4-24-2023

Effects of Cyanobacteria Harmful Algal Blooms on the Microbial Community Within Lake Okeechobee, FL, USA

Paisley S. Samuel
Nova Southeastern University

Thesis of
Paisley S. Samuel

Submitted in Partial Fulfillment of the Requirements for the Degree of

Master of Science
Marine Science

Nova Southeastern University
Halmos College of Arts and Sciences

April 2023

Approved:
Thesis Committee

Committee Chair: Jose Lopez, Ph.D.

Committee Member: Lauren Krausfeldt, Ph.D.

Committee Member: Matthew Johnston, Ph.D.

NOVA SOUTHEASTERN UNIVERSITY

HALMOS COLLEGE OF ARTS AND SCIENCES

Effects of Cyanobacteria Harmful Algal Blooms on the Microbial Community within Lake Okeechobee, FL, USA

By:

Paisley S. Samuel

Submitted to the Faculty of
Halmos College of Arts and Sciences
in partial fulfillment of the requirements for the degree of
Master of Science with a specialty in:

Marine Science

Nova Southeastern University

April 2023

ABSTRACT

The Lake Okeechobee (Lake O) watershed is a Floridian freshwater ecosystem that has been affected by the increased frequency and intensity of harmful cyanobacterial bloom (cyanoHAB) events occurring over recent decades. Lake O has several ecological and economic purposes such as providing habitats for various organisms and providing drinking water to urban communities surrounding the lake. Toxic cyanoHAB events are posing a threat to the ecosystem and economy of the lake due to the degradation of water quality. This study investigates how the microbial community structure within Lake O is affected by annual cyanobacterial harmful algal blooms over several years by assessing the dominant taxa, temporal patterns, and spatial patterns within the microbial communities and determining if cyanoHABs alter the microbial diversity in Lake O. Filtered surface water samples and public environmental data were collected from 21 routinely monitored sites within and connecting to Lake O from March 2019 to October 2021. DNA extraction, purification, and polymerase chain reactions on the V4 region of the 16S rRNA gene were used to create amplicon libraries for high-throughput sequencing on 541 samples, generating an average of over 40,000 reads per sample. After characterizing the dominant taxa within Lake O, the top four phyla include Proteobacteria, Bacteroidota, Cyanobacteria, and Actinobacteriota, which remained consistent across the sampling period. Microbial alpha diversity exhibited both spatial and temporal changes from year-to-year. The significant spatial differences observed across all three years suggested that there are stable biogeographical patterns within Lake O. Different environmental variables across the sampling period were found to drive beta diversity of the microbial communities in Lake O, with TN:TP ratio, turbidity, ammonia, total phosphate, nitrate + nitrite, dissolved oxygen, and pH remaining consistent in all years. *Microcystis* relative abundance was found to influence the alpha and beta diversity of the microbial communities, decreasing alpha diversity, and decreasing correlating beta diversity as well. *Microcystis* relative abundance also correlated with several environmental factors including temperature, total depth, and nitrate + nitrite concentrations. After observing such strong correlations to *Microcystis*, a co-occurrence network was created and has suggested that specific taxa may influence mutualistic or antagonistic relationships with *Microcystis*.

Keywords: Lake Okeechobee, *Microcystis*, cyanoHABs, microbial community, cyanobacteria, blooms, freshwater ecosystems, high-throughput sequencing

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

INTRODUCTION

Cyanobacteria and Harmful algal blooms

Cyanobacteria are photoautotrophic, gram-negative, prokaryotic bacteria that can be found within numerous environments all over the world, including some extreme environments (Gaysina *et al.*, 2019; Mataloni and Komárek, 2004; Whitton and Potts, 2000a, b). Cyanobacteria contain chlorophyll a, a pigment that allows them to perform photosynthesis and produce oxygen as a product. It was due to this ability to photosynthesize that allowed cyanobacteria to spark the oxidation of Earth's atmosphere around 3 billion years ago (Huisman *et al.*, 2018). Cyanobacteria are often referred to as blue-green algae; however, they are not algae but true bacteria and were initially confused with being algae since they possessed the photosynthetic abilities and pigments like eukaryotic algae. In addition, cyanobacteria are not always blue green in color, as there are other species of cyanobacteria that exhibit various other colors such as numerous shades of green, red, and brown (Huisman *et al.*, 2018; Stomp *et al.*, 2007).

Cyanobacteria are able to rapidly proliferate to form dense accumulations of biomass known as blooms (Larkin & Adams, 2007). Some of these cyanobacteria blooms can either be harmless or harmful to their surrounding environment. Cyanobacteria are primarily responsible for causing harmful blooms (cyanoHABs) in freshwater environments (Rosen *et al.*, 2017). These cyanoHABs can result from water quality changes, which is primarily due to changes in nutrient levels. During photosynthesis, cyanobacteria utilize nutrients, such as carbon, potassium, iron, etc., along with solar energy to aid in their cell growth. However, nutrients must be present in a certain amount to promote cyanobacteria populations to bloom, if there is a deficiency in any of the nutrients then a bloom cannot occur (Markou *et al.*, 2014). The nutrient level changes associated with degraded water quality are primarily attributed to the increase in nitrogen (N) and phosphorus (P) levels in the environment. Levels of N and P in freshwater ecosystems often serve as limiting nutrients and, when low, allow for good water quality and higher microbial diversity within the ecosystem (Facey, Apte, & Mitrovic, 2019). When there are high levels of N and P due to agricultural fertilizer runoff, these populations can bloom and create very dense mats on the surface. There are many other factors that produce favorable conditions for and exacerbate cyanobacterial blooms, including stagnant water and high temperatures (Paerl & Huisman, 2008).

CyanoHABs can further decrease water quality by producing cyanotoxins, water-soluble chemical metabolites that are toxic to the environment. Cyanotoxins are grouped into four groups: hepatotoxins, which attack the liver (microcystins and cylindrospermopsin); neurotoxins, which attack the nervous system (anatoxins and saxitoxins); dermatotoxins, which attack the skin (lyngbyatoxins and aplysiatoxin); and irritant toxins, which attack both skin and organs if contact is made (Wiegand & Pflugmacher, 2005; Williams *et al.*, 2007; Bláha, Babica, & Maršálek, 2009). As these toxins reach high enough concentrations in these freshwater ecosystems, they can threaten the health of the organisms in and around those ecosystems and the ecosystem itself. For example, there have been a number of incidents where cyanotoxins from the cyanoHABs caused animal and human poisonings (Bláha, Babica, & Maršálek, 2009). These impacts are derived from the structure of these blooms. Both harmless and harmful blooms create thick, dense mats at the surface of the water. These mats prevent sunlight from penetrating into the water column, decreasing the light needed for photosynthetic organisms residing deeper in the water column. Additionally, when these blooms begin to decay, they create an anoxic environment as large amounts of dissolved oxygen are used up thus reducing the amount of dissolved oxygen that other organisms in the lake need to survive and causing many organisms to die (Anderson, 2009). These negative impacts caused by cyanoHABs can have severe impacts on ecosystem functioning, such as changes in biodiversity, bioaccumulation of cyanotoxins within organisms, and food web disturbances (Zamora-Barrios *et al.*, 2019; McQuaid, 2019; Bláha, Babica, & Maršálek, 2009). Despite immense research on cyanobacterial blooms and the factors that drive them, they remain difficult to predict and mitigate, and there is much more to be studied on the triggers of cyanoHABs (Facey, Apte, & Mitrovic, 2019; Bowling, 1994).

CyanoHABs in Lake Okeechobee, Florida

CyanoHABs occur within many Floridian freshwater ecosystems, including Floridian lakes, rivers, streams, and canals. Toxin-producing cyanoHABs have been recorded in Florida's freshwater systems and the adverse effects of these cyanoHABs appear to have increased over the decades (Myer *et al.*, 2020). Lake Okeechobee is one freshwater ecosystem experiencing these increasing numbers of toxic cyanoHABs events.

Also known as "Florida's Inland Sea," Lake Okeechobee is the largest lake in the southeastern United States and is located at the center of Florida's Everglades ecosystem (Lecher,

2021). Lake Okeechobee was once larger and deeper flowing north to south and provided a constant water source to the Everglades ecosystem. However, beginning in the late 19th century, the size, depth, and direction of flow of the lake were permanently altered as a series of major drainage projects transformed the land around the lake to become a foundation for urban communities and agriculture (Lecher, 2021). These major drainage projects included the channelization of the Kissimmee River and the dredging of numerous canals (Lecher, 2021). The last major drainage project of Lake Okeechobee that is still managed today was the construction of the Herbert Hoover Dike in the 1930s to 1940s (U.S. Army Corps of Engineers, 2021). After the destruction and deaths caused by the storm surges and flooding from the 1920s hurricanes, the federal government passed the "Rivers and Harbors act of 1930" which demanded the construction of the 31-feet (9.4m) tall Hoover Dike to aid in the water flow management of Lake Okeechobee and further serve as flood protection for the communities residing around the lake (Lecher, 2021). Consequently, these water management projects greatly impacted the ecosystem and the water quality of the lake. Throughout the 1950s and 1960s, the water quality of Lake Okeechobee began to decline rapidly as the nutrient levels continually increased, primarily phosphorus levels, from agricultural land use (Canfield & Hoyer, 1988), thus further increasing the nutrient input of an already eutrophic environment that was initially limited in nitrogen rather than phosphorus (Missimer *et al.*, 2021).

As a result of the nutrient pollution and degrading water quality, cyanoHABs are a common occurrence in Lake Okeechobee, and in recent decades, these bloom events have increased in both abundance and prevalence (Rosen *et al.*, 2017). The freshwater toxic cyanoHABs that occur in Florida are primarily caused by the genus *Microcystis*, but blooms caused by the genera *Dolichospermum*, and *Cylindrospermopsis* also occur. The toxins produced during blooms caused by these genera include microcystins, which are produced by *Microcystis*, some *Dolichospermum* species, and some *Cylindrospermopsis* species; anatoxin-a, which is produced by *Dolichospermum* and some *Cylindrospermopsis* species; saxitoxins, which is produced by *Cylindrospermopsis*; and cylindrospermopsin, which is produced by *Cylindrospermopsis* (Myer *et al.*, 2020). In 2016, after a long period of rain and warm, sunny weather, massive toxic cyanoHABs formed in Lake Okeechobee, St. Lucie River, and Caloosahatchee River. Metcalf *et al.* (2018) documented that the dominant blooming species was *Microcystis aeruginosa*. In fact, *Microcystis aeruginosa* is one of the most common bloom-forming and microcystin-producing cyanobacterium in the lake and is

also found in freshwater ecosystems around the world (Harke, *et al.*, 2016). For decades, there have been annual cyanoHAB events within the lake and neighboring rivers/canals, and it can only be assumed that these cyanoHAB events will further increase due to anthropogenic eutrophication and climate change (Huisman *et al.*, 2018; Van Wichelen *et al.*, 2016; Okello *et al.*, 2010).

Heterotrophic bacteria and cyanoHABs

Traditionally, cyanoHABs are considered to be predominantly driven by abiotic factors (Rollwagen-Bollens *et al.*, 2018; Visser *et al.*, 2016; Paerl & Scott, 2010). However, Shen *et al.* (2011) documented that some heterotrophic bacterioplankton can coexist with these bloom-forming cyanobacteria, which has led to speculation that the microbial community may also play a role during these cyanoHAB events (Wang *et al.*, 2021; Van Wichelen *et al.*, 2016). The interactions between photoautotrophic bacteria, which use sunlight and carbon dioxide, and heterotrophic bacteria, which consume organic material to obtain energy, play fundamental roles in aquatic ecosystems. As described by Zheng *et al.* (2018), heterotrophs utilize fixed carbon and other nutrients supplied by photoautotrophs and, in turn, provide these photoautotrophs with essential vitamins and amino acids. *Synechococcus* (Zheng *et al.*, 2018) and *Microcystis* (Van Wichelen *et al.*, 2016; Tu *et al.*, 2019) colonies frequently contain heterotrophic bacteria, and the colonies obtained from nature contain heterotrophic bacteria communities as well.

Certainly, there must be a diverse microbial community within Lake Okeechobee although there have not been any studies done to characterize this diverse community until recently (Krausfeldt *et al.*, *submitted*). This microbial diversity could allow for the interaction of the bloom-forming cyanobacteria before, during, and after cyanoHAB events within Lake Okeechobee. Some studies have been done to investigate what roles the microbial community may play in the overall development and maintenance of these cyanoHABs, suggesting that these microbes who thrive alongside the bloom-forming cyanobacteria may have an important impact on the cyanobacterial growth and populations (Eiler & Bertilsson, 2004; Sigee, 2005). Microbes can also aid in the degradation of the organic material produced by the bloom, which contributes to the anoxic conditions that follow bloom degradation (Anderson, 2009; Havens, 2007).

When a cyanoHAB event occurs, there is essentially a proliferation of one species of bacteria that continues to multiply within the lake. As this cyanobacterial species continues to grow in abundance, the other bacterial species may become outnumbered or driven out of the area due

to competition of resources with the blooming cyanobacteria. The movement of bacteria out of the area would decrease the diversity of that area of the lake since there are now fewer species inhabiting that area of the lake. Ultimately, the local communities scattered across the lake show less diversity between them, thus exhibiting a decrease in microbial diversity throughout the lake. So, understanding the interactions between the microbial community and these bloom-forming cyanobacteria and how microbial diversity changes during cyanoHABs may provide scientists the knowledge of key factors driving or sustaining blooms, serve as a biological indicator, and may aid efforts to reduce or mitigate the occurrences of these blooms.

High throughput sequencing of the 16S rRNA gene

High-throughput sequencing (HTS) is used to comprehensively study microbial communities. HTS is the second generation of sequencing technology and has been the most used method of sequencing for over half a century (Zhu *et al.*, 2014). The methods used within HTS have been modeled after the first generation of sequencing technology, Sanger sequencing, developed in 1977 by Frederick Sanger and his colleagues (Sanger *et al.*, 1977). However, it was not until the development of HTS techniques that scientists began to understand various biological systems and the impacts of various conditions on organism microbiomes.

As described by Byrne *et al.* (2018), the 16S rRNA gene encodes small subunit ribosomal RNA molecules of ribosomes, responsible for converting genetic code into functional cell components within an organism. Discovered by the works of Dubnau *et al.* in the 1960s and Woese and Fox in 1977, the 16S rRNA gene sequence in bacteria contains multiple conserved and highly variable regions (Dubnau *et al.*, 1965; Woese & Fox, 1977). There are a total of nine variable regions found within the 16S rRNA gene (V1-V9), and they are widely used in the identification, classification, and phylogenetic analysis of various bacteria. Various studies have found that the V2 and V4 regions of the gene are best used for classification due to their low error rates. Additionally, the V3 region of the gene can identify the genus of pathogenic bacteria better than the V2 region. To properly detect these variable regions, various universal primers were created, and polymerase chain reactions (PCR) were used to amplify these regions (including the primers) to aid in identifying specific species of bacterium.

Woese & Fox (1977) were the pioneers of using the 16S rRNA gene to aid in the phylogenetic analyses of bacterial and archaeal species. Within the past decade, these regions of

the 16S gene have also been used in large-scale genomic projects, including the human microbiome project (conducted to understand the human-body microbiome) and the Earth Microbiome Project (conducted to understand the microbiomes of the organisms that inhabit this planet). In the Microbiology and Genetics Laboratory at Nova Southeastern University's Halmos College of Arts and Sciences (NSU HCAS), HTS is commonly used to analyze various microbiomes (Campbell, Fleisher, Sinigalliano, White, & Lopez, 2015; Donnelly, 2018; Easson & Lopez, 2019; Freed, 2018; Karns, 2017; O'Connell, Gao, McCorquodale, Fleisher, & Lopez, 2018).

AIMS AND HYPOTHESES

The primary objective of this study was to investigate how the structure of microbial communities within Lake Okeechobee is affected by annual cyanoHABs over several years. To address this, the alpha and beta diversity of the microbial community were examined using statistical analyses (as described in the methodology section below). The temporal and spatial trends were assessed in the microbial community of Lake Okeechobee by comparing the alpha and beta diversity values of the microbial communities across the years, months, seasons, stations, and ecological zones.

This study was broken down further to address several aims and hypotheses:

Aim 1. Compare the dominant taxa and species diversity (alpha and beta diversity) of the microbial communities in Lake Okeechobee across three years.

H_1: The dominant taxa and microbial diversity of Lake Okeechobee will remain the same across three years.

Aim 2. Explore the spatial differences in alpha and beta diversity of the microbial communities within Lake Okeechobee across three years.

H_2: Spatial differences will be observed in the alpha and beta diversity of each year based on ecological zones and stations.

Aim 3. Determine if cyanoHABs alter microbial diversity in Lake Okeechobee.

H_3: CyanoHABs will decrease the alpha and beta diversity of the microbial community within Lake Okeechobee.

METHODOLOGY

Sample and environmental data collection

Beginning in March of 2019, surface water samples were collected monthly by the South Florida Water Management District (SFWMD) at 21 routinely sampled stations. These stations included 19 stations dispersed within Lake Okeechobee, one station located near the W.P. Franklin Lock along the Caloosahatchee River (S79), and another station located near the St. Lucie River lock (Figure 1). After collection, the water samples were kept on ice and shipped overnight to the USGS Water Science Center in Orlando, Florida, where each sample was filtered through two 0.22µm Sterivex filters (Millipore, SVGP01050), stored at -20°C, then transported on ice to the Microbiology and Genomics Lab at Nova Southeastern University (NSU) for further sample processing. This workflow of sample collection and processing was repeated until October of 2021.

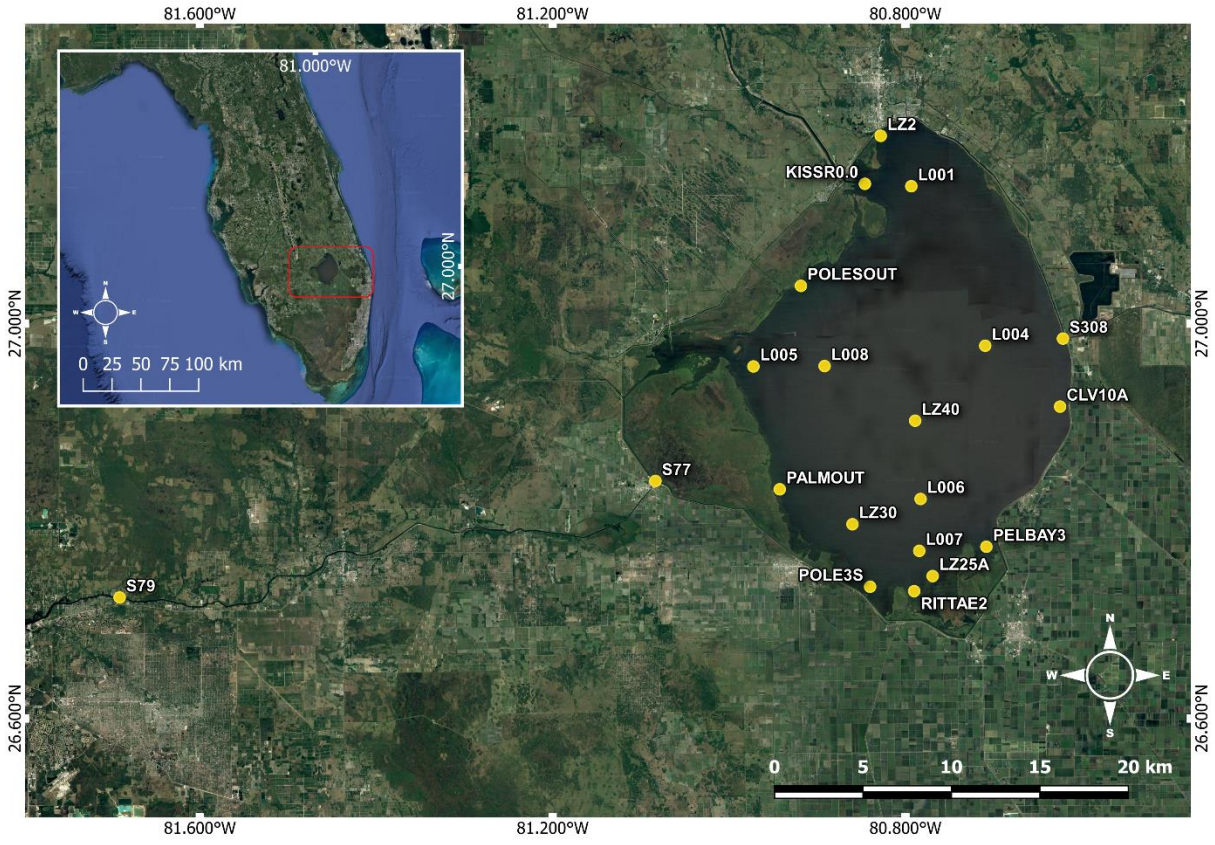

Figure 2. Map of sampling stations found within and connected to Lake Okeechobee. 19 stations are located within the lake while one is located within the Caloosahatchee River (S79).

Environmental data was collected from SFWMD's environmental database, DBHYDRO, that contains hydrologic, meteorologic, hydrogeologic, and water quality data (http://my.sfwmd.gov/dbhydroplsql/show_dbkey_info.main_menu). Environmental variables that were collected include: chlorophyll a (chl a, µg/L), pheophytin a (µg/L), secchi disk depth (m), silica (mg/L), turbidity (NTU), sulfate (mg/L), alkalinity (as total $CaCO_3$, mg/L), ammonia (NH_4, mg/L), total depth (m), pH, dissolved oxygen (mg/L), nitrate+nitrite (NO_3+NO_2, mg/L), total phosphate (PO_4, mg/L), temperature (temp, °Celsius), total nitrogen (TN, mg/L), total phosphorus (TP, mg/L), TN and TP ratio, and three toxins associated with cyanoHABs, Anatoxin-a (µg/L), Cylindrospermopsin (µg/L), and Microcystin (µg/L). Additional variables were also considered for each sample, including month (1-12), season (wet or dry), year (1-3), station (CLV10A, KISR0.0, L001, L004, L005, L006, L007, L008, LZ2, LZ25A, LZ30, LZ40, PALMOUT, PELBAY3, POLE3S, POLESOUT, RITTAE2, S308, S77, and S79), and ecological zone (inflow, nearshore, pelagic, or S79). To note, the wet and dry seasons of Florida were defined by NOAA, with the wet season occurring from May to October and the dry season occurring from November to April (U.S. Department of Commerce, n.d.). After retrieval, the environmental data was then corresponded to the collected samples for DNA extraction and sequencing.

Sample Processing

Once the collected samples were received at NSU, the sterivex filters were cut from their plastic tubing and DNA was extracted from the filters using the Qiagen® DNeasy® PowerLyzer® PowerSoil® kit (Qiagen, 12855-100) by following the manufacturer's protocol. Negative controls in the form of blank 'reagent-only' extractions were also included to detect any DNA contamination within the reagents. Following successful DNA extractions, an 1.5% agarose gel underwent an agarose gel electrophoresis protocol to confirm the presence of intact DNA in each sample.

Following the confirmation of intact DNA, a test polymerase chain reaction (PCR) was performed on each sample to confirm the successful amplification of PCR products. In short, a master mix was made using Invitrogen Platinum Hot Start PCR Master Mix (2X; ThermoFisher, 13000014), nuclease-free water, and universal primers 515F and 806R. DNA was then added and underwent amplification in a thermal cycler following the Earth Microbiome Project (EMP) 16S Illumina Amplicon protocol (Caporaso, 2018). 515F and 806R primers are used to target and

amplify the V4 region of the 16S rRNA gene. A 1.5% agarose gel electrophoresis was also done to confirm the production of successful PCR products. To note, if the test PCR was unsuccessful—evidence that the concentration of extracted DNA was low—the sample was concentrated using a CentriVap DNA Vacuum Concentrator (©Labconco, Cat. No. 7970010), ran through another test PCR, and ran again on a 1.5% agarose gel to verify successful amplification. With the successful production of PCR products, barcoded 515F and 806R primers were then used, with each sample receiving identical barcoded 515F primer sequences and unique barcoded 806R primer sequences. A final 1.5% agarose gel was run to confirm the successful barcoding of the samples. Afterwards, the samples are cleaned using a modified AMPure XP beads protocol (PCR purification with Beckman Coulter AMPure XP magnetic beads and the VIAFLO 96, 2020), quantified using Qubit 3.0 and Qubit 4.0 Fluorometers (Life Technologies), and diluted to 4nM using nuclease-free water. The now-diluted barcoded samples were then pooled together and checked for quality and contamination using the Agilent TapeStation 4150 (Product #G2992AA). The final library pool was then loaded into the Illumina MiSeq system (Product #SY-410-1003) using the MiSeq Reagent Kit v3 at 600 cycles (Product #MS-102-3003) following a modified protocol.

Sequence analysis

The raw sequence data generated from the Illumina MiSeq system was transferred to a hard drive and initial bioinformatic analysis began within a command-line program known as QIIME2. QIIME2 (Quantitative Insights into Microbial Ecology, version 2022.2) is a next-generation, open-source bioinformatics pipeline used for performing microbiome analysis from raw DNA sequence data (Bolyen *et al.*, 2019). Within the QIIME2 environment, the forward and reverse read sequence data (in the form of FASTQ files) were paired and demultiplexed to produce the sequence reads for each sample. The sample sequences were then trimmed, checked for chimeras, and quality filtered (Q-scores > 29) using the DADA2 software package built into the QIIME2 prorgam. There was a total of 11 sequencing runs included within this study, thus the raw sequence data for each run underwent demultiplexing, trimming, and quality filtering before being merged as one dataset. Lastly, the merged sequencing data set was assigned taxonomy using the SILVA 138 classifier (silva-138-99-515-806-nb-classifier.qza). The resulting dataset was then cleaned to ensure it did not contain any unwanted ASVs. A rarefaction curve was created to determine the sequence read cut-off point for any samples that were not fully sequenced. Any ASVs that were found in the

negative controls were removed and the negative control samples were also removed from the sample pool. Any duplicate samples were removed by choosing the sample that obtained the most sequence reads and removing the other replicates. To ensure that the dataset contained no eukaryotes, ASVs that represented chloroplast or mitochondrial DNA were also removed. A final cleaning and normalization were performed using the 'vegan' package using the statistical computing language, R, in the RStudio software (version 4.2.0) where singletons, doubletons, and ASVs occurring less than 0.01% were removed.

Batch Correction

Due to the large-scale nature of this study, the hundreds of samples that were sequenced could be affected by differences in sample preparation and data acquisition conditions, for example, different individuals working on the sample preparation, different reagent batches, or even changes in instrumentation (Cuklina, *et al.,* 2021). This is known as the "batch effect" and can introduce noise that would in turn reduce the statistical power of the analyses (Cuklina, *et al.*, 2021). Taking this into consideration, the data was tested for any significant batch effects before moving on to further downstream analyses. The test was performed using the 'MMUPHin' and 'vegan' packages in R. An ANOSIM was performed to determine if the variation in the data caused by batch were significant ($p < 0.05$). If significant differences caused by batch were found in the data, the package 'MMUPHin' was used to conduct a batch correction.

Taxonomy analyses and visualization using QGIS

Taxonomic and statistical analyses were performed on the cleaned, normalized, batch corrected dataset using R. The 'phyloseq' package was used to determine the minimum, maximum, and average sequence read amounts, total number of unique ASVs, and number of unique phyla found in the data set. Top 10 taxa were calculated using packages 'phyloseq' and 'microbiome' and visualized using bar plots made using 'ggplot2' package for each year and station. QGIS, an analytical mapping software, was used to visualize the microbial community taxonomic distributions and patterns within Lake Okeechobee across the entire sampling period and within each year. An aerial satellite image of Lake Okeechobee was retrieved from Google Earth via the QGIS software and utilized as the raster layer. Point layers were created using the latitude and longitude coordinates retrieved from DBHYDRO for each station. Pie charts of the top 10 phyla found within each station were created for both the entire sampling period and within each year.

Diversity analyses

Alpha diversity, which describes the number of different species and how evenly distributed they are within a particular community (Thukral, 2017), was assessed using the 'vegan' package and visualized using the 'base' and 'ggplot2' packages. Alpha diversity was measured by calculating the total number of species (species richness), species evenness (also known as Pielou's evenness index) (J), Shannon diversity index (H), and inverse Simpson's diversity index (inv. D). Shannon and inverse Simpson diversity indices take into consideration species richness and evenness when examining alpha diversity. Shannon diversity index assumes all species are represented and sampled randomly but can be less effective against rare species. The inverse Simpson index removes bias by pooling the total diversity so that the average of the pooled communities is greater than or equal to the diversity within communities (Lande, 1996). Differences between these alpha diversity indices were analyzed between samples. If the data was normally distributed, then an analysis of variance (ANOVA) was used, otherwise a Kruskal-Wallis test was to be used. If there were significant differences found, a pairwise Wilcoxon test (for Kruskal-Wallis analyses) or Tukey test (for ANOVA analyses) was used as a post-hoc test to determine where the differences lie.

Beta diversity, which describes the differences between communities (Thukral, 2017), was assessed using the 'vegan' package and visualized using the 'base' and 'ggplot2' packages as well. Beta diversity was measured by calculating Bray-Curtis dissimilarity between sites. These distance matrices were then used to produce non-metric multidimensional scaling (nMDS) plots in R to further visualize the distances between sites. To create the nMDS plots, the relative abudance data was transformed using the "total" method found within the 'decostand' function in 'vegan'. Functions 'betadisper' and 'permutest' in 'vegan', were used to calculate variances within each group and to determine if the variances differ by group. If the variances between groups were not significant, a permutational multivariate ANOVA (PERMANOVA) with 999 permutations was performed. If the variances between groups were significant, an analysis of similarity (ANOSIM) with 999 permutations was performed. Canonical correspondence analysis (CCA) was also performed using the 'cca' function in 'vegan' to detect the interactions between the selected environmental variables and ASVs. The function 'envfit' was then used to get the p-value of

correlation of each variable with overall bacterial communities and the p-value of each correlation between each ASV and all variables. Only significant ($p<0.05$) environmental variables with R^2 values higher than 0.3 were plotted as vectors overlaying the CCA plot.

Venn diagram and co-occurrence network

Using the 'eulerr' package in R, a venn diagram was made to compare core taxa that appeared across the years (1, 2, and 3). Core taxa included any ASVs that was detected in a relative abundance of at least 0.1% and in at least 75% of the samples. Afterwards, a co-occurrence network was created to further investigate what taxa could be co-occurring with the genus *Microcystis*. This was done using the package 'Hmisc' in R and Cytoscape (version 3.9.1) (Shannon, et al., 2003), a software used to create interactive networks. In R, a Pearson correlation matrix was created using the sample count data and making pairs of all 8,340 ASVs from the entire sampling period. The correlation matrix was then converted into a table format so that the individual R^2 values and their associated p-values could be extracted between each interaction pair that was created. Only the significant interactions ($p<0.05$) and the strongest correlations ($R^2 > 0.7$ OR $R^2 < -0.7$) were extracted from the table. This resulting table was then imported into Cytoscape (version 3.9.1) as a network, where it was filtered further to only include the network nodes and edges that interact with *Microcystis*.

RESULTS

Sequencing statistics

Across the sampling period (March 2019 to October 2021), there were a total of 59,862,979 sequencing reads and 70,605 ASVs generated across all samples in this study. To determine the sequencing depth, or the total number of usable reads, that best represented the microbial communities of Lake O, total sequence reads were calculated for each sample and a rarefaction curve was generated to aid in determining the minimum sequence read cut-off point. The resulting rarefaction curve reached an inflection point at relatively 10,000 reads, thus, any samples that were below this amount were removed (Figure 2). As a result, 65,294 ASVs and 541 samples, with an average of 44,535 reads per sample, were used for further analysis (Table S1). Additional filtering for singletons, doubletons, and exceptionally low abundance ASVs (occurring less than 0.01%) was completed, resulting in 8,340 ASVs being utilized for further diversity analyses.

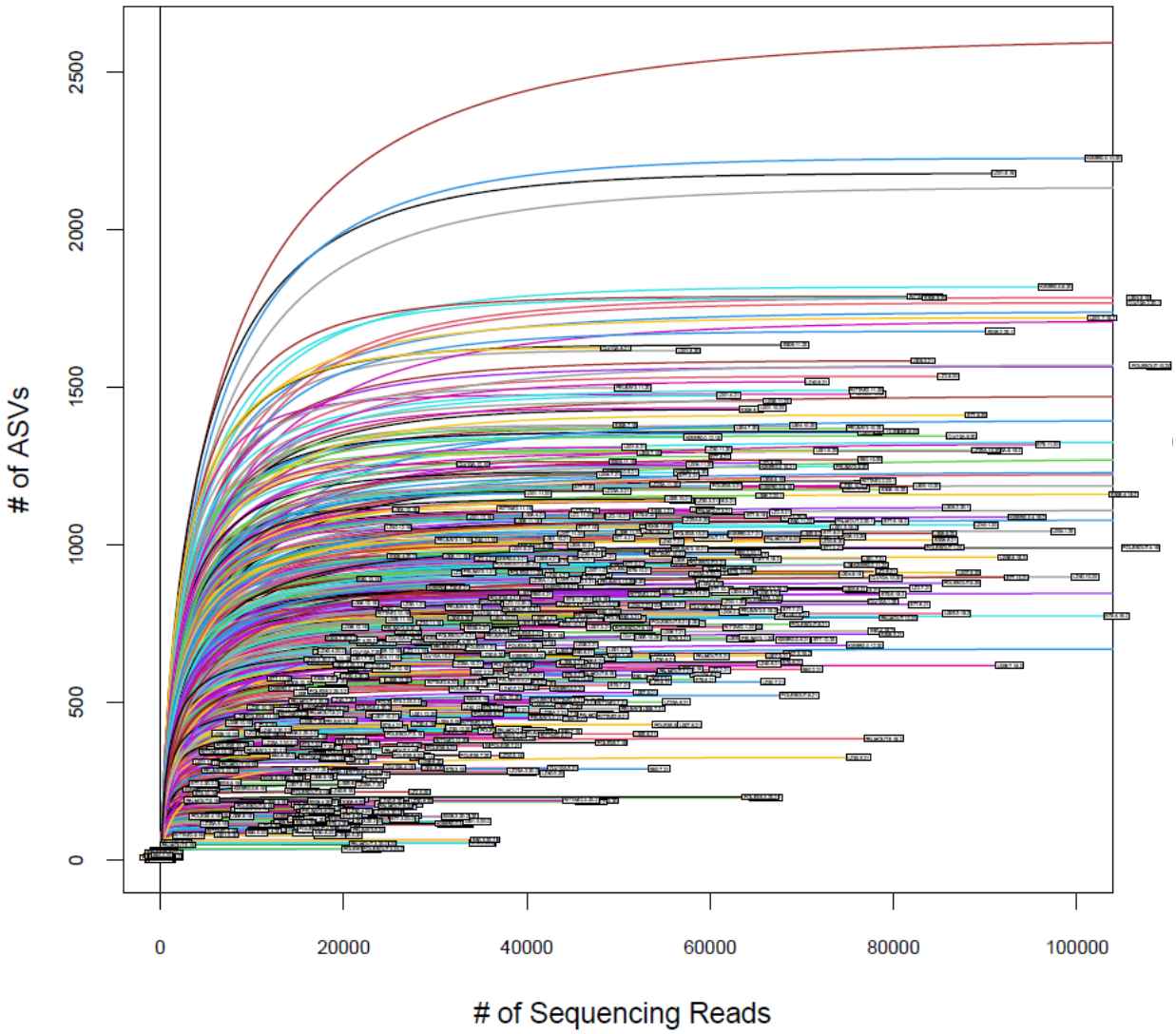

Figure 2. Rarefaction curve for number of sequencing reads versus number of ASVs to determine final samples for analysis. Each line represents one sample. Inflection point occurred at roughly 10,000 reads.

Dominant Phyla and Species diversity

The top ten phyla found in Lake O over the entire sampling period were Proteobacteria (24.7%), Bacteroidota (22.1%), Cyanobacteria (16.8%), Actinobacteriota (11.3%), Verrucomicrobiota (7.9%), Planctomycetota (6.8%), Bdellovibrionota (3.2%), Acidobacteriota (3.0%), Chloroflexi (2.2%), and Gemmatimonadota (1.9%) (Figure 3). The top ten phyla within each year varied within their makeups, with year 3 being the only year containing phylum Gemmatimonadota (Table 1, Figure 3). These phyla can also be seen within each station with Proteobacteria, Bacteroidota, and Cyanobacteria being the top three phyla found in each station (Figure 4). Additionally, when considering individual stations, the top 10 phyla also differed—both within all years overall (Figure S1) and between each year (Figures S2-S4).

Year 1 was the only year that included the phylum SAR324_ clade (marine group B) within the top 10 phyla of only 2 stations, POLESOUT and S79 (Figure 5, Figure S2). Year 2 had 13 unique phyla appear within the top 10 phyla of each station—one phylum short of years 1 and 3, both of which had 14 unique phyla each in their top 10 phyla across each station. Furthermore, year 2 was the only year that included the phylum Armatimonadota within the top 10 phyla occurring at only one station, KISSR0.0 (Figure 6, Figure S3). Year 2 also was the only year that did not have the phylum Myxococcota within the top 10 phyla of any station. Year 3 was the only year that included the phylum Patescibacteria within the top 10 phyla of only 2 stations, L004 and L006 (Figure 7, Figure S4).

Table 1. Average proportion and standard deviation of the relative abundances of the top 10 phyla in Lake Okeechobee by year.

Phylum	Year 1 (2019)	Year 2 (2020)	Year 3 (2021)
Proteobacteria	0.236 ± 0.057	0.215 ± 0.073	0.226 ± 0.055
Bacteroidota	0.217 ± 0.082	0.200 ± 0.071	0.196 ± 0.079
Cyanobacteria	0.119 ± 0.096	0.169 ± 0.102	0.159 ± 0.098
Actinobacteriota	0.105 ± 0.055	0.115 ± 0.041	0.099 ± 0.042
Planctomycetota	0.071 ± 0.025	0.060 ± 0.026	0.063 ± 0.023
Verrucomicrobiota	0.069 ± 0.031	0.068 ± 0.032	0.075 ± 0.031
Bdellovibrionota	0.033 ± 0.018	0.022 ± 0.014	0.027 ± 0.014
Acidobacteriota	0.029 ± 0.020	0.027 ± 0.018	0.029 ± 0.019
Chloroflexi	0.021 ± 0.009	0.021 ± 0.009	0.021 ± 0.008
Crenarchaeota	0.018 ± 0.028	0.018 ± 0.025	–
Gemmatimonadota	–	–	0.019 ± 0.011

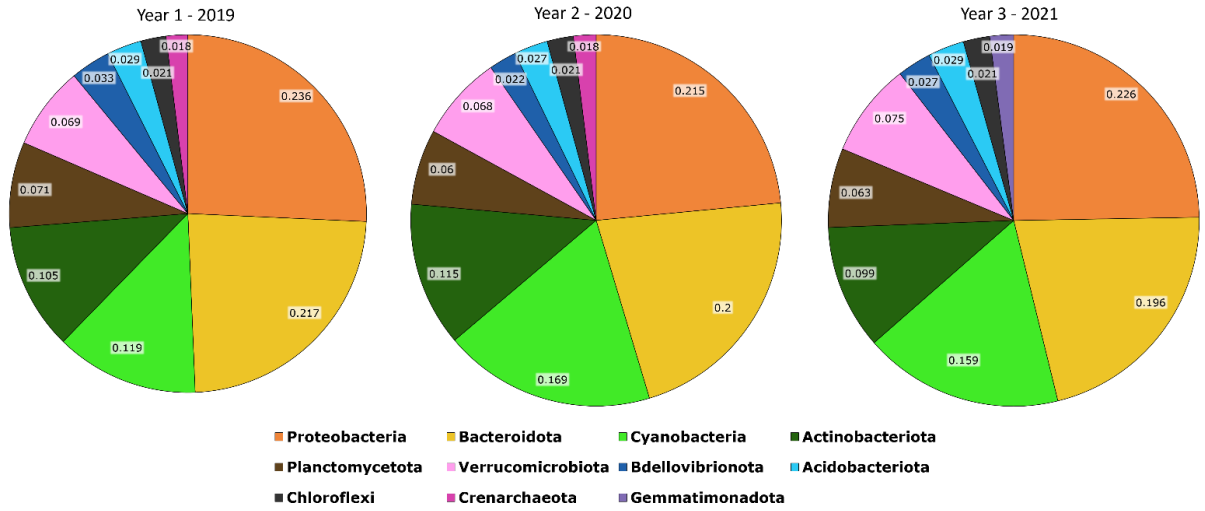

Figure 3. Pie charts depicting the proportions of the top 10 phyla within each year. The numbers indicate the total relative abundance of the respective year.

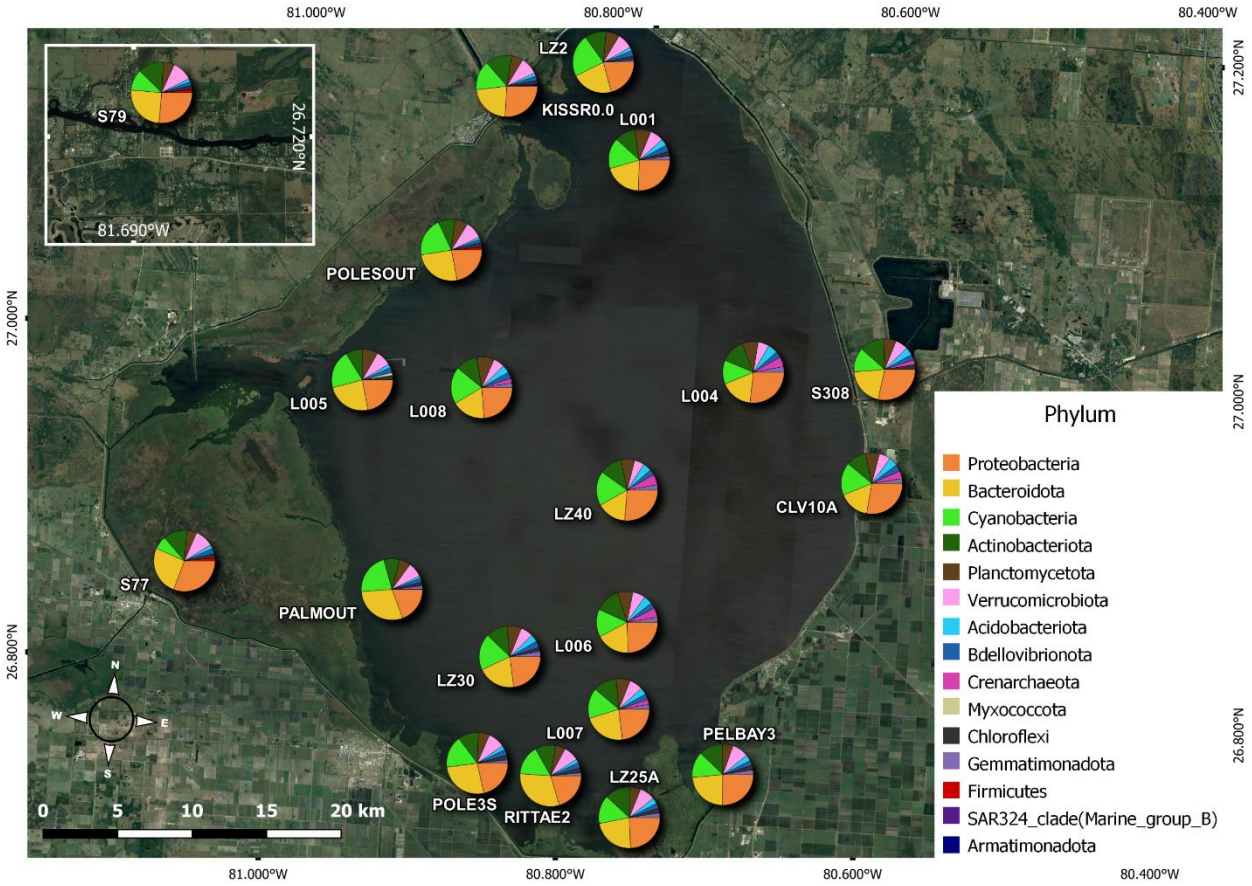

Figure 4. Pie charts showing the top phyla found in each station in Lake O over the sampling period.

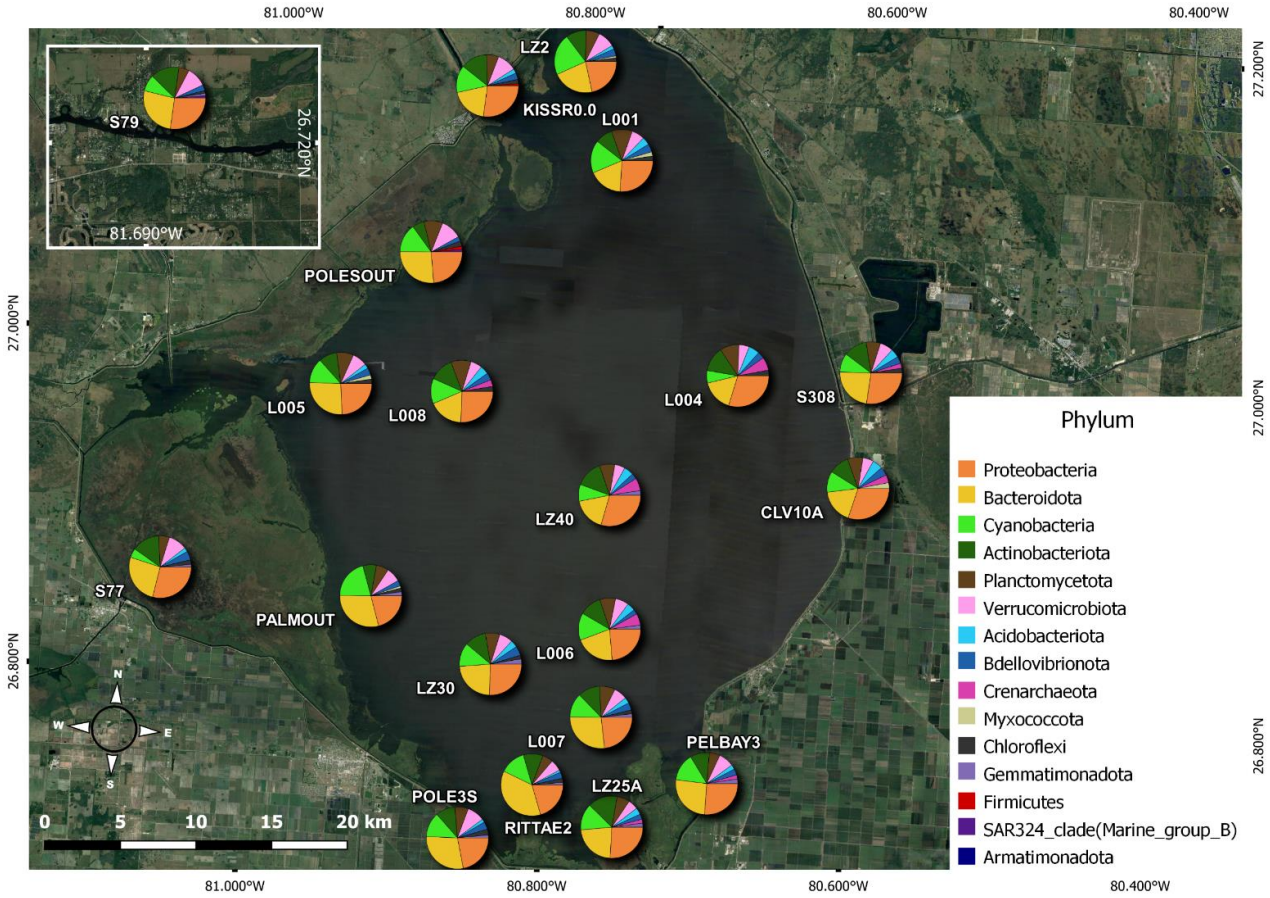

Figure 5. Pie charts showing the top phyla found in each station in Lake O within year 1 (2019).

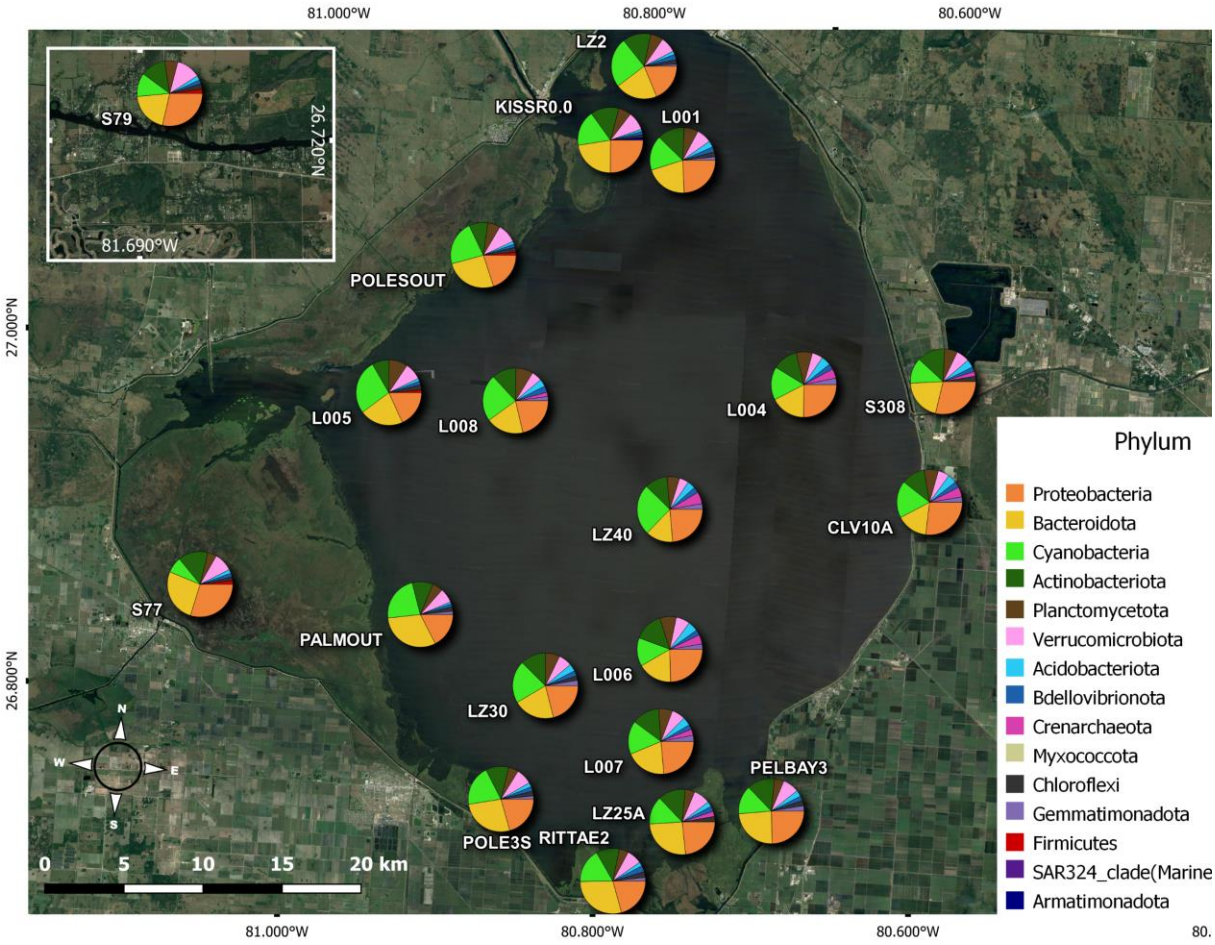

Figure 6. Pie charts showing the top phyla found in each station in Lake O within year 2 (2020).

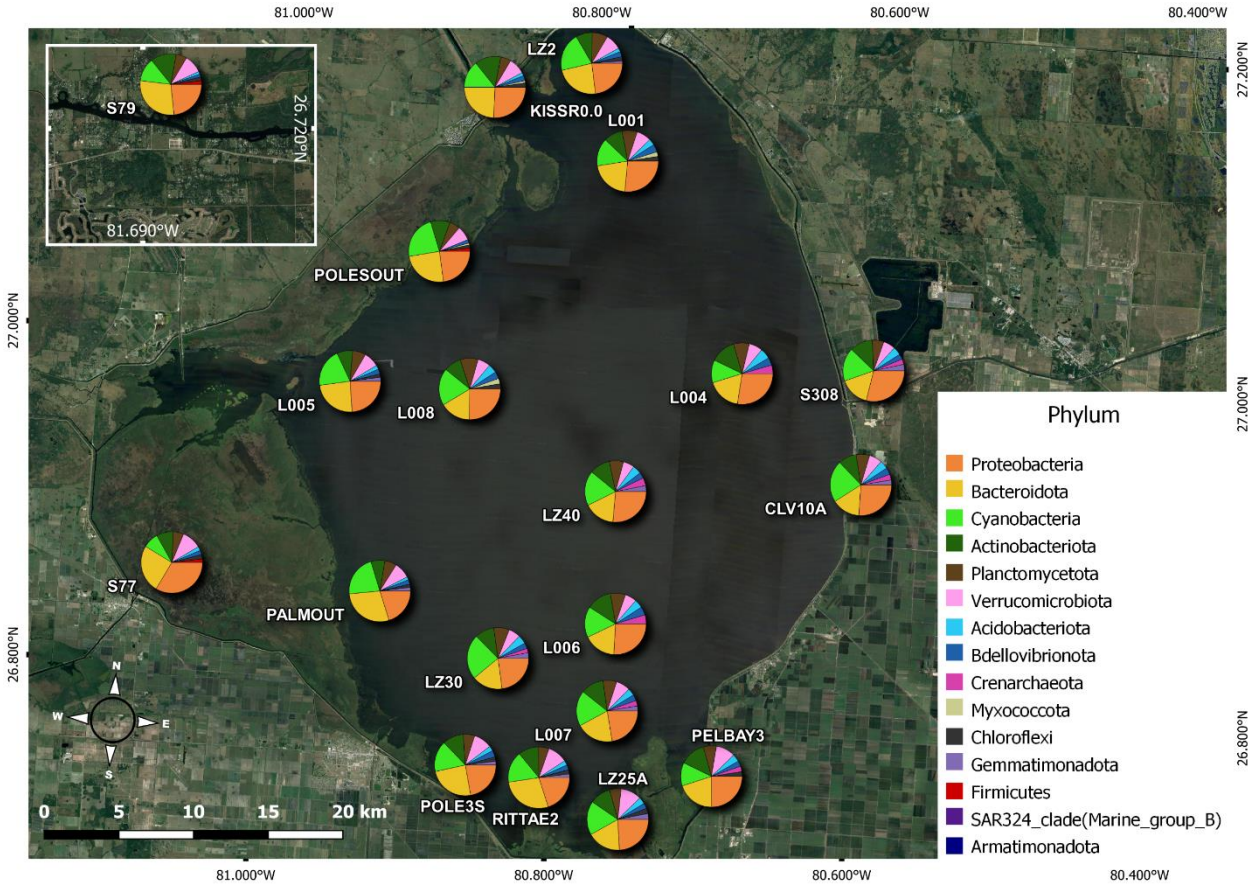

Figure 7. Pie charts showing the top phyla found in each station in Lake O within year 3 (2021).

Alpha diversity analyses

Alpha diversity was calculated using the Shannon diversity index, species evenness, species richness, and inverse Simpson diversity index. Year 3 (2021) exhibited significantly higher species richness than the previous two years (2019 and 2020, respectively) (year 1 vs. year 3, p = 0.0006; year 2 vs. year 3, p=0.0098) (Figure 8). Year 1 showed significantly higher species evenness throughout the microbial community compared to years 2 and 3, but year 2 was similar in species evenness compared to both years 1 and 3 (year 1 vs. year 2, p =0.042; year 1 vs. year 3, p=0.00013; year 2 vs. year 3, p=0.028) (Figure 8).

Within each year, alpha diversity differed by month (Table 3). The trends over time appeared to be seasonal, and analysis comparing season within each year showed that evenness specifically differed in year 2 (p = 0.00084) and year 3 (p = 0.037) (Figures 9-11). Alpha diversity also differed by zones across years 1 and 3, with year 2 showing no differences within all alpha diversity measures (Table 3, Figures 12-14). Alpha diversity differed by station within each year as well, with year 1 showing no significant differences in species evenness, year 2 only showing differences in species evenness, and year 3 showing differences in all the alpha diversity measures (Table 4).

Overall, the environmental variables measured did not strongly correlate to the alpha diversity in Lake O (Figure 15). Regarding species evenness, microcystin concentration showed the strongest correlation out of all the environmental variables (Pearson R^2 = -0.49) (Figure 15). Other environmental variables that correlated to species evenness included ammonia (Pearson R^2 = 0.11), nitrate + nitrite (Pearson R^2 = -0.10), and total phosphate (Pearson R^2 = -0.11) (Figure 15). Environmental variables that correlated to species richness include total nitrogen (Pearson R^2 = 0.17), TN:TP ratio (Pearson R^2 = -0.13), and total phosphorus (Pearson R^2 = 0.18) (Figure 15). The environmental variables that correlated to the diversity indices, Shannon and inverse Simpson, included microcystin (Pearson R^2 , shannon = -0.23; inv. Simpson = -0.20), nitrate + nitrite (Pearson R^2 , inv. Simpson = -0.10), total nitrogen (Pearson R^2 , shannon = 0.13; inv. Simpson = 0.17), total phosphorus (Pearson R^2 , shannon = 0.06; inv. Simpson = 0.10) and total phosphate (Pearson R^2 , inv. Simpson = -0.12) (Figure 15). There were no correlations between any of the alpha diversity measures and chlorophyll a, temperature, nor pH (Figure 15). *Microcystis* relative abundance had a srtong, negative correlation with species evenness (Pearson R^2 = -0.72), with

additional negative correlations with Shannon diversity index (Pearson R^2 = -0.23), and inverse Simpson diversity index (Pearson R^2 = -0.22) (Figure 15).

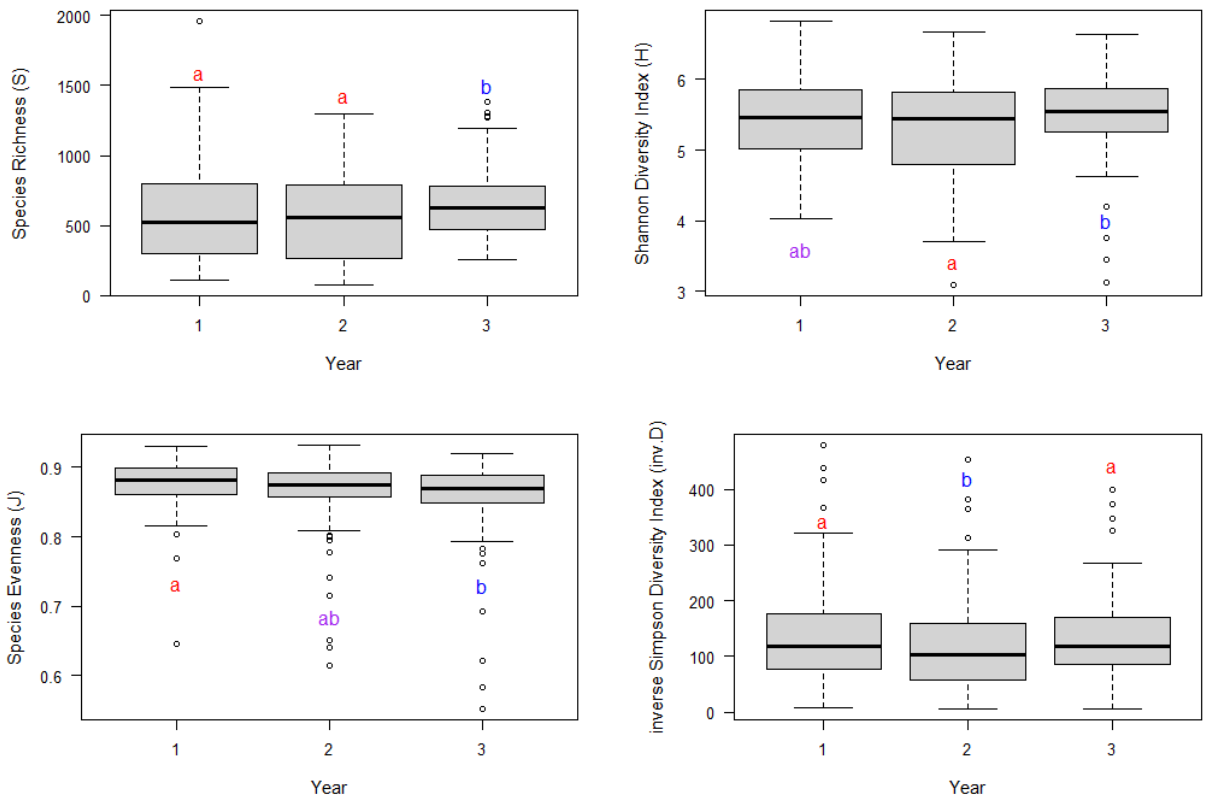

Figure 8. Alpha diversity comparison between years. Letters and colors represent the significant differences between each year; same letter and color indicate no differences and different letters and colors indicate significant differences are present ($p < 0.05$). Year 1 = 2019, Year 2 = 2020, and Year 3 = 2021.

Table 2. Kruskal-Wallis p-values for alpha diversity measure by month across each year. A star indicates that the p-value was significant ($p < 0.05$).

Alpha Diversity measure	Year 1	Year 2	Year 3
Species richness (S)	0.0017*	< 2.2e-16*	8.819e-08*
Species evenness (J)	0.13	0.00025*	2.848e-05*
Shannon Diversity Index (H)	0.0024*	< 2.2e-16*	8.126e-07*
Inverse Simpson Diversity Index (inv.D)	0.027*	< 2.2e-16*	1.383e-05

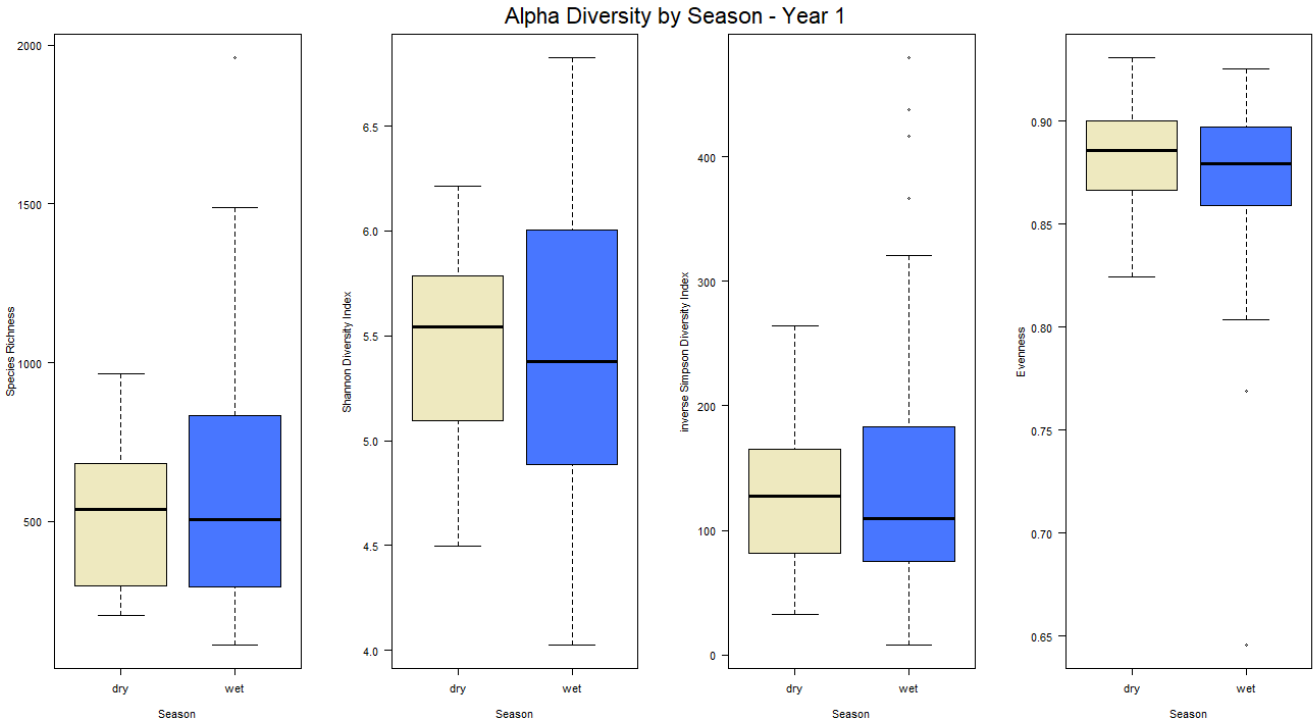

Figure 9. Alpha diversity measures across seasons in year 1. There were no significant differences between season and each alpha diversity measure. Tan = dry season; blue = wet season. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

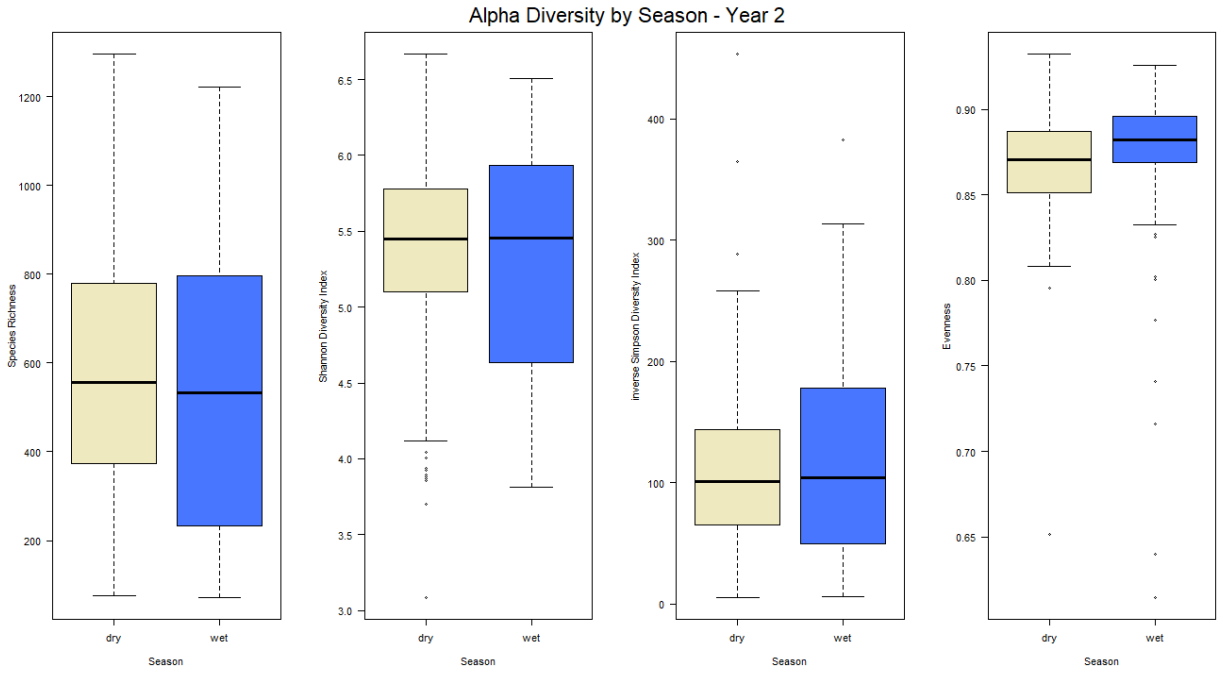

Figure 10. Alpha diversity measures across seasons in year 2. Significant differences were found in species evenness between seasons (p = 0.001). Tan = dry season; blue = wet season. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

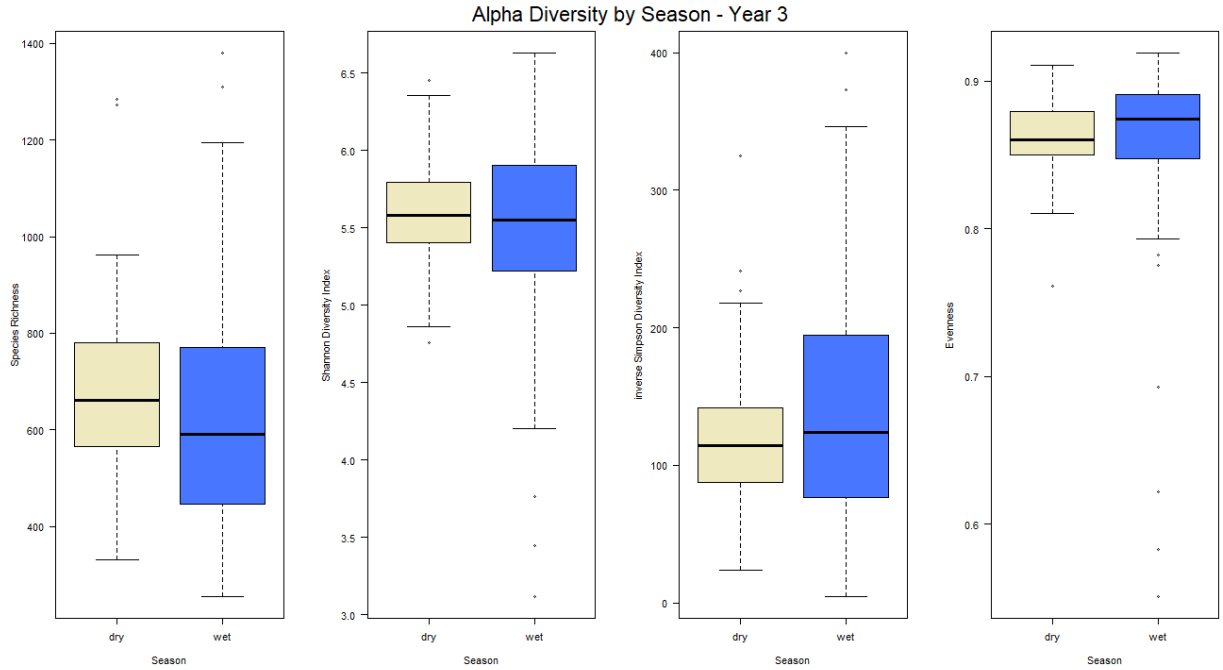

Figure 11. Alpha diversity measures across seasons in year 3. Significant differences were found in species evenness between seasons (p = 0.001). Tan = dry season; blue = wet season. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

Table 3. Kruskal-Wallis p-values for alpha diversity measure by zone across each year. A star indicates that the p-value was significant ($p < 0.05$).

Alpha Diversity measure	Year 1	Year 2	Year 3
Species richness (S)	0.0073*	0.54	0.00040*
Species evenness (J)	0.0033*	0.10	0.0015*
Shannon Diversity Index (H)	0.0082*	0.82	0.0020*
Inverse Simpson Diversity Index (inv.D)	0.035*	0.54	0.0034*

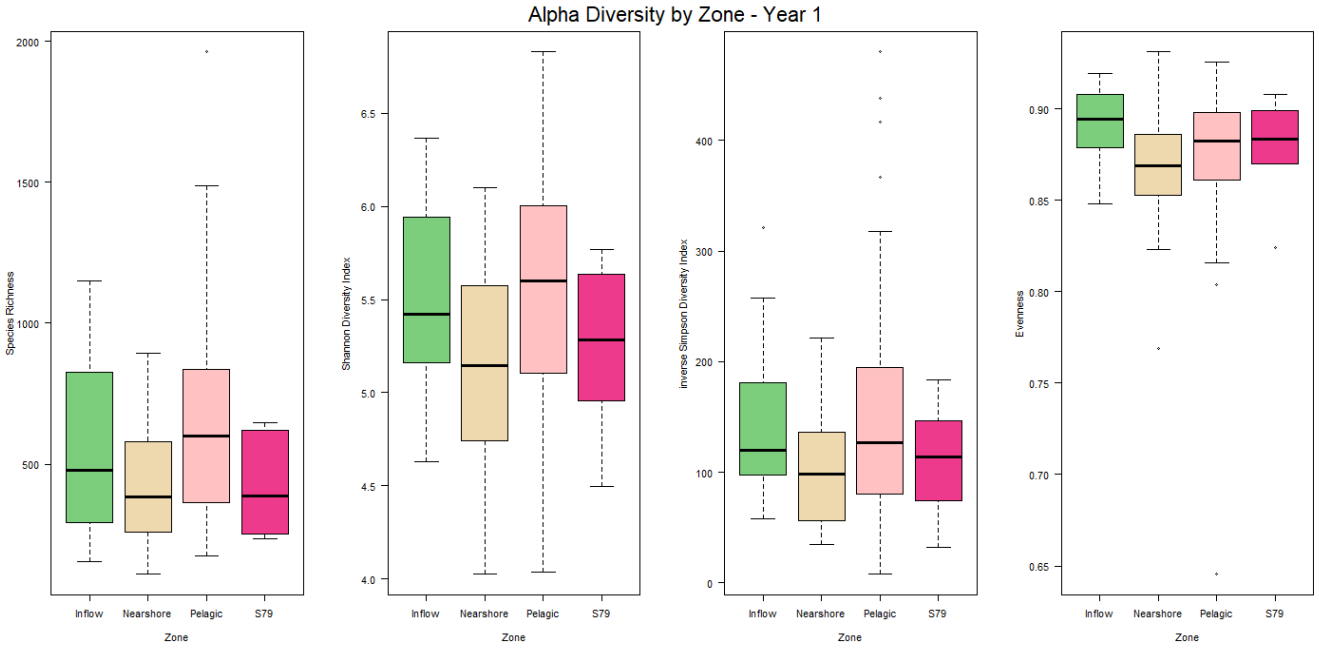

Figure 12. Alpha diversity measures across zones in year 1. Green = Inflow zone; Beige = Nearshore zone; Light pink = Pelagic zone; Bright pink = zone S79. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

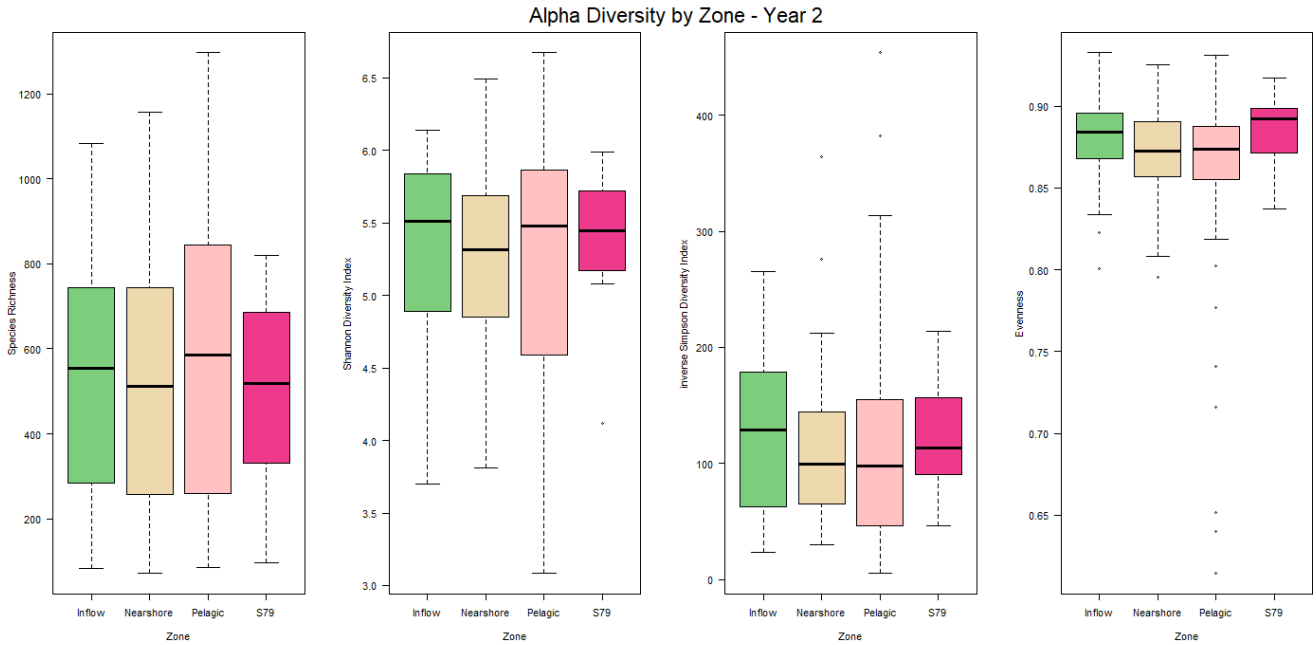

Figure 13. Alpha diversity measures across zones in year 2. Green = Inflow zone; Beige = Nearshore zone; Light pink = Pelagic zone; Bright pink = zone S79. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

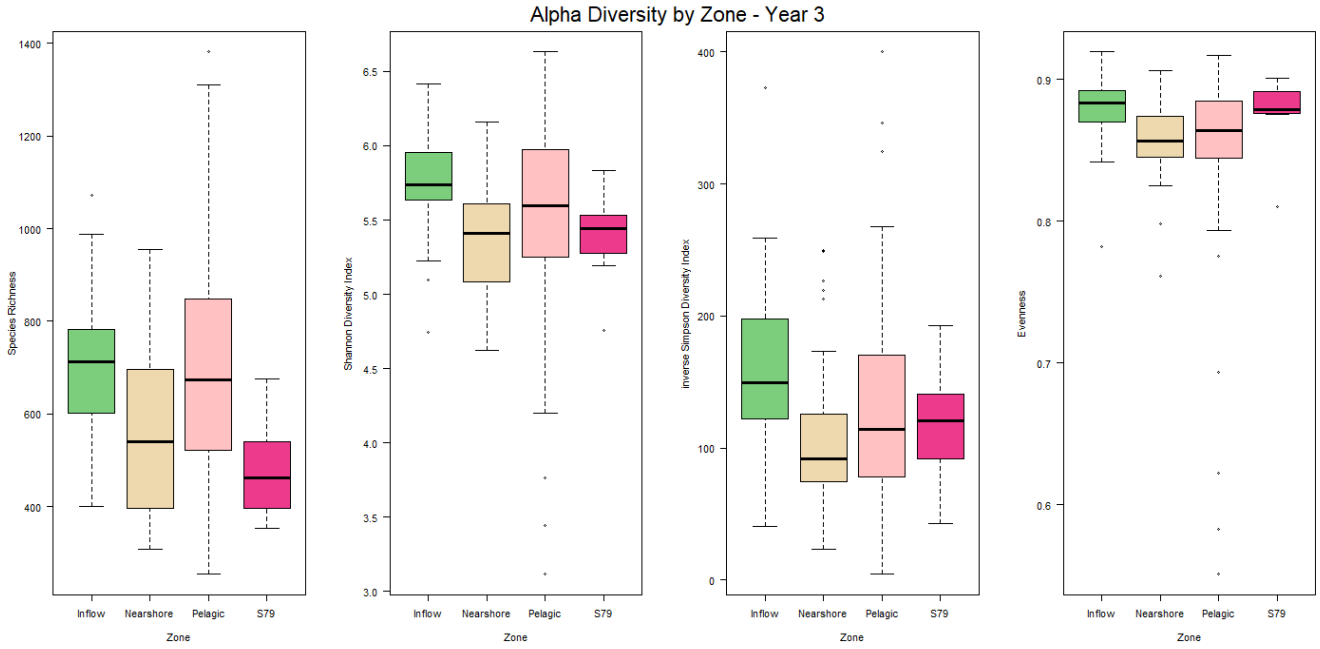

Figure 14. Alpha diversity measures across zones in year 3. Green = Inflow zone; Beige = Nearshore zone; Light pink = Pelagic zone; Bright pink = zone S79. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

Table 4. Kruskal-Wallis p-values for alpha diversity measure by station across each year. A star indicates that the p-value was significant ($p < 0.05$).

Alpha Diversity measure	Year 1	Year 2	Year 3
Species richness (S)	0.0054*	0.99	0.0091*
Species evenness (J)	0.016[a]	0.0080*	0.0015*
Shannon Diversity Index (H)	0.0025*	0.88	0.0068*
Inverse Simpson Diversity Index (inv.D)	0.0028*	0.31	0.0017*

[a]Although the p-value was significant, there were no differences found between the stations.

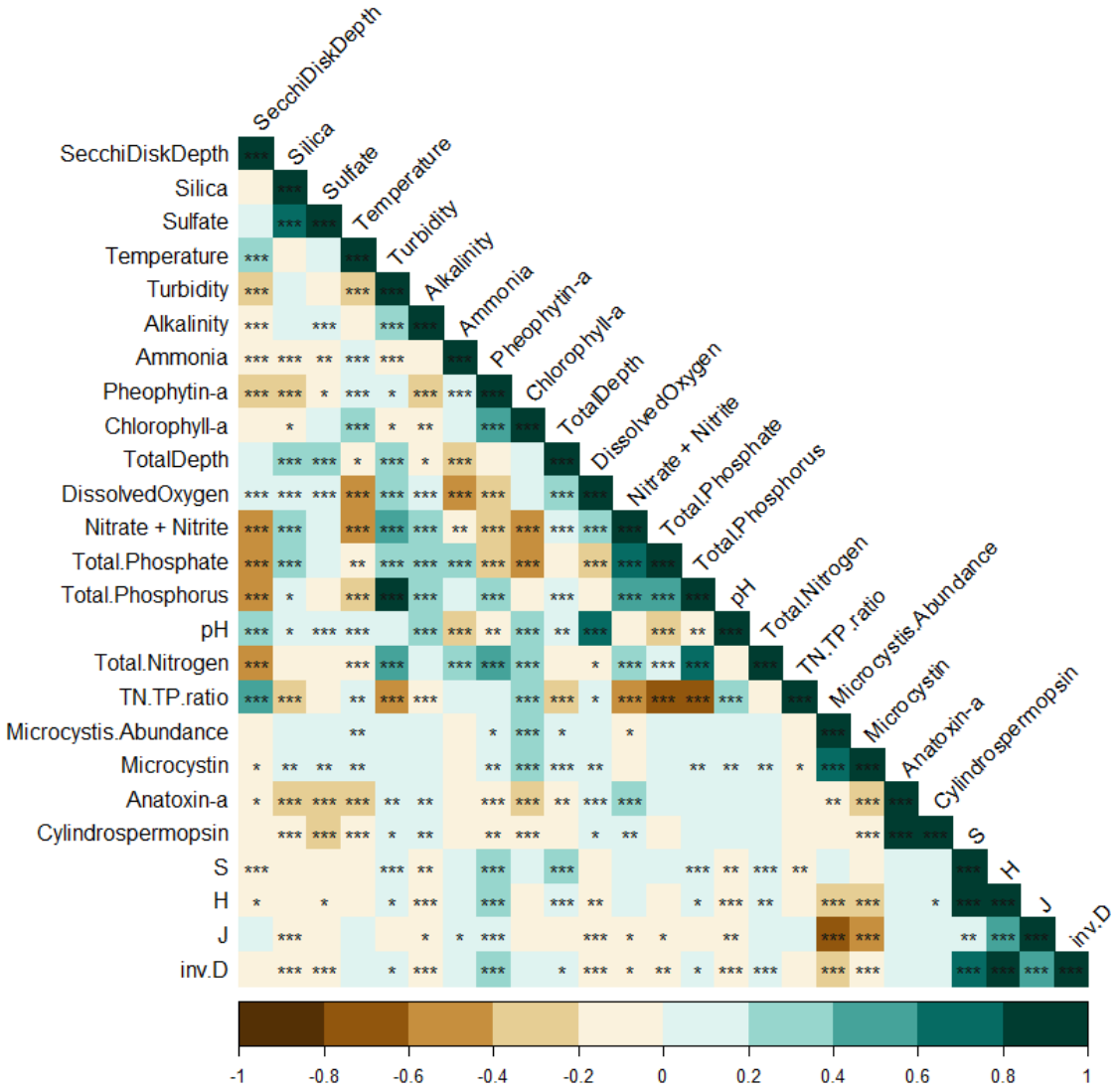

Figure 15. **Correlation heat map between the environmental variables and the alpha diversity indices.** Stars indicate the significance level; * = 0.05, ** = 0.01, *** = 0.001. No star indicates that the relationship is not significant. Alpha diversity measures can be found at the bottom of the heatmap: S = species richness, H = Shannon diversity index, J = species evenness, inv.D = inverse Simpson diversity index. TN.TP.ratio = ratio of total nitrogen and total phosphorus.

Venn diagram of core taxa between years

Each sampling year may have shared unique core taxa. To reiterate, core taxa is defined as any ASVs that were detected at a relative abundance of at least 0.1% and in at least 75% of the samples. A Venn diagram was created between each year, and it showed that all years shared 12 core taxa (Figure 16). Years 1 and 2 did not have any core taxa that was unique to them, nor did they share any core taxa (Figure 16). Year 3, however, had 14 unique core taxa, shared 4 core taxa with year 2, and shared 2 core taxa with year 1 (Figure 16). The taxonomic information for each taxon placed in the venn diagram can be found in Table 5. It can be seen from the table that the phylum Cyanobacteria are only found in the core taxa shared between years 2 and 3 and within the unique core taxa of year 3 (Table 5). Verrucomicrobiota was the only phylum of heterotrophic bacteria found within the shared taxa between year 2 and year 3 (Figure 16, Table 5).

Figure 16. Venn diagram of the number of shared core taxa between years across the sampling period. Year 1 = red; Year 2 = blue; Year 3 = green. Numbers represent the number of taxa.

Table 5. Core taxa comparisons between years (corresponding to venn diagram). Taxonomic information is structured by phylum, class, order, family, and genus. Dashes indicate that there were no shared taxa between specified years.

	Taxonomic Information
Year 1 Only	——
Year 2 Only	——
Year 3 Only	1. Actinobacteriota, Acidimicrobiia, Microtrichales, Ilumatobacteraceae, CL500-29_marine_group 2. Actinobacteriota, Actinobacteria, Frankiales, Sporichthyaceae, 3. Actinobacteriota, MB-A2-108, MB-A2-108, MB-A2-108, MB-A2-108 4. Verrucomicrobiota, Verrucomicrobiae, Pedosphaerales, Pedosphaeraceae, SH3-11 5. Proteobacteria, Gammaproteobacteria 6. Proteobacteria, Gammaproteobacteria, Burkholderiales, Oxalobacteraceae, 7. Proteobacteria, Gammaproteobacteria, Gammaproteobacteria_Incertae_Sedis, Unknown_Family, Acidibacter 8. Proteobacteria, Gammaproteobacteria, JG36-TzT-191, JG36-TzT-191, JG36-TzT-191 9. Proteobacteria, Gammaproteobacteria, Oceanospirillales, Pseudohongiellaceae, BIyi10 10. Bacteroidota, Bacteroidia, Sphingobacteriales, AKYH767, AKYH767 11. Bacteroidota, Bacteroidia, Sphingobacteriales, env.OPS_17, env.OPS_17 12. Bacteroidota, Bacteroidia, Sphingobacteriales, NS11-12_marine_group, NS11-12_marine_group 13. Cyanobacteria, Cyanobacteriia, Synechococcales, Cyanobiaceae, Cyanobium_PCC-6307 14. Gemmatimonadota, Gemmatimonadetes, Gemmatimonadales, Gemmatimonadaceae
Years 1 & 2	——
Years 1 & 3	1. Actinobacteriota, Actinobacteria, Frankiales, Sporichthyaceae, hgcI_clade 2. Proteobacteria, Alphaproteobacteria, Rhizobiales, Rhizobiales_Incertae_Sedis, uncultured
Years 2 & 3	1. Verrucomicrobiota, Verrucomicrobiae, Opitutales, Opitutaceae, Opitutus 2. Cyanobacteria, Cyanobacteriia, Synechococcales, Cyanobiaceae, Cyanobium_PCC-6307 3. Cyanobacteria, Cyanobacteriia, Synechococcales, Cyanobiaceae, Cyanobium_PCC-6307 4. Cyanobacteria, Cyanobacteriia, Synechococcales, Cyanobiaceae, Cyanobium_PCC-6307

ALL years	1. Actinobacteriota, Acidimicrobiia, Microtrichales, Ilumatobacteraceae, CL500-29_marine_group 2. Actinobacteriota, Acidimicrobiia, Microtrichales, Ilumatobacteraceae, CL500-29_marine_group 3. Actinobacteriota, Acidimicrobiia, Microtrichales, Ilumatobacteraceae, CL500-29_marine_group 4. Actinobacteriota, Actinobacteria, Frankiales, Sporichthyaceae, hgcI_clade 5. Bacteroidota, Bacteroidia, Chitinophagales, Saprospiraceae, Candidatus_Aquirestis 6. Bacteroidota, Bacteroidia, Flavobacteriales, Crocinitomicaceae, Fluviicola 7. Bacteroidota, Kapabacteria, Kapabacteriales, Kapabacteriales, Kapabacteriales 8. Verrucomicrobiota, Verrucomicrobiae, Methylacidiphilales, Methylacidiphilaceae, uncultured 9. Proteobacteria, Alphaproteobacteria, Rickettsiales, Rickettsiaceae, Candidatus_Megaira 10. Chloroflexi, SL56_marine_group, SL56_marine_group, SL56_marine_group, SL56_marine_group 11. Planctomycetota, Phycisphaerae, Phycisphaerales, Phycisphaeraceae, CL500-3 12. Proteobacteria, Gammaproteobacteria, Burkholderiales, Burkholderiaceae, Limnobacter

Beta diversity analyses

Beta diversity was calculated using Bray-Curtis dissimilarity. Following ANOSIM and PERMANOVA analyses, it was revealed that there were significant differences between stations (ANOSIM R = 0.1967; p = 0.01) across all sampling years. However, there were no significant differences in year (p = 0.75), season (p = 0.78), month (p = 0.91), nor zone (p = 0.19) across the sampling years. When investigating within each year, there were significant differences by station across each year (year 1, p = 0.001; year 2, p = 0.001; year 3, p = 0.001) and there were significant differences by zone within year 1 (p = 0.001) and year 3 (p = 0.001).

Environmental variables were fitted onto a CCA plot through vectors to show which environmental variables may be driving the differences in the microbial community within the lake across the sampling period and within each year (Figures 18-21). The length of the vector is proportional to its importance and the angle between two vectors reflects the degree of correlation between variables (Sarker, et al., 2014). To reiterate, the environmental variable vectors that were included in the CCA plots exhibited a significant effect ($p < 0.05$) and correlation (Pearson $R^2 > 0.3$) on the microbial community of Lake O. Across all three years, the environmental variables accounted for about 14.47% of the variation within the microbial communities in Lake O and these variables included TN:TP ratio (Pearson $R^2 = 0.57$), pH (Pearson $R^2 = 0.34$), nitrate + nitrite (Pearson $R^2 = 0.55$), dissolved oxygen (Pearson $R^2 = 0.43$), turbidity (Pearson $R^2 = 0.42$), total phosphate ("phosphate.ortho"; Pearson $R^2 = 0.48$), and ammonia (Pearson $R^2 = 0.34$) (Figure 18). In year 1, the environmental variables accounted for about 17.44% of the variation within the microbial communities in Lake O and these variables included TN:TP ratio (Pearson $R^2 = 0.65$), pH (Pearson $R^2 = 0.51$), nitrate + nitrite (Pearson $R^2 = 0.46$), dissolved oxygen (Pearson $R^2 = 0.49$), turbidity (Pearson $R^2 = 0.31$), secchi disk depth (Pearson $R^2 = 0.30$), and ammonia (Pearson $R^2 = 0.60$) (Figure 19). In year 2, the environmental variables accounted for about 17.26% of the variation within the microbial communities in Lake O and these variables included TN:TP ratio (Pearson $R^2 = 0.62$), pH (Pearson $R^2 = 0.69$), nitrate + nitrite (Pearson $R^2 = 0.55$), dissolved oxygen (Pearson $R^2 = 0.51$), turbidity (Pearson $R^2 = 0.52$), total phosphate ("phosphate.ortho"; Pearson $R^2 = 0.35$), ammonia (Pearson $R^2 = 0.35$), and chlorophyll a (Pearson $R^2 = 0.35$) (Figure 20). In year 3, the environmental variables accounted for about 20.69% of the variation within the microbial communities in Lake O and these variables included TN:TP ratio (Pearson $R^2 = 0.36$), nitrate +

nitrite (Pearson $R^2 = 0.67$), dissolved oxygen (Pearson $R^2 = 0.30$), alkalinity (Pearson $R^2 = 0.31$), temperature (Pearson $R^2 = 0.36$), total phosphate ("phosphate.ortho"; Pearson $R^2 = 0.44$), *Microcystis* relative abundance (Pearson $R^2 = 0.55$), and chlorophyll a (Pearson $R^2 = 0.39$) (Figure 21). When comparing the environmental variables that influenced microbial community composition across the sampling years, year 1 was the only year in which secchi disk depth influenced microbial community composition (Figure 18). Total phosphate concentration and chlorophyll a concentration were environmental variables shared between year 2 and year 3 that were not included in year 1 that drove microbial community composition (Figures 19 and 20). The environmental variables unique to year 3 in driving the microbial community composition included alkalinity, temperature, and *Microcystis* abundance.

Across the entire sampling period, the microbial community composition of year 3 was closely associated with total phosphate ("phosphate. ortho" in figure 18), nitrate + nitrite, and turbidity (Figure 18). In year 1 and year 3, nearshore and pelagic zones were similar in microbial community composition while inflow and S79 zones were similar in microbial community composition (Figures 19 and 21). In year 1, the microbial community composition of the nearshore and pelagic zones was driven mostly by nitrate + nitrite, turbidity, and TN:TP ratio, while the communities of the inflow and S79 zones were driven mostly by ammonia (Figure 19). In year 3, the microbial community composition of the nearshore and pelagic zones was driven by nitrate + nitrite, total phosphate, *Microcystis* abundance, chlorophyll-a, and temperature. The microbial community composition of the inflow and S79, however, doesn't seem to be driven primarily by any of the environmental factors shown in the plot (Figure 22). Year 2 had significant differences between stations (Figure 20) and no significant differences between zones (Figure 21). However, each station is located within a certain ecological zone in the lake. Thus, to better interpret the station plot, the zone plot will be used. When looking at the zones of each station, the stations located in the nearshore and pelagic zones were clustered together and mostly driven by nitrate + nitrite concentrations, turbidity, with TN:TP ratio also driving microbial community within the nearshore zone (Figure 20 and figure 22). Stations located in the inflow and S79 zones were also clustered together but there were some stations from the pelagic and inflow zones that were driven by the same environmental variables (chlorophyll a, TN:TP ratio, and ammonia) (Figure 20 and figure 22).

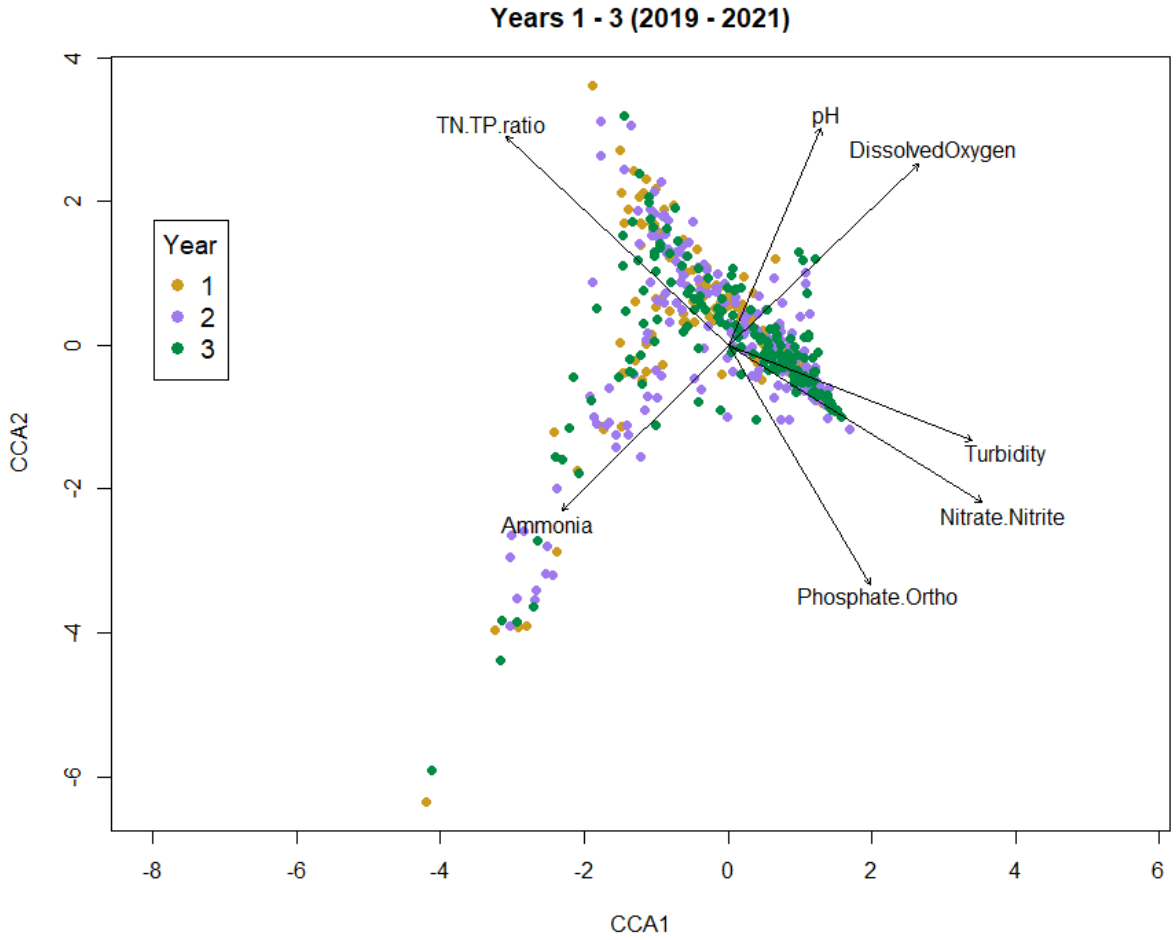

Figure 17. CCA plot based on species composition of each sample over the sampling period by year. Arrows indicate the direction and magnitude of the environmental variables that showed a significant effect ($p<0.05$) and correlation ($R^2 \geq 0.3$).

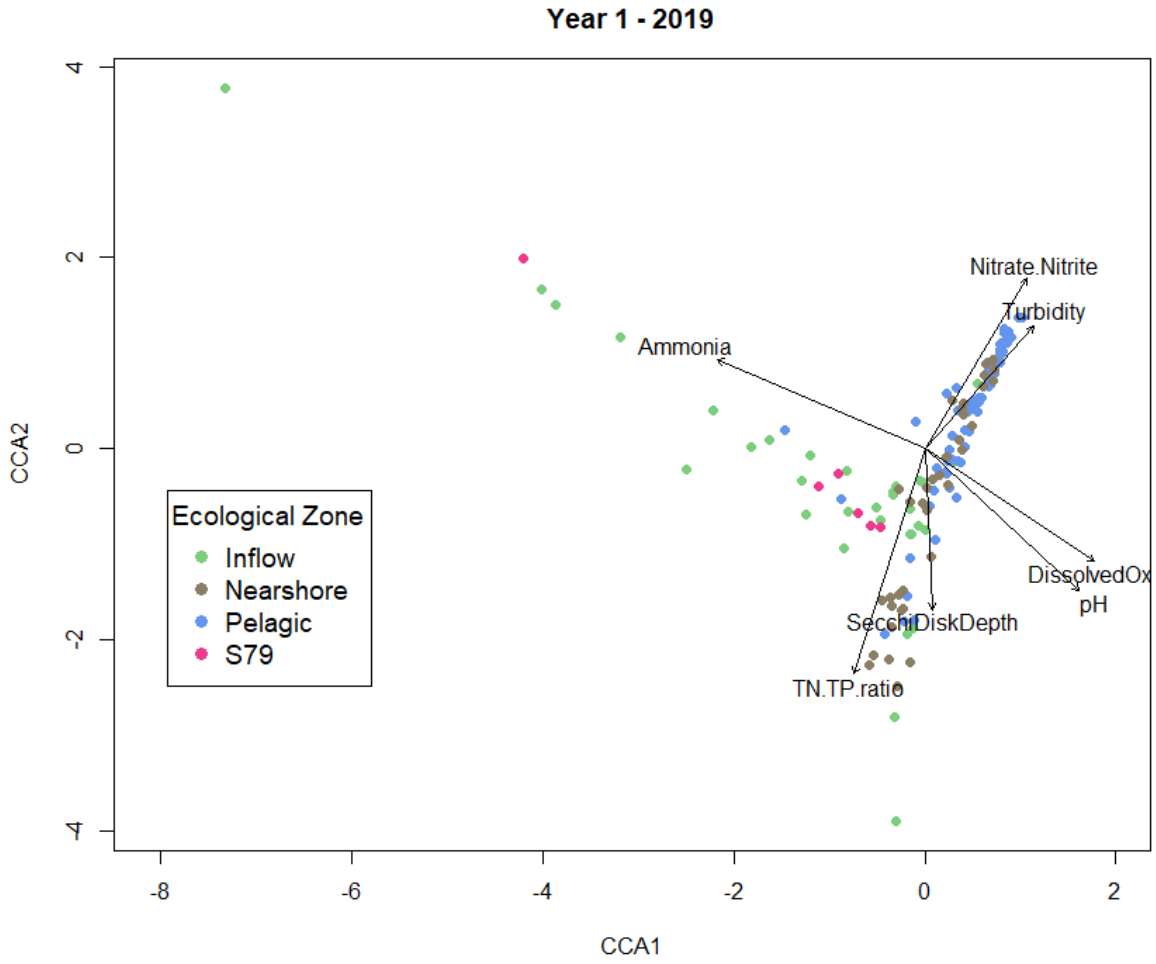

Figure 18. CCA plot based on species composition of each sample in year 1 by zone. Arrows indicate the direction and magnitude of the environmental variables that showed a significant effect ($p<0.05$) and correlation ($R^2 \geq 0.3$).

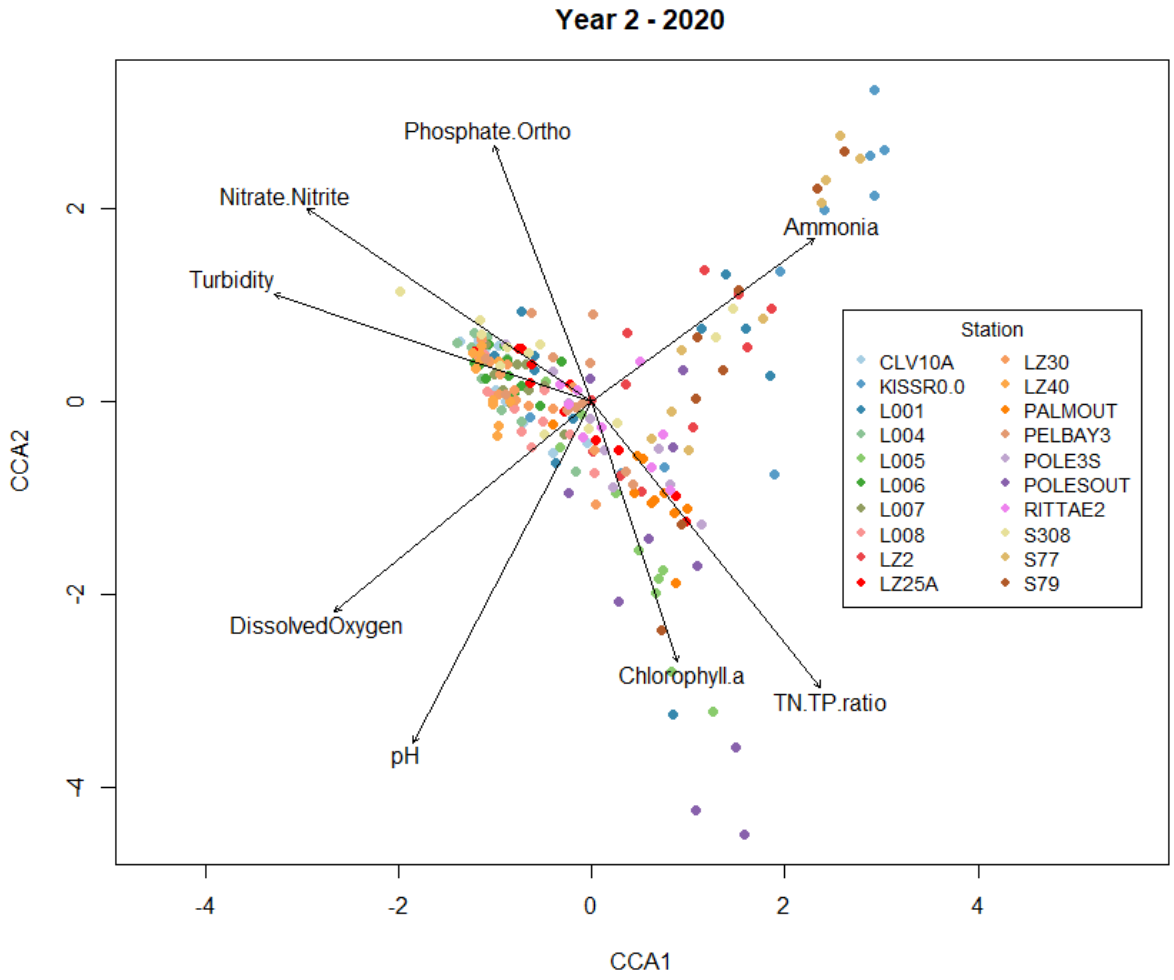

Figure 19. CCA plot based on species composition of each sample in year 2 by station. Arrows indicate the direction and magnitude of the environmental variables that showed a significant effect ($p<0.05$) and correlation ($R^2 \geq 0.3$).

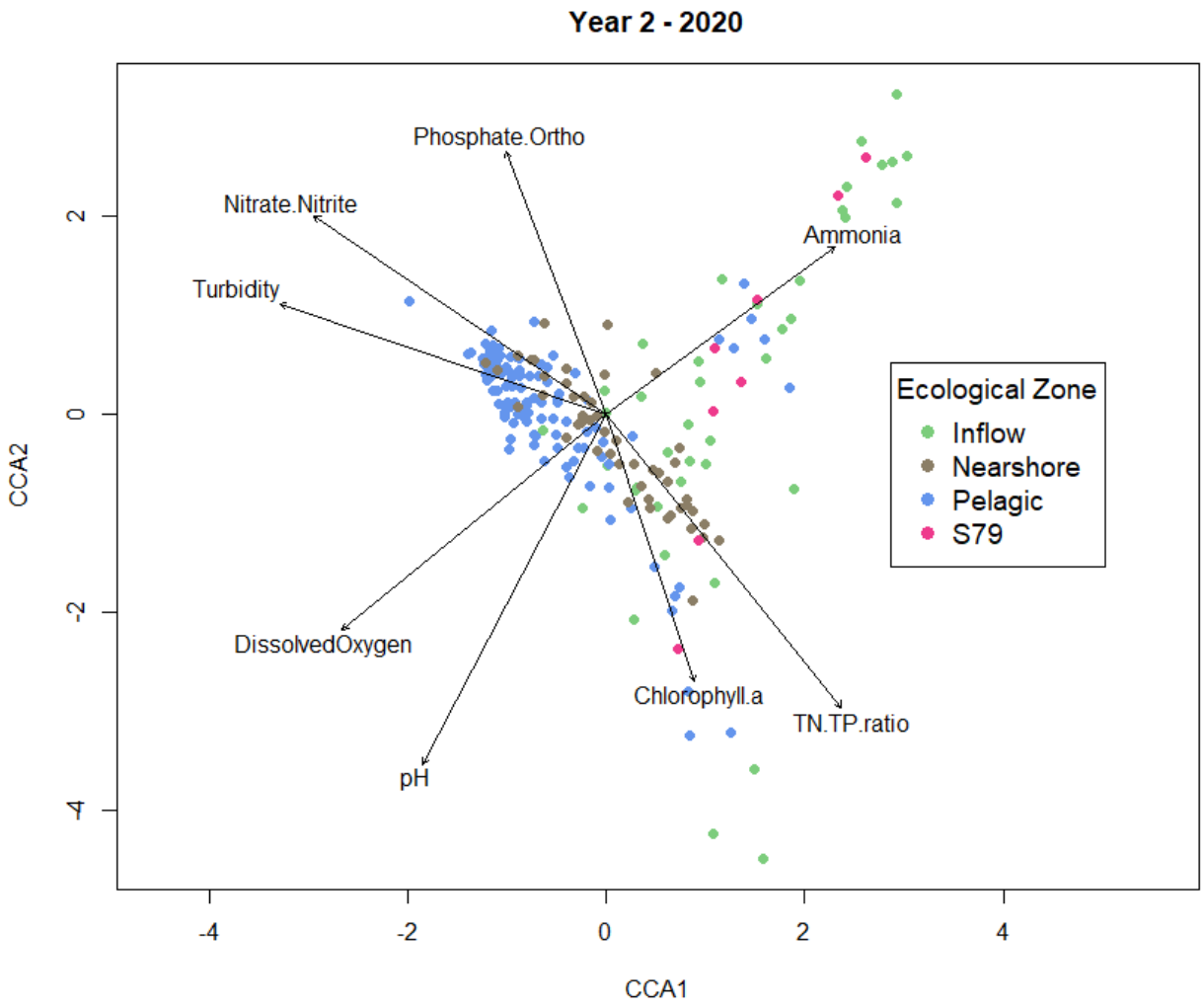

Figure 20. CCA plot based on species composition of each sample in year 2 by zone. Arrows indicate the direction and magnitude of the environmental variables that showed a significant effect ($p<0.05$) and correlation ($R^2 \geq 0.3$).

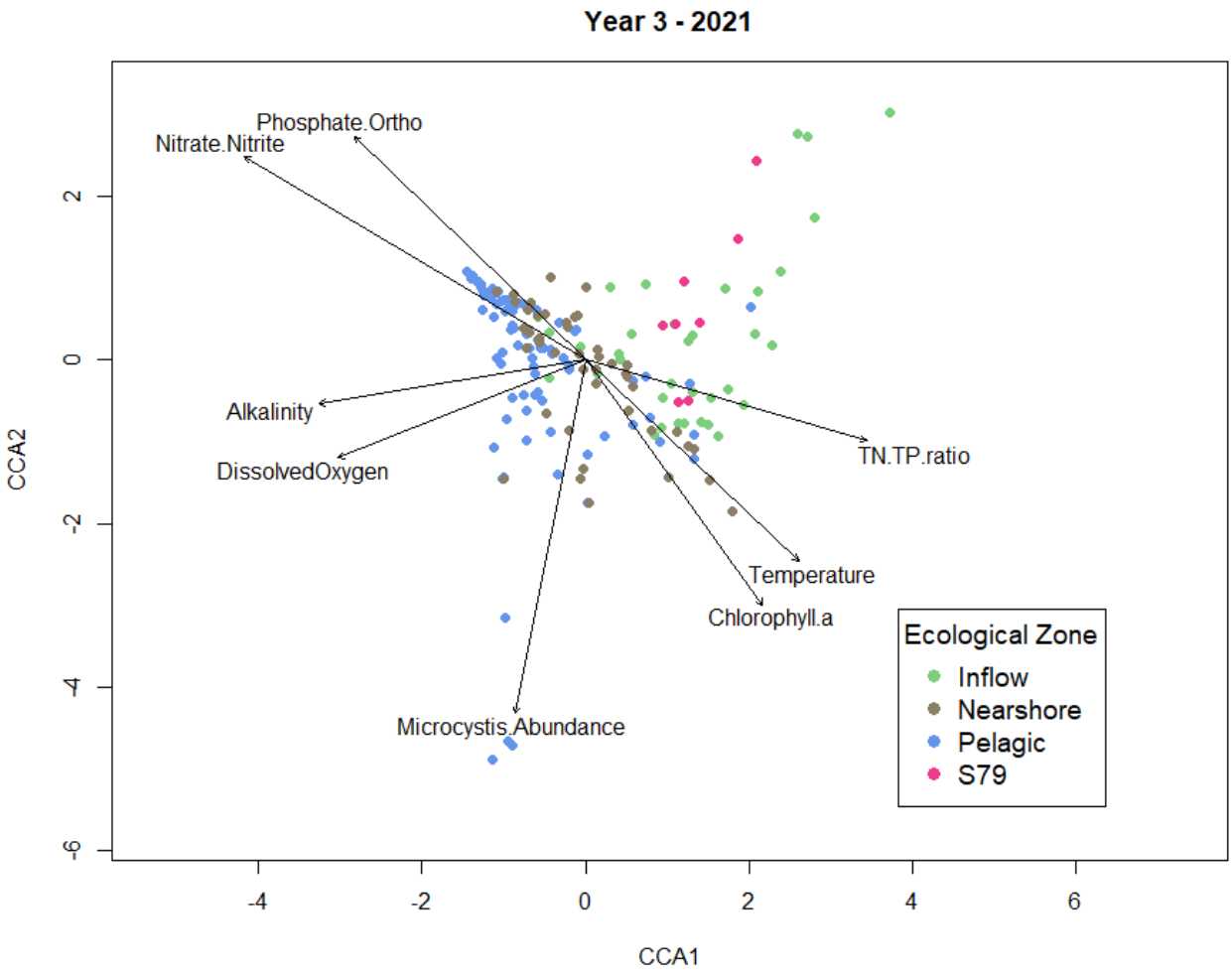

Figure 21. CCA plot based on species composition of each sample in year 3 by zone. Arrows indicate the direction and magnitude of the environmental variables that showed a significant effect ($p<0.05$) and correlation ($R^2 \geq 0.3$).

Co-occurrence network with Microcystis

There was a total of 22 bacteria taxa that appeared to co-occur with the genus *Microcystis* (Figure 22). The network consisted of two clusters around *Microcystis*, one with 18 taxa and another with 4 taxa. Most of the bacteria fall under the phylum Proteobacteria with some occurring in other phyla such as Bacteroidota and Gemmatimonadota. The three strongest relationships shared with *Microcystis* were between uncultured bacteria belonging to the family Sutterallaceae (Pearson R = 0.836), the genus *Pseudanabaena_PCC-7429* (Pearson R = 0.811), and the genus *Silanimonas* (Pearson R = 0.807). It is evident that the genus *Microcystis* co-occurs primarily with heterotrophic bacterial taxa, with only two relationships with other Cyanobacteria taxa (Figure 22).

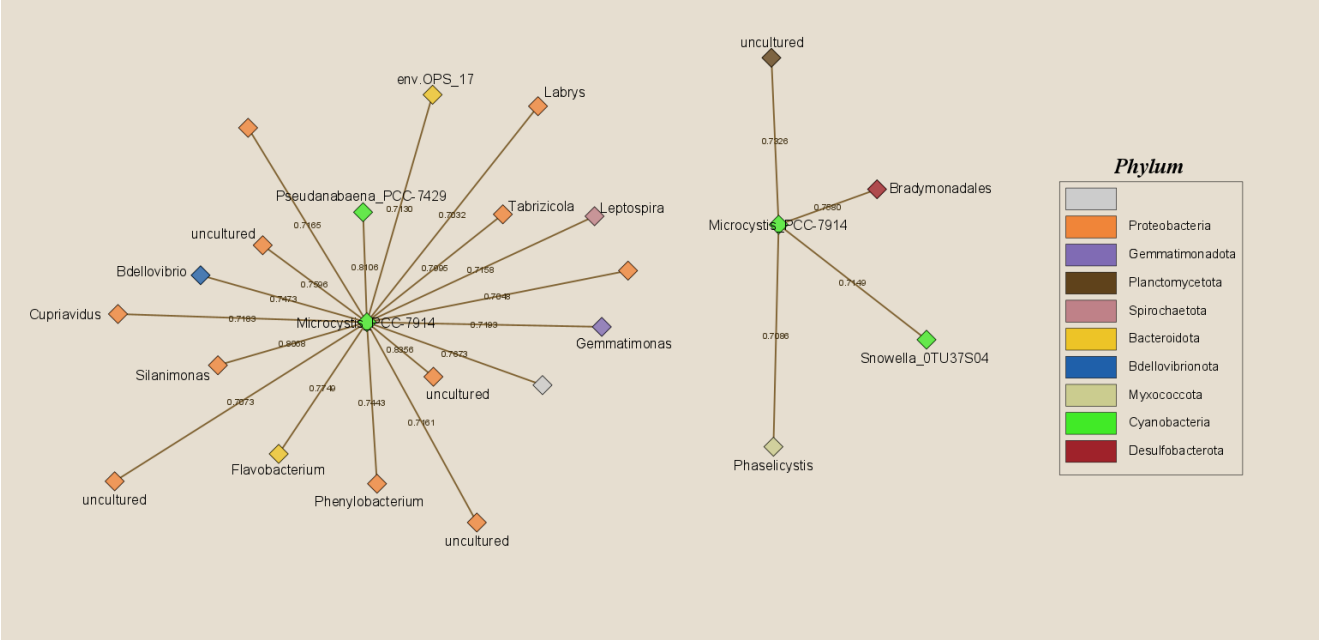

Figure 22. Co-occurrence network of genera sharing a significantly strong positive correlation (p = 0.05; R^2 > 0.7) with the genus *Microcystis.* Node color indicates the phylum corresponding to the genera shown. The numbers shown on the edges of the network signify the R^2 values of the relationship.

Environmental variables over sampling period

After uncovering which environmental variables were in close association with the microbial community beta diversity, selected environmental variables were plotted against the sampling period (by month across the years) (Figures 23-34). The only environmental variable that stayed relative constant with minor changes across the sampling period was pH (Figure 29). However, there were several instances of decreased pH within year 2 and year 3 during the late summer to winter months (7-12) (Figure 29). TN:TP ratio and nitrate + nitrite concentration showed some seasonal changes (Figure 31 and Figure 28, respectively). TN:TP ratio showed a decrease during spring months (3-5) and began to increase into the summer months (6-7) across all three years. Year 1 experienced instances of the highest TN:TP ratio compared to year 2 and year 3 (Figure 31). Nitrate + nitrite concentrations showed an overall decrease in concentration during the summer months into early fall months (6-9) (Figure 28). Year 2 experienced several instances of the highest concentration of nitrate + nitrite compared to year 1 and year 3 (Figure 28).

Most of the remaining selected environmental variables displayed changes from year-to-year. The total depth of Lake O was lower in year 1 while year 2 and year 3 experienced increasing average depths (Figure 33). Year 1 and year 3 experienced warmer water temperatures for a longer period compared to year 2, which exhibited a smoother transition between water temperature gradients across months (Figure 30). Ammonia concentrations remained constant in year 1, with only three instances being substantially higher than average (Figure 24a). Year 3 also portrayed the same pattern; however, there was only one instance where the concentration was substantially above average (Figure 24c). Year 2 showed the most instances that were above average concentrations compared to the other two years (Figure 24b). Both *Microcystis* relative abundance and microcystin concentration were higher during year 2 and year 3 and lowest during year 1 (Figure 27 and Figure 26, respectively). Chlorophyll a concentration exhibited the same pattern—with year 1 exhibiting lower concentrations than year 2 and year 3 (Figure 25). Year 1 and year 3 exhibited an unstable increase-decrease cycle in total nitrogen concentration across the monthly averages, while year 2 experienced only two increase averages during March and November (Figure 32). Total phosphorus also experienced this pattern in concentration (Figure 23). The

average concentration of total phosphate stayed within the same range across the years until it began to decrease during July of year 3 (Figure 34).

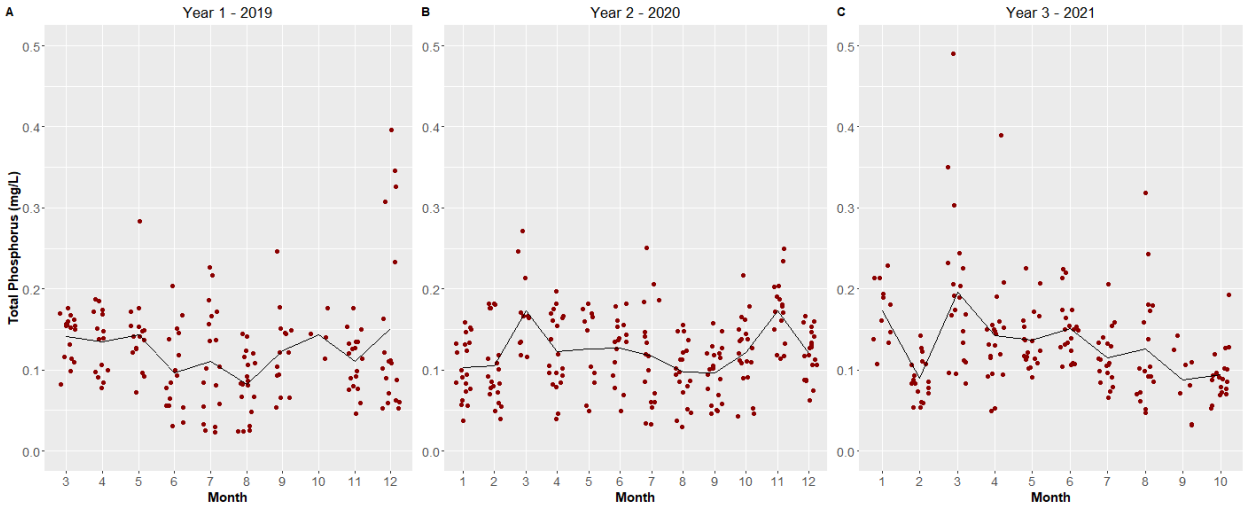

Figure 23. Scatterplot of total phosphorus concentrations (mg/L) over the sampling period. The black line depicts the average concentration per month across the years.

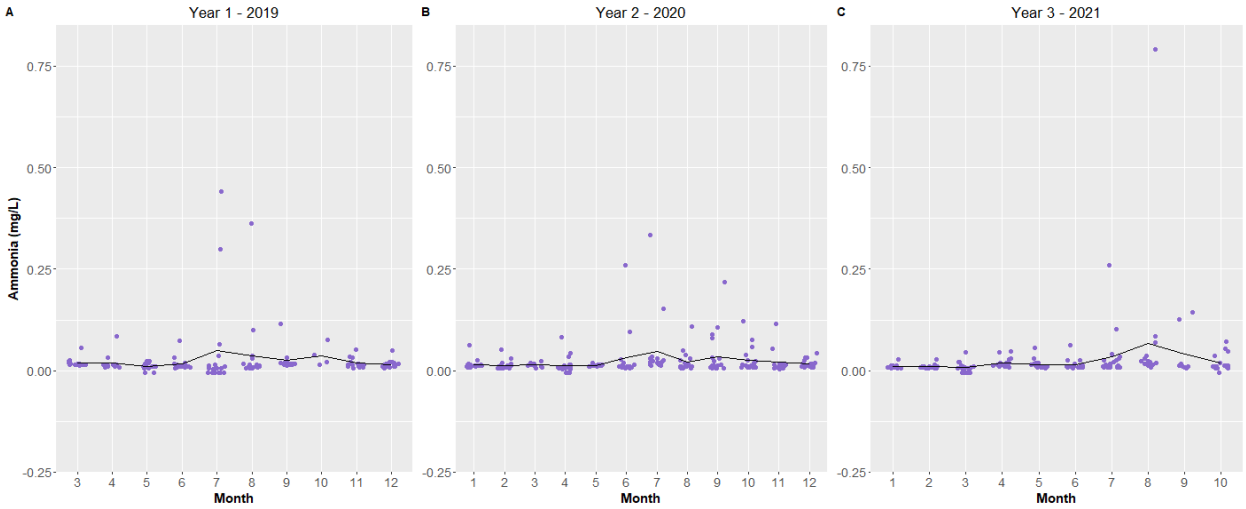

Figure 24. Scatterplot of ammonia concentrations (mg/L) over the sampling period. The black line depicts the average concentration per month across the years.

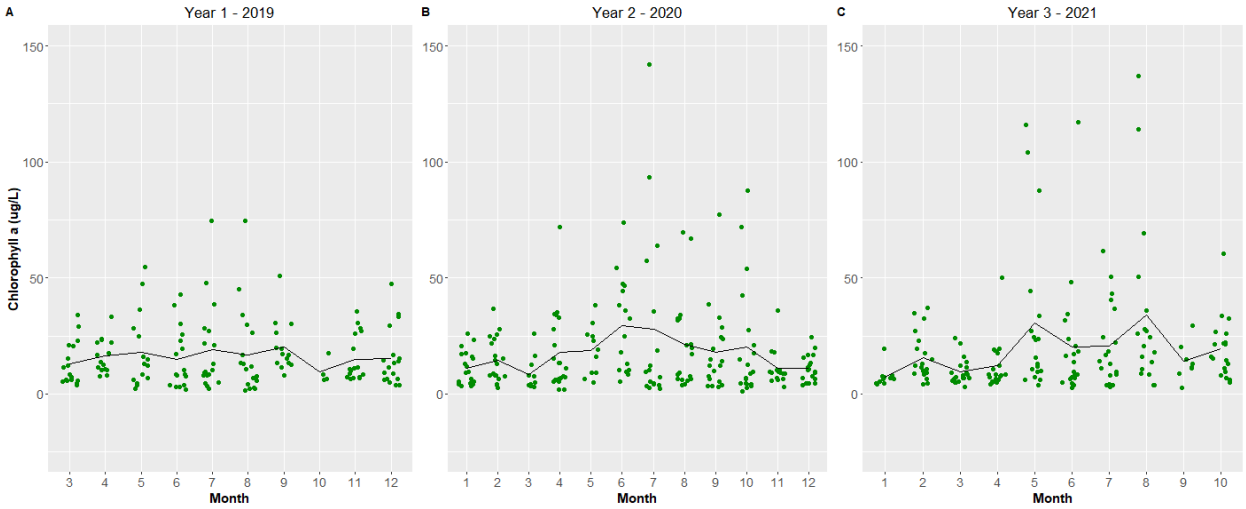

Figure 25. Scatterplot of total chlorophyll a concentration (µg/L) over the sampling period. The black line depicts the average concentration per month across the years.

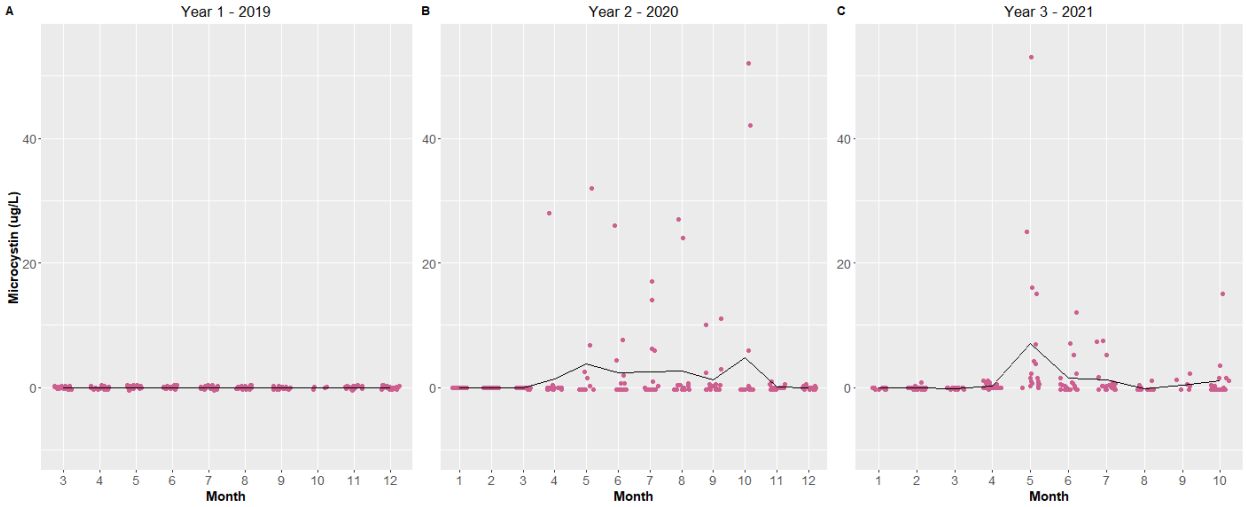

Figure 26. Scatterplot of microcystin concentrations (µg/L) over the sampling period. The black line depicts the average concentration per month across the years.

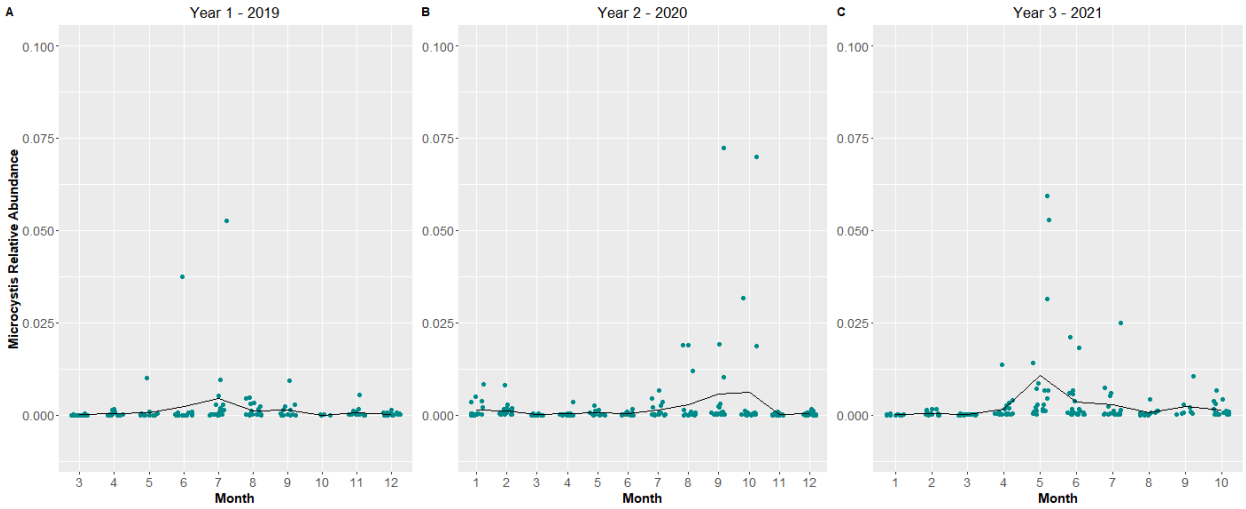

Figure 27. Scatterplot of *Microcystis* relative abundance over the sampling period. The black line depicts the average abundance per month across the years.

Figure 28. Scatterplot of nitrate + nitrite concentration (mg/L) over the sampling period. The black line depicts the average concentration per month across the years.

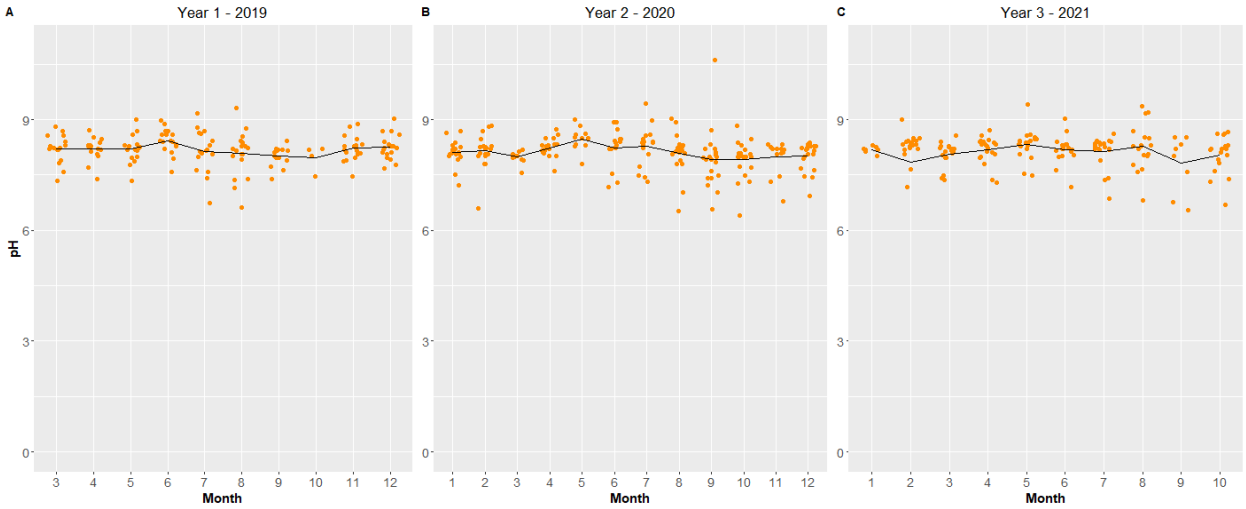

Figure 29. Scatterplot of surface water pH over the sampling period. The black line depicts the average pH per month across the years.

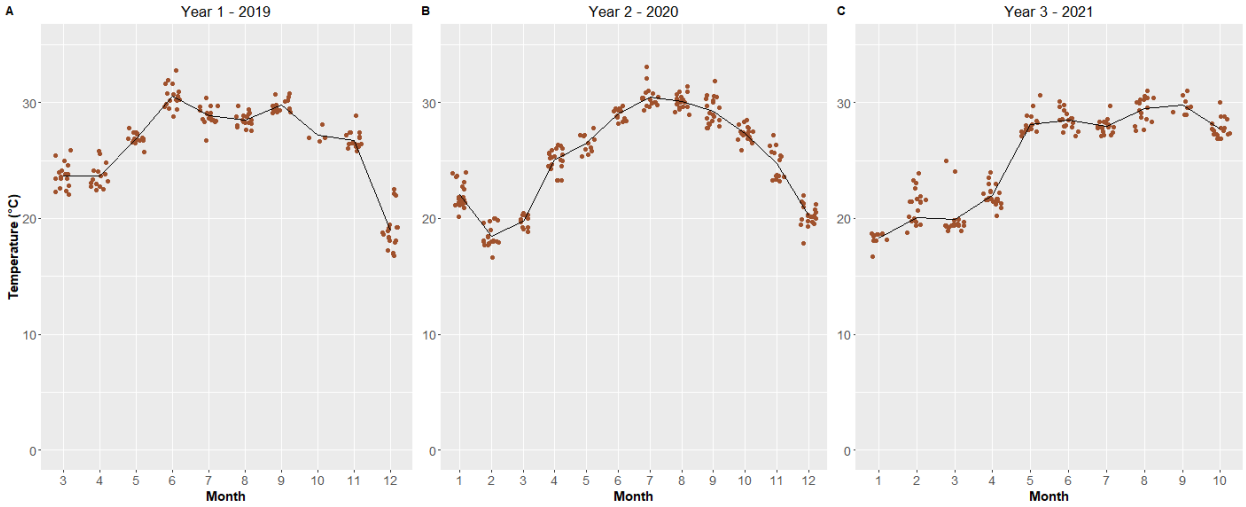

Figure 30. Scatterplot of surface water temperature (°C) over the sampling period. The black line depicts the average temperature per month across the years.

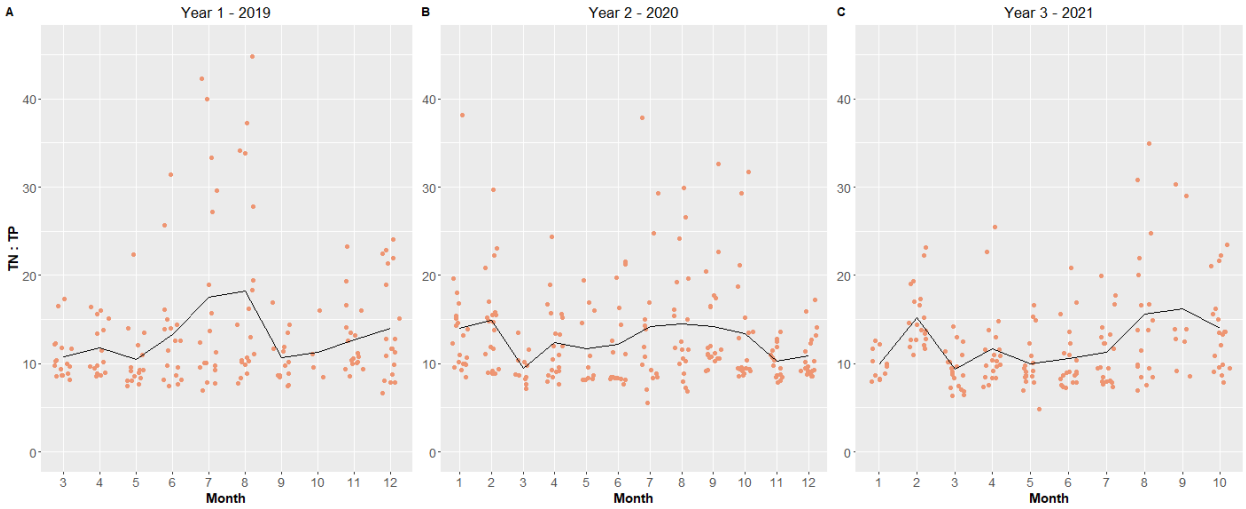

Figure 31. Scatterplot of the ratio of total nitrogen and total phosphorus over the sampling period. The black line depicts the average ratio per month across the years.

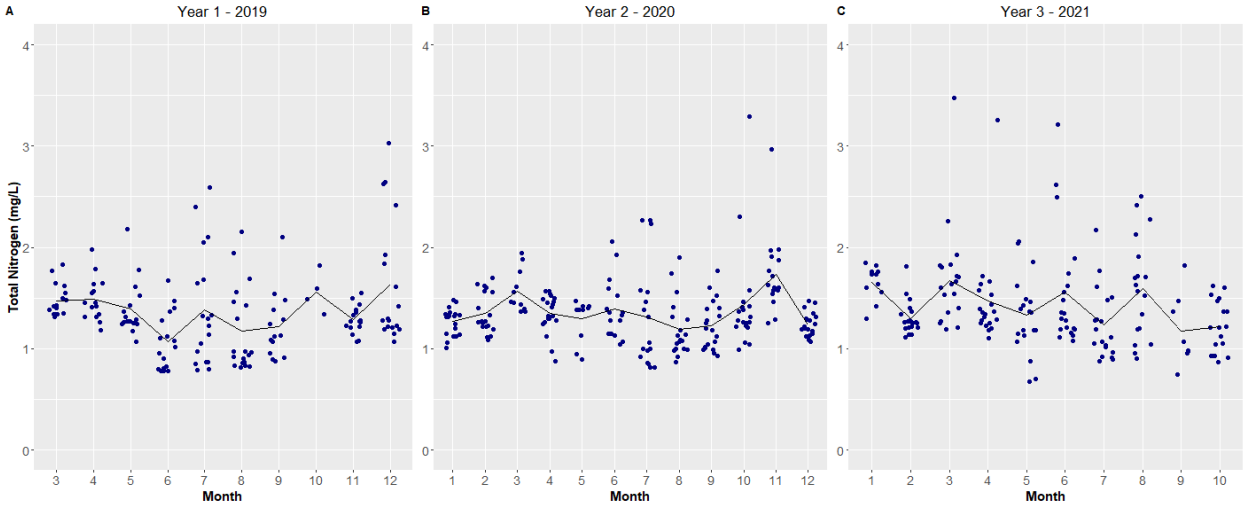

Figure 32. Scatterplot of total nitrogen concentrations (mg/L) over the sampling period. The black line depicts the average concentration per month across the years.

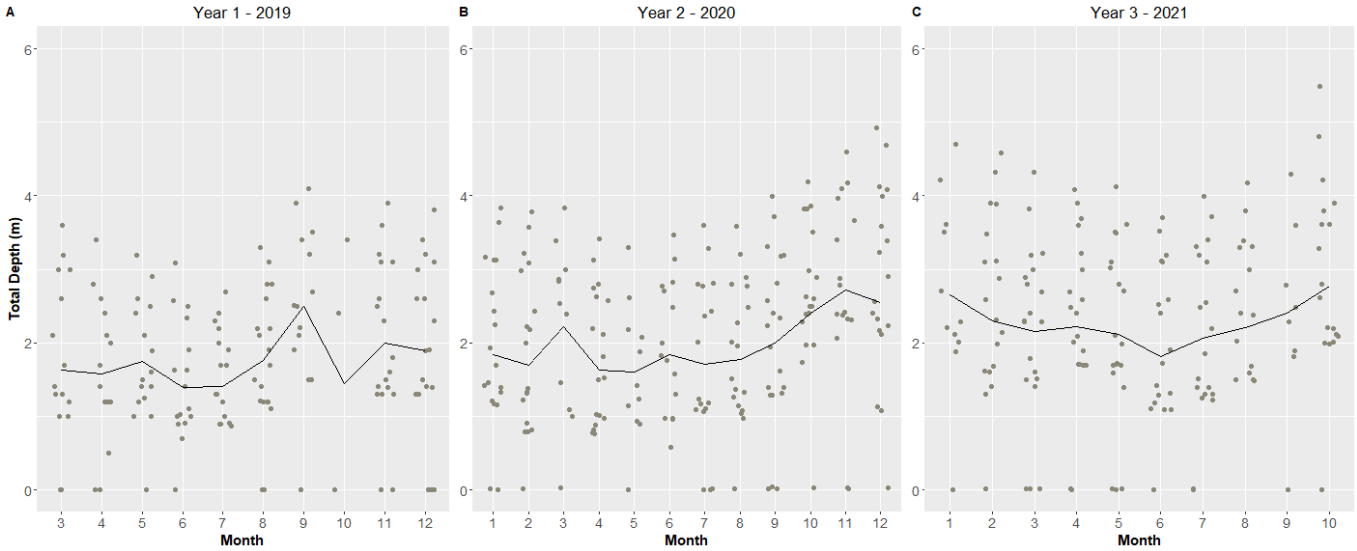

Figure 33. Scatterplot of the total depth (m) of the lake over the sampling period. The black line depicts the average depth per month across the years.

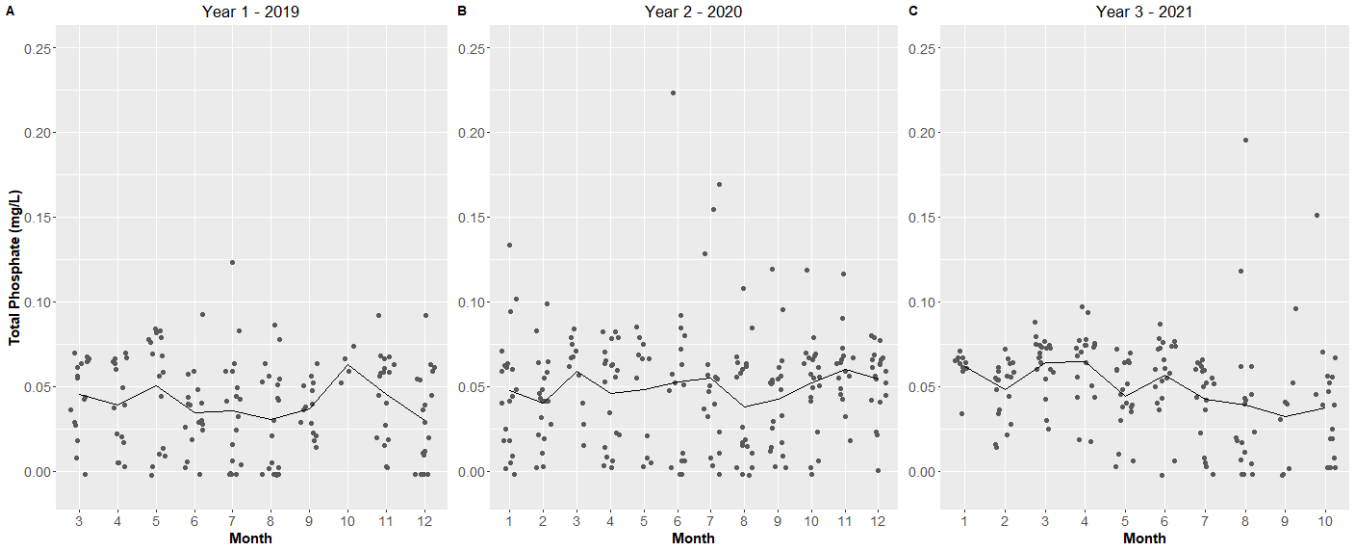

Figure 34. Scatterplot of the total phosphate (mg/L) concentration over the sampling period. The black line depicts the average concentration per month across the years.

DISCUSSION

Bloom effects on microbial community diversity

Most of the cyanobacterial harmful algal bloom (cyanoHAB) research done on Lake Okeechobee (Lake O) primarily focuses on bloom management via the control of nutrients going into the lake. However, there is a growing amount of research suggesting that nutrient levels may not be the only factor influencing these blooms to occur so frequently (Wilhelm *et al.*, 2020). There have not been many studies done on Lake O that assess how these cyanoHABs are affecting the other microbial communities within the lake during these blooms or how these other microbes could be influencing the blooms. The conclusions reached in this study provide a glimpse into the effects of cyanoHABs caused by *Microcystis* may have on the microbial community make-up within Lake O.

This study has found that the diversity of microbial communities in Lake O are affected by the occurrence of *Microcystis*, one of the main cyanobacteria genera causing cyanoHABs both in Lake O and around the world. The microbial communities within Lake O appeared to show both temporal and spatial differences in diversity. However, more significant differences were found between stations and ecological zones within all three years together and between each year. This result was expected due to the different environmental conditions experienced by the ecological zones found throughout the lake. *Microcystis* is known to "lie-in-wait" for the proper environmental conditions that are favorable for their populations to proliferate and bloom; they even tend to overwinter in the sediments at the bottom of the lake until these conditions are present (Cai *et al.*, 2021; Reynolds, 1973). Over the three sampling years (2019-2021), there was an evident increase in bloom intensity and longevity. The peak average relative abundance of *Microcystis* and the average concentration of microcystin could be seen increasing over the years with year 3 (2021) experiencing the highest abundance and concentration (Figures 27 and 26, respectively). There were also changes in environmental conditions within 2021 that may have contributed to the increase of bloom intensity. For instance, 2021 was seen to have warmer average temperatures and a lower TN:TP ratio during the months (May to July) that blooms occurred (Figures 30 and 31, respectively). Numerous studies have shown that cyanobacteria favor higher temperatures thus increasing their growth rates during warmer periods of the year (Wilhelm *et al.*, 2020; Paerl & Hulsman, 2008; Jöhnk K. D., *et al.*, 2008; Reynolds, 2006). Xie *et al.* (2003)

uncovered that when *Microcystis* populations were exposed to sufficient amounts of nitrogen (N) but differing amounts of phosphorus (P), *Microcystis* blooms occurred only in the environments with higher P concentrations. However, as these blooms progressed, both N and P concentrations declined, hence resulting in lower TN:TP ratios. Therefore, as an increase in temperature influences the growth of *Microcystis* blooms, there is a decrease in TN:TP ratio due to the increases use of the nutrients in the water column.

Beta diversity patterns of the microbial community composition

There were some evident spatial patterns throughout the data. The spatial variables of interest in this study were the monitoring stations in the lake and the ecological zones of the lake. When looking at the ecological zones of the lake, there was an obvious coupling between the zones: the inflow zone was always coupled with the zone S79, and the pelagic zone was always coupled with the nearshore zone; giving the idea that these couples have similar microbial community composition. As mentioned in a previous study, although these zones exhibit differing physiochemical properties, these zones do not have clearly defined borders between them, hence these zones can be dynamic (Krausfeldt *et al.*, *submitted*). The results of this study further supported this concept as 2020 (year 2) showed no significant differences between zone when 2019 and 2021 (year 1 and year 3, respectively) did show significant differences; showing that there was less of a differentiation between zones in 2020 compared to the other years. However, the members of each coupling did not come to a surprise as the zone S79 is within the Caloosahatchee River, which has a mouth into the lake, so it is in contact with the inflow zone of the lake. Additionally, the pelagic and nearshore zones also come into contact with one another despite their physiochemical differences.

Rare microbial taxa in Lake Okeechobee

The taxonomic make-up of Lake O was dominated primarily by four common bacterial phyla: Proteobacteria, Bacteroidota, Cyanobacteria, and Actinobacteriota (Table 1, Figure 3). These phyla appeared to change in distribution, along with the less-dominant taxa present, both temporally (Figure 3) and spatially (Figures 5-7). However, there were some phyla that irregular in both their distribution around the lake and their presence across the years. In 2019 (year 1), there was one phylum that appeared in the top phyla of only two stations within Lake O and was found in no other year—SAR324 (marine_clade group B). SAR324 is a novel phylum that has been

recently classified as its own phylum after initially being classified as "marine_clade group B" under the phylum Deltaproteobacteria (Malfertheiner *et al.*, 2022; Parks *et al.*, 2018; Pommier *et al.*, 2005). SAR324 is known to be present only in marine environments; however, Malfertheiner and colleagues (2022) discovered that this phylum can also be found in terrestrial aquifers. (Malfertheiner *et al.*, 2022) Lake O could possibly be subjected to saltwater intrusion (Prinos, 2016; Barlow & Reichard, 2010), or the movement of seawater into freshwater aquifers, due to the water level being heavily managed. The SFWMD stated that saltwater intrusion is at a higher risk of occurring in Lake O starting at a depth of 10½ feet (or 3.2 meters) and compromising the Caloosahatchee lock at a starting depth of 9½ feet (or 2.9 meters) (SFWMD, "Impacts of Operating Lake Okeechobee at Lower Water Levels"). Yet, throughout the majority of 2019, the total depth of Lake O was sustained between about 1 and 3 meters (3.3 feet and 9.8 feet). These conditions put Lake O in the position of the increased risk of saltwater intrusion, especially at the Caloosahatchee River lock (station S79). Coincidentally, SAR324 appears as one of the dominant taxa in stations S79 and POLESOUT (Figure S2); thus, whether SAR324 appears due to saltwater intrusion, or it is naturally occurring in the terrestrial aquifer is unknown.

A non-ubiquitous phylum that was found in 2020 and no other year was Armatimonadota (Figure S3). This phylum was part of the top phyla within the station, KISSR0.0, which is located in the inflow zone and the mouth of the Kissimmee River (Figure 1). Armatimonadota was originally known as candidate phylum OP10 before its reclassification into a new phylum by Hugenholtz and colleagues in 1998 (Hugenholtz *et al.*, 1998b). Isolated sequences of Armatimonadota were isolated from a variety of environments such as aerobic and anaerobic wastewater treatment processes, contaminated and regular soil and sediments (Im *et al.*, 2012). Lake O and its connecting rivers, St. Lucie, Kissimmee, Caloosahatchee, etc. all are experiencing nutrient pollution due to the agricultural and urban lands surrounding them. Furthermore, between 2019 and 2020, there was an increase in the average concentrations of total phosphate (Figure 34), total nitrogen (Figure 32), nitrate + nitrite (Figure 28), and total phosphorus (Figure 23). Hence, it is unknown what kind of contamination occurred during the initial collection and isolation of the bacteria belonging to Armatimonadota, but there may be a connection with the increase in nutrient pollution and the presence of this phyla.

An additional non-ubiquitous phylum, Patescibacteria, appeared only in 2021 at two stations within the lake (Figure S4). Patescibacteria, formerly known as the 'candidate phyla radiation'(CPR), included the discovery of an immense microbial diversion within the bacterial tree of life in 2016 (Herrman *et al.*, 2019). However, in 2018, Parks *et al.* (2018) suggested classifying the CPR into a new phylum, Patescibacteria. There are 14 classes of bacteria known so far in this phylum and they all inhabit a range of environments including groundwater and other aquifer environments, freshwater sediments, and deep-sea sediments (Herrman *et al.*, 2019; Proctor *et al.*, 2018; Leon-Zayas *et al.*, 2017; Luef *et al.*, 2015; Brown *et al.*, 2015). There is a high abundance of Patescibacteria that found in groundwater environments—making up around 38% of the total microbiomes (Herrmann *et al.*, 2019; Bruno *et al.*, 2017; Kumar *et al.*, 2017). In Lake O, Patescibacteria were found only in 2021 (year 3) at two stations, L004 and L006, both of which are in the pelagic zone of the lake. The pelagic zone is the deepest part of the lake but also experiences the most turbidity (Krausfeldt *et al., submitted*). The higher turbidity and reduced water clarity of the water column suggests that there may be sediment resuspension occurring within the pelagic zone (Krausfeldt *et al., submitted*), thus possibly allowing this phylum to be collected in surface waters.

Bacterial co-occurrences with Microcystis

It is well-known that *Microcystis* blooms are influenced by abiotic factors such as environmental variables and nutrient inputs of freshwater ecosystems. There has been increasing curiosity of how the heterotrophic bacterial community plays a role in the aggregation and proliferation of the colonies and how they could be maintaining these cyanobacterial harmful algal blooms (cyanoHABs) created by *Microcystis*. Studies have shown evidence that there are heterotrophic bacteria that live within and surrounding *Microcystis* colonies, with either mutualistic or antagonistic effects (Tu *et al.*, 2019; Shen *et al.*, 2011; Shi *et al.*, 2009; Maruyama *et al.*, 2003; Imamura *et al.*, 2001; Pankow, 1986). As mentioned previously, several results in this study suggested that *Microcystis* can alter the microbial community of Lake O through cyanoHABs. Both *Microcystis* and its related toxin, microcystin, showed strong negative correlations to species evenness and species diversity (Figure 8). In year 3 (2021)—the year with the most intense blooms of the entire sampling period—*Microcystis* appeared as one of the strongest correlated variables, along with other environmental variables, to drive variation in the

microbial communities in Lake O (Figure 21). After revealing that *Microcystis* can alter the microbial communities, the curiosity of knowing who else can possibly be changing with *Microcystis* resulted in the creation of a co-occurrence network involving any bacteria that has appeared with this genus. The co-occurrence network showed 22 significantly strong positive correlations between *Microcystis* and other heterotrophic bacteria; with two exceptions being cyanobacteria *(Pseudanabaena_PCC-7429* and *Snowella_OTU37S04*) (Figure 22). Although some negative correlations did exist between *Microcystis* and other bacteria, their relationships were not strong enough to document as strong correlations (R^2 = -0.7 or less).

Some of the heterotrophic bacteria genera that co-occur with *Microcystis* may indicate that there is a commensal relationship between them. *Bradymonadales* belongs to the phylum Desulfobacterota which is located under the phylum Deltaproteobacteria. *Bradymonadales* are predatory bacteria, which is broken up into two categories, obligatory and facultative (Mu *et al.*; 2020). Mu and colleagues (2020) found that *Bradymonadales* displays unique living strategies that allow for these bacteria to present a novel method of predation: a transition between being obligate and facultative predators. Some of the main bacteria that are highly preyed on by *Bradymonadales* include Bacteroidetes, Flavobacteria, and Proteobacteria. Intriguingly, 11 of the 22 co-occurring bacteria with *Microcystis* belong to the phylum Proteobacteria with an additional two belonging to Bacteroidetes and Flavobacteria. Thus, *Bradymonadales* may be utilizing *Microcystis* colonies during the blooms as a feeding ground for its prey items. *Bdellovibrio exovorus* is another predatory bacteria species that was seen to co-exist with *Microcystis*. First described in 1963 (Koval *et al.*, 2013; Stolp & Starr, 1963), *Bdellovibrio exovorus* belongs to a group of like predatory bacteria known as Bdellovibrio and like organisms (BALOs) (Ezzedine *et al.*, 2020). BALOs were the first records of predatory bacteria and continue to be used as a baseline for the discovery of novel predatory bacteria like *Bradymonadales* which was previously mentioned above. Similar to *Bradymonadales*, *B. exovorus* is also obligatory predators on primarily other Proteobacteria. However, it is important to note that some species of BALOs have been found to kill cyanobacterial cells. Caiola and Pellegrini (1984) found that BALOs were able to lyse *Microcystis aeruginosa* cells via penetration and proposed that these and other algicidal bacteria could be the reason for the dying out of cyanobacteria bloom events.

There were only two taxa that were not heterotrophic bacteria that shared strong positive correlations with *Microcystis*, genera *Pseudanabaena_PCC-7429* and *Snowella_OTU37S04*, which are also part of the phylum Cyanobacteria. The genus *Pseudanabaena* is an epiphytic cyanobacterial taxon that is commonly found embedded within or attached to the mucilaginous sheath of *Microcystis* colonies (Li *et al.*, 2020). Both taxa are frequently observed to be highly correlated during cyanoHABs and this study also provides evidence of this pattern (Li *et al.*, 2020; Berry *et al.*, 2017; Ilhe, 2008). In the 1980s, *Pseudanabaena* was primarily described as a parasitic organism to *Microcystis* colonies (Chang, 1985; Gorham *et al.*, 1982). Further investigation was conducted regarding the interactions between *Pseudanabaena* and *Microcystis*, which investigated the interaction directly (Agha *et al.*, 2016). Agha and colleagues (2016) discovered that *Pseudanabaena* is not selective on the species of *Microcystis* but on their mucilage structure. They also uncovered that *Pseudanabaena* is detrimental to *Microcystis* colonies both directly via cell lysis and indirectly via cell sedimentation. Thus, it may be possible that *Pseudanabaena* may also contribute to the dying out of cyanoHAB events. Conversely, although the genus *Snowella* was also found to be highly correlated to *Microcystis* in a previous study, not much is known about their ecology and their interaction with *Microcystis* (Mankiewicz-Boczek & Font-Nájera, 2022).

Another interesting taxa that was highly correlated with *Microcystis* is the genera *env.OP_17* (Figure 22). There is not much information solely about the bacterium *env.OP_17*, however, it is part of the order Sphingobacteriales and this order is known to be potential algicidal bacteria that favor the uptake of cyanobacterial excretions and decaying material (Mankiewicz-Boczek & Font-Nájera, 2022). Furthermore, Mankiewicz-Boczek & Font-Nájera (2022) found that *env. OP_17* increased in abundance after a bloom, suggesting that this taxon might be a part of the "clean-up team" once a cyanoHAB dies out. Though this study presented results focused primarily on the highly correlated relationships between other bacteria and *Microcystis* in Lake O, there was another bacterial genus, *Streptomyces*, that is known to exhibit algicidal activity towards *Microcystis* that was present in microbial community of Lake O (Zhang *et al.*, 2023). On the contrary, the genus *Phenylobacterium*—another taxon that was found with a high correlation with *Microcystis* (Figure 22)—was found to aid in the growth and dominance of toxic *Microcystis* strains during cyanoHAB events. As mentioned previously, there are toxic and non-toxic bloom-forming strains of *Microcystis* and in a study conducted by Zuo *et al.* (2021), they saw that *Phenylobacterium* was one of the few genera that strongly positively co-existed with

toxic strains of *Microcystis*. After further investigation in the field and in the laboratory, they found that there were three strains of *Phenylobacterium* that promoted the growth of these toxic strains of *Microcystis*, suggesting that *Phenylobacterium* may be a heterotrophic bacterium that could be aiding in the longevity of these blooms (Zuo *et al.*, 2021). Unfortunately, there needs to be further investigation into the mechanisms by which *Phenylobacterium* interact with these toxic strains of *Microcystis* that allow *Microcystis* to remain dominant throughout the cyanoHAB event.

Microcystis, temperature, pH, and nutrients

Although it is also important to investigate the biotic factors that influence cyanoHABs, such as the interactions between the blooming cyanobacteria and other microbes, there is still plenty of evidence of how abiotic factors influence cyanoHABs, and vice versa, all over the world. During this study, in addition to characterizing the microbial community of the lake, certain environmental variables were also collected to consider how these variables could be influencing these blooms along with the microbial community. Besides nutrient levels in the lake, one important physical characteristic that affects cyanoHABs is temperature. Temperature affects the growth of cyanobacterial species. In general, higher temperatures promote the growth of cyanobacteria, often temperatures that are above 25°C (Paerl & Huisman, 2008; Jöhnk *et al.*,2008; Reynolds, 2006). When temperatures increase, the water column becomes more stable and stratified since the increase in temperature weakens the amount of vertical mixing in the water column (Paerl & Huisman, 2008; Paerl & Fulton III, 2006; Reynolds, 2006; Husiman, Matthijs, & Visser, 2005). *Microcystis aeruginosa*, the dominant bloom-forming cyanobacteria species in Lake O, can take advantage of these more stratified conditions using their gas vesicles. The gas vesicles formed by *M. aeruginosa* give them the buoyancy they need to effectively migrate through the water column during favorable conditions, such as high temperatures and increased light availability (Dick *et al.*, 2021; Huisman *et al.*, 2018; Komárek, 2003). This buoyancy also provides *M. aeruginosa* the ability to form "mats" of biomass at the surface of the water; hence, cyanoHAB events tend to increase in frequency in the summer (You *et al.*, 2017; Litchman *et al.*, 2010). Across the sampling period, especially in 2021, temperatures reached between 25°C and 30°C each year from May through to September—around the same months where microcystin concentrations (Figure 26) and *Microcystis* relative abundances (Figure 27) were the highest (Figure 30). Certainly, global warming is becoming a concerning topic as increasing temperatures

are affecting the various environments of the planet. Further research should be done on Lake O and other lakes affected by cyanoHABs to look at the trend of bloom frequencies as the global temperature continues to rise over time.

In addition to increased water temperatures, pH is also known to be a factor associated with *Microcystis* blooms. This importance was evident as pH was included as an environmental factor driving the differences found in the microbial community composition across the sampling period (Figure 17). During a dense bloom, the cyanobacteria rapidly consume inorganic carbon (in the form of dissolved CO_2) that is available in the upper water column, in turn increasing the pH of the surface water to above 9 (Ji *et al.*, 2020; Wilhelm *et al.*, 2020). Across the sampling period, there were an increasing number of instances where the surface water pH was measured above 9 (Figure 29). With this increase in pH, the equilibrium of carbon in the water is shifted from inorganic carbon (dissolved CO_2) to bicarbonate (HCO_3^-) and carbonate (CO_3^{2-}) (Ji *et al.*, 2020; Huisman *et al.*, 2018). *Microcystis*, although also adaptive to high concentrations of CO_2 concentrations, can utilize bicarbonate as a carbon source through the use of carbonic anhydrase found in cyanobacteria—further allowing these blooms to thrive during these alkaline conditions (Ji *et al.*, 2020; Wilhelm *et al.*, 2020; Huisman *et al.*, 2018). Alkaline pH conditions also allow for the conversion of ammonium ions (NH_4^+) to ammonia (NH_3). During the months where microcystin concentrations (Figure 26) and *Microcystis* relative abundances (Figure 27) were the highest (May to September), there was also an increase in ammonia during those months.

CONCLUSION

This study provides a glimpse into the effects of cyanoHABs within the microbial community of the Floridian freshwater lake, Lake Okeechobee. This study provides an initial look into the taxonomic classification of the dynamic microbial community of Lake O over several years and the spatial changes that were seen within these communities. We found that the cyanoHABs that have been commonly occurring in Lake O do in fact alter the microbial community composition of the lake. Further investigation of these changes within the microbial community composition yielded the identification of possible relationships between these microbial communities and *Microcystis*. With the identification of these possible relationships, future investigation should be conducted to see how the functions of these taxa are incorporated into their interaction with *Microcystis*. With that, we might be able to identify bacteria that may serve as possible bioindicators for these cyanoHAB events and aid in preventing or managing these recurring blooms in the lake.

Lake Okeechobee is indeed an essential part of south Florida's ecosystems as it serves as a source of drinking water for nearby towns, irrigation for the agricultural lands surrounding the border of the lake, critical water supply for the environment, and as habitat for various organisms in the water and on the land (South Florida Water Management District (SFWMD)). With the degrading water quality of the lake, there is great concern for life both within and around the lake. To date, numerous studies have been conducted on reducing the nutrient loading into the lake (Canfield Jr. *et al.*, 2021; Schelske, 1989; Canfield Jr. & Hoyer, 1988) and investigating the possible control of these recurring blooms (Pokrzywinski *et al.*, 2022), primarily focusing on the cyanobacteria involved in these blooms. Not many studies have been done on Lake Okeechobee that explore the taxonomic structure, temporal distributions, and spatial distributions of the microbial communities before, during, and after annual cyanoHABs. Furthermore, whether the microbial community taxonomic structure, temporal and spatial distributions rebound after a bloom event also has yet to be studied.

To enable scientists to enhance their comprehension of the ongoing cyanoHABs in Lake Okeechobee and their interactions with the surrounding environment, particularly the microbial community, it is essential to fill these existing knowledge gaps. With that, scientists will be able to examine the variations in the diversity and trophic structure of the lake before, during, and after

the occurrence of these harmful blooms—bringing scientists closer to fully understanding the impact of cyanoHABs on Lake Okeechobee's microbial communities.

REFERENCES

Agha, R., Del Mar Labrador, M., De Los Ríos, A., & Quesada, A.. (2016). Selectivity and detrimental effects of epiphytic *Pseudanabaena* on *Microcystis* colonies. *Hydrobiologia*, *777*(1), 139–148. doi:10.1007/s10750-016-2773-z

Anderson, D. M. (2009). Approaches to monitoring, control, and management of harmful algal blooms (HABs). *Ocean Coast Manag.* doi:10.1016/j.ocecoaman.2009.04.006

Barlow, P., & Reichard, E. (2010). Saltwater intrusion in coastal regions of North America. *Hydrogeology Journal*, *18*, 247–260. doi:10.1007/s10040-009-0514-3

Berry, M. A., Davis, T. W., Cory, R. M., Duhaime, M. B., Johengen, T. H., Kling, G. W., Marino, J. A., DeuUyl, P. A., Gossiaux, D., Dick, G. J. & Denef, V. J. (2017). Cyanobacterial harmful algal blooms are a biological disturbance to western Lake Erie bacterial communities. *Environ. Microbiol*. 19:1149-62.

Bláha, L., Babica, P., & Maršálek, B. (2009). Toxins produced in cyanobacterial water blooms - toxicity and risks. *Interdisc. Toxicol., 2*. doi:10.2478/v10102-009-0006-2

Bolyen, E., Rideout, J.R., Dillon, M.R. *et al.* (2019). Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2. *Nat Biotechnol* **37**, 852–857. doi:10.1038/s41587-019-0209-9

Bowling, L. (1994). Occurrence and possible causes of a severe cyanobacterial bloom in Lake Cargelligo, New South Wales. *Mar. Freshw. Res., 45*(5). doi:10.1071/MF9940737

Brown C. T., Hug L. A., Thomas B. C., Sharon I., Castelle C. J., Singh A., *et al.* (2015). Unusual biology across a group comprising more than 15% of domain Bacteria. *Nature*. 523:208. Doi: 10.1038/nature14486

Bruno A., Sandionigi A., Rizzi E., Bernasconi M., Vicario S., Galimberti A., *et al.* (2017). Exploring the under-investigated “microbial dark matter” of drinking water treatment plants. *Sci Rep*. 7:44350. Doi: 10.1038/srep44350

Byrne, S., Butler, C. A., Reynolds, E. C., & Dashper, S. G. (2018). Chapter 7 - Taxonomy of Oral Bacteria. *Methods in Microbiology, 45*. doi:10.1016/bs.mim.2018.07.001

Cai, P.; Cai, Q.; He, F.; Huang, Y.; Tian, C.;Wu, X.;Wang, C.; Xiao, B. (2021). Flexibility of *Microcystis* Overwintering Strategy in Response to Winter Temperatures. *Microorganisms*. 9, 2278. doi:10.3390/microorganisms9112278

Caiola, M.G., and Pellegrini, S. (1984) Lysis of *Microcystis aeruginosa* (Kutz.) by Bdellovibrio-like Bacteria1. *J Phycol* 20: 471–475.

Campbell, A. M., Fleisher, J., Sinigalliano, C., White, J. R., & Lopez, J. V. (2015). Dynamics of marine bacterial community diversity of the coastal waters of the reefs, inlets, and wastewater outfalls of southeast Florida. *Microbiology Open*, *4*(3), 390–408. doi:10.1002/mbo3.245

Canfield, D., & Hoyer, M. (1988). The Eutrophication of Lake Okeechobee. *Lake and Reservoir Management*.

Canfield Jr. D. E., Bachmann, R. W. & Hoyer, M. V. (2021) Restoration of Lake Okeechobee, Florida: mission impossible?, *Lake and Reservoir Management*, 37:1, 95-111, doi: 10.1080/10402381.2020.1839607

Chang, T.-P. (1985). Selective inhabitation of parasitice Cyanophyte *Pseudanabaena* in water-bloom *Microcystis* colonies. *Arch. Hydrobiol.*

Chapman, R. L. (2013). Algae: the world's most important "plants"—an introduction. *Mitig. Adapt. Strateg. Glob. Change, 18,* 5-12. doi:10.1007/s11027-010-9255-9.

Cuklina, J., Lee, C. H., Williams, E. G., Sajic, T., Collins, B. C., Rodriguez Martinez, M., . . . Pedrioli, P. G. (2021). Diagnostics and correction of batch effects in large-scale proteomic studies: a tutorial. *Molecular Systems Biology*(17). doi:10.15252/msb.202110240

Dick, G.J. (2021). The genetic and ecophysiological diversity of *Microcystis*. *Environ. Microbiol*. doi:10.1111/1462-2920.15615

Donnelly, C.P. 2018. *Microbial Ecology of South Florida Surface Waters: Examining the Potential for Anthropogenic Influences.* Master's thesis. Nova Southeastern University.

Dubnau, D., Smith, I., Morell, P., & Marmur, J. (1965). Gene conservation in Bacillus species. I. Conserved genetic and nucleic acid base sequence homologies. *Proc Natl Acad Sci U S A., 54*. doi:10.1073/pnas.54.2.491

Easson, C. G., & Lopez, J. V. (2019). Depth-Dependent Environmental Drivers of Microbial Plankton Community Structure in the Northern Gulf of Mexico. *Frontiers in microbiology*, *9*, 3175. doi:10.3389/fmicb.2018.03175

Eiler A, Bertilsson S. (2004). Composition of freshwater bacterial communities associated with cyanobacterial blooms in four Swedish lakes. *Environ Microbiol* **6**: 1228–1243.

Ezzedine, J. A., Desdevises, Y., & Jacquet, S. (2022). *Bdellovibrio* and like organisms: current understanding and knowledge gaps of the smallest cellular hunters of the microbial world. *Critical reviews in microbiology*, *48*(4), 428–449. doi:10.1080/1040841X.2021.1979464

Facey, J. A., Apte, S. C., & Mitrovic, S. M. (2019). A Review of the Effect of Trace Metals on Freshwater Cyanobacterial Growth and Toxin Production. *Toxins, 11*. doi:10.3390/toxins11110643

Freed, L.L. (2018). Characterization of the bioluminescent symbionts from ceratioids collected in the Gulf of Mexico. Masters thesis. Halmos College of Natural Sciences and Oceanography, Nova Southeastern University.

Gaysina, L. A., Saraf, A., and Singh, P. (2019) Chapter 1 - Cyanobacteria in Diverse Habitats. Academic Press. doi: 10.1016/B978-0-12-814667-5.00001-5.

Gorham, P., S. McNicholas & E. D. Allen. (1982). Problems encountered in searching for new strains of toxic planktonic cyanobacteria. *South African Journal of Science*. 78: 357.

Harke, M. J. *et al.* (2016). A review of the global ecology, genomics, and biogeography of the toxic cyanobacterium *Microcystis* spp. *Harmful Algae* 54, 4–20. doi: 10. 1016/j. hal. 2015. 12. 007.

Harrell Jr, F. (2023). _Hmisc: Harrell Miscellaneous_. R package version 5.0-1. https://CRAN.R-project.org/package=Hmisc.

Havens, KE. (2007). Cyanobacteria blooms: effects on aquatic ecosystems. In: Hudnell KH (ed). *Cyanobacterial Harmful Algal Blooms: State of the Science and Research*, vol. 619. Springer: New York, pp 675–732.

Herrmann, M., Wegner, C. E., Taubert, M., Geesink, P., Lehmann, K., Yan, L., Lehmann, R., Totsche, K. U., & Küsel, K. (2019). Predominance of *Cand*. Patescibacteria in Groundwater Is Caused by Their Preferential Mobilization From Soils and Flourishing Under Oligotrophic Conditions. *Frontiers in microbiology*, *10*, 1407. doi:10.3389/fmicb.2019.01407

Hugenholtz P, Pitulle C, Hershberger KL, Pace NR (1998) Novel division level bacterial diversity in a yellowstone hot spring. *J Bacteriol* 180:366–376

Huisman, J. M., Matthijs, H. C. P., & Visser, P. M. (2005). Harmful Cyanobacteria Springer Aquatic Ecology Series 3. *Dordrecht, The Netheralands*.

Huisman, J., Codd, G. A., Paerl, H. W., Ibelings, B. W., Verspagen, J. M., & Visser, P. M. (2018). Cyanobacterial blooms. *Nature Reviews Microbiology*, *16*(8), 471-483.

Ilhe, T. (2008). The Spatiotemporal Variation of *Microcystis* spp. (Cyanophyceae) and Microcystins in Quitzdorf reservoir (Sachsen). Die raum-zeitliche Variation von *Microcystis* spp. (Cyanophyceae) und Microcystinen in der Talsperre Quitzdorf (Sachsen). Ph.D. dissertation. Universität, Dresden, Germany.

Im, W.-T., Hu, Z.-Y., Kim, K.-H., Rhee, S.-K., Meng, H., Lee, S.-T., & Quan, Z.-X. (2012). Description of Fimbriimonas ginsengisoli gen. nov., sp. nov. within the Fimbriimonadia class nov., of the phylum Armatimonadetes. *Antonie van Leeuwenhoek*. doi:10.1007/s10482-012-9739-6

Imamura, N., Motoike, I., Shimada, N., Nishikori, M., Morisaki, H., & Fukami, H. (2001). An Efficient Screening Approach for Anti-*Microcystis* Compounds: Based on Knowledge of Aquatic Microbial Ecosystem. *The Journal of Antibiotics*.

J. Greg Caporaso, G. A.-L. (2018). EMP 16S Illumina Amplicon Protocol. *PLOS One*. doi:10.17504/protocols.io.nuudeww

Ji X, Verspagen JMH, Van de Waal DB, Rost B, Huisman J. (2020). Phenotypic plasticity of carbon fixation stimulates cyanobacterial blooms at elevated CO2. Sci Adv 6: eaax2926. doi:10.1126/sciadv.aax2926.

Jöhnk, K.D., Huisman, J., Sharples, J., Sommeijer, B., Visser, P.M. And Stroom, J.M. (2008), Summer heatwaves promote blooms of harmful cyanobacteria. *Global Change Biology*, 14: 495-512. doi:10.1111/j.1365-2486.2007.01510.x

Karns, R. C. 2017. *Microbial Community Richness Distinguishes Shark Species Microbiomes in South Florida.* Master's thesis. Nova Southeastern University.

Kolmonen, E., Sivonen, K., Rapala, J., & Haukka, K. (2004). Diversity of cyanobacteria and heterotrophic bacteria in cyanobacterial blooms in Lake Joutikas, Finland. *Aquatic Microbial Ecology, 36*.

Komárek, J. (2003) Coccoid and colonial Cyanobacteria. Freshwater Algae of North America. Amsterdam: *Elsevier*,pp. 59–116.

Koval, S.F., Hynes, S.H., Flannagan, R.S., Pasternak, Z., Davidov, Y., and Jurkevitch, E. (2013) *Bdellovibrio exovorus* sp. nov., a novel predator of Caulobacter crescentus. *Int J Syst Evol Microbiol*. 63: 146–151.

Krausfeldt, L. E., Shmakova, E., Lee, H., Mazzei, V., Loftin, K. A., Smith, R. P., . . . Lopez, J. V. (submitted). Microbial biodiversity and phage-host interactions are linked to the occurrence of cyanobacterial blooms.

Kumar S., Herrmann M., Thamdrup B., Schwab V. F., Geesink P., Trumbore S. E., *et al.* (2017). Nitrogen loss from pristine carbonate-rock aquifers of the Hainich Critical Zone Exploratory (Germany) is primarily driven by chemolithoautotrophic anammox processes. *Front. Microbiol*. 8:1951. doi: 10.3389/fmicb.2017.01951

Lahti, L. *et al.* microbiome R package. URL: http://microbiome.github.io

Lande, R. (1996). Statistics and Partitioning of Species Diversity, and Similarity among Multiple Communities. *Oikos*, *76*(1), 5–13. doi: 10.2307/3545743

Larkin, S. L., & Adams, C. M. (2007). Harmful Algal Blooms and Coastal Business: Economic Consequences in Florida. *Society and Natural Resources, 20*. doi:10.1080/08941920601171683

Larsson, J. (2022). _eulerr: Area-Proportional Euler and Venn Diagrams with Ellipses_. R package version 7.0.0. https://CRAN.R-project.org/package=eulerr.

Lecher, A. L. (2021). A Brief History of Lake Okeechobee: A Narrative of Conflict. *Journal of Floria Studies, 1*(9). Retrieved from https://www.journaloffloridastudies.org/files/vol0109/lecher-brief-history-lake-okeechobee.pdf

Léon-Zayas R., Peoples L., Biddle J. F., Podell S., Novotny M., Cameron J., *et al.* (2017). The metabolic potential of the single cell genomes obtained from the Challenger Deep, Mariana Trench within the candidate superphylum Parcubacteria (OD1). *Environ. Microbiol. 19.* 2769–2784. doi: 10.1111/1462-2920.13789.

Li, Z. K., Dai, G. Z., Zhang, Y., Xu, K., Bretherton, L., Finkel, Z. V., Irwin, A. J., Juneau, P., & Qiu, B. S. (2020). Photosynthetic adaptation to light availability shapes the ecological

success of bloom-forming cyanobacterium *Pseudanabaena* to iron limitation. *Journal of phycology*, *56*(6), 1457–1467. doi:10.1111/jpy.13040

Litchman, E., de Tezanos Pinto, P., Klausmeier, C. A., Thomas, M. K., & Yoshiyama, K. (2010). Linking traits to species diversity and community structure in phytoplankton. *Hydrobiologia, 653*, 15-28.

Luef B., Frischkorn K. R., Wrighton K. C., Holman H.-Y. N., Birarda G., Thomas B. C., *et al.* (2015). Diverse uncultivated ultra-small bacterial cells in groundwater. *Nat. Commun*. 6:6372. doi: 10.1038/ncomms7372

Ma, S. (2023). _MMUPHin: Meta-analysis Methods with Uniform Pipeline for Heterogeneity in Microbiome Studies_. R package version 1.12.1.

Malfertheiner, L.; Martínez-Pérez, C.; Zhao, Z.; Herndl, G.J.; Baltar, F. (2022). Phylogeny and Metabolic Potential of the Candidate Phylum SAR324. *Biology, 11*, 599. doi:10.3390/ biology11040599

Mankiewicz-Boczek, J., & Font-Najera, A. (2022). Temporal and functional interrelationships between bacterioplankton communities and the development of a toxigenic *Microcystis* bloom in a lowland European reservoir. *Nature Scientific Reports*. doi:10.1038/s41598-022-23671-2

Markou, G., Vandamme, D., & Muylaert, K.. (2014). Microalgal and cyanobacterial cultivation: The supply of nutrients. *Water Research*, *65*, 186–202. doi: 10.1016/j.watres.2014.07.025

Maruyama T., Kato K., Yokoyama A., Tanaka T., Hiaishi A. & Park H.D. (2003) Dynamics of microcystin degrading bacteria in mucilage of *Microcystis*. *Microbial Ecology*, 46, 279–288.

Mataloni, G., Komarek, J., (2004). *Gloeocapsopsis aurea*, a new subaerophytic cyanobacterium from maritime Antarctica. *Polar Biol*. 27, 623–628.

McMurdie, P.J. and Holmes, S. (2013). An R package for reproducible interactive analysis and graphics of microbiome census data. *PLoS ONE* 8(4):e61217.

McQuaid, A. L. (2019). The Bioaccumulation of Cyanotoxins in Aquatic Food Webs. *Doctoral Dissertations, 2481*. https://scholars.unh.edu/dissertation/2481

Metcalf, J. S., Banack, S. A., Powell, J. T., Tymm, F. J., Murch, S. J., Brand, L. E., & Cox, P. A. (2018). Public health responses to toxic cyanobacterial blooms: perspectives from the 2016 Florida event. *Water Policy, 20*, 919-932. doi:10.2166/wp.2018.012

Missimer, T.M.; Thomas, S.; Rosen, B.H. (2021). Legacy Phosphorus in Lake Okeechobee (Florida, USA) Sediments: A Review and New Perspective. Water, 13, 39. doi:10.3390/w13010039

Mu, DS., Wang, S., Liang, QY. *et al.* (2020). Bradymonabacteria, a novel bacterial predator group with versatile survival strategies in saline environments. *Microbiome* **8**, 126. Doi: 10.1186/s40168-020-00902-0

Myer, M. H., Urquhart, E., Schaeffer, B. A., & Johnston, J. M. (2020). Spatio-Temporal Modeling for Forecasting High-Risk Freshwater Cyanobacterial Harmful Algal Blooms in Florida. *Frontiers in Environmental Science, 8*, 1-13. doi:10.3389/fenvs.2020.581091

O'Connell, L.M., Gao, S., McCorquodale, D.S., Fleisher, J., & Lopez, J.V. (2018). Fine grained compositional analysis of Port Everglades Inlet microbiome using high throughput DNA sequencing. *PeerJ, 6*.

Okello, W., Portmann, C., Erhard, M., Gademann, K. and Kurmayer, R. (2010), Occurrence of microcystin-producing cyanobacteria in Ugandan freshwater habitats. Environ. Toxicol., 25: 367-380. doi:10.1002/tox.20522

Oksanen *et al.* (2022). _vegan: Community Ecology Package_. R package version 2.6-4. https://CRAN.R-project.org/package=vegan

Paerl, H., & Scott, J. (2010). Throwing Fuel on the Fire: Synergistic Effects of Excessive Nitrogen Inputs and Global Warming on Harmful Algal Blooms. *Environ. Sci. Technol., 44*. doi:10.1021/es102665e

Paerl HW, Huisman J. (2008). Blooms like it hot. Science 320:57–58. doi:10.1126/science.1155398.

Paerl, Hans & Fulton, Rolland. (2006). *Ecology of Harmful Cyanobacteria*. doi:10.1007/978-3-540-32210-8_8.

Pankow, H. (1986). About endophytic and epiphytic algae in or on the mucilage envelope of *Microcystis* colonies. *Arch. Protistenkd*. 132, 377–380.

Parks, D., Chuvochina, M., Waite, D., Rinke, C., Skarshewski, A., Chaumeil, P.-A., & Philip, H. (2018). A standardized bacterial taxonomy based on genome phylogeny substantially revises the tree of life. *Nature Biotechnology, 36*. doi:10.1038/nbt.4229

PCR purification with Beckman Coulter AMPure XP magnetic beads and the VIAFLO 96. (2020). Retrieved from INTEGRA: https://www.integra-biosciences.com/global/en/applications/pcr-purification-beckman-coulter-ampure-xp-magnetic-beads-and-viaflo-96#top

Pokrzywinski, K.L.; Bishop, W.M.; Grasso, C.R.; Fernando, B.M.; Sperry, B.P.; Berthold, D.E.; Laughinghouse, H.D., IV; Van Goethem, E.M.; Volk, K.; Heilman, M.; *et al.* (2022). Evaluation of a Peroxide-Based Algaecide for Cyanobacteria Control: A Mesocosm Trial in Lake Okeechobee, FL, USA. *Water*, 14, 169. doi:10.3390/w14020169

Pommier, T., Pinhassi, J., & Hagstrom, A. (2005). Biogeographic analysis of ribosomal RNA clusters from marine bacterioplankton. *Aquatic Microbial Ecology*, *41*(1), 79–89. doi:10.3354/ame041079

Prinos, S. T. (2016). *Saltwater intrusion monitoring in Florida*.

Proctor C. R., Besmer M. D., Langenegger T., Beck K., Walser J.-C., Ackermann M., *et al.* (2018). Phylogenetic clustering of small low nucleic acid-content bacteria across diverse freshwater ecosystems. *ISME J*. 12 1344–1359. doi: 10.1038/s41396-018-0070-78.

Quast C, Pruesse E, Yilmaz P, Gerken J, Schweer T, Yarza P, Peplies J, Glöckner FO (2013) The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. *Nucl. Acids Res*. 41 (D1): D590-D596.

R Core Team. (2022). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Reynolds, C.S. (2006). *Ecology of Phytoplankton*. Cambridge Univ. Press, Cambridge.

Reynolds, C. S. (1973). Growth and buoyancy of *Microcystis aeruginosa* Kütz. emend. Elenkin in a shallow eutrophic lake. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, *184*(1074), 29-50.

Rollwagen-Bollens, G., Lee, T., Rose, V., & Bollens, S. M. (2018). Beyond Eutrophication: Vancouver Lake, WA, USA as a Model System for Assessing Multiple, Interacting Biotic and Abiotic Drivers of Harmful Cyanobacterial Blooms. *Water, 10*. doi:10.3390/w10060757

Rosen, B. H., Davis, T. W., Gobler, C. J., Kramer, B. J., & Loftin, K. A. (2017). *Cyanobacteria of the 2016 Lake Okeechobee and Okeechobee Waterway Harmful Algal Bloom: U.S. Geological Survey Open-File Report 2017–1054.* doi:10.3133/ofr20171054

Sanger, F., Nicklen, S., & Coulson, A. R. (1977). DNA sequencing with chain-temrinating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America, 74*.

Schelske, C. L. (1989). Assessment of Nutrient Effects and Nutrient Limitation in Lake Okeechobee. *Water Resources Bulletin, 25*.

Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, Amin N, Schwikowski B, Ideker T. (2003). Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Research*. 13(*11*):2498-504

Shen, H., Niu, Y., Xie, P., Tao, M., & Yang, X. (2011). Morphological and physiological changes in *Microcystis aeruginosa* as a result of interactions with heterotrophic bacteria. *Freshwater Biology*, *56*, 1065-1080. doi:10.1111/j.1365-2427.2010.02551.x

Shi L., Cai Y., Yang H., Xing P., Li P., Kong L. *et al.* (2009) Phylogenetic diversity and specificity of bacteria associated with *Microcystis aeruginosa* and other cyanobacteria. *Journal of Environmental Sciences (China)*, 21, 1581–1590.

Sigee D. (2005). *Freshwater Microbiology. Biodiversity and Dynamic Interactions of Microorganisms in the Aquatic Environment*. John Wiley & Sons: Chichester, UK, pp 328–338.

Smayda, T. J. (1997). What is a bloom? A commentary. *Limnol. Oceanogr., 42*(5), 1132-1136.

South Florida Water Management District. (n.d.). Retrieved from DBHYDRO: https://my.sfwmd.gov/dbhydroplsql/show_dbkey_info.main_menu

South Florida Water Management District (SFWMD). (n.d.). Lake Okeechobee: In Review. Retrieved from https://www.sfwmd.gov/

South Florida Water Management District. (n.d.). *Impacts of Operating Lake Okeechobee at Lower Water Levels* [Infographic]. SFWMD. https://www.sfwmd.gov/sites/default/files/documents/infographic_lake_okee_depth.pdf

Stolp, H., and Starr, M.P. (1963) Bdellovibrio bacteriovorus gen. et sp. n., a predatory, ectoparasitic, and bacteriolytic microorganism. *Antonie Van Leeuwenhoek*. 29: 217–248.

Stomp, M. *et al.* (2007). Colourful coexistence of red and green picocyanobacteria in lakes and seas. *Ecol. Lett.* 10, 290–298.

Thurkal, A. K. (2017). A REVIEW ON MEASUREMENT OF ALPHA DIVERSITY IN BIOLOGY. *Agric Res J*. doi:10.5958/2395-146X.2017.00001.1

Tian, R., Ning, D., He, Z. *et al.* (2020). Small and mighty: adaptation of superphylum *Patescibacteria* to groundwater environment drives their genome simplicity. *Microbiome* 8, 51. doi:10.1186/s40168-020-00825-w

Tu, J., Chen, L., Gao, S., Zhang, J., Bi, C., Tao, Y., . . . Lu, Z. (2019). Obtaining Genome Sequences of Mutualistic Bacteria in Single *Microcystis* Colonies. *Int. J. Mol. Sci., 20*. doi:10.3390/ijms20205047

U.S. Army Corps of Engineers, J. D. (2021). *Home*. Herbert Hoover Dike. https://www.saj.usace.army.mil/HHD/

US Department of Commerce, N. (n.d.). Florida Dry Season Forecast and El Niño-Southern Oscillation (EÑSO). Www.weather.gov. https://www.weather.gov/mlb/enso_florida_climate_forecast

Van Wichelen, J., Vanormelingen, P., Codd, G. A., & Vyverman, W. (2016). The common bloom-forming cyanobacterium Microcystis is prone to wide array of microbial antagonists. Harmful Algae, 55, 97-111. doi:10.1016/j.hal.2016.02.009

Visser, P., Verspagen, J., Sandrini, G., Stal, L., Matthijs, H., Davis, T., . . . Huisman, J. (2016). How rising CO2 and global warming may stimulate harmful cyanobacterial blooms. *Harmful Algae, 54*.

Wang, K., Mou, X., Cao, H., Struewing, I., Allen, J., & Lu, J. (2021). Co-occurring microorganisms regulate the succession of cyanobacterial harmful algal blooms. *Environmental Pollution, 288*, 117682. doi:10.1016/j.envpol.2021.117682

Whitton, B.A., Potts, M., (2000a). The Ecology of Cyanobacteria. Kluwer Academic Publishers, Dordrecht.

Whitton, B.A., Potts, M., (2000b). Introduction of cyanobacteria. In: Whitton, B.A., Potts, M. (Eds.), The Ecology of Cyanobacteria. Their Diversity in Time and Space. Kluwer Academic, Dordrecht, pp. 1–10.

Wickham, H. (2016). ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York.

Wiegand, C., & Pflugmacher, S. (2005). Ecotoxicological eff ects of selected cyanobacterial secondary metabolites a short review. *Toxicology and Applied Pharmacology, 203*.

Wilhelm, S. W., Bullerjahn, G. S., & McKay, R. M. L. (2020). The Complicated and Confusing Ecology of *Microcystis* Blooms. *MBio*, *11*(3), e00529-20. doi:10.1128/mBio.00529-20

Williams, C. D., Aubel, M. T., Chapman, A. D., & D'Aiuto, P. E. (2007). Identification of cyanobacterial toxins in Florida's freshwater systems. *Lake and Reservoir Management, 23*(2), 144-152. doi:10.1080/07438140709353917

Woese, C. R., & Fox, G. E. (1977). Phylogenetic structure of the prokaryotic domain: The primary kingdoms. *Proc Natl Acad Sci USA*, *74*, 5088-5090. doi:10.1073/pnas.74.11.5088

Xie, L. Q., Xie, P., & Tang, H. J. (2003). Enhancement of dissolved phosphorus release from sediment to lake water by *Microcystis* blooms—an enclosure experiment in a hyper-eutrophic, subtropical Chinese lake. *Environmental Pollution*, *122*(3), 391–399. doi:10.1016/S0269-7491(02)00305-6

You, J., Mallery, K., Hong, J., & Hondzo, M. (2017). Temperature effects on growth and buoyancy of *Microcystis aeruginosa*. *Journal of Plankton Research*, *40*(1), 16–28. doi:10.1093/plankt/fbx059

Zamora-Barrios, C. A., Nandini, S., & Sarma, S. S. (2019). Bioaccumulation of microcystins in seston, zooplankton and fish: A case study in Lake Zumpango, Mexico. *Environmental Pollution, 249*. doi:10.1016/j.envpol.2019.03.029

Zhang, H.; Xie, Y.; Zhang, R.; Zhang, Z.; Hu, X.; Cheng, Y.; Geng, R.; Ma, Z.; Li, R. (2023). Discovery of a High-Efficient Algicidal Bacterium against *Microcystis aeruginosa* Based on Examinations toward Culture Strains and Natural Bloom Samples. *Toxins*, 15, 220. doi:10.3390/toxins15030220

Zheng, Q., Wang, Y., Xie, R., Lang, A., Liu, Y., Lu, J., . . . Nianzhi, J. (2018). Dynamics of Heterotrophic Bacterial Assemblages within Synechococcus Cultures. *Applied and Environmental Microbiology*, *84*(3). doi:10.1128/AEM.01517-17

Zhu, Q., Shi, L., Peng, G., & Fei-shi, L. (2014). High-throughput Sequencing Technology and Its Application. *Journal of Northeast Agricultural University (English Edition)*, 21. doi:10.1016/S1006-8104(14)60073-8

Zuo, Jun & Hu, Lili & Shen, Wei & Zeng, Jiaying & Li, Lin & Gan, Nanqin. (2021). The involvement of α-proteobacteria *Phenylobacterium* in maintaining the dominance of toxic *Microcystis* blooms in Lake Taihu, China. *Environmental Microbiology*. 23. 1066–1078. 10.1111/1462-2920.15301.

APPENDIX

I. Sample read table

Table S1. Final samples and their total amount of sequencing reads.

Sample	# of reads				
CLV10A_1_20	76,624	**L007_5_20**	14,306	**PALMOUT_9_20**	60,268
CLV10A_1_21	46,642	**L007_5_21**	60,799	**PALMOUT_9_21**	17,598
CLV10A_10_19	12,394	**L007_6_19**	25,096	**PELBAY3_1_20**	62,250
CLV10A_10_20	76,075	**L007_6_20**	14,750	**PELBAY3_1_21**	32,522
CLV10A_10_21	27,728	**L007_6_21**	36,790	**PELBAY3_10_20**	72,870
CLV10A_11_19	31,983	**L007_7_19**	38,943	**PELBAY3_10_21**	53,021
CLV10A_12_19	31,518	**L007_7_21**	50,726	**PELBAY3_11_19**	28,589
CLV10A_12_20	46,448	**L007_8_19**	53,470	**PELBAY3_11_20**	49,393
CLV10A_2_20	100,350	**L007_8_20**	36,822	**PELBAY3_12_19**	31,417
CLV10A_2_21	23,458	**L007_8_21**	56,065	**PELBAY3_12_20**	46,412
CLV10A_3_20	17,910	**L007_9_19**	13,578	**PELBAY3_2_20**	105,663
CLV10A_3_21	52,702	**L007_9_20**	81,952	**PELBAY3_2_21**	27,543
CLV10A_4_19	22,167	**L007_9_21**	51,459	**PELBAY3_3_19**	15,933
CLV10A_4_20	23,094	**L008_1_20**	42,067	**PELBAY3_3_21**	43,612
CLV10A_4_21	34,584	**L008_10_20**	71,738	**PELBAY3_4_20**	10,029
CLV10A_5_19	21,015	**L008_10_21**	44,244	**PELBAY3_4_21**	28,973
CLV10A_5_21	39,585	**L008_11_19**	29,332	**PELBAY3_5_19**	60,939
CLV10A_6_19	33,664	**L008_11_20**	60,226	**PELBAY3_5_20**	21,305
CLV10A_6_20	15,985	**L008_12_19**	20,267	**PELBAY3_5_21**	35,182
CLV10A_6_21	53,886	**L008_12_20**	19,467	**PELBAY3_6_19**	50,764
CLV10A_7_19	120,120	**L008_2_20**	58,702	**PELBAY3_6_20**	13,069
CLV10A_7_20	20,116	**L008_2_21**	34,817	**PELBAY3_6_21**	36,587
CLV10A_7_21	55,550	**L008_3_19**	33,247	**PELBAY3_7_19**	39,502
CLV10A_8_19	98,094	**L008_3_20**	21,043	**PELBAY3_7_20**	16,049
CLV10A_8_20	39,276	**L008_3_21**	79,741	**PELBAY3_7_21**	35,714
CLV10A_8_21	46,501	**L008_4_20**	10,088	**PELBAY3_8_19**	43,571
CLV10A_9_19	85,121	**L008_4_21**	38,117	**PELBAY3_8_20**	25,457
CLV10A_9_20	82,088	**L008_5_19**	60,352	**PELBAY3_8_21**	35,761
KISSR0.0_1_20	36,658	**L008_5_20**	11,508	**PELBAY3_9_19**	38,412
KISSR0.0_10_20	98,425	**L008_5_21**	47,189	**PELBAY3_9_20**	71,440
KISSR0.0_10_21	65,812	**L008_6_19**	25,457	**POLE3S_1_20**	30,299
KISSR0.0_11_19	11,587	**L008_6_20**	13,623	**POLE3S_1_21**	31,623
KISSR0.0_11_20	74,182	**L008_6_21**	47,807	**POLE3S_10_20**	73,885
KISSR0.0_12_19	51,148	**L008_7_19**	49,147	**POLE3S_10_21**	53,517
KISSR0.0_12_20	74,553	**L008_7_20**	15,851	**POLE3S_11_20**	36,478
KISSR0.0_2_20	63,076	**L008_7_21**	48,710	**POLE3S_12_19**	24,108
KISSR0.0_2_21	39,407	**L008_8_19**	59,179	**POLE3S_12_20**	31,633
KISSR0.0_3_19	16,094	**L008_8_20**	41,239	**POLE3S_2_20**	35,025
KISSR0.0_3_21	33,783	**L008_8_21**	41,213	**POLE3S_2_21**	34,632
KISSR0.0_4_19	86,959	**L008_9_19**	18,340	**POLE3S_3_19**	14,424
KISSR0.0_4_20	14,190	**L008_9_20**	78,876	**POLE3S_3_21**	57,108
KISSR0.0_4_21	28,525	**LZ2_1_20**	53,511	**POLE3S_4_20**	21,753
KISSR0.0_5_19	142,791	**LZ2_10_20**	72,031	**POLE3S_4_21**	30,637

KISSR0.0_5_20	11,072	**LZ2_10_21**	47,220	**POLE3S_5_19**	30,597
KISSR0.0_5_21	45,548	**LZ2_11_19**	23,380	**POLE3S_5_21**	38,883
KISSR0.0_6_20	25,235	**LZ2_11_20**	41,657	**POLE3S_6_19**	14,647
KISSR0.0_6_21	61,426	**LZ2_12_19**	18,663	**POLE3S_6_21**	30,355
KISSR0.0_7_19	15,071	**LZ2_12_20**	38,681	**POLE3S_7_19**	47,995
KISSR0.0_7_21	60,634	**LZ2_2_20**	15,620	**POLE3S_7_20**	33,503
KISSR0.0_8_19	126,671	**LZ2_2_21**	50,842	**POLE3S_7_21**	34,565
KISSR0.0_8_20	56,130	**LZ2_3_19**	41,948	**POLE3S_8_19**	53,491
KISSR0.0_8_21	73,235	**LZ2_3_21**	40,141	**POLE3S_8_20**	25,946
KISSR0.0_9_19	63,718	**LZ2_4_19**	16,436	**POLE3S_8_21**	30,494
KISSR0.0_9_20	94,116	**LZ2_4_20**	15,464	**POLE3S_9_20**	45,210
KISSR0.0_9_21	40,703	**LZ2_4_21**	30,621	**POLESOUT_1_20**	79,181
L001_1_20	69,121	**LZ2_5_19**	100,830	**POLESOUT_10_20**	105,561
L001_10_20	62,372	**LZ2_5_20**	25,241	**POLESOUT_10_21**	46,118
L001_10_21	40,366	**LZ2_5_21**	63,438	**POLESOUT_11_19**	33,973
L001_11_19	23,869	**LZ2_6_19**	30,662	**POLESOUT_11_20**	46,080
L001_11_20	38,398	**LZ2_6_20**	10,071	**POLESOUT_12_20**	50,735
L001_12_19	30,015	**LZ2_6_21**	74,326	**POLESOUT_2_20**	36,634
L001_12_20	25,130	**LZ2_7_20**	17,943	**POLESOUT_2_21**	33,648
L001_2_20	20,447	**LZ2_7_21**	73,048	**POLESOUT_3_19**	18,616
L001_2_21	41,243	**LZ2_8_19**	60,425	**POLESOUT_3_21**	46,797
L001_3_19	55,974	**LZ2_8_20**	31,421	**POLESOUT_4_19**	97,611
L001_3_20	33,450	**LZ2_8_21**	50,740	**POLESOUT_4_20**	15,640
L001_3_21	47,455	**LZ2_9_19**	10,507	**POLESOUT_4_21**	26,357
L001_4_19	62,834	**LZ2_9_20**	81,905	**POLESOUT_5_19**	25,865
L001_4_20	16,301	**LZ25A_1_20**	60,637	**POLESOUT_5_21**	49,238
L001_4_21	59,802	**LZ25A_1_21**	36,929	**POLESOUT_6_19**	14,811
L001_5_19	65,666	**LZ25A_10_20**	83,654	**POLESOUT_6_20**	15,163
L001_5_21	43,676	**LZ25A_10_21**	30,907	**POLESOUT_6_21**	65,067
L001_6_19	55,827	**LZ25A_11_19**	17,080	**POLESOUT_7_19**	64,203
L001_6_20	15,917	**LZ25A_11_20**	52,790	**POLESOUT_7_20**	25,430
L001_6_21	66,222	**LZ25A_12_19**	16,615	**POLESOUT_7_21**	35,781
L001_7_19	89,208	**LZ25A_12_20**	37,878	**POLESOUT_8_19**	50,673
L001_7_20	21,657	**LZ25A_2_20**	51,477	**POLESOUT_8_20**	38,149
L001_7_21	72,399	**LZ25A_2_21**	33,158	**POLESOUT_8_21**	67,504
L001_8_19	150,654	**LZ25A_3_19**	12,262	**POLESOUT_9_20**	86,132
L001_8_20	53,550	**LZ25A_3_20**	33,491	**POLESOUT_9_21**	48,871
L001_8_21	49,305	**LZ25A_3_21**	46,665	**RITTAE2_1_20**	54,298
L001_9_19	87,813	**LZ25A_4_19**	17,755	**RITTAE2_1_21**	48,316
L001_9_20	70,594	**LZ25A_4_20**	31,183	**RITTAE2_10_20**	71,018
L001_9_21	37,013	**LZ25A_4_21**	40,967	**RITTAE2_10_21**	51,779
L004_1_20	94,846	**LZ25A_5_20**	19,997	**RITTAE2_11_19**	34,798
L004_10_20	64,665	**LZ25A_5_21**	42,305	**RITTAE2_11_20**	72,037
L004_10_21	34,233	**LZ25A_6_19**	15,634	**RITTAE2_12_19**	23,292
L004_11_19	21,572	**LZ25A_6_21**	52,604	**RITTAE2_12_20**	27,845
L004_11_20	56,382	**LZ25A_7_19**	56,424	**RITTAE2_2_20**	68,756
L004_12_19	24,092	**LZ25A_7_20**	22,123	**RITTAE2_2_21**	24,529
L004_12_20	29,549	**LZ25A_7_21**	32,884	**RITTAE2_3_19**	14,624
L004_2_20	46,557	**LZ25A_8_19**	43,506	**RITTAE2_3_20**	43,584

L004_2_21	48,272	**LZ25A_8_20**	23,717	**RITTAE2_3_21**	41,907
L004_3_19	31,177	**LZ25A_9_19**	42,993	**RITTAE2_4_19**	17,614
L004_3_20	11,902	**LZ25A_9_20**	54,018	**RITTAE2_4_20**	20,993
L004_3_21	56,711	**LZ30_1_20**	57,864	**RITTAE2_4_21**	30,636
L004_4_20	16,779	**LZ30_1_21**	26,041	**RITTAE2_5_21**	41,138
L004_4_21	41,409	**LZ30_10_19**	10,086	**RITTAE2_6_19**	26,345
L004_5_19	27,050	**LZ30_10_20**	68,400	**RITTAE2_6_21**	46,329
L004_6_20	22,960	**LZ30_10_21**	40,942	**RITTAE2_7_19**	38,362
L004_6_21	43,553	**LZ30_11_19**	25,644	**RITTAE2_7_21**	46,134
L004_7_20	60,488	**LZ30_11_20**	57,308	**RITTAE2_8_19**	107,571
L004_7_21	45,275	**LZ30_12_19**	16,537	**RITTAE2_8_20**	28,133
L004_8_19	93,248	**LZ30_12_20**	25,439	**RITTAE2_8_21**	42,628
L004_8_20	54,394	**LZ30_2_20**	193,677	**RITTAE2_9_20**	43,811
L004_8_21	58,656	**LZ30_2_21**	22,063	**S308_1_20**	40,604
L004_9_19	64,591	**LZ30_3_20**	18,107	**S308_1_21**	52,580
L004_9_20	104,024	**LZ30_3_21**	53,517	**S308_10_19**	14,110
L005_1_20	45,496	**LZ30_4_19**	26,237	**S308_10_20**	75,491
L005_10_20	81,641	**LZ30_4_20**	18,544	**S308_11_19**	47,368
L005_10_21	55,821	**LZ30_4_21**	40,009	**S308_11_20**	66,475
L005_11_19	23,774	**LZ30_5_19**	33,743	**S308_12_19**	16,025
L005_11_20	47,251	**LZ30_5_20**	16,504	**S308_12_20**	49,384
L005_12_19	23,328	**LZ30_5_21**	65,446	**S308_2_20**	85,800
L005_12_20	48,266	**LZ30_6_19**	21,000	**S308_2_21**	35,427
L005_2_20	27,477	**LZ30_6_20**	23,343	**S308_3_19**	25,070
L005_2_21	39,338	**LZ30_6_21**	30,066	**S308_3_20**	32,080
L005_3_19	22,299	**LZ30_7_19**	39,048	**S308_3_21**	40,605
L005_3_21	41,506	**LZ30_7_20**	42,374	**S308_4_19**	87,900
L005_4_19	24,271	**LZ30_7_21**	63,949	**S308_4_20**	23,923
L005_4_20	21,645	**LZ30_8_19**	55,304	**S308_4_21**	32,531
L005_4_21	35,763	**LZ30_8_20**	30,177	**S308_5_19**	29,336
L005_5_19	81,630	**LZ30_8_21**	52,160	**S308_5_20**	16,791
L005_5_21	58,292	**LZ30_9_19**	118,379	**S308_5_21**	75,525
L005_6_19	15,663	**LZ30_9_20**	60,589	**S308_6_19**	103,414
L005_6_20	16,835	**LZ40_1_20**	85,675	**S308_6_20**	19,425
L005_6_21	68,739	**LZ40_1_21**	45,185	**S308_6_21**	78,761
L005_7_19	60,728	**LZ40_10_20**	98,781	**S308_7_19**	46,758
L005_7_20	15,387	**LZ40_10_21**	50,590	**S308_7_20**	17,488
L005_7_21	32,169	**LZ40_11_19**	32,832	**S308_8_20**	54,558
L005_8_19	38,863	**LZ40_12_19**	24,196	**S308_9_19**	11,619
L005_8_20	48,457	**LZ40_12_20**	24,497	**S308_9_20**	79,367
L005_8_21	50,637	**LZ40_2_20**	52,952	**S77_1_20**	37,138
L005_9_19	68,266	**LZ40_2_21**	41,099	**S77_10_19**	13,226
L005_9_20	65,398	**LZ40_3_19**	54,293	**S77_10_20**	59,970
L005_9_21	34,062	**LZ40_3_20**	22,912	**S77_10_21**	92,000
L006_1_20	121,402	**LZ40_3_21**	40,265	**S77_11_19**	12,042
L006_1_21	52,987	**LZ40_4_19**	62,015	**S77_12_19**	18,217
L006_10_20	69,771	**LZ40_4_20**	17,216	**S77_12_20**	68,352
L006_10_21	42,768	**LZ40_4_21**	41,690	**S77_2_20**	62,899
L006_11_19	38,256	**LZ40_5_19**	43,714	**S77_2_21**	69,613

L006_11_20	47,760	**LZ40_5_20**	34,480	**S77_3_19**	19,081
L006_12_19	17,868	**LZ40_5_21**	32,484	**S77_3_21**	52,008
L006_12_20	33,623	**LZ40_6_19**	17,476	**S77_4_19**	16,483
L006_2_20	57,514	**LZ40_6_20**	19,539	**S77_4_20**	15,182
L006_2_21	34,194	**LZ40_6_21**	62,535	**S77_4_21**	44,716
L006_3_20	19,579	**LZ40_7_19**	46,131	**S77_5_19**	20,176
L006_3_21	58,233	**LZ40_7_20**	19,153	**S77_5_21**	46,643
L006_4_20	36,761	**LZ40_7_21**	51,159	**S77_6_19**	72,832
L006_4_21	34,467	**LZ40_8_19**	50,468	**S77_6_20**	20,274
L006_5_19	25,542	**LZ40_8_20**	39,749	**S77_6_21**	79,554
L006_5_20	14,984	**LZ40_8_21**	51,857	**S77_7_19**	41,984
L006_5_21	38,195	**LZ40_9_19**	29,685	**S77_7_20**	43,760
L006_6_20	22,846	**LZ40_9_20**	113,292	**S77_7_21**	66,225
L006_6_21	43,750	**LZ40_9_21**	68,220	**S77_8_19**	110,263
L006_7_19	86,105	**PALMOUT_1_20**	54,149	**S77_8_20**	42,614
L006_7_20	15,198	**PALMOUT_1_21**	36,386	**S77_8_21**	58,774
L006_7_21	53,061	**PALMOUT_10_20**	47,151	**S77_9_20**	86,750
L006_8_19	84,425	**PALMOUT_10_21**	37,175	**S77_9_21**	72,055
L006_8_20	29,947	**PALMOUT_11_19**	51,800	**S79_1_20**	61,708
L006_8_21	43,461	**PALMOUT_12_19**	24,689	**S79_10_20**	59,110
L006_9_19	15,469	**PALMOUT_12_20**	74,662	**S79_10_21**	50,775
L006_9_20	75,004	**PALMOUT_2_20**	73,824	**S79_11_20**	93,690
L007_1_20	46,361	**PALMOUT_2_21**	33,631	**S79_12_19**	20,703
L007_1_21	40,718	**PALMOUT_3_19**	40,431	**S79_12_20**	52,718
L007_10_20	67,773	**PALMOUT_3_21**	54,149	**S79_2_20**	25,122
L007_10_21	22,909	**PALMOUT_4_19**	18,118	**S79_2_21**	30,920
L007_11_19	24,758	**PALMOUT_4_20**	14,841	**S79_3_19**	31,458
L007_11_20	42,615	**PALMOUT_4_21**	39,178	**S79_3_21**	54,142
L007_12_19	20,666	**PALMOUT_5_20**	18,801	**S79_4_19**	100,406
L007_12_20	37,320	**PALMOUT_5_21**	55,075	**S79_4_20**	16,238
L007_2_20	125,116	**PALMOUT_6_19**	75,209	**S79_4_21**	23,991
L007_2_21	24,990	**PALMOUT_6_20**	24,379	**S79_5_21**	43,355
L007_3_19	19,507	**PALMOUT_6_21**	42,458	**S79_6_19**	69,562
L007_3_20	11,319	**PALMOUT_7_19**	25,724	**S79_6_21**	63,097
L007_3_21	43,102	**PALMOUT_7_20**	14,861	**S79_7_19**	40,023
L007_4_19	15,803	**PALMOUT_7_21**	53,459	**S79_7_20**	15,841
L007_4_20	20,307	**PALMOUT_8_19**	45,322	**S79_7_21**	42,838
L007_4_21	38,413	**PALMOUT_8_20**	23,660	**S79_8_19**	11,343
L007_5_19	61,612	**PALMOUT_8_21**	39,399	**S79_8_21**	52,020
				S79_9_20	50,447

II. Supplemental Figures

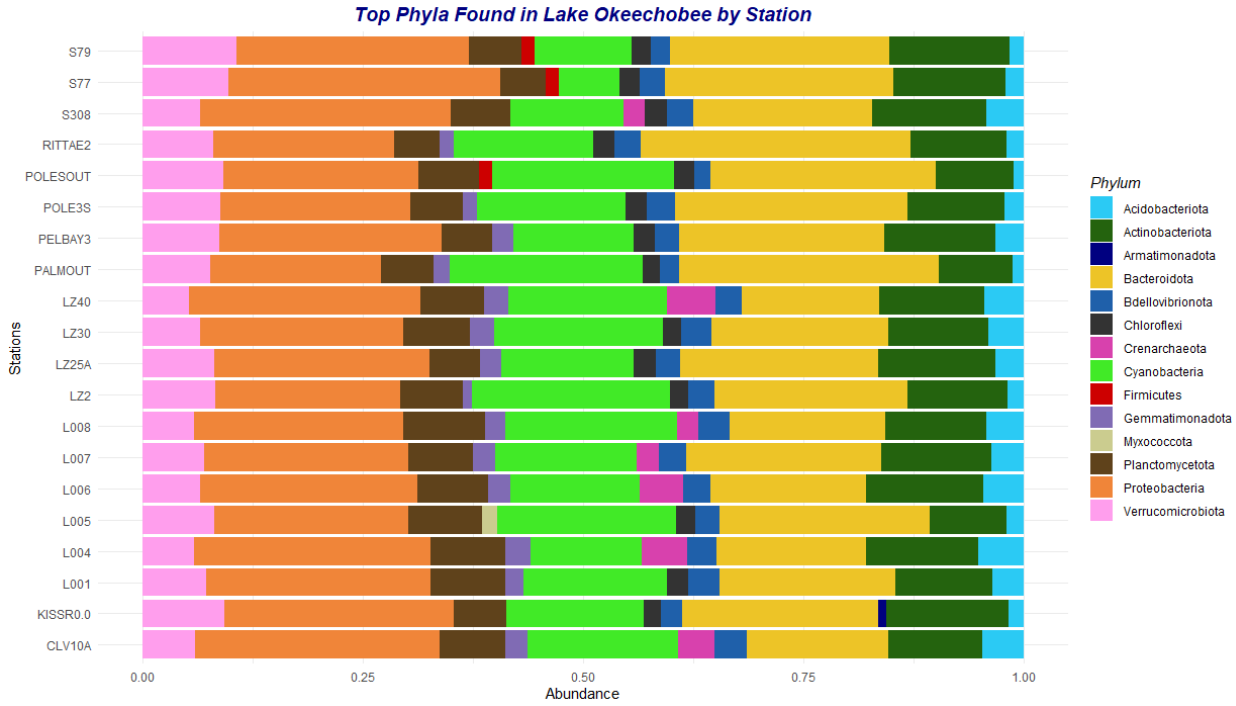

Figure S1. Top 10 phyla within each station over the sampling period (2019-2021).

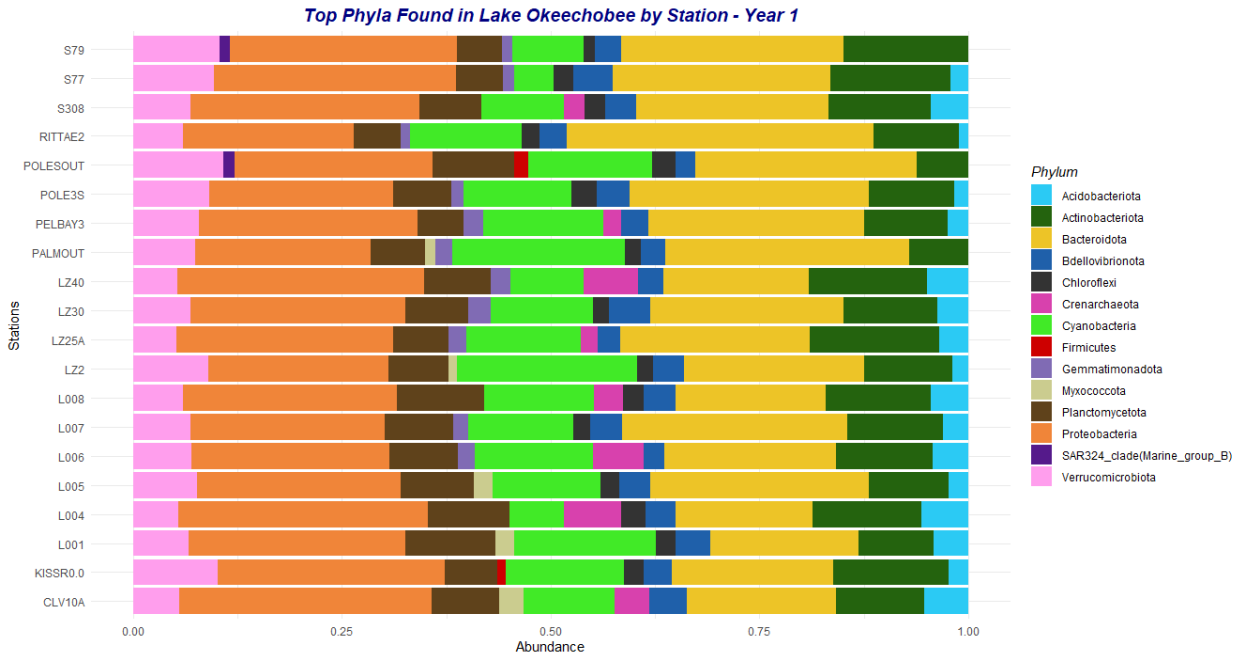

Figure S2. Top 10 phyla within each station during year 1 (2019).

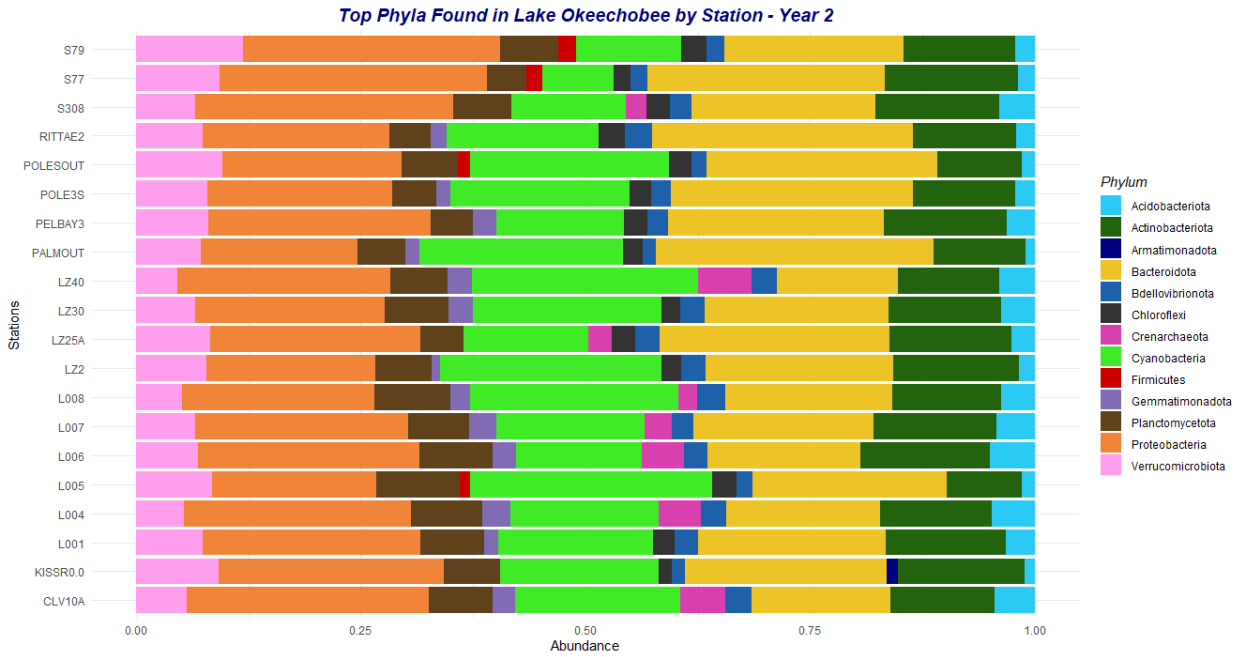

Figure S3. Top 10 phyla within each station during year 2 (2020).

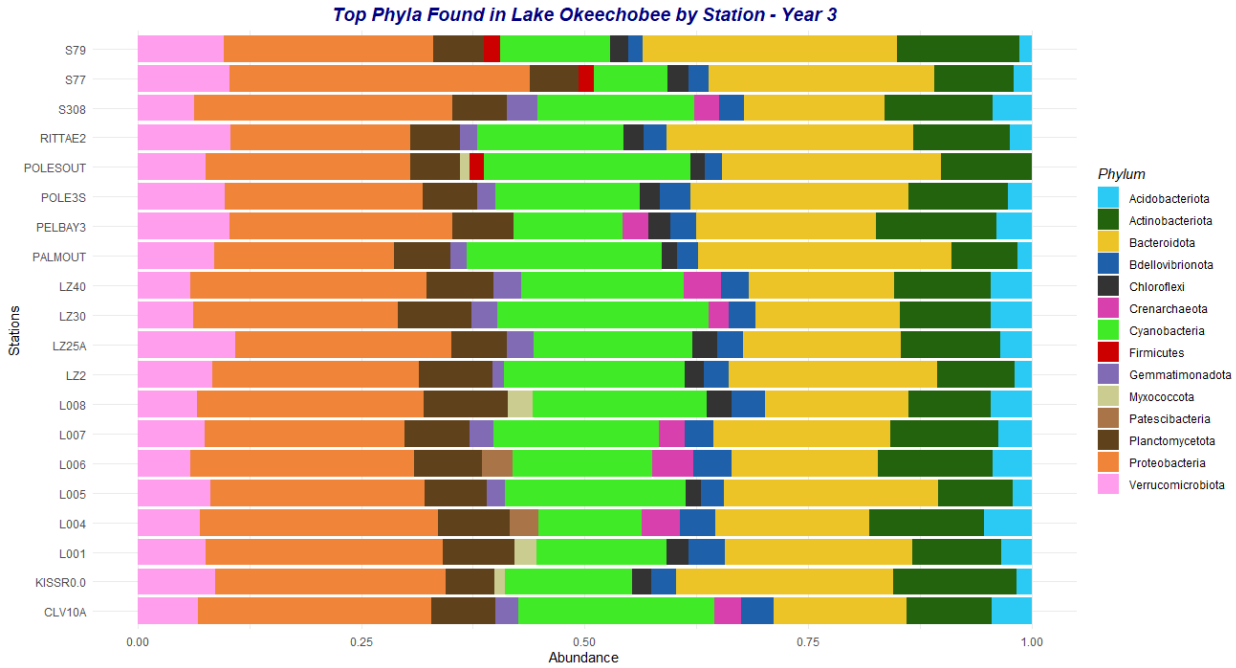

Figure S4. Top 10 phyla within each station during year 3 (2021).

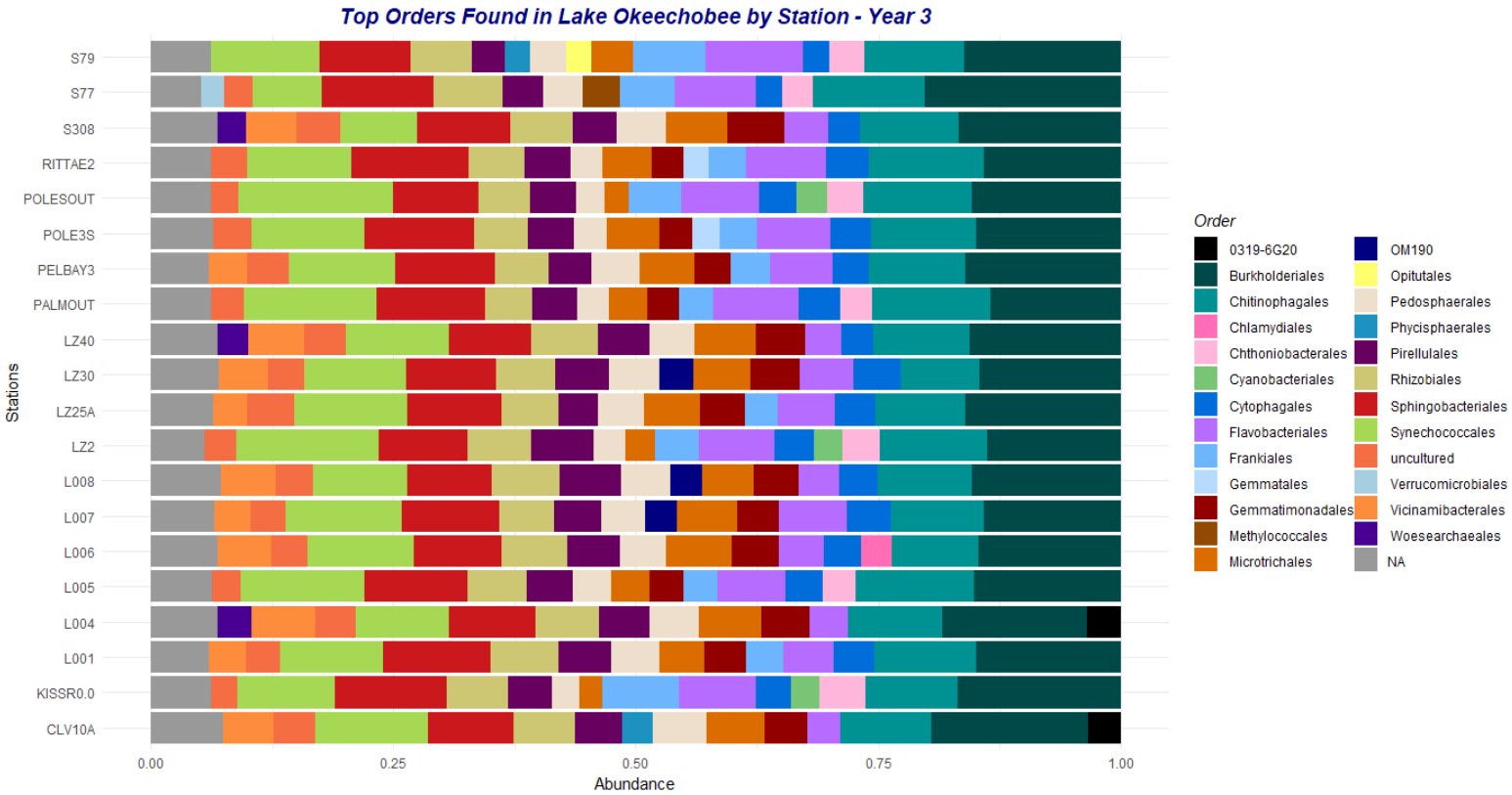

Figure S5. Top 15 orders within each station over the sampling period (2019-2021).

III. Draft Manuscript – Style of Frontiers in Water

Microcystis blooms alter the microbial community within Lake Okeechobee, FL across several years

Paisley S. Samuel[1*], Lauren E. Krausfeldt[1], Jose V. Lopez[1]

[1] Department of Biological Sciences, Nova Southeastern University, Guy Harvey Oceanographic Center, Dania Beach, FL, USA

*** Correspondence:**
Paisley S. Samuel
7309 SW 7th St, North Lauderdale, FL 33068
(954) 643-8390
paisley.samuel.98@gmail.com

Keywords: Lake Okeechobee, *Microcystis*, cyanoHABs, microbial community, cyanobacteria, blooms, freshwater ecosystems, high-throughput sequencing

Abstract

The Lake Okeechobee (Lake O) watershed is a Floridian freshwater ecosystem that has been affected by the increased frequency and intensity of harmful cyanobacterial bloom (cyanoHAB) events occurring over recent decades. Toxic cyanoHAB events are posing a threat to the ecosystem and economy of the lake due to the degradation of water quality. This study investigates how the microbial community structure within Lake O is affected by annual cyanobacterial harmful algal blooms over several years by characterizing the microbial community of Lake O and determining if cyanoHABs alter the microbial diversity in Lake O. Filtered surface water samples and public environmental data were collected from 21 routinely monitored sites within and connecting to Lake O from March 2019 to October 2021. DNA extraction, purification, and polymerase chain reactions on the V4 region of the 16S rRNA gene were used to create amplicon libraries for high-throughput sequencing on 541 samples, generating an average of over 40,000 reads per sample. After characterizing the dominant taxa within Lake O, the top four phyla include Proteobacteria, Bacteroidota, Cyanobacteria, and Actinobacteriota, which remained consistent across the sampling period. Microbial alpha diversity exhibited both spatial and temporal changes from year-to-year. The significant spatial differences observed across all three years suggest that there are stable biogeographical patterns within Lake O. Different environmental variables across the sampling period were found to drive beta diversity of the microbial communities in Lake O, with TN:TP ratio, turbidity, ammonia, total phosphate, nitrate + nitrite, dissolved oxygen, and pH remaining consistent in all years. *Microcystis* relative abundance was found to influence the alpha and beta diversity of the microbial communities, decreasing alpha diversity, and thus decreasing beta diversity as well. *Microcystis* relative

abundance also correlated with several environmental factors including temperature, total depth, and nitrate + nitrite concentrations. After observing such strong correlations to *Microcystis*, a co-occurrence network was created and has demonstrated that specific taxa may influence mutualistic or antagonistic relationships with *Microcystis*.

Introduction

Cyanobacteria are photoautotrophic, gram-negative, prokaryotic bacteria that can be found within numerous environments all over the world, including some extreme environments (Gaysina *et al.*, 2019; Mataloni and Komárek, 2004; Whitton and Potts, 2000a, b). Cyanobacteria, despite being commonly referred to as blue green algae, are true bacteria that perform photosynthesis, as they contain chlorophyll a. Cyanobacteria are able to rapidly proliferate to form dense accumulations of biomass known as blooms (Larkin & Adams, 2007). Some of these cyanobacteria blooms can either be harmless or harmful to their surrounding environment. Cyanobacteria are primarily responsible for causing harmful blooms (cyanoHABs) in freshwater environments (Rosen *et al.*, 2017). These cyanoHABs can result from water quality changes, which is primarily due to changes in nutrient levels especially in nitrogen (N) and phosphorus (P) levels. During photosynthesis, cyanobacteria utilize nutrients, such as carbon, potassium, iron, etc., along with solar energy to aid in their cell growth. However, nutrients must be present in a certain amount to promote cyanobacteria populations to bloom, if there is a deficiency in any of the nutrients then a bloom cannot occur (Markou *et al.*, 2014). When there are high levels of N and P due to agricultural fertilizer runoff, these cyanobacteria populations can bloom and create very dense mats on the surface. There are many other factors that produce favorable conditions for and exacerbate cyanobacterial blooms, including stagnant water and high temperatures (Paerl & Huisman, 2008).

CyanoHABs can further decrease water quality by producing cyanotoxins, water-soluble chemical metabolites that are toxic to the environment. Cyanotoxins can threaten the health of organisms in and around those ecosystems and the ecosystem itself. For example, there have been a number of incidents where cyanotoxins from the cyanoHABs caused animal and human poisonings (Bláha, Babica, & Maršálek, 2009). The thick, dense mats formed at the surface of the water also prevents sunlight from penetrating into the water column, decreasing the light needed for photosynthetic organisms residing deeper in the water column. Additionally, when these blooms begin to decay, they create an anoxic environment as large amounts of dissolved oxygen are used up thus reducing the amount of dissolved oxygen that other organisms in the lake need to survive and causing many organisms to die (Anderson, 2009). These negative impacts caused by cyanoHABs can have severe impacts on ecosystem functioning (Zamora-Barrios *et al.*, 2019; McQuaid, 2019; Bláha, Babica, & Maršálek, 2009). Despite immense research on cyanobacterial blooms and the factors that drive them, they remain difficult to predict and mitigate, and there is much more to be studied on the triggers of cyanoHABs (Facey, Apte, & Mitrovic, 2019; Bowling, 1994).

Lake Okeechobee is the largest lake in the southeastern United States and is located at the center of Florida's Everglades ecosystem (Lecher, 2021). Lake Okeechobee was once larger and deeper flowing north to south and provided a constant water source to the Everglades ecosystem. However, beginning in the late 19th century, the size, depth, and direction of flow of the lake were permanently altered as a series of major drainage projects (including the channelization of the

Kissimmee River, dredging of numerous canals, and construction of Hoover Dike) transformed the land around the lake to become a foundation for urban communities and agriculture (Lecher, 2021). Consequently, these water management projects greatly impacted the ecosystem and the water quality of the lake. Throughout the 1950s and 1960s, the water quality of Lake Okeechobee began to decline rapidly as the nutrient levels continually increased, primarily phosphorus levels, from agricultural land use (Canfield & Hoyer, 1988), thus further increasing the nutrient input of an already eutrophic environment that was initially limited in nitrogen rather than phosphorus (Missimer *et al.*, 2021).

As a result of the nutrient pollution and degrading water quality, cyanoHABs are a common occurrence in Lake Okeechobee, and in recent decades, these bloom events have increased in both abundance and prevalence (Rosen *et al.*, 2017). The freshwater toxic cyanoHABs that occur in Florida are primarily caused by the genus *Microcystis*, but blooms caused by the genera *Dolichospermum*, and *Cylindrospermopsis* also occur. The toxins produced during blooms caused by these genera include microcystins, anatoxin-a, saxitoxins, and cylindrospermopsin (Myer *et al.*, 2020). Metcalf *et al.* (2018) documented that the dominant blooming species in Lake Okeechobee was *Microcystis aeruginosa*. In fact, *Microcystis aeruginosa* is one of the most common bloom-forming and microcystin-producing cyanobacterium in the lake and is also found in freshwater ecosystems around the world (Harke, *et al.*, 2016).

Traditionally, cyanoHABs are considered to be predominantly driven by abiotic factors (Rollwagen-Bollens *et al.*, 2018; Visser *et al.*, 2016; Paerl & Scott, 2010). However, Shen *et al.* (2011) documented that some heterotrophic bacterioplankton can coexist with these bloom-forming cyanobacteria, which has led to speculation that the microbial community may also play a role during these cyanoHAB events (Wang *et al.*, 2021; Van Wichelen *et al.*, 2016). The interactions between photoautotrophic and heterotrophic bacteria play fundamental roles in aquatic ecosystems. As described by Zheng *et al.* (2018), heterotrophs utilize fixed carbon and other nutrients supplied by photoautotrophs and, in turn, provide these photoautotrophs with essential vitamins and amino acids. *Synechococcus* (Zheng *et al.*, 2018) and *Microcystis* (Van Wichelen *et al.*, 2016; Tu *et al.*, 2019) colonies frequently contain heterotrophic bacteria, and the colonies obtained from nature contain heterotrophic bacteria communities as well.

Certainly, there must be a diverse microbial community within Lake Okeechobee, yet, there has not been any studies done to characterize this diverse community until recently (Krausfeldt *et al.*, submitted). This microbial diversity could allow for the interaction of the bloom-forming cyanobacteria before, during, and after cyanoHAB events within Lake Okeechobee. Some studies have been done to investigate what roles the microbial community may play in the overall development and maintenance of these cyanoHABs, suggesting that these microbes who thrive alongside the bloom-forming cyanobacteria may have an important impact on the cyanobacterial growth and populations (Eiler & Bertilsson, 2004; Sigee, 2005). These microbes can also aid in the degradation of the organic material produced by the bloom, which contributes to the anoxic conditions that follow bloom degradation (Anderson, 2009; Havens, 2007). Understanding the interactions between the microbial community and these bloom-forming cyanobacteria and how microbial diversity changes during cyanoHABs may provide scientists the knowledge of key factors driving or sustaining blooms, serve as a biological indicator, and may aid efforts to reduce or mitigate the occurrences of these blooms.

In this study, we used 16S rRNA high-throughput sequencing to investigate how the structure of microbial communities within Lake Okeechobee (Lake O) is affected by annual cyanoHABs over several years. An initial characterization of the microbial community of Lake Okeechobee (Lake O) was conducted to look at the taxa that inhabit the lake. Afterwards, diversity indices were used, along with *Microcystis* abundance and microcystin concentrations, to determine whether the cyanoHABs occurring in Lake O do, in fact, alter the microbial community of Lake O.

Materials and Methods

Sample and environmental data collection

Beginning in March of 2019, surface water samples were collected monthly by the South Florida Water Management District (SFWMD) at 21 routinely sampled stations. These stations included 19 stations dispersed within Lake Okeechobee, one station located near the W.P. Franklin Lock along the Caloosahatchee River (S79), and another station located near the St. Lucie River lock (Figure 1). After collection, the water samples were kept on ice and shipped overnight to the USGS Water Science Center in Orlando, Florida, where each sample was filtered through two 0.22µm Sterivex filters (Millipore, SVGP01050), stored at -20°C, then transported on ice to the Microbiology and Genomics Lab at Nova Southeastern University (NSU) for further sample processing. This workflow of sample collection and processing was repeated until October of 2021.

Environmental data was collected from SFWMD's environmental database, DBHYDRO, that contains hydrologic, meteorologic, hydrogeologic, and water quality data (http://my.sfwmd.gov/dbhydroplsql/show_dbkey_info.main_menu). Environmental variables that were collected include: chlorophyll a (chl a, µg/L), pheophytin a (µg/L), secchi disk depth (m), silica (mg/L), turbidity (NTU), sulfate (mg/L), alkalinity (as total CaCO3, mg/L), ammonia (NH4, mg/L), total depth (m), pH, dissolved oxygen (mg/L), nitrate+nitrite (NO3+NO2, mg/L), total phosphate (PO4, mg/L), temperature (temp, °Celsius), total nitrogen (TN, mg/L), total phosphorus (TP, mg/L), TN and TP ratio, and three toxins associated with cyanoHABs, Anatoxin-a (µg/L), Cylindrospermopsin (µg/L), and Microcystin (µg/L). Additional variables were also considered for each sample, including month (1-12), season (wet or dry), year (1-3), station (CLV10A, KISR0.0, L001, L004, L005, L006, L007, L008, LZ2, LZ25A, LZ30, LZ40, PALMOUT, PELBAY3, POLE3S, POLESOUT, RITTAE2, S308, S77, and S79), and ecological zone (inflow, nearshore, pelagic, or S79). After retrieval, the environmental data was then corresponded to the collected samples for DNA extraction and sequencing.

Sample Processing

Once the collected samples were received at NSU, the sterivex filters were cut from their plastic tubing and DNA was extracted from the filters using the Qiagen® DNeasy® PowerLyzer® PowerSoil® kit (Qiagen, 12855-100) by following the manufacturer's protocol. Negative controls in the form of blank 'reagent-only' extractions were also included to detect any DNA contamination within the reagents. Following successful DNA extractions, an 1.5% agarose gel underwent an agarose gel electrophoresis protocol to confirm the presence of intact DNA in each sample.

Following the confirmation of intact DNA, a test polymerase chain reaction (PCR) was performed on each sample to confirm the successful amplification of PCR products. In short, a master mix was made using Invitrogen Platinum Hot Start PCR Master Mix (2X; ThermoFisher, 13000014), nuclease-free water, and universal primers 515F and 806R. DNA was then added and underwent amplification in a thermal cycler following the Earth Microbiome Project (EMP) 16S Illumina Amplicon protocol (Caporaso, 2018). 515F and 806R primers are used to target and amplify the V4 region of the 16S rRNA gene. A 1.5% agarose gel electrophoresis was also done to confirm the production of successful PCR products. To note, if the test PCR was unsuccessful—evidence that the concentration of extracted DNA was low—the sample was concentrated using a CentriVap DNA Vacuum Concentrator (©Labconco, Cat. No. 7970010), ran through another test PCR, and ran again on a 1.5% agarose gel to verify successful amplification. With the successful production of PCR products, barcoded 515F and 806R primers were then used, with each sample receiving identical barcoded 515F primer sequences and unique barcoded 806R primer sequences. A final 1.5% agarose gel was run to confirm the successful barcoding of the samples. Afterwards, the samples are cleaned using a modified AMPure XP beads protocol (PCR purification with Beckman Coulter AMPure XP magnetic beads and the VIAFLO 96, 2020), quantified using Qubit 3.0 and Qubit 4.0 Fluorometers (Life Technologies), and diluted to 4nM using nuclease-free water. The now-diluted barcoded samples were then pooled together and checked for quality and contamination using the Agilent TapeStation 4150 (Product #G2992AA). The final library pool was then loaded into the Illumina MiSeq system (Product #SY-410-1003) using the MiSeq Reagent Kit v3 at 600 cycles (Product #MS-102-3003) following a modified protocol.

Sequence analysis

The raw sequence data generated from the Illumina MiSeq system underwent initial bioinformatic analyses within a command-line program known as QIIME2. QIIME2 (Quantitative Insights into Microbial Ecology, version 2022.2) is a next-generation, open-source bioinformatics pipeline used for performing microbiome analysis from raw DNA sequence data (Bolyen *et al.*, 2019). Within the QIIME2 environment, the forward and reverse read sequence data (in the form of FASTQ files) were paired and demultiplexed to produce the sequence reads for each sample. The sample sequences were then trimmed, checked for chimeras, and quality filtered (Q-scores > 29) using the DADA2 software package built into the QIIME2 prorgam. There was a total of 11 sequencing runs included within this study, thus the raw sequence data for each run underwent demultiplexing, trimming, and quality filtering before being merged as one dataset. Lastly, the merged sequencing data set was assigned taxonomy using the SILVA 138 classifier (silva-138-99-515-806-nb-classifier.qza). The resulting dataset was then cleaned to ensure it did not contain any unwanted ASVs. A rarefaction curve was created to determine the sequence read cut-off point for any samples that were not fully sequenced. Any ASVs that were found in the negative controls were removed and the negative control samples were also removed from the sample pool. Any duplicate samples were removed by choosing the sample that obtained the most sequence reads and removing the other replicates. To ensure that the dataset contained no eukaryotes, ASVs that represented chloroplast or mitochondrial DNA were also removed. A final cleaning and normalization were performed using the 'vegan' package using the statistical computing language, R, in the RStudio software (version 4.2.0) where singletons, doubletons, and ASVs occurring less than 0.01% were removed.

Batch Correction

Due to the large-scale nature of this study, the hundreds of samples that were sequenced could be affected by differences in sample preparation and data acquisition conditions, for example, different individuals working on the sample preparation, different reagent batches, or even changes in instrumentation (Cuklina, *et al.*, 2021). This is known as the "batch effect" and can introduce noise that would in turn reduce the statistical power of the analyses (Cuklina, *et al.*, 2021). Taking this into consideration, the data was tested for any significant batch effects before moving on to further downstream analyses. The test was performed using the 'MMUPHin' and 'vegan' packages in R. An ANOSIM was performed to determine if the variation in the data caused by batch were significant ($p < 0.05$). If significant differences caused by batch were found in the data, the package 'MMUPHin' was used to conduct a batch correction.

Taxonomy analyses and visualization using QGIS

Taxonomic and statistical analyses were performed on the cleaned, normalized, batch corrected dataset using R. The 'phyloseq' package was used to determine the minimum, maximum, and average sequence read amounts, total number of unique ASVs, and number of unique phyla found in the data set. Top 10 taxa were calculated using packages 'phyloseq' and 'microbiome' and visualized using bar plots made using 'ggplot2' package for each year and station. QGIS, an analytical mapping software, was used to visualize the microbial community taxonomic distributions and patterns within Lake Okeechobee across the entire sampling period and within each year. An aerial satellite image of Lake Okeechobee was retrieved from Google Earth via the QGIS software and utilized as the raster layer. Point layers were created using the latitude and longitude coordinates retrieved from DBHYDRO for each station. Pie charts of the top 10 phyla found within each station were created for both the entire sampling period and within each year.

Diversity analyses

Alpha diversity, which describes the number of different species and how evenly distributed they are within a particular community, was assessed using the 'vegan' package and visualized using the 'base' and 'ggplot2' packages. Alpha diversity was measured by calculating the total number of species (species richness), species evenness (also known as Pielou's evenness index) (J), Shannon diversity index (H), and inverse Simpson's diversity index (inv. D). Differences between these alpha diversity indices were analyzed between samples. If the data was normally distributed, then an analysis of variance (ANOVA) would be used, otherwise a Kruskal-Wallis test was to be used. If there were significant differences found, a pairwise Wilcoxon test (for Kruskal-Wallis analyses) or Tukey test (for ANOVA analyses) was used as a post-hoc test to determine where the differences lie.

Beta diversity, which describes the differences between communities, was assessed using the 'vegan' package and visualized using the 'base' and 'ggplot2' packages as well. Beta diversity was measured by calculating Bray-Curtis dissimilarity between sites. These distance matrices were then used to produce non-metric multidimensional scaling (nMDS) plots in R to further visualize the distances between sites. To create the nMDS plots, the relative abudance data was transformed using the "total" method found within the 'decostand' function in 'vegan'. Functions 'betadisper' and 'permutest' in 'vegan', were used to calculate variances within each group and to determine if the variances differ by group. If the variances between groups were not significant, a

permutational multivariate ANOVA (PERMANOVA) with 999 permutations was performed. If the variances between groups were significant, an analysis of similarity (ANOSIM) with 999 permutations was performed. Canonical correspondence analysis (CCA) was also performed using the 'cca' function in 'vegan' to detect the interactions between the selected environmental variables and ASVs. The function 'envfit' was then used to get the p-value of correlation of each variable with overall bacterial communities and the p-value of each correlation between each ASV and all variables. Only significant ($p<0.05$) environmental variables with R^2 values higher than 0.3 were plotted as vectors overlaying the CCA plot.

Venn diagram and Co-occurrence network

Using the 'eulerr' package in R, a venn diagram was made to compare core taxa that appeared across the years (1, 2, and 3). Core taxa included any ASVs that was detected in a relative abundance of at least 0.1% and in at least 75% of the samples. Afterwards, a co-occurrence network was created to further investigate what taxa could be co-occurring with the genus *Microcystis*. This was done using the package 'Hmisc' in R and Cytoscape (version 3.9.1), a software used to create interactive networks. In R, a Pearson correlation matrix was created using the sample count data and making pairs of all 8,340 ASVs from the entire sampling period. The correlation matrix was then converted into a table format so that the individual R^2 values and their associated p-values could be extracted between each interaction pair that was created. Only the significant interactions ($p<0.05$) and the strongest correlations ($R^2 > 0.7$ OR $R^2 < -0.7$) were extracted from the table. This resulting table was then imported into Cytoscape (version 3.9.1) as a network, where it was filtered further to only include the network nodes and edges that interact with *Microcystis*.

Results

Sequencing statistics

Across the sampling period (March 2019 to October 2021), there were a total of 59,862,979 sequencing reads and 70,605 ASVs generated across all samples in this study. To determine the sequencing depth, or the total number of usable reads, that best represented the microbial communities of Lake O, total sequence reads were calculated for each sample and a rarefaction curve was generated to aid in determining the minimum sequence read cut-off point. The resulting rarefaction curve reached an inflection point at relatively 10,000 reads, thus, any samples that were below this amount were removed (Figure 2). As a result, 65,294 ASVs and 541 samples, with an average of 44,535 reads per sample, were used for further analysis (Table S1). Additional filtering for singletons, doubletons, and exceptionally low abundance ASVs (occurring less than 0.01%) was completed, resulting in 8,340 ASVs being utilized for further diversity analyses.

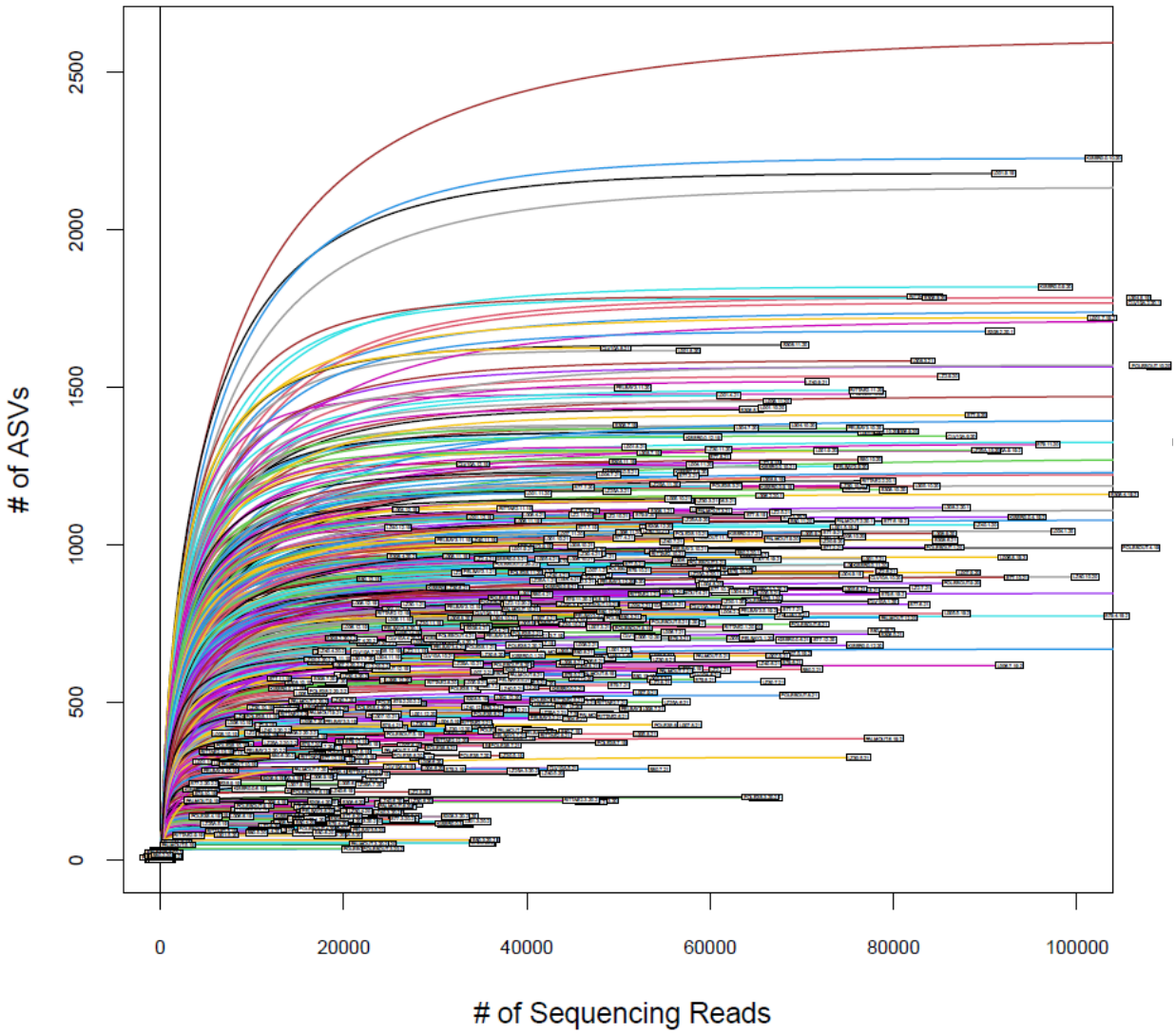

Figure 2. Rarefaction curve for number of sequencing reads versus number of ASVs to determine final samples for analysis. Each line represents one sample. Inflection point occurred at roughly 10,000 reads.

Dominant Phyla and Species diversity

The top ten phyla found in Lake O over the entire sampling period were Proteobacteria (24.7%), Bacteroidota (22.1%), Cyanobacteria (16.8%), Actinobacteriota (11.3%), Verrucomicrobiota (7.9%), Planctomycetota (6.8%), Bdellovibrionota (3.2%), Acidobacteriota (3.0%), Chloroflexi (2.2%), and Gemmatimonadota (1.9%) (Figure 3). The top ten phyla within each year varied within their makeups, with year 3 being the only year containing phylum Gemmatimonadota (Table 1, Figure 3). These phyla can also be seen within each station with Proteobacteria, Bacteroidota, and Cyanobacteria being the top three phyla found in each station (Figure 4). Additionally, when considering individual stations, the top 10 phyla also differed—both within all years overall (Figure S1) and between each year (Figures S2-S4).

Year 1 was the only year that included the phylum SAR324_ clade (marine group B) within the top 10 phyla of only 2 stations, POLESOUT and S79 (Figure 5, Figure S2). Year 2 had 13 unique phyla appear within the top 10 phyla of each station—one phylum short of years 1 and 3, both of which had 14 unique phyla each in their top 10 phyla across each station. Furthermore, year 2 was the only year that included the phylum Armatimonadota within the top 10 phyla occurring at only one station, KISSR0.0 (Figure 6, Figure S3). Year 2 also was the only year that did not have the phylum Myxococcota within the top 10 phyla of any station. Year 3 was the only year that included the phylum Patescibacteria within the top 10 phyla of only 2 stations, L004 and L006 (Figure 7, Figure S4).

Table 1. Average proportion and standard deviation of the relative abundances of the top 10 phyla in Lake Okeechobee by year.

Phylum	Year 1 (2019)	Year 2 (2020)	Year 3 (2021)
Proteobacteria	0.236 ± 0.057	0.215 ± 0.073	0.226 ± 0.055
Bacteroidota	0.217 ± 0.082	0.200 ± 0.071	0.196 ± 0.079
Cyanobacteria	0.119 ± 0.096	0.169 ± 0.102	0.159 ± 0.098
Actinobacteriota	0.105 ± 0.055	0.115 ± 0.041	0.099 ± 0.042
Planctomycetota	0.071 ± 0.025	0.060 ± 0.026	0.063 ± 0.023
Verrucomicrobiota	0.069 ± 0.031	0.068 ± 0.032	0.075 ± 0.031
Bdellovibrionota	0.033 ± 0.018	0.022 ± 0.014	0.027 ± 0.014
Acidobacteriota	0.029 ± 0.020	0.027 ± 0.018	0.029 ± 0.019
Chloroflexi	0.021 ± 0.009	0.021 ± 0.009	0.021 ± 0.008
Crenarchaeota	0.018 ± 0.028	0.018 ± 0.025	–
Gemmatimonadota	–	–	0.019 ± 0.011

308

309

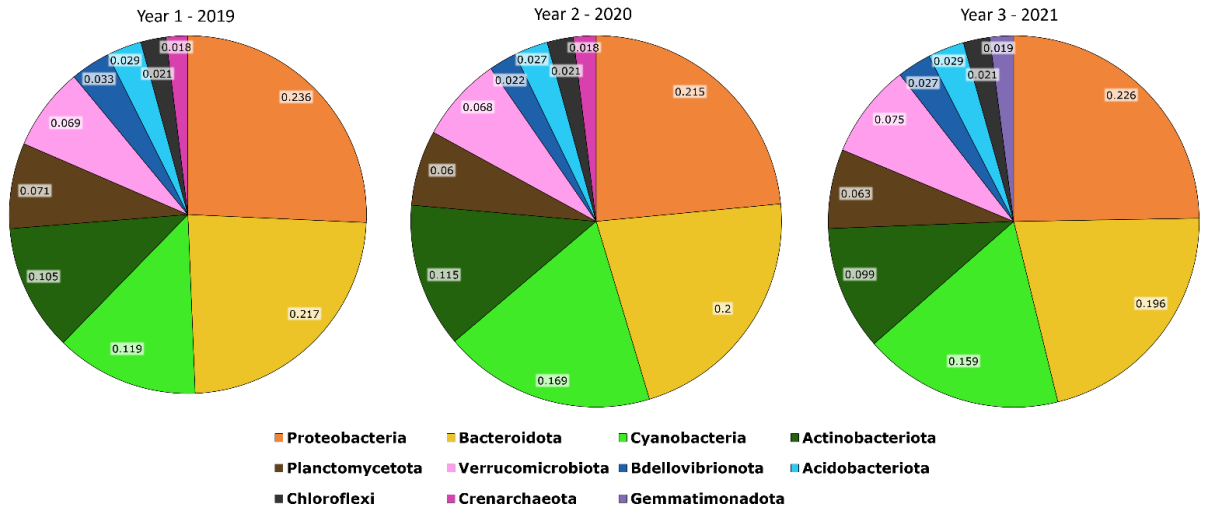

Figure 3. Pie charts depicting the proportions of the top 10 phyla within each year. The numbers indicate the total relative abundance of the respective year.

310

311

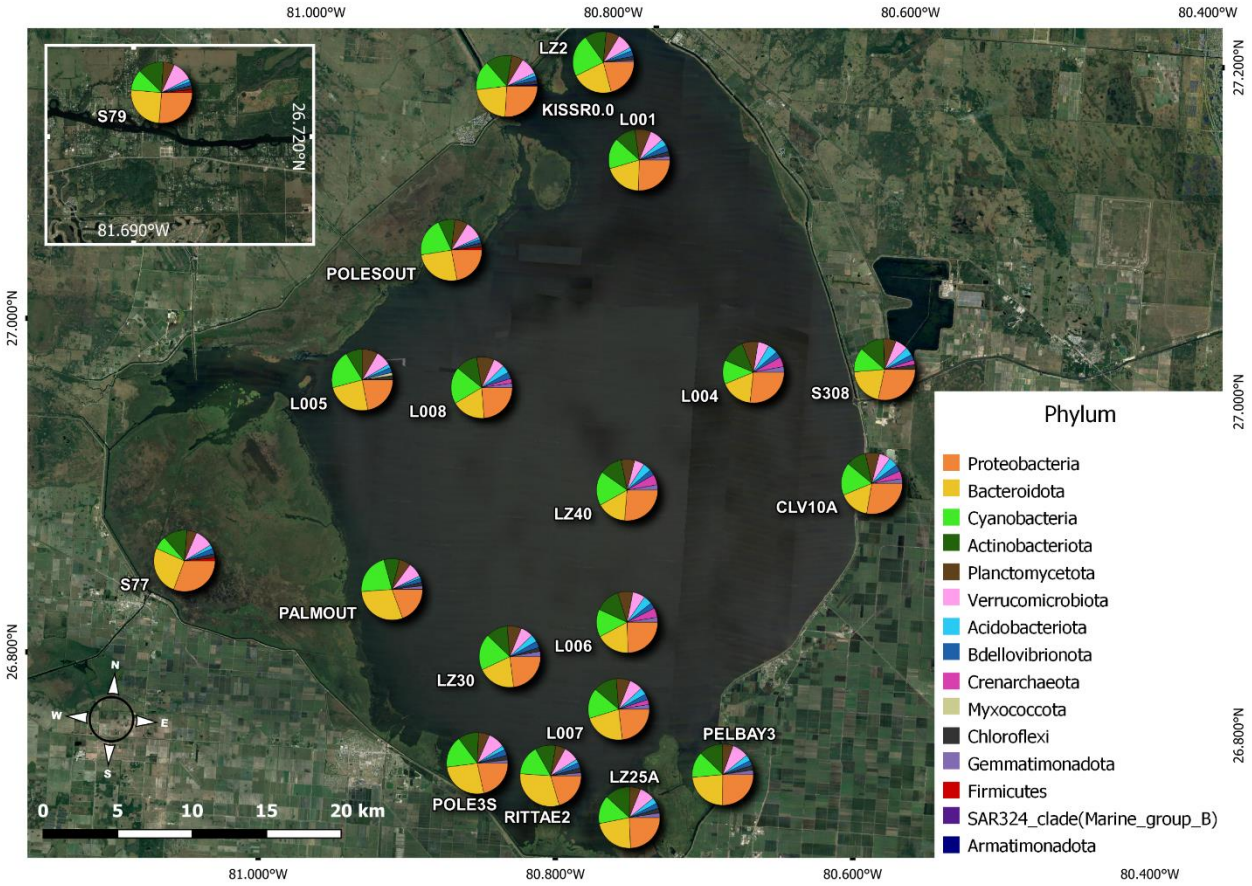

Figure 4. Pie charts showing the top phyla found in each station in Lake O over the sampling period.

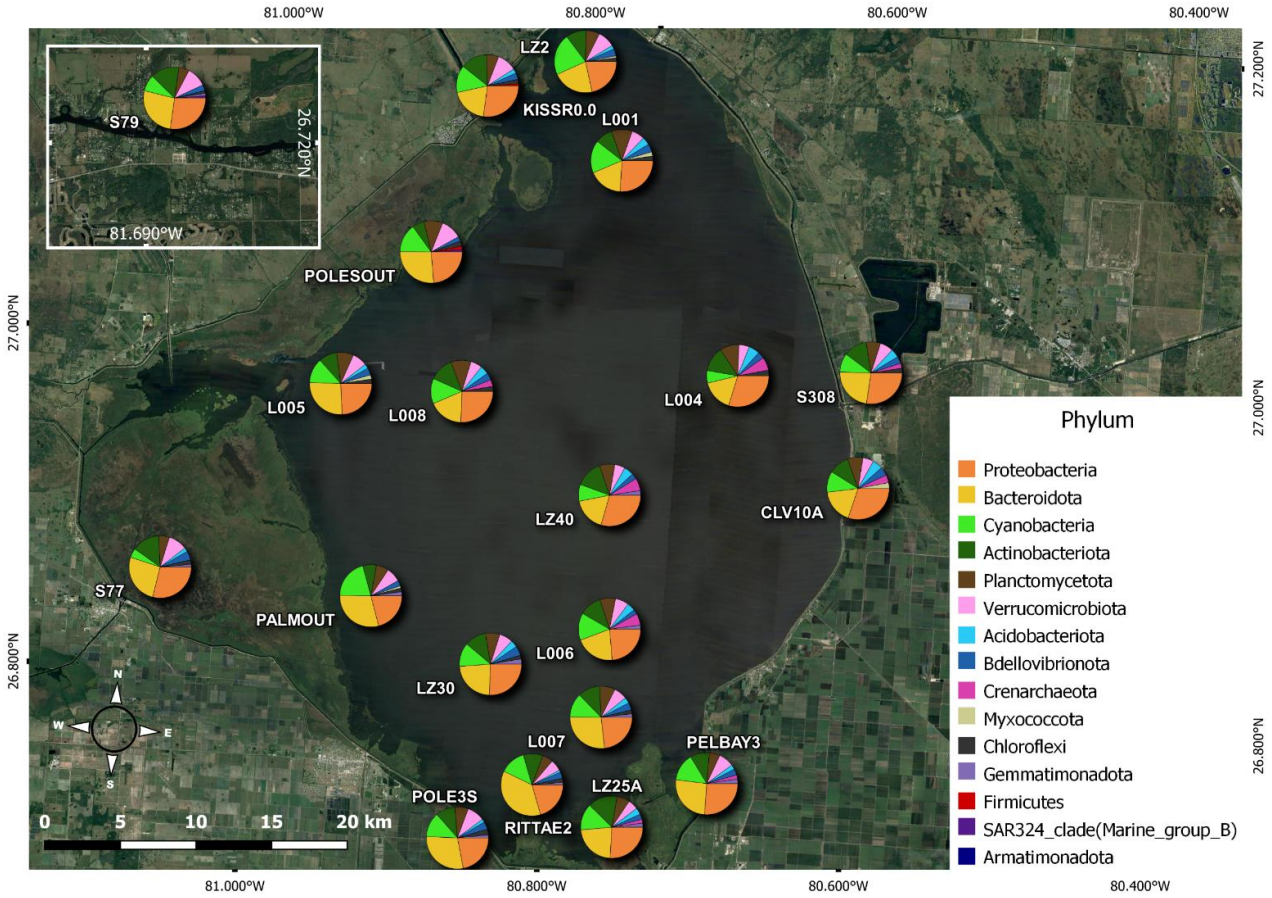

Figure 5. Pie charts showing the top phyla found in each station in Lake O within year 1 (2019).

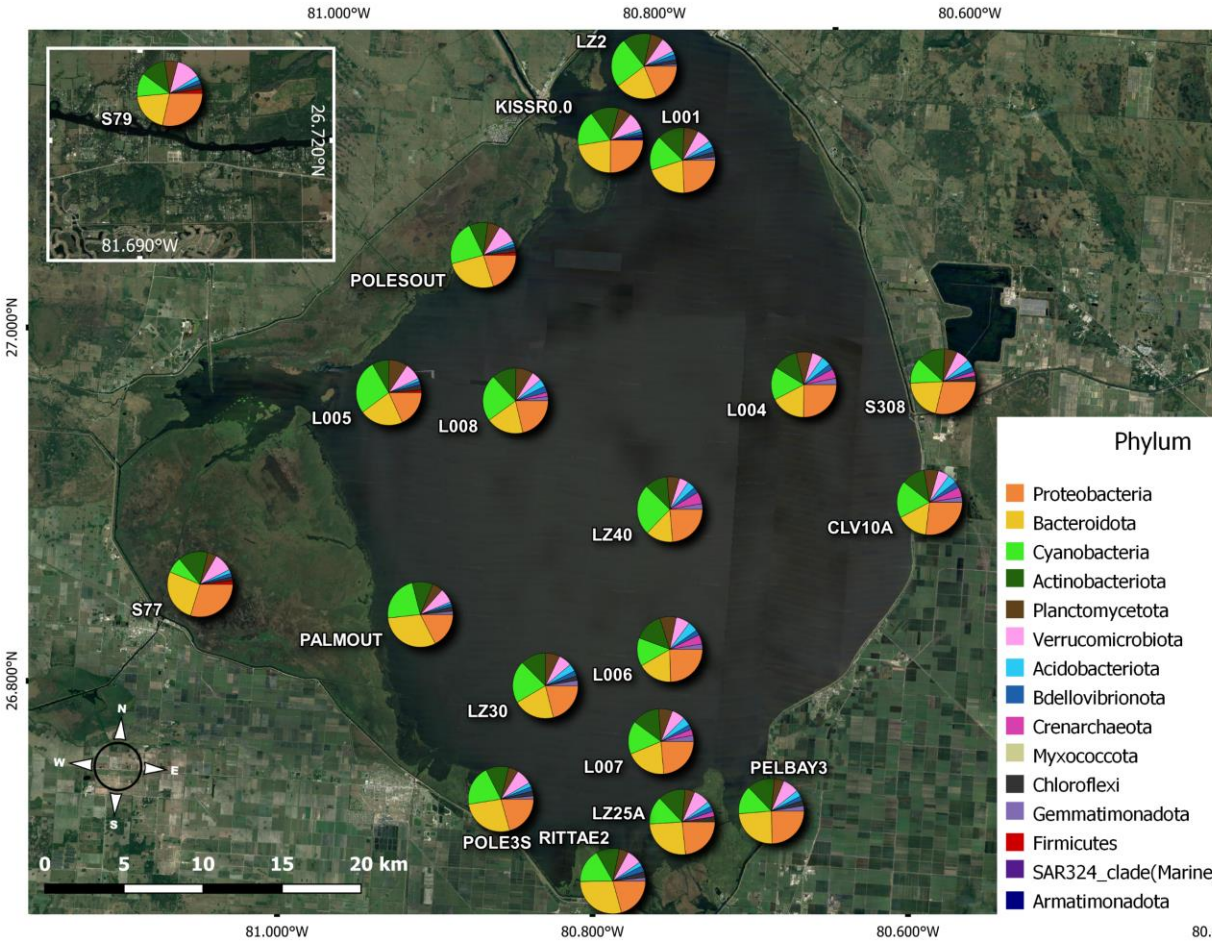

Figure 6. Pie charts showing the top phyla found in each station in Lake O within year 2 (2020).

327

328

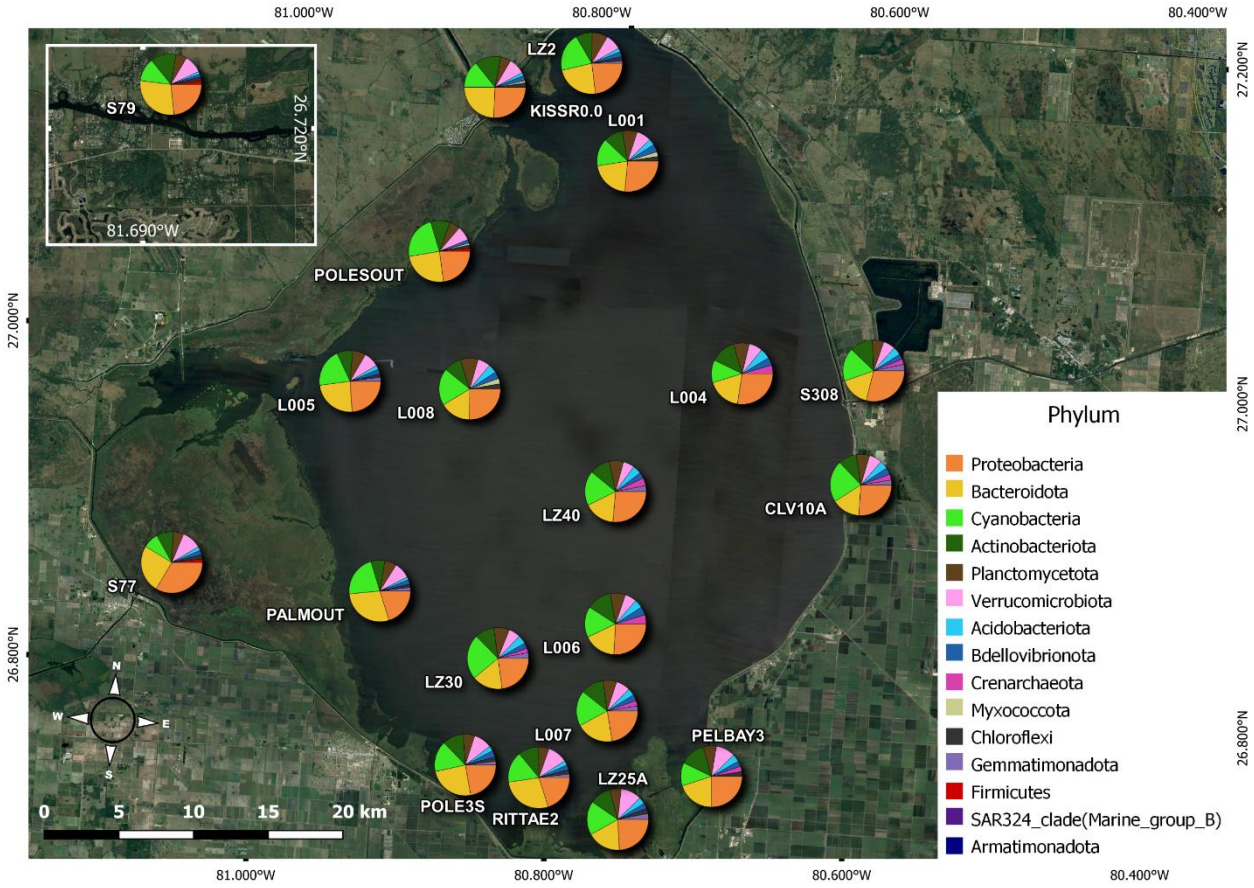

Figure 7. Pie charts showing the top phyla found in each station in Lake O within year 3 (2021).

329

330

Alpha diversity analyses

Alpha diversity was calculated using the Shannon diversity index, species evenness, species richness, and inverse Simpson diversity index. Year 3 (2021) exhibited significantly higher species richness than the previous two years (2019 and 2020, respectively) (year 1 vs. year 3, p = 0.0006; year 2 vs. year 3, p=0.0098) (Figure 8). Year 1 showed significantly higher species evenness throughout the microbial community compared to years 2 and 3, but year 2 was similar in species evenness compared to both years 1 and 3 (year 1 vs. year 2, p =0.042; year 1 vs. year 3, p=0.00013; year 2 vs. year 3, p=0.028) (Figure 8).

Within each year, alpha diversity differed by month (Table 3). The trends over time appeared to be seasonal, and analysis comparing season within each year showed that evenness specifically differed in year 2 (p = 0.00084) and year 3 (p = 0.037) (Figures 9-11). Alpha diversity also differed by zones across years 1 and 3, with year 2 showing no differences within all alpha diversity measures (Table 3, Figures 12-14). Alpha diversity differed by station within each year as well, with year 1 showing no significant differences in species evenness, year 2 only showing differences in species evenness, and year 3 showing differences in all the alpha diversity measures (Table 4).

Overall, the environmental variables measured did not strongly correlate to the alpha diversity in Lake O (Figure 15). Regarding species evenness, microcystin concentration showed the strongest correlation out of all the environmental variables (Pearson R^2 = -0.49) (Figure 15). Other environmental variables that correlated to species evenness included ammonia (Pearson R^2 = 0.11), nitrate + nitrite (Pearson R^2 = -0.10), and total phosphate (Pearson R^2 = -0.11) (Figure 15). Environmental variables that correlated to species richness include total nitrogen (Pearson R^2 = 0.17), TN:TP ratio (Pearson R^2 = -0.13), and total phosphorus (Pearson R^2 = 0.18) (Figure 15). The environmental variables that correlated to the diversity indices, Shannon and inverse Simpson, included microcystin (Pearson R^2 , shannon = -0.23; inv. Simpson = -0.20), nitrate + nitrite (Pearson R^2 , inv. Simpson = -0.10), total nitrogen (Pearson R^2 , shannon = 0.13; inv. Simpson = 0.17), total phosphorus (Pearson R^2 , shannon = 0.06; inv. Simpson = 0.10) and total phosphate (Pearson R^2 , inv. Simpson = -0.12) (Figure 15). There were no correlations between any of the alpha diversity measures and chlorophyll a, temperature, nor pH (Figure 15). *Microcystis* relative abundance had a srtong, negative correlation with species evenness (Pearson R^2 = -0.72), with additional negative correlations with Shannon diversity index (Pearson R^2 = -0.23), and inverse Simpson diversity index (Pearson R^2 = -0.22) (Figure 15).

363

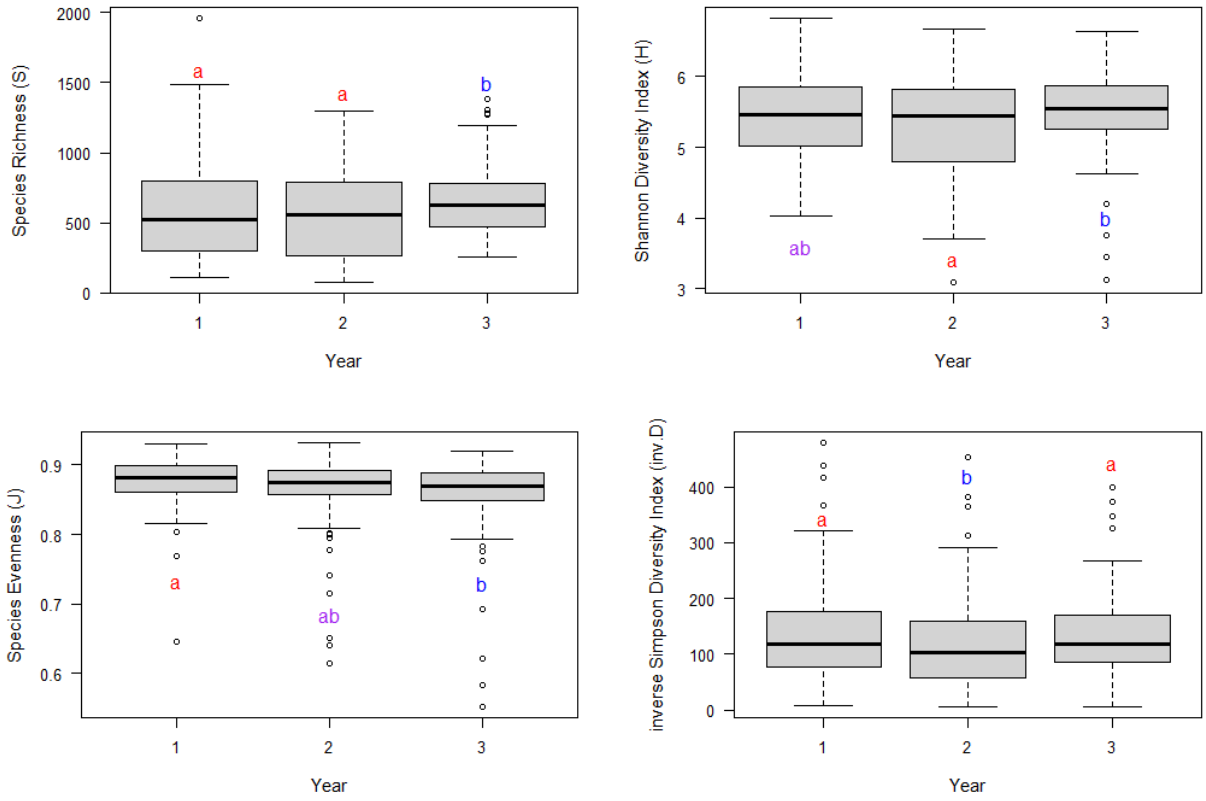

Figure 8. Alpha diversity comparison between years. Letters and colors represent the significant differences between each year; same letter and color indicate no differences and different letters and colors indicate significant differences are present ($p < 0.05$). Year 1 = 2019, Year 2 = 2020, and Year 3 = 2021.

Table 2. Kruskal-Wallis p-values for alpha diversity measure by month across each year. A star indicates that the p-value was significant ($p < 0.05$).

Alpha Diversity measure	Year 1	Year 2	Year 3
Species richness (S)	0.0017*	< 2.2e-16*	8.819e-08*
Species evenness (J)	0.13	0.00025*	2.848e-05*
Shannon Diversity Index (H)	0.0024*	< 2.2e-16*	8.126e-07*
Inverse Simpson Diversity Index (inv.D)	0.027*	< 2.2e-16*	1.383e-05

364

365

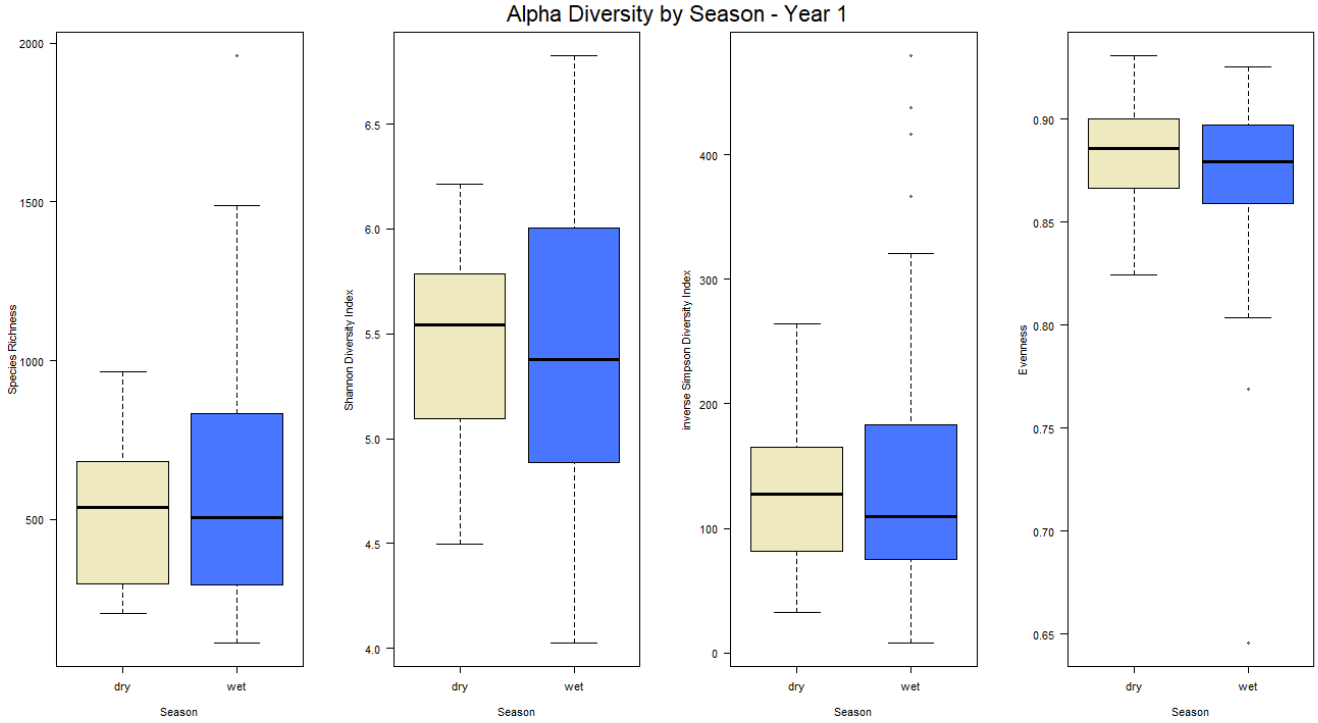

Figure 9. Alpha diversity measures across seasons in year 1. There were no significant differences between season and each alpha diversity measure. Tan = dry season; blue = wet season. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

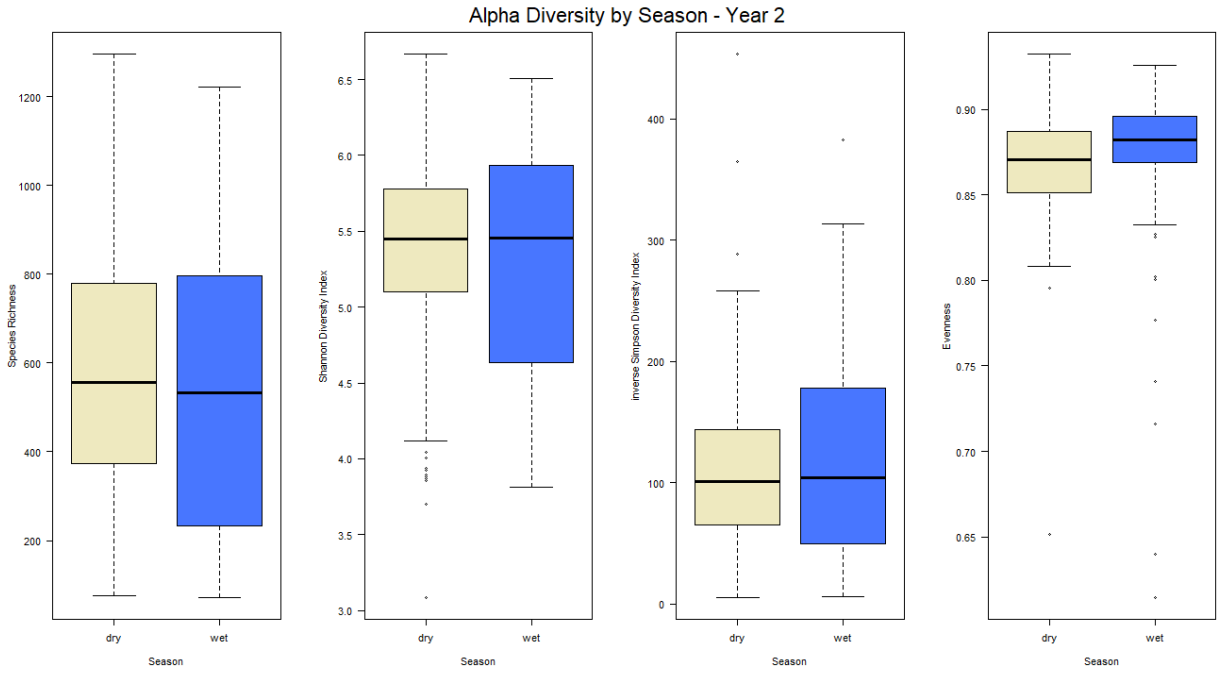

Figure 10. Alpha diversity measures across seasons in year 2. Significant differences were found in species evenness between seasons (p = 0.001). Tan = dry season; blue = wet season. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

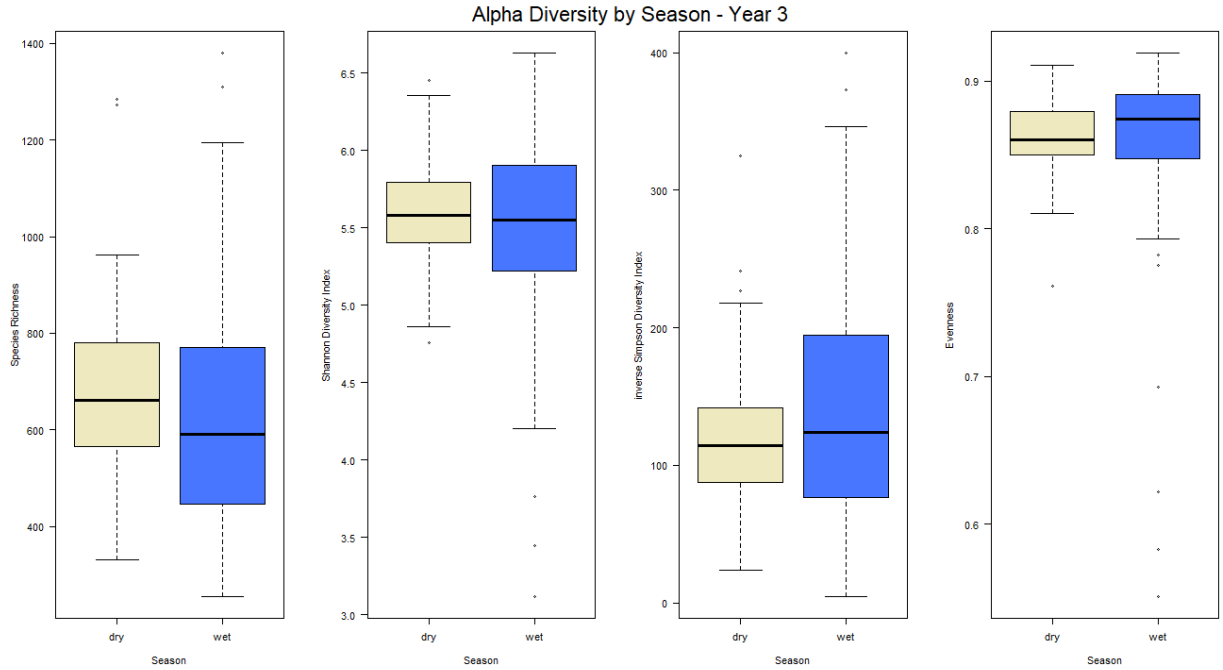

Figure 11. Alpha diversity measures across seasons in year 3. Significant differences were found in species evenness between seasons (p = 0.001). Tan = dry season; blue = wet season. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

366

Table 3. Kruskal-Wallis p-values for alpha diversity measure by zone across each year. A star indicates that the p-value was significant ($p < 0.05$).

Alpha Diversity measure	Year 1	Year 2	Year 3
Species richness (S)	0.0073*	0.54	0.00040*
Species evenness (J)	0.0033*	0.10	0.0015*
Shannon Diversity Index (H)	0.0082*	0.82	0.0020*
Inverse Simpson Diversity Index (inv.D)	0.035*	0.54	0.0034*

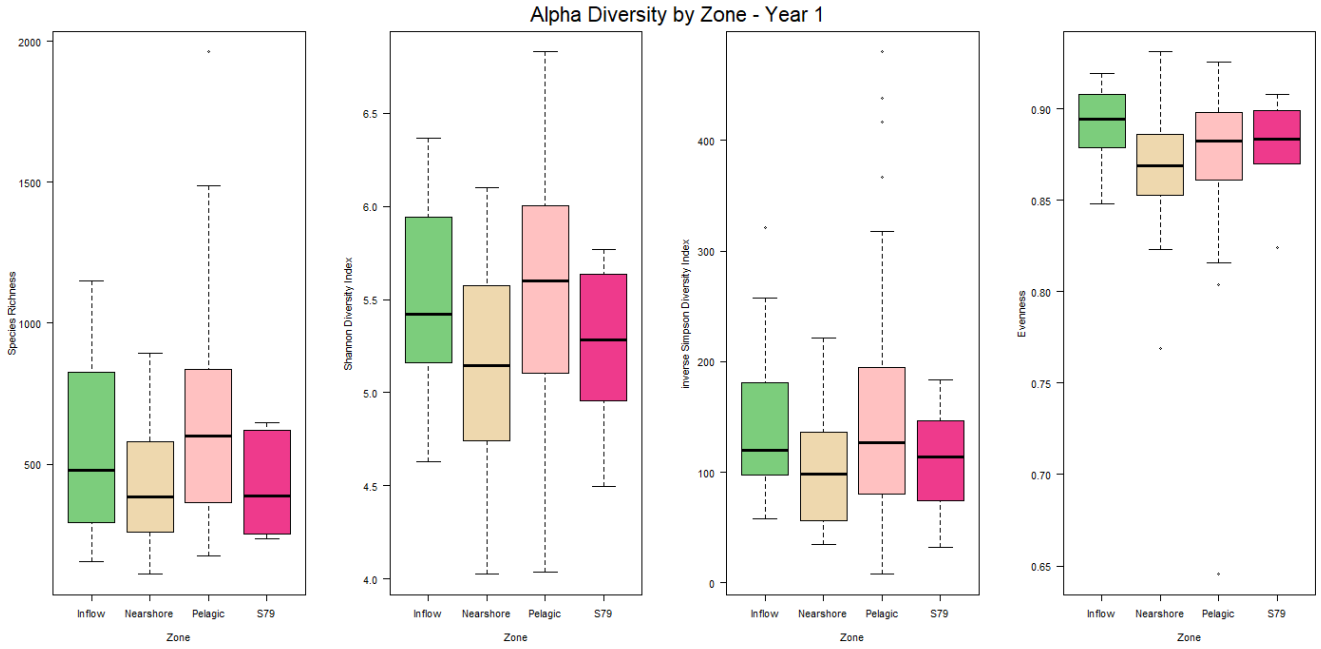

Figure 12. Alpha diversity measures across zones in year 1. Green = Inflow zone; Beige = Nearshore zone; Light pink = Pelagic zone; Bright pink = zone S79. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

377

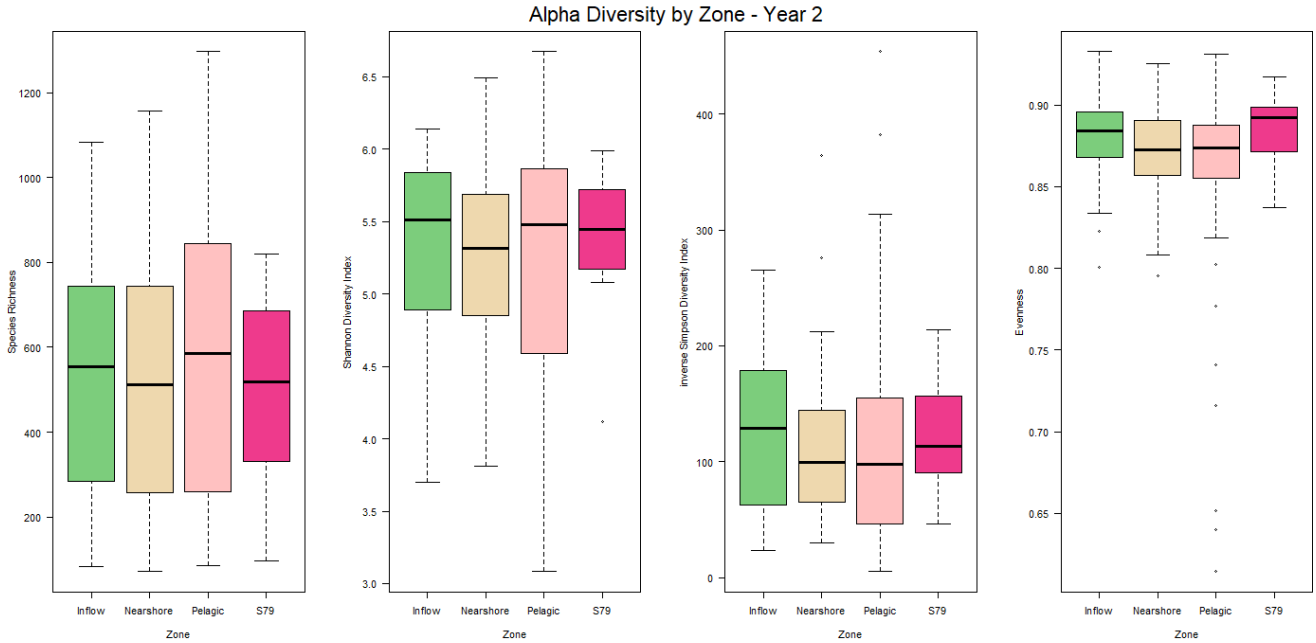

Figure 13. Alpha diversity measures across zones in year 2. Green = Inflow zone; Beige = Nearshore zone; Light pink = Pelagic zone; Bright pink = zone S79. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

378

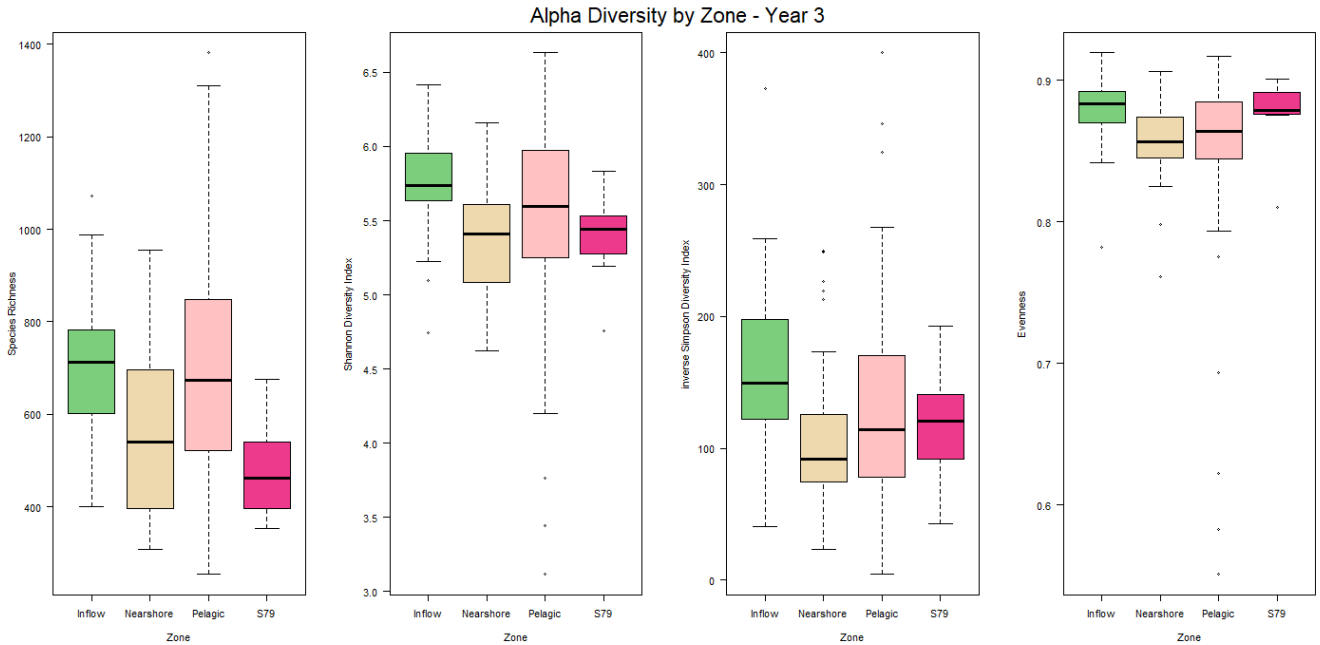

Figure 14. Alpha diversity measures across zones in year 3. Green = Inflow zone; Beige = Nearshore zone; Light pink = Pelagic zone; Bright pink = zone S79. Panels from left to right: species richness, Shannon diversity index, inverse Simpson index, and species evenness.

Table 4. Kruskal-Wallis p-values for alpha diversity measure by station across each year. A star indicates that the p-value was significant ($p < 0.05$).

Alpha Diversity measure	Year 1	Year 2	Year 3
Species richness (S)	0.0054*	0.99	0.0091*
Species evenness (J)	0.016[a]	0.0080*	0.0015*
Shannon Diversity Index (H)	0.0025*	0.88	0.0068*
Inverse Simpson Diversity Index (inv.D)	0.0028*	0.31	0.0017*

[a]Although the p-value was significant, there were no differences found between the stations.

379

380
381
382

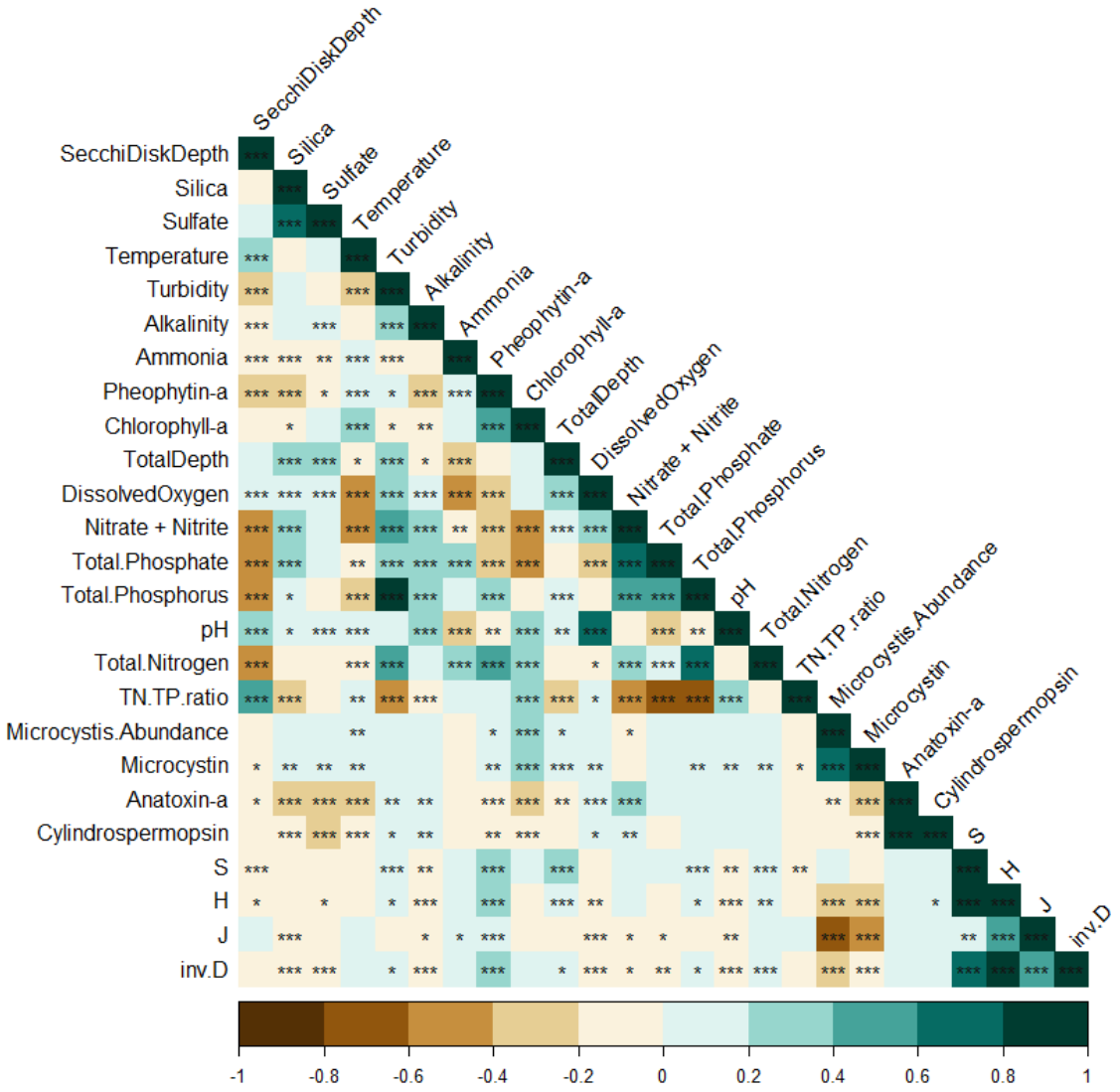

Figure 15. **Correlation heat map between the environmental variables and the alpha diversity indices.** Stars indicate the significance level; * = 0.05, ** = 0.01, *** = 0.001. No star indicates that the relationship is not significant. Alpha diversity measures can be found at the bottom of the heatmap: S = species richness, H = Shannon diversity index, J = species evenness, inv.D = inverse Simpson diversity index. TN.TP.ratio = ratio of total nitrogen and total phosphorus.

383

Venn diagram of core taxa between years

Each sampling year may have shared unique core taxa. To reiterate, core taxa is defined as any ASVs that were detected at a relative abundance of at least 0.1% and in at least 75% of the samples. A Venn diagram was created between each year, and it showed that all years shared 12 core taxa (Figure 16). Years 1 and 2 did not have any core taxa that was unique to them, nor did they share any core taxa (Figure 16). Year 3, however, had 14 unique core taxa, shared 4 core taxa with year 2, and shared 2 core taxa with year 1 (Figure 16). The taxonomic information for each taxon placed in the venn diagram can be found in Table 5. It can be seen from the table that the phylum Cyanobacteria are only found in the core taxa shared between years 2 and 3 and within the unique core taxa of year 3 (Table 5). Verrucomicrobiota was the only phylum of heterotrophic bacteria found within the shared taxa between year 2 and year 3 (Figure 16, Table 5).

Figure 16. Venn diagram of the number of shared core taxa between years across the sampling period. Year 1 = red; Year 2 = blue; Year 3 = green. Numbers represent the number of taxa.

Table 5. Core taxa comparisons between years (corresponding to venn diagram). Taxonomic information is structured by phylum, class, order, family, and genus. Dashes indicate that there were no shared taxa between specified years.

	Taxonomic Information
Year 1 Only	——
Year 2 Only	——
Year 3 Only	1. Actinobacteriota, Acidimicrobiia, Microtrichales, Ilumatobacteraceae, CL500-29_marine_group 2. Actinobacteriota, Actinobacteria, Frankiales, Sporichthyaceae, 3. Actinobacteriota, MB-A2-108, MB-A2-108, MB-A2-108, MB-A2-108 4. Verrucomicrobiota, Verrucomicrobiae, Pedosphaerales, Pedosphaeraceae, SH3-11 5. Proteobacteria, Gammaproteobacteria 6. Proteobacteria, Gammaproteobacteria, Burkholderiales, Oxalobacteraceae, 7. Proteobacteria, Gammaproteobacteria, Gammaproteobacteria_Incertae_Sedis, Unknown_Family, Acidibacter 8. Proteobacteria, Gammaproteobacteria, JG36-TzT-191, JG36-TzT-191, JG36-TzT-191 9. Proteobacteria, Gammaproteobacteria, Oceanospirillales, Pseudohongiellaceae, BIyi10 10. Bacteroidota, Bacteroidia, Sphingobacteriales, AKYH767, AKYH767 11. Bacteroidota, Bacteroidia, Sphingobacteriales, env.OPS_17, env.OPS_17 12. Bacteroidota, Bacteroidia, Sphingobacteriales, NS11-12_marine_group, NS11-12_marine_group 13. Cyanobacteria, Cyanobacteriia, Synechococcales, Cyanobiaceae, Cyanobium_PCC-6307 14. Gemmatimonadota, Gemmatimonadetes, Gemmatimonadales, Gemmatimonadaceae
Years 1 & 2	——
Years 1 & 3	1. Actinobacteriota, Actinobacteria, Frankiales, Sporichthyaceae, hgcI_clade 2. Proteobacteria, Alphaproteobacteria, Rhizobiales, Rhizobiales_Incertae_Sedis, uncultured
Years 2 & 3	1. Verrucomicrobiota, Verrucomicrobiae, Opitutales, Opitutaceae, Opitutus 2. Cyanobacteria, Cyanobacteriia, Synechococcales, Cyanobiaceae, Cyanobium_PCC-6307 3. Cyanobacteria, Cyanobacteriia, Synechococcales, Cyanobiaceae, Cyanobium_PCC-6307 4. Cyanobacteria, Cyanobacteriia, Synechococcales, Cyanobiaceae, Cyanobium_PCC-6307

ALL years	1. Actinobacteriota, Acidimicrobiia, Microtrichales, Ilumatobacteraceae, CL500-29_marine_group 2. Actinobacteriota, Acidimicrobiia, Microtrichales, Ilumatobacteraceae, CL500-29_marine_group 3. Actinobacteriota, Acidimicrobiia, Microtrichales, Ilumatobacteraceae, CL500-29_marine_group 4. Actinobacteriota, Actinobacteria, Frankiales, Sporichthyaceae, hgcI_clade 5. Bacteroidota, Bacteroidia, Chitinophagales, Saprospiraceae, Candidatus_Aquirestis 6. Bacteroidota, Bacteroidia, Flavobacteriales, Crocinitomicaceae, Fluviicola 7. Bacteroidota, Kapabacteria, Kapabacteriales, Kapabacteriales, Kapabacteriales 8. Verrucomicrobiota, Verrucomicrobiae, Methylacidiphilales, Methylacidiphilaceae, uncultured 9. Proteobacteria, Alphaproteobacteria, Rickettsiales, Rickettsiaceae, Candidatus_Megaira 10. Chloroflexi, SL56_marine_group, SL56_marine_group, SL56_marine_group, SL56_marine_group 11. Planctomycetota, Phycisphaerae, Phycisphaerales, Phycisphaeraceae, CL500-3 12. Proteobacteria, Gammaproteobacteria, Burkholderiales, Burkholderiaceae, Limnobacter

401

Beta diversity analyses

Beta diversity was calculated using Bray-Curtis dissimilarity. Following ANOSIM and PERMANOVA analyses, it was revealed that there were significant differences between stations (ANOSIM R = 0.1967; p = 0.01) across all sampling years. However, there were no significant differences in year (p = 0.75), season (p = 0.78), month (p = 0.91), nor zone (p = 0.19) across the sampling years. When investigating within each year, there were significant differences by station across each year (year 1, p = 0.001; year 2, p = 0.001; year 3, p = 0.001) and there were significant differences by zone within year 1 (p = 0.001) and year 3 (p = 0.001).

Environmental variables were fitted onto a CCA plot through vectors to show which environmental variables may be driving the differences in the microbial community within the lake across the sampling period and within each year (Figures 18-21). The length of the vector is proportional to its importance and the angle between two vectors reflects the degree of correlation between variables. (Sarker, et al., 2014) To reiterate, the environmental variable vectors that were included in the CCA plots exhibited a significant effect ($p < 0.05$) and correlation (Pearson $R^2 > 0.3$) on the microbial community of Lake O. Across all three years, the environmental variables accounted for about 14.47% of the variation within the microbial communities in Lake O and these variables included TN:TP ratio (Pearson $R^2 = 0.57$), pH (Pearson $R^2 = 0.34$), nitrate + nitrite (Pearson $R^2 = 0.55$), dissolved oxygen (Pearson $R^2 = 0.43$), turbidity (Pearson $R^2 = 0.42$), total phosphate ("phosphate.ortho"; Pearson $R^2 = 0.48$), and ammonia (Pearson $R^2 = 0.34$) (Figure 18). In year 1, the environmental variables accounted for about 17.44% of the variation within the microbial communities in Lake O and these variables included TN:TP ratio (Pearson $R^2 = 0.65$), pH (Pearson $R^2 = 0.51$), nitrate + nitrite (Pearson $R^2 = 0.46$), dissolved oxygen (Pearson $R^2 = 0.49$), turbidity (Pearson $R^2 = 0.31$), secchi disk depth (Pearson $R^2 = 0.30$), and ammonia (Pearson $R^2 = 0.60$) (Figure 19). In year 2, the environmental variables accounted for about 17.26% of the variation within the microbial communities in Lake O and these variables included TN:TP ratio (Pearson $R^2 = 0.62$), pH (Pearson $R^2 = 0.69$), nitrate + nitrite (Pearson $R^2 = 0.55$), dissolved oxygen (Pearson $R^2 = 0.51$), turbidity (Pearson $R^2 = 0.52$), total phosphate ("phosphate.ortho"; Pearson $R^2 = 0.35$), ammonia (Pearson $R^2 = 0.35$), and chlorophyll a (Pearson $R^2 = 0.35$) (Figure 20). In year 3, the environmental variables accounted for about 20.69% of the variation within the microbial communities in Lake O and these variables included TN:TP ratio (Pearson $R^2 = 0.36$), nitrate + nitrite (Pearson $R^2 = 0.67$), dissolved oxygen (Pearson $R^2 = 0.30$), alkalinity (Pearson $R^2 = 0.31$), temperature (Pearson $R^2 = 0.36$), total phosphate ("phosphate.ortho"; Pearson $R^2 = 0.44$), *Microcystis* relative abundance (Pearson $R^2 = 0.55$), and chlorophyll a (Pearson $R^2 = 0.39$) (Figure 21). When comparing the environmental variables that influenced microbial community composition across the sampling years, year 1 was the only year in which secchi disk depth influenced microbial community composition (Figure 18). Total phosphate concentration and chlorophyll a concentration were environmental variables shared between year 2 and year 3 that were not included in year 1 that drove microbial community composition (Figures 19 and 20). The environmental variables unique to year 3 in driving the microbial community composition included alkalinity, temperature, and *Microcystis* abundance.

Across the entire sampling period, the microbial community composition of year 3 was closely associated with total phosphate ("phosphate. ortho" in figure 18), nitrate + nitrite, and turbidity (Figure 18). In year 1 and year 3, nearshore and pelagic zones were similar in microbial community composition while inflow and S79 zones were similar in microbial community

composition (Figures 19 and 21). In year 1, the microbial community composition of the nearshore and pelagic zones was driven mostly by nitrate + nitrite, turbidity, and TN:TP ratio, while the communities of the inflow and S79 zones were driven mostly by ammonia (Figure 19). In year 3, the microbial community composition of the nearshore and pelagic zones was driven by nitrate + nitrite, total phosphate, *Microcystis* abundance, chlorophyll-a, and temperature. The microbial community composition of the inflow and S79, however, doesn't seem to be driven primarily by any of the environmental factors shown in the plot (Figure 22). Year 2 had significant differences between stations (Figure 20) and no significant differences between zones (Figure 21). However, each station is located within a certain ecological zone in the lake. Thus, to better interpret the station plot, the zone plot will be used. When looking at the zones of each station, the stations located in the nearshore and pelagic zones were clustered together and mostly driven by nitrate + nitrite concentrations, turbidity, with TN:TP ratio also driving microbial community within the nearshore zone (Figure 20 and figure 22). Stations located in the inflow and S79 zones were also clustered together but there were some stations from the pelagic and inflow zones that were driven by the same environmental variables (chlorophyll a, TN:TP ratio, and ammonia) (Figure 20 and figure 22).

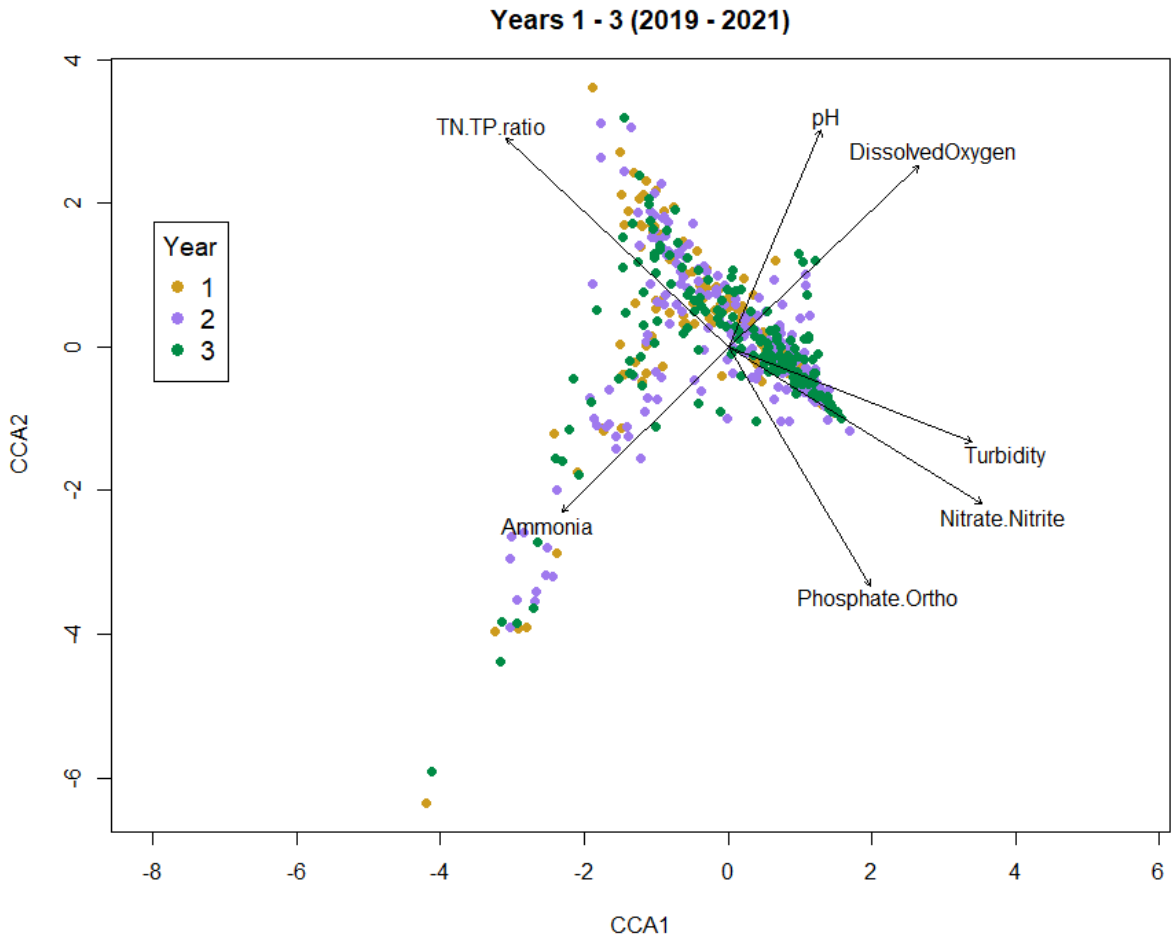

Figure 17. CCA plot based on species composition of each sample over the sampling period by year. Arrows indicate the direction and magnitude of the environmental variables that showed a significant effect ($p<0.05$) and correlation ($R^2 \geq 0.3$).

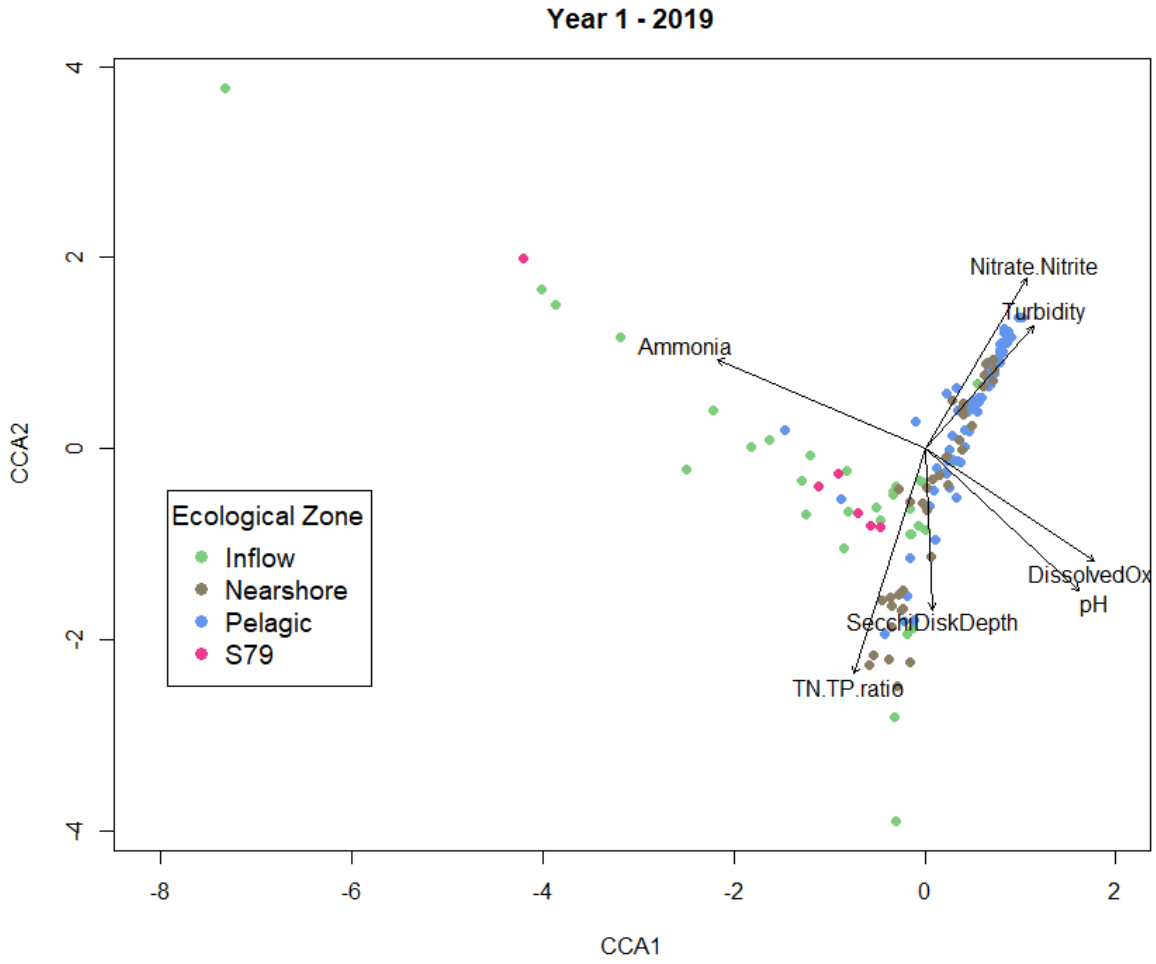

Figure 18. CCA plot based on species composition of each sample in year 1 by zone. Arrows indicate the direction and magnitude of the environmental variables that showed a significant effect ($p<0.05$) and correlation ($R^2 \geq 0.3$).

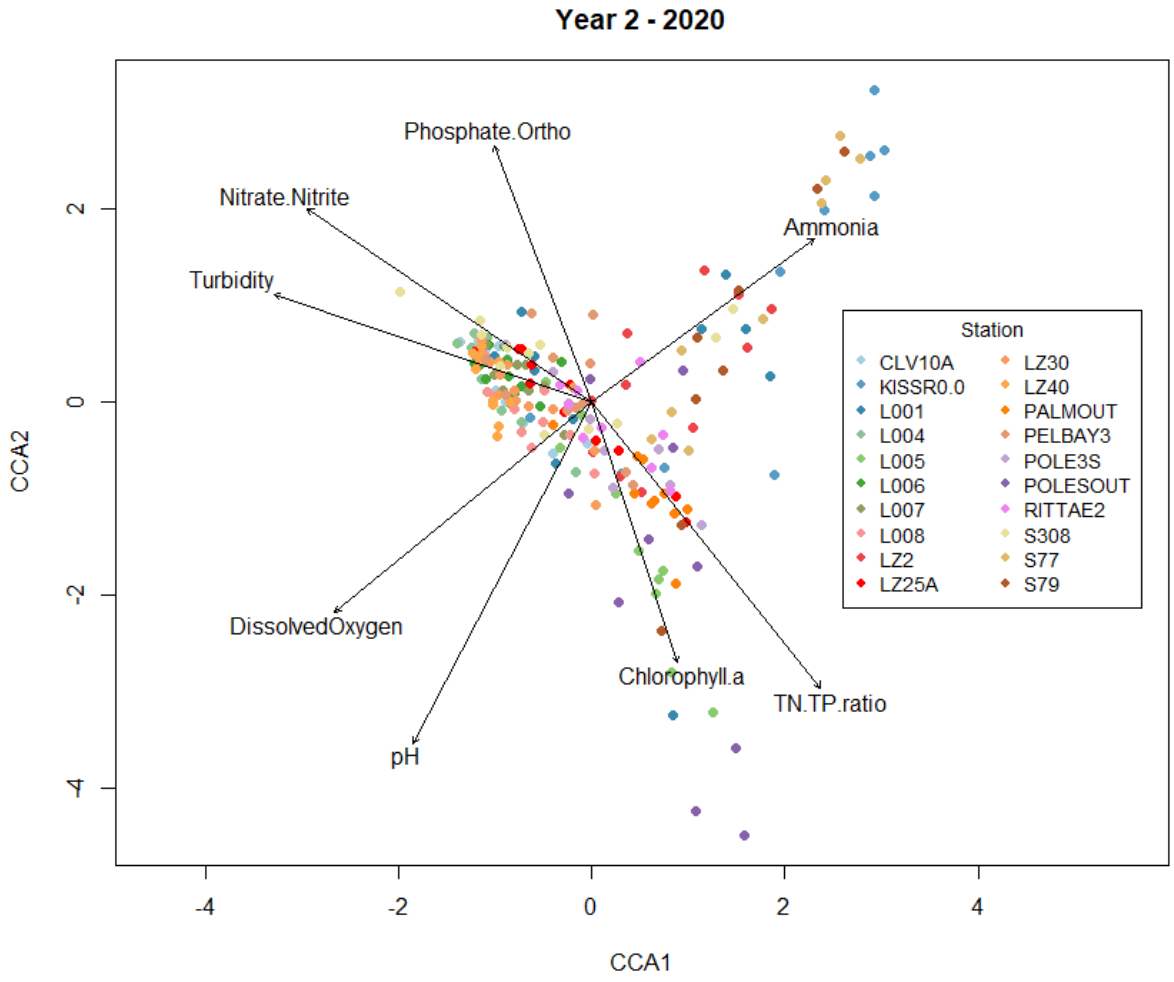

Figure 19. CCA plot based on species composition of each sample in year 2 by station. Arrows indicate the direction and magnitude of the environmental variables that showed a significant effect ($p<0.05$) and correlation ($R^2 \geq 0.3$).

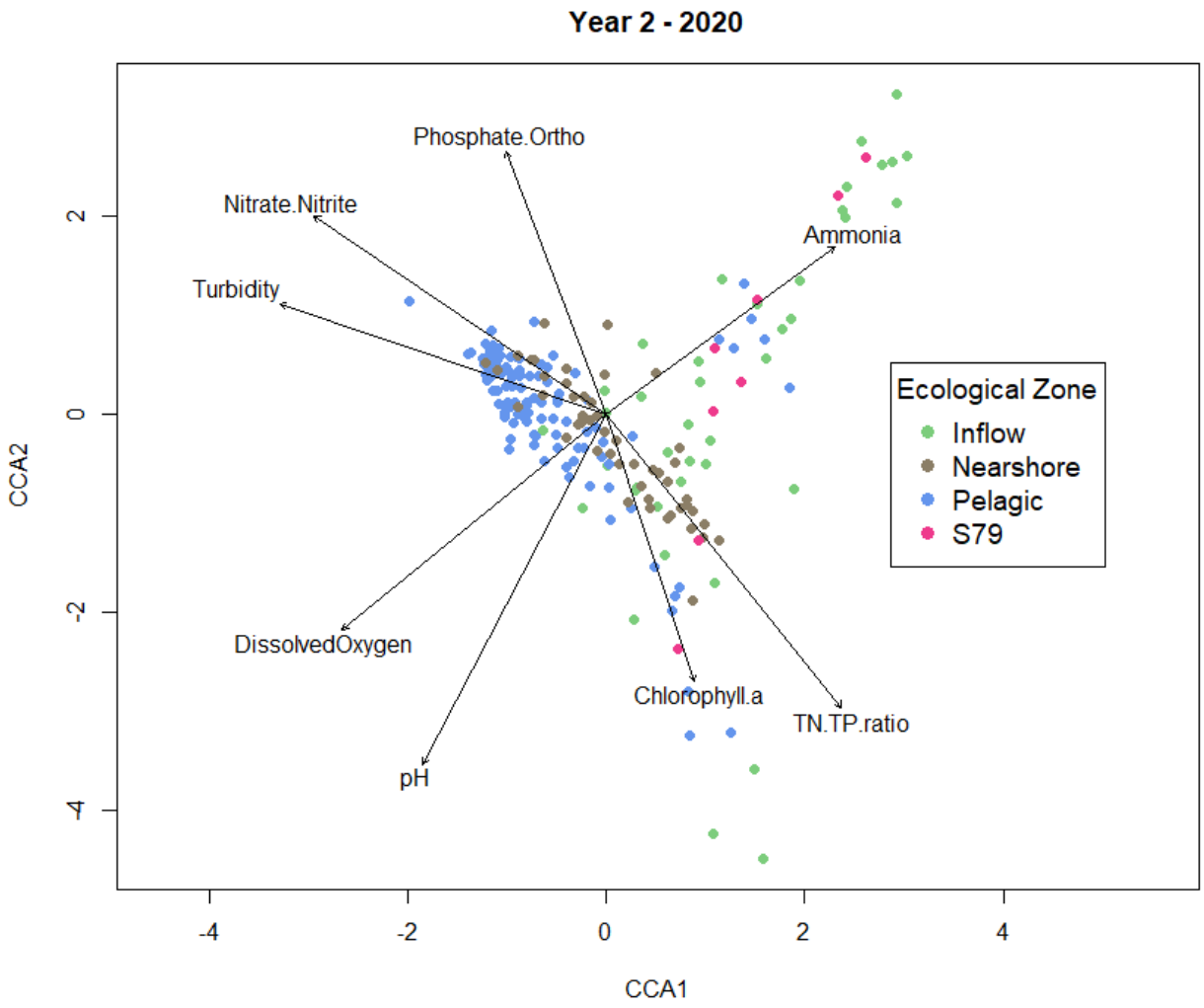

Figure 20. CCA plot based on species composition of each sample in year 2 by zone. Arrows indicate the direction and magnitude of the environmental variables that showed a significant effect ($p<0.05$) and correlation ($R^2 \geq 0.3$).

469

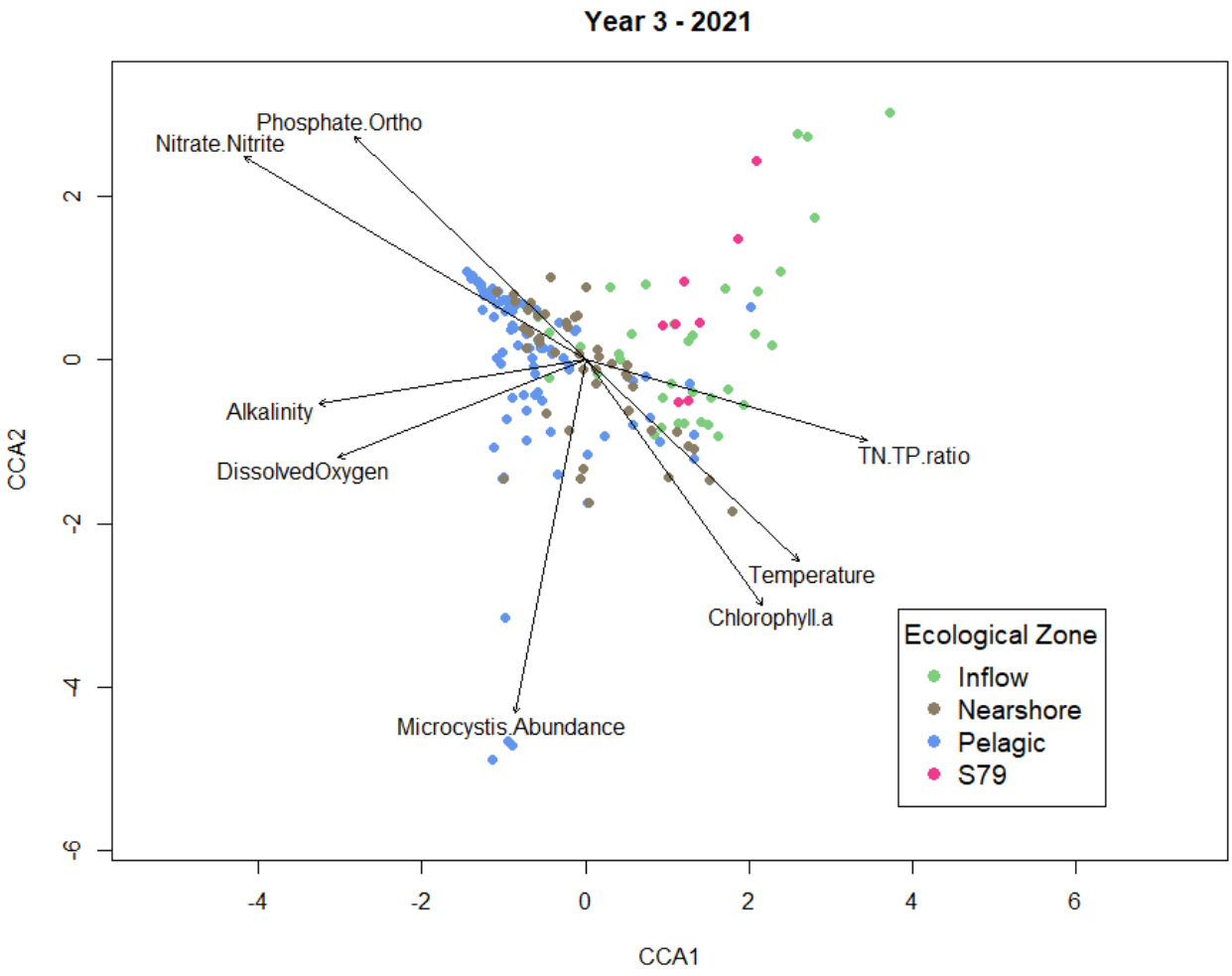

Figure 21. CCA plot based on species composition of each sample in year 3 by zone. Arrows indicate the direction and magnitude of the environmental variables that showed a significant effect ($p<0.05$) and correlation ($R^2 \geq 0.3$).

Co-occurrence network with *Microcystis*

There was a total of 22 bacteria taxa that appeared to co-occur with the genus *Microcystis* (Figure 22). The network consisted of two clusters around *Microcystis*, one with 18 taxa and another with 4 taxa. Most of the bacteria fall under the phylum Proteobacteria with some occurring in other phyla such as Bacteroidota and Gemmatimonadota. The three strongest relationships shared with *Microcystis* were between uncultured bacteria belonging to the family Sutterallaceae (Pearson R = 0.836), the genus *Pseudanabaena_PCC-7429* (Pearson R = 0.811), and the genus *Silanimonas* (Pearson R = 0.807). It is evident that the genus *Microcystis* co-occurs primarily with heterotrophic bacterial taxa, with only two relationships with other Cyanobacteria taxa (Figure 22).

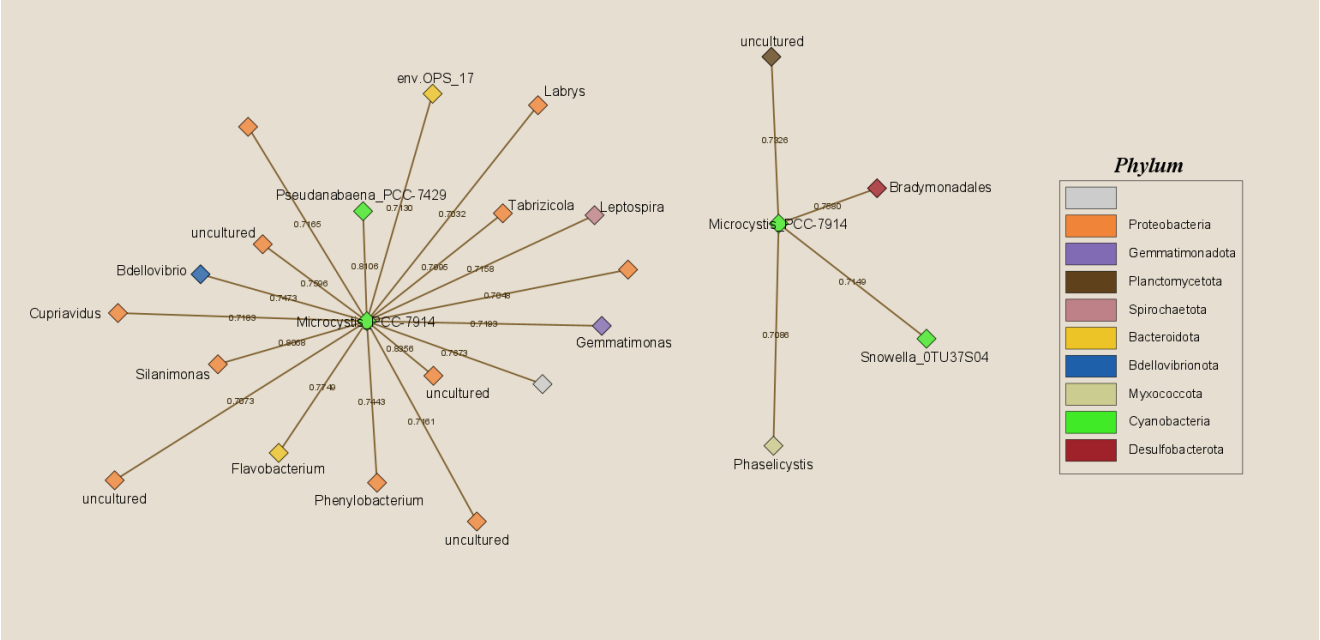

Figure 22. Co-occurrence network of genera sharing a significantly strong positive correlation (p = 0.05; R^2 > 0.7) with the genus *Microcystis*. Node color indicates the phylum corresponding to the genera shown. The numbers shown on the edges of the network signify the R^2 values of the relationship.

Environmental variables over sampling period

After uncovering which environmental variables were in close association with the microbial community beta diversity, selected environmental variables were plotted against the sampling period (by month across the years) (Figures 23-34). The only environmental variable that stayed relative constant with minor changes across the sampling period was pH (Figure 29). However, there were several instances of decreased pH within year 2 and year 3 during the late summer to winter months (7-12) (Figure 29). TN:TP ratio and nitrate + nitrite concentration showed some seasonal changes (Figure 31 and Figure 28, respectively). TN:TP ratio showed a decrease during spring months (3-5) and began to increase into the summer months (6-7) across all three years. Year 1 experienced instances of the highest TN:TP ratio compared to year 2 and year 3 (Figure 31). Nitrate + nitrite concentrations showed an overall decrease in concentration during the summer months into early fall months (6-9) (Figure 28). Year 2 experienced several instances of the highest concentration of nitrate + nitrite compared to year 1 and year 3 (Figure 28).

Most of the remaining selected environmental variables displayed changes from year-to-year. The total depth of Lake O was lower in year 1 while year 2 and year 3 experienced increasing average depths (Figure 33). Year 1 and year 3 experienced warmer water temperatures for a longer period compared to year 2, which exhibited a smoother transition between water temperature gradients across months (Figure 30). Ammonia concentrations remained constant in year 1, with only three instances being substantially higher than average (Figure 24a). Year 3 also portrayed the same pattern; however, there was only one instance where the concentration was substantially above average (Figure 24c). Year 2 showed the most instances that were above average concentrations compared to the other two years (Figure 24b). Both *Microcystis* relative abundance and microcystin concentration were higher during year 2 and year 3 and lowest during year 1 (Figure 27 and Figure 26, respectively). Chlorophyll a concentration exhibited the same pattern—with year 1 exhibiting lower concentrations than year 2 and year 3 (Figure 25). Year 1 and year 3 exhibited an unstable increase-decrease cycle in total nitrogen concentration across the monthly averages, while year 2 experienced only two increase averages during March and November (Figure 32). Total phosphorus also experienced this pattern in concentration (Figure 23). The average concentration of total phosphate stayed within the same range across the years until it began to decrease during July of year 3 (Figure 34).

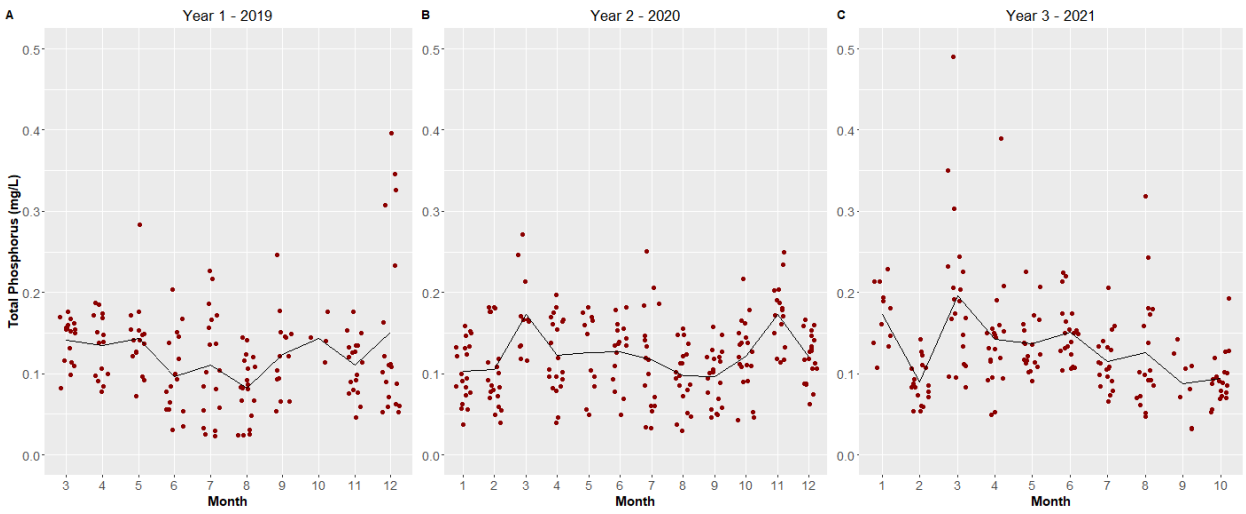

Figure 23. Scatterplot of total phosphorus concentrations (mg/L) over the sampling period. The black line depicts the average concentration per month across the years.

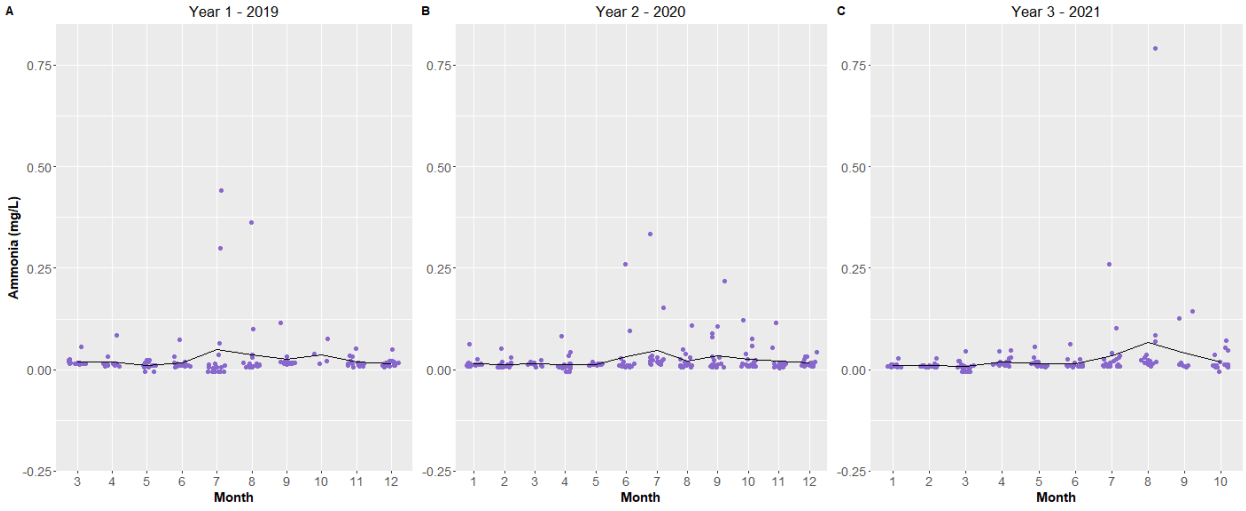

Figure 24. Scatterplot of ammonia concentrations (mg/L) over the sampling period. The black line depicts the average concentration per month across the years.

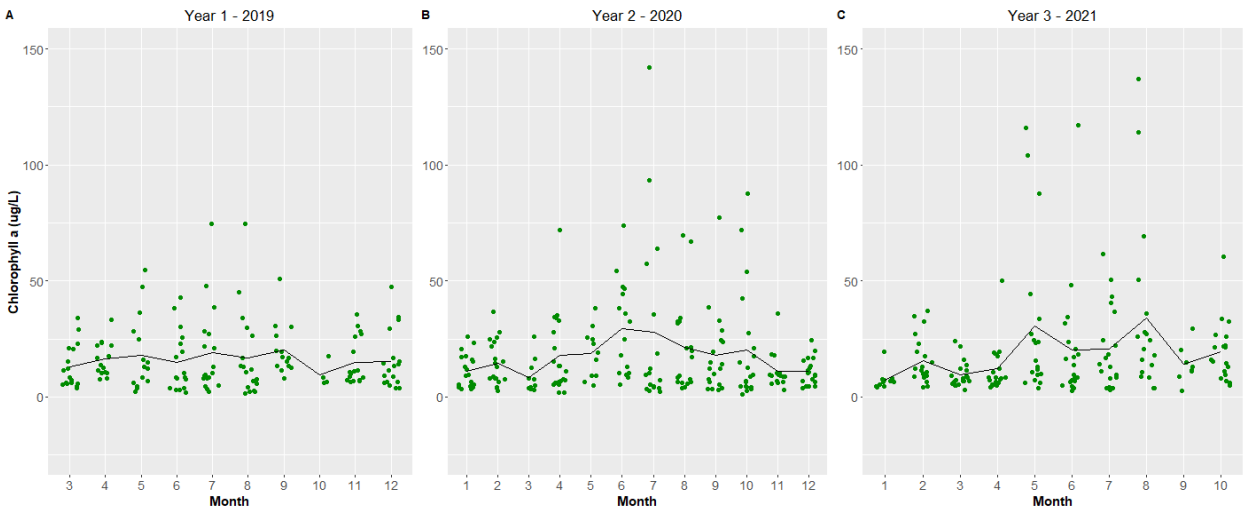

Figure 25. Scatterplot of total chlorophyll a concentration (µg/L) over the sampling period. The black line depicts the average concentration per month across the years.

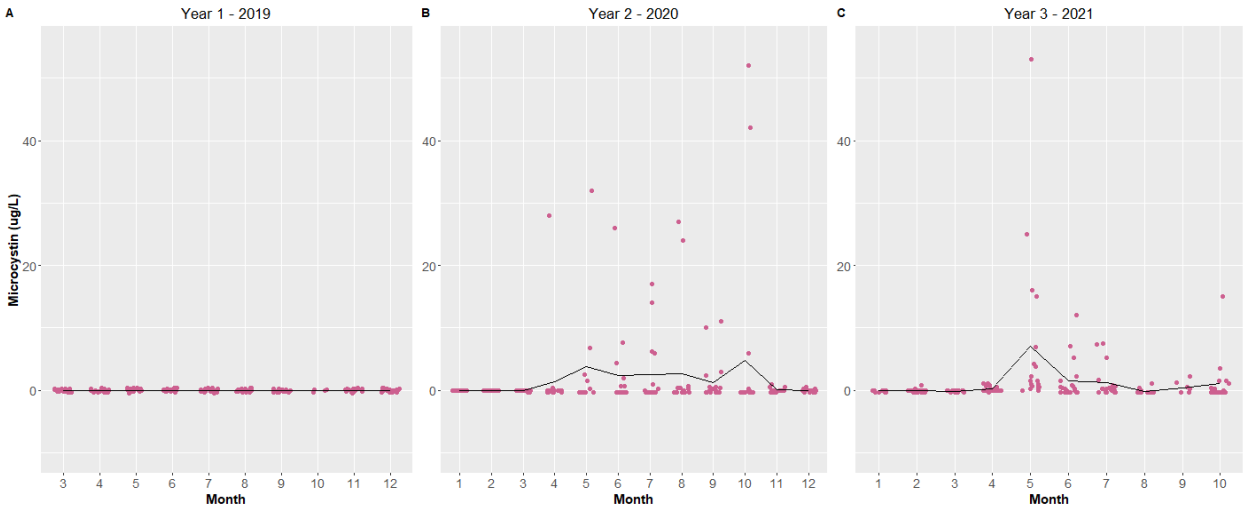

Figure 26. Scatterplot of microcystin concentrations (µg/L) over the sampling period. The black line depicts the average concentration per month across the years.

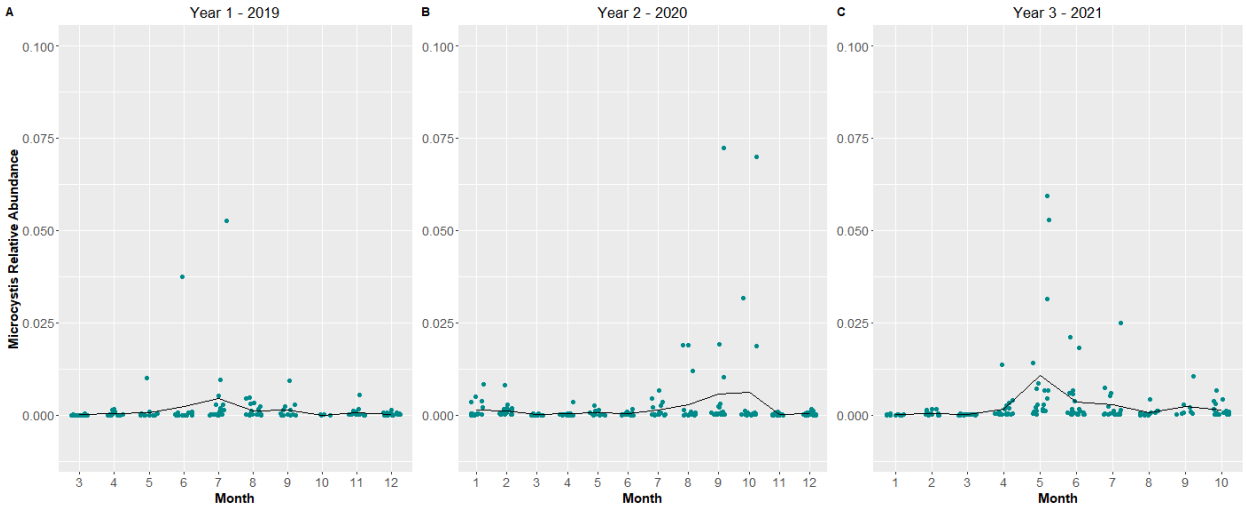

Figure 27. Scatterplot of *Microcystis* relative abundance over the sampling period. The black line depicts the average abundance per month across the years.

Figure 28. Scatterplot of nitrate + nitrite concentration (mg/L) over the sampling period. The black line depicts the average concentration per month across the years.

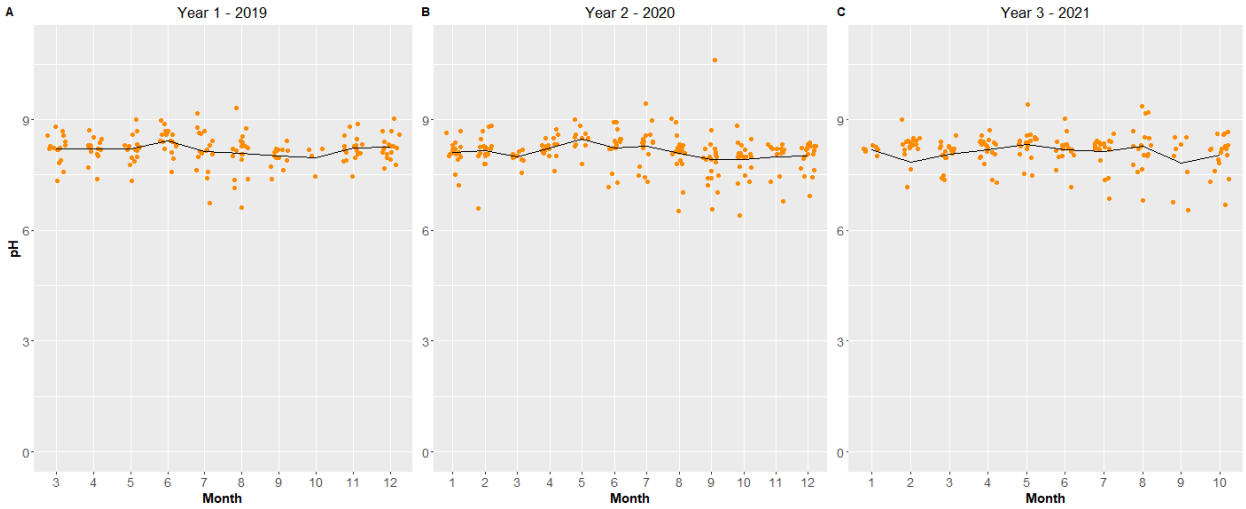

Figure 29. Scatterplot of surface water pH over the sampling period. The black line depicts the average pH per month across the years.

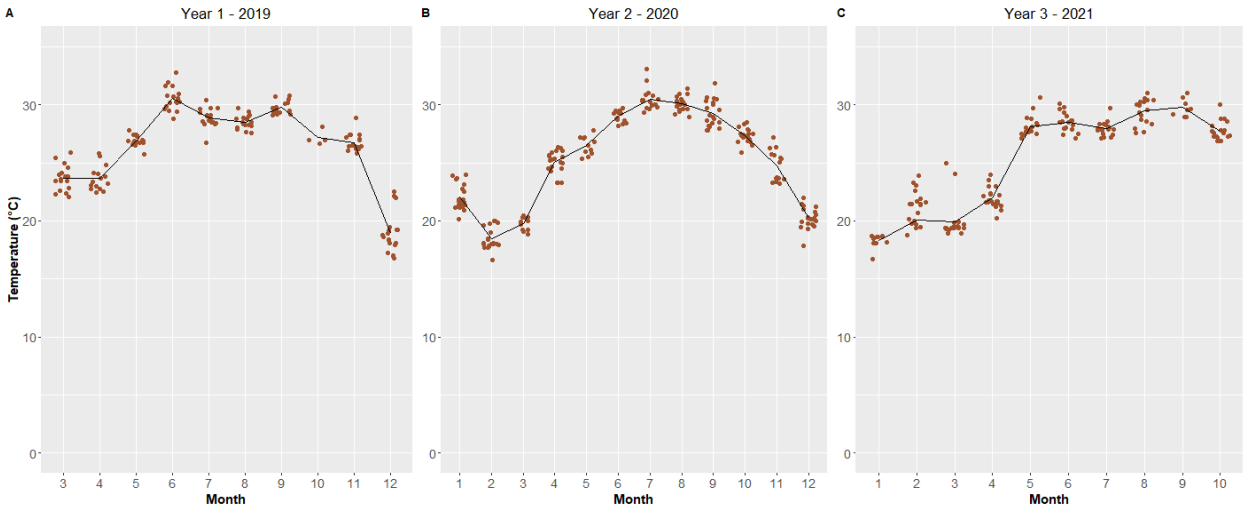

Figure 30. Scatterplot of surface water temperature (°C) over the sampling period. The black line depicts the average temperature per month across the years.

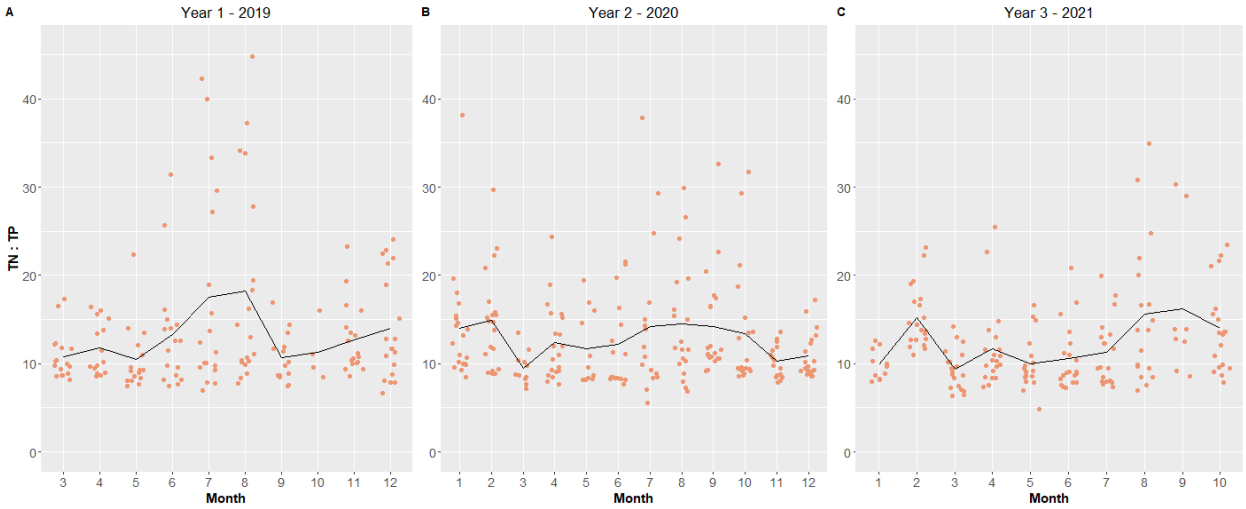

Figure 31. Scatterplot of the ratio of total nitrogen and total phosphorus over the sampling period. The black line depicts the average ratio per month across the years.

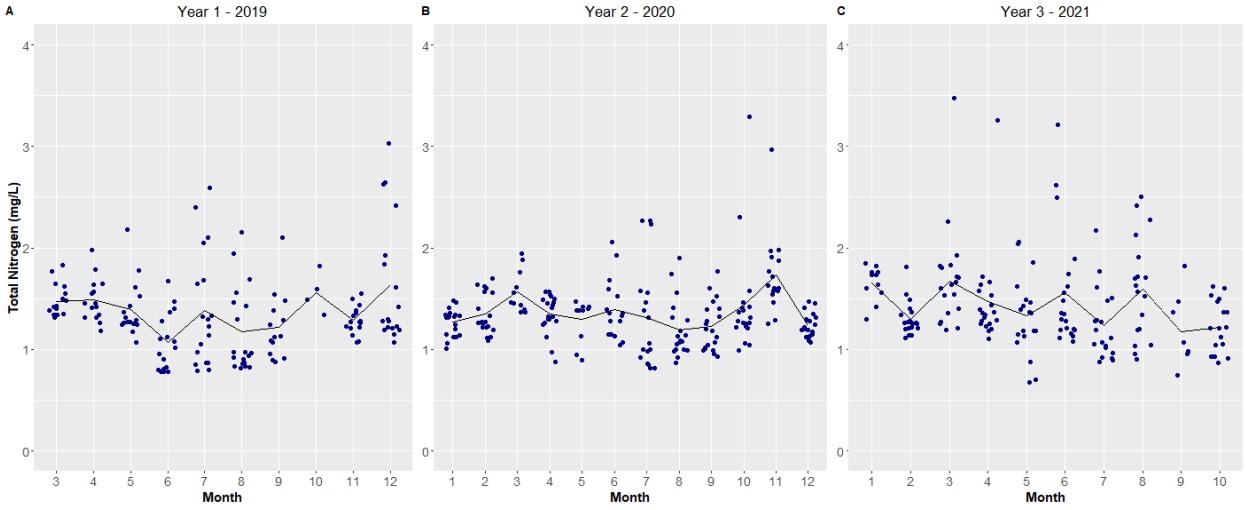

Figure 32. Scatterplot of total nitrogen concentrations (mg/L) over the sampling period. The black line depicts the average concentration per month across the years.

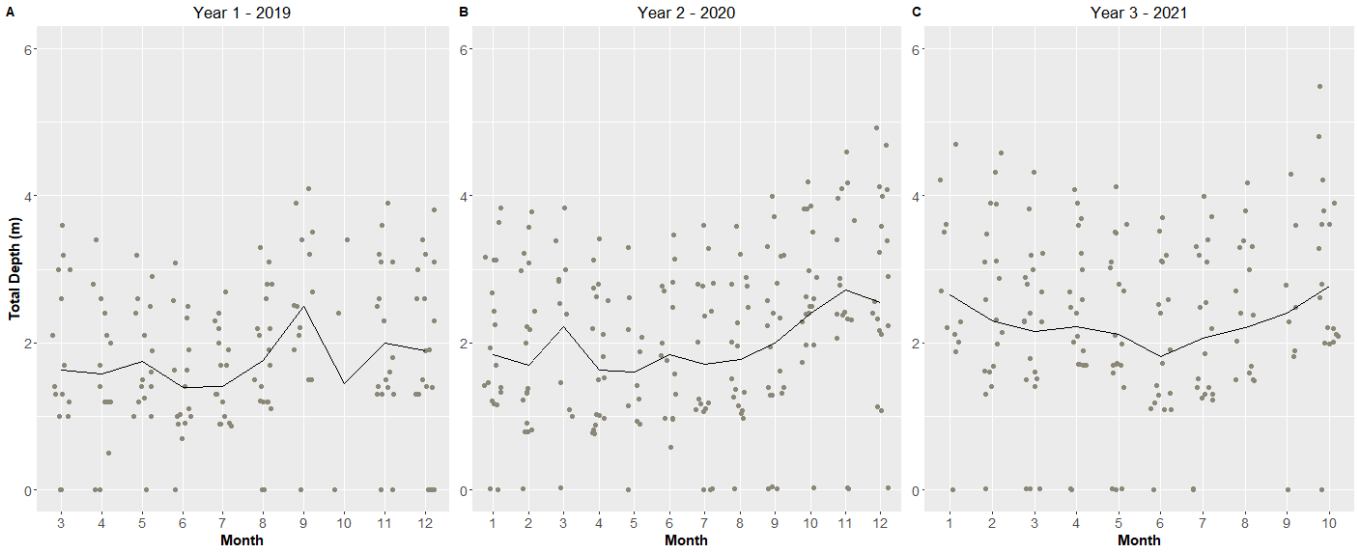

Figure 33. Scatterplot of the total depth (m) of the lake over the sampling period. The black line depicts the average depth per month across the years.

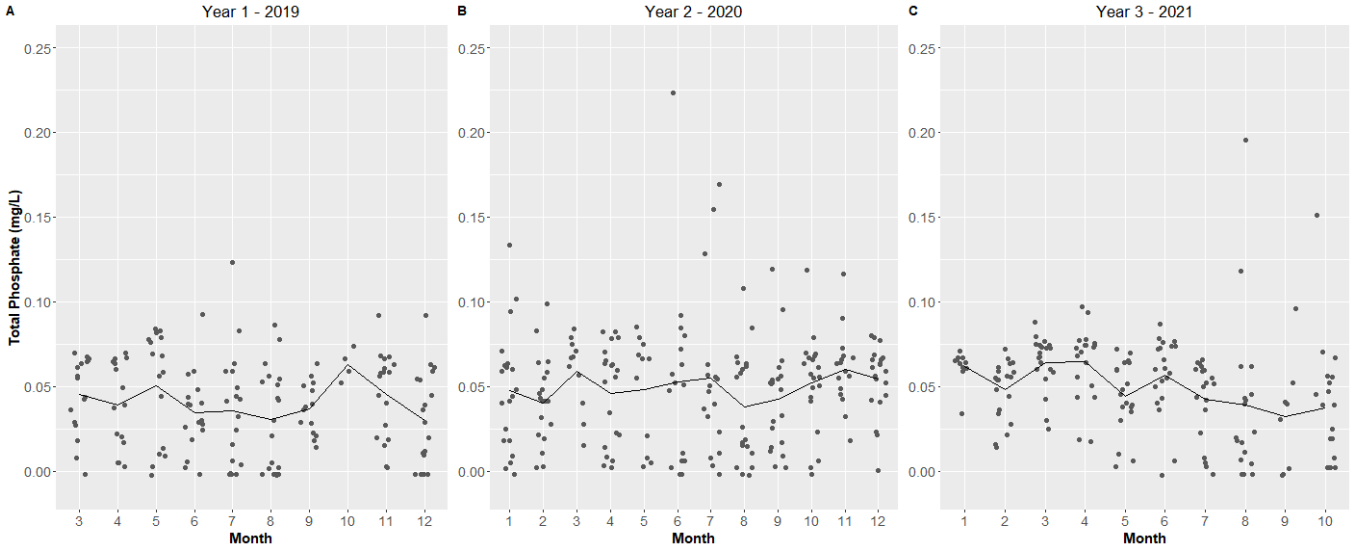

Figure 34. Scatterplot of the total phosphate (mg/L) concentration over the sampling period. The black line depicts the average concentration per month across the years.

Discussion

Bloom effects on microbial community diversity

Most of the cyanobacterial harmful algal bloom (cyanoHAB) research done on Lake Okeechobee (Lake O) primarily focuses on bloom management via the control of nutrients going into the lake. However, there is a growing amount of research suggesting that nutrient levels may

not be the only factor influencing these blooms to occur so frequently (Wilhelm *et al.*, 2020). There have not been many studies done on Lake O that assess how these cyanoHABs are affecting the other microbial communities within the lake during these blooms or how these other microbes could be influencing the blooms. The conclusions reached in this study provide a glimpse into the effects of cyanoHABs caused by *Microcystis* may have on the microbial community make-up within Lake O.

This study has found that the diversity of microbial communities in Lake O are affected by the occurrence of *Microcystis*, one of the main cyanobacteria genera causing cyanoHABs both in Lake O and around the world. The microbial communities within Lake O appeared to show both temporal and spatial differences in diversity. However, more significant differences were found between stations and ecological zones within all three years together and between each year. This result was expected due to the different environmental conditions experienced by the ecological zones found throughout the lake. *Microcystis* is known to "lie-in-wait" for the proper environmental conditions that are favorable for their populations to proliferate and bloom; they even tend to overwinter in the sediments at the bottom of the lake until these conditions are present (Cai *et al.*, 2021; Reynolds, 1973). Over the three sampling years (2019-2021), there was an evident increase in bloom intensity and longevity. The peak average relative abundance of *Microcystis* and the average concentration of microcystin could be seen increasing over the years with year 3 (2021) experiencing the highest abundance and concentration (Figures 27 and 26, respectively). There were also changes in environmental conditions within 2021 that may have contributed to the increase of bloom intensity. For instance, 2021 was seen to have warmer average temperatures and a lower TN:TP ratio during the months (May to July) that blooms occurred (Figures 30 and 31, respectively). Numerous studies have shown that cyanobacteria favor higher temperatures thus increasing their growth rates during warmer periods of the year (Wilhelm *et al.*, 2020; Paerl & Hulsman, 2008; Jöhnk K. D., *et al.*, 2008; Reynolds, 2006). Xie *et al.* (2003) uncovered that when *Microcystis* populations were exposed to sufficient amounts of nitrogen (N) but differing amounts of phosphorus (P), *Microcystis* blooms occurred only in the environments with higher P concentrations. However, as these blooms progressed, both N and P concentrations declined, hence resulting in lower TN:TP ratios. Therefore, as an increase in temperature influences the growth of *Microcystis* blooms, there is a decrease in TN:TP ratio due to the increases use of the nutrients in the water column.

Beta diversity patterns of the microbial community composition

There were some evident spatial patterns throughout the data. The spatial variables of interest in this study were the monitoring stations in the lake and the ecological zones of the lake. When looking at the ecological zones of the lake, there was an obvious coupling between the zones: the inflow zone was always coupled with the zone S79, and the pelagic zone was always coupled with the nearshore zone; giving the idea that these couples have similar microbial community composition. As mentioned in a previous study, although these zones exhibit differing physiochemical properties, these zones do not have clearly defined borders between them, hence these zones can be dynamic (Krausfeldt *et al.*, submitted). The results of this study further supported this concept as 2020 (year 2) showed no significant differences between zone when 2019 and 2021 (year 1 and year 3, respectively) did show significant differences; showing that there was less of a differentiation between zones in 2020 compared to the other years. However, the members of each coupling did not come to a surprise as the zone S79 is within the

Caloosahatchee River, which has a mouth into the lake, so it is in contact with the inflow zone of the lake. Additionally, the pelagic and nearshore zones also come into contact with one another despite their physiochemical differences.

Rare microbial taxa in Lake Okeechobee

The taxonomic make-up of Lake O was dominated primarily by four common bacterial phyla: Proteobacteria, Bacteroidota, Cyanobacteria, and Actinobacteriota (Table 1, Figure 3). These phyla appeared to change in distribution, along with the less-dominant taxa present, both temporally (Figure 3) and spatially (Figures 5-7). However, there were some phyla that irregular in both their distribution around the lake and their presence across the years. In 2019 (year 1), there was one phylum that appeared in the top phyla of only two stations within Lake O and was found in no other year—SAR324 (marine_clade group B). SAR324 is a novel phylum that has been recently classified as its own phylum after initially being classified as "marine_clade group B" under the phylum Deltaproteobacteria (Malfertheiner *et al.*, 2022; Parks *et al.*, 2018; Pommier *et al.*, 2005). SAR324 is known to be present only in marine environments; however, Malfertheiner and colleagues (2022) discovered that this phylum can also be found in terrestrial aquifers. (Malfertheiner *et al.*, 2022) Lake O could possibly be subjected to saltwater intrusion (Prinos, 2016; Barlow & Reichard, 2010), or the movement of seawater into freshwater aquifers, due to the water level being heavily managed. The SFWMD stated that saltwater intrusion is at a higher risk of occurring in Lake O starting at a depth of 10½ feet (or 3.2 meters) and compromising the Caloosahatchee lock at a starting depth of 9½ feet (or 2.9 meters) (SFWMD, "Impacts of Operating Lake Okeechobee at Lower Water Levels"). Yet, throughout the majority of 2019, the total depth of Lake O was sustained between about 1 and 3 meters (3.3 feet and 9.8 feet). These conditions put Lake O in the position of the increased risk of saltwater intrusion, especially at the Caloosahatchee River lock (station S79). Coincidentally, SAR324 appears as one of the dominant taxa in stations S79 and POLESOUT (Figure S2); thus, whether SAR324 appears due to saltwater intrusion, or it is naturally occurring in the terrestrial aquifer is unknown.

A non-ubiquitous phylum that was found in 2020 and no other year was Armatimonadota (Figure S3). This phylum was part of the top phyla within the station, KISSR0.0, which is located in the inflow zone and the mouth of the Kissimmee River (Figure 1). Armatimonadota was originally known as candidate phylum OP10 before its reclassification into a new phylum by Hugenholtz and colleagues in 1998 (Hugenholtz *et al.*, 1998b). Isolated sequences of Armatimonadota were isolated from a variety of environments such as aerobic and anaerobic wastewater treatment processes, contaminated and regular soil and sediments (Im *et al.*, 2012). Lake O and its connecting rivers, St. Lucie, Kissimmee, Caloosahatchee, etc. all are experiencing nutrient pollution due to the agricultural and urban lands surrounding them. Furthermore, between 2019 and 2020, there was an increase in the average concentrations of total phosphate (Figure 34), total nitrogen (Figure 32), nitrate + nitrite (Figure 28), and total phosphorus (Figure 23). Hence, it is unknown what kind of contamination occurred during the initial collection and isolation of the bacteria belonging to Armatimonadota, but there may be a connection with the increase in nutrient pollution and the presence of this phyla.

An additional non-ubiquitous phylum, Patescibacteria, appeared only in 2021 at two stations within the lake (Figure S4). Patescibacteria, formerly known as the 'candidate phyla radiation'(CPR), included the discovery of an immense microbial diversion within the bacterial

tree of life in 2016 (Herrman *et al.*, 2019). However, in 2018, Parks *et al.* (2018) suggested classifying the CPR into a new phylum, Patescibacteria. There are 14 classes of bacteria known so far in this phylum and they all inhabit a range of environments including groundwater and other aquifer environments, freshwater sediments, and deep-sea sediments (Herrman *et al.*, 2019; Proctor *et al.*, 2018; Leon-Zayas *et al.*, 2017; Luef *et al.*, 2015; Brown *et al.*, 2015). There is a high abundance of Patescibacteria that found in groundwater environments—making up around 38% of the total microbiomes (Herrmann *et al.*, 2019; Bruno *et al.*, 2017; Kumar *et al.*, 2017). In Lake O, Patescibacteria were found only in 2021 (year 3) at two stations, L004 and L006, both of which are in the pelagic zone of the lake. The pelagic zone is the deepest part of the lake but also experiences the most turbidity (Krausfeldt *et al.*, submitted). The higher turbidity and reduced water clarity of the water column suggests that there may be sediment resuspension occurring within the pelagic zone (Krausfeldt *et al.*, submitted), thus possibly allowing this phylum to be collected in surface waters.

Bacterial co-occurrences with *Microcystis*

It is well-known that *Microcystis* blooms are influenced by abiotic factors such as environmental variables and nutrient inputs of freshwater ecosystems. There has been increasing curiosity of how the heterotrophic bacterial community plays a role in the aggregation and proliferation of the colonies and how they could be maintaining these cyanobacterial harmful algal blooms (cyanoHABs) created by *Microcystis*. Studies have shown evidence that there are heterotrophic bacteria that live within and surrounding *Microcystis* colonies, with either mutualistic or antagonistic effects (Tu *et al.*, 2019; Shen *et al.*, 2011; Shi *et al.*, 2009; Maruyama *et al.*, 2003; Imamura *et al.*, 2001; Pankow, 1986). As mentioned previously, several results in this study suggested that *Microcystis* can alter the microbial community of Lake O through cyanoHABs. Both *Microcystis* and its related toxin, microcystin, showed strong negative correlations to species evenness and species diversity (Figure 8). In year 3 (2021)—the year with the most intense blooms of the entire sampling period—*Microcystis* appeared as one of the strongest correlated variables, along with other environmental variables, to drive variation in the microbial communities in Lake O (Figure 21). After revealing that *Microcystis* can alter the microbial communities, the curiosity of knowing who else can possibly be changing with *Microcystis* resulted in the creation of a co-occurrence network involving any bacteria that has appeared with this genus. The co-occurrence network showed 22 significantly strong positive correlations between *Microcystis* and other heterotrophic bacteria; with two exceptions being cyanobacteria (*Pseudanabaena*_PCC-7429 and *Snowella*_OTU37S04) (Figure 22). Although some negative correlations did exist between *Microcystis* and other bacteria, their relationships were not strong enough to document as strong correlations (R2 = -0.7 or less).

Some of the heterotrophic bacteria genera that co-occur with *Microcystis* may indicate that there is a commensal relationship between them. *Bradymonadales* belongs to the phylum Desulfobacterota which is located under the phylum Deltaproteobacteria. *Bradymonadales* are predatory bacteria, which is broken up into two categories, obligatory and facultative (Mu *et al.*; 2020). Mu and colleagues (2020) found that *Bradymonadales* displays unique living strategies that allow for these bacteria to present a novel method of predation: a transition between being obligate and facultative predators. Some of the main bacteria that are highly preyed on by *Bradymonadales* include Bacteroidetes, Flavobacteria, and Proteobacteria. Intriguingly, 11 of the 22 co-occurring bacteria with *Microcystis* belong to the phylum Proteobacteria with an additional two belonging

to Bacteroidetes and Flavobacteria. Thus, *Bradymonadales* may be utilizing *Microcystis* colonies during the blooms as a feeding ground for its prey items. *Bdellovibrio exovorus* is another predatory bacteria species that was seen to co-exist with *Microcystis*. First described in 1963 (Koval *et al.*, 2013; Stolp & Starr, 1963), *Bdellovibrio exovorus* belongs to a group of like predatory bacteria known as Bdellovibrio and like organisms (BALOs) (Ezzedine *et al.*, 2020). BALOs were the first records of predatory bacteria and continue to be used as a baseline for the discovery of novel predatory bacteria like *Bradymonadales* which was previously mentioned above. Similar to *Bradymonadales*, *B. exovorus* is also obligatory predators on primarily other Proteobacteria. However, it is important to note that some species of BALOs have been found to kill cyanobacterial cells. Caiola and Pellegrini (1984) found that BALOs were able to lyse *Microcystis aeruginosa* cells via penetration and proposed that these and other algicidal bacteria could be the reason for the dying out of cyanobacteria bloom events.

There were only two taxa that were not heterotrophic bacteria that shared strong positive correlations with *Microcystis*, genera *Pseudanabaena*_PCC-7429 and *Snowella*_OTU37S04, which are also part of the phylum Cyanobacteria. The genus *Pseudanabaena* is an epiphytic cyanobacterial taxon that is commonly found embedded within or attached to the mucilaginous sheath of *Microcystis* colonies (Li *et al.*, 2020). Both taxa are frequently observed to be highly correlated during cyanoHABs and this study also provides evidence of this pattern (Li *et al.*, 2020; Berry *et al.*, 2017; Ilhe, 2008). In the 1980s, *Pseudanabaena* was primarily described as a parasitic organism to *Microcystis* colonies (Chang, 1985; Gorham *et al.*, 1982). Further investigation was conducted regarding the interactions between *Pseudanabaena* and *Microcystis*, which investigated the interaction directly (Agha *et al.*, 2016). Agha and colleagues (2016) discovered that *Pseudanabaena* is not selective on the species of *Microcystis* but on their mucilage structure. They also uncovered that *Pseudanabaena* is detrimental to *Microcystis* colonies both directly via cell lysis and indirectly via cell sedimentation. Thus, it may be possible that *Pseudanabaena* may also contribute to the dying out of cyanoHAB events. Conversely, although the genus *Snowella* was also found to be highly correlated to *Microcystis* in a previous study, not much is known about their ecology and their interaction with *Microcystis* (Mankiewicz-Boczek & Font-Nájera, 2022).

Another interesting taxa that was highly correlated with *Microcystis* is the genera env.OP_17 (Figure 22). There is not much information solely about the bacterium env.OP_17, however, it is part of the order Sphingobacteriales and this order is known to be potential algicidal bacteria that favor the uptake of cyanobacterial excretions and decaying material (Mankiewicz-Boczek & Font-Nájera, 2022). Furthermore, Mankiewicz-Boczek & Font-Nájera (2022) found that env. OP_17 increased in abundance after a bloom, suggesting that this taxon might be a part of the "clean-up team" once a cyanoHAB dies out. Though this study presented results focused primarily on the highly correlated relationships between other bacteria and *Microcystis* in Lake O, there was another bacterial genus, Streptomyces, that is known to exhibit algicidal activity towards *Microcystis* that was present in microbial community of Lake O (Zhang *et al.*, 2023). On the contrary, the genus *Phenylobacterium*—another taxon that was found with a high correlation with *Microcystis* (Figure 22)—was found to aid in the growth and dominance of toxic *Microcystis* strains during cyanoHAB events. As mentioned previously, there are toxic and non-toxic bloom-forming strains of *Microcystis* and in a study conducted by Zuo *et al.* (2021), they saw that *Phenylobacterium* was one of the few genera that strongly positively co-existed with toxic strains of *Microcystis*. After further investigation in the field and in the laboratory, they found that there were three strains of *Phenylobacterium* that promoted the growth of these toxic strains of

Microcystis, suggesting that *Phenylobacterium* may be a heterotrophic bacterium that could be aiding in the longevity of these blooms (Zuo *et al.*, 2021). Unfortunately, there needs to be further investigation into the mechanisms by which *Phenylobacterium* interact with these toxic strains of *Microcystis* that allow *Microcystis* to remain dominant throughout the cyanoHAB event.

Microcystis, temperature, pH, and nutrients

Although it is also important to investigate the biotic factors that influence cyanoHABs, such as the interactions between the blooming cyanobacteria and other microbes, there is still plenty of evidence of how abiotic factors influence cyanoHABs, and vice versa, all over the world. During this study, in addition to characterizing the microbial community of the lake, certain environmental variables were also collected to consider how these variables could be influencing these blooms along with the microbial community. Besides nutrient levels in the lake, one important physical characteristic that affects cyanoHABs is temperature. Temperature affects the growth of cyanobacterial species. In general, higher temperatures promote the growth of cyanobacteria, often temperatures that are above 25°C (Paerl & Huisman, 2008; Jöhnk *et al.*,2008; Reynolds, 2006). When temperatures increase, the water column becomes more stable and stratified since the increase in temperature weakens the amount of vertical mixing in the water column (Paerl & Huisman, 2008; Paerl & Fulton III, 2006; Reynolds, 2006; Husiman, Matthijs, & Visser, 2005). *Microcystis aeruginosa*, the dominant bloom-forming cyanobacteria species in Lake O, can take advantage of these more stratified conditions using their gas vesicles. The gas vesicles formed by M. aeruginosa give them the buoyancy they need to effectively migrate through the water column during favorable conditions, such as high temperatures and increased light availability (Dick *et al.*, 2021; Huisman *et al.*, 2018; Komárek, 2003). This buoyancy also provides M. aeruginosa the ability to form "mats" of biomass at the surface of the water; hence, cyanoHAB events tend to increase in frequency in the summer (You *et al.*, 2017; Litchman *et al.*, 2010). Across the sampling period, especially in 2021, temperatures reached between 25°C and 30°C each year from May through to September—around the same months where microcystin concentrations (Figure 26) and *Microcystis* relative abundances (Figure 27) were the highest (Figure 30). Certainly, global warming is becoming a concerning topic as increasing temperatures are affecting the environments of the planet. Further research should be done on Lake O and other lakes affected by cyanoHABs to look at the trend of bloom frequencies as the global temperature continues to rise over time.

In addition to rising temperatures, pH is also known to be a factor associated with *Microcystis* blooms. This importance was evident as pH was included as an environmental factor driving the differences found in the microbial community composition across the sampling period (Figure 17). During a dense bloom, the cyanobacteria rapidly consume inorganic carbon (in the form of dissolved CO¬¬2) that is available in the upper water column, in turn increasing the pH of the surface water to above 9 (Ji *et al.*, 2020; Wilhelm *et al.*, 2020). Across the sampling period, there were an increasing number of instances where the surface water pH was measured above 9 (Figure 29). With this increase in pH, the equilibrium of carbon in the water is shifted from inorganic carbon (dissolved CO¬¬2) to bicarbonate (HCO3-) and carbonate (CO32-) (Ji *et al.*, 2020; Huisman *et al.*, 2018). *Microcystis*, although also adaptive to high concentrations of CO2 concentrations, can utilize bicarbonate as a carbon source through the use of carbonic anhydrase found in cyanobacteria—further allowing these blooms to thrive during these alkaline conditions (Ji *et al.*, 2020; Wilhelm *et al.*, 2020; Huisman *et al.*, 2018). Alkaline pH conditions also allow for

the conversion of ammonium ions (NH_4^+) to ammonia (NH_3). During the months where microcystin concentrations (Figure 26) and *Microcystis* relative abundances (Figure 27) were the highest (May to September), there was also an increase in ammonia during those months.

Conclusion

This study provides a glimpse into the effects of cyanoHABs within the microbial community of the freshwater lake, Lake Okeechobee. This study provides an initial look into the taxonomic classification of the dynamic microbial community of Lake O over several years and the spatial changes that were seen within these communities. We found that the cyanoHABs that have been commonly occurring in Lake O do in fact alter the microbial community composition of the lake. Further investigation of these changes within the microbial community composition yielded the identification of possible relationships between these microbial communities and *Microcystis*. With the identification of these possible relationships, future investigation should be conducted to see how the functions of these taxa are incorporated into their interaction with *Microcystis*. With that, we might be able to identify bacteria that may serve as possible bioindicators for these cyanoHAB events and aid in preventing or managing these recurring blooms in the lake.

Lake Okeechobee is indeed an essential part of south Florida's ecosystems as it serves as a source of drinking water for nearby towns, irrigation for the agricultural lands surrounding the border of the lake, critical water supply for the environment, and as habitat for various organisms in the water and on the land (South Florida Water Management District (SFWMD)). With the degrading water quality of the lake, there is great concern for life both within and around the lake. To date, numerous studies have been conducted on reducing the nutrient loading into the lake (Canfield Jr. *et al.*, 2021; Schelske, 1989; Canfield Jr. & Hoyer, 1988) and investigating the possible control of these recurring blooms (Pokrzywinski *et al.*, 2022), primarily focusing on the cyanobacteria involved in these blooms. Not many studies have been done on Lake Okeechobee that explore the taxonomic structure, temporal distributions, and spatial distributions of the microbial communities before, during, and after annual cyanoHABs. Furthermore, whether the microbial community taxonomic structure, temporal and spatial distributions rebound after a bloom event also has yet to be studied.

To enable scientists to enhance their comprehension of the ongoing cyanoHABs in Lake Okeechobee and their interactions with the surrounding environment, particularly the microbial community, it is essential to fill these existing knowledge gaps. With that scientists will be able to examine the variations in the diversity and trophic structure of the lake before, during, and after the occurrence of these harmful blooms—bringing scientists closer to fully understanding the impact of cyanoHABs on Lake Okeechobee's microbial communities.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

PS wrote the manuscript, assisted with sample processing and sequencing, performed data analysis, and generated the figures. LK assisted with sample processing and sequencing. All authors contributed to editing the manuscript.

Funding

This work was funded by the US Army Corps of Engineers' Engineer Research and Development Center (ERDC) and facilitated by South Florida and Caribbean Cooperative Ecosystems Studies Unit (SFC CESU).

Acknowledgments

We would like to thank South Florida Water Management District and USGS for collecting and filtering the water samples. We would like to also thank the high school, undergraduate, and graduate students that helped tremendously with sample processing and sequencing across the project years

References

1. Agha, R., Del Mar Labrador, M., De Los Ríos, A., & Quesada, A.. (2016). Selectivity and detrimental effects of epiphytic *Pseudanabaena* on *Microcystis* colonies. *Hydrobiologia*, *777*(1), 139–148. doi:10.1007/s10750-016-2773-z
2. Anderson, D. M. (2009). Approaches to monitoring, control, and management of harmful algal blooms (HABs). *Ocean Coast Manag.* doi:10.1016/j.ocecoaman.2009.04.006
3. Barlow, P., & Reichard, E. (2010). Saltwater intrusion in coastal regions of North America. *Hydrogeology Journal*, *18*, 247–260. doi:10.1007/s10040-009-0514-3
4. Berry, M. A., Davis, T. W., Cory, R. M., Duhaime, M. B., Johengen, T. H., Kling, G. W., Marino, J. A., DeuUyl, P. A., Gossiaux, D., Dick, G. J. & Denef, V. J. (2017). Cyanobacterial harmful algal blooms are a biological disturbance to western Lake Erie bacterial communities. *Environ. Microbiol*. 19:1149-62.
5. Bláha, L., Babica, P., & Maršálek, B. (2009). Toxins produced in cyanobacterial water blooms - toxicity and risks. *Interdisc. Toxicol., 2*. doi:10.2478/v10102-009-0006-2
6. Bolyen, E., Rideout, J.R., Dillon, M.R. *et al.* (2019). Reproducible, interactive, scalable and extensible microbiome data science using QIIME 2. *Nat Biotechnol* **37**, 852–857. doi:10.1038/s41587-019-0209-9
7. Bowling, L. (1994). Occurrence and possible causes of a severe cyanobacterial bloom in Lake Cargelligo, New South Wales. *Mar. Freshw. Res., 45*(5). doi:10.1071/MF9940737
8. Brown C. T., Hug L. A., Thomas B. C., Sharon I., Castelle C. J., Singh A., *et al.* (2015). Unusual biology across a group comprising more than 15% of domain Bacteria. *Nature*. 523:208. Doi: 10.1038/nature14486
9. Bruno A., Sandionigi A., Rizzi E., Bernasconi M., Vicario S., Galimberti A., *et al.* (2017). Exploring the under-investigated "microbial dark matter" of drinking water treatment plants. *Sci Rep*. 7:44350. Doi: 10.1038/srep44350
10. Byrne, S., Butler, C. A., Reynolds, E. C., & Dashper, S. G. (2018). Chapter 7 - Taxonomy of Oral Bacteria. *Methods in Microbiology, 45*. doi:10.1016/bs.mim.2018.07.001

11. Cai, P.; Cai, Q.; He, F.; Huang, Y.; Tian, C.;Wu, X.;Wang, C.; Xiao, B. (2021). Flexibility of *Microcystis* Overwintering Strategy in Response to Winter Temperatures. Microorganisms 2021, 9, 2278. doi:10.3390/microorganisms9112278
12. Caiola, M.G., and Pellegrini, S. (1984) Lysis of *Microcystis aeruginosa* (Kutz.) by Bdellovibrio-like Bacteria1. J Phycol 20: 471–475.
13. Campbell, A. M., Fleisher, J., Sinigalliano, C., White, J. R., & Lopez, J. V. (2015). Dynamics of marine bacterial community diversity of the coastal waters of the reefs, inlets, and wastewater outfalls of southeast Florida. *Microbiology Open*, *4*(3), 390–408. doi:10.1002/mbo3.245
14. Canfield, D., & Hoyer, M. (1988). The Eutrophication of Lake Okeechobee. *Lake and Reservoir Management*.
15. Canfield Jr. D. E., Bachmann, R. W. & Hoyer, M. V. (2021) Restoration of Lake Okeechobee, Florida: mission impossible?, Lake and Reservoir Management, 37:1, 95-111, doi: 10.1080/10402381.2020.1839607
16. Chang, T.-P. (1985). Selective inhabitation of parasitice Cyanophyte *Pseudanabaena* in water-bloom *Microcystis* colonies. *Arch. Hydrobiol.*
17. Chapman, R. L. (2013). Algae: the world's most important "plants"—an introduction. *Mitig. Adapt. Strateg. Glob. Change*, *18*, 5-12. doi:10.1007/s11027-010-9255-9.
18. Cuklina, J., Lee, C. H., Williams, E. G., Sajic, T., Collins, B. C., Rodriguez Martinez, M., . . . Pedrioli, P. G. (2021). Diagnostics and correction of batch effects in large-scale proteomic studies: a tutorial. *Molecular Systems Biology*(17). doi:10.15252/msb.202110240
19. Dick, G.J. (2021). The genetic and ecophysiological diversity of *Microcystis*. *Environ. Microbiol*. doi:10.1111/1462-2920.15615
20. Donnelly, C.P. 2018. *Microbial Ecology of South Florida Surface Waters: Examining the Potential for Anthropogenic Influences.* Master's thesis. Nova Southeastern University.
21. Dubnau, D., Smith, I., Morell, P., & Marmur, J. (1965). Gene conservation in Bacillus species. I. Conserved genetic and nucleic acid base sequence homologies. *Proc Natl Acad Sci U S A.*, *54*. doi:10.1073/pnas.54.2.491
22. Easson, C. G., & Lopez, J. V. (2019). Depth-Dependent Environmental Drivers of Microbial Plankton Community Structure in the Northern Gulf of Mexico. *Frontiers in microbiology*, *9*, 3175. doi:10.3389/fmicb.2018.03175
23. Eiler A, Bertilsson S. (2004). Composition of freshwater bacterial communities associated with cyanobacterial blooms in four Swedish lakes. *Environ Microbiol* **6**: 1228–1243.
24. Ezzedine, J. A., Desdevises, Y., & Jacquet, S. (2022). *Bdellovibrio* and like organisms: current understanding and knowledge gaps of the smallest cellular hunters of the microbial world. *Critical reviews in microbiology*, *48*(4), 428–449. doi:10.1080/1040841X.2021.1979464
25. Facey, J. A., Apte, S. C., & Mitrovic, S. M. (2019). A Review of the Effect of Trace Metals on Freshwater Cyanobacterial Growth and Toxin Production. *Toxins, 11*. doi:10.3390/toxins11110643
26. Freed, L.L. (2018). Characterization of the bioluminescent symbionts from ceratioids collected in the Gulf of Mexico. Masters thesis. Halmos College of Natural Sciences and Oceanography, Nova Southeastern University.
27. Gaysina, L. A., Saraf, A., and Singh, P. (2019) Chapter 1 - Cyanobacteria in Diverse Habitats. Academic Press. doi: 10.1016/B978-0-12-814667-5.00001-5.

28. Gorham, P., S. McNicholas & E. D. Allen. (1982). Problems encountered in searching for new strains of toxic planktonic cyanobacteria. South African Journal of Science. 78: 357.
29. Harke, M. J. *et al.* (2016). A review of the global ecology, genomics, and biogeography of the toxic cyanobacterium *Microcystis* spp. Harmful Algae 54, 4–20. https:// doi. org/ 10. 1016/j. hal. 2015. 12. 007.
30. Harrell Jr, F. (2023). _Hmisc: Harrell Miscellaneous_. R package version 5.0-1. https://CRAN.R-project.org/package=Hmisc.
31. Havens, KE. (2007). Cyanobacteria blooms: effects on aquatic ecosystems. In: Hudnell KH (ed). *Cyanobacterial Harmful Algal Blooms: State of the Science and Research*, vol. 619. Springer: New York, pp 675–732.
32. Herrmann, M., Wegner, C. E., Taubert, M., Geesink, P., Lehmann, K., Yan, L., Lehmann, R., Totsche, K. U., & Küsel, K. (2019). Predominance of *Cand*. Patescibacteria in Groundwater Is Caused by Their Preferential Mobilization From Soils and Flourishing Under Oligotrophic Conditions. *Frontiers in microbiology*, *10*, 1407. doi:10.3389/fmicb.2019.01407
33. Hugenholtz P, Pitulle C, Hershberger KL, Pace NR (1998) Novel division level bacterial diversity in a yellowstone hot spring. J Bacteriol 180:366–376
34. Huisman, J. M., Matthijs, H. C. P., & Visser, P. M. (2005). Harmful Cyanobacteria Springer Aquatic Ecology Series 3. *Dordrecht, The Netheralands*.
35. Huisman, J., Codd, G. A., Paerl, H. W., Ibelings, B. W., Verspagen, J. M., & Visser, P. M. (2018). Cyanobacterial blooms. *Nature Reviews Microbiology*, *16*(8), 471-483.
36. Ilhe, T. (2008). The Spatiotemporal Variation of *Microcystis* spp. (Cyanophyceae) and Microcystins in Quitzdorf reservoir (Sachsen). Die raum-zeitliche Variation von *Microcystis* spp. (Cyanophyceae) und Microcystinen in der Talsperre Quitzdorf (Sachsen). Ph.D. dissertation. Universität, Dresden, Germany.
37. Im, W.-T., Hu, Z.-Y., Kim, K.-H., Rhee, S.-K., Meng, H., Lee, S.-T., & Quan, Z.-X. (2012). Description of Fimbriimonas ginsengisoli gen. nov., sp. nov. within the Fimbriimonadia class nov., of the phylum Armatimonadetes. *Antonie van Leeuwenhoek*. doi:10.1007/s10482-012-9739-6
38. Imamura, N., Motoike, I., Shimada, N., Nishikori, M., Morisaki, H., & Fukami, H. (2001). An Efficient Screening Approach for Anti-*Microcystis* Compounds: Based on Knowledge of Aquatic Microbial Ecosystem. *The Journal of Antibiotics*.
39. J. Greg Caporaso, G. A.-L. (2018). EMP 16S Illumina Amplicon Protocol. *PLOS One*. doi:10.17504/protocols.io.nuudeww
40. Ji X, Verspagen JMH, Van de Waal DB, Rost B, Huisman J. (2020). Phenotypic plasticity of carbon fixation stimulates cyanobacterial blooms at elevated CO2. Sci Adv 6: eaax2926. doi:10.1126/sciadv.aax2926.
41. Jöhnk, K.D., Huisman, J., Sharples, J., Sommeijer, B., Visser, P.M. And Stroom, J.M. (2008), Summer heatwaves promote blooms of harmful cyanobacteria. Global Change Biology, 14: 495-512. doi:10.1111/j.1365-2486.2007.01510.x
42. Karns, R. C. 2017. *Microbial Community Richness Distinguishes Shark Species Microbiomes in South Florida.* Master's thesis. Nova Southeastern University.
43. Kolmonen, E., Sivonen, K., Rapala, J., & Haukka, K. (2004). Diversity of cyanobacteria and heterotrophic bacteria in cyanobacterial blooms in Lake Joutikas, Finland. *Aquatic Microbial Ecology*, *36*.

44. Komárek, J. (2003) Coccoid and colonial Cyanobacteria. Freshwater Algae of North America. Amsterdam: Elsevier,pp. 59–116.
45. Koval, S.F., Hynes, S.H., Flannagan, R.S., Pasternak, Z., Davidov, Y., and Jurkevitch, E. (2013) *Bdellovibrio exovorus* sp. nov., a novel predator of Caulobacter crescentus. *Int J Syst Evol Microbiol*. 63: 146–151.
46. Krausfeldt, L. E., Shmakova, E., Lee, H., Mazzei, V., Loftin, K. A., Smith, R. P., . . . Lopez, J. V. (submitted). Microbial biodiversity and phage-host interactions are linked to the occurrence of cyanobacterial blooms.
47. Kumar S., Herrmann M., Thamdrup B., Schwab V. F., Geesink P., Trumbore S. E., *et al.* (2017). Nitrogen loss from pristine carbonate-rock aquifers of the Hainich Critical Zone Exploratory (Germany) is primarily driven by chemolithoautotrophic anammox processes. *Front. Microbiol*. 8:1951. Doi: 10.3389/fmicb.2017.01951
48. Lahti, L. *et al.* microbiome R package. URL: http://microbiome.github.io
49. Lande, R. (1996). Statistics and Partitioning of Species Diversity, and Similarity among Multiple Communities. *Oikos*, *76*(1), 5–13. doi: 10.2307/3545743
50. Larkin, S. L., & Adams, C. M. (2007). Harmful Algal Blooms and Coastal Business: Economic Consequences in Florida. *Society and Natural Resources, 20*. doi:10.1080/08941920601171683
51. Larsson, J. (2022). _eulerr: Area-Proportional Euler and Venn Diagrams with Ellipses_. R package version 7.0.0. https://CRAN.R-project.org/package=eulerr.
52. Lecher, A. L. (2021). A Brief History of Lake Okeechobee: A Narrative of Conflict. *Journal of Floria Studies, 1*(9). Retrieved from https://www.journaloffloridastudies.org/files/vol0109/lecher-brief-history-lake-okeechobee.pdf
53. Léon-Zayas R., Peoples L., Biddle J. F., Podell S., Novotny M., Cameron J., *et al.* (2017). The metabolic potential of the single cell genomes obtained from the Challenger Deep, Mariana Trench within the candidate superphylum Parcubacteria (OD1). *Environ. Microbiol. 19.* 2769–2784. doi: 10.1111/1462-2920.13789.
54. Li, Z. K., Dai, G. Z., Zhang, Y., Xu, K., Bretherton, L., Finkel, Z. V., Irwin, A. J., Juneau, P., & Qiu, B. S. (2020). Photosynthetic adaptation to light availability shapes the ecological success of bloom-forming cyanobacterium *Pseudanabaena* to iron limitation. *Journal of phycology*, *56*(6), 1457–1467. doi:10.1111/jpy.13040
55. Litchman, E., de Tezanos Pinto, P., Klausmeier, C. A., Thomas, M. K., & Yoshiyama, K. (2010). Linking traits to species diversity and community structure in phytoplankton. *Hydrobiologia*, *653*, 15-28.
56. Luef B., Frischkorn K. R., Wrighton K. C., Holman H.-Y. N., Birarda G., Thomas B. C., *et al.* (2015). Diverse uncultivated ultra-small bacterial cells in groundwater. *Nat. Commun*. 6:6372. doi: 10.1038/ncomms7372
57. Ma, S. (2023). _MMUPHin: Meta-analysis Methods with Uniform Pipeline for Heterogeneity in Microbiome Studies_. R package version 1.12.1.
58. Malfertheiner, L.; Martínez-Pérez, C.; Zhao, Z.; Herndl, G.J.; Baltar, F. (2022). Phylogeny and Metabolic Potential of the Candidate Phylum SAR324. *Biology, 11*, 599. doi:10.3390/biology11040599
59. Mankiewicz-Boczek, J., & Font-Najera, A. (2022). Temporal and functional interrelationships between bacterioplankton communities and the development of a

toxigenic *Microcystis* bloom in a lowland European reservoir. *Nature Scientific Reports*. doi:10.1038/s41598-022-23671-2
60. Markou, G., Vandamme, D., & Muylaert, K.. (2014). Microalgal and cyanobacterial cultivation: The supply of nutrients. *Water Research, 65,* 186–202. doi: 10.1016/j.watres.2014.07.025
61. Maruyama T., Kato K., Yokoyama A., Tanaka T., Hiaishi A. & Park H.D. (2003) Dynamics of microcystin degrading bacteria in mucilage of *Microcystis*. Microbial Ecology, 46, 279–288.
62. Mataloni, G., Komarek, J., (2004). *Gloeocapsopsis aurea*, a new subaerophytic cyanobacterium from maritime Antarctica. Polar Biol. 27, 623–628.
63. McMurdie, P.J. and Holmes, S. (2013). An R package for reproducible interactive analysis and graphics of microbiome census data.PLoS ONE 8(4):e61217.
64. McQuaid, A. L. (2019). The Bioaccumulation of Cyanotoxins in Aquatic Food Webs. *Doctoral Dissertations, 2481*. Retrieved from https://scholars.unh.edu/dissertation/2481
65. Metcalf, J. S., Banack, S. A., Powell, J. T., Tymm, F. J., Murch, S. J., Brand, L. E., & Cox, P. A. (2018). Public health responses to toxic cyanobacterial blooms: perspectives from the 2016 Florida event. *Water Policy, 20*, 919-932. doi:10.2166/wp.2018.012
66. Missimer, T.M.; Thomas, S.; Rosen, B.H. (2021). Legacy Phosphorus in Lake Okeechobee (Florida, USA) Sediments: A Review and New Perspective. Water, 13, 39. doi:10.3390/w13010039
67. Mu, DS., Wang, S., Liang, QY. *et al.* (2020). Bradymonabacteria, a novel bacterial predator group with versatile survival strategies in saline environments. *Microbiome* **8**, 126. Doi: 10.1186/s40168-020-00902-0
68. Myer, M. H., Urquhart, E., Schaeffer, B. A., & Johnston, J. M. (2020). Spatio-Temporal Modeling for Forecasting High-Risk Freshwater Cyanobacterial Harmful Algal Blooms in Florida. *Frontiers in Environmental Science, 8*, 1-13. doi:10.3389/fenvs.2020.581091
69. O'Connell, L.M., Gao, S., McCorquodale, D.S., Fleisher, J., & Lopez, J.V. (2018). Fine grained compositional analysis of Port Everglades Inlet microbiome using high throughput DNA sequencing. *PeerJ, 6*.
70. Okello, W., Portmann, C., Erhard, M., Gademann, K. and Kurmayer, R. (2010), Occurrence of microcystin-producing cyanobacteria in Ugandan freshwater habitats. Environ. Toxicol., 25: 367-380. doi:10.1002/tox.20522
71. Oksanen *et al.* (2022). _vegan: Community Ecology Package_. R package version 2.6-4. https://CRAN.R-project.org/package=vegan
72. Paerl, H., & Scott, J. (2010). Throwing Fuel on the Fire: Synergistic Effects of Excessive Nitrogen Inputs and Global Warming on Harmful Algal Blooms. *Environ. Sci. Technol., 44*. doi:10.1021/es102665e
73. Paerl HW, Huisman J. (2008). Blooms like it hot. Science 320:57–58. doi:10.1126/science.1155398.
74. Paerl, Hans & Fulton, Rolland. (2006). Ecology of Harmful Cyanobacteria. doi:10.1007/978-3-540-32210-8_8.
75. Pankow, H. (1986). About endophytic and epiphytic algae in or on the mucilage envelope of *Microcystis* colonies. Arch. Protistenkd. 132, 377–380.
76. Parks, D., Chuvochina, M., Waite, D., Rinke, C., Skarshewski, A., Chaumeil, P.-A., & Philip, H. (2018). A standardized bacterial taxonomy based on genome phylogeny substantially revises the tree of life. *Nature Biotechnology, 36*. doi:10.1038/nbt.4229

77. *PCR purification with Beckman Coulter AMPure XP magnetic beads and the VIAFLO 96*. (2020). Retrieved from INTEGRA: https://www.integra-biosciences.com/global/en/applications/pcr-purification-beckman-coulter-ampure-xp-magnetic-beads-and-viaflo-96#top
78. Pokrzywinski, K.L.; Bishop, W.M.; Grasso, C.R.; Fernando, B.M.; Sperry, B.P.; Berthold, D.E.; Laughinghouse, H.D., IV; Van Goethem, E.M.; Volk, K.; Heilman, M.; *et al.* (2022). Evaluation of a Peroxide-Based Algaecide for Cyanobacteria Control: A Mesocosm Trial in Lake Okeechobee, FL, USA. *Water*, 14, 169. doi:10.3390/w14020169
79. Pommier, T., Pinhassi, J., & Hagstrom, A. (2005). Biogeographic analysis of ribosomal RNA clusters from marine bacterioplankton. *Aquatic Microbial Ecology*, *41*(1), 79–89. doi:10.3354/ame041079
80. Prinos, S. T. (2016). *Saltwater intrusion monitoring in Florida*.
81. Proctor C. R., Besmer M. D., Langenegger T., Beck K., Walser J.-C., Ackermann M., *et al.* (2018). Phylogenetic clustering of small low nucleic acid-content bacteria across diverse freshwater ecosystems. *ISME J*. 12 1344–1359. doi: 10.1038/s41396-018-0070-78.
82. Quast C, Pruesse E, Yilmaz P, Gerken J, Schweer T, Yarza P, Peplies J, Glöckner FO (2013) The SILVA ribosomal RNA gene database project: improved data processing and web-based tools. Nucl. Acids Res. 41 (D1): D590-D596.
83. R Core Team. (2022). R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.
84. Reynolds, C.S. (2006). *Ecology of Phytoplankton*. Cambridge Univ. Press, Cambridge.
85. Reynolds, C. S. (1973). Growth and buoyancy of *Microcystis aeruginosa* Kütz. emend. Elenkin in a shallow eutrophic lake. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, *184*(1074), 29-50.
86. Rollwagen-Bollens, G., Lee, T., Rose, V., & Bollens, S. M. (2018). Beyond Eutrophication: Vancouver Lake, WA, USA as a Model System for Assessing Multiple, Interacting Biotic and Abiotic Drivers of Harmful Cyanobacterial Blooms. *Water, 10*. doi:10.3390/w10060757
87. Rosen, B. H., Davis, T. W., Gobler, C. J., Kramer, B. J., & Loftin, K. A. (2017). *Cyanobacteria of the 2016 Lake Okeechobee and Okeechobee Waterway Harmful Algal Bloom: U.S. Geological Survey Open-File Report 2017–1054.* doi:10.3133/ofr20171054
88. Sanger, F., Nicklen, S., & Coulson, A. R. (1977). DNA sequencing with chain-temrinating inhibitors. *Proceedings of the National Academy of Sciences of the United States of America, 74*.
89. Schelske, C. L. (1989). Assessment of Nutrient Effects and Nutrient Limitation in Lake Okeechobee. *Water Resources Bulletin, 25*.
90. Shannon P, Markiel A, Ozier O, Baliga NS, Wang JT, Ramage D, Amin N, Schwikowski B, Ideker T. (2003). Cytoscape: a software environment for integrated models of biomolecular interaction networks Genome Research. 13(*11*):2498-504
91. Shen, H., Niu, Y., Xie, P., Tao, M., & Yang, X. (2011). Morphological and physiological changes in *Microcystis aeruginosa* as a result of interactions with heterotrophic bacteria. *Freshwater Biology*, *56*, 1065-1080. doi:10.1111/j.1365-2427.2010.02551.x
92. Shi L., Cai Y., Yang H., Xing P., Li P., Kong L. *et al.* (2009) Phylogenetic diversity and specificity of bacteria associated with *Microcystis aeruginosa* and other cyanobacteria. Journal of Environmental Sciences (China), 21, 1581–1590.

93. Sigee D. (2005). *Freshwater Microbiology. Biodiversity and Dynamic Interactions of Microorganisms in the Aquatic Environment*. John Wiley & Sons: Chichester, UK, pp 328–338.
94. Smayda, T. J. (1997). What is a bloom? A commentary. *Limnol. Oceanogr., 42*(5), 1132-1136.
95. South Florida Water Management District. (n.d.). Retrieved from DBHYDRO: https://my.sfwmd.gov/dbhydroplsql/show_dbkey_info.main_menu
96. South Florida Water Management District (SFWMD). (n.d.). Lake Okeechobee: In Review. Retrieved from https://www.sfwmd.gov/
97. South Florida Water Management District. (n.d.). *Impacts of Operating Lake Okeechobee at Lower Water Levels* [Infographic]. SFWMD. https://www.sfwmd.gov/sites/default/files/documents/infographic_lake_okee_depth.pdf
98. Stolp, H., and Starr, M.P. (1963) Bdellovibrio bacteriovorus gen. et sp. n., a predatory, ectoparasitic, and bacteriolytic microorganism. *Antonie Van Leeuwenhoek*. 29: 217–248.
99. Stomp, M. *et al.* (2007). Colourful coexistence of red and green picocyanobacteria in lakes and seas. Ecol. Lett. 10, 290–298.
100. Thurkal, A. K. (2017). A REVIEW ON MEASUREMENT OF ALPHA DIVERSITY IN BIOLOGY. *Agric Res J*. doi:10.5958/2395-146X.2017.00001.1
101. Tian, R., Ning, D., He, Z. *et al.* (2020). Small and mighty: adaptation of superphylum *Patescibacteria* to groundwater environment drives their genome simplicity. *Microbiome* 8, 51. doi:10.1186/s40168-020-00825-w
102. Tu, J., Chen, L., Gao, S., Zhang, J., Bi, C., Tao, Y., . . . Lu, Z. (2019). Obtaining Genome Sequences of Mutualistic Bacteria in Single *Microcystis* Colonies. *Int. J. Mol. Sci., 20*. doi:10.3390/ijms20205047
103. U.S. Army Corps of Engineers, J. D. (2021). *Home*. Herbert Hoover Dike. https://www.saj.usace.army.mil/HHD/
104. Van Wichelen, J., Vanormelingen, P., Codd, G. A., & Vyverman, W. (2016). The common bloom-forming cyanobacterium *Microcystis* is prone to wide array of microbial antagonists. *Harmful Algae, 55*, 97-111. doi:10.1016/j.hal.2016.02.009
105. Visser, P., Verspagen, J., Sandrini, G., Stal, L., Matthijs, H., Davis, T., . . . Huisman, J. (2016). How rising CO2 and global warming may stimulate harmful cyanobacterial blooms. *Harmful Algae, 54*.
106. Wang, K., Mou, X., Cao, H., Struewing, I., Allen, J., & Lu, J. (2021). Co-occurring microorganisms regulate the succession of cyanobacterial harmful algal blooms. *Environmental Pollution, 288*, 117682. doi:10.1016/j.envpol.2021.117682
107. Whitton, B.A., Potts, M., (2000a). The Ecology of Cyanobacteria. Kluwer Academic Publishers, Dordrecht.
108. Whitton, B.A., Potts, M., (2000b). Introduction of cyanobacteria. In: Whitton, B.A., Potts, M. (Eds.), The Ecology of Cyanobacteria. Their Diversity in Time and Space. Kluwer Academic, Dordrecht, pp. 1–10.
109. Wickham, H. (2016). ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York.
110. Wiegand, C., & Pflugmacher, S. (2005). Ecotoxicological eff ects of selected cyanobacterial secondary metabolites a short review. *Toxicology and Applied Pharmacology, 203*.

111. Wilhelm, S. W., Bullerjahn, G. S., & McKay, R. M. L. (2020). The Complicated and Confusing Ecology of *Microcystis* Blooms. *MBio*, *11*(3), e00529-20. doi:10.1128/mBio.00529-20
112. Williams, C. D., Aubel, M. T., Chapman, A. D., & D'Aiuto, P. E. (2007). Identification of cyanobacterial toxins in Florida's freshwater systems. *Lake and Reservoir Management, 23*(2), 144-152. doi:10.1080/07438140709353917
113. Woese, C. R., & Fox, G. E. (1977). Phylogenetic structure of the prokaryotic domain: The primary kingdoms. *Proc Natl Acad Sci USA, 74*, 5088-5090. doi:10.1073/pnas.74.11.5088
114. Xie, L. Q., Xie, P., & Tang, H. J. (2003). Enhancement of dissolved phosphorus release from sediment to lake water by *Microcystis* blooms—an enclosure experiment in a hyper-eutrophic, subtropical Chinese lake. *Environmental Pollution*, *122*(3), 391–399. doi:10.1016/S0269-7491(02)00305-6
115. You, J., Mallery, K., Hong, J., & Hondzo, M. (2017). Temperature effects on growth and buoyancy of *Microcystis aeruginosa*. *Journal of Plankton Research*, *40*(1), 16–28. doi:10.1093/plankt/fbx059
116. Zamora-Barrios, C. A., Nandini, S., & Sarma, S. S. (2019). Bioaccumulation of microcystins in seston, zooplankton and fish: A case study in Lake Zumpango, Mexico. *Environmental Pollution, 249*. doi:10.1016/j.envpol.2019.03.029
117. Zhang, H.; Xie, Y.; Zhang, R.; Zhang, Z.; Hu, X.; Cheng, Y.; Geng, R.; Ma, Z.; Li, R. (2023). Discovery of a High-Efficient Algicidal Bacterium against *Microcystis aeruginosa* Based on Examinations toward Culture Strains and Natural Bloom Samples. Toxins, 15, 220. doi:10.3390/toxins15030220
118. Zheng, Q., Wang, Y., Xie, R., Lang, A., Liu, Y., Lu, J., . . . Nianzhi, J. (2018). Dynamics of Heterotrophic Bacterial Assemblages within Synechococcus Cultures. *Applied and Environmental Microbiology*, *84*(3). doi:10.1128/AEM.01517-17
119. Zhu, Q., Shi, L., Peng, G., & Fei-shi, L. (2014). High-throughput Sequencing Technology and Its Application. Journal of Northeast Agricultural University (English Edition), 21. doi:10.1016/S1006-8104(14)60073-8
120. Zuo, Jun & Hu, Lili & Shen, Wei & Zeng, Jiaying & Li, Lin & Gan, Nanqin. (2021). The involvement of α-proteobacteria *Phenylobacterium* in maintaining the dominance of toxic *Microcystis* blooms in Lake Taihu, China. Environmental Microbiology. 23. 1066–1078. 10.1111/1462-2920.15301.

IV. R Script

```
###### BATCH CORRECTION & ASSOCIATED ANALYSES ######

##First had to go through and manually assign batches to the samples within the
##metadata file (based on mapping files)
## 10 KNOWN SEQUENCING RUNS IN TOTAL (an unknown sequence run making 11 "UNK")

###### SET WORKING DIRECTORY AND SEED ####
setwd("F:/Paise_Thesis/LakeO_Data/2019-2021_LakeO_Data/Analyses/LakeO_BatchCorrected/Analyses_Corrected")
#or setwd("/Volumes/PaiseSSD-T7/Paise_Thesis/LakeO_Data/2019-
2021_LakeO_Data/Analyses/LakeO_BatchCorrected/Analyses_Corrected") for use on the lab computer
set.seed(1998)\

###### Packages ######
library(vegan)
library(ggplot2)
library(tidyverse)
library(reshape2)
library(BiocManager)
library(MMUPHin)

#updating BiocManager and installing mmuphin
# if (!require("BiocManager", quietly = TRUE))
#   install.packages("BiocManager")
# BiocManager::install(version = "3.16")
# BiocManager::install("MMUPHin")

###### Creating relative abundance data ######
set.seed(1998)
dat<-read.csv("feature_Y123_nobcmASVs-nobelow10korDupes.csv", header=TRUE, row.names = 1)
dat<-data.matrix(dat)
typeof(dat) #"integer"
dat <- t(dat)
row.names(dat) # row names should now be the sample names
metadata <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
typeof(metadata) ## "list"
dat <- as.data.frame(dat)
typeof(dat)
common.rownames <- intersect(rownames(dat), rownames(metadata))
dat <- dat[common.rownames,]
metadata <- metadata[common.rownames,]
all.equal(rownames(dat),rownames(metadata))
otu.abund<-which(colSums(dat)>2)
dat.dom<-dat[,otu.abund] #dominant taxa
dat.pa<-decostand(dat.dom, method ="pa") #presence/absence data
dat.otus.01per<-which(colSums(dat.pa) > (0.01*nrow(dat.pa)))
dat.01per<-dat.dom[,dat.otus.01per] #removed ASVs that occur less than 0.1%; 8,340 taxa present
dat.otus.001per<-which(colSums(dat.pa) > (0.001*nrow(dat.pa)))
dat.001per<-dat.dom[,dat.otus.001per] #removed ASVs that occur less than 0.01%; 44,623 taxa present
                                      #increases the number of ASVs - includes more "microdiversity"
dat.ra<-decostand(dat.01per, method = "total") #relative abundance of >1% taxa

###### ANOSIM by Sequencing Batch ######
set.seed(1998)
##create relative abundance table in above code
##create Bray-Curtis dissimilarity distance matrix
ra.bc.dist<-vegdist(dat.ra, method = "bray")

##betadisper calculates dispersion (variances) within each group
dis.Batch <- betadisper(ra.bc.dist,metadata$Batch)

##permutest determines if the variances differ by groups (If differences are SIGNIFICANT - use ANOSIM
##                                                        if not use PERMANOVA (adonis))
permutest(dis.Batch, pairwise=TRUE, permutations=999)
#           Df Sum Sq  Mean Sq      F N.Perm Pr(>F)
# Groups     10 1.0196 0.101957 10.832    999  0.001 ***   SIGNIFICANT - USE ANOSIM!!
# Residuals 530 4.9886 0.009413
# ---

# Pairwise comparisons:
```

```
#   (Observed p-value below diagonal, permuted p-value above diagonal)
# ELIZA2    ELIZA23     ELIZA3       LO22      LO310     LO8382       NOAA      PAIS1      PAIS2      PAIS3
UNK
# ELIZA2             2.5800e-01 2.2800e-01 4.0000e-03 1.0000e-03 2.8000e-02 7.0300e-01 5.4400e-01 2.0000e-02
2.0000e-03 0.001
# ELIZA23 2.4836e-01            2.9000e-02 4.5000e-02 1.0000e-03 9.7000e-02 3.1600e-01 4.7000e-01 1.6100e-01
4.9000e-02 0.001
# ELIZA3  1.9314e-01 2.3483e-02            3.0000e-03 1.0000e-03 6.0000e-03 4.5400e-01 1.1300e-01 3.0000e-03
1.0000e-03 0.001
# LO22    4.2982e-03 4.4251e-02 1.9370e-03            1.1800e-01 9.9300e-01 7.5000e-02 1.0000e-02 5.1900e-01
7.1500e-01 0.005
# LO310   1.4327e-04 5.5957e-04 8.7490e-06 1.2846e-01            1.8300e-01 3.0000e-03 1.0000e-03 3.1000e-02
4.1000e-02 0.966
# LO8382  2.2967e-02 9.5642e-02 4.8732e-03 9.9098e-01 1.8031e-01            1.1600e-01 3.4000e-02 6.0500e-01
7.5900e-01 0.029
# NOAA    7.1430e-01 3.1669e-01 4.7882e-01 7.6250e-02 3.0428e-03 1.0361e-01            5.3600e-01 1.1600e-01
7.0000e-02 0.001
# PAIS1   5.5294e-01 4.9884e-01 9.5982e-02 8.2362e-03 2.4554e-04 3.9403e-02 5.4000e-01            5.3000e-02
4.0000e-03 0.001
# PAIS2   1.9845e-02 1.6321e-01 2.9914e-03 5.0465e-01 2.4943e-02 5.7184e-01 1.0857e-01 4.4072e-02
7.0500e-01 0.002
# PAIS3   1.6726e-03 4.7249e-02 1.0649e-03 6.9056e-01 3.8967e-02 7.4158e-01 7.1715e-02 3.8772e-03 7.0437e-01
0.001
# UNK     6.1433e-14 3.1874e-10 7.8632e-11 4.5319e-03 9.6747e-01 2.3258e-02 3.6162e-05 3.0384e-14 5.3552e-05
3.8664e-05

##ANOSIM - determining if the differences between two or more groups are significant.
## The ANOSIM statistic "R" compares the mean of ranked dissimilarities between groups to
## the mean of ranked dissimilarities within groups. An R value close to "1" suggests
## dissimilarity between groups while an R value close to "0" suggests an even distribution of
## high and low ranks within and between groups"
## the higher the R value, the more dissimilar your groups are in terms of microbial community composition.

anosim(ra.bc.dist, metadata$Batch, permutations = 999)
# ANOSIM statistic R: 0.1486
# Significance: 0.001

anosim(ra.bc.dist, metadata$Batch, permutations = 9999)
# ANOSIM statistic R: 0.1486
# Significance: 0.0001

## Conclusion? There are significantly weak differences between batches so the
## data needs to be batch corrected and ALL analyses redone.

###### BATCH CORRECTION ######
set.seed(1998)
library(MMUPHin)
library(vegan)

## Loading in feature- and metadata
dat <- read.csv("feature_Y123_nobcmASVs-nobelow10korDupes.csv", header=TRUE, row.names = 1)
dat <- data.matrix(dat)
typeof(dat) #"integer"
dat <- t(dat) #transposing data matrix
row.names(dat) # row names should now be the sample names
metadata <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
typeof(metadata) ## "list"
dat <- as.data.frame(dat)
typeof(dat)
common.rownames <- intersect(rownames(dat), rownames(metadata))
dat <- dat[common.rownames,]
metadata <- metadata[common.rownames,]
all.equal(rownames(dat),rownames(metadata)) #TRUE

## Batch Correction (following Harvard tutorial)
#looking at how many samples are in each batch
table(metadata$Batch)
# ELIZA2 ELIZA23  ELIZA3   LO22   LO310   LO8382    NOAA   PAIS1   PAIS2   PAIS3   UNK
#   62      50      11      38      20      20       6      98      40      72     124

#Adjusting (removing) batch effect
#taxa should be rows in feature table and samples should be rows in metadata
#feature table should be a matrix while metadata should be a dataframe
fit_adjust_batch <- adjust_batch(feature_abd = t(dat),
                                 batch = "Batch",
                                 data = metadata)
```

```
Lake_abd_adj <- fit_adjust_batch$feature_abd_adj #now adjusted feature table MATRIX
Lake_abd_adj <- as.data.frame(Lake_abd_adj) #converting to data frame
write.csv(Lake_abd_adj, "feature_Y123_ADJUSTED.csv") #saving as csv


###### Creating a rarefaction curve on the read counts ######
library(vegan)

#load in data with NO blank samples or blank ASVs
rardat<-read.csv("feature_Y123_noblanksorbASVs.csv", header=TRUE, row.names=1, sep=',')

#as you can see the samples are in columns and need to be in the rows so we need to flip or transpose the file
#transpose the data to rows
trans.rardat <- t(rardat)
## check file to make sure it worked
trans.rardat[1:5,1:5] #shows rows 1 through 5 and the samples should now be the rows
##making the transformed data matrix into main
rardat <- trans.rardat
##changing back into data frame instead of matrix (transforming the data frame turned it into a matrix)
rardat <-as.data.frame(rardat)
#check data file to make sure it looks okay
View(rardat)

rowSums(rardat) #sums the value of each row in the data frame

#### Creating the rarefaction curve
#count the number of species within each sample
S <- specnumber(rardat)
raremax <- min(rowSums(rardat)) ## takes the sample with the lowest sample size which is 0 in this dataset

#creating color palette for curve
colors() ## lists the color names that are built into R
cc <- palette()
palette(c(cc,"purple","brown"))    ## creating the color ramp for the plot
cc <- palette()

#plotting the rarefaction curves
## auto removes samples that have no reads
pars <- expand.grid()
Hklim <- rarecurve(rardat, step = 2000, sample=raremax, col = cc, label = TRUE, main="Rarefaction Curve for Lake
O read counts",
          cex= 0.14, cex.axis= 0.7, cex.lab= 1, xlim=c(0,100000), xlab = "# of Reads", ylab = "# of ASVs", tidy
= T)

#### ####



###### ANALYSES ON BATCH CORRECTED DATA ######
###### SET WORKING DIRECTORY AND SEED ####
setwd("F:/Paise_Thesis/LakeO_Data/2019-2021_LakeO_Data/Analyses/LakeO_BatchCorrected/Analyses_Corrected")
#or setwd("/Volumes/PaiseSSD-T7/Paise_Thesis/LakeO_Data/2019-
2021_LakeO_Data/Analyses/LakeO_BatchCorrected/Analyses_Corrected")
#for use on the lab computer
set.seed(1998)

###### PACKAGES ######
library(phyloseq)
library(vegan)
library(ggplot2)
library(tidyverse)
library(RVAideMemoire)
library(DESeq2)
library(corrplot)
library(multcompView)
library(pgirmess)
library(data.table)
library(microbiome)
library(BiocManager)
library(ggthemes)
library(gplots)
library(RColorBrewer)
library(co-occur)
library(visNetwork)
library(Hmisc)
library(cowplot)
```

```
library(reshape2)
library(sjmisc)
library(MASS)
library(scales)
library(forcats)
library(leaflet)
library(eulerr)
library(microbiomeutilities)

##Installing packages
BiocManager::install("DESeq2")
BiocManager::install("lefser")
BiocManager::install("ALDEx2")
BiocManager::install("ANCOMBC")
BiocManager::install("phyloseq")
BiocManager::install("microbiome")
BiocManager::install("microbiomeutilities")

##Had to install using binaries (3/9/23 on iMAC)
install.packages("tibble", type="binary")
install.packages("Hmisc", type="binary")

## Notes on packages:
# pgirmess = Kruskal-Wallis Test
# RVAideMemoire = PERMANOVA
# cowplot = making multiple plots using ggplots objects

###### Prepping data for analyses #####

### import feature-table data ###
##change to csv or import as a tsv using read.table function
dat<-read.csv("feature_Y123_ADJUSTED.csv", header=TRUE, row.names = 1) ## do not add "header =" or "row.names ="
for merging
# 561 samples; 65294 taxa

dat<-data.matrix(dat) ##if data is not recognized as a data.frame numeric
typeof(dat) #"integer"
#check data file to make sure it looks okay


#as you can see the samples are in columns and need to be in the rows so we need to flip or transpose the file
#transpose the data to rows
trans.dat <- t(dat)

## check file to make sure it worked
trans.dat[1:5,1:5] #shows rows 1 through 5 and the samples should now be the rows


##set transposed data to main data variable
dat <-trans.dat
row.names(dat) # row names should now be the sample names

### import metadata ###
###(if you intend to do any statistical analyses in R)
##If not skip to refining and normalizing steps
metadata <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)

##should read "list"
typeof(metadata) ## "list"
dat <- as.data.frame(dat) ## had to change dat back into a data frame to check for matching rows
typeof(dat) ## "list"

##check to make sure the sample names match and are correct
common.rownames <- intersect(rownames(dat), rownames(metadata))
##541 rows are in common (20 S80 samples NOT included)

##if there are any rows that do not match, they will not be included in the statistical analysis or relative
abundance tables
dat <- dat[common.rownames,]
metadata <- metadata[common.rownames,]

##check that all rows match
all.equal(rownames(dat),rownames(metadata)) #TRUE so yes they all match
dat[1:5,1:3] ## double-checking that everything looks good
```

```
##merging the working feature and taxonomy tables
feat <- dat
tax <- read.csv("taxonomy_Y123_edited&cleaned.csv")
feattax <- merge.data.frame(feat, tax, by= "FeatureID", all.x=TRUE, all.y = TRUE)
write.csv(feattax, "feat-tax_Y123_cleaned.csv")


## CONTINUE HERE IF YOU ARE IGNORING METADATA ###
## refining and normalizing data #
##remove singletons and doubletons -  ASVs that only show up once or twice
##this can be modified or removed if desired. Depends on what you want to know
library(vegan)

otu.abund<-which(colSums(dat)>2)
dat.dom<-dat[,otu.abund] #46838 taxa

##all this will get rid of ASVs that appear less than a certain percent in the data
##this is not always something that you should do depending on your question.
dat.pa<-decostand(dat.dom, method ="pa")  #"pa" = standardization method that scales your data to
presence/absence (0/1)
##remove ASVs that occur <0.01 ***
dat.otus.01per<-which(colSums(dat.pa) > (0.01*nrow(dat.pa)))
dat.01per<-dat.dom[,dat.otus.01per]
# 8,340 taxa
write.csv(as.data.frame(t(dat.01per)), "feature_Y123_0.01per.csv")

##remove ASVs that occur <0.001 ---> increases the number of ASVs - includes more "micro-diversity"
dat.otus.001per<-which(colSums(dat.pa) > (0.001*nrow(dat.pa)))
dat.001per<-dat.dom[,dat.otus.001per]
# 46,838 taxa

## relative abundance --> normalization ##
dat.ra<-decostand(dat.01per, method = "total") #"total" = standardization method that divides your data by
margin total (def. margin = 1)

##export relative abundance table(s)
write.csv(dat.ra, "relative-abundance_Y123.csv")

## SHORTCUT WITH NO EXPLANATIONS
## re-creating relative abundance table
set.seed(1998)
dat<-read.csv("feature_Y123_ADJUSTED.csv", header=TRUE, row.names = 1)
dat<-data.matrix(dat)
typeof(dat)
dat <- t(dat)
row.names(dat)
metadata <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
typeof(metadata)
dat <- as.data.frame(dat)
typeof(dat)
common.rownames <- intersect(rownames(dat), rownames(metadata))
dat <- dat[common.rownames,]
metadata <- metadata[common.rownames,]
all.equal(rownames(dat),rownames(metadata))
otu.abund<-which(colSums(dat)>2)
dat.dom<-dat[,otu.abund]
dat.pa<-decostand(dat.dom, method ="pa")
dat.otus.01per<-which(colSums(dat.pa) > (0.01*nrow(dat.pa)))
dat.01per<-dat.dom[,dat.otus.01per]
dat.otus.001per<-which(colSums(dat.pa) > (0.001*nrow(dat.pa)))
dat.001per<-dat.dom[,dat.otus.001per]
dat.ra<-decostand(dat.01per, method = "total")


###### Merging relative abundance with taxonomy and getting averages ######
Yr1 <- read.csv("Year1_RA.csv")
Yr2 <- read.csv("Year2_RA.csv")
Yr3 <- read.csv("Year3_RA.csv")
tax <- read.csv("taxonomy_Y123_edited&cleaned.csv")
Yr1t <- merge.data.frame(Yr1,tax,by= "FeatureID", all.x = TRUE)
Yr2t <- merge.data.frame(Yr2,tax,by= "FeatureID", all.x = TRUE)
Yr3t <- merge.data.frame(Yr3,tax,by= "FeatureID", all.x = TRUE)
write.csv(Yr1t, "Year1_RA.csv")
write.csv(Yr2t, "Year2_RA.csv")
write.csv(Yr3t, "Year3_RA.csv")
```

```
### Average and St.dev abundance of each phylum in each year
library(tidyverse)

## Year 1
#first merge data with matching taxonomy and load csv
Yr1 <- read.csv("Year1_RA.csv", row.names = 1)
#Sum by phylum across samples
physumY1 <- Yr1 %>%
  group_by(Phylum) %>%
  summarise(across(where(is.numeric), sum))
#Average phylum across samples
Y1mean <- apply(physumY1[,-1], 1, mean, na.rm=TRUE)
#Standard deviation across samples
Y1std <- apply(physumY1[,-1], 1, sd, na.rm=TRUE)
#merge average and st.dev with rows
Y1avsd <- as.data.frame(cbind(physumY1$Phylum,Y1mean, Y1std))
#Renaming columns and saving as csv
colnames(Y1avsd)[1] ="Phylum"
colnames(Y1avsd)[2] ="Average"
colnames(Y1avsd)[3] ="Stand.Dev"
write.csv(Y1avsd, "Year1_AvSD-UPDATED.csv")
#Extract top 10 phyla and save as csv
top101 <- names(top10phy.names.Y1)
Y1avsd10 <- filter(Y1avsd,
                   Y1avsd$Phylum %in% top101)
write.csv(Y1avsd10, "Year1_AvSD_TOP10-UPDATED.csv")

## Year 2
Yr2 <- read.csv("Year2_RA.csv", row.names = 1)
#Sum by phylum across samples
physumY2 <- Yr2 %>%
  group_by(Phylum) %>%
  summarise(across(where(is.numeric), sum))
#Average phylum across samples
Y2mean <- apply(physumY2[,-1], 1, mean, na.rm=TRUE)
#Standard deviation across samples
Y2std <- apply(physumY2[,-1], 1, sd, na.rm=TRUE)
#merge average and st.dev with rows
Y2avsd <- as.data.frame(cbind(physumY2$Phylum,Y2mean, Y2std))
#Renaming columns and saving as csv
colnames(Y2avsd)[1] ="Phylum"
colnames(Y2avsd)[2] ="Average"
colnames(Y2avsd)[3] ="Stand.Dev"
write.csv(Y2avsd, "Year2_AvSD-UPDATED.csv")
#Extract top 10 phyla and save as csv
top102 <- names(top10phy.names.Y2)
Y2avsd10 <- filter(Y2avsd,
                   Y2avsd$Phylum %in% top102)
write.csv(Y2avsd10, "Year2_AvSD_TOP10-UPDATED.csv")

## Year 3
Yr3 <- read.csv("Year3_RA.csv", row.names = 1)
#Sum by phylum across samples
physumY3 <- Yr3 %>%
  group_by(Phylum) %>%
  summarise(across(where(is.numeric), sum))
#Average phylum across samples
Y3mean <- apply(physumY3[,-1], 1, mean, na.rm=TRUE)
#Standard deviation across samples
Y3std <- apply(physumY3[,-1], 1, sd, na.rm=TRUE)
#merge average and st.dev with rows
Y3avsd <- as.data.frame(cbind(physumY3$Phylum,Y3mean, Y3std))
#Renaming columns and saving as csv
colnames(Y3avsd)[1] ="Phylum"
colnames(Y3avsd)[2] ="Average"
colnames(Y3avsd)[3] ="Stand.Dev"
write.csv(Y3avsd, "Year3_AvSD-UPDATED.csv")
#Extract top 10 phyla and save as csv
top103 <- names(top10phy.names.Y3)
Y3avsd10 <- filter(Y3avsd,
                   Y3avsd$Phylum %in% top103)
write.csv(Y3avsd10, "Year3_AvSD_TOP10-UPDATED.csv")

# Merge all years together and save as csv
#Original lists
#put all data frames into list
```

```
Y123avstd <- list(Y1avsd, Y2avsd, Y3avsd)
#merge all data frames in list
all <- Y123avstd %>% reduce(full_join, by='Phylum')
#renaming columns
colnames(all)[2] ="Y1mean"
colnames(all)[3] ="Y1std"
colnames(all)[4] ="Y2mean"
colnames(all)[5] ="Y2std"
colnames(all)[6] ="Y3mean"
colnames(all)[7] ="Y3std"

#Top 10 lists
Y123avstd10 <- list(Y1avsd10, Y2avsd10, Y3avsd10)
top10 <- Y123avstd10 %>% reduce(full_join, by='Phylum')
colnames(top10)[2] ="Y1mean"
colnames(top10)[3] ="Y1std"
colnames(top10)[4] ="Y2mean"
colnames(top10)[5] ="Y2std"
colnames(top10)[6] ="Y3mean"
colnames(top10)[7] ="Y3std"

#Save as csvs
write.csv(all, "Year123_AvSD.csv")
write.csv(top10, "Year123_AvSD_TOP10.csv")

###### Separating feature table by Station (CSVs) ######
CLV <- as.data.frame(t(dat.ra[grep("^CLV10A", rownames(dat.ra)),]))
KISS <- as.data.frame(t(dat.ra[grep("^KISSR0.0", rownames(dat.ra)),]))
L1 <- as.data.frame(t(dat.ra[grep("^L001", rownames(dat.ra)),]))
L4 <- as.data.frame(t(dat.ra[grep("^L004", rownames(dat.ra)),]))
L5 <- as.data.frame(t(dat.ra[grep("^L005", rownames(dat.ra)),]))
L6 <- as.data.frame(t(dat.ra[grep("^L006", rownames(dat.ra)),]))
L7 <- as.data.frame(t(dat.ra[grep("^L007", rownames(dat.ra)),]))
L8 <- as.data.frame(t(dat.ra[grep("^L008", rownames(dat.ra)),]))
LZ2 <- as.data.frame(t(dat.ra[grep("^LZ2_", rownames(dat.ra)),]))
Z25A <- as.data.frame(t(dat.ra[grep("^LZ25A", rownames(dat.ra)),]))
Z30 <- as.data.frame(t(dat.ra[grep("^LZ30", rownames(dat.ra)),]))
Z40 <- as.data.frame(t(dat.ra[grep("^LZ40", rownames(dat.ra)),]))
PALM <- as.data.frame(t(dat.ra[grep("^PALMOUT", rownames(dat.ra)),]))
PEL <- as.data.frame(t(dat.ra[grep("^PELBAY3", rownames(dat.ra)),]))
POLE3S <- as.data.frame(t(dat.ra[grep("^POLE3S", rownames(dat.ra)),]))
PO <- as.data.frame(t(dat.ra[grep("^POLESOUT", rownames(dat.ra)),]))
RIT <- as.data.frame(t(dat.ra[grep("^RITTAE2", rownames(dat.ra)),]))
S308 <- as.data.frame(t(dat.ra[grep("^S308", rownames(dat.ra)),]))
S77 <- as.data.frame(t(dat.ra[grep("^S77", rownames(dat.ra)),]))
S79 <- as.data.frame(t(dat.ra[grep("^S79", rownames(dat.ra)),]))

#S80 not included in adjusted dataset

###### Separating feature table by Year then Station (CSVs) ######
dat1 <- as.data.frame(t(dat.ra[grep("_19$", rownames(dat.ra)),]))
dat2 <- as.data.frame(t(dat.ra[grep("_20$", rownames(dat.ra)),]))
dat3 <- as.data.frame(t(dat.ra[grep("_21$", rownames(dat.ra)),]))
write.csv(dat1,"feature_Y1r_ADJUSTED.csv")
write.csv(dat2,"feature_Y2r_ADJUSTED.csv")
write.csv(dat3,"feature_Y3r_ADJUSTED.csv")
dat1 <- as.data.frame(t(dat1))
dat2 <- as.data.frame(t(dat2))
dat3 <- as.data.frame(t(dat3))

#Year 1 Stations
CLV <- as.data.frame(t(dat1[grep("^CLV10A", rownames(dat1)),]))
KISS <- as.data.frame(t(dat1[grep("^KISSR0.0", rownames(dat1)),]))
L1 <- as.data.frame(t(dat1[grep("^L001", rownames(dat1)),]))
L4 <- as.data.frame(t(dat1[grep("^L004", rownames(dat1)),]))
L5 <- as.data.frame(t(dat1[grep("^L005", rownames(dat1)),]))
L6 <- as.data.frame(t(dat1[grep("^L006", rownames(dat1)),]))
L7 <- as.data.frame(t(dat1[grep("^L007", rownames(dat1)),]))
L8 <- as.data.frame(t(dat1[grep("^L008", rownames(dat1)),]))
LZ2 <- as.data.frame(t(dat1[grep("^LZ2_", rownames(dat1)),]))
Z25A <- as.data.frame(t(dat1[grep("^LZ25A", rownames(dat1)),]))
Z30 <- as.data.frame(t(dat1[grep("^LZ30", rownames(dat1)),]))
Z40 <- as.data.frame(t(dat1[grep("^LZ40", rownames(dat1)),]))
PALM <- as.data.frame(t(dat1[grep("^PALMOUT", rownames(dat1)),]))
PEL <- as.data.frame(t(dat1[grep("^PELBAY3", rownames(dat1)),]))
POLE3S <- as.data.frame(t(dat1[grep("^POLE3S", rownames(dat1)),]))
```

```
PO <- as.data.frame(t(dat1[grep("^POLESOUT", rownames(dat1)),]))
RIT <- as.data.frame(t(dat1[grep("^RITTAE2", rownames(dat1)),]))
S308 <- as.data.frame(t(dat1[grep("^S308", rownames(dat1)),]))
S77 <- as.data.frame(t(dat1[grep("^S77", rownames(dat1)),]))
S79 <- as.data.frame(t(dat1[grep("^S79", rownames(dat1)),]))

#Year 2 Stations
CLV <- as.data.frame(t(dat2[grep("^CLV10A", rownames(dat2)),]))
KISS <- as.data.frame(t(dat2[grep("^KISSR0.0", rownames(dat2)),]))
L1 <- as.data.frame(t(dat2[grep("^L001", rownames(dat2)),]))
L4 <- as.data.frame(t(dat2[grep("^L004", rownames(dat2)),]))
L5 <- as.data.frame(t(dat2[grep("^L005", rownames(dat2)),]))
L6 <- as.data.frame(t(dat2[grep("^L006", rownames(dat2)),]))
L7 <- as.data.frame(t(dat2[grep("^L007", rownames(dat2)),]))
L8 <- as.data.frame(t(dat2[grep("^L008", rownames(dat2)),]))
LZ2 <- as.data.frame(t(dat2[grep("^LZ2_", rownames(dat2)),]))
Z25A <- as.data.frame(t(dat2[grep("^LZ25A", rownames(dat2)),]))
Z30 <- as.data.frame(t(dat2[grep("^LZ30", rownames(dat2)),]))
Z40 <- as.data.frame(t(dat2[grep("^LZ40", rownames(dat2)),]))
PALM <- as.data.frame(t(dat2[grep("^PALMOUT", rownames(dat2)),]))
PEL <- as.data.frame(t(dat2[grep("^PELBAY3", rownames(dat2)),]))
POLE3S <- as.data.frame(t(dat2[grep("^POLE3S", rownames(dat2)),]))
PO <- as.data.frame(t(dat2[grep("^POLESOUT", rownames(dat2)),]))
RIT <- as.data.frame(t(dat2[grep("^RITTAE2", rownames(dat2)),]))
S308 <- as.data.frame(t(dat2[grep("^S308", rownames(dat2)),]))
S77 <- as.data.frame(t(dat2[grep("^S77", rownames(dat2)),]))
S79 <- as.data.frame(t(dat2[grep("^S79", rownames(dat2)),]))

#Year 3 Stations
CLV <- as.data.frame(t(dat3[grep("^CLV10A", rownames(dat3)),]))
KISS <- as.data.frame(t(dat3[grep("^KISSR0.0", rownames(dat3)),]))
L1 <- as.data.frame(t(dat3[grep("^L001", rownames(dat3)),]))
L4 <- as.data.frame(t(dat3[grep("^L004", rownames(dat3)),]))
L5 <- as.data.frame(t(dat3[grep("^L005", rownames(dat3)),]))
L6 <- as.data.frame(t(dat3[grep("^L006", rownames(dat3)),]))
L7 <- as.data.frame(t(dat3[grep("^L007", rownames(dat3)),]))
L8 <- as.data.frame(t(dat3[grep("^L008", rownames(dat3)),]))
LZ2 <- as.data.frame(t(dat3[grep("^LZ2_", rownames(dat3)),]))
Z25A <- as.data.frame(t(dat3[grep("^LZ25A", rownames(dat3)),]))
Z30 <- as.data.frame(t(dat3[grep("^LZ30", rownames(dat3)),]))
Z40 <- as.data.frame(t(dat3[grep("^LZ40", rownames(dat3)),]))
PALM <- as.data.frame(t(dat3[grep("^PALMOUT", rownames(dat3)),]))
PEL <- as.data.frame(t(dat3[grep("^PELBAY3", rownames(dat3)),]))
POLE3S <- as.data.frame(t(dat3[grep("^POLE3S", rownames(dat3)),]))
PO <- as.data.frame(t(dat3[grep("^POLESOUT", rownames(dat3)),]))
RIT <- as.data.frame(t(dat3[grep("^RITTAE2", rownames(dat3)),]))
S308 <- as.data.frame(t(dat3[grep("^S308", rownames(dat3)),]))
S77 <- as.data.frame(t(dat3[grep("^S77", rownames(dat3)),]))
S79 <- as.data.frame(t(dat3[grep("^S79", rownames(dat3)),]))


###### TOP 10 TAXA BAR CHART - ALL YEARS TOGETHER ######
asvdat <- as.data.frame(t(dat.ra))
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
#Merging metadata, taxonomy, and ASV tables into one phyloseq object
physeq <- phyloseq(ASV,TAX,META)
#Use transform functions from microbiome package
transform <- microbiome::transform
#Merge rare taxa in to "Other"
physeq_transform <- transform(physeq, "compositional")
ASV  # 8,340 taxa & 541 samples
TAX  # 8,340 taxa by 7 tax. ranks
META # 541 samples by 42 sample variables

### Basic stats of seq. reads
#Check number of microbes observed in each sample
sample_sums(physeq)
##Basic stats for reads of samples
sum(sample_sums(physeq))
#Total reads = 24,093,755
```

```
mean(sample_sums(physeq))
#Mean = 44,535.59
min(sample_sums(physeq))
#Min= 10,029
max(sample_sums(physeq))
#Max = 193,655
sd(sample_sums(physeq))
#Stan.Dev = 24,782.95
ntaxa(physeq)
#Total ASVs = 65,294

physeq
# phyloseq-class experiment-level object
# otu_table()   OTU Table:         [ 8340 taxa and 541 samples ]
# sample_data() Sample Data:       [ 541 samples by 42 sample variables ]
# tax_table()   Taxonomy Table:    [ 8340 taxa by 7 taxonomic ranks ]

##Retrieves the unique taxonomic ranks observed in the data set
##[#] = rank (starting from Domain and onward DPCOFGS)
get_taxa_unique(physeq, taxonomic.rank=rank_names(physeq)[7], errorIfNULL=TRUE)
#Unique Domains = 4
#Unique Phyla = 56
#Unique Classes = 142
#Unique Orders = 351
#Unique Families = 508
#Unique Genera = 728
#Unique Species = 317

## make sure there is a phyloseq object which includes the data, metadata, and taxonomy ##

## Aggregating by Taxa levels
phyPhy <- aggregate_taxa(physeq, 'Phylum')
phyClass <- aggregate_taxa(physeq, 'Class')
phyOrd <- aggregate_taxa(physeq, 'Order')
phyGen <- aggregate_taxa(physeq, 'Genus')
LakeOPhy <- as.data.frame(taxa_sums(phyPhy))
LakeOClass <- as.data.frame(taxa_sums(phyClass))
LakeOOrd <- as.data.frame(taxa_sums(phyOrd))
LakeOGenus <- as.data.frame(taxa_sums(phyGen))
#Saving each table as CSV
write.csv(LakeOPhy, "LakeOPhylaTotals.csv")
write.csv(LakeOClass, "LakeOClassesTotals.csv")
write.csv(LakeOOrd, "LakeOOrdersTotals.csv")
write.csv(LakeOGenus, "LakeOGeneraTotals.csv")

## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names <- sort(tapply(taxa_sums(physeq_transform), tax_table(physeq_transform)[, "Phylum"], sum),
TRUE)[1:10]
## write.csv(top10phy.names, "Top10PhylaLakeO.csv")
# Proteobacteria       Bacteroidota      Cyanobacteria   Actinobacteriota  Verrucomicrobiota    Planctomycetota
Acidobacteriota
# 121.550676         110.168874          81.682736          57.976055          38.301827          34.610471
15.164802
# Bdellovibrionota        Chloroflexi    Gemmatimonadota
# 14.615002          11.278973           9.640009
#Cut down the physeq data to only the top 10 Phyla
top10phyla <- subset_taxa(physeq_transform, Phylum %in% names(top10phy.names))

## Plotting taxa stacked bar based on Zone
LakePhylaZ <- plot_bar(top10phyla, x="Zone", y="Abundance", fill="Phylum")
LakePhylaZ <- LakePhylaZ +
  geom_bar(aes(fill=Phylum), stat="identity", position="fill", width = 0.96)+   #width=0.96 removes any space
between bars
  ggtitle("Top 10 Phyla in Lake Okeechobee by Zone - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free",
             labeller = as_labeller(c('1'='Year 1 (2019)',
                                      '2'='Year 2 (2020)',
                                      '3'='Year 3 (2021)')))+   #scales=free -> allows ggplot to change the axes
for the data shown in each facet
  theme_light()+                                                #labeller -> changing the labels of the grid
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+   #vjust= moves the x-axis text labels
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+   #hjust= 0.5
centers the title
  theme(legend.title = element_text(face="italic"))
##Changing the color (by changing the default in ggplot2 [from HELP])
```

```
LakeOTop10 <- c("#2bcaf4","#24630e","#edc427","#1f60aa","#333333",
                        "#41ea27","#806bb4","#5f421b","#f08539","#ff9eed")
                        ## listed by phyla in alphabetical order
withr::with_options(list(ggplot2.discrete.fill = LakeOTop10, ggplot2.discrete.colour =
LakeOTop10),print(LakePhylaZ))

###### Top 10 phyla each year (CSVs) ######
#Subsetting original ASV table by year
Y1r <- dat.ra[grep("_19$", rownames(dat.ra)),]
Y2r <- dat.ra[grep("_20$", rownames(dat.ra)),]
Y3r <- dat.ra[grep("_21$", rownames(dat.ra)),]
write.csv(t(Y1), "Year1_RA.csv")
write.csv(t(Y2), "Year2_RA.csv")
write.csv(t(Y3), "Year3_RA.csv")

# OR

#Load in data if already exported to CSVs
Y1r <- read.csv("Year1_RA.csv", row.names = 1)
Y2r <- read.csv("Year2_RA.csv", row.names = 1)
Y3r <- read.csv("Year3_RA.csv", row.names = 1)

#Top 10 phyla in each year
##Year 1
asvdat <- Y1r
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyY1<- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyY1_transform <- transform(phyY1, "compositional")
### Assigning Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.Y1 <- sort(tapply(taxa_sums(phyY1_transform), tax_table(phyY1_transform)[, "Phylum"], sum),
TRUE)[1:10]
top10phy.names.Y1
# Proteobacteria       Bacteroidota      Cyanobacteria   Actinobacteriota    Planctomycetota  Verrucomicrobiota
Bdellovibrionota
# 37.118712             34.048403          18.633005          16.562391          11.084878          10.877789
5.230602
# Acidobacteriota        Chloroflexi      Crenarchaeota
# 4.481436                3.249468           2.864711
#Cut down the physeq data to only the top 10 Phyla
top10phylaY1 <- subset_taxa(phyY1_transform, Phylum %in% names(top10phy.names.Y1))
#Saving names and proportions as a data frame then saving as csv
topphylaY1 <- as.data.frame(top10phy.names.Y1)
colnames(topphylaY1)[1] ="Abundance"
write.csv(topphylaY1, "Top10Phyla_Year1.csv")

##Year 2
asvdat <- Y2r
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyY2<- phyloseq(ASV,TAX,META)
phyY2_transform <- transform(phyY2, "compositional")
### Assigning Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.Y2 <- sort(tapply(taxa_sums(phyY2_transform), tax_table(phyY2_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaY2 <- subset_taxa(phyY2_transform, Phylum %in% names(top10phy.names.Y2))
#Saving names and proportions as a data frame then saving as csv
topphylaY2 <- as.data.frame(top10phy.names.Y2)
colnames(topphylaY2)[1] ="Abundance"
write.csv(topphylaY2, "Top10Phyla_Year2.csv")

##Year 3
```

```
asvdat <- Y3r
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyY3<- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyY3_transform <- transform(phyY3, "compositional")
### Assigning Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.Y3 <- sort(tapply(taxa_sums(phyY3_transform), tax_table(phyY3_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaY3 <- subset_taxa(phyY3_transform, Phylum %in% names(top10phy.names.Y3))
#Saving names and proportions as a data frame then saving as csv
topphylaY3 <- as.data.frame(top10phy.names.Y3)
colnames(topphylaY3)[1] ="Abundance"
write.csv(topphylaY3, "Top10Phyla_Year3.csv")


###### Top 10 by Stations (CSVs) - ALL YEARS TOGETHER ######

## Use sample name order from Metadata file to keep samples in chronological order
#Note: psmelt() turns phyloseq object into a large dataframe that is in LONG format

## CLV10A
asvdat <- CLV
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyCLV <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyCLV_transform <- transform(phyCLV, "compositional")
### Assigning Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.CLV <- sort(tapply(taxa_sums(phyCLV_transform), tax_table(phyCLV_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaCLV <- subset_taxa(phyCLV_transform, Phylum %in% names(top10phy.names.CLV))
#Saving names and proportions as a data frame then saving as csv
topphylaCLV <- as.data.frame(top10phy.names.CLV)
colnames(topphylaCLV)[1] ="Abundance"
write.csv(topphylaCLV, "Top10Phyla_CLV.csv")


## KISSR0.0 - (Firmicutes removed-> KISSR0.0_3_20)
asvdat <- KISS
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyKISS <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyKISS_transform <- transform(phyKISS, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.KISS <- sort(tapply(taxa_sums(phyKISS_transform), tax_table(phyKISS_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaKISS <- subset_taxa(phyKISS_transform, Phylum %in% names(top10phy.names.KISS))
#Saving names and proportions as a data frame then saving as csv
topphylaKISS <- as.data.frame(top10phy.names.KISS)
colnames(topphylaKISS)[1] ="Abundance"
write.csv(topphylaKISS, "Top10Phyla_KISS.csv")
```

```
## L001
asvdat <- L1
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL1 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL1_transform <- transform(phyL1, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L1 <- sort(tapply(taxa_sums(phyL1_transform), tax_table(phyL1_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL1 <- subset_taxa(phyL1_transform, Phylum %in% names(top10phy.names.L1))
#Saving names and proportions as a data frame then saving as csv
topphylaL1 <- as.data.frame(top10phy.names.L1)
colnames(topphylaL1)[1] ="Abundance"
write.csv(topphylaL1, "Top10Phyla_L001.csv")


## L004
asvdat <- L4
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL4 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL4_transform <- transform(phyL4, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L4 <- sort(tapply(taxa_sums(phyL4_transform), tax_table(phyL4_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL4 <- subset_taxa(phyL4_transform, Phylum %in% names(top10phy.names.L4))
#Saving names and proportions as a data frame then saving as csv
topphylaL4 <- as.data.frame(top10phy.names.L4)
colnames(topphylaL4)[1] ="Abundance"
write.csv(topphylaL4, "Top10Phyla_L004.csv")


## L005 (Firmicutes removed-> L005_3_20)
asvdat <- L5
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL5 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL5_transform <- transform(phyL5, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L5 <- sort(tapply(taxa_sums(phyL5_transform), tax_table(phyL5_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL5 <- subset_taxa(phyL5_transform, Phylum %in% names(top10phy.names.L5))
#Saving names and proportions as a data frame then saving as csv
topphylaL5 <- as.data.frame(top10phy.names.L5)
colnames(topphylaL5)[1] ="Abundance"
write.csv(topphylaL5, "Top10Phyla_L005.csv")


## L006
asvdat <- L6
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
```

```
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL6 <- phyloseq(ASV,TAX,META)
phyL6_transform <- transform(phyL6, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L6 <- sort(tapply(taxa_sums(phyL6_transform), tax_table(phyL6_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL6 <- subset_taxa(phyL6_transform, Phylum %in% names(top10phy.names.L6))
#Saving names and proportions as a data frame then saving as csv
topphylaL6 <- as.data.frame(top10phy.names.L6)
colnames(topphylaL6)[1] ="Abundance"
write.csv(topphylaL6, "Top10Phyla_L006.csv")


## L007
asvdat <- L7
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL7 <- phyloseq(ASV,TAX,META)
phyL7_transform <- transform(phyL7, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L7 <- sort(tapply(taxa_sums(phyL7_transform), tax_table(phyL7_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL7 <- subset_taxa(phyL7_transform, Phylum %in% names(top10phy.names.L7))
#Saving names and proportions as a data frame then saving as csv
topphylaL7 <- as.data.frame(top10phy.names.L7)
colnames(topphylaL7)[1] ="Abundance"
write.csv(topphylaL7, "Top10Phyla_L007.csv")

## L008
asvdat <- L8
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL8 <- phyloseq(ASV,TAX,META)
phyL8_transform <- transform(phyL8, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L8 <- sort(tapply(taxa_sums(phyL8_transform), tax_table(phyL8_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL8 <- subset_taxa(phyL8_transform, Phylum %in% names(top10phy.names.L8))
#Saving names and proportions as a data frame then saving as csv
topphylaL8 <- as.data.frame(top10phy.names.L8)
colnames(topphylaL8)[1] ="Abundance"
write.csv(topphylaL8, "Top10Phyla_L008.csv")

## LZ25A
asvdat <- Z25A
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy25A <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phy25A_transform <- transform(phy25A, "compositional")
## Top 10 Phyla
```

```
#Sort Phylum by abundance and pick the top 10
top10phy.names.25A <- sort(tapply(taxa_sums(phy25A_transform), tax_table(phy25A_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla25A <- subset_taxa(phy25A_transform, Phylum %in% names(top10phy.names.25A))
#Saving names and proportions as a data frame then saving as csv
topphyla25A <- as.data.frame(top10phy.names.25A)
colnames(topphyla25A)[1] ="Abundance"
write.csv(topphyla25A, "Top10Phyla_LZ25A.csv")

## LZ2 (Firmicutes contam. removed LZ2_3_20)
asvdat <- LZ2
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyLZ2 <- phyloseq(ASV,TAX,META)
phyLZ2_transform <- transform(phyLZ2, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.LZ2 <- sort(tapply(taxa_sums(phyLZ2_transform), tax_table(phyLZ2_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaLZ2 <- subset_taxa(phyLZ2_transform, Phylum %in% names(top10phy.names.LZ2))
#Saving names and proportions as a data frame then saving as csv
topphylaLZ2 <- as.data.frame(top10phy.names.LZ2)
colnames(topphylaLZ2)[1] ="Abundance"
write.csv(topphylaLZ2, "Top10Phyla_LZ2.csv")

## LZ30
asvdat <- Z30
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy30 <- phyloseq(ASV,TAX,META)
phy30_transform <- transform(phy30, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.30 <- sort(tapply(taxa_sums(phy30_transform), tax_table(phy30_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla30 <- subset_taxa(phy30_transform, Phylum %in% names(top10phy.names.30))
#Saving names and proportions as a data frame then saving as csv
topphyla30 <- as.data.frame(top10phy.names.30)
colnames(topphyla30)[1] ="Abundance"
write.csv(topphyla30, "Top10Phyla_LZ30.csv")

## LZ40
asvdat <- Z40
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy40 <- phyloseq(ASV,TAX,META)
phy40_transform <- transform(phy40, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.40 <- sort(tapply(taxa_sums(phy40_transform), tax_table(phy40_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla40 <- subset_taxa(phy40_transform, Phylum %in% names(top10phy.names.40))
#Saving names and proportions as a data frame then saving as csv
topphyla40 <- as.data.frame(top10phy.names.40)
colnames(topphyla40)[1] ="Abundance"
write.csv(topphyla40, "Top10Phyla_LZ40.csv")
```

```
## PALMOUT (Firmicutes contam. removed PALMOUT_3_20)
asvdat <- PALM
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPALM <- phyloseq(ASV,TAX,META)
phyPALM_transform <- transform(phyPALM, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PALM <- sort(tapply(taxa_sums(phyPALM_transform), tax_table(phyPALM_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPALM <- subset_taxa(phyPALM_transform, Phylum %in% names(top10phy.names.PALM))
#Saving names and proportions as a data frame then saving as csv
topphylaPALM <- as.data.frame(top10phy.names.PALM)
colnames(topphylaPALM)[1] ="Abundance"
write.csv(topphylaPALM, "Top10Phyla_PALM.csv")

## PELBAY3 - DONE ON 11/12/22
asvdat <- PEL
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPEL <- phyloseq(ASV,TAX,META)
phyPEL_transform <- transform(phyPEL, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PEL <- sort(tapply(taxa_sums(phyPEL_transform), tax_table(phyPEL_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPEL <- subset_taxa(phyPEL_transform, Phylum %in% names(top10phy.names.PEL))
#Saving names and proportions as a data frame then saving as csv
topphylaPEL <- as.data.frame(top10phy.names.PEL)
colnames(topphylaPEL)[1] ="Abundance"
write.csv(topphylaPEL, "Top10Phyla_PEL.csv")

## POLE3S - DONE ON 11/12/22 (Firmicutes contam. removed POLE3S_3_20)
asvdat <- POLE3S
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPOLE3S <- phyloseq(ASV,TAX,META)
phyPOLE3S_transform <- transform(phyPOLE3S, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.POLE3S <- sort(tapply(taxa_sums(phyPOLE3S_transform), tax_table(phyPOLE3S_transform)[, "Phylum"],
sum), TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPOLE3S <- subset_taxa(phyPOLE3S_transform, Phylum %in% names(top10phy.names.POLE3S))
#Saving names and proportions as a data frame then saving as csv
topphylaPOLE3S <- as.data.frame(top10phy.names.POLE3S)
colnames(topphylaPOLE3S)[1] ="Abundance"
write.csv(topphylaPOLE3S, "Top10Phyla_POLE3S.csv")

## POLESOUT - DONE ON 11/12/22 (Firmicutes contam. removed POLESOUT_3_20)
asvdat <- PO
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPO <- phyloseq(ASV,TAX,META)
```

```
phyPO_transform <- transform(phyPO, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PO <- sort(tapply(taxa_sums(phyPO_transform), tax_table(phyPO_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPO <- subset_taxa(phyPO_transform, Phylum %in% names(top10phy.names.PO))
#Saving names and proportions as a data frame then saving as csv
topphylaPO <- as.data.frame(top10phy.names.PO)
colnames(topphylaPO)[1] ="Abundance"
write.csv(topphylaPO, "Top10Phyla_PO.csv")

## RITTAE2 - DONE ON 11/12/22 (Firmicutes contam. removed RITTAE2_3_20)
asvdat <- RIT
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyRIT <- phyloseq(ASV,TAX,META)
phyRIT_transform <- transform(phyRIT, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.RIT <- sort(tapply(taxa_sums(phyRIT_transform), tax_table(phyRIT_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaRIT <- subset_taxa(phyRIT_transform, Phylum %in% names(top10phy.names.RIT))
#Saving names and proportions as a data frame then saving as csv
topphylaRIT <- as.data.frame(top10phy.names.RIT)
colnames(topphylaRIT)[1] ="Abundance"
write.csv(topphylaRIT, "Top10Phyla_RIT.csv")

## S308
asvdat <- S308
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS308 <- phyloseq(ASV,TAX,META)
phyS308_transform <- transform(phyS308, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S308 <- sort(tapply(taxa_sums(phyS308_transform), tax_table(phyS308_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS308 <- subset_taxa(phyS308_transform, Phylum %in% names(top10phy.names.S308))
#Saving names and proportions as a data frame then saving as csv
topphylaS308 <- as.data.frame(top10phy.names.S308)
colnames(topphylaS308)[1] ="Abundance"
write.csv(topphylaS308, "Top10Phyla_S308.csv")

## S77  (Firmicutes contam. removed S77_3_20)
asvdat <- S77
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS77 <- phyloseq(ASV,TAX,META)
phyS77_transform <- transform(phyS77, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S77 <- sort(tapply(taxa_sums(phyS77_transform), tax_table(phyS77_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS77 <- subset_taxa(phyS77_transform, Phylum %in% names(top10phy.names.S77))
#Saving names and proportions as a data frame then saving as csv
topphylaS77 <- as.data.frame(top10phy.names.S77)
colnames(topphylaS77)[1] ="Abundance"
```

```
write.csv(topphylaS77, "Top10Phyla_S77.csv")

## S79 (Firmicutes contam. removed S79_3_20)
asvdat <- S79
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS79 <- phyloseq(ASV,TAX,META)
phyS79_transform <- transform(phyS79, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S79 <- sort(tapply(taxa_sums(phyS79_transform), tax_table(phyS79_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS79 <- subset_taxa(phyS79_transform, Phylum %in% names(top10phy.names.S79))
#Saving names and proportions as a data frame then saving as csv
topphylaS79 <- as.data.frame(top10phy.names.S79)
colnames(topphylaS79)[1] ="Abundance"
write.csv(topphylaS79, "Top10Phyla_S79.csv")

###### Plotting Taxonomy Bar plots using phyloseq - ALL YEARS TOGETHER ######
#Defining the initial plot
CLV <- plot_bar(top10phylaCLV, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
CLV$data$Sample <- as.factor(CLV$data$Sample) #Assigning the samples as factors so I can manually put the levels
in order
levels(CLV$data$Sample) #making sure each sample name is a level (should be 28 levels)
#Samples ARE NOT in chronological order here
CLV$data$Sample <- factor(CLV$data$Sample,
levels=c("CLV10A_4_19","CLV10A_5_19","CLV10A_6_19","CLV10A_7_19","CLV10A_8_19",

"CLV10A_9_19","CLV10A_10_19","CLV10A_11_19","CLV10A_12_19","CLV10A_1_20",

"CLV10A_2_20","CLV10A_3_20","CLV10A_4_20","CLV10A_6_20","CLV10A_7_20",

"CLV10A_8_20","CLV10A_9_20","CLV10A_10_20","CLV10A_12_20","CLV10A_1_21",

"CLV10A_2_21","CLV10A_3_21","CLV10A_4_21","CLV10A_5_21","CLV10A_6_21",
                                                    "CLV10A_7_21","CLV10A_8_21","CLV10A_10_21"))
levels(CLV$data$Sample) #Samples ARE in chronological order now
#Customizing the plot using ggplot2's geom_bar
CLV +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.96)+   #width=0.96
removes any space between bars
  ggtitle("Top 10 Phyla at CLV10A - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free",
             labeller = as_labeller(c('1'='Year 1 (2019)',
                                      '2'='Year 2 (2020)',
                                      '3'='Year 3 (2021)')))+   #scales=free -> allows ggplot to change the axes
for the data shown in each facet
  theme_light()+                                                #labeller -> changing the labels of the grid
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+   #vjust= moves the x-axis text labels
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+   #hjust= 0.5
centers the title
  theme(legend.title = element_text(face="italic"))
#facet_grid - splits up the graph into the variable specified
#position=fill - bars go up to 1.00, while position=stack - bar shows actual abundance (bars don't line up)

#Defining the initial plot
KISS <- plot_bar(top10phylaKISS, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
KISS$data$Sample <- as.factor(KISS$data$Sample)
levels(KISS$data$Sample)
KISS$data$Sample <- factor(KISS$data$Sample,
levels=c("KISSR0.0_3_19","KISSR0.0_4_19","KISSR0.0_5_19","KISSR0.0_7_19","KISSR0.0_8_19","KISSR0.0_9_19",

"KISSR0.0_11_19","KISSR0.0_12_19","KISSR0.0_1_20","KISSR0.0_2_20","KISSR0.0_4_20",

"KISSR0.0_5_20","KISSR0.0_6_20","KISSR0.0_8_20","KISSR0.0_9_20","KISSR0.0_10_20","KISSR0.0_11_20",

"KISSR0.0_12_20","KISSR0.0_2_21","KISSR0.0_3_21","KISSR0.0_4_21","KISSR0.0_5_21","KISSR0.0_6_21",
```

```
"KISSR0.0_7_21","KISSR0.0_8_21","KISSR0.0_9_21","KISSR0.0_10_21"))
levels(KISS$data$Sample)
#Customizing the plot using ggplot2's geom_bar
KISS +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at KISSR0.0 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

#Defining the initial plot
L1 <- plot_bar(top10phylaL1, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
L1$data$Sample <- as.factor(L1$data$Sample)
levels(L1$data$Sample)
L1$data$Sample <- factor(L1$data$Sample,
levels=c("L001_3_19","L001_4_19","L001_5_19","L001_6_19","L001_7_19","L001_8_19","L001_9_19",

"L001_11_19","L001_12_19","L001_1_20","L001_2_20","L001_3_20","L001_4_20",

"L001_6_20","L001_7_20","L001_8_20","L001_9_20","L001_10_20","L001_11_20",

"L001_12_20","L001_2_21","L001_3_21","L001_4_21","L001_5_21","L001_6_21",
                                                 "L001_7_21","L001_8_21","L001_9_21","L001_10_21"))
levels(L1$data$Sample)
#Customizing the plot using ggplot2's geom_bar
L1 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at L001 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

#Defining the initial plot
L4 <- plot_bar(top10phylaL4, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
L4$data$Sample <- as.factor(L4$data$Sample)
levels(L4$data$Sample)
L4$data$Sample <- factor(L4$data$Sample, levels=c("L004_3_19","L004_5_19","L004_8_19","L004_9_19",

"L004_11_19","L004_12_19","L004_1_20","L004_2_20","L004_3_20","L004_4_20",

"L004_6_20","L004_7_20","L004_8_20","L004_9_20","L004_10_20","L004_11_20",
                                                 "L004_12_20","L004_2_21","L004_3_21","L004_4_21","L004_6_21",
                                                 "L004_7_21","L004_8_21","L004_10_21"))
levels(L4$data$Sample)
#Customizing the plot using ggplot2's geom_bar
L4 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at L004 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
## Top 10 Classes - 12/01/22
#Sort Class by abundance and pick the top 10
top10class.names.L4 <- sort(tapply(taxa_sums(phyL4_transform), tax_table(phyL4_transform)[, "Class"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 classes
top10classL4 <- subset_taxa(phyL4_transform, Class %in% names(top10class.names.L4))
#Saving names and proportions as a data frame then saving as csv
topclassL4 <- as.data.frame(top10class.names.L4)
colnames(topclassL4)[1] ="Abundance"
write.csv(topclassL4, "Top10Classes_L004.csv")
```

```
### Plotting the graph -PHYLUM
#Defining the initial plot
L4c <- plot_bar(top10classL4, x="Sample", y="Abundance", fill = "Class")
#Reordering the samples so they plot in chronological order
L4c$data$Sample <- as.factor(L4c$data$Sample)
levels(L4c$data$Sample)
L4c$data$Sample <- factor(L4c$data$Sample, levels=c("L004_3_19","L004_5_19","L004_8_19","L004_9_19",

"L004_11_19","L004_12_19","L004_1_20","L004_2_20","L004_3_20","L004_4_20",

"L004_6_20","L004_7_20","L004_8_20","L004_9_20","L004_10_20","L004_11_20",

"L004_12_20","L004_2_21","L004_3_21","L004_4_21","L004_6_21",
                                                    "L004_7_21","L004_8_21","L004_10_21"))
levels(L4c$data$Sample)
#Customizing the plot using ggplot2's geom_bar
L4c +
  geom_bar(aes(color=Class, fill=Class), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Classes at L004 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

#Defining the initial plot
L5 <- plot_bar(top10phylaL5, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
L5$data$Sample <- as.factor(L5$data$Sample)
levels(L5$data$Sample)
L5$data$Sample <- factor(L5$data$Sample,
levels=c("L005_3_19","L005_4_19","L005_5_19","L005_6_19","L005_7_19","L005_8_19","L005_9_19",
                                                  "L005_11_19","L005_12_19","L005_1_20","L005_2_20","L005_4_20",

"L005_6_20","L005_7_20","L005_8_20","L005_9_20","L005_10_20","L005_11_20",

"L005_12_20","L005_2_21","L005_3_21","L005_4_21","L005_5_21","L005_6_21",
                                                  "L005_7_21","L005_8_21","L005_9_21","L005_10_21"))
levels(L5$data$Sample)
#Customizing the plot using ggplot2's geom_bar
L5 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at L005 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

#Defining the initial plot
L6 <- plot_bar(top10phylaL6, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
L6$data$Sample <- as.factor(L6$data$Sample)
levels(L6$data$Sample)
L6$data$Sample <- factor(L6$data$Sample, levels=c("L006_5_19","L006_7_19","L006_8_19","L006_9_19",

"L006_11_19","L006_12_19","L006_1_20","L006_2_20","L006_3_20","L006_4_20",

"L006_5_20","L006_6_20","L006_7_20","L006_8_20","L006_9_20","L006_10_20","L006_11_20",

"L006_12_20","L006_1_21","L006_2_21","L006_3_21","L006_4_21","L006_5_21","L006_6_21",
                                                  "L006_7_21","L006_8_21","L006_10_21"))
levels(L6$data$Sample)
#Customizing the plot using ggplot2's geom_bar
L6 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at L006 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
```

```
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

### Plotting the graph
#Defining the initial plot
L7 <- plot_bar(top10phylaL7, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
L7$data$Sample <- as.factor(L7$data$Sample)
levels(L7$data$Sample)
L7$data$Sample <- factor(L7$data$Sample,
levels=c("L007_3_19","L007_4_19","L007_5_19","L007_6_19","L007_7_19","L007_8_19","L007_9_19",

"L007_11_19","L007_12_19","L007_1_20","L007_2_20","L007_3_20","L007_4_20","L007_5_20",
                                                  "L007_6_20","L007_8_20","L007_9_20","L007_10_20","L007_11_20",

"L007_12_20","L007_1_21","L007_2_21","L007_3_21","L007_4_21","L007_5_21","L007_6_21",
                                                  "L007_7_21","L007_8_21","L007_9_21","L007_10_21"))
levels(L7$data$Sample)
#Customizing the plot using ggplot2's geom_bar
L7 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at L007 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

### Plotting the graph
#Defining the initial plot
L8 <- plot_bar(top10phylaL8, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
L8$data$Sample <- as.factor(L8$data$Sample)
levels(L8$data$Sample)
L8$data$Sample <- factor(L8$data$Sample,
levels=c("L008_3_19","L008_5_19","L008_6_19","L008_7_19","L008_8_19","L008_9_19",

"L008_11_19","L008_12_19","L008_1_20","L008_2_20","L008_3_20","L008_4_20","L008_5_20",

"L008_6_20","L008_7_20","L008_8_20","L008_9_20","L008_10_20","L008_11_20",

"L008_12_20","L008_2_21","L008_3_21","L008_4_21","L008_5_21","L008_6_21",
                                                  "L008_7_21","L008_8_21","L008_10_21"))
levels(L8$data$Sample)
#Customizing the plot using ggplot2's geom_bar
L8 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at L008 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

### Plotting the graph
#Defining the initial plot
Z25A <- plot_bar(top10phyla25A, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
Z25A$data$Sample <- as.factor(Z25A$data$Sample)
levels(Z25A$data$Sample)
Z25A$data$Sample <- factor(Z25A$data$Sample,
levels=c("LZ25A_3_19","LZ25A_4_19","LZ25A_6_19","LZ25A_7_19","LZ25A_8_19","LZ25A_9_19",

"LZ25A_11_19","LZ25A_12_19","LZ25A_1_20","LZ25A_2_20","LZ25A_3_20","LZ25A_4_20",

"LZ25A_5_20","LZ25A_7_20","LZ25A_8_20","LZ25A_9_20","LZ25A_10_20","LZ25A_11_20",

"LZ25A_12_20","LZ25A_1_21","LZ25A_2_21","LZ25A_3_21","LZ25A_4_21","LZ25A_5_21","LZ25A_6_21",
                                                    "LZ25A_7_21","LZ25A_10_21"))
levels(Z25A$data$Sample)
#Customizing the plot using ggplot2's geom_bar
Z25A +
```

```
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at LZ25A - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

### Plotting the graph
#Defining the initial plot
LZ2 <- plot_bar(top10phylaLZ2, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
LZ2$data$Sample <- as.factor(LZ2$data$Sample)
levels(LZ2$data$Sample)
LZ2$data$Sample <- factor(LZ2$data$Sample,
levels=c("LZ2_3_19","LZ2_4_19","LZ2_5_19","LZ2_6_19","LZ2_8_19","LZ2_9_19",
                                                    "LZ2_11_19","LZ2_12_19","LZ2_1_20","LZ2_2_20","LZ2_4_20",

"LZ2_5_20","LZ2_6_20","LZ2_7_20","LZ2_8_20","LZ2_9_20","LZ2_10_20","LZ2_11_20",

"LZ2_12_20","LZ2_2_21","LZ2_3_21","LZ2_4_21","LZ2_5_21","LZ2_6_21",
                                                    "LZ2_7_21","LZ2_8_21","LZ2_10_21"))
levels(LZ2$data$Sample)
#Customizing the plot using ggplot2's geom_bar
LZ2 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at LZ2 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))


### Plotting the graph
#Defining the initial plot
Z30 <- plot_bar(top10phyla30, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
Z30$data$Sample <- as.factor(Z30$data$Sample)
levels(Z30$data$Sample)
Z30$data$Sample <- factor(Z30$data$Sample,
levels=c("LZ30_4_19","LZ30_5_19","LZ30_6_19","LZ30_7_19","LZ30_8_19","LZ30_9_19","LZ30_10_19",

"LZ30_11_19","LZ30_12_19","LZ30_1_20","LZ30_2_20","LZ30_3_20","LZ30_4_20","LZ30_5_20",

"LZ30_6_20","LZ30_7_20","LZ30_8_20","LZ30_9_20","LZ30_10_20","LZ30_11_20",

"LZ30_12_20","LZ30_1_21","LZ30_2_21","LZ30_3_21","LZ30_4_21","LZ30_5_21","LZ30_6_21",
                                                    "LZ30_7_21","LZ30_8_21","LZ30_10_21"))
levels(Z30$data$Sample)
#Customizing the plot using ggplot2's geom_bar
Z30 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at LZ30 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

### Plotting the graph
#Defining the initial plot
Z40 <- plot_bar(top10phyla40, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
Z40$data$Sample <- as.factor(Z40$data$Sample)
levels(Z40$data$Sample)
Z40$data$Sample <- factor(Z40$data$Sample,
levels=c("LZ40_3_19","LZ40_4_19","LZ40_5_19","LZ40_6_19","LZ40_7_19","LZ40_8_19","LZ40_9_19",

"LZ40_11_19","LZ40_12_19","LZ40_1_20","LZ40_2_20","LZ40_3_20","LZ40_4_20",
```

```
"LZ40_5_20","LZ40_6_20","LZ40_7_20","LZ40_8_20","LZ40_9_20","LZ40_10_20",

"LZ40_12_20","LZ40_1_21","LZ40_2_21","LZ40_3_21","LZ40_4_21","LZ40_5_21","LZ40_6_21",
                                                    "LZ40_7_21","LZ40_8_21","LZ40_9_21","LZ40_10_21"))
levels(Z40$data$Sample)
#Customizing the plot using ggplot2's geom_bar
Z40 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at LZ40 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

### Plotting the graph
#Defining the initial plot
PALM <- plot_bar(top10phylaPALM, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
PALM$data$Sample <- as.factor(PALM$data$Sample)
levels(PALM$data$Sample)
PALM$data$Sample <- factor(PALM$data$Sample,
levels=c("PALMOUT_3_19","PALMOUT_4_19","PALMOUT_6_19","PALMOUT_7_19","PALMOUT_8_19",

"PALMOUT_11_19","PALMOUT_12_19","PALMOUT_1_20","PALMOUT_2_20","PALMOUT_4_20",

"PALMOUT_5_20","PALMOUT_6_20","PALMOUT_7_20","PALMOUT_8_20","PALMOUT_9_20","PALMOUT_10_20",

"PALMOUT_12_20","PALMOUT_1_21","PALMOUT_2_21","PALMOUT_3_21","PALMOUT_4_21","PALMOUT_5_21","PALMOUT_6_21",

"PALMOUT_7_21","PALMOUT_8_21","PALMOUT_9_21","PALMOUT_10_21"))
levels(PALM$data$Sample)

#Customizing the plot using ggplot2's geom_bar
PALM +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at PALMOUT - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))

### Plotting the graph
#Defining the initial plot
PEL <- plot_bar(top10phylaPEL, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
PEL$data$Sample <- as.factor(PEL$data$Sample)
levels(PEL$data$Sample)
PEL$data$Sample <- factor(PEL$data$Sample,
levels=c("PELBAY3_3_19","PELBAY3_5_19","PELBAY3_6_19","PELBAY3_7_19","PELBAY3_8_19","PELBAY3_9_19",

"PELBAY3_11_19","PELBAY3_12_19","PELBAY3_1_20","PELBAY3_2_20","PELBAY3_4_20","PELBAY3_5_20",

"PELBAY3_6_20","PELBAY3_7_20","PELBAY3_8_20","PELBAY3_9_20","PELBAY3_10_20","PELBAY3_11_20",

"PELBAY3_12_20","PELBAY3_1_21","PELBAY3_2_21","PELBAY3_3_21","PELBAY3_4_21","PELBAY3_5_21","PELBAY3_6_21",
                                                    "PELBAY3_7_21","PELBAY3_8_21","PELBAY3_10_21"))
levels(PEL$data$Sample)
#Customizing the plot using ggplot2's geom_bar
PEL +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at PELBAY3 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
```

```
### Plotting the graph
#Defining the initial plot
POLE3S <- plot_bar(top10phylaPOLE3S, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
POLE3S$data$Sample <- as.factor(POLE3S$data$Sample)
levels(POLE3S$data$Sample)
POLE3S$data$Sample <- factor(POLE3S$data$Sample,
levels=c("POLE3S_3_19","POLE3S_5_19","POLE3S_6_19","POLE3S_7_19","POLE3S_8_19",

"POLE3S_12_19","POLE3S_1_20","POLE3S_2_20","POLE3S_4_20",

"POLE3S_7_20","POLE3S_8_20","POLE3S_9_20","POLE3S_10_20","POLE3S_11_20",

"POLE3S_12_20","POLE3S_1_21","POLE3S_2_21","POLE3S_3_21","POLE3S_4_21","POLE3S_5_21","POLE3S_6_21",
                                                          "POLE3S_7_21","POLE3S_8_21","POLE3S_10_21"))
levels(POLE3S$data$Sample)
#Customizing the plot using ggplot2's geom_bar and exporting as PNG file
png("Top10PhylaPOLE3S.png", width = 885, height = 575) # creates a named png file in your working directory
POLE3S +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at POLE3S - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
dev.off() #stops writing to the png file and saves it

### Plotting the graph
#Defining the initial plot
PO <- plot_bar(top10phylaPO, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
PO$data$Sample <- as.factor(PO$data$Sample)
levels(PO$data$Sample)
PO$data$Sample <- factor(PO$data$Sample,
levels=c("POLESOUT_3_19","POLESOUT_4_19","POLESOUT_5_19","POLESOUT_6_19","POLESOUT_7_19","POLESOUT_8_19",

"POLESOUT_11_19","POLESOUT_1_20","POLESOUT_2_20","POLESOUT_4_20",

"POLESOUT_6_20","POLESOUT_7_20","POLESOUT_8_20","POLESOUT_9_20","POLESOUT_10_20","POLESOUT_11_20",

"POLESOUT_12_20","POLESOUT_2_21","POLESOUT_3_21","POLESOUT_4_21","POLESOUT_5_21","POLESOUT_6_21",

"POLESOUT_7_21","POLESOUT_8_21","POLESOUT_9_21","POLESOUT_10_21"))
levels(PO$data$Sample)
#Customizing the plot using ggplot2's geom_bar
png("Top10PhylaPOLESOUT.png", width = 885, height = 575)
PO +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at POLESOUT - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
dev.off()

### Plotting the graph
#Defining the initial plot
RIT <- plot_bar(top10phylaRIT, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
RIT$data$Sample <- as.factor(RIT$data$Sample)
levels(RIT$data$Sample)
RIT$data$Sample <- factor(RIT$data$Sample,
levels=c("RITTAE2_3_19","RITTAE2_4_19","RITTAE2_6_19","RITTAE2_7_19","RITTAE2_8_19",

"RITTAE2_11_19","RITTAE2_12_19","RITTAE2_1_20","RITTAE2_2_20","RITTAE2_4_20",

"RITTAE2_8_20","RITTAE2_9_20","RITTAE2_10_20","RITTAE2_11_20",

"RITTAE2_12_20","RITTAE2_1_21","RITTAE2_2_21","RITTAE2_3_21","RITTAE2_4_21","RITTAE2_5_21","RITTAE2_6_21",
                                                  "RITTAE2_7_21","RITTAE2_8_21","RITTAE2_10_21"))
```

```
levels(RIT$data$Sample)
#Customizing the plot using ggplot2's geom_bar
png("Top10PhylaRITTAE2.png", width = 885, height = 575)
RIT +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at RITTAE2 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
dev.off()

### Plotting the graph
#Defining the initial plot
S308 <- plot_bar(top10phylaS308, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
S308$data$Sample <- as.factor(S308$data$Sample)
levels(S308$data$Sample)
S308$data$Sample <- factor(S308$data$Sample,
levels=c("S308_3_19","S308_4_19","S308_5_19","S308_6_19","S308_7_19","S308_9_19","S308_10_19",

"S308_11_19","S308_12_19","S308_1_20","S308_2_20","S308_3_20","S308_4_20","S308_5_20",

"S308_6_20","S308_7_20","S308_8_20","S308_9_20","S308_10_20","S308_11_20",

"S308_12_20","S308_1_21","S308_2_21","S308_3_21","S308_4_21","S308_5_21","S308_6_21"))
levels(S308$data$Sample)
#Customizing the plot using ggplot2's geom_bar
png("Top10PhylaS308.png", width = 885, height = 575)
S308 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at S308 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
dev.off()

### Plotting the graph
#Defining the initial plot
S77 <- plot_bar(top10phylaS77, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
S77$data$Sample <- as.factor(S77$data$Sample)
levels(S77$data$Sample)
S77$data$Sample <- factor(S77$data$Sample,
levels=c("S77_3_19","S77_4_19","S77_5_19","S77_6_19","S77_7_19","S77_8_19","S77_10_19",
                                                    "S77_11_19","S77_12_19","S77_1_20","S77_2_20","S77_4_20",
                                                    "S77_6_20","S77_7_20","S77_8_20","S77_9_20","S77_10_20",

"S77_12_20","S77_2_21","S77_3_21","S77_4_21","S77_5_21","S77_6_21",
                                                    "S77_7_21","S77_8_21","S77_9_21","S77_10_21"))
levels(S77$data$Sample)
#Customizing the plot using ggplot2's geom_bar
png("Top10PhylaS77.png", width = 885, height = 575)
S77 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at S77 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
dev.off()

### Plotting the graph
#Defining the initial plot
S79 <- plot_bar(top10phylaS79, x="Sample", y="Abundance", fill = "Phylum")
#Reordering the samples so they plot in chronological order
```

```
S79$data$Sample <- as.factor(S79$data$Sample)
levels(S79$data$Sample)
S79$data$Sample <- factor(S79$data$Sample,
levels=c("S79_3_19","S79_4_19","S79_6_19","S79_7_19","S79_8_19","S79_12_19",
                                                    "S79_1_20","S79_2_20","S79_4_20",
                                                    "S79_7_20","S79_9_20","S79_10_20","S79_11_20",

"S79_12_20","S79_2_21","S79_3_21","S79_4_21","S79_5_21","S79_6_21",
                                                    "S79_7_21","S79_8_21","S79_10_21"))
levels(S79$data$Sample)
#Customizing the plot using ggplot2's geom_bar
png("Top10PhylaS79.png", width = 885, height = 575)
S79 +
  geom_bar(aes(color=Phylum, fill=Phylum), stat="identity", position="fill", width = 0.98)+
  ggtitle("Top 10 Phyla at S79 - March 2019 to October 2021")+
  facet_grid(.~Year, scales = "free", labeller = as_labeller(c('1'='Year 1 (2019)',
                                                               '2'='Year 2 (2020)',
                                                               '3'='Year 3 (2021)')))+
  theme_light()+
  theme(axis.text.x = element_text(angle = 90, vjust = 0.28))+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
dev.off()


###### Top 10 by Stations each Year - exporting CSVs ######
#### Year 1
## CLV10A
asvdat <- CLV
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyCLV <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyCLV_transform <- transform(phyCLV, "compositional")
### Assigning Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.CLV <- sort(tapply(taxa_sums(phyCLV_transform), tax_table(phyCLV_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaCLV <- subset_taxa(phyCLV_transform, Phylum %in% names(top10phy.names.CLV))
#Saving names and proportions as a data frame then saving as csv
topphylaCLV <- as.data.frame(top10phy.names.CLV)
colnames(topphylaCLV)[1] ="Abundance"
write.csv(topphylaCLV, "Top10Phyla_CLV_Y1.csv")

## KISSR0.0 - (Firmicutes removed-> KISSR0.0_3_20)
asvdat <- KISS
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyKISS <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyKISS_transform <- transform(phyKISS, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.KISS <- sort(tapply(taxa_sums(phyKISS_transform), tax_table(phyKISS_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaKISS <- subset_taxa(phyKISS_transform, Phylum %in% names(top10phy.names.KISS))
#Saving names and proportions as a data frame then saving as csv
topphylaKISS <- as.data.frame(top10phy.names.KISS)
colnames(topphylaKISS)[1] ="Abundance"
write.csv(topphylaKISS, "Top10Phyla_KISS_Y1.csv")

## L001
asvdat <- L1
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
```

```
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL1 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL1_transform <- transform(phyL1, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L1 <- sort(tapply(taxa_sums(phyL1_transform), tax_table(phyL1_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL1 <- subset_taxa(phyL1_transform, Phylum %in% names(top10phy.names.L1))
#Saving names and proportions as a data frame then saving as csv
topphylaL1 <- as.data.frame(top10phy.names.L1)
colnames(topphylaL1)[1] ="Abundance"
write.csv(topphylaL1, "Top10Phyla_L001_Y1.csv")

## L004
asvdat <- L4
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL4 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL4_transform <- transform(phyL4, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L4 <- sort(tapply(taxa_sums(phyL4_transform), tax_table(phyL4_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL4 <- subset_taxa(phyL4_transform, Phylum %in% names(top10phy.names.L4))
#Saving names and proportions as a data frame then saving as csv
topphylaL4 <- as.data.frame(top10phy.names.L4)
colnames(topphylaL4)[1] ="Abundance"
write.csv(topphylaL4, "Top10Phyla_L004_Y1.csv")

## L005 (Firmicutes removed-> L005_3_20)
asvdat <- L5
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL5 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL5_transform <- transform(phyL5, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L5 <- sort(tapply(taxa_sums(phyL5_transform), tax_table(phyL5_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL5 <- subset_taxa(phyL5_transform, Phylum %in% names(top10phy.names.L5))
#Saving names and proportions as a data frame then saving as csv
topphylaL5 <- as.data.frame(top10phy.names.L5)
colnames(topphylaL5)[1] ="Abundance"
write.csv(topphylaL5, "Top10Phyla_L005_Y1.csv")

## L006
asvdat <- L6
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL6 <- phyloseq(ASV,TAX,META)
```

```
phyL6_transform <- transform(phyL6, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L6 <- sort(tapply(taxa_sums(phyL6_transform), tax_table(phyL6_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL6 <- subset_taxa(phyL6_transform, Phylum %in% names(top10phy.names.L6))
#Saving names and proportions as a data frame then saving as csv
topphylaL6 <- as.data.frame(top10phy.names.L6)
colnames(topphylaL6)[1] ="Abundance"
write.csv(topphylaL6, "Top10Phyla_L006_Y1.csv")

## L007
asvdat <- L7
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL7 <- phyloseq(ASV,TAX,META)
phyL7_transform <- transform(phyL7, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L7 <- sort(tapply(taxa_sums(phyL7_transform), tax_table(phyL7_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL7 <- subset_taxa(phyL7_transform, Phylum %in% names(top10phy.names.L7))
#Saving names and proportions as a data frame then saving as csv
topphylaL7 <- as.data.frame(top10phy.names.L7)
colnames(topphylaL7)[1] ="Abundance"
write.csv(topphylaL7, "Top10Phyla_L007_Y1.csv")

## L008
asvdat <- L8
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL8 <- phyloseq(ASV,TAX,META)
phyL8_transform <- transform(phyL8, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L8 <- sort(tapply(taxa_sums(phyL8_transform), tax_table(phyL8_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL8 <- subset_taxa(phyL8_transform, Phylum %in% names(top10phy.names.L8))
#Saving names and proportions as a data frame then saving as csv
topphylaL8 <- as.data.frame(top10phy.names.L8)
colnames(topphylaL8)[1] ="Abundance"
write.csv(topphylaL8, "Top10Phyla_L008_Y1.csv")

## LZ25A
asvdat <- Z25A
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy25A <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phy25A_transform <- transform(phy25A, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.25A <- sort(tapply(taxa_sums(phy25A_transform), tax_table(phy25A_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla25A <- subset_taxa(phy25A_transform, Phylum %in% names(top10phy.names.25A))
#Saving names and proportions as a data frame then saving as csv
topphyla25A <- as.data.frame(top10phy.names.25A)
```

```
colnames(topphyla25A)[1] ="Abundance"
write.csv(topphyla25A, "Top10Phyla_LZ25A_Y1.csv")

## LZ2 (Firmicutes contam. removed LZ2_3_20)
asvdat <- LZ2
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyLZ2 <- phyloseq(ASV,TAX,META)
phyLZ2_transform <- transform(phyLZ2, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.LZ2 <- sort(tapply(taxa_sums(phyLZ2_transform), tax_table(phyLZ2_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaLZ2 <- subset_taxa(phyLZ2_transform, Phylum %in% names(top10phy.names.LZ2))
#Saving names and proportions as a data frame then saving as csv
topphylaLZ2 <- as.data.frame(top10phy.names.LZ2)
colnames(topphylaLZ2)[1] ="Abundance"
write.csv(topphylaLZ2, "Top10Phyla_LZ2_Y1.csv")

## LZ30
asvdat <- Z30
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy30 <- phyloseq(ASV,TAX,META)
phy30_transform <- transform(phy30, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.30 <- sort(tapply(taxa_sums(phy30_transform), tax_table(phy30_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla30 <- subset_taxa(phy30_transform, Phylum %in% names(top10phy.names.30))
#Saving names and proportions as a data frame then saving as csv
topphyla30 <- as.data.frame(top10phy.names.30)
colnames(topphyla30)[1] ="Abundance"
write.csv(topphyla30, "Top10Phyla_LZ30_Y1.csv")

## LZ40
asvdat <- Z40
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy40 <- phyloseq(ASV,TAX,META)
phy40_transform <- transform(phy40, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.40 <- sort(tapply(taxa_sums(phy40_transform), tax_table(phy40_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla40 <- subset_taxa(phy40_transform, Phylum %in% names(top10phy.names.40))
#Saving names and proportions as a data frame then saving as csv
topphyla40 <- as.data.frame(top10phy.names.40)
colnames(topphyla40)[1] ="Abundance"
write.csv(topphyla40, "Top10Phyla_LZ40_Y1.csv")

## PALMOUT (Firmicutes contam. removed PALMOUT_3_20)
asvdat <- PALM
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
```

```
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPALM <- phyloseq(ASV,TAX,META)
phyPALM_transform <- transform(phyPALM, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PALM <- sort(tapply(taxa_sums(phyPALM_transform), tax_table(phyPALM_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPALM <- subset_taxa(phyPALM_transform, Phylum %in% names(top10phy.names.PALM))
#Saving names and proportions as a data frame then saving as csv
topphylaPALM <- as.data.frame(top10phy.names.PALM)
colnames(topphylaPALM)[1] ="Abundance"
write.csv(topphylaPALM, "Top10Phyla_PALM_Y1.csv")

## PELBAY3 - DONE ON 11/12/22
asvdat <- PEL
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPEL <- phyloseq(ASV,TAX,META)
phyPEL_transform <- transform(phyPEL, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PEL <- sort(tapply(taxa_sums(phyPEL_transform), tax_table(phyPEL_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPEL <- subset_taxa(phyPEL_transform, Phylum %in% names(top10phy.names.PEL))
#Saving names and proportions as a data frame then saving as csv
topphylaPEL <- as.data.frame(top10phy.names.PEL)
colnames(topphylaPEL)[1] ="Abundance"
write.csv(topphylaPEL, "Top10Phyla_PEL_Y1.csv")

## POLE3S (Firmicutes contam. removed POLE3S_3_20)
asvdat <- POLE3S
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPOLE3S <- phyloseq(ASV,TAX,META)
phyPOLE3S_transform <- transform(phyPOLE3S, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.POLE3S <- sort(tapply(taxa_sums(phyPOLE3S_transform), tax_table(phyPOLE3S_transform)[, "Phylum"],
sum), TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPOLE3S <- subset_taxa(phyPOLE3S_transform, Phylum %in% names(top10phy.names.POLE3S))
#Saving names and proportions as a data frame then saving as csv
topphylaPOLE3S <- as.data.frame(top10phy.names.POLE3S)
colnames(topphylaPOLE3S)[1] ="Abundance"
write.csv(topphylaPOLE3S, "Top10Phyla_POLE3S_Y1.csv")

## POLESOUT (Firmicutes contam. removed POLESOUT_3_20)
asvdat <- PO
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPO <- phyloseq(ASV,TAX,META)
phyPO_transform <- transform(phyPO, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PO <- sort(tapply(taxa_sums(phyPO_transform), tax_table(phyPO_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPO <- subset_taxa(phyPO_transform, Phylum %in% names(top10phy.names.PO))
```

```
#Saving names and proportions as a data frame then saving as csv
topphylaPO <- as.data.frame(top10phy.names.PO)
colnames(topphylaPO)[1] ="Abundance"
write.csv(topphylaPO, "Top10Phyla_PO_Y1.csv")

## RITTAE2 (Firmicutes contam. removed RITTAE2_3_20)
asvdat <- RIT
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyRIT <- phyloseq(ASV,TAX,META)
phyRIT_transform <- transform(phyRIT, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.RIT <- sort(tapply(taxa_sums(phyRIT_transform), tax_table(phyRIT_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaRIT <- subset_taxa(phyRIT_transform, Phylum %in% names(top10phy.names.RIT))
#Saving names and proportions as a data frame then saving as csv
topphylaRIT <- as.data.frame(top10phy.names.RIT)
colnames(topphylaRIT)[1] ="Abundance"
write.csv(topphylaRIT, "Top10Phyla_RIT_Y1.csv")

## S308
asvdat <- S308
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS308 <- phyloseq(ASV,TAX,META)
phyS308_transform <- transform(phyS308, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S308 <- sort(tapply(taxa_sums(phyS308_transform), tax_table(phyS308_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS308 <- subset_taxa(phyS308_transform, Phylum %in% names(top10phy.names.S308))
#Saving names and proportions as a data frame then saving as csv
topphylaS308 <- as.data.frame(top10phy.names.S308)
colnames(topphylaS308)[1] ="Abundance"
write.csv(topphylaS308, "Top10Phyla_S308_Y1.csv")

## S77  (Firmicutes contam. removed S77_3_20)
asvdat <- S77
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS77 <- phyloseq(ASV,TAX,META)
phyS77_transform <- transform(phyS77, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S77 <- sort(tapply(taxa_sums(phyS77_transform), tax_table(phyS77_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS77 <- subset_taxa(phyS77_transform, Phylum %in% names(top10phy.names.S77))
#Saving names and proportions as a data frame then saving as csv
topphylaS77 <- as.data.frame(top10phy.names.S77)
colnames(topphylaS77)[1] ="Abundance"
write.csv(topphylaS77, "Top10Phyla_S77_Y1.csv")

## S79 (Firmicutes contam. removed S79_3_20)
asvdat <- S79
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
```

```
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS79 <- phyloseq(ASV,TAX,META)
phyS79_transform <- transform(phyS79, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S79 <- sort(tapply(taxa_sums(phyS79_transform), tax_table(phyS79_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS79 <- subset_taxa(phyS79_transform, Phylum %in% names(top10phy.names.S79))
#Saving names and proportions as a data frame then saving as csv
topphylaS79 <- as.data.frame(top10phy.names.S79)
colnames(topphylaS79)[1] ="Abundance"
write.csv(topphylaS79, "Top10Phyla_S79_Y1.csv")

#### Year 2
## CLV10A
asvdat <- CLV
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyCLV <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyCLV_transform <- transform(phyCLV, "compositional")
### Assigning Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.CLV <- sort(tapply(taxa_sums(phyCLV_transform), tax_table(phyCLV_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaCLV <- subset_taxa(phyCLV_transform, Phylum %in% names(top10phy.names.CLV))
#Saving names and proportions as a data frame then saving as csv
topphylaCLV <- as.data.frame(top10phy.names.CLV)
colnames(topphylaCLV)[1] ="Abundance"
write.csv(topphylaCLV, "Top10Phyla_CLV_Y2.csv")

## KISSR0.0 - (Firmicutes removed-> KISSR0.0_3_20)
asvdat <- KISS
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyKISS <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyKISS_transform <- transform(phyKISS, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.KISS <- sort(tapply(taxa_sums(phyKISS_transform), tax_table(phyKISS_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaKISS <- subset_taxa(phyKISS_transform, Phylum %in% names(top10phy.names.KISS))
#Saving names and proportions as a data frame then saving as csv
topphylaKISS <- as.data.frame(top10phy.names.KISS)
colnames(topphylaKISS)[1] ="Abundance"
write.csv(topphylaKISS, "Top10Phyla_KISS_Y2.csv")

## L001
asvdat <- L1
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL1 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL1_transform <- transform(phyL1, "compositional")
```

```
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L1 <- sort(tapply(taxa_sums(phyL1_transform), tax_table(phyL1_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL1 <- subset_taxa(phyL1_transform, Phylum %in% names(top10phy.names.L1))
#Saving names and proportions as a data frame then saving as csv
topphylaL1 <- as.data.frame(top10phy.names.L1)
colnames(topphylaL1)[1] ="Abundance"
write.csv(topphylaL1, "Top10Phyla_L001_Y2.csv")

## L004
asvdat <- L4
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL4 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL4_transform <- transform(phyL4, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L4 <- sort(tapply(taxa_sums(phyL4_transform), tax_table(phyL4_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL4 <- subset_taxa(phyL4_transform, Phylum %in% names(top10phy.names.L4))
#Saving names and proportions as a data frame then saving as csv
topphylaL4 <- as.data.frame(top10phy.names.L4)
colnames(topphylaL4)[1] ="Abundance"
write.csv(topphylaL4, "Top10Phyla_L004_Y2.csv")

## L005 (Firmicutes removed-> L005_3_20)
asvdat <- L5
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL5 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL5_transform <- transform(phyL5, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L5 <- sort(tapply(taxa_sums(phyL5_transform), tax_table(phyL5_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL5 <- subset_taxa(phyL5_transform, Phylum %in% names(top10phy.names.L5))
#Saving names and proportions as a data frame then saving as csv
topphylaL5 <- as.data.frame(top10phy.names.L5)
colnames(topphylaL5)[1] ="Abundance"
write.csv(topphylaL5, "Top10Phyla_L005_Y2.csv")

## L006
asvdat <- L6
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL6 <- phyloseq(ASV,TAX,META)
phyL6_transform <- transform(phyL6, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L6 <- sort(tapply(taxa_sums(phyL6_transform), tax_table(phyL6_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL6 <- subset_taxa(phyL6_transform, Phylum %in% names(top10phy.names.L6))
#Saving names and proportions as a data frame then saving as csv
topphylaL6 <- as.data.frame(top10phy.names.L6)
```

```
colnames(topphylaL6)[1] ="Abundance"
write.csv(topphylaL6, "Top10Phyla_L006_Y2.csv")

## L007
asvdat <- L7
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL7 <- phyloseq(ASV,TAX,META)
phyL7_transform <- transform(phyL7, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L7 <- sort(tapply(taxa_sums(phyL7_transform), tax_table(phyL7_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL7 <- subset_taxa(phyL7_transform, Phylum %in% names(top10phy.names.L7))
#Saving names and proportions as a data frame then saving as csv
topphylaL7 <- as.data.frame(top10phy.names.L7)
colnames(topphylaL7)[1] ="Abundance"
write.csv(topphylaL7, "Top10Phyla_L007_Y2.csv")

## L008
asvdat <- L8
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL8 <- phyloseq(ASV,TAX,META)
phyL8_transform <- transform(phyL8, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L8 <- sort(tapply(taxa_sums(phyL8_transform), tax_table(phyL8_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL8 <- subset_taxa(phyL8_transform, Phylum %in% names(top10phy.names.L8))
#Saving names and proportions as a data frame then saving as csv
topphylaL8 <- as.data.frame(top10phy.names.L8)
colnames(topphylaL8)[1] ="Abundance"
write.csv(topphylaL8, "Top10Phyla_L008_Y2.csv")

## LZ25A
asvdat <- Z25A
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy25A <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phy25A_transform <- transform(phy25A, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.25A <- sort(tapply(taxa_sums(phy25A_transform), tax_table(phy25A_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla25A <- subset_taxa(phy25A_transform, Phylum %in% names(top10phy.names.25A))
#Saving names and proportions as a data frame then saving as csv
topphyla25A <- as.data.frame(top10phy.names.25A)
colnames(topphyla25A)[1] ="Abundance"
write.csv(topphyla25A, "Top10Phyla_LZ25A_Y2.csv")

## LZ2 (Firmicutes contam. removed LZ2_3_20)
asvdat <- LZ2
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
```

```
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyLZ2 <- phyloseq(ASV,TAX,META)
phyLZ2_transform <- transform(phyLZ2, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.LZ2 <- sort(tapply(taxa_sums(phyLZ2_transform), tax_table(phyLZ2_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaLZ2 <- subset_taxa(phyLZ2_transform, Phylum %in% names(top10phy.names.LZ2))
#Saving names and proportions as a data frame then saving as csv
topphylaLZ2 <- as.data.frame(top10phy.names.LZ2)
colnames(topphylaLZ2)[1] ="Abundance"
write.csv(topphylaLZ2, "Top10Phyla_LZ2_Y2.csv")

## LZ30
asvdat <- Z30
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy30 <- phyloseq(ASV,TAX,META)
phy30_transform <- transform(phy30, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.30 <- sort(tapply(taxa_sums(phy30_transform), tax_table(phy30_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla30 <- subset_taxa(phy30_transform, Phylum %in% names(top10phy.names.30))
#Saving names and proportions as a data frame then saving as csv
topphyla30 <- as.data.frame(top10phy.names.30)
colnames(topphyla30)[1] ="Abundance"
write.csv(topphyla30, "Top10Phyla_LZ30_Y2.csv")

## LZ40
asvdat <- Z40
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy40 <- phyloseq(ASV,TAX,META)
phy40_transform <- transform(phy40, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.40 <- sort(tapply(taxa_sums(phy40_transform), tax_table(phy40_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla40 <- subset_taxa(phy40_transform, Phylum %in% names(top10phy.names.40))
#Saving names and proportions as a data frame then saving as csv
topphyla40 <- as.data.frame(top10phy.names.40)
colnames(topphyla40)[1] ="Abundance"
write.csv(topphyla40, "Top10Phyla_LZ40_Y2.csv")

## PALMOUT (Firmicutes contam. removed PALMOUT_3_20)
asvdat <- PALM
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPALM <- phyloseq(ASV,TAX,META)
phyPALM_transform <- transform(phyPALM, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PALM <- sort(tapply(taxa_sums(phyPALM_transform), tax_table(phyPALM_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
```

```
top10phylaPALM <- subset_taxa(phyPALM_transform, Phylum %in% names(top10phy.names.PALM))
#Saving names and proportions as a data frame then saving as csv
topphylaPALM <- as.data.frame(top10phy.names.PALM)
colnames(topphylaPALM)[1] ="Abundance"
write.csv(topphylaPALM, "Top10Phyla_PALM_Y2.csv")

## PELBAY3 - DONE ON 11/12/22
asvdat <- PEL
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPEL <- phyloseq(ASV,TAX,META)
phyPEL_transform <- transform(phyPEL, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PEL <- sort(tapply(taxa_sums(phyPEL_transform), tax_table(phyPEL_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPEL <- subset_taxa(phyPEL_transform, Phylum %in% names(top10phy.names.PEL))
#Saving names and proportions as a data frame then saving as csv
topphylaPEL <- as.data.frame(top10phy.names.PEL)
colnames(topphylaPEL)[1] ="Abundance"
write.csv(topphylaPEL, "Top10Phyla_PEL_Y2.csv")

## POLE3S (Firmicutes contam. removed POLE3S_3_20)
asvdat <- POLE3S
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPOLE3S <- phyloseq(ASV,TAX,META)
phyPOLE3S_transform <- transform(phyPOLE3S, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.POLE3S <- sort(tapply(taxa_sums(phyPOLE3S_transform), tax_table(phyPOLE3S_transform)[, "Phylum"],
sum), TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPOLE3S <- subset_taxa(phyPOLE3S_transform, Phylum %in% names(top10phy.names.POLE3S))
#Saving names and proportions as a data frame then saving as csv
topphylaPOLE3S <- as.data.frame(top10phy.names.POLE3S)
colnames(topphylaPOLE3S)[1] ="Abundance"
write.csv(topphylaPOLE3S, "Top10Phyla_POLE3S_Y2.csv")

## POLESOUT (Firmicutes contam. removed POLESOUT_3_20)
asvdat <- PO
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPO <- phyloseq(ASV,TAX,META)
phyPO_transform <- transform(phyPO, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PO <- sort(tapply(taxa_sums(phyPO_transform), tax_table(phyPO_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPO <- subset_taxa(phyPO_transform, Phylum %in% names(top10phy.names.PO))
#Saving names and proportions as a data frame then saving as csv
topphylaPO <- as.data.frame(top10phy.names.PO)
colnames(topphylaPO)[1] ="Abundance"
write.csv(topphylaPO, "Top10Phyla_PO_Y2.csv")

## RITTAE2 (Firmicutes contam. removed RITTAE2_3_20)
asvdat <- RIT
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
```

```
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyRIT <- phyloseq(ASV,TAX,META)
phyRIT_transform <- transform(phyRIT, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.RIT <- sort(tapply(taxa_sums(phyRIT_transform), tax_table(phyRIT_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaRIT <- subset_taxa(phyRIT_transform, Phylum %in% names(top10phy.names.RIT))
#Saving names and proportions as a data frame then saving as csv
topphylaRIT <- as.data.frame(top10phy.names.RIT)
colnames(topphylaRIT)[1] ="Abundance"
write.csv(topphylaRIT, "Top10Phyla_RIT_Y2.csv")

## S308
asvdat <- S308
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS308 <- phyloseq(ASV,TAX,META)
phyS308_transform <- transform(phyS308, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S308 <- sort(tapply(taxa_sums(phyS308_transform), tax_table(phyS308_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS308 <- subset_taxa(phyS308_transform, Phylum %in% names(top10phy.names.S308))
#Saving names and proportions as a data frame then saving as csv
topphylaS308 <- as.data.frame(top10phy.names.S308)
colnames(topphylaS308)[1] ="Abundance"
write.csv(topphylaS308, "Top10Phyla_S308_Y2.csv")

## S77  (Firmicutes contam. removed S77_3_20)
asvdat <- S77
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS77 <- phyloseq(ASV,TAX,META)
phyS77_transform <- transform(phyS77, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S77 <- sort(tapply(taxa_sums(phyS77_transform), tax_table(phyS77_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS77 <- subset_taxa(phyS77_transform, Phylum %in% names(top10phy.names.S77))
#Saving names and proportions as a data frame then saving as csv
topphylaS77 <- as.data.frame(top10phy.names.S77)
colnames(topphylaS77)[1] ="Abundance"
write.csv(topphylaS77, "Top10Phyla_S77_Y2.csv")

## S79 (Firmicutes contam. removed S79_3_20)
asvdat <- S79
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS79 <- phyloseq(ASV,TAX,META)
phyS79_transform <- transform(phyS79, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
```

```
top10phy.names.S79 <- sort(tapply(taxa_sums(phyS79_transform), tax_table(phyS79_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS79 <- subset_taxa(phyS79_transform, Phylum %in% names(top10phy.names.S79))
#Saving names and proportions as a data frame then saving as csv
topphylaS79 <- as.data.frame(top10phy.names.S79)
colnames(topphylaS79)[1] ="Abundance"
write.csv(topphylaS79, "Top10Phyla_S79_Y2.csv")

#### Year 3
## CLV10A
asvdat <- CLV
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyCLV <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyCLV_transform <- transform(phyCLV, "compositional")
### Assigning Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.CLV <- sort(tapply(taxa_sums(phyCLV_transform), tax_table(phyCLV_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaCLV <- subset_taxa(phyCLV_transform, Phylum %in% names(top10phy.names.CLV))
#Saving names and proportions as a data frame then saving as csv
topphylaCLV <- as.data.frame(top10phy.names.CLV)
colnames(topphylaCLV)[1] ="Abundance"
write.csv(topphylaCLV, "Top10Phyla_CLV_Y3.csv")

## KISSR0.0 - (Firmicutes removed-> KISSR0.0_3_20)
asvdat <- KISS
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyKISS <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyKISS_transform <- transform(phyKISS, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.KISS <- sort(tapply(taxa_sums(phyKISS_transform), tax_table(phyKISS_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaKISS <- subset_taxa(phyKISS_transform, Phylum %in% names(top10phy.names.KISS))
#Saving names and proportions as a data frame then saving as csv
topphylaKISS <- as.data.frame(top10phy.names.KISS)
colnames(topphylaKISS)[1] ="Abundance"
write.csv(topphylaKISS, "Top10Phyla_KISS_Y3.csv")

## L001
asvdat <- L1
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL1 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL1_transform <- transform(phyL1, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L1 <- sort(tapply(taxa_sums(phyL1_transform), tax_table(phyL1_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL1 <- subset_taxa(phyL1_transform, Phylum %in% names(top10phy.names.L1))
#Saving names and proportions as a data frame then saving as csv
topphylaL1 <- as.data.frame(top10phy.names.L1)
```

```
colnames(topphylaL1)[1] ="Abundance"
write.csv(topphylaL1, "Top10Phyla_L001_Y3.csv")

## L004
asvdat <- L4
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL4 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL4_transform <- transform(phyL4, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L4 <- sort(tapply(taxa_sums(phyL4_transform), tax_table(phyL4_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL4 <- subset_taxa(phyL4_transform, Phylum %in% names(top10phy.names.L4))
#Saving names and proportions as a data frame then saving as csv
topphylaL4 <- as.data.frame(top10phy.names.L4)
colnames(topphylaL4)[1] ="Abundance"
write.csv(topphylaL4, "Top10Phyla_L004_Y3.csv")

## L005 (Firmicutes removed-> L005_3_20)
asvdat <- L5
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL5 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phyL5_transform <- transform(phyL5, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L5 <- sort(tapply(taxa_sums(phyL5_transform), tax_table(phyL5_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL5 <- subset_taxa(phyL5_transform, Phylum %in% names(top10phy.names.L5))
#Saving names and proportions as a data frame then saving as csv
topphylaL5 <- as.data.frame(top10phy.names.L5)
colnames(topphylaL5)[1] ="Abundance"
write.csv(topphylaL5, "Top10Phyla_L005_Y3.csv")

## L006
asvdat <- L6
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL6 <- phyloseq(ASV,TAX,META)
phyL6_transform <- transform(phyL6, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L6 <- sort(tapply(taxa_sums(phyL6_transform), tax_table(phyL6_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL6 <- subset_taxa(phyL6_transform, Phylum %in% names(top10phy.names.L6))
#Saving names and proportions as a data frame then saving as csv
topphylaL6 <- as.data.frame(top10phy.names.L6)
colnames(topphylaL6)[1] ="Abundance"
write.csv(topphylaL6, "Top10Phyla_L006_Y3.csv")

## L007
asvdat <- L7
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
```

```
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL7 <- phyloseq(ASV,TAX,META)
phyL7_transform <- transform(phyL7, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L7 <- sort(tapply(taxa_sums(phyL7_transform), tax_table(phyL7_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL7 <- subset_taxa(phyL7_transform, Phylum %in% names(top10phy.names.L7))
#Saving names and proportions as a data frame then saving as csv
topphylaL7 <- as.data.frame(top10phy.names.L7)
colnames(topphylaL7)[1] ="Abundance"
write.csv(topphylaL7, "Top10Phyla_L007_Y3.csv")

## L008
asvdat <- L8
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyL8 <- phyloseq(ASV,TAX,META)
phyL8_transform <- transform(phyL8, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.L8 <- sort(tapply(taxa_sums(phyL8_transform), tax_table(phyL8_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaL8 <- subset_taxa(phyL8_transform, Phylum %in% names(top10phy.names.L8))
#Saving names and proportions as a data frame then saving as csv
topphylaL8 <- as.data.frame(top10phy.names.L8)
colnames(topphylaL8)[1] ="Abundance"
write.csv(topphylaL8, "Top10Phyla_L008_Y3.csv")

## LZ25A
asvdat <- Z25A
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy25A <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
phy25A_transform <- transform(phy25A, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.25A <- sort(tapply(taxa_sums(phy25A_transform), tax_table(phy25A_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla25A <- subset_taxa(phy25A_transform, Phylum %in% names(top10phy.names.25A))
#Saving names and proportions as a data frame then saving as csv
topphyla25A <- as.data.frame(top10phy.names.25A)
colnames(topphyla25A)[1] ="Abundance"
write.csv(topphyla25A, "Top10Phyla_LZ25A_Y3.csv")

## LZ2 (Firmicutes contam. removed LZ2_3_20)
asvdat <- LZ2
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyLZ2 <- phyloseq(ASV,TAX,META)
phyLZ2_transform <- transform(phyLZ2, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
```

```
top10phy.names.LZ2 <- sort(tapply(taxa_sums(phyLZ2_transform), tax_table(phyLZ2_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaLZ2 <- subset_taxa(phyLZ2_transform, Phylum %in% names(top10phy.names.LZ2))
#Saving names and proportions as a data frame then saving as csv
topphylaLZ2 <- as.data.frame(top10phy.names.LZ2)
colnames(topphylaLZ2)[1] ="Abundance"
write.csv(topphylaLZ2, "Top10Phyla_LZ2_Y3.csv")

## LZ30
asvdat <- Z30
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy30 <- phyloseq(ASV,TAX,META)
phy30_transform <- transform(phy30, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.30 <- sort(tapply(taxa_sums(phy30_transform), tax_table(phy30_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla30 <- subset_taxa(phy30_transform, Phylum %in% names(top10phy.names.30))
#Saving names and proportions as a data frame then saving as csv
topphyla30 <- as.data.frame(top10phy.names.30)
colnames(topphyla30)[1] ="Abundance"
write.csv(topphyla30, "Top10Phyla_LZ30_Y3.csv")

## LZ40
asvdat <- Z40
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phy40 <- phyloseq(ASV,TAX,META)
phy40_transform <- transform(phy40, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.40 <- sort(tapply(taxa_sums(phy40_transform), tax_table(phy40_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phyla40 <- subset_taxa(phy40_transform, Phylum %in% names(top10phy.names.40))
#Saving names and proportions as a data frame then saving as csv
topphyla40 <- as.data.frame(top10phy.names.40)
colnames(topphyla40)[1] ="Abundance"
write.csv(topphyla40, "Top10Phyla_LZ40_Y3.csv")

## PALMOUT (Firmicutes contam. removed PALMOUT_3_20)
asvdat <- PALM
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPALM <- phyloseq(ASV,TAX,META)
phyPALM_transform <- transform(phyPALM, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PALM <- sort(tapply(taxa_sums(phyPALM_transform), tax_table(phyPALM_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPALM <- subset_taxa(phyPALM_transform, Phylum %in% names(top10phy.names.PALM))
#Saving names and proportions as a data frame then saving as csv
topphylaPALM <- as.data.frame(top10phy.names.PALM)
colnames(topphylaPALM)[1] ="Abundance"
write.csv(topphylaPALM, "Top10Phyla_PALM_Y3.csv")

## PELBAY3 - DONE ON 11/12/22
```

```
asvdat <- PEL
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPEL <- phyloseq(ASV,TAX,META)
phyPEL_transform <- transform(phyPEL, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PEL <- sort(tapply(taxa_sums(phyPEL_transform), tax_table(phyPEL_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPEL <- subset_taxa(phyPEL_transform, Phylum %in% names(top10phy.names.PEL))
#Saving names and proportions as a data frame then saving as csv
topphylaPEL <- as.data.frame(top10phy.names.PEL)
colnames(topphylaPEL)[1] ="Abundance"
write.csv(topphylaPEL, "Top10Phyla_PEL_Y3.csv")

## POLE3S (Firmicutes contam. removed POLE3S_3_20)
asvdat <- POLE3S
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPOLE3S <- phyloseq(ASV,TAX,META)
phyPOLE3S_transform <- transform(phyPOLE3S, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.POLE3S <- sort(tapply(taxa_sums(phyPOLE3S_transform), tax_table(phyPOLE3S_transform)[, "Phylum"],
sum), TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPOLE3S <- subset_taxa(phyPOLE3S_transform, Phylum %in% names(top10phy.names.POLE3S))
#Saving names and proportions as a data frame then saving as csv
topphylaPOLE3S <- as.data.frame(top10phy.names.POLE3S)
colnames(topphylaPOLE3S)[1] ="Abundance"
write.csv(topphylaPOLE3S, "Top10Phyla_POLE3S_Y3.csv")

## POLESOUT (Firmicutes contam. removed POLESOUT_3_20)
asvdat <- PO
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyPO <- phyloseq(ASV,TAX,META)
phyPO_transform <- transform(phyPO, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.PO <- sort(tapply(taxa_sums(phyPO_transform), tax_table(phyPO_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaPO <- subset_taxa(phyPO_transform, Phylum %in% names(top10phy.names.PO))
#Saving names and proportions as a data frame then saving as csv
topphylaPO <- as.data.frame(top10phy.names.PO)
colnames(topphylaPO)[1] ="Abundance"
write.csv(topphylaPO, "Top10Phyla_PO_Y3.csv")

## RITTAE2 (Firmicutes contam. removed RITTAE2_3_20)
asvdat <- RIT
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyRIT <- phyloseq(ASV,TAX,META)
phyRIT_transform <- transform(phyRIT, "compositional")
```

```
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.RIT <- sort(tapply(taxa_sums(phyRIT_transform), tax_table(phyRIT_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaRIT <- subset_taxa(phyRIT_transform, Phylum %in% names(top10phy.names.RIT))
#Saving names and proportions as a data frame then saving as csv
topphylaRIT <- as.data.frame(top10phy.names.RIT)
colnames(topphylaRIT)[1] ="Abundance"
write.csv(topphylaRIT, "Top10Phyla_RIT_Y3.csv")

## S308
asvdat <- S308
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS308 <- phyloseq(ASV,TAX,META)
phyS308_transform <- transform(phyS308, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S308 <- sort(tapply(taxa_sums(phyS308_transform), tax_table(phyS308_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS308 <- subset_taxa(phyS308_transform, Phylum %in% names(top10phy.names.S308))
#Saving names and proportions as a data frame then saving as csv
topphylaS308 <- as.data.frame(top10phy.names.S308)
colnames(topphylaS308)[1] ="Abundance"
write.csv(topphylaS308, "Top10Phyla_S308_Y3.csv")

## S77  (Firmicutes contam. removed S77_3_20)
asvdat <- S77
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS77 <- phyloseq(ASV,TAX,META)
phyS77_transform <- transform(phyS77, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S77 <- sort(tapply(taxa_sums(phyS77_transform), tax_table(phyS77_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS77 <- subset_taxa(phyS77_transform, Phylum %in% names(top10phy.names.S77))
#Saving names and proportions as a data frame then saving as csv
topphylaS77 <- as.data.frame(top10phy.names.S77)
colnames(topphylaS77)[1] ="Abundance"
write.csv(topphylaS77, "Top10Phyla_S77_Y3.csv")

## S79 (Firmicutes contam. removed S79_3_20)
asvdat <- S79
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
phyS79 <- phyloseq(ASV,TAX,META)
phyS79_transform <- transform(phyS79, "compositional")
## Top 10 Phyla
#Sort Phylum by abundance and pick the top 10
top10phy.names.S79 <- sort(tapply(taxa_sums(phyS79_transform), tax_table(phyS79_transform)[, "Phylum"], sum),
TRUE)[1:10]
#Cut down the physeq data to only the top 10 Phyla
top10phylaS79 <- subset_taxa(phyS79_transform, Phylum %in% names(top10phy.names.S79))
#Saving names and proportions as a data frame then saving as csv
topphylaS79 <- as.data.frame(top10phy.names.S79)
colnames(topphylaS79)[1] ="Abundance"
write.csv(topphylaS79, "Top10Phyla_S79_Y3.csv")
```

```
###### Top 10 Phyla by Station - ALL 3 YEARS ######
### You need tidyverse package in order to do this

## Loading in each station on their own (make sure the two columns in the csv is labeled 'Phylum' 'Station
Name')
CLV <- read.csv("Top10Phyla_CLV.csv")
KISS <- read.csv("Top10Phyla_KISS.csv")
L1 <- read.csv("Top10Phyla_L001.csv")
L4 <- read.csv("Top10Phyla_L004.csv")
L5 <- read.csv("Top10Phyla_L005.csv")
L6 <- read.csv("Top10Phyla_L006.csv")
L7 <- read.csv("Top10Phyla_L007.csv")
L8 <- read.csv("Top10Phyla_L008.csv")
LZ2 <- read.csv("Top10Phyla_LZ2.csv")
Z25A <- read.csv("Top10Phyla_LZ25A.csv")
Z30 <- read.csv("Top10Phyla_LZ30.csv")
Z40 <- read.csv("Top10Phyla_LZ40.csv")
PALM <- read.csv("Top10Phyla_PALM.csv")
PEL <- read.csv("Top10Phyla_PEL.csv")
POLE3S <- read.csv("Top10Phyla_POLE3S.csv")
PO <- read.csv("Top10Phyla_PO.csv")
RIT <- read.csv("Top10Phyla_RIT.csv")
S308 <- read.csv("Top10Phyla_S308.csv")
S77 <- read.csv("Top10Phyla_S77.csv")
S79 <- read.csv("Top10Phyla_S79.csv")
## Creating a list of the stations
Stations <- list(CLV, KISS, L1, L4, L5, L6, L7, L8, LZ2, Z25A, Z30, Z40, PALM,
                 PEL, POLE3S, PO, RIT, S308, S77, S79)
## Merging all of the data frames in the list (USES TIDYVERSE)
Station_merge <- Stations %>% reduce(full_join, by= "Phylum")
Station_merge[is.na(Station_merge)] = 0  #replacing the NAs with zeros
## Saving merged data frame as CSV
write.csv(Station_merge, "Top10Phyla-Stations.csv")


## Testing to see if I can create a stacked bar chart using the merged station data frame
## Converting the data frame into long format (which converts it into a tibble)
S_tibble <-Station_merge %>% pivot_longer(cols=c(2:21),names_to= "Station",values_to= "Abundance")
write.csv(S_tibble, "StationPhyla_long.csv")
# ## Reloading in previous data frame (went into excel and replaced NA with 0)
# StationPhyla <- read.csv("StationPhyla_long.csv", header = T) or SKIP AND GO TO NEXT LINE!!
StationPhyla <- S_tibble

## Plotting using custom colors
Top10Station <- ggplot(StationPhyla, aes(fill=Phylum, x=Abundance, y=Station)) +
  geom_bar(position='fill', stat='identity')+      #position="fill" creates a stacked bar plot with abundance as
a percentage
  theme_minimal()+
  labs(x='Abundance', y='Stations', title='Top Phyla Found in Lake Okeechobee by Station')+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
Top10Stat <- c("#2bcaf4","#24630e","#000080","#edc427","#1f60aa","#333333","#d841ad",
                        "#41ea27","red3","#806bb4","#cbcc8f","#5f421b","#f08539","#ff9eed")
                        ## listed by phyla in alphabetical order
withr::with_options(list(ggplot2.discrete.fill = Top10Stat),print(Top10Station))

###### Top 10 Phyla by Station - EACH YEAR ######
### You need tidyverse package in order to do this

##Year 1
CLV <- read.csv("Top10Phyla_CLV_Y1.csv")
KISS <- read.csv("Top10Phyla_KISS_Y1.csv")
L1 <- read.csv("Top10Phyla_L001_Y1.csv")
L4 <- read.csv("Top10Phyla_L004_Y1.csv")
L5 <- read.csv("Top10Phyla_L005_Y1.csv")
L6 <- read.csv("Top10Phyla_L006_Y1.csv")
L7 <- read.csv("Top10Phyla_L007_Y1.csv")
L8 <- read.csv("Top10Phyla_L008_Y1.csv")
LZ2 <- read.csv("Top10Phyla_LZ2_Y1.csv")
Z25A <- read.csv("Top10Phyla_LZ25A_Y1.csv")
Z30 <- read.csv("Top10Phyla_LZ30_Y1.csv")
Z40 <- read.csv("Top10Phyla_LZ40_Y1.csv")
PALM <- read.csv("Top10Phyla_PALM_Y1.csv")
PEL <- read.csv("Top10Phyla_PEL_Y1.csv")
POLE3S <- read.csv("Top10Phyla_POLE3S_Y1.csv")
```

```
PO <- read.csv("Top10Phyla_PO_Y1.csv")
RIT <- read.csv("Top10Phyla_RIT_Y1.csv")
S308 <- read.csv("Top10Phyla_S308_Y1.csv")
S77 <- read.csv("Top10Phyla_S77_Y1.csv")
S79 <- read.csv("Top10Phyla_S79_Y1.csv")
## Creating a list of the stations
Stations <- list(CLV, KISS, L1, L4, L5, L6, L7, L8, LZ2, Z25A, Z30, Z40, PALM,
                 PEL, POLE3S, PO, RIT, S308, S77, S79)
## Merging all of the data frames in the list (USES TIDYVERSE)
Station_merge <- Stations %>% reduce(full_join, by= "Phylum")
Station_merge[is.na(Station_merge)] = 0  #replacing the NAs with zeros
## Saving merged data frame as CSV
write.csv(Station_merge, "Top10Phyla-Stations_Y1.csv")


## Testing to see if I can create a stacked bar chart using the merged station data frame
## Converting the data frame into long format (which converts it into a tibble)
S_tibble <-Station_merge %>% pivot_longer(cols=c(2:21),names_to= "Station",values_to= "Abundance")
write.csv(S_tibble, "StationPhyla_long_Y1.csv")
# StationPhyla <- read.csv("StationPhyla_long_Y1.csv", header = T) or SKIP AND GO TO NEXT LINE!!
StationPhyla <- S_tibble

## Plotting using custom colors
Top10Station <- ggplot(StationPhyla, aes(fill=Phylum, x=Abundance, y=Station)) +
  geom_bar(position='fill', stat='identity')+      #position="fill" creates a stacked bar plot with abundance as
a percentage
  theme_minimal()+
  labs(x='Abundance', y='Stations', title='Top Phyla Found in Lake Okeechobee by Station - Year 1')+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
Top10Stat <- c("#2bcaf4","#24630e","#edc427","#1f60aa","#333333","#d841ad",
                        "#41ea27","red3","#806bb4","#cbcc8f","#5f421b","#f08539","purple4","#ff9eed")
                        ## listed by phyla in alphabetical order
withr::with_options(list(ggplot2.discrete.fill = Top10Stat),print(Top10Station))

##Year 2
CLV <- read.csv("Top10Phyla_CLV_Y2.csv")
KISS <- read.csv("Top10Phyla_KISS_Y2.csv")
L1 <- read.csv("Top10Phyla_L001_Y2.csv")
L4 <- read.csv("Top10Phyla_L004_Y2.csv")
L5 <- read.csv("Top10Phyla_L005_Y2.csv")
L6 <- read.csv("Top10Phyla_L006_Y2.csv")
L7 <- read.csv("Top10Phyla_L007_Y2.csv")
L8 <- read.csv("Top10Phyla_L008_Y2.csv")
LZ2 <- read.csv("Top10Phyla_LZ2_Y2.csv")
Z25A <- read.csv("Top10Phyla_LZ25A_Y2.csv")
Z30 <- read.csv("Top10Phyla_LZ30_Y2.csv")
Z40 <- read.csv("Top10Phyla_LZ40_Y2.csv")
PALM <- read.csv("Top10Phyla_PALM_Y2.csv")
PEL <- read.csv("Top10Phyla_PEL_Y2.csv")
POLE3S <- read.csv("Top10Phyla_POLE3S_Y2.csv")
PO <- read.csv("Top10Phyla_PO_Y2.csv")
RIT <- read.csv("Top10Phyla_RIT_Y2.csv")
S308 <- read.csv("Top10Phyla_S308_Y2.csv")
S77 <- read.csv("Top10Phyla_S77_Y2.csv")
S79 <- read.csv("Top10Phyla_S79_Y2.csv")
## Creating a list of the stations
Stations <- list(CLV, KISS, L1, L4, L5, L6, L7, L8, LZ2, Z25A, Z30, Z40, PALM,
                 PEL, POLE3S, PO, RIT, S308, S77, S79)
## Merging all of the data frames in the list (USES TIDYVERSE)
Station_merge <- Stations %>% reduce(full_join, by= "Phylum")
Station_merge[is.na(Station_merge)] = 0  #replacing the NAs with zeros
## Saving merged data frame as CSV
write.csv(Station_merge, "Top10Phyla-Stations_Y2.csv")


## Testing to see if I can create a stacked bar chart using the merged station data frame
## Converting the data frame into long format (which converts it into a tibble)
S_tibble <-Station_merge %>% pivot_longer(cols=c(2:21),names_to= "Station",values_to= "Abundance")
write.csv(S_tibble, "StationPhyla_long_Y2.csv")
# StationPhyla <- read.csv("StationPhyla_long_Y2.csv", header = T) or SKIP AND GO TO NEXT LINE!!
StationPhyla <- S_tibble

## Plotting using custom colors
Top10Station <- ggplot(StationPhyla, aes(fill=Phylum, x=Abundance, y=Station)) +
```

```
  geom_bar(position='fill', stat='identity')+      #position="fill" creates a stacked bar plot with abundance as
a percentage
  theme_minimal()+
  labs(x='Abundance', y='Stations', title='Top Phyla Found in Lake Okeechobee by Station - Year 2')+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
Top10Stat <- c("#2bcaf4","#24630e","#000080","#edc427","#1f60aa","#333333","#d841ad",
                        "#41ea27","red3","#806bb4","#5f421b","#f08539","#ff9eed")
                        ## listed by phyla in alphabetical order
withr::with_options(list(ggplot2.discrete.fill = Top10Stat),print(Top10Station))

##Year 3
CLV <- read.csv("Top10Phyla_CLV_Y3.csv")
KISS <- read.csv("Top10Phyla_KISS_Y3.csv")
L1 <- read.csv("Top10Phyla_L001_Y3.csv")
L4 <- read.csv("Top10Phyla_L004_Y3.csv")
L5 <- read.csv("Top10Phyla_L005_Y3.csv")
L6 <- read.csv("Top10Phyla_L006_Y3.csv")
L7 <- read.csv("Top10Phyla_L007_Y3.csv")
L8 <- read.csv("Top10Phyla_L008_Y3.csv")
LZ2 <- read.csv("Top10Phyla_LZ2_Y3.csv")
Z25A <- read.csv("Top10Phyla_LZ25A_Y3.csv")
Z30 <- read.csv("Top10Phyla_LZ30_Y3.csv")
Z40 <- read.csv("Top10Phyla_LZ40_Y3.csv")
PALM <- read.csv("Top10Phyla_PALM_Y3.csv")
PEL <- read.csv("Top10Phyla_PEL_Y3.csv")
POLE3S <- read.csv("Top10Phyla_POLE3S_Y3.csv")
PO <- read.csv("Top10Phyla_PO_Y3.csv")
RIT <- read.csv("Top10Phyla_RIT_Y3.csv")
S308 <- read.csv("Top10Phyla_S308_Y3.csv")
S77 <- read.csv("Top10Phyla_S77_Y3.csv")
S79 <- read.csv("Top10Phyla_S79_Y3.csv")
## Creating a list of the stations
Stations <- list(CLV, KISS, L1, L4, L5, L6, L7, L8, LZ2, Z25A, Z30, Z40, PALM,
                 PEL, POLE3S, PO, RIT, S308, S77, S79)
## Merging all of the data frames in the list (USES TIDYVERSE)
Station_merge <- Stations %>% reduce(full_join, by= "Phylum")
Station_merge[is.na(Station_merge)] = 0  #replacing the NAs with zeros
## Saving merged data frame as CSV
write.csv(Station_merge, "Top10Phyla-Stations_Y3.csv")


## Testing to see if I can create a stacked bar chart using the merged station data frame
## Converting the data frame into long format (which converts it into a tibble)
S_tibble <-Station_merge %>% pivot_longer(cols=c(2:21),names_to= "Station",values_to= "Abundance")
write.csv(S_tibble, "StationPhyla_long_Y3.csv")
# StationPhyla <- read.csv("StationPhyla_long_Y3.csv", header = T) or SKIP AND GO TO NEXT LINE!!
StationPhyla <- S_tibble

## Plotting using custom colors
Top10Station <- ggplot(StationPhyla, aes(fill=Phylum, x=Abundance, y=Station)) +
  geom_bar(position='fill', stat='identity')+      #position="fill" creates a stacked bar plot with abundance as
a percentage
  theme_minimal()+
  labs(x='Abundance', y='Stations', title='Top Phyla Found in Lake Okeechobee by Station - Year 3')+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
Top10Stat <- c("#2bcaf4","#24630e","#edc427","#1f60aa","#333333","#d841ad",
                        "#41ea27","red3","#806bb4","#cbcc8f","#a97548","#5f421b","#f08539","#ff9eed")
                        ## listed by phyla in alphabetical order
withr::with_options(list(ggplot2.discrete.fill = Top10Stat),print(Top10Station))


###### Top 15 Orders in Year 3 by Station ######
##Merge feature table with taxonomy and save
dat.Y3 <- read.csv("feature_Y3r_ADJUSTED.csv")
tax <- read.csv("taxonomy_Y123_edited&cleaned.csv")
Yr3t <- merge.data.frame(dat.Y3,tax,by= "FeatureID", all.x = TRUE)
write.csv(Yr3t, "feature_Y3r_ADJUSTED_tax.csv")

##Load feature/tax table
dat.Y3 <- as.data.frame(t(read.csv("feature_Y3r_ADJUSTED_tax.csv", row.names = 1)))

##Separate Station and create master list of top 15
CLV <- as.data.frame(t(dat.Y3[grep("^CLV10A", rownames(dat.Y3)),]))
KISS <- as.data.frame(t(dat.Y3[grep("^KISSR0.0", rownames(dat.Y3)),]))
```

```
L1 <- as.data.frame(t(dat.Y3[grep("^L001", rownames(dat.Y3)),]))
L4 <- as.data.frame(t(dat.Y3[grep("^L004", rownames(dat.Y3)),]))
L5 <- as.data.frame(t(dat.Y3[grep("^L005", rownames(dat.Y3)),]))
L6 <- as.data.frame(t(dat.Y3[grep("^L006", rownames(dat.Y3)),]))
L7 <- as.data.frame(t(dat.Y3[grep("^L007", rownames(dat.Y3)),]))
L8 <- as.data.frame(t(dat.Y3[grep("^L008", rownames(dat.Y3)),]))
LZ2 <- as.data.frame(t(dat.Y3[grep("^LZ2_", rownames(dat.Y3)),]))
Z25A <- as.data.frame(t(dat.Y3[grep("^LZ25A", rownames(dat.Y3)),]))
Z30 <- as.data.frame(t(dat.Y3[grep("^LZ30", rownames(dat.Y3)),]))
Z40 <- as.data.frame(t(dat.Y3[grep("^LZ40", rownames(dat.Y3)),]))
PALM <- as.data.frame(t(dat.Y3[grep("^PALMOUT", rownames(dat.Y3)),]))
PEL <- as.data.frame(t(dat.Y3[grep("^PELBAY3", rownames(dat.Y3)),]))
POLE3S <- as.data.frame(t(dat.Y3[grep("^POLE3S", rownames(dat.Y3)),]))
PO <- as.data.frame(t(dat.Y3[grep("^POLESOUT", rownames(dat.Y3)),]))
RIT <- as.data.frame(t(dat.Y3[grep("^RITTAE2", rownames(dat.Y3)),]))
S308 <- as.data.frame(t(dat.Y3[grep("^S308", rownames(dat.Y3)),]))
S77 <- as.data.frame(t(dat.Y3[grep("^S77", rownames(dat.Y3)),]))
S79 <- as.data.frame(t(dat.Y3[grep("^S79", rownames(dat.Y3)),]))

##Assigning top 15 orders by Station
## CLV10A
asvdat <- CLV
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordCLV <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
ordCLV_transform <- transform(ordCLV, "compositional")
### Assigning Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.CLV <- sort(tapply(taxa_sums(ordCLV_transform), tax_table(ordCLV_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordCLV <- subset_taxa(ordCLV_transform, Order %in% names(top15ord.names.CLV))
#Saving names and proportions as a data frame then saving as csv
topordCLV <- as.data.frame(top15ord.names.CLV)
colnames(topordCLV)[1] ="Abundance"
write.csv(topordCLV, "Top15Ord_CLV.csv")


## KISSR0.0 - (Firmicutes removed-> KISSR0.0_3_20)
asvdat <- KISS
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordKISS <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
ordKISS_transform <- transform(ordKISS, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.KISS <- sort(tapply(taxa_sums(ordKISS_transform), tax_table(ordKISS_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordKISS <- subset_taxa(ordKISS_transform, Order %in% names(top15ord.names.KISS))
#Saving names and proportions as a data frame then saving as csv
topordKISS <- as.data.frame(top15ord.names.KISS)
colnames(topordKISS)[1] ="Abundance"
write.csv(topordKISS, "Top15Ord_KISS.csv")


## L001
asvdat <- L1
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
```

```
META <- sample_data(meta)
ordL1 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
ordL1_transform <- transform(ordL1, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.L1 <- sort(tapply(taxa_sums(ordL1_transform), tax_table(ordL1_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordL1 <- subset_taxa(ordL1_transform, Order %in% names(top15ord.names.L1))
#Saving names and proportions as a data frame then saving as csv
topordL1 <- as.data.frame(top15ord.names.L1)
colnames(topordL1)[1] ="Abundance"
write.csv(topordL1, "Top15Ord_L001.csv")


## L004
asvdat <- L4
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordL4 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
ordL4_transform <- transform(ordL4, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.L4 <- sort(tapply(taxa_sums(ordL4_transform), tax_table(ordL4_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordL4 <- subset_taxa(ordL4_transform, Order %in% names(top15ord.names.L4))
#Saving names and proportions as a data frame then saving as csv
topordL4 <- as.data.frame(top15ord.names.L4)
colnames(topordL4)[1] ="Abundance"
write.csv(topordL4, "Top15Ord_L004.csv")


## L005 (Firmicutes removed-> L005_3_20)
asvdat <- L5
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordL5 <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
ordL5_transform <- transform(ordL5, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.L5 <- sort(tapply(taxa_sums(ordL5_transform), tax_table(ordL5_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordL5 <- subset_taxa(ordL5_transform, Order %in% names(top15ord.names.L5))
#Saving names and proportions as a data frame then saving as csv
topordL5 <- as.data.frame(top15ord.names.L5)
colnames(topordL5)[1] ="Abundance"
write.csv(topordL5, "Top15Ord_L005.csv")


## L006
asvdat <- L6
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordL6 <- phyloseq(ASV,TAX,META)
ordL6_transform <- transform(ordL6, "compositional")
## Top 15 ord
```

```
#Sort Order by abundance and pick the top 15
top15ord.names.L6 <- sort(tapply(taxa_sums(ordL6_transform), tax_table(ordL6_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordL6 <- subset_taxa(ordL6_transform, Order %in% names(top15ord.names.L6))
#Saving names and proportions as a data frame then saving as csv
topordL6 <- as.data.frame(top15ord.names.L6)
colnames(topordL6)[1] ="Abundance"
write.csv(topordL6, "Top15Ord_L006.csv")


## L007
asvdat <- L7
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordL7 <- phyloseq(ASV,TAX,META)
ordL7_transform <- transform(ordL7, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.L7 <- sort(tapply(taxa_sums(ordL7_transform), tax_table(ordL7_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordL7 <- subset_taxa(ordL7_transform, Order %in% names(top15ord.names.L7))
#Saving names and proportions as a data frame then saving as csv
topordL7 <- as.data.frame(top15ord.names.L7)
colnames(topordL7)[1] ="Abundance"
write.csv(topordL7, "Top15Ord_L007.csv")

## L008
asvdat <- L8
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordL8 <- phyloseq(ASV,TAX,META)
ordL8_transform <- transform(ordL8, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.L8 <- sort(tapply(taxa_sums(ordL8_transform), tax_table(ordL8_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordL8 <- subset_taxa(ordL8_transform, Order %in% names(top15ord.names.L8))
#Saving names and proportions as a data frame then saving as csv
topordL8 <- as.data.frame(top15ord.names.L8)
colnames(topordL8)[1] ="Abundance"
write.csv(topordL8, "Top15Ord_L008.csv")

## LZ25A
asvdat <- Z25A
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ord25A <- phyloseq(ASV,TAX,META)
transform <- microbiome::transform
ord25A_transform <- transform(ord25A, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.25A <- sort(tapply(taxa_sums(ord25A_transform), tax_table(ord25A_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ord25A <- subset_taxa(ord25A_transform, Order %in% names(top15ord.names.25A))
#Saving names and proportions as a data frame then saving as csv
topord25A <- as.data.frame(top15ord.names.25A)
colnames(topord25A)[1] ="Abundance"
```

```
write.csv(topord25A, "Top15Ord_LZ25A.csv")

## LZ2 (Firmicutes contam. removed LZ2_3_20)
asvdat <- LZ2
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordLZ2 <- phyloseq(ASV,TAX,META)
ordLZ2_transform <- transform(ordLZ2, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.LZ2 <- sort(tapply(taxa_sums(ordLZ2_transform), tax_table(ordLZ2_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordLZ2 <- subset_taxa(ordLZ2_transform, Order %in% names(top15ord.names.LZ2))
#Saving names and proportions as a data frame then saving as csv
topordLZ2 <- as.data.frame(top15ord.names.LZ2)
colnames(topordLZ2)[1] ="Abundance"
write.csv(topordLZ2, "Top15Ord_LZ2.csv")

## LZ30
asvdat <- Z30
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ord30 <- phyloseq(ASV,TAX,META)
ord30_transform <- transform(ord30, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.30 <- sort(tapply(taxa_sums(ord30_transform), tax_table(ord30_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ord30 <- subset_taxa(ord30_transform, Order %in% names(top15ord.names.30))
#Saving names and proportions as a data frame then saving as csv
topord30 <- as.data.frame(top15ord.names.30)
colnames(topord30)[1] ="Abundance"
write.csv(topord30, "Top15Ord_LZ30.csv")

## LZ40
asvdat <- Z40
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ord40 <- phyloseq(ASV,TAX,META)
ord40_transform <- transform(ord40, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.40 <- sort(tapply(taxa_sums(ord40_transform), tax_table(ord40_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ord40 <- subset_taxa(ord40_transform, Order %in% names(top15ord.names.40))
#Saving names and proportions as a data frame then saving as csv
topord40 <- as.data.frame(top15ord.names.40)
colnames(topord40)[1] ="Abundance"
write.csv(topord40, "Top15Ord_LZ40.csv")

## PALMOUT (Firmicutes contam. removed PALMOUT_3_20)
asvdat <- PALM
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
```

```
META <- sample_data(meta)
ordPALM <- phyloseq(ASV,TAX,META)
ordPALM_transform <- transform(ordPALM, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.PALM <- sort(tapply(taxa_sums(ordPALM_transform), tax_table(ordPALM_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordPALM <- subset_taxa(ordPALM_transform, Order %in% names(top15ord.names.PALM))
#Saving names and proportions as a data frame then saving as csv
topordPALM <- as.data.frame(top15ord.names.PALM)
colnames(topordPALM)[1] ="Abundance"
write.csv(topordPALM, "Top15Ord_PALM.csv")

## PELBAY3 - DONE ON 11/12/22
asvdat <- PEL
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordPEL <- phyloseq(ASV,TAX,META)
ordPEL_transform <- transform(ordPEL, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.PEL <- sort(tapply(taxa_sums(ordPEL_transform), tax_table(ordPEL_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordPEL <- subset_taxa(ordPEL_transform, Order %in% names(top15ord.names.PEL))
#Saving names and proportions as a data frame then saving as csv
topordPEL <- as.data.frame(top15ord.names.PEL)
colnames(topordPEL)[1] ="Abundance"
write.csv(topordPEL, "Top15Ord_PEL.csv")

## POLE3S - DONE ON 11/12/22 (Firmicutes contam. removed POLE3S_3_20)
asvdat <- POLE3S
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordPOLE3S <- phyloseq(ASV,TAX,META)
ordPOLE3S_transform <- transform(ordPOLE3S, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.POLE3S <- sort(tapply(taxa_sums(ordPOLE3S_transform), tax_table(ordPOLE3S_transform)[, "Order"],
sum), TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordPOLE3S <- subset_taxa(ordPOLE3S_transform, Order %in% names(top15ord.names.POLE3S))
#Saving names and proportions as a data frame then saving as csv
topordPOLE3S <- as.data.frame(top15ord.names.POLE3S)
colnames(topordPOLE3S)[1] ="Abundance"
write.csv(topordPOLE3S, "Top15Ord_POLE3S.csv")

## POLESOUT - DONE ON 11/12/22 (Firmicutes contam. removed POLESOUT_3_20)
asvdat <- PO
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordPO <- phyloseq(ASV,TAX,META)
ordPO_transform <- transform(ordPO, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.PO <- sort(tapply(taxa_sums(ordPO_transform), tax_table(ordPO_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordPO <- subset_taxa(ordPO_transform, Order %in% names(top15ord.names.PO))
#Saving names and proportions as a data frame then saving as csv
```

```
topordPO <- as.data.frame(top15ord.names.PO)
colnames(topordPO)[1] ="Abundance"
write.csv(topordPO, "Top15Ord_PO.csv")

## RITTAE2 - DONE ON 11/12/22 (Firmicutes contam. removed RITTAE2_3_20)
asvdat <- RIT
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordRIT <- phyloseq(ASV,TAX,META)
ordRIT_transform <- transform(ordRIT, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.RIT <- sort(tapply(taxa_sums(ordRIT_transform), tax_table(ordRIT_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordRIT <- subset_taxa(ordRIT_transform, Order %in% names(top15ord.names.RIT))
#Saving names and proportions as a data frame then saving as csv
topordRIT <- as.data.frame(top15ord.names.RIT)
colnames(topordRIT)[1] ="Abundance"
write.csv(topordRIT, "Top15Ord_RIT.csv")

## S308
asvdat <- S308
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordS308 <- phyloseq(ASV,TAX,META)
ordS308_transform <- transform(ordS308, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.S308 <- sort(tapply(taxa_sums(ordS308_transform), tax_table(ordS308_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordS308 <- subset_taxa(ordS308_transform, Order %in% names(top15ord.names.S308))
#Saving names and proportions as a data frame then saving as csv
topordS308 <- as.data.frame(top15ord.names.S308)
colnames(topordS308)[1] ="Abundance"
write.csv(topordS308, "Top15Ord_S308.csv")

## S77  (Firmicutes contam. removed S77_3_20)
asvdat <- S77
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordS77 <- phyloseq(ASV,TAX,META)
ordS77_transform <- transform(ordS77, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.S77 <- sort(tapply(taxa_sums(ordS77_transform), tax_table(ordS77_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordS77 <- subset_taxa(ordS77_transform, Order %in% names(top15ord.names.S77))
#Saving names and proportions as a data frame then saving as csv
topordS77 <- as.data.frame(top15ord.names.S77)
colnames(topordS77)[1] ="Abundance"
write.csv(topordS77, "Top15Ord_S77.csv")

## S79 (Firmicutes contam. removed S79_3_20)
asvdat <- S79
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
```

```
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
ordS79 <- phyloseq(ASV,TAX,META)
ordS79_transform <- transform(ordS79, "compositional")
## Top 15 ord
#Sort Order by abundance and pick the top 15
top15ord.names.S79 <- sort(tapply(taxa_sums(ordS79_transform), tax_table(ordS79_transform)[, "Order"], sum),
TRUE)[1:15]
#Cut down the phyloseq data to only the top 15 ord
top15ordS79 <- subset_taxa(ordS79_transform, Order %in% names(top15ord.names.S79))
#Saving names and proportions as a data frame then saving as csv
topordS79 <- as.data.frame(top15ord.names.S79)
colnames(topordS79)[1] ="Abundance"
write.csv(topordS79, "Top15Ord_S79.csv")


## Creating a list of the stations
CLV <- read.csv("Top15Ord_CLV.csv")
KISS <- read.csv("Top15Ord_KISS.csv")
L1 <- read.csv("Top15Ord_L001.csv")
L4 <- read.csv("Top15Ord_L004.csv")
L5 <- read.csv("Top15Ord_L005.csv")
L6 <- read.csv("Top15Ord_L006.csv")
L7 <- read.csv("Top15Ord_L007.csv")
L8 <- read.csv("Top15Ord_L008.csv")
LZ2 <- read.csv("Top15Ord_LZ2.csv")
Z25A <- read.csv("Top15Ord_LZ25A.csv")
Z30 <- read.csv("Top15Ord_LZ30.csv")
Z40 <- read.csv("Top15Ord_LZ40.csv")
PALM <- read.csv("Top15Ord_PALM.csv")
PEL <- read.csv("Top15Ord_PEL.csv")
POLE3S <- read.csv("Top15Ord_POLE3S.csv")
PO <- read.csv("Top15Ord_PO.csv")
RIT <- read.csv("Top15Ord_RIT.csv")
S308 <- read.csv("Top15Ord_S308.csv")
S77 <- read.csv("Top15Ord_S77.csv")
S79 <- read.csv("Top15Ord_S79.csv")
## Creating a list of the stations (fix in Excel before moving on!)
Stations <- list(CLV, KISS, L1, L4, L5, L6, L7, L8, LZ2, Z25A, Z30, Z40, PALM,
                 PEL, POLE3S, PO, RIT, S308, S77, S79)
## Merging all of the data frames in the list (USES TIDYVERSE)
Station_merge <- Stations %>% reduce(full_join, by= "Order")
Station_merge[is.na(Station_merge)] = 0  #replacing the NAs with zeros
Station_merge[5,1] <- "NA" #renaming a cell in the dataframe
## Saving merged data frame as CSV
write.csv(Station_merge, "Top15Order-Stations_Y3.csv")


## Testing to see if I can create a stacked bar chart using the merged station data frame
## Converting the data frame into long format (which converts it into a tibble)
S_tibble <-Station_merge %>% pivot_longer(cols=c(2:21),names_to= "Station",values_to= "Abundance")
write.csv(S_tibble, "StationOrders_long_Y3.csv")
# StationOrd <- read.csv("StationPhyla_long_Y3.csv", header = T) or SKIP AND GO TO NEXT LINE!!
StationOrd <- S_tibble

## Plotting using custom colors
Top15Station <- ggplot(StationOrd, aes(fill=Order, x=Abundance, y=Station)) +
  geom_bar(position='fill', stat='identity')+      #position="fill" creates a stacked bar plot with abundance as
a percentage
  theme_minimal()+
  labs(x='Abundance', y='Stations', title='Top Orders Found in Lake Okeechobee by Station - Year 3')+
  theme(plot.title = element_text(color="navyblue", size=14, face="bold.italic", hjust = 0.5))+
  theme(legend.title = element_text(face="italic"))
Top15Stat <- c("#000000","#004949","#009292","#ff6db6","#ffb6db",
                               "#78C675","#006ddb","#b66dff","#6db6ff","#b6dbff",
                               "#920000","#924900","#db6d00","navy","#ffff6d",
                          "antiquewhite2", "#1D91C0", "#67005F", "khaki3", "#CB181D",
                          "#A6D854", "#F46D43", "#A6CEE3", "#FD8D3C", "#490092", "#999999")
## 15-color palette, colorlblind friendly
withr::with_options(list(ggplot2.discrete.fill = Top15Stat),print(Top15Station))


###### Environmental variable - Scatter plots by Year ######
library(ggplot2)
library(cowplot)
```

```
#Loading in metadata
metadata <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
#Subsetting metadata table by year
met1 <- metadata[grep("_19$", rownames(metadata)),]
met2 <- metadata[grep("_20$", rownames(metadata)),]
met3 <- metadata[grep("_21$", rownames(metadata)),]
write.csv(met1, "Metadata_BATCH_Y1.csv")
write.csv(met2, "Metadata_BATCH_Y2.csv")
write.csv(met3, "Metadata_BATCH_Y3.csv")

### PLOTTING
#Chlorophyll a
ch1 <- ggplot(met1, aes(x = as.factor(Month), y = Chlorophyll.a)) +
  geom_jitter(size = 2, color = "green4", width = 0.25)+
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "Chlorophyll a (ug/L)")+
  ylim(-25, 150)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

ch2 <- ggplot(met2, aes(x = as.factor(Month), y = Chlorophyll.a)) +
  geom_jitter(size = 2, color = "green4", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-25, 150)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

ch3 <- ggplot(met3, aes(x = as.factor(Month), y = Chlorophyll.a)) +
  geom_jitter(size = 2, color = "green4", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-25, 150)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph and saving as png
png(file="Chla_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(ch1, ch2, ch3, ncol = 3, labels = "AUTO")
graphics.off()


#Total Phosphorus
tp1 <- ggplot(met1, aes(x = as.factor(Month), y = Phosphate.Total)) +
  geom_jitter(size = 2, color = "darkred", width = 0.25)+
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "Total Phosphorus (mg/L)")+
  ylim(0, 0.5)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

tp2 <- ggplot(met2, aes(x = as.factor(Month), y = Phosphate.Total)) +
```

```
  geom_jitter(size = 2, color = "darkred", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 0.5)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

tp3 <- ggplot(met3, aes(x = as.factor(Month), y = Phosphate.Total)) +
  geom_jitter(size = 2, color = "darkred", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 0.5)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph and saving png
png(file="TP_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(tp1, tp2, tp3, ncol = 3, labels = "AUTO")
graphics.off()

#Nitrate + Nitrite
tn1 <- ggplot(met1, aes(x = as.factor(Month), y = Nitrate.Nitrite)) +
  geom_jitter(size = 2, color = "dodgerblue2", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "Nitrate + Nitrite (mg/L)")+
  ylim(-0.2, 0.6)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

tn2 <- ggplot(met2, aes(x = as.factor(Month), y = Nitrate.Nitrite)) +
  geom_jitter(size = 2, color = "dodgerblue2", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-0.2, 0.6)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

tn3 <- ggplot(met3, aes(x = as.factor(Month), y = Nitrate.Nitrite)) +
  geom_jitter(size = 2, color = "dodgerblue2", width = 0.25)+
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-0.2, 0.6)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph and saving png
```

```
png(file="Nit_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(tn1, tn2, tn3, ncol = 3, labels = "AUTO")
graphics.off()

#Ammonia
a1 <- ggplot(met1, aes(x = as.factor(Month), y = Ammonia)) +
  geom_jitter(size = 2, color = "mediumpurple3", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "Ammonia (mg/L)")+
  ylim(-0.2, 0.8)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

a2 <- ggplot(met2, aes(x = as.factor(Month), y = Ammonia)) +
  geom_jitter(size = 2, color = "mediumpurple3", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-0.2, 0.8)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

a3 <- ggplot(met3, aes(x = as.factor(Month), y = Ammonia)) +
  geom_jitter(size = 2, color = "mediumpurple3", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-0.2, 0.8)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph and saving png
png(file="Ammonia_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(a1, a2, a3, ncol = 3, labels = "AUTO")
graphics.off()

#Temperature
t1 <- ggplot(met1, aes(x = as.factor(Month), y = Temperature)) +
  geom_jitter(size = 2, color = "sienna", width = 0.25)+
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "Temperature (°C)")+
  ylim(0, 35)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

t2 <- ggplot(met2, aes(x = as.factor(Month), y = Temperature)) +
  geom_jitter(size = 2, color = "sienna", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 35)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
```

```
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

t3 <- ggplot(met3, aes(x = as.factor(Month), y = Temperature)) +
  geom_jitter(size = 2, color = "sienna", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 35)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph
png(file="Temp_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(t1, t2, t3, ncol = 3, labels = "AUTO")
graphics.off()

#Microcystin.LR
m1 <- ggplot(met1, aes(x = as.factor(Month), y = Microcystin.LR)) +
  geom_jitter(size = 2, color = "hotpink3", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "Microcystin (ug/L)")+
  ylim(-10, 55)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

m2 <- ggplot(met2, aes(x = as.factor(Month), y = Microcystin.LR)) +
  geom_jitter(size = 2, color = "hotpink3", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-10, 55)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

m3 <- ggplot(met3, aes(x = as.factor(Month), y = Microcystin.LR)) +
  geom_jitter(size = 2, color = "hotpink3", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-10, 55)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph and saving png
png(file="MicrocystinLR_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(m1, m2, m3, ncol = 3, labels = "AUTO")
graphics.off()

#pH
p1 <- ggplot(met1, aes(x = as.factor(Month), y = pH)) +
  geom_jitter(size = 2, color = "darkorange", width = 0.25)+
  stat_summary(fun=mean, aes(group=1), geom="line",
```

```
              colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "pH")+
  ylim(0, 11)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

p2 <- ggplot(met2, aes(x = as.factor(Month), y = pH)) +
  geom_jitter(size = 2, color = "darkorange", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
              colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 11)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

p3 <- ggplot(met3, aes(x = as.factor(Month), y = pH)) +
  geom_jitter(size = 2, color = "darkorange", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
              colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 11)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph
png(file="PH_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(p1, p2, p3, ncol = 3, labels = "AUTO")
graphics.off()

#Total Nitrogen
tn4 <- ggplot(met1, aes(x = as.factor(Month), y = Total.Nitrogen)) +
  geom_jitter(size = 2, color = "navy", width = 0.25)+
  stat_summary(fun=mean, aes(group=1), geom="line",
              colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "Total Nitrogen (mg/L)")+
  ylim(0, 4)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

tn5 <- ggplot(met2, aes(x = as.factor(Month), y = Total.Nitrogen)) +
  geom_jitter(size = 2, color = "navy", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
              colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 4)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

tn6 <- ggplot(met3, aes(x = as.factor(Month), y = Total.Nitrogen)) +
  geom_jitter(size = 2, color = "navy", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
```

```
              colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 4)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph
png(file="TotN_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(tn4, tn5, tn6, ncol = 3, labels = "AUTO")
graphics.off()

#TN:TP
np1 <- ggplot(met1, aes(x = as.factor(Month), y = TN.TP.ratio)) +
  geom_jitter(size = 2, color = "lightsalmon2", width = 0.25)+
  stat_summary(fun=mean, aes(group=1), geom="line",
              colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "TN : TP")+
  ylim(0, 46)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

np2 <- ggplot(met2, aes(x = as.factor(Month), y = TN.TP.ratio)) +
  geom_jitter(size = 2, color = "lightsalmon2", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
              colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 46)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

np3 <- ggplot(met3, aes(x = as.factor(Month), y = TN.TP.ratio)) +
  geom_jitter(size = 2, color = "lightsalmon2", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
              colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 46)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph
png(file="TNTP_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(np1, np2, np3, ncol = 3, labels = "AUTO")
graphics.off()

#Total Depth
d1 <- ggplot(met1, aes(x = as.factor(Month), y = TotalDepth)) +
  geom_jitter(size = 2, color = "cornsilk4", width = 0.25)+
  stat_summary(fun=mean, aes(group=1), geom="line",
              colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "Total Depth (m)")+
  ylim(0, 6)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
```

```
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

d2 <- ggplot(met2, aes(x = as.factor(Month), y = TotalDepth)) +
  geom_jitter(size = 2, color = "cornsilk4", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 6)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

d3 <- ggplot(met3, aes(x = as.factor(Month), y = TotalDepth)) +
  geom_jitter(size = 2, color = "cornsilk4", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(0, 6)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph and saving as png
png(file="Depth_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(d1, d2, d3, ncol = 3, labels = "AUTO")
graphics.off()

#Total Phosphate
tph1 <- ggplot(met1, aes(x = as.factor(Month), y = Phosphate.Ortho)) +
  geom_jitter(size = 2, color = "grey35", width = 0.25)+
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "Total Phosphate (mg/L)")+
  ylim(-0.01, 0.25)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

tph2 <- ggplot(met2, aes(x = as.factor(Month), y = Phosphate.Ortho)) +
  geom_jitter(size = 2, color = "grey35", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-0.01, 0.25)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

tph3 <- ggplot(met3, aes(x = as.factor(Month), y = Phosphate.Ortho)) +
  geom_jitter(size = 2, color = "grey35", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-0.01, 0.25)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
```

```
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph and saving png
png(file="TPhos_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(tph1, tph2, tph3, ncol = 3, labels = "AUTO")
graphics.off()

###### Viewing Microcystis RA over time ######
#Loading in metadata
metadata <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
#Subsetting metadata table by year
met1 <- metadata[grep("_19$", rownames(metadata)),]
met2 <- metadata[grep("_20$", rownames(metadata)),]
met3 <- metadata[grep("_21$", rownames(metadata)),]

#Plotting
mc1 <- ggplot(met1, aes(x = as.factor(Month), y = Microcystis.Abundance)) +
  geom_jitter(size = 1.8, color = "darkcyan", width = 0.25)+
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = "Microcystis Relative Abundance")+
  ylim(-0.01, 0.1)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 1 - 2019") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

mc2 <- ggplot(met2, aes(x = as.factor(Month), y = Microcystis.Abundance)) +
  geom_jitter(size = 1.8, color = "darkcyan", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-0.01, 0.1)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 2 - 2020") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

mc3 <- ggplot(met3, aes(x = as.factor(Month), y = Microcystis.Abundance)) +
  geom_jitter(size = 1.8, color = "darkcyan", width = 0.25) +
  stat_summary(fun=mean, aes(group=1), geom="line",
               colour="black", linewidth= 0.7)+
  theme_grey()+
  labs(x = "Month", y = NULL)+
  ylim(-0.01, 0.1)+
  theme(legend.position="none")+
  theme(axis.title = element_text(size = 15,face = "bold"))+
  theme(axis.text = element_text(size = 14))+
  labs(title = "Year 3 - 2021") +
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

#Viewing all plots in one graph
png(file="Microcystis_scatter.png", width=1406, height=573, bg="transparent")
plot_grid(mc1, mc2, mc3, ncol = 3, labels = "AUTO")
graphics.off()

###### Alpha Diversity - Measures ######
#### alpha diversity: the species richness that occurs within a given area within a region
#### that is smaller than the entire distribution of the species (Moore, 2013)
#### uses the relative abundance data

###Diversity by Sample (MAKE SURE YOU ONLY HAVE vegan INSTALLED!!)
# Species richness:
S <- as.data.frame(specnumber(dat.01per))
colnames(S)[1] ="Species Richness"
## Species richness: the number of species within a region (Moore, 2013)
#No. individuals:
N <- as.data.frame(rowSums(dat.01per))
```

```
colnames(N)[1] ="No. of Individuals"
#Shannon-Weiner Diversity:
H <- as.data.frame(diversity(dat.ra), index="shannon")
colnames(H)[1] ="Shannon Diverisity Index"
## Shannon index: a measure of the information content of a community rather than of the particular species
##                that is present (Moore, 2013) [species richness index]
## strongly influenced by species richness and by rare species (so sample size is negligible)
#Pielou's Evenness:
J = H/log(S)
colnames(J)[1] ="Species Evenness"
## Pielou's evenness: an index that measures diversity along with the species richness
## Formula - J = H/log(S) (aka Shannon evenness index)
## evenness = the count of individuals of each species in an area; 0 is no evenness & 1 is complete evenness
#Simpson's Diversity (1/D) (inverse):
inv.D <- as.data.frame(diversity(dat.ra, index="inv"))
colnames(inv.D)[1] ="inverse Simpson Diversity Index"
## gives the Simpson index the property of increasing as diversity increases (the dominance of
## a few species decreases)

#Combine data together into a single new data frame, export as CSV
diversitybysample <- cbind(S, N, H, J,inv.D)
write.csv(diversitybysample, "AlphaDiversityBATCH.csv")

#merging with metadata table and export as csv (edited OUTSIDE of R in Excel)
diversitybysample <- read.csv("AlphaDiversityBATCH.csv", row.names = 1)
met <- read.csv("Metadata-Diversity_BATCH.csv", row.names = 1)
adivmet <- cbind(diversitybysample,met)
write.csv(adivmet,"Metadata-Diversity_BATCH.csv")

###### Alpha Diversity Stats. - ALL YEARS TOGETHER ######
# Packages Used
library(vegan)
library(stats)
library(ggplot2)
library(ggfortify)

#### Alpha Diversities analyses
metadata <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)

#### Testing Statistical Significance
## Normality - Shapiro Test (only done on NUMERIC data)
## p <= 0.05 = H0 REJECTED -> DATA IS NOT NORMAL
## p > 0.05 = H0 ACCEPTED -> DATA IS NORMAL
## Attempted to transform twice using log and sqrt

#Alpha Diversity Variables
shapiro.test(metadata$S) #NOT NORMAL
#W = 0.97921, p-value = 5.777e-07
shapiro.test(metadata$N) #NOT NORMAL
#W = 0.91551, p-value < 2.2e-16
shapiro.test(metadata$H) #NOT NORMAL
#W = 0.96059, p-value = 7.456e-11
shapiro.test(metadata$J) #NOT NORMAL
#W = 0.72606, p-value < 2.2e-16
shapiro.test(metadata$inv.D) #NOT NORMAL
#W = 0.9247, p-value = 8.049e-16



## NOT NORMAL -> Transformations also didn't work -> Non-parametric test (KRUSKAL-WALLIS)
library(pgirmess)
library(multcompView)

#### Hypothesis 1 Comparisons (Diversity & Year)
# Kruskal Wallis: Nonparametric Data (not normal)
## Pairwise Wilcox Test - calculate pairwise comparisons between group levels
##                        with corrections for multiple testing (non-parametric)

kruskal.test(metadata$S ~ metadata$Year)
#Kruskal-Wallis chi-squared = 13.385, df = 2, p-value = 0.00124 (< 0.05; reject null - significant)
pairwise.wilcox.test(metadata$S, metadata$Year, p.adjust.method = "fdr") #Difference between year 1 and 3 & year
2 and 3
kmc <- kruskalmc(metadata$S ~ metadata$Year) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
```

```
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#  1   2   3
# "a" "a" "b"

kruskal.test(metadata$N ~ metadata$Year)
#Kruskal-Wallis chi-squared = 19.73, df = 2, p-value = 5.196e-05 (< 0.05; reject null - significant)
pairwise.wilcox.test(metadata$N, metadata$Year, p.adjust.method = "fdr") #Difference between year 1 and 3 & year
2 and 3
kmc <- kruskalmc(metadata$N ~ metadata$Year) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#  1   2   3
# "a" "a" "b"


kruskal.test(metadata$H ~ metadata$Year)
#Kruskal-Wallis chi-squared = 8.5305, df = 2, p-value = 0.01405 (< 0.05; reject null - significant)
pairwise.wilcox.test(metadata$H, metadata$Year, p.adjust.method = "fdr") #Difference between year 2 and 3
kmc <- kruskalmc(metadata$H ~ metadata$Year) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#  1   2   3
# "ab" "a" "b"

kruskal.test(metadata$J ~ metadata$Year)
#Kruskal-Wallis chi-squared = 16.987, df = 2, p-value = 0.0002048 (< 0.05; reject null - significant)
pairwise.wilcox.test(metadata$J, metadata$Year, p.adjust.method = "fdr") #Difference between year 1 and 2 & 1
and 3
kmc <- kruskalmc(metadata$J ~ metadata$Year) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#  1   2   3
# "a" "ab" "b"

kruskal.test(metadata$inv.D ~ metadata$Year)
#Kruskal-Wallis chi-squared = 16.987, df = 2, p-value = 0.0002048 (< 0.05; reinv.Dect null - significant)
pairwise.wilcox.test(metadata$inv.D, metadata$Year, p.adjust.method = "fdr") #Difference between year 1 and 2 &
1 and 3
kmc <- kruskalmc(metadata$inv.D ~ metadata$Year) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#  1   2   3
# "a" "b" "a"


## Plotting boxplots of alpha diversity by year
# Creating pdf for the plots to populate
pdf("AlphaDiverisityPlots.pdf")
```

```
par(mfrow=c(2,2))
par(mar=c(5,6,2,2)+0.1)
# plot each boxplot on its own page
boxplot(S~Year, data=metadata, horizontal = F, las=1, ylab = "", xlab = "")
title(xlab="Year", line = 3, cex.lab=1.15)
title(ylab="Species Richness (S)", line=4.25, cex.lab=1.15)
text(y=1500, x=3, labels="b", col="blue", cex=1.2)
text(y=1420, x=2, labels="a", col="red", cex=1.2)        # labeling which groups are significantly different
than the other
text(y=1585, x=1, labels="a", col="red", cex=1.2)

par(mar=c(5,4.5,2,2)+0.1)
boxplot(H~Year, data=metadata, horizontal = F, las=1, ylab = "", xlab = "")
title(xlab="Year", line = 3, cex.lab=1.15)
title(ylab="Shannon Diversity Index (H)", line=2.8, cex.lab=1.15)
text(y=4, x=3, labels="b", col="blue", cex=1.2)
text(y=3.4, x=2, labels="a", col="red", cex=1.2)
text(y=3.6, x=1, labels="ab", col="purple", cex=1.2)

boxplot(J~Year, data=metadata, horizontal = F, las=1, ylab = "", xlab = "")
title(xlab="Year", line = 3, cex.lab=1.15)
title(ylab="Species Evenness (J)", line=3, cex.lab=1.15)
text(y=0.73, x=3, labels="b", col="blue", cex=1.2)
text(y=0.685, x=2, labels="ab", col="purple", cex=1.2)
text(y=0.73, x=1, labels="a", col="red", cex=1.2)

par(mar=c(5,6,2,2)+0.1)
boxplot(inv.D~Year, data=metadata, horizontal = F, las=1, ylab = "", xlab = "")
title(xlab="Year", line = 3, cex.lab=1.15)
title(ylab="inverse Simpson Diversity Index (inv.D)", line=3.6, cex.lab=1.15)
text(y=440, x=3, labels="a", col="red", cex=1.2)
text(y=420, x=2, labels="b", col="blue", cex=1.2)
text(y=340, x=1, labels="a", col="red", cex=1.2)

boxplot(N~Year, data=metadata, horizontal = F, las=1, ylab = "", xlab = "")
title(xlab="Year", line = 3, cex.lab=1.15)
title(ylab="No. of Individuals (N)", line=4.25, cex.lab=1.15)
text(y=90000, x=3, labels="b", col="blue", cex=1.2)
text(y=130000, x=2, labels="a", col="red", cex=1.2)
text(y=160000, x=1, labels="a", col="red", cex=1.2)

# stop saving to pdf
dev.off()

###### Alpha Diversity by Year ######
Y1 <- metadata[grep("_19$", rownames(metadata)),]
Y2 <- metadata[grep("_20$", rownames(metadata)),]
Y3 <- metadata[grep("_21$", rownames(metadata)),]
## Packages
library(pgirmess)
library(multcompView)
library(vegan)

###### Differences by ZONE - Richness, Shannon, inv. Simpson, Evenness ####
# Boxplot colors by zone (4 different zones so 4 different colors)
Zones <- c("palegreen3","wheat2","rosybrown1","violetred2")


#Year 1
kruskal.test(Y1$S ~ Y1$Zone)
#Kruskal-Wallis chi-squared = 12.026, df = 3, p-value = 0.007295
pairwise.wilcox.test(Y1$S, Y1$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$S ~ Y1$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# Inflow  Nearshore   Pelagic     S79
#  "ab"       "a"       "b"      "ab"

kruskal.test(Y1$H ~ Y1$Zone)
#Kruskal-Wallis chi-squared = 11.77, df = 3, p-value = 0.008214
```

```
pairwise.wilcox.test(Y1$H, Y1$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$H ~ Y1$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# Inflow Nearshore   Pelagic       S79
# "ab"       "a"       "b"      "ab"

kruskal.test(Y1$inv.D ~ Y1$Zone)
#Kruskal-Wallis chi-squared = 8.5961, df = 3, p-value = 0.03517
pairwise.wilcox.test(Y1$inv.D, Y1$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$inv.D ~ Y1$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# Inflow Nearshore   Pelagic   S79
# "a"       "a"       "a"      "a"  -> NO DIFFERENCES

kruskal.test(Y1$J ~ Y1$Zone)
#Kruskal-Wallis chi-squared = 13.726, df = 3, p-value = 0.003303
pairwise.wilcox.test(Y1$J, Y1$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$J ~ Y1$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# Inflow Nearshore   Pelagic    S79
# "a"       "b"      "ab"      "ab"

## Plotting all the Year 1 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Zone, data=Y1, las=1, col= Zones, ylab = "Species Richness")
boxplot(H~Zone, data=Y1, las=1,col= Zones, ylab = "Shannon Diversity Index")
boxplot(inv.D~Zone, data=Y1, las=1,col= Zones, ylab = "inverse Simpson Diversity Index")
boxplot(J~Zone, data=Y1, las=1,col= Zones, ylab = "Evenness")
#Creating main title
mtext("Alpha Diversity by Zone - Year 1", side = 3, line = - 2.4, outer = TRUE, cex = 1.4)


#Year 2 - NO SIGINIFICANT DIFFERENCES!
kruskal.test(Y2$S ~ Y2$Zone)
#Kruskal-Wallis chi-squared = 2.1354, df = 3, p-value = 0.5448
pairwise.wilcox.test(Y2$S, Y2$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$S ~ Y2$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGNIFICANT DIFFERENCES FOUND!!

kruskal.test(Y2$H ~ Y2$Zone)
#Kruskal-Wallis chi-squared = 0.90469, df = 3, p-value = 0.8243
pairwise.wilcox.test(Y2$H, Y2$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$H ~ Y2$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
```

```
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGNIFICANT DIFFERENCES FOUND!!

kruskal.test(Y2$inv.D ~ Y2$Zone)
#Kruskal-Wallis chi-squared = 2.1509, df = 3, p-value = 0.5417
pairwise.wilcox.test(Y2$inv.D, Y2$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$inv.D ~ Y2$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGNIFICANT DIFFERENCES FOUND!!

kruskal.test(Y2$J ~ Y2$Zone)
#Kruskal-Wallis chi-squared = 6.2334, df = 3, p-value = 0.1008
pairwise.wilcox.test(Y2$J, Y2$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$J ~ Y2$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGNIFICANT DIFFERENCES FOUND!!

## Plotting all the Year 2 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Zone, data=Y2, las=1, col= Zones, ylab = "Species Richness")
boxplot(H~Zone, data=Y2, las=1,col= Zones, ylab = "Shannon Diversity Index")
boxplot(inv.D~Zone, data=Y2, las=1,col= Zones, ylab = "inverse Simpson Diversity Index")
boxplot(J~Zone, data=Y2, las=1,col= Zones, ylab = "Evenness")
#Creating main title
mtext("Alpha Diversity by Zone - Year 2", side = 3, line = - 2.4, outer = TRUE, cex = 1.4)


#Year 3
kruskal.test(Y3$S ~ Y3$Zone)
#Kruskal-Wallis chi-squared = 18.21, df = 3, p-value = 0.0003981
pairwise.wilcox.test(Y3$S, Y3$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$S ~ Y3$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# Inflow Nearshore   Pelagic    S79
# "a"       "b"       "a"       "b"

kruskal.test(Y3$H ~ Y3$Zone)
#Kruskal-Wallis chi-squared = 14.781, df = 3, p-value = 0.002014
pairwise.wilcox.test(Y3$H, Y3$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$H ~ Y3$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# Inflow Nearshore   Pelagic    S79
#  "a"       "b"      "ab"      "ab"
```

```
kruskal.test(Y3$inv.D ~ Y3$Zone)
#Kruskal-Wallis chi-squared = 13.68, df = 3, p-value = 0.003374
pairwise.wilcox.test(Y3$inv.D, Y3$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$inv.D ~ Y3$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#Inflow Nearshore   Pelagic    S79
# "a"       "b"       "b"      "ab"

kruskal.test(Y3$J ~ Y3$Zone)
#Kruskal-Wallis chi-squared = 15.472, df = 3, p-value = 0.001454
pairwise.wilcox.test(Y3$J, Y3$Zone, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$J ~ Y3$Zone) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#Inflow Nearshore   Pelagic   S79
#"a"       "b"       "b"      "ab"

## Plotting all the Year 3 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Zone, data=Y3, las=1, col= Zones, ylab = "Species Richness")
boxplot(H~Zone, data=Y3, las=1,col= Zones, ylab = "Shannon Diversity Index")
boxplot(inv.D~Zone, data=Y3, las=1,col= Zones, ylab = "inverse Simpson Diversity Index")
boxplot(J~Zone, data=Y3, las=1,col= Zones, ylab = "Evenness")
#Creating main title
mtext("Alpha Diversity by Zone - Year 3", side = 3, line = -2.4, outer = TRUE, cex = 1.4)

###### Differences by SEASON - Richness, Shannon, inv. Simpson, Evenness ####
# Boxplot colors by season (2 seasons so 2 different colors)
Seasons <- c("lemonchiffon2","royalblue1")

#Year 1 - NO SIGNIFICANT DIFFERENCES ALL AROUND!
kruskal.test(Y1$S ~ Y1$Season)
#Kruskal-Wallis chi-squared = 0.10935, df = 1, p-value = 0.7409
pairwise.wilcox.test(Y1$S, Y1$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$S ~ Y1$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGINIFICANT DIFFERENCES FOUND!!

kruskal.test(Y1$H ~ Y1$Season)
#Kruskal-Wallis chi-squared = 0.18617, df = 1, p-value = 0.6661
pairwise.wilcox.test(Y1$H, Y1$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$H ~ Y1$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGINIFICANT DIFFERENCES FOUND!!

kruskal.test(Y1$inv.D ~ Y1$Season)
#Kruskal-Wallis chi-squared = 0.16256, df = 1, p-value = 0.6868
```

```
pairwise.wilcox.test(Y1$inv.D, Y1$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$inv.D ~ Y1$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGINIFICANT DIFFERENCES FOUND!!

kruskal.test(Y1$J ~ Y1$Season)
#Kruskal-Wallis chi-squared = 1.5322, df = 1, p-value = 0.2158
pairwise.wilcox.test(Y1$J, Y1$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$J ~ Y1$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGINIFICANT DIFFERENCES FOUND!!

## Plotting all the Year 1 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Season, data=Y1, las=1, col= Seasons, ylab = "Species Richness")
boxplot(H~Season, data=Y1, las=1,col= Seasons, ylab = "Shannon Diversity Index")
boxplot(inv.D~Season, data=Y1, las=1,col= Seasons, ylab = "inverse Simpson Diversity Index")
boxplot(J~Season, data=Y1, las=1,col= Seasons, ylab = "Evenness")
#Creating main title
mtext("Alpha Diversity by Season - Year 1", side = 3, line = - 2.4, outer = TRUE, cex = 1.4)

#Year 2 - Difference found in evenness
kruskal.test(Y2$S ~ Y2$Season)
#Kruskal-Wallis chi-squared = 0.0066879, df = 1, p-value = 0.9348
pairwise.wilcox.test(Y2$S, Y2$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$S ~ Y2$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGINIFICANT DIFFERENCES FOUND!!

kruskal.test(Y2$H ~ Y2$Season)
#Kruskal-Wallis chi-squared = 0.018269, df = 1, p-value = 0.8925
pairwise.wilcox.test(Y2$H, Y2$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$H ~ Y2$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGINIFICANT DIFFERENCES FOUND!!

kruskal.test(Y2$inv.D ~ Y2$Season)
#Kruskal-Wallis chi-squared = 0.17949, df = 1, p-value = 0.6718
pairwise.wilcox.test(Y2$inv.D, Y2$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$inv.D ~ Y2$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
```

```
let # significant letters for the multiple comparison test
#NO SIGINIFICANT DIFFERENCES FOUND!!

kruskal.test(Y2$J ~ Y2$Season)
#Kruskal-Wallis chi-squared = 11.159, df = 1, p-value = 0.0008365
pairwise.wilcox.test(Y2$J, Y2$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$J ~ Y2$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# dry  wet
# "a"  "b"

## Plotting all the Year 2 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Season, data=Y2, las=1, col= Seasons, ylab = "Species Richness")
boxplot(H~Season, data=Y2, las=1,col= Seasons, ylab = "Shannon Diversity Index")
boxplot(inv.D~Season, data=Y2, las=1,col= Seasons, ylab = "inverse Simpson Diversity Index")
boxplot(J~Season, data=Y2, las=1,col= Seasons, ylab = "Evenness")
#Creating main title
mtext("Alpha Diversity by Season - Year 2", side = 3, line = - 2.4, outer = TRUE, cex = 1.4)


#Year 3 - Differences found in evenness
kruskal.test(Y3$S ~ Y3$Season)
#Kruskal-Wallis chi-squared = 2.0537, df = 1, p-value = 0.1518
pairwise.wilcox.test(Y3$S, Y3$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$S ~ Y3$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGINIFICANT DIFFERENCES FOUND!!

kruskal.test(Y3$H ~ Y3$Season)
#Kruskal-Wallis chi-squared = 0.075109, df = 1, p-value = 0.784
pairwise.wilcox.test(Y3$H, Y3$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$H ~ Y3$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGINIFICANT DIFFERENCES FOUND!!

kruskal.test(Y3$inv.D ~ Y3$Season)
#Kruskal-Wallis chi-squared = 0.41548, df = 1, p-value = 0.5192
pairwise.wilcox.test(Y3$inv.D, Y3$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$inv.D ~ Y3$Season) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGINIFICANT DIFFERENCES FOUND!!

kruskal.test(Y3$J ~ Y3$Season)
#Kruskal-Wallis chi-squared = 4.3677, df = 1, p-value = 0.03663
pairwise.wilcox.test(Y3$J, Y3$Season, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$J ~ Y3$Season) # multiple-comparison test
```

```
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# dry  wet
# "a"  "b"

## Plotting all the Year 3 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Season, data=Y3, las=1, col= Seasons, ylab = "Species Richness")
boxplot(H~Season, data=Y3, las=1,col= Seasons, ylab = "Shannon Diversity Index")
boxplot(inv.D~Season, data=Y3, las=1,col= Seasons, ylab = "inverse Simpson Diversity Index")
boxplot(J~Season, data=Y3, las=1,col= Seasons, ylab = "Evenness")
#Creating main title
mtext("Alpha Diversity by Season - Year 3", side = 3, line = -2.4, outer = TRUE, cex = 1.4)

###### Differences by STATION - Richness, Shannon, inv. Simpson, Evenness ####
# Boxplot colors by station
## Expanding the color palette using color ramp
library(RColorBrewer)
nb.cols <- 20 #defines the number of colors you want
Stations <- colorRampPalette(brewer.pal(12, "Paired"))(nb.cols) #now the color ramp has 20 colors


#Year 1
kruskal.test(Y1$S ~ Y1$Station)
#Kruskal-Wallis chi-squared = 38.321, df = 19, p-value = 0.0054
pairwise.wilcox.test(Y1$S, Y1$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$S ~ Y1$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# CLV10A KISSR0.0    L001     L004     L005     L006     L007     L008      LZ2    LZ25A     LZ30     LZ40
PALMOUT  PELBAY3   POLE3S
# "ab"     "ab"      "a"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"
"ab"     "ab"      "b"
# POLESOUT  RITTAE2     S308      S77      S79
#     "ab"     "ab"     "ab"     "ab"     "ab"

kruskal.test(Y1$H ~ Y1$Station)
#Kruskal-Wallis chi-squared = 40.886, df = 19, p-value = 0.002499
pairwise.wilcox.test(Y1$H, Y1$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$H ~ Y1$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# CLV10A KISSR0.0    L001     L004     L005     L006     L007     L008      LZ2    LZ25A     LZ30     LZ40
PALMOUT  PELBAY3   POLE3S
# "ab"     "ab"      "a"     "ab"     "ab"     "ab"      "b"     "ab"     "ab"      "b"     "ab"     "ab"
"ab"     "ab"      "b"
# POLESOUT  RITTAE2     S308      S77      S79
# "ab"      "b"      "ab"     "ab"     "ab"

kruskal.test(Y1$inv.D ~ Y1$Station)
#Kruskal-Wallis chi-squared = 40.482, df = 19, p-value = 0.002827
pairwise.wilcox.test(Y1$inv.D, Y1$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$inv.D ~ Y1$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
```

```
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# CLV10A KISSR0.0   L001     L004     L005     L006     L007     L008      LZ2    LZ25A     LZ30     LZ40
PALMOUT  PELBAY3   POLE3S
# "ab"     "ab"      "a"     "ab"     "ab"     "b"     "b"     "ab"     "ab"      "b"     "ab"     "ab"     "ab"
"ab"      "b"
# POLESOUT  RITTAE2     S308      S77      S79
# "ab"     "b"         "ab"     "ab"     "ab"

kruskal.test(Y1$J ~ Y1$Station)
#Kruskal-Wallis chi-squared = 34.478, df = 19, p-value = 0.01613
pairwise.wilcox.test(Y1$J, Y1$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$J ~ Y1$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGNIFICANT DIFFERENCES FOUND!!

## Plotting all the Year 1 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Station, data=Y1, las=2, col= Stations, ylab = "Species Richness", xlab = "", cex.axis = 0.88)
boxplot(H~Station, data=Y1, las=2,col= Stations, ylab = "Shannon Diversity Index", xlab = "", cex.axis = 0.88)
boxplot(inv.D~Station, data=Y1, las=2,col= Stations, ylab = "inverse Simpson Diversity Index", xlab = "",
cex.axis = 0.88)
boxplot(J~Station, data=Y1, las=2,col= Stations, ylab = "Evenness", xlab = "", cex.axis = 0.88)
#Creating main title
mtext("Alpha Diversity by Station - Year 1", side = 3, line = - 2.4, outer = TRUE, cex = 1.4)


#Year 2 - Differences found in evenness
kruskal.test(Y2$S ~ Y2$Station)
#Kruskal-Wallis chi-squared = 7.7969, df = 19, p-value = 0.9886
pairwise.wilcox.test(Y2$S, Y2$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$S ~ Y2$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGNIFICANT DIFFERENCES FOUND!!

kruskal.test(Y2$H ~ Y2$Station)
#Kruskal-Wallis chi-squared = 12.192, df = 19, p-value = 0.8772
pairwise.wilcox.test(Y2$H, Y2$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$H ~ Y2$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGNIFICANT DIFFERENCES FOUND!!

kruskal.test(Y2$inv.D ~ Y2$Station)
#Kruskal-Wallis chi-squared = 21.503, df = 19, p-value = 0.3097
pairwise.wilcox.test(Y2$inv.D, Y2$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$inv.D ~ Y2$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
```

```
                      Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGNIFICANT DIFFERENCES FOUND!!

kruskal.test(Y2$J ~ Y2$Station)
#Kruskal-Wallis chi-squared = 36.956, df = 19, p-value = 0.008036
pairwise.wilcox.test(Y2$J, Y2$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$J ~ Y2$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                      Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# CLV10A KISSR0.0     L001     L004     L005     L006     L007     L008      LZ2    LZ25A     LZ30
# "ab"     "ab"      "a"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"
# LZ40  PALMOUT  PELBAY3   POLE3S POLESOUT  RITTAE2     S308      S77      S79
# "b"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"

## Plotting all the Year 2 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Station, data=Y2, las=2, col= Stations, ylab = "Species Richness", xlab = "", cex.axis = 0.88)
boxplot(H~Station, data=Y2, las=2,col= Stations, ylab = "Shannon Diversity Index", xlab = "", cex.axis = 0.88)
boxplot(inv.D~Station, data=Y2, las=2,col= Stations, ylab = "inverse Simpson Diversity Index", xlab = "",
cex.axis = 0.88)
boxplot(J~Station, data=Y2, las=2,col= Stations, ylab = "Evenness", xlab = "", cex.axis = 0.88)
#Creating main title
mtext("Alpha Diversity by Station - Year 2", side = 3, line = -2.4, outer = TRUE, cex = 1.4)



#Year 3
kruskal.test(Y3$S ~ Y3$Station)
#Kruskal-Wallis chi-squared = 36.513, df = 19, p-value = 0.009123
pairwise.wilcox.test(Y3$S, Y3$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$S ~ Y3$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                      Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# CLV10A KISSR0.0    L001     L004     L005     L006     L007     L008      LZ2    LZ25A     LZ30     LZ40
PALMOUT  PELBAY3   POLE3S
# "ab"     "ab"      "a"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"
"ab"     "ab"     "ab"
# POLESOUT  RITTAE2     S308      S77      S79
# "ab"     "ab"          "ab"     "ab"      "b"

kruskal.test(Y3$H ~ Y3$Station)
#Kruskal-Wallis chi-squared = 37.551, df = 19, p-value = 0.006766
pairwise.wilcox.test(Y3$H, Y3$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$H ~ Y3$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                      Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# CLV10A KISSR0.0     L001     L004     L005     L006     L007     L008      LZ2    LZ25A     LZ30     LZ40
PALMOUT  PELBAY3   POLE3S
# "ab"     "ab"        "a"     "ab"     "ab"     "ab"      "b"     "ab"     "ab"     "ab"     "ab"     "ab"
"ab"     "ab"     "ab"
# POLESOUT  RITTAE2     S308      S77      S79
# "ab"     "ab"          "ab"     "ab"     "ab"

kruskal.test(Y3$inv.D ~ Y3$Station)
#Kruskal-Wallis chi-squared = 42.098, df = 19, p-value = 0.001719
pairwise.wilcox.test(Y3$inv.D, Y3$Station, p.adjust.method = "fdr")
```

```
kmc <- kruskalmc(Y3$inv.D ~ Y3$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# CLV10A KISSR0.0     L001     L004     L005     L006     L007     L008      LZ2    LZ25A     LZ30     LZ40
PALMOUT  PELBAY3   POLE3S
# "ab"      "ab"        "a"     "ab"     "ab"     "ab"      "b"   "ab"      "ab"     "ab"     "ab"     "ab"
"ab"     "ab"     "ab"
# POLESOUT  RITTAE2     S308      S77      S79
# "ab"     "ab"         "ab"     "ab"     "ab"

kruskal.test(Y3$J ~ Y3$Station)
#Kruskal-Wallis chi-squared = 42.614, df = 19, p-value = 0.001463
pairwise.wilcox.test(Y3$J, Y3$Station, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$J ~ Y3$Station) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# CLV10A KISSR0.0     L001     L004     L005     L006     L007     L008      LZ2    LZ25A     LZ30
# "ab"     "ab"      "a"     "ab"     "ab"     "ab"      "b"     "ab"     "ab"     "ab"     "ab"
# LZ40  PALMOUT  PELBAY3   POLE3S POLESOUT  RITTAE2     S308      S77      S79
# "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"     "ab"

## Plotting all the Year 3 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Station, data=Y3, las=2, col= Stations, ylab = "Species Richness", xlab = "", cex.axis = 0.88)
boxplot(H~Station, data=Y3, las=2,col= Stations, ylab = "Shannon Diversity Index", xlab = "", cex.axis = 0.88)
boxplot(inv.D~Station, data=Y3, las=2,col= Stations, ylab = "inverse Simpson Diversity Index", xlab = "",
cex.axis = 0.88)
boxplot(J~Station, data=Y3, las=2,col= Stations, ylab = "Evenness", xlab = "", cex.axis = 0.88)
#Creating main title
mtext("Alpha Diversity by Station - Year 3", side = 3, line = - 2.4, outer = TRUE, cex = 1.4)

###### Differences by MONTH - Richness, Shannon, inv. Simpson, Evenness ####
# Boxplot colors by month (different for each year)
Year1col <- c("lightgoldenrod1","goldenrod1","green3","cadetblue2","dodgerblue2",
                            "mediumpurple2","lightpink1","tan","sienna","seashell3")
Year2col <- c("firebrick2","darkorange1","lightgoldenrod1","goldenrod1","green3",
                          "cadetblue2","dodgerblue2","mediumpurple2","lightpink1",
                          "tan","sienna","seashell3")
Year3col <- c("firebrick2","darkorange1","lightgoldenrod1","goldenrod1","green3",
                          "cadetblue2","dodgerblue2","mediumpurple2","lightpink1",
                          "tan")
#Year 1
kruskal.test(Y1$S ~ Y1$Month)
#Kruskal-Wallis chi-squared = 26.535, df = 9, p-value = 0.001669
pairwise.wilcox.test(Y1$S, Y1$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$S ~ Y1$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#   3     4     5     6     7     8     9    10    11    12
# "ab" "abc" "abc"   "a" "abc"   "c" "abc" "abc"  "bc" "abc"

kruskal.test(Y1$H ~ Y1$Month)
#Kruskal-Wallis chi-squared = 25.593, df = 9, p-value = 0.002381
pairwise.wilcox.test(Y1$H, Y1$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$H ~ Y1$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
```

```
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#   3    4    5    6    7    8    9   10   11   12
# "ab" "ab" "ab"  "a" "ab" "ab" "ab" "ab"  "b" "ab"

kruskal.test(Y1$inv.D ~ Y1$Month)
#Kruskal-Wallis chi-squared = 18.778, df = 9, p-value = 0.02715
pairwise.wilcox.test(Y1$inv.D, Y1$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$inv.D ~ Y1$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGNIFICANT DIFFERENCES FOUND!!

kruskal.test(Y1$J ~ Y1$Month)
#Kruskal-Wallis chi-squared = 13.89, df = 9, p-value = 0.1263
pairwise.wilcox.test(Y1$J, Y1$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y1$J ~ Y1$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#NO SIGNIFICANT DIFFERENCES FOUND!!

## Plotting all the Year 1 boxplots on one graph
#defining plotting area as one row and 4 columns
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Month, data=Y1, las=1, col= Year1col, ylab = "Species Richness")
boxplot(H~Month, data=Y1, las=1,col= Year1col, ylab = "Shannon Diversity Index")
boxplot(inv.D~Month, data=Y1, las=1,col= Year1col, ylab = "inverse Simpson Diversity Index")
boxplot(J~Month, data=Y1, las=1,col= Year1col, ylab = "Evenness")
#Creating main title
mtext("Alpha Diversity by Month - Year 1", side = 3, line = -2.4, outer = TRUE, cex = 1.4)


#Year 2
kruskal.test(Y2$S ~ Y2$Month)
#Kruskal-Wallis chi-squared = 144.03, df = 11, p-value < 2.2e-16
pairwise.wilcox.test(Y2$S, Y2$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$S ~ Y2$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# 1      2      3      4      5      6      7      8      9     10     11     12
# "abc"  "abc"    "d"   "de"    "d"    "d"  "ade" "abce"   "bc"    "b"    "b" "acde"

kruskal.test(Y2$H ~ Y2$Month)
#Kruskal-Wallis chi-squared = 131.82, df = 11, p-value < 2.2e-16
pairwise.wilcox.test(Y2$H, Y2$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$H ~ Y2$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
```

```
let # significant letters for the multiple comparison test
# 1     2     3     4     5     6     7     8     9    10    11    12
# "ab"  "ab"   "c"  "cd"  "cd"  "cd" "acd"  "ab"  "ab"  "ab"   "b"  "ad"

kruskal.test(Y2$inv.D ~ Y2$Month)
#Kruskal-Wallis chi-squared = 104.87, df = 11, p-value < 2.2e-16
pairwise.wilcox.test(Y2$inv.D, Y2$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$inv.D ~ Y2$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
# 1     2     3     4     5     6     7     8     9    10    11    12
# "a"   "a"   "b"  "bc"  "bc"  "bc" "abc"   "a"   "a"   "a"   "a"  "ac"

kruskal.test(Y2$J ~ Y2$Month)
#Kruskal-Wallis chi-squared = 34.984, df = 11, p-value = 0.0002494
pairwise.wilcox.test(Y2$J, Y2$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y2$J ~ Y2$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#   1    2    3    4    5    6    7    8    9   10   11   12
# "ab"  "a" "ab" "ab" "ab"  "b" "ab" "ab" "ab" "ab"  "b" "ab"

## Plotting all the Year 2 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Month, data=Y2, las=1, col= Year2col, ylab = "Species Richness")
boxplot(H~Month, data=Y2, las=1,col= Year2col, ylab = "Shannon Diversity Index")
boxplot(inv.D~Month, data=Y2, las=1,col= Year2col, ylab = "inverse Simpson Diversity Index")
boxplot(J~Month, data=Y2, las=1,col= Year2col, ylab = "Evenness")
#Creating main title
mtext("Alpha Diversity by Month - Year 2", side = 3, line = - 2.4, outer = TRUE, cex = 1.4)


#Year 3
kruskal.test(Y3$S ~ Y3$Month)
#Kruskal-Wallis chi-squared = 50.462, df = 9, p-value = 8.819e-08
pairwise.wilcox.test(Y3$S, Y3$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$S ~ Y3$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#   1     2     3     4     5     6     7     8     9    10
# "ab"   "c"   "a"  "ab" "abc"  "bc"  "bc"  "bc" "abc"  "ab"

kruskal.test(Y3$H ~ Y3$Month)
#Kruskal-Wallis chi-squared = 45.298, df = 9, p-value = 8.126e-07
pairwise.wilcox.test(Y3$H, Y3$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$H ~ Y3$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#    1     2     3     4     5     6     7     8     9    10
# "abc"   "a"   "b" "abc"  "ac"   "a" "abc" "abc" "abc"  "bc"
```

```
kruskal.test(Y3$inv.D ~ Y3$Month)
#Kruskal-Wallis chi-squared = 38.56, df = 9, p-value = 1.383e-05
pairwise.wilcox.test(Y3$inv.D, Y3$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$inv.D ~ Y3$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#   1    2    3    4    5    6    7    8    9   10
# "ab"  "a"  "b" "ab" "ab"  "a" "ab" "ab" "ab"  "b"

kruskal.test(Y3$J ~ Y3$Month)
#Kruskal-Wallis chi-squared = 36.807, df = 9, p-value = 2.848e-05
pairwise.wilcox.test(Y3$J, Y3$Month, p.adjust.method = "fdr")
kmc <- kruskalmc(Y3$J ~ Y3$Month) # multiple-comparison test
kmc # comparisons TRUE= significantly different or FALSE= not significantly different
# To look for homogeneous groups, and give each group a code (letter):
test <- kmc$dif.com$difference # select logical vector
names(test) <- row.names(kmc$dif.com)# add comparison names
# create a list with "homogeneous groups" coded by letter
let <- multcompLetters(test, compare="<", threshold=0.05,
                       Letters=c(letters, LETTERS, "."),reversed = FALSE)
let # significant letters for the multiple comparison test
#   1    2    3    4    5    6    7    8    9   10
# "ab"  "a"  "b" "ab" "ab" "ab" "ab"  "b" "ab"  "b"

## Plotting all the Year 3 boxplots on one graph
par(mfrow = c(1,4))
#plotting the boxplots for each alpha diversity variable
boxplot(S~Month, data=Y3, las=1, col= Year3col, ylab = "Species Richness")
boxplot(H~Month, data=Y3, las=1,col= Year3col, ylab = "Shannon Diversity Index")
boxplot(inv.D~Month, data=Y3, las=1,col= Year3col, ylab = "inverse Simpson Diversity Index")
boxplot(J~Month, data=Y3, las=1,col= Year3col, ylab = "Evenness")
#Creating main title
mtext("Alpha Diversity by Month - Year 3", side = 3, line = -2.4, outer = TRUE, cex = 1.4)



###### Correlation of alpha diversity measures and chlorophyll a ######
metadata <- read.csv("Metadata-Diversity_BATCH.csv", row.names = 1)
par(mfrow=c(2,2))
#Shannon vs. Chl.a
#calculating correlation (-1 to 0 to +1; negatively correlated to none to positively correlated)
cor.test(metadata$Chlorophyll.a, metadata$H, method ="pearson")
#t = -0.74435, df = 539, p-value = 0.457, Pearson coeff. = -0.03204502 <- NOT SIGNIFICANT
#plotting them against each other
plot(metadata$Chlorophyll.a, metadata$H, pch = 19, col = "gray52", xlab = "", ylab = "")
# Adding text
title(main="Shannon Diversity vs Chlorophyll-a Correlation",
      xlab = "Chlorophyll a (ug/L)",
      ylab = "Shannon Diversity Index")
#inv.Simpson vs. Chl.a
cor.test(metadata$Chlorophyll.a, metadata$inv.D, method ="pearson")
# t = 1.1217, df = 539, p-value = 0.2625, Pearson coeff. = 0.04825728 <- NOT SIGNIFICANT
plot(metadata$Chlorophyll.a, metadata$inv.D, pch = 19, col = "gray52", xlab = "", ylab = "")
title(main="inverse Simpson Diversity vs Chlorophyll-a Correlation",
      xlab = "Chlorophyll a (ug/L)",
      ylab = "inverse Simpson Diversity Index")
#Richness vs. Chl.a
cor.test(metadata$Chlorophyll.a, metadata$S, method ="pearson")
# t = 0.49649, df = 539, p-value = 0.6198, Pearson coeff. = 0.0213804 <- NOT SIGNIFICANT
plot(metadata$Chlorophyll.a, metadata$S, pch = 19, col = "gray52", xlab = "", ylab = "")
title(main="Species Richness vs Chlorophyll-a Correlation",
      xlab = "Chlorophyll a (ug/L)",
      ylab = "Species Richness")
#Evenness vs. Chl.a
cor.test(metadata$Chlorophyll.a, metadata$J, method ="pearson")
# t = -1.9153, df = 539, p-value = 0.05599, Pearson coeff. = -0.08221648   <- NOT SIGNIFICANT
plot(metadata$Chlorophyll.a, metadata$J, pch = 19, col = "gray52", xlab = "", ylab = "")
title(main="Evenness vs Chlorophyll-a Correlation",
      xlab = "Chlorophyll a (ug/L)",
      ylab = "Evenness")
```

```
###### Correlation of Microcystis  vs. Chl a (and Microcystin LR) ######
metadata <- read.csv("Metadata-Diversity_BATCH.csv", row.names = 1)
## Chl a
#calculating correlation (-1 to 0 to +1; negatively correlated to no correlation to positively correlated)
cor.test(metadata$Chlorophyll.a, metadata$Microcystis.Abundance, method ="pearson")
# t = 5.4696, df = 539, p-value = 6.914e-08, Pearson coeff. = 0.229314    -> weakly positive (SIGNIFICANT)
#plotting them against each other
plot(metadata$Microcystis.Abundance, metadata$Chlorophyll.a, pch = 19, xlab = "", ylab = "")
lines(lowess(metadata$Microcystis.Abundance, metadata$Chlorophyll.a), col = 2, lwd = 2)
# Adding text
title(main="Microcystis Relative Abundance vs Chlorophyll-a Correlation",
      xlab = "Microcystis Relative Abundance",
      ylab = "Chlorophyll a (ug/L)")
text(0.063,136,"Pearson R: 0.23", cex=1.05)
## Microcystin
cor.test(metadata$Microcystin.LR, metadata$Microcystis.Abundance, method ="pearson")
# t = 17.318, df = 539, p-value < 2.2e-16, Pearson coeff. = 0.5979055     -> positive (SIGNIFICANT)
#plotting them against each other
plot(metadata$Microcystis.Abundance, metadata$Microcystin.LR, pch = 19, xlab = "", ylab = "")
lines(lowess(metadata$Microcystis.Abundance, metadata$Microcystin.LR), col = 2, lwd = 2)
# Adding text
title(main="Microcystis Relative Abundance vs Microcystin (ug/L) Correlation",
      xlab = "Microcystis Relative Abundance",
      ylab = "Microcystin (ug/L)")
text(0.063,45,"Pearson R: 0.60", cex=1.05)

###### Alpha Diversity vs Microcystis Abundance ######
metadata <- read.csv("Metadata-Diversity_BATCH.csv", row.names = 1)
##Scatter plots
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$Microcystis.Abundance, metadata$S, pch = 19, xlab= "Microcystis Relative Abundance",
     ylab = "Species Richness")
lines(lowess(metadata$Microcystis.Abundance, metadata$S), col = 2, lwd = 2)
title(main="Species richness vs Microcystis Relative Abundance", cex.main = 1)

plot(metadata$Microcystis.Abundance, metadata$H, pch = 19, xlab= "Microcystis Relative Abundance",
     ylab = "Shannon Diversity Index")
lines(lowess(metadata$Microcystis.Abundance, metadata$H), col = 2, lwd = 2)
text(0.062,6,"Pearson's r = -0.23", cex=0.9)
title(main="Shannon Diversity vs Microcystis Relative Abundance", cex.main = 1)

plot(metadata$Microcystis.Abundance, metadata$J, pch = 19, xlab= "Microcystis Relative Abundance",
     ylab = "Evenness")
lines(lowess(metadata$Microcystis.Abundance, metadata$J), col = 2, lwd = 2)
text(0.062,0.88,"Pearson's r = -0.72", cex=0.9)
title(main="Species Evenness vs Microcystis Relative Abundance", cex.main = 1)

plot(metadata$Microcystis.Abundance, metadata$inv.D, pch = 19, xlab= "Microcystis Relative Abundance",
     ylab = "inverse Simpson Diversity Index")
lines(lowess(metadata$Microcystis.Abundance, metadata$inv.D), col = 2, lwd = 2)
text(0.062,400,"Pearson's r = -0.22", cex=0.9)
title(main="inverse Simpson Diversity vs Microcystis Relative Abundance", cex.main = 1)


## Looking at the correlations
cor.test(metadata$Microcystis.Abundance, metadata$S, method ="pearson")
#t = 1.4678, df = 539, Pearson coeff. = 0.0630954 , p-value = 0.1427 -> NOT SIGNIFICANT (NO CORRELATION)
cor.test(metadata$Microcystis.Abundance, metadata$H, method ="pearson")
#t = -5.5028, df = 539, Pearson coeff. = -0.2306343, p-value = 5.785e-08 -> SIGNIFICANT (NEG. CORRELATION)
cor.test(metadata$Microcystis.Abundance, metadata$J, method ="pearson")
#t = -24.34, df = 539, Pearson coeff. = -0.7236151, p-value < 2.2e-16 -> SIGNIFICANT (NEG. CORRELATION)
cor.test(metadata$Microcystis.Abundance, metadata$inv.D, method ="pearson")
#t = -5.3297, df = 539, Pearson coeff. = -0.2237471, p-value = 1.448e-07 -> SIGNIFICANT (NEG. CORRELATION)

###### Alpha Diversity vs Environmental Variables - Scatter plots ######
metadata <- read.csv("Metadata-Diversity_BATCH.csv", row.names = 1)
##Scatter plots
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph

#Chlorophyll a
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$Chlorophyll.a, metadata$S, pch = 19, xlab= "Chlorophyll a (ug/L)",
     ylab = "Species Richness", col="grey54")
title(main="Species richness vs Chlorophyll a (ug/L)", cex.main = 1)
plot(metadata$Chlorophyll.a, metadata$H, pch = 19, xlab= "Chlorophyll a (ug/L)",
```

```
     ylab = "Shannon Diversity Index", col="grey54")
title(main="Shannon Diversity vs Chlorophyll a (ug/L)", cex.main = 1)
plot(metadata$Chlorophyll.a, metadata$J, pch = 19, xlab= "Chlorophyll a (ug/L)",
     ylab = "Evenness", col="grey54")
title(main="Species Evenness vs Chlorophyll a (ug/L)", cex.main = 1)
plot(metadata$Chlorophyll.a, metadata$inv.D, pch = 19, xlab= "Chlorophyll a (ug/L)",
     ylab = "inverse Simpson Diversity Index", col="grey54")
title(main="inverse Simpson Diversity vs Chlorophyll a (ug/L)", cex.main = 1)

#Ammonia
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$Ammonia, metadata$S, pch = 19, xlab= "Ammonia (mg/L)",
     ylab = "Species Richness", col="grey54")
title(main="Species richness vs Ammonia (mg/L)", cex.main = 1)
plot(metadata$Ammonia, metadata$H, pch = 19, xlab= "Ammonia (mg/L)",
     ylab = "Shannon Diversity Index", col="grey54")
title(main="Shannon Diversity vs Ammonia (mg/L)", cex.main = 1)
plot(metadata$Ammonia, metadata$J, pch = 19, xlab= "Ammonia (mg/L)",
     ylab = "Evenness")
lines(lowess(metadata$Ammonia, metadata$J), col = 2, lwd = 2)
text(0.68,0.8,"Pearson's r = 0.11", cex=0.9)
title(main="Species Evenness vs Ammonia (mg/L)", cex.main = 1)
plot(metadata$Ammonia, metadata$inv.D, pch = 19, xlab= "Ammonia (mg/L)",
     ylab = "inverse Simpson Diversity Index", col="grey54")
title(main="inverse Simpson Diversity vs Ammonia (mg/L)", cex.main = 1)

#Nitrate(ite)
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$Nitrate.Nitrite, metadata$S, pch = 19, xlab= "Nitrate + Nitrite (mg/L)",
     ylab = "Species Richness", col="grey54")
title(main="Species richness vs Nitrate + Nitrite (mg/L)", cex.main = 1)
plot(metadata$Nitrate.Nitrite, metadata$H, pch = 19, xlab= "Nitrate + Nitrite (mg/L)",
     ylab = "Shannon Diversity Index", col="grey54")
title(main="Shannon Diversity vs Nitrate + Nitrite (mg/L)", cex.main = 1)
plot(metadata$Nitrate.Nitrite, metadata$J, pch = 19, xlab= "Nitrate + Nitrite (mg/L)",
     ylab = "Evenness")
lines(lowess(metadata$Nitrate.Nitrite, metadata$J), col = 2, lwd = 2)
text(0.5,0.7,"Pearson's r = -0.10", cex=0.9)
title(main="Species Evenness vs Nitrate + Nitrite (mg/L)", cex.main = 1)
plot(metadata$Nitrate.Nitrite, metadata$inv.D, pch = 19, xlab= "Nitrate + Nitrite (mg/L)",
     ylab = "inverse Simpson Diversity Index")
lines(lowess(metadata$Nitrate.Nitrite, metadata$inv.D), col = 2, lwd = 2)
text(0.52,400,"Pearson's r = -0.10", cex=0.9)
title(main="inverse Simpson Diversity vs Nitrate + Nitrite (mg/L)", cex.main = 1)

#Total Phosphorus
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$Phosphate.Total, metadata$S, pch = 19, xlab= "Total Phosphorus (mg/L)",
     ylab = "Species Richness")
lines(lowess(metadata$Phosphate.Total, metadata$S), col = 2, lwd = 2)
text(0.45,1750,"Pearson's r = 0.18", cex=0.9)
title(main="Species richness vs Total Phosphorus (mg/L)", cex.main = 1)
plot(metadata$Phosphate.Total, metadata$H, pch = 19, xlab= "Total Phosphorus (mg/L)",
     ylab = "Shannon Diversity Index")
lines(lowess(metadata$Phosphate.Total, metadata$H), col = 2, lwd = 2)
text(0.44,6.4,"Pearson's r = 0.06", cex=0.9)
title(main="Shannon Diversity vs Total Phosphorus (mg/L)", cex.main = 1)
plot(metadata$Phosphate.Total, metadata$J, pch = 19, xlab= "Total Phosphorus (mg/L)",
     ylab = "Evenness", col="grey54")
title(main="Species Evenness vs Total Phosphorus (mg/L)", cex.main = 1)
plot(metadata$Phosphate.Total, metadata$inv.D, pch = 19, xlab= "Total Phosphorus (mg/L)",
     ylab = "inverse Simpson Diversity Index")
lines(lowess(metadata$Phosphate.Total, metadata$inv.D), col = 2, lwd = 2)
text(0.44,400,"Pearson's r = 0.10", cex=0.9)
title(main="inverse Simpson Diversity vs Total Phosphorus (mg/L)", cex.main = 1)

#Microcystin
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$Microcystin.LR, metadata$S, pch = 19, xlab= "Microcystin (ug/L)",
     ylab = "Species Richness", col="grey54")
title(main="Species richness vs Microcystin (ug/L)", cex.main = 1)
plot(metadata$Microcystin.LR, metadata$H, pch = 19, xlab= "Microcystin (ug/L)",
     ylab = "Shannon Diversity Index")
lines(lowess(metadata$Microcystin.LR, metadata$H), col = 2, lwd = 2)
text(48,6.2,"Pearson's r = -0.23", cex=0.9)
title(main="Shannon Diversity vs Microcystin (ug/L)", cex.main = 1)
```

```
plot(metadata$Microcystin.LR, metadata$J, pch = 19, xlab= "Microcystin (ug/L)",
     ylab = "Evenness")
lines(lowess(metadata$Microcystin.LR, metadata$J), col = 2, lwd = 2)
text(48,0.88,"Pearson's r = -0.49", cex=0.9)
title(main="Species Evenness vs Microcystin (ug/L)", cex.main = 1)
plot(metadata$Microcystin.LR, metadata$inv.D, pch = 19, xlab= "Microcystin (ug/L)",
     ylab = "inverse Simpson Diversity Index")
lines(lowess(metadata$Microcystin.LR, metadata$inv.D), col = 2, lwd = 2)
text(46,375,"Pearson's r = -0.20", cex=0.9)
title(main="inverse Simpson Diversity vs Microcystin (ug/L)", cex.main = 1)

#Temperature
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$Temperature, metadata$S, pch = 19, xlab= "Temperature (°C)",
     ylab = "Species Richness", col="grey54")
title(main="Species richness vs Temperature (°C)", cex.main = 1)
plot(metadata$Temperature, metadata$H, pch = 19, xlab= "Temperature (°C)",
     ylab = "Shannon Diversity Index", col="grey54")
title(main="Shannon Diversity vs Temperature (°C)", cex.main = 1)
plot(metadata$Temperature, metadata$J, pch = 19, xlab= "Temperature (°C)",
     ylab = "Evenness", col="grey54")
title(main="Species Evenness vs Temperature (°C)", cex.main = 1)
plot(metadata$Temperature, metadata$inv.D, pch = 19, xlab= "Temperature (°C)",
     ylab = "inverse Simpson Diversity Index", col="grey54")
title(main="inverse Simpson Diversity vs Temperature (°C)", cex.main = 1)

#Total Nitrogen
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$Total.Nitrogen, metadata$S, pch = 19, xlab= "Total Nitrogen (mg/L)",
     ylab = "Species Richness")
lines(lowess(metadata$Total.Nitrogen, metadata$S), col = 2, lwd = 2)
text(3.15,1750,"Pearson's r = 0.17", cex=0.9)
title(main="Species richness vs Total Nitrogen (mg/L)", cex.main = 1)
plot(metadata$Total.Nitrogen, metadata$H, pch = 19, xlab= "Total Nitrogen (mg/L)",
     ylab = "Shannon Diversity Index")
lines(lowess(metadata$Total.Nitrogen, metadata$H), col = 2, lwd = 2)
text(3,6.8,"Pearson's r = 0.13", cex=0.9)
title(main="Shannon Diversity vs Total Nitrogen (mg/L)", cex.main = 1)
plot(metadata$Total.Nitrogen, metadata$J, pch = 19, xlab= "Total Nitrogen (mg/L)",
     ylab = "Evenness", col="grey54")
title(main="Species Evenness vs Total Nitrogen (mg/L)", cex.main = 1)
plot(metadata$Total.Nitrogen, metadata$inv.D, pch = 19, xlab= "Total Nitrogen (mg/L)",
     ylab = "inverse Simpson Diversity Index")
lines(lowess(metadata$Total.Nitrogen, metadata$inv.D), col = 2, lwd = 2)
text(3.1,440,"Pearson's r = 0.17", cex=0.9)
title(main="inverse Simpson Diversity vs Total Nitrogen (mg/L)", cex.main = 1)

#pH
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$pH, metadata$S, pch = 19, xlab= "pH",
     ylab = "Species Richness")
lines(lowess(metadata$pH, metadata$S), col = 2, lwd = 2)
text(2,1730,"Pearson's r = -0.13", cex=0.9)
title(main="Species richness vs pH", cex.main = 1)
plot(metadata$pH, metadata$H, pch = 19, xlab= "pH",
     ylab = "Shannon Diversity Index")
lines(lowess(metadata$pH, metadata$H), col = 2, lwd = 2)
text(2,6.4,"Pearson's r = -0.15", cex=0.9)
title(main="Shannon Diversity vs pH", cex.main = 1)
plot(metadata$pH, metadata$J, pch = 19, xlab= "pH",
     ylab = "Evenness")
lines(lowess(metadata$pH, metadata$J), col = 2, lwd = 2)
text(2,0.74,"Pearson's r = -0.11", cex=0.9)
title(main="Species Evenness vs pH", cex.main = 1)
plot(metadata$pH, metadata$inv.D, pch = 19, xlab= "pH",
     ylab = "inverse Simpson Diversity Index")
lines(lowess(metadata$pH, metadata$inv.D), col = 2, lwd = 2)
text(2,400,"Pearson's r = -0.16", cex=0.9)
title(main="inverse Simpson Diversity vs pH", cex.main = 1)

#TN:TP
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$TN.TP.ratio, metadata$S, pch = 19, xlab= "TN : TP",
     ylab = "Species Richness")
lines(lowess(metadata$TN.TP.ratio, metadata$S), col = 2, lwd = 2)
text(40,1780,"Pearson's r = -0.13", cex=0.9)
```

```
title(main="Species richness vs TN : TP", cex.main = 1)
plot(metadata$TN.TP.ratio, metadata$H, pch = 19, xlab= "TN : TP",
     ylab = "Shannon Diversity Index", col="grey54")
title(main="Shannon Diversity vs TN : TP", cex.main = 1)
plot(metadata$TN.TP.ratio, metadata$J, pch = 19, xlab= "TN : TP",
     ylab = "Evenness", col="grey54")
title(main="Species Evenness vs TN : TP", cex.main = 1)
plot(metadata$TN.TP.ratio, metadata$inv.D, pch = 19, xlab= "TN : TP",
     ylab = "inverse Simpson Diversity Index", col="grey54")
title(main="inverse Simpson Diversity vs TN : TP", cex.main = 1)

#Total Phosphate
par(mfrow = c(2,2), mar = c(4, 4, 3, 3)) ## all plots on one graph
plot(metadata$Phosphate.Ortho, metadata$S, pch = 19, xlab= "Total Phosphate (mg/L)",
     ylab = "Species Richness", col="grey54")
title(main="Species richness vs Total Phosphate", cex.main = 1)
plot(metadata$Phosphate.Ortho, metadata$H, pch = 19, xlab= "Total Phosphate (mg/L)",
     ylab = "Shannon Diversity Index", col="grey54")
title(main="Shannon Diversity vs Total Phosphate", cex.main = 1)
plot(metadata$Phosphate.Ortho, metadata$J, pch = 19, xlab= "Total Phosphate (mg/L)",
     ylab = "Evenness")
lines(lowess(metadata$Phosphate.Ortho, metadata$J), col = 2, lwd = 2)
text(0.17,0.82,"Pearson's r = -0.11", cex=0.9)
title(main="Species Evenness vs Total Phosphate", cex.main = 1)
plot(metadata$Phosphate.Ortho, metadata$inv.D, pch = 19, xlab= "Total Phosphate (mg/L)",
     ylab = "inverse Simpson Diversity Index")
lines(lowess(metadata$Phosphate.Ortho, metadata$inv.D), col = 2, lwd = 2)
text(0.17,400,"Pearson's r = -0.12", cex=0.9)
title(main="inverse Simpson Diversity vs Total Phosphate", cex.main = 1)


###### Alpha Diversity vs Environmental Variables - Correlation Heat map ######
library(corrplot)
library(reshape2)

#load in metadata
metadata <- read.csv("Metadata-Diversity_BATCH.csv", row.names = 1)

#Making a dataframe with only env. variables and a-div measures
alphaenv <- metadata[, c(7:24,33,36:38,40:42)]
#changing some column names
colnames(alphaenv)[8] ="Pheophytin-a"
colnames(alphaenv)[9] ="Chlorophyll-a"
colnames(alphaenv)[12] ="Nitrate + Nitrite"
colnames(alphaenv)[13] ="Total.Phosphate"
colnames(alphaenv)[14] ="Total.Phosphorus"
colnames(alphaenv)[19] ="Microcystin"
colnames(alphaenv)[20] ="Anatoxin-a"

#Before making heatmap, we must first calculate the correlation coefficient
#between each variable using cor() and then transform the results into a usable
#format using the melt() function from the reshape2 package

#calculate correlation coefficients, rounded to 2 decimal places
envcor <- round(cor(alphaenv), 2) #this a correlation matrix
testRes <- cor.mtest(alphaenv, conf.level = 0.95) #generates a table of p-values

#creating heatmap
corrplot(envcor,
         type = "lower",
         method = 'color',
         col = COL2('BrBG', 10),
         p.mat = testRes$p,
         insig = 'label_sig',
         pch.cex = 0.98,
         pch.col = 'grey8',
         sig.level = c(0.001, 0.01, 0.05),
         order = 'original',
         number.cex = 0.8,
         tl.col = 'black',
         cl.ratio = 0.2,
         tl.srt = 45)


###### Beta Diversity - Creating Bray Curtis matrix #################
## re-creating relative abundance table
```

```
set.seed(1998)
dat<-read.csv("feature_Y123_ADJUSTED.csv", header=TRUE, row.names = 1)
dat<-data.matrix(dat)
typeof(dat)
dat <- t(dat)
row.names(dat)
metadata <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
typeof(metadata)
dat <- as.data.frame(dat)
typeof(dat)
common.rownames <- intersect(rownames(dat), rownames(metadata))
dat <- dat[common.rownames,]
metadata <- metadata[common.rownames,]
all.equal(rownames(dat),rownames(metadata))
otu.abund<-which(colSums(dat)>2)
dat.dom<-dat[,otu.abund]
dat.pa<-decostand(dat.dom, method ="pa")
dat.otus.01per<-which(colSums(dat.pa) > (0.01*nrow(dat.pa)))
dat.01per<-dat.dom[,dat.otus.01per]
dat.otus.001per<-which(colSums(dat.pa) > (0.001*nrow(dat.pa)))
dat.001per<-dat.dom[,dat.otus.001per]
dat.ra<-decostand(dat.01per, method = "total")

#use relative abundance table created
#creating Bray-Curtis dissimilarity distance matrix
ra.bc.dist<-vegdist(dat.ra, method = "bray")

#Separating into the different years
Y1r <- dat.ra[grep("_19$", rownames(dat.ra)),]
Y2r <- dat.ra[grep("_20$", rownames(dat.ra)),]
Y3r <- dat.ra[grep("_21$", rownames(dat.ra)),]
ra.bc.d.Y1<-vegdist(Y1r, method = "bray")
ra.bc.d.Y2<-vegdist(Y2r, method = "bray")
ra.bc.d.Y3<-vegdist(Y3r, method = "bray")
metadata <-read.csv("Metadata-Diversity_BATCH.csv", row.names = 1)



###### Plotting NMDS by Year - 2D ######
nmds2d <- metaMDS(ra.bc.dist,k=2,autotransform = F,trymax=20)
#Dimensions = 2
#Stress = 0.1705273
stressplot(nmds2d)
#Shepard plot "shows scatter around the regression between the inter-point
#distances in the final configuration (i.e., the distances between each pair of communities)
#against their original dissimilarities"


#Fitting environmental vectors to NMDS plot
ef.cca<- envfit(cca.p,metadata[,c(7,8,16)])
ef.cca$vectors$pvals

nmds.plot <- ordiplot(nmds2d,display="sites")
## Adding ellipses to group years
ordihull(nmds.plot,groups=metadata$Year,draw="lines",col=c("tomato3","steelblue3","springgreen3"))
##adjust colors to match each year, pch=20 makes it bullet points
points(nmds.plot,"sites", pch=20, col= "tomato4", select = metadata$Year == "1")
points(nmds.plot,"sites", pch=20, col= "steelblue4", select = metadata$Year == "2")
points(nmds.plot,"sites", pch=20, col= "springgreen4", select = metadata$Year == "3")
##Add Stress Value
text(1.2,1.5,"2D Stress: 0.17", cex=0.9)
##Adding legend
legend("topleft",legend= c("Year 1","Year 2", "Year 3"),
       title = "Year",
       col=c("tomato4","steelblue4","springgreen4"),
       pch=19, cex=1)
##Adding title
title(main="nMDS of Relative Abundances by Year")
#NMDS by Season
nmds.plot <- ordiplot(nmds2d,display="sites")
ordihull(nmds.plot,groups=metadata$Season,draw="lines",col = c("sienna4","royalblue3"))
points(nmds.plot,"sites", pch=20, col= "sienna4", select = metadata$Season == "dry")
points(nmds.plot,"sites", pch=20, col= "royalblue3", select = metadata$Season == "wet")
text(1.2,1.5,"2D Stress: 0.17", cex=0.9)
legend("topleft",legend= c("dry","wet"),
       title = "Season",
```

```
       col=c("sienna4","royalblue3"),
       pch=19, cex=1)
title(main="nMDS of Relative Abundances by Season")
#NMDS by Zone
nmds.plot <- ordiplot(nmds2d,display="sites")
ordihull(nmds.plot,groups=metadata$Zone,draw="lines",col =
c("palegreen3","wheat4","cornflowerblue","violetred2"))
points(nmds.plot,"sites", pch=20, col= "palegreen3", select = metadata$Zone == "Inflow")
points(nmds.plot,"sites", pch=20, col= "wheat4", select = metadata$Zone == "Nearshore")
points(nmds.plot,"sites", pch=20, col= "cornflowerblue", select = metadata$Zone == "Pelagic")
points(nmds.plot,"sites", pch=20, col= "violetred2", select = metadata$Zone == "S79")
text(1.2,1.5,"2D Stress: 0.17", cex=0.9)
legend("topleft",legend= c("Inflow","Nearshore","Pelagic", "S79"),
       title = "Zone",
       col=c("palegreen3","wheat4","cornflowerblue","violetred2"),
       pch=19, cex=1)
title(main="nMDS of Relative Abundances by Zone")
#NMDS by Month
nmds.plot <- ordiplot(nmds2d,display="sites")
ordihull(nmds.plot,groups=metadata$Month,draw="lines",col=c("firebrick2","darkorange1","gray34","goldenrod2","gr
een3","cadetblue2","dodgerblue2",

"mediumpurple2","hotpink","tan","sienna","purple4"))
points(nmds.plot,"sites", pch=19, col= "firebrick2", select = metadata$Month == "1")
points(nmds.plot,"sites", pch=19, col= "darkorange1", select = metadata$Month == "2")
points(nmds.plot,"sites", pch=19, col= "gray34", select = metadata$Month == "3")
points(nmds.plot,"sites", pch=19, col= "goldenrod2", select = metadata$Month == "4")
points(nmds.plot,"sites", pch=19, col= "green3", select = metadata$Month == "5")
points(nmds.plot,"sites", pch=19, col= "cadetblue2", select = metadata$Month == "6")
points(nmds.plot,"sites", pch=19, col= "dodgerblue2", select = metadata$Month == "7")
points(nmds.plot,"sites", pch=19, col= "mediumpurple2", select = metadata$Month == "8")
points(nmds.plot,"sites", pch=19, col= "hotpink", select = metadata$Month == "9")
points(nmds.plot,"sites", pch=19, col= "tan", select = metadata$Month == "10")
points(nmds.plot,"sites", pch=19, col= "sienna", select = metadata$Month == "11")
points(nmds.plot,"sites", pch=19, col= "purple4", select = metadata$Month == "12")
text(1.8,1.5,"2D Stress: 0.17", cex=0.9)
legend("topleft",legend= c("1","2","3","4","5", "6","7","8","9","10","11","12"), title = "Month",
       col=c("firebrick2","darkorange1","gray34","goldenrod2","green3","cadetblue2","dodgerblue2",
                        "mediumpurple2","hotpink","tan","sienna","purple4"), pch=19,ncol=2, cex=0.88)
title(main="nMDS of Relative Abundances by Month")
#NMDS by Station
nmds.plot <- ordiplot(nmds2d,display="sites")
ordihull(nmds.plot,groups=metadata$Station,draw="lines",col=c("#A6CEE3","#579CC7","#3688AD",
                                                              "#8BC395","#89CB6C",
                                                              "#40A635","#919D5F",
                                                              "#F99392","#EB494A",
                                                              "#E83C2D","#F79C5D",
                                                              "#FDA746","#FE8205",
                                                              "#E39970", "#BFA5CF",
                                                              "#8861AC","#917099",
                                                              "#E7E099","#DEB969",
                                                              "#B15928"))
points(nmds.plot,"sites", pch=19, col= "#A6CEE3", select = metadata$Station == "CLV10A")
points(nmds.plot,"sites", pch=19, col= "#579CC7", select = metadata$Station == "KISSR0.0")
points(nmds.plot,"sites", pch=19, col= "#3688AD", select = metadata$Station == "L001")
points(nmds.plot,"sites", pch=19, col= "#8BC395", select = metadata$Station == "L004")
points(nmds.plot,"sites", pch=19, col= "#89CB6C", select = metadata$Station == "L005")
points(nmds.plot,"sites", pch=19, col= "#40A635", select = metadata$Station == "L006")
points(nmds.plot,"sites", pch=19, col= "#919D5F", select = metadata$Station == "L007")
points(nmds.plot,"sites", pch=19, col= "#F99392", select = metadata$Station == "L008")
points(nmds.plot,"sites", pch=19, col= "#EB494A", select = metadata$Station == "LZ2")
points(nmds.plot,"sites", pch=19, col= "#E83C2D", select = metadata$Station == "LZ25A")
points(nmds.plot,"sites", pch=19, col= "#F79C5D", select = metadata$Station == "LZ30")
points(nmds.plot,"sites", pch=19, col= "#FDA746", select = metadata$Station == "LZ40")
points(nmds.plot,"sites", pch=19, col= "#FE8205", select = metadata$Station == "PALMOUT")
points(nmds.plot,"sites", pch=19, col= "#E39970", select = metadata$Station == "PELBAY3")
points(nmds.plot,"sites", pch=19, col= "#BFA5CF", select = metadata$Station == "POLE3S")
points(nmds.plot,"sites", pch=19, col= "#8861AC", select = metadata$Station == "POLESOUT")
points(nmds.plot,"sites", pch=19, col= "#917099", select = metadata$Station == "RITTAE2")
points(nmds.plot,"sites", pch=19, col= "#E7E099", select = metadata$Station == "S308")
points(nmds.plot,"sites", pch=19, col= "#DEB969", select = metadata$Station == "S77")
points(nmds.plot,"sites", pch=19, col= "#B15928", select = metadata$Station == "S79")
text(1.8,1.5,"2D Stress: 0.17", cex=0.9)
legend("topleft",legend= c("CLV10A","KISSR0.0","L001","L004","L005","L006","L007",
                           "L008","LZ2","LZ25A","LZ30","LZ40","PALMOUT","PELBAY3",
                           "POLE3S","POLESOUT","RITTAE2","S308","S77","S79"),title = "Station",
```

```
       col=c("#A6CEE3","#579CC7","#3688AD","#8BC395","#89CB6C","#40A635","#919D5F",
                      "#F99392","#EB494A","#E83C2D","#F79C5D","#FDA746","#FE8205",
                      "#E39970","#BFA5CF","#8861AC","#917099","#E7E099","#DEB969",
                      "#B15928"),ncol=2,pch=19, cex=0.74)
title(main="nMDS of Relative Abundances by Station")

#Statistics
anosim(ra.bc.dist, metadata$Year, permutations = 999, distance = "bray")
# ANOSIM statistic R: -0.003354
# Significance: 0.748 -> NOT SIGNIFICANT
anosim(ra.bc.dist, metadata$Season, permutations = 999, distance = "bray")
# ANOSIM statistic R: -0.004122
# Significance: 0.78  -> NOT SIGNIFICANT
anosim(ra.bc.dist, metadata$Month, permutations = 999, distance = "bray")
# ANOSIM statistic R: -0.00777
# Significance: 0.913 -> NOT SIGNIFICANT
anosim(ra.bc.dist, metadata$Zone, permutations = 999, distance = "bray")
# ANOSIM statistic R: 0.01493
# Significance: 0.191 -> NOT SIGNIFICANT
anosim(ra.bc.dist, metadata$Station, permutations = 999, distance = "bray")
# ANOSIM statistic R: 0.1967
# Significance: 0.001

###### Plotting NMDS separated by Year - 2D ONLY ####
###Year 1
nmdsY1 <- metaMDS(ra.bc.d.Y1,k=2,autotransform = F,trymax=20)
# Dimensions: 2
# Stress: 0.1672539
stressplot(nmdsY1)
#Base Plot and title
nmds.plot.Y1 <- ordiplot(nmdsY1,display="sites")
title(main="nMDS of Relative Abundances - Year 1")
text(1,1.5,"2D Stress: 0.17", cex=0.9)

#Month
ordihull(nmds.plot.Y1,groups=met1$Month,draw="lines",col=c("gray34","goldenrod2","green3","cadetblue2","dodgerbl
ue2",

"mediumpurple2","hotpink","tan","sienna","purple4"))
points(nmds.plot.Y1,"sites", pch=19, col= "gray34", select = met1$Month == "3")
points(nmds.plot.Y1,"sites", pch=19, col= "goldenrod2", select = met1$Month == "4")
points(nmds.plot.Y1,"sites", pch=19, col= "green3", select = met1$Month == "5")
points(nmds.plot.Y1,"sites", pch=19, col= "cadetblue2", select = met1$Month == "6")
points(nmds.plot.Y1,"sites", pch=19, col= "dodgerblue2", select = met1$Month == "7")
points(nmds.plot.Y1,"sites", pch=19, col= "mediumpurple2", select = met1$Month == "8")
points(nmds.plot.Y1,"sites", pch=19, col= "hotpink", select = met1$Month == "9")
points(nmds.plot.Y1,"sites", pch=19, col= "tan", select = met1$Month == "10")
points(nmds.plot.Y1,"sites", pch=19, col= "sienna", select = met1$Month == "11")
points(nmds.plot.Y1,"sites", pch=19, col= "purple4", select = met1$Month == "12")
text(1,1.5,"2D Stress: 0.17", cex=0.9)
legend("topleft",legend= c("3","4","5", "6","7","8","9","10","11","12"),
       title = "Month",ncol=2, col=c("gray34","goldenrod2","green3","cadetblue2",
                                     "dodgerblue2","mediumpurple2","hotpink",
                                     "tan","sienna","purple4"),
                                                                                pch=19, cex=0.92)
title(main="nMDS of Relative Abundances by Month - Year 1")


#Season
nmds.plot.Y1 <- ordiplot(nmdsY1,display="sites")
ordihull(nmds.plot.Y1,groups=met1$Season,draw="lines",col = c("sienna4","royalblue3"))
points(nmds.plot.Y1,"sites", pch=19, col= "sienna4", select = met1$Season == "dry")
points(nmds.plot.Y1,"sites", pch=19, col= "royalblue3", select = met1$Season == "wet")
text(1,1.5,"2D Stress: 0.17", cex=0.9)
legend("topleft",legend= c("dry","wet"), title = "Season",
       col=c("sienna4","royalblue3"), pch=19, cex=0.92)
title(main="nMDS of Relative Abundances by Season - Year 1")

#Zone
nmds.plot.Y1 <- ordiplot(nmdsY1,display="sites")
ordihull(nmds.plot.Y1,groups=met1$Zone,draw="lines",col =
c("palegreen3","wheat4","cornflowerblue","violetred2"))
points(nmds.plot.Y1,"sites", pch=19, col= "palegreen3", select = met1$Zone == "Inflow")
points(nmds.plot.Y1,"sites", pch=19, col= "wheat4", select = met1$Zone == "Nearshore")
points(nmds.plot.Y1,"sites", pch=19, col= "cornflowerblue", select = met1$Zone == "Pelagic")
points(nmds.plot.Y1,"sites", pch=19, col= "violetred2", select = met1$Zone == "S79")
```

```
text(1,1.5,"2D Stress: 0.17", cex=0.9)
legend("topleft",legend= c("Inflow","Nearshore","Pelagic", "S79"),
       title = "Zone",col=c("palegreen3","wheat4","mediumblue","violetred2"),
       pch=19, cex=0.92)
title(main="nMDS of Relative Abundances by Zone - Year 1")

#Station
nmds.plot.Y1 <- ordiplot(nmdsY1,display="sites")
ordihull(nmds.plot.Y1,groups=met1$Station,draw="lines",col=c("#A6CEE3","#579CC7","#3688AD",
                                                             "#8BC395","#89CB6C",
                                                             "#40A635","#919D5F",
                                                             "#F99392","#EB494A",
                                                             "#E83C2D","#F79C5D",
                                                             "#FDA746","#FE8205",
                                                             "#E39970", "#BFA5CF",
                                                             "#8861AC","#917099",
                                                             "#E7E099","#DEB969",
                                                             "#B15928"))
points(nmds.plot.Y1,"sites", pch=19, col= "#A6CEE3", select = met1$Station == "CLV10A")
points(nmds.plot.Y1,"sites", pch=19, col= "#579CC7", select = met1$Station == "KISSR0.0")
points(nmds.plot.Y1,"sites", pch=19, col= "#3688AD", select = met1$Station == "L001")
points(nmds.plot.Y1,"sites", pch=19, col= "#8BC395", select = met1$Station == "L004")
points(nmds.plot.Y1,"sites", pch=19, col= "#89CB6C", select = met1$Station == "L005")
points(nmds.plot.Y1,"sites", pch=19, col= "#40A635", select = met1$Station == "L006")
points(nmds.plot.Y1,"sites", pch=19, col= "#919D5F", select = met1$Station == "L007")
points(nmds.plot.Y1,"sites", pch=19, col= "#F99392", select = met1$Station == "L008")
points(nmds.plot.Y1,"sites", pch=19, col= "#EB494A", select = met1$Station == "LZ2")
points(nmds.plot.Y1,"sites", pch=19, col= "#E83C2D", select = met1$Station == "LZ25A")
points(nmds.plot.Y1,"sites", pch=19, col= "#F79C5D", select = met1$Station == "LZ30")
points(nmds.plot.Y1,"sites", pch=19, col= "#FDA746", select = met1$Station == "LZ40")
points(nmds.plot.Y1,"sites", pch=19, col= "#FE8205", select = met1$Station == "PALMOUT")
points(nmds.plot.Y1,"sites", pch=19, col= "#E39970", select = met1$Station == "PELBAY3")
points(nmds.plot.Y1,"sites", pch=19, col= "#BFA5CF", select = met1$Station == "POLE3S")
points(nmds.plot.Y1,"sites", pch=19, col= "#8861AC", select = met1$Station == "POLESOUT")
points(nmds.plot.Y1,"sites", pch=19, col= "#917099", select = met1$Station == "RITTAE2")
points(nmds.plot.Y1,"sites", pch=19, col= "#E7E099", select = met1$Station == "S308")
points(nmds.plot.Y1,"sites", pch=19, col= "#DEB969", select = met1$Station == "S77")
points(nmds.plot.Y1,"sites", pch=19, col= "#B15928", select = met1$Station == "S79")
text(1,1.5,"2D Stress: 0.17", cex=0.9)
legend("topleft",legend= c("CLV10A","KISSR0.0","L001","L004","L005","L006","L007",
                           "L008","LZ2","LZ25A","LZ30","LZ40","PALMOUT","PELBAY3",
                           "POLE3S","POLESOUT","RITTAE2","S308","S77","S79"),title = "Station",
       col=c("#A6CEE3","#579CC7","#3688AD","#8BC395","#89CB6C","#40A635","#919D5F",
                       "#F99392","#EB494A","#E83C2D","#F79C5D","#FDA746","#FE8205",
                       "#E39970","#BFA5CF","#8861AC","#917099","#E7E099","#DEB969",
                       "#B15928"),ncol=2,pch=19, cex=0.72)
title(main="nMDS of Relative Abundances by Station - Year 1")

### Year 2
nmdsY2 <- metaMDS(ra.bc.d.Y2,k=2,autotransform = F,trymax=20)
# Dimensions: 2
# Stress: 0.1773041
stressplot(nmdsY2)
#Base Plot and title
nmds.plot.Y2 <- ordiplot(nmdsY2,display="sites")
title(main="nMDS of Relative Abundances - Year 2")


#Month
nmds.plot.Y2 <- ordiplot(nmdsY2,display="sites")
ordihull(nmds.plot.Y2,groups=met2$Month,draw="lines",col=c("firebrick2","darkorange1","gray34","goldenrod2","gre
en3","cadetblue2","dodgerblue2",

"mediumpurple2","hotpink","tan","sienna","purple4"))
points(nmds.plot.Y2,"sites", pch=19, col= "firebrick2", select = met2$Month == "1")
points(nmds.plot.Y2,"sites", pch=19, col= "darkorange1", select = met2$Month == "2")
points(nmds.plot.Y2,"sites", pch=19, col= "gray34", select = met2$Month == "3")
points(nmds.plot.Y2,"sites", pch=19, col= "goldenrod2", select = met2$Month == "4")
points(nmds.plot.Y2,"sites", pch=19, col= "green3", select = met2$Month == "5")
points(nmds.plot.Y2,"sites", pch=19, col= "cadetblue2", select = met2$Month == "6")
points(nmds.plot.Y2,"sites", pch=19, col= "dodgerblue2", select = met2$Month == "7")
points(nmds.plot.Y2,"sites", pch=19, col= "mediumpurple2", select = met2$Month == "8")
points(nmds.plot.Y2,"sites", pch=19, col= "hotpink", select = met2$Month == "9")
points(nmds.plot.Y2,"sites", pch=19, col= "tan", select = met2$Month == "10")
points(nmds.plot.Y2,"sites", pch=19, col= "sienna", select = met2$Month == "11")
points(nmds.plot.Y2,"sites", pch=19, col= "purple4", select = met2$Month == "12")
```

```
text(1.8,1.4,"2D Stress: 0.18", cex=0.9)
legend("topleft",legend= c("1","2","3","4","5", "6","7","8","9","10","11","12"), title = "Month",
       col=c("firebrick2","darkorange1","gray34","goldenrod2","green3","cadetblue2","dodgerblue2",
                          "mediumpurple2","hotpink","tan","sienna","purple4"), pch=19,ncol=2, cex=0.88)
title(main="nMDS of Relative Abundances by Month - Year 2")

#Season
nmds.plot.Y2 <- ordiplot(nmdsY2,display="sites")
ordihull(nmds.plot.Y2,groups=met2$Season,draw="lines",col = c("sienna4","royalblue3"))
points(nmds.plot.Y2,"sites", pch=19, col= "sienna4", select = met2$Season == "dry")
points(nmds.plot.Y2,"sites", pch=19, col= "royalblue3", select = met2$Season == "wet")
text(1.8,1.4,"2D Stress: 0.18", cex=0.9)
legend("topleft",legend= c("dry","wet"),  title = "Season",col=c("sienna4","royalblue3"),  pch=19, cex=0.92)
title(main="nMDS of Relative Abundances by Season - Year 2")

#Zone
nmds.plot.Y2 <- ordiplot(nmdsY2,display="sites")
ordihull(nmds.plot.Y2,groups=met2$Zone,draw="lines",col =
c("palegreen3","wheat4","cornflowerblue","violetred2"))
points(nmds.plot.Y2,"sites", pch=19, col= "palegreen3", select = met2$Zone == "Inflow")
points(nmds.plot.Y2,"sites", pch=19, col= "wheat4", select = met2$Zone == "Nearshore")
points(nmds.plot.Y2,"sites", pch=19, col= "cornflowerblue", select = met2$Zone == "Pelagic")
points(nmds.plot.Y2,"sites", pch=19, col= "violetred2", select = met2$Zone == "S79")
text(1.8,1.4,"2D Stress: 0.18", cex=0.9)
legend("topleft",legend= c("Inflow","Nearshore","Pelagic", "S79"), title = "Zone",
       col=c("palegreen3","wheat4","mediumblue","violetred2"), pch=19, cex=0.92)
title(main="nMDS of Relative Abundances by Zone - Year 2")

#Station
nmds.plot.Y2 <- ordiplot(nmdsY2,display="sites")
ordihull(nmds.plot.Y2,groups=met2$Station,draw="lines",col=c("#A6CEE3","#579CC7","#3688AD","#8BC395","#89CB6C","
#40A635","#919D5F","#F99392","#EB494A","#E83C2D","#F79C5D","#FDA746","#FE8205","#E39970","#BFA5CF","#8861AC","#9
17099","#E7E099","#DEB969","#B15928"))
points(nmds.plot.Y2,"sites", pch=19, col= "#A6CEE3", select = met2$Station == "CLV10A")
points(nmds.plot.Y2,"sites", pch=19, col= "#579CC7", select = met2$Station == "KISSR0.0")
points(nmds.plot.Y2,"sites", pch=19, col= "#3688AD", select = met2$Station == "L001")
points(nmds.plot.Y2,"sites", pch=19, col= "#8BC395", select = met2$Station == "L004")
points(nmds.plot.Y2,"sites", pch=19, col= "#89CB6C", select = met2$Station == "L005")
points(nmds.plot.Y2,"sites", pch=19, col= "#40A635", select = met2$Station == "L006")
points(nmds.plot.Y2,"sites", pch=19, col= "#919D5F", select = met2$Station == "L007")
points(nmds.plot.Y2,"sites", pch=19, col= "#F99392", select = met2$Station == "L008")
points(nmds.plot.Y2,"sites", pch=19, col= "#EB494A", select = met2$Station == "LZ2")
points(nmds.plot.Y2,"sites", pch=19, col= "#E83C2D", select = met2$Station == "LZ25A")
points(nmds.plot.Y2,"sites", pch=19, col= "#F79C5D", select = met2$Station == "LZ30")
points(nmds.plot.Y2,"sites", pch=19, col= "#FDA746", select = met2$Station == "LZ40")
points(nmds.plot.Y2,"sites", pch=19, col= "#FE8205", select = met2$Station == "PALMOUT")
points(nmds.plot.Y2,"sites", pch=19, col= "#E39970", select = met2$Station == "PELBAY3")
points(nmds.plot.Y2,"sites", pch=19, col= "#BFA5CF", select = met2$Station == "POLE3S")
points(nmds.plot.Y2,"sites", pch=19, col= "#8861AC", select = met2$Station == "POLESOUT")
points(nmds.plot.Y2,"sites", pch=19, col= "#917099", select = met2$Station == "RITTAE2")
points(nmds.plot.Y2,"sites", pch=19, col= "#E7E099", select = met2$Station == "S308")
points(nmds.plot.Y2,"sites", pch=19, col= "#DEB969", select = met2$Station == "S77")
points(nmds.plot.Y2,"sites", pch=19, col= "#B15928", select = met2$Station == "S79")
text(1.8,1.4,"2D Stress: 0.18", cex=0.9)
legend("topleft",legend= c("CLV10A","KISSR0.0","L001","L004","L005",
                           "L006","L007","L008","LZ2","LZ25A","LZ30","LZ40",
                           "PALMOUT","PELBAY3","POLE3S","POLESOUT","RITTAE2",
                           "S308","S77","S79"),title = "Station",
       col=c("#A6CEE3","#579CC7","#3688AD","#8BC395","#89CB6C","#40A635","#919D5F",
                     "#F99392","#EB494A","#E83C2D","#F79C5D","#FDA746","#FE8205",
                     "#E39970", "#BFA5CF","#8861AC","#917099","#E7E099","#DEB969",
                     "#B15928"),pch=19, ncol=2,cex=0.64)
title(main="nMDS of Relative Abundances by Station - Year 2")

### Year 3
nmdsY3 <- metaMDS(ra.bc.d.Y3,k=2,autotransform = F,trymax=20)
# Dimensions: 2
# Stress: 0.1471427
stressplot(nmdsY3)
#Base Plot and title
nmds.plot.Y3 <- ordiplot(nmdsY3,display="sites")
title(main="nMDS of Relative Abundances - Year 3")



#Month
```

```
nmds.plot.Y3 <- ordiplot(nmdsY3,display="sites")
ordihull(nmds.plot.Y3,groups=met3$Month,draw="lines",col=c("firebrick2","darkorange1","gray34","goldenrod2","gre
en3","cadetblue2","dodgerblue2","mediumpurple2","hotpink","tan"))
points(nmds.plot.Y3,"sites", pch=19, col= "firebrick2", select = met3$Month == "1")
points(nmds.plot.Y3,"sites", pch=19, col= "darkorange1", select = met3$Month == "2")
points(nmds.plot.Y3,"sites", pch=19, col= "gray34", select = met3$Month == "3")
points(nmds.plot.Y3,"sites", pch=19, col= "goldenrod2", select = met3$Month == "4")
points(nmds.plot.Y3,"sites", pch=19, col= "green3", select = met3$Month == "5")
points(nmds.plot.Y3,"sites", pch=19, col= "cadetblue2", select = met3$Month == "6")
points(nmds.plot.Y3,"sites", pch=19, col= "dodgerblue2", select = met3$Month == "7")
points(nmds.plot.Y3,"sites", pch=19, col= "mediumpurple2", select = met3$Month == "8")
points(nmds.plot.Y3,"sites", pch=19, col= "hotpink", select = met3$Month == "9")
points(nmds.plot.Y3,"sites", pch=19, col= "tan", select = met3$Month == "10")
text(-0.85,1.3,"2D Stress: 0.15", cex=0.9)
legend("topright",legend= c("1","2","3","4","5", "6","7","8","9","10"),
       title = "Month",

col=c("firebrick2","darkorange1","gray34","goldenrod2","green3","cadetblue2","dodgerblue2","mediumpurple2","hotp
ink","tan"),
       pch=19, ncol=2,cex=1)
title(main="nMDS of Relative Abundances by Month - Year 3")

#Season
nmds.plot.Y3 <- ordiplot(nmdsY3,display="sites")
ordihull(nmds.plot.Y3,groups=met3$Season,draw="lines",col = c("sienna4","royalblue3"))
points(nmds.plot.Y3,"sites", pch=19, col= "sienna4", select = met3$Season == "dry")
points(nmds.plot.Y3,"sites", pch=19, col= "royalblue3", select = met3$Season == "wet")
text(-0.85,1.3,"2D Stress: 0.15", cex=0.9)
legend("topright",legend= c("dry","wet"), title = "Season",col=c("sienna4","royalblue3"),pch=19, cex=1.4)
title(main="nMDS of Relative Abundances by Season - Year 3")

#Zone
nmds.plot.Y3 <- ordiplot(nmdsY3,display="sites")
ordihull(nmds.plot.Y3,groups=met3$Zone,draw="lines",col =
c("palegreen3","wheat4","cornflowerblue","violetred2"))
points(nmds.plot.Y3,"sites", pch=19, col= "palegreen3", select = met3$Zone == "Inflow")
points(nmds.plot.Y3,"sites", pch=19, col= "wheat4", select = met3$Zone == "Nearshore")
points(nmds.plot.Y3,"sites", pch=19, col= "cornflowerblue", select = met3$Zone == "Pelagic")
points(nmds.plot.Y3,"sites", pch=19, col= "violetred2", select = met3$Zone == "S79")
text(-0.85,1.3,"2D Stress: 0.15", cex=0.9)
legend("topright",legend= c("Inflow","Nearshore","Pelagic", "S79"),title = "Zone",
       col=c("palegreen3","wheat4","mediumblue","violetred2"),pch=19, cex=0.9)
title(main="nMDS of Relative Abundances by Zone - Year 3")

#Station
nmds.plot.Y3 <- ordiplot(nmdsY3,display="sites")
ordihull(nmds.plot.Y3,groups=met3$Station,draw="lines",col=c("#A6CEE3","#579CC7","#3688AD","#8BC395","#89CB6C","
#40A635","#919D5F","#F99392","#EB494A","#E83C2D","#F79C5D","#FDA746","#FE8205","#E39970",
"#BFA5CF","#8861AC","#917099","#E7E099","#DEB969","#B15928"))
points(nmds.plot.Y3,"sites", pch=19, col= "#A6CEE3", select = met3$Station == "CLV10A")
points(nmds.plot.Y3,"sites", pch=19, col= "#579CC7", select = met3$Station == "KISSR0.0")
points(nmds.plot.Y3,"sites", pch=19, col= "#3688AD", select = met3$Station == "L001")
points(nmds.plot.Y3,"sites", pch=19, col= "#8BC395", select = met3$Station == "L004")
points(nmds.plot.Y3,"sites", pch=19, col= "#89CB6C", select = met3$Station == "L005")
points(nmds.plot.Y3,"sites", pch=19, col= "#40A635", select = met3$Station == "L006")
points(nmds.plot.Y3,"sites", pch=19, col= "#919D5F", select = met3$Station == "L007")
points(nmds.plot.Y3,"sites", pch=19, col= "#F99392", select = met3$Station == "L008")
points(nmds.plot.Y3,"sites", pch=19, col= "#EB494A", select = met3$Station == "LZ2")
points(nmds.plot.Y3,"sites", pch=19, col= "#E83C2D", select = met3$Station == "LZ25A")
points(nmds.plot.Y3,"sites", pch=19, col= "#F79C5D", select = met3$Station == "LZ30")
points(nmds.plot.Y3,"sites", pch=19, col= "#FDA746", select = met3$Station == "LZ40")
points(nmds.plot.Y3,"sites", pch=19, col= "#FE8205", select = met3$Station == "PALMOUT")
points(nmds.plot.Y3,"sites", pch=19, col= "#E39970", select = met3$Station == "PELBAY3")
points(nmds.plot.Y3,"sites", pch=19, col= "#BFA5CF", select = met3$Station == "POLE3S")
points(nmds.plot.Y3,"sites", pch=19, col= "#8861AC", select = met3$Station == "POLESOUT")
points(nmds.plot.Y3,"sites", pch=19, col= "#917099", select = met3$Station == "RITTAE2")
points(nmds.plot.Y3,"sites", pch=19, col= "#E7E099", select = met3$Station == "S308")
points(nmds.plot.Y3,"sites", pch=19, col= "#DEB969", select = met3$Station == "S77")
points(nmds.plot.Y3,"sites", pch=19, col= "#B15928", select = met3$Station == "S79")
text(-0.85,1.3,"2D Stress: 0.15", cex=0.9)
legend("topright",legend= c("CLV10A","KISSR0.0","L001","L004","L005","L006",
                            "L007","L008","LZ2","LZ25A","LZ30","LZ40","PALMOUT",
                            "PELBAY3","POLE3S","POLESOUT","RITTAE2","S308","S77","S79"),
       title = "Station",col=c("#A6CEE3","#579CC7","#3688AD","#8BC395","#89CB6C",
                                    "#40A635","#919D5F","#F99392","#EB494A",
                                    "#E83C2D","#F79C5D","#FDA746","#FE8205",
```

```
                                        "#E39970","#BFA5CF","#8861AC","#917099",
                                        "#E7E099","#DEB969","#B15928"),
                                        ncol=2,pch=19, cex=0.8)
title(main="nMDS of Relative Abundances by Station - Year 3")

###### Beta Diversity Stat. Analyses for each year ######
##betadisper calculates dispersion (variances) within each group

#Loading in metadata
metadata <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
#Subsetting metadata table by year
met1 <- metadata[grep("_19$", rownames(metadata)),]
met2 <- metadata[grep("_20$", rownames(metadata)),]
met3 <- metadata[grep("_21$", rownames(metadata)),]

#Year 1
dis.Z1 <-betadisper(ra.bc.d.Y1,met1$Zone)
dis.S1 <-betadisper(ra.bc.d.Y1,met1$Season)
dis.St1 <-betadisper(ra.bc.d.Y1,met1$Station)
dis.M1 <-betadisper(ra.bc.d.Y1,met1$Month)
#Year 2
dis.Z2 <-betadisper(ra.bc.d.Y2,met2$Zone)
dis.S2 <-betadisper(ra.bc.d.Y2,met2$Season)
dis.St2 <-betadisper(ra.bc.d.Y2,met2$Station)
dis.M2 <-betadisper(ra.bc.d.Y2,met2$Month)
#Year 3
dis.Z3 <-betadisper(ra.bc.d.Y3,met3$Zone)
dis.S3 <-betadisper(ra.bc.d.Y3,met3$Season)
dis.St3 <-betadisper(ra.bc.d.Y3,met3$Station)
dis.M3 <-betadisper(ra.bc.d.Y3,met3$Month)

##permutest determines if the variances differ by groups (If differences are SIGNIFICANT - use ANOSIM
##                                                        if not use PERMANOVA (adonis))
#Year 1
permutest(dis.Z1, pairwise=TRUE, permutations=999)
#            Df Sum Sq  Mean Sq      F N.Perm Pr(>F)
# Groups      3 0.0448 0.014934 1.4207    999  0.238 -> NOT SIGNIFICANT
# Residuals 153 1.6082 0.010511
# ---

permutest(dis.S1, pairwise=TRUE, permutations=999)
#            Df  Sum Sq   Mean Sq      F N.Perm Pr(>F)
# Groups      1 0.00001 0.0000127 0.0013    999  0.968 -> NOT SIGNIFICANT
# Residuals 155 1.45375 0.0093790
# ---

permutest(dis.M1, pairwise=TRUE, permutations=999)
#            Df  Sum Sq   Mean Sq      F N.Perm Pr(>F)
# Groups      9 0.07056 0.0078398 0.7765    999  0.651 -> NOT SIGNIFICANT
# Residuals 147 1.48410 0.0100959
# ---

permutest(dis.St1, pairwise=TRUE, permutations=999)
#            Df  Sum Sq  Mean Sq      F N.Perm Pr(>F)
# Groups     19 0.28943 0.015233 1.1881    999  0.279 -> NOT SIGNIFICANT
# Residuals 137 1.75652 0.012821


## USE PERMANOVA/adonis!!


##PERMANOVA - determining if the differences between two or more groups are significant
adonis2(ra.bc.d.Y1~met1$Station, permutations = 999)
#               Df SumOfSqs      R2      F Pr(>F)
# met1$Station  19   10.764 0.23512 2.2165  0.001 ***
# Residual     137   35.016 0.76488
# Total        156   45.779 1.00000
# ___
#Pairwise perMANOVA to see what sites have the differences
Y1Stat <- pairwise.perm.manova(ra.bc.d.Y1, met1$Station,nperm = 999,p.method = "fdr")
# Get p-values in a dataframe
Y1Stp <- Y1Stat$p.value
# Convert the data to a table
m <- as.data.frame(Y1Stp)
# Plot p-values
library(gplots)
```

```
ggballoonplot(m,
            main ="p.values",
            xlab ="",
            ylab="",
            label = T, label.size=0.6, #adds the p value number to the plot
            show.margins = F)
ggballoonplot(
  m, main = "Year 1 by Station - p-value comparison",
  size = "value",
  size.range = c(1, 10),
  shape = 21,
  color = "black",
  fill = "value",
  show.label = F, legend = ggplot2::lims(0.05,0.8),
  font.label = list(size = 6, color = "black"),
  rotate.x.text = TRUE,
  ggtheme = theme_minimal())
#_____________________________________

adonis2(ra.bc.d.Y1~met1$Season, permutations = 999)
#              Df SumOfSqs      R2      F Pr(>F)
# met1$Season   1    0.244 0.00533 0.8308  0.672 -> NOT SIGNIFICANT
# Residual    155   45.535 0.99467
# Total       156   45.779 1.00000

adonis2(ra.bc.d.Y1~met1$Zone, permutations = 999)
#            Df SumOfSqs      R2      F Pr(>F)
# met1$Zone   3    1.791 0.03911 2.0759  0.001 ***
# Residual  153   43.989 0.96089
# Total     156   45.779 1.00000
# ___
#PerMANOVA to see what sites have the differences
Y1Zone <- pairwise.perm.manova(ra.bc.d.Y1, met1$Zone,nperm = 999,p.method = "fdr")
# Significant differences found between all zones

adonis2(ra.bc.d.Y1~met1$Month, permutations = 999)
#             Df SumOfSqs      R2      F Pr(>F)
# met1$Month   1    0.157 0.00342 0.5322  0.994 -> NOT SIGNIFICANT
# Residual   155   45.622 0.99658
# Total      156   45.779 1.00000


#Year 2
permutest(dis.Z2, pairwise=TRUE, permutations=999)
#            Df  Sum Sq  Mean Sq     F N.Perm Pr(>F)
# Groups      3 0.17468 0.058226 6.558    999  0.002 **
# Residuals 206 1.82900 0.008879
# ---
#
# Pairwise comparisons:
#   (Observed p-value below diagonal, permuted p-value above diagonal)
#              Inflow  Nearshore    Pelagic   S79
# Inflow               2.2100e-01 3.1000e-02 0.018
# Nearshore 2.2085e-01            3.6200e-01 0.002
# Pelagic   1.9483e-02 3.3873e-01            0.001
# S79       2.3672e-02 8.2715e-04 3.5696e-05
permutest(dis.S2, pairwise=TRUE, permutations=999)
#            Df  Sum Sq   Mean Sq     F N.Perm Pr(>F)
# Groups      1 0.00219 0.0021948 0.258    999  0.614
# Residuals 208 1.76932 0.0085063
# ---
permutest(dis.M2, pairwise=TRUE, permutations=999)
#            Df  Sum Sq   Mean Sq      F N.Perm Pr(>F)
# Groups     11 0.05232 0.0047561 0.5497    999  0.858
# Residuals 198 1.71297 0.0086514
permutest(dis.St2, pairwise=TRUE, permutations=999)
#            Df  Sum Sq  Mean Sq      F N.Perm Pr(>F)
# Groups     19 0.67528 0.035541 2.8946    999  0.001 ***
# Residuals 190 2.33290 0.012278

## USE ANOSIM FOR ZONE AND STATION, USE PERMANOVA FOR SEASON AND MONTH!!


##ANOSIM - determining if the differences between two or more groups are significant
anosim(ra.bc.d.Y2,met2$Zone, permutations = 999, distance = "bray")
# ANOSIM statistic R: 0.01148
```

```
# Significance: 0.314 -> NOT SIGINFICANT

anosim(ra.bc.d.Y2,met2$Station, permutations = 999, distance = "bray")
# ANOSIM statistic R: 0.2535
# Significance: 0.001
Y2Stat <- pairwise.perm.manova(ra.bc.d.Y2, met2$Station,nperm = 999,p.method = "fdr")

##PERMANOVA
adonis2(ra.bc.d.Y2~met2$Month, permutations = 999)
#             Df SumOfSqs    R2      F Pr(>F)
# met2$Month   1    0.184 0.003 0.6265  0.945 -> NOT SIGINFICANT
# Residual   208   61.122 0.997
# Total      209   61.306 1.000
adonis2(ra.bc.d.Y2~met2$Season, permutations = 999)
#              Df SumOfSqs      R2      F Pr(>F)
# met2$Season   1    0.172 0.00281 0.5857  0.977 -> NOT SIGINFICANT
# Residual    208   61.134 0.99719
# Total       209   61.306 1.00000

#Year 3
permutest(dis.Z3, pairwise=TRUE, permutations=999)
#            Df  Sum Sq  Mean Sq      F N.Perm Pr(>F)
# Groups      3 0.18912 0.063039 5.1907    999  0.007 **
# Residuals 170 2.06459 0.012145
# ---
# Pairwise comparisons:
#   (Observed p-value below diagonal, permuted p-value above diagonal)
#              Inflow  Nearshore    Pelagic   S79
# Inflow               0.01000000 0.16800000 0.463
# Nearshore 0.01207560            0.00100000 0.068
# Pelagic   0.15407191 0.00012197            0.975
# S79       0.46792457 0.05697194 0.96831209
permutest(dis.S3, pairwise=TRUE, permutations=999)
#            Df  Sum Sq  Mean Sq      F N.Perm Pr(>F)
# Groups      1 0.03421 0.034209 3.3793    999  0.074 . -> NOT SIGNIFICANT
# Residuals 172 1.74117 0.010123
# ---
permutest(dis.M3, pairwise=TRUE, permutations=999)
#            Df  Sum Sq   Mean Sq      F N.Perm Pr(>F)
# Groups      9 0.06587 0.0073193 0.7267    999  0.721 -> NOT SIGNIFICANT
# Residuals 164 1.65174 0.0100716
permutest(dis.St3, pairwise=TRUE, permutations=999)
#            Df  Sum Sq  Mean Sq     F N.Perm Pr(>F)
# Groups     19 0.70017 0.036851 3.009    999  0.003 **
# Residuals 154 1.88604 0.012247

## USE ANOSIM FOR ZONE AND STATION, USE PERMANOVA FOR SEASON AND MONTH!!


##ANOSIM - determining if the differences between two or more groups are significant
anosim(ra.bc.d.Y3,met3$Zone, permutations = 999, distance = "bray")
# ANOSIM statistic R: 0.4239
# Significance: 0.001
Y3Zone <- pairwise.perm.manova(ra.bc.d.Y3, met3$Zone,nperm = 999,p.method = "fdr")
# Significant differences found between all zones

anosim(ra.bc.d.Y3,met3$Station, permutations = 999, distance = "bray")
# ANOSIM statistic R: 0.2877
# Significance: 0.001
Y3Stat <- pairwise.perm.manova(ra.bc.d.Y3, met3$Station,nperm = 999,p.method = "fdr")

##PERMANOVA
adonis2(ra.bc.d.Y3~met3$Season, permutations = 999)
#              Df SumOfSqs      R2      F Pr(>F)
# met3$Season   1    0.265 0.00598 1.0348   0.33 -> NOT SIGNIFICANT
# Residual    172   44.122 0.99402
# Total       173   44.387 1.00000
adonis2(ra.bc.d.Y3~met3$Month, permutations = 999)
#             Df SumOfSqs      R2      F Pr(>F)
# met3$Month   1    0.193 0.00434 0.7504  0.735 -> NOT SIGNIFICANT
# Residual   172   44.195 0.99566
# Total      173   44.387 1.00000


###### Beta Diversity - Stat. Analyses - ALL YEARS TOGETHER ######
set.seed(1998)
```

```
##betadisper calculates dispersion (variances) within each group
#values should be non-significant in order to use PERMANOVA
dis.Zone <-betadisper(ra.bc.dist,metadata$Zone)
dis.Season <-betadisper(ra.bc.dist,metadata$Season)
dis.Year <-betadisper(ra.bc.dist,metadata$Year)
dis.Station <-betadisper(ra.bc.dist,metadata$Station)
dis.Month <-betadisper(ra.bc.dist,metadata$Month)

##permutest determines if the variances differ by groups (If differences are SIGNIFICANT - use ANOSIM
##                                                        if not use PERMANOVA (adonis))
permutest(dis.Zone, pairwise=TRUE, permutations=999)
#            Df Sum Sq  Mean Sq      F N.Perm Pr(>F)
# Groups      3 0.1605 0.053487 5.3955    999  0.001 ***
# Residuals 537 5.3235 0.009913
# ---
# Pairwise comparisons:
# (Observed p-value below diagonal, permuted p-value above diagonal)
#              Inflow Nearshore   Pelagic   S79
# Inflow              0.0030000 0.5910000 0.051
# Nearshore 0.0025931           0.0010000 0.713
# Pelagic   0.5842551 0.0011149           0.057
# S79       0.0309406 0.7291803 0.0427081
permutest(dis.Season, pairwise=TRUE, permutations=999)
#            Df Sum Sq   Mean Sq      F N.Perm Pr(>F)
# Groups      1 0.0038 0.0037558 0.4045    999  0.532 -> NOT SIGNIFICANT
# Residuals 539 5.0041 0.0092840
# ---
permutest(dis.Year, pairwise=TRUE, permutations=999)
#            Df Sum Sq   Mean Sq      F N.Perm Pr(>F)
# Groups      2 0.0042 0.0021079 0.2258    999  0.809 -> NOT SIGNIFICANT
# Residuals 538 5.0226 0.0093358
# ---
permutest(dis.Station, pairwise=TRUE, permutations=999) #look at pairwise in R (very large)
#            Df Sum Sq  Mean Sq      F N.Perm Pr(>F)
# Groups     19 1.0197 0.053670 5.1682    999  0.001 ***
# Residuals 521 5.4105 0.010385
permutest(dis.Month, pairwise=TRUE, permutations=999)
#            Df Sum Sq   Mean Sq      F N.Perm Pr(>F)
# Groups     11 0.0580 0.0052772 0.5639    999  0.851 -> NOT SIGNIFICANT
# Residuals 529 4.9508 0.0093589
# ---


## USE ANOSIM FOR ZONE AND STATION AND USE PERMANOVA FOR SEASON, YEAR, AND MONTH


##ANOSIM - determining if the differences between two or more groups are significant.
## The ANOSIM statistic "R" compares the mean of ranked dissimilarities between groups to
## the mean of ranked dissimilarities within groups. An R value close to "1" suggests
## dissimilarity between groups while an R value close to "0" suggests an even distribution of
## high and low ranks within and between groups"
## the higher the R value, the more dissimilar your groups are in terms of microbial community composition.

anosim(ra.bc.dist, metadata$Zone, permutations = 999, distance = "bray")
# ANOSIM statistic R: 0.01493
# Significance: 0.205 -> NOT SIGNIFICANT
anosim(ra.bc.dist, metadata$Station, permutations = 999, distance = "bray")
# ANOSIM statistic R: 0.1967
# Significance: 0.001

##PERMANOVA
adonis2(ra.bc.dist~metadata$Month, permutations = 999)
#                Df SumOfSqs      R2     F Pr(>F)
# metadata$Month   1    0.195 0.00127 0.683  0.909 -> NOT SIGNIFICANT
# Residual       539  154.113 0.99873
# Total          540  154.309 1.00000
adonis2(ra.bc.dist~metadata$Year, permutations = 999)
#               Df SumOfSqs      R2      F Pr(>F)
# metadata$Year   1    0.171 0.00111 0.5987  0.974 -> NOT SIGNIFICANT
# Residual      539  154.137 0.99889
# Total         540  154.309 1.00000
adonis2(ra.bc.dist~metadata$Season, permutations = 999)
#                 Df SumOfSqs      R2      F Pr(>F)
# metadata$Season   1    0.204 0.00132 0.7127  0.881 -> NOT SIGNIFICANT
# Residual        539  154.105 0.99868
```

```
# Total           540  154.309 1.00000

## USE MANTEL TEST FOR CONTINUOUS VARIABLES
##Mantel tests are correlation tests that determine the correlation between two
##matrices (rather than two variables). A significant Mantel test will tell you
##that the distances between samples in one matrix are correlated with the distances
##between samples in the other matrix. Therefore, as the distance between samples
##increases with respect to one matrix, the distances between the same samples also
##increases in the other matrix

#abundance dissim. matrix
dist.abund <- ra.bc.dist
#Microcystis/Bloom distance using euclidean
MA <- metadata$Microcystis.Abundance
CHL <- metadata$Chlorophyll.a
dist.MA <- dist(MA, method = "euclidean")
dist.CHL <- dist(CHL, method = "euclidean")

#Mantel test - Microcystis
mantel(dist.abund, dist.MA, method = "spearman", permutations = 999)
# Mantel statistic r: 0.008024
# Significance: 0.4 -> NOT SIGINIFCANT

#Mantel test -  Chlorophyll a
mantel(dist.abund, dist.CHL, method = "spearman", permutations = 999)
# Mantel statistic r: 0.01756
# Significance: 0.225 -> NOT SIGNIFICANT

##Plotting beta diversity against significant variables
#create vectors of matrices
cc <- as.vector(dist.CHL)
mm <- as.vector(dist.MA)
aa <- as.vector(dist.abund)
#new data frame with vectorized distance matrices
mat <- data.frame(cc,aa,mm)
#PLOT - Chlorophyll a
ggplot(mat, aes(y = aa, x = cc)) +
  geom_point(size = 2, alpha = 0.75, colour = "black",shape = 21) +
  labs(x = "Chlorophyll a (ug/L)", y = "Bray-Curtis Dissimilarity") +
  theme( axis.text.x = element_text(face = "bold",colour = "black", size = 12),
         axis.text.y = element_text(face = "bold", size = 11, colour = "black"),
         axis.title= element_text(face = "bold", size = 14, colour = "black"),
         panel.background = element_blank(),
         panel.border = element_rect(fill = NA, colour = "black"))
#PLOT - Microcystis
ggplot(mat, aes(y = aa, x = mm)) +
  geom_point(size = 2, alpha = 0.75, colour = "black",shape = 21) +
  labs(x = "Microcystis Relative Abundance", y = "Bray-Curtis Dissimilarity") +
  theme( axis.text.x = element_text(face = "bold",colour = "black", size = 12),
         axis.text.y = element_text(face = "bold", size = 11, colour = "black"),
         axis.title= element_text(face = "bold", size = 14, colour = "black"),
         panel.background = element_blank(),
         panel.border = element_rect(fill = NA, colour = "black"))



###### Venn Diagram of ASVs (Year, Zone, Season) ######
##Packages
library(eulerr)
library(microbiome)
library(microbiomeutilities)
#library(devtools) ##used to install microbiome utilities package
#devtools::install_github('microsud/microbiomeutilities') ## only run if need to install package



## Making phyloseq objects (WHOLE DATA SET)
asvdat <- as.data.frame(t(dat.01per)) #species has to be rows so the df was transformed
taxdat <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE, row.names = 1)
meta <- read.csv("Metadata-Diversity_BATCH.csv", header = TRUE, row.names = 1)
asvmat <- data.matrix(asvdat)
taxmat <- as.matrix(taxdat) # use as.matrix NOT as.data.matrix as the data will convert the data into numbers
ASV <- otu_table(asvmat, taxa_are_rows = TRUE)
TAX <- tax_table(taxmat)
META <- sample_data(meta)
pseq <- phyloseq(ASV,TAX,META)
```

```
# simple way to count number of samples in each group
table(meta(pseq)$Year, useNA = "always")
##
##   1       2        3    <NA>
##  157     210     174      0
table(meta(pseq)$Zone, useNA = "always")
##
# Inflow  Nearshore   Pelagic     S79       <NA>
#   107       131       281        22         0
table(meta(pseq)$Season, useNA = "always")
##
##  dry     wet   <NA>
##  247     294     0

#convert to relative abundance
transform <- microbiome::transform
pseq_rel <- transform(pseq, "compositional")

#Make a list of Years
years <- unique(as.character(meta(pseq_rel)$Year))
print(years)
# [1] "1" "2" "3"

#Make a list of Zones
zones <- unique(as.character(meta(pseq_rel)$Zone))
print(zones)
# [1] "Inflow"    "Pelagic"   "Nearshore" "S79"

#Make a list of Seasons
seasons <- unique(as.character(meta(pseq_rel)$Season))
print(seasons)
# [1] "dry" "wet"

#### YEAR
#Write a for loop to go through each of the years
#one by one and combine identified core taxa into a list.
list_core <- c() # an empty object to store information

for (n in years){ # for each variable n in Year
  #print(paste0("Identifying Core Taxa for ", n))

  ps.sub <- subset_samples(pseq_rel, Year == n) # Choose sample from Year by n

  core_m <- core_members(ps.sub, # ps.sub is phyloseq selected with only samples from g
                         detection = 0.001,
                         prevalence = 0.75)
  print(paste0("No. of core taxa in ", n, " : ", length(core_m))) # print core taxa identified in each year.
  list_core[[n]] <- core_m # add to a list core taxa for each group.
  #print(list_core)
}
# [1] "No. of core taxa in 1 : 14"
# [1] "No. of core taxa in 2 : 16"        WHOLE DATASET
# [1] "No. of core taxa in 3 : 32"


##Adding taxa information
print(list_core) # can see that its the ASV id w/ NO taxa info

taxa_names(pseq_rel)[1:5] #shows ASV id
# [1] "0885965c051f3034c0e28043193bc5d2" "51e00e866016fba8a19581249b811ec4"
# [3] "dfd3874c0e70ae177e8cdc4fb6961e7d" "ac879ef0bc703ee2637bc55f0ef97afc"
# [5] "41714fa1a258e8098d51d03a1e1b3304"

#format names and checking
pseq_rel_f <- format_to_besthit(pseq_rel)
taxa_names(pseq_rel_f)[1:5]

#rerun 'for' loop with better taxa information
for (n in years){
  ps.sub <- subset_samples(pseq_rel_f, Year == n)
  core_m <- core_members(ps.sub,
                         detection = 0.001,
                         prevalence = 0.75)
  print(paste0("No. of core taxa in ", n, " : ", length(core_m)))
```

```
  list_core[[n]] <- core_m
}
print(list_core) #shows ASV id with taxa information
#converting lists to dfs and saving as CSVs
Year1VennTaxa <- as.data.frame(list_core[["1"]])
Year2VennTaxa <- as.data.frame(list_core[["2"]])
Year3VennTaxa <- as.data.frame(list_core[["3"]])
write.csv(Year1VennTaxa, "CoreTaxaYear1-Venn.csv")
write.csv(Year2VennTaxa, "CoreTaxaYear2-Venn.csv")
write.csv(Year3VennTaxa, "CoreTaxaYear3-Venn.csv")

###Comparing venn diagram packages to see which to use (1.31.23)
##Plotting venn diagram using eulerr
plot(venn(list_core),fills = c("tomato3", "steelblue3", "springgreen3"))

#### ZONE
list_core <- c()
for (n in zones){
  ps.sub <- subset_samples(pseq_rel_f, Zone == n)
  core_m <- core_members(ps.sub,
                         detection = 0.001,
                         prevalence = 0.75)
  print(paste0("No. of core taxa in ", n, " : ", length(core_m)))
  list_core[[n]] <- core_m
}
# [1] "No. of core taxa in Inflow : 15"
# [1] "No. of core taxa in Pelagic : 45"
# [1] "No. of core taxa in Nearshore : 31"
# [1] "No. of core taxa in S79 : 33"

print(list_core) #shows ASV id with taxa information

#converting lists to dfs and saving as CSVs
InflowVennTaxa <- as.data.frame(list_core[["Inflow"]])
NearVennTaxa <- as.data.frame(list_core[["Nearshore"]])
PelVennTaxa <- as.data.frame(list_core[["Pelagic"]])
S79VennTaxa <- as.data.frame(list_core[["S79"]])
write.csv(InflowVennTaxa, "CoreTaxaInflow-Venn.csv")
write.csv(NearVennTaxa, "CoreTaxaNear-Venn.csv")
write.csv(PelVennTaxa, "CoreTaxaPelagic-Venn.csv")
write.csv(S79VennTaxa, "CoreTaxaS79-Venn.csv")

##Plotting venn diagram
plot(venn(list_core),fills = c("palegreen3","cornflowerblue","wheat4","violetred2"))

##Plotting venn diagram using VennDiagram
#downfall - creates a png file for the venn diagram BUT there is a workaround to view it in R
#         - does not allow for less than 4 variables
install.packages("VennDiagram")
# Helper function to display Venn diagram
display_venn <- function(x, ...){
  library(VennDiagram)
  grid.newpage()
  venn_object <- venn.diagram(x, filename = NULL, ...)
  grid.draw(venn_object)
}
display_venn(
  list_core,
  category.names = c("Inflow" , "Pelagic" , "Nearshore", "S79"),
  # Circles
  lwd = 2,
  lty = 'blank',
  fill = c("palegreen3","cornflowerblue","wheat4","violetred2"),
  # Numbers
  cex = 1,
  # Set names
  cat.cex = 1.26,
  cat.fontface = "bold",
  cat.default.pos = "outer",
  cat.dist = c(0.055, 0.055, 0.1, 0.1)
)

#### SEASON
list_core <- c()
for (n in seasons){
  ps.sub <- subset_samples(pseq_rel_f, Season == n)
```

```
  core_m <- core_members(ps.sub,
                         detection = 0.001,
                         prevalence = 0.75)
  print(paste0("No. of core taxa in ", n, " : ", length(core_m)))
  list_core[[n]] <- core_m
}
# [1] "No. of core taxa in dry : 29"
# [1] "No. of core taxa in wet : 17"

print(list_core) #shows ASV id with taxa information
#converting lists to dfs and saving as CSVs
DryVennTaxa <- as.data.frame(list_core[["dry"]])
WetVennTaxa <- as.data.frame(list_core[["wet"]])
write.csv(DryVennTaxa, "CoreTaxaDry-Venn.csv")
write.csv(WetVennTaxa, "CoreTaxaWet-Venn.csv")

##Plotting venn diagram
plot(venn(list_core),fills = c("lemonchiffon2","royalblue1"))

##Core line plots
# Determine core microbiota across various abundance/prevalence thresholds with
# the blanket analysis (Salonen et al. CMI, 2012) based on various signal and
# prevalences.

# With compositional (relative) abundances
det <- c(0, 0.1, 0.5, 2, 5, 20)/100
prevalences <- seq(.05, 1, .05)

plot_core(pseq_rel_f, prevalences = prevalences,
          detections = det, plot.type = "lineplot") +
  xlab("Relative Abundance (%)") +
  theme_bw()

##Core heatmaps
# This visualization method has been used for instance in Intestinal microbiome
# landscaping: Insight in community assemblage and implications for microbial
# modulation strategies. Shetty et al. FEMS Microbiology Reviews fuw045, 2017.

#Note that you can order the taxa on the heatmap with the order.taxa argument.

# Core with compositionals:
prevalences <- seq(.05, 1, .05)
detections <- round(10^seq(log10(1e-2), log10(.2), length = 10), 3)

#Deletes "ASV" from taxa_names, e.g. ASV1 --> 1
#taxa_names(ps.m3.rel) = taxa_names(ps.m3.rel) %>% str_replace("ASV", "")
# Also define gray color palette
gray <- gray(seq(0,1,length=5))

p1 <- plot_core(pseq_rel_f,
                plot.type = "heatmap",
                colours = gray,
                prevalences = prevalences,
                detections = detections, min.prevalence = .05) +
  xlab("Detection Threshold (Relative Abundance (%))")

p1 <- p1 + theme_bw() + ylab("ASVs")
p1


###### CCA Analysis - Overall and Year-to-Year ######
set.seed(1998)
#ALL YEARS
ccamodel <- cca(dat.ra~., metadata[,c(7:37)]) #run 1
# If VIF>10, the variable presents colinearity with another or other variables.
# In that case, delete the variable from initial dataset and redo the analysis.
# VIF = 1 for completely independent variables,and values above 10 or 20
# (depending on your taste) are regarded as highly multicollinear (dependent on others).

ccamodel <- cca(dat.ra~., metadata[,c(7:19,21:24,31,33)]) #run 2
anova.cca(finalmodel, by="terms")
#                         Df ChiSquare       F Pr(>F)
#   SecchiDiskDepth        1    0.1574  9.9667  0.001 ***
#   Silica                 1    0.0667  4.2218  0.001 ***
#   Sulfate                1    0.0552  3.4962  0.001 ***
```

```
#   Temperature             1    0.1163  7.3647  0.001 ***
#   Turbidity               1    0.1578  9.9912  0.001 ***
#   Alkalinity              1    0.1466  9.2843  0.001 ***
#   Ammonia                 1    0.1299  8.2251  0.006 **
#   Pheophytin.a            1    0.0678  4.2934  0.001 ***
#   Chlorophyll.a           1    0.1273  8.0613  0.001 ***
#   TotalDepth              1    0.0952  6.0274  0.001 ***
#   DissolvedOxygen         1    0.0584  3.6952  0.004 **
#   Nitrate.Nitrite         1    0.0654  4.1389  0.001 ***
#   Phosphate.Ortho         1    0.0530  3.3573  0.001 ***
#   pH                      1    0.0321  2.0298  0.014 *
#   Total.Nitrogen          1    0.0360  2.2828  0.004 **
#   TN.TP.ratio             1    0.0677  4.2882  0.001 ***
#   Microcystis.Abundance   1    0.1738 11.0048  0.001 ***
#   Microcystin.LA          1    0.0144  0.9097  0.383   -> REMOVE
#   Microcystin.LR          1    0.0243  1.5367  0.038 *
#   Residual              521    8.2273
# ---

ccamodel <- cca(dat.ra~., metadata[,c(7:19,21:24,33)]) #run 3
finalmodel<- ordistep(ccamodel, scope=formula(ccamodel))
vif.cca(finalmodel) ## everything is under 10
finalmodel ## Note that "Total Inertia" is the total variance in species (observations matrix) distributions.
## "Constrained Inertia" is the variance explained by the environmental variables (gradients matrix).
## The "Proportion" values represent the percentages of variance of species distributions explained
## by Constrained (environmental) and Unconstrained variables. Eigenvalues of constrained and
## unconstrained axes represent the amount of variance explained by each CCA axis (graphs usually
## present the first two constrained axes, so take a look at their values).
#Total Inertia = total variance in species (observdistributions
#Unconstrained Inertia = the variance explained by the environmental variables

#               Inertia Proportion Rank
# Total           9.872      1.000
# Constrained     1.629      0.165   18
# Unconstrained   8.243      0.835  522
# Inertia is scaled Chi-square

R2.adj.cca <- RsquareAdj(finalmodel)
# adjusting the R-squared value: The adjusted R2 tells you the percentage of
# variation explained by only the independent variables that actually affect
# the dependent variable
#indicates how well terms fit a curve or line, but adjusts for the number of terms in a model
R2.adj.cca
# r.squared: 0.173352
# adj.r.squared: 0.1446893

# Testing the significance of the CCA model
anova.cca(finalmodel) #should be significant
#           Df ChiSquare      F Pr(>F)
# Model     18    1.6290 5.7307  0.001 ***
# Residual 522    8.2434
# ---

# Testing the significance of terms (environmental variables)
anova.cca(finalmodel, by="terms")
#                          Df ChiSquare       F Pr(>F)
#   SecchiDiskDepth         1    0.1574  9.9663  0.001 ***
#   Silica                  1    0.0667  4.2216  0.001 ***
#   Sulfate                 1    0.0552  3.4961  0.001 ***
#   Temperature             1    0.1163  7.3644  0.001 ***
#   Turbidity               1    0.1578  9.9908  0.001 ***
#   Alkalinity              1    0.1466  9.2839  0.001 ***
#   Ammonia                 1    0.1299  8.2248  0.003 **
#   Pheophytin.a            1    0.0678  4.2932  0.001 ***
#   Chlorophyll.a           1    0.1273  8.0610  0.001 ***
#   TotalDepth              1    0.0952  6.0272  0.001 ***
#   DissolvedOxygen         1    0.0584  3.6951  0.002 **
#   Nitrate.Nitrite         1    0.0654  4.1387  0.001 ***
#   Phosphate.Ortho         1    0.0530  3.3572  0.002 **
#   pH                      1    0.0321  2.0297  0.008 **
#   Total.Nitrogen          1    0.0360  2.2827  0.003 **
#   TN.TP.ratio             1    0.0677  4.2880  0.001 ***
#   Microcystis.Abundance   1    0.1738 11.0044  0.001 ***
#   Microcystin.LR          1    0.0225  1.4273  0.064 .   -> Make sure to specify that it had a p-value of 0.06
#   Residual              522    8.2434
# ---
```

```
summary(finalmodel)

## Correlation between the significant environmental variables
cor(metadata[,c(7:19,21:24,33)], method ="pearson")
#create pairs plot to see the correlation statistics between each variable
library(psych)
pairs.panels(metadata[,c(7:19,21:24,33)])



#Year-by-year
#Year 1
ccamodel <- cca(Y1r~., met1[,c(7:37)]) #run1
ccamodel <- cca(Y1r~., met1[,c(7:18,21,23,24,28)]) #run2
finalmodel<- ordistep(ccamodel, scope=formula(ccamodel))
vif.cca(finalmodel)
finalmodel
#               Inertia Proportion Rank
# Total          8.5646     1.0000
# Constrained    2.0371     0.2379   14
# Unconstrained  6.5275     0.7621  142
# Inertia is scaled Chi-square
# 588 species (variables) deleted due to missingness

R2.adj.cca <- RsquareAdj(finalmodel)
R2.adj.cca
# r.squared:0.2591125
# adj.r.squared: 0.1743835

# Testing the significance of the CCA model
anova.cca(finalmodel)
#           Df ChiSquare      F Pr(>F)
# Model     16    2.2068 3.0372  0.001 ***
# Residual 140    6.3578
# ---

# Testing the significance of terms (environmental variables)
anova.cca(finalmodel, by="terms")
# Microcystin             1    0.0339 0.7469  0.746  -> NOT SIG.

#create pairs plot to see the correlation statistics between each variable
library(psych)
pairs.panels(met1[,c(7:18,21,23,24)])


#Year 2
ccamodel <- cca(Y2r~., met2[,c(7:37)]) #run1
ccamodel <- cca(Y2r~., met2[,c(7:19,21:24,28,31,33,36,37)]) #run2
finalmodel<- ordistep(ccamodel, scope=formula(ccamodel))
vif.cca(finalmodel)
finalmodel
#               Inertia Proportion Rank
# Total          9.3746     1.0000
# Constrained    2.3486     0.2505   22
# Unconstrained  7.0260     0.7495  187
# Inertia is scaled Chi-square


R2.adj.cca <- RsquareAdj(finalmodel)
R2.adj.cca
# r.squared:0.2593453
# adj.r.squared:0.172592


anova.cca(finalmodel)
#           Df ChiSquare      F Pr(>F)
# Model     22    2.3486 2.8413  0.001 ***
# Residual 187    7.0260
# ---



#create pairs plot to see the correlation statistics between each variable
library(psych)
pairs.panels(met2[,c(7:19,21:24,28,31,33,36,37)])
```

```
#Year 3
ccamodel <- cca(Y3r~., met3[,c(7:37)]) #run1
ccamodel <- cca(Y3r~., met3[,c(7:10,12:19,21,23,24,31,33)]) #run2
finalmodel<- ordistep(ccamodel, scope=formula(ccamodel))
vif.cca(finalmodel)
finalmodel
#               Inertia Proportion Rank
# Total          6.9434     1.0000
# Constrained    1.9044     0.2743   15
# Unconstrained  5.0390     0.7257  158
# Inertia is scaled Chi-square
# 669 species (variables) deleted due to missingness


R2.adj.cca <- RsquareAdj(finalmodel)
R2.adj.cca
# r.squared: 0.2852408
# adj.r.squared: 0.2068729


anova.cca(finalmodel)
#           Df ChiSquare      F Pr(>F)
# Model     17    1.9617 3.6136  0.001 ***
# Residual 156    4.9817
# ---


#create pairs plot to see the correlation statistics between each variable
library(psych)
pairs.panels(met3[,c(7:10,12:19,21,23,24,31,33)])


###### Plotting CCAs ######
cca.p <- plot(finalmodel,type = "none")

#Fitting of the environmental variables to the CCA plot
ef.cca<- envfit(cca.p,met3[,c(7:10,12:19,21,23,24,31,33)])
#Creating R2 threshold for vectors (found function code on research gate)
#Function: select.envfit - Setting r2 cutoff values to display in an
#                          ordination.r.select<-0.3 # correlation threshold,
#                          see function below
#__FUNCTION: select.envfit__#
# function (select.envfit) filters the resulting list of function (envfit) based on their p values. This allows
to display only significant values in the final plot.
# just run this
select.envfit<-function(fit, r.select){ #needs two sorts of input: fit= result of envfit, r.select= numeric,
correlation minimum threshold
  for (i in 1:length(fit$vectors$r)) { #run for-loop through the entire length of the column r in object
fit$vectors$r starting at i=1
    if (fit$vectors$r[i]<r.select) { #Check wether r<r.select, i.e. if the correlation is weaker than the
threshold value. Change this Parameter for r-based selection
      fit$vectors$arrows[i,]=NA #If the above statement is TRUE, i.e. r is smaller than r.select, then the
coordinates of the vectors are set to NA, so they cannot be displayed
      i=i+1 #increase the running parameter i from 1 to 2, i.e. check the next value in the column until every
value has been checked
    } #close if-loop
  } #close for-loop
  return(fit) #return fit as the result of the function
} #close the function

#Running select function on actual data
ef.cca<- select.envfit(ef.cca, 0.3) #selecting from a weak positive correlation and stronger


## R2 VALUES
#All years
# SecchiDiskDepth           Silica              Sulfate          Temperature            Turbidity
# 0.27094972           0.07374569           0.05479263           0.12495445           0.42287517
# Alkalinity              Ammonia         Pheophytin.a        Chlorophyll.a           TotalDepth
# 0.25428886           0.33806540           0.05730124           0.23852181           0.21004233
# DissolvedOxygen       Nitrate.Nitrite       Phosphate.Ortho                pH      Total.Nitrogen
# 0.42767606           0.54789964           0.47798414           0.34217550           0.05233525
# TN.TP.ratio   Microcystis.Abundance        Microcystin.LR
```

```
# 0.57227444            0.03451549            0.03085789

#Year 1
# SecchiDiskDepth           Silica              Sulfate          Temperature            Turbidity
# 0.304765766           0.059737589           0.006162602           0.025615940           0.314931560
# Alkalinity              Ammonia         Pheophytin.a         Chlorophyll.a            TotalDepth
# 0.210544801           0.597196549           0.054743998           0.175220168           0.220000596
# DissolvedOxygen       Nitrate.Nitrite              pH            TN.TP.ratio    Microcystis.Abundance
# 0.485019703           0.462306509           0.514576526           0.652323571           0.004472664
# Microcystin
# 0.006837999

#Year 2
# SecchiDiskDepth          Silica              Sulfate          Temperature            Turbidity
# 0.18704153            0.07288965            0.14544517            0.14922633            0.52276802
# Alkalinity              Ammonia         Pheophytin.a         Chlorophyll.a            TotalDepth
# 0.25794197            0.35220927            0.08725580            0.35052294            0.18683669
# DissolvedOxygen       Nitrate.Nitrite       Phosphate.Ortho           pH              Total.Nitrogen
# 0.51390253            0.54838242            0.34838408            0.68891135            0.01746031
# TN.TP.ratio   Microcystis.Abundance        Microcystin        Microcystin.LA        Microcystin.LR
# 0.62322767            0.01788581            0.00223627            0.02135175            0.03884635
# Anatoxin.a    Cylindrospermopsin
# 0.04925972            0.03583364

#Year 3
# SecchiDiskDepth           Silica              Sulfate          Temperature            Alkalinity
# 0.12798686            0.14790446            0.16111518            0.36344282            0.30968020
# Ammonia             Pheophytin.a         Chlorophyll.a            TotalDepth       DissolvedOxygen
# 0.18427317            0.09774378            0.38622539            0.20853791            0.30090864
# Nitrate.Nitrite       Phosphate.Ortho           pH                TN.TP.ratio    Microcystis.Abundance
# 0.67163554            0.44076153            0.11917088            0.36285678            0.55155892
# Microcystin.LA        Microcystin.LR
# 0.03009204            0.38899517

#Microcystin LR strongly correlated to Microcystis abundance so removing that vector
ef.cca$vectors$arrows["Microcystin.LR",]=NA


#Setting up base plot
#ALL Years
par(mar=c(5.1, 6.1, 3.1, 4.1))
plot(finalmodel,type = "none")
abline(h = 0, v = 0, col = "white", lwd = 2)
box()
#Year 1
par(mar=c(5.1, 6.1, 3.1, 4.1))
plot(finalmodel,type = "none")
abline(h = 0, v = 0, col = "white", lwd = 2)
box()
#Year 2
par(mar=c(5.1, 6.1, 3.1, 4.1))
plot(finalmodel,type = "none")
abline(h = 0, v = 0, col = "white", lwd = 2)
box()
#Year 3
par(mar=c(5.1, 6.1, 3.1, 4.1))
plot(finalmodel,type = "none")
abline(h = 0, v = 0, col = "white", lwd = 2)
box()


#Adding the points

#Year
#Adding the points
points(cca.p,"sites", pch=19, col= "goldenrod3", select = metadata$Year == "1")
points(cca.p,"sites", pch=19, col= "mediumpurple2", select = metadata$Year == "2")
points(cca.p,"sites", pch=19, col= "springgreen4", select = metadata$Year == "3")
#Plotting envfit vectors
plot(ef.cca, col = "black", p.max=0.05)
#Add legend (click to place legend on the outside of the plot) & Title
legend(locator(1),legend=c("1","2", "3"),
       col=c("goldenrod3","mediumpurple2", "springgreen4"), pch=19, cex=1.2,
       title = "Year")
title(main="Years 1 - 3 (2019 - 2021)")
```

```
#Zone
points(cca.p,"sites", pch=19, col= "palegreen3", select = met3$Zone == "Inflow")
points(cca.p,"sites", pch=19, col= "cornflowerblue", select = met3$Zone == "Pelagic")
points(cca.p,"sites", pch=19, col= "wheat4", select = met3$Zone == "Nearshore")
points(cca.p,"sites", pch=19, col= "violetred2", select = met3$Zone == "S79")
#Plotting envfit vectors
plot(ef.cca, col = "black", p.max=0.05)
#Add legend (click to place legend on the outside of the plot) & Title
legend(locator(1),legend=c("Inflow","Nearshore","Pelagic","S79"),
       col=c("palegreen3","wheat4","cornflowerblue","violetred2"), pch=19, cex=1.2,
       title = "Ecological Zone")
title(main="Years 1 - 3 (2019 - 2021)")
title(main="Year 1 - 2019")
title(main="Year 2 - 2020")
title(main="Year 3 - 2021")
#Season
#Adding the points
points(cca.p,"sites", pch=19, col= "lemonchiffon3", select = met3$Season == "dry")
points(cca.p,"sites", pch=19, col= "royalblue1", select = met3$Season == "wet")
#Plotting envfit vectors
plot(ef.cca, col = "black", p.max=0.05)
#Add legend (click to place legend on the outside of the plot) & Title
legend(locator(1),legend=c("Dry","Wet"),
       col=c("lemonchiffon3","royalblue1"), pch=19, cex=1.2, title = "Season")
title(main="Years 1 - 3 (2019 - 2021)")
title(main="Year 1 - 2019")
title(main="Year 2 - 2020")
title(main="Year 3 - 2021")

#Month
#Adding the points
points(cca.p,"sites", pch=19, col= "firebrick2", select = met3$Month == "1")
points(cca.p,"sites", pch=19, col= "darkorange1", select = met3$Month == "2")
points(cca.p,"sites", pch=19, col= "gray38", select = met3$Month == "3")
points(cca.p,"sites", pch=19, col= "goldenrod1", select = met3$Month == "4")
points(cca.p,"sites", pch=19, col= "green4", select = met3$Month == "5")
points(cca.p,"sites", pch=19, col= "cadetblue2", select = met3$Month == "6")
points(cca.p,"sites", pch=19, col= "dodgerblue2", select = met3$Month == "7")
points(cca.p,"sites", pch=19, col= "mediumpurple2", select = met3$Month == "8")
points(cca.p,"sites", pch=19, col= "hotpink", select = met3$Month == "9")
points(cca.p,"sites", pch=19, col= "tan", select = met3$Month == "10")
points(cca.p,"sites", pch=19, col= "saddlebrown", select = met3$Month == "11")
points(cca.p,"sites", pch=19, col= "purple4", select = met3$Month == "12")
#Plotting envfit vectors
plot(ef.cca, col = "black", p.max=0.05)
#Add legend (click to place legend on the outside of the plot) & Title
legend(locator(1),legend= c("3","4","5","6","7","8","9","10","11","12"),
       title = "Month",ncol = 2,
       col=c("gray34","goldenrod2","green3",
                        "cadetblue2","dodgerblue2","mediumpurple2","hotpink","tan","saddlebrown","purple4"),
                        pch=19, cex=1.2)
legend(locator(1),legend= c("1","2","3","4","5","6","7","8","9","10","11","12"),
       title = "Month",ncol = 2,
       col=c("firebrick2","darkorange1","gray34","goldenrod2","green3",
                        "cadetblue2","dodgerblue2","mediumpurple2","hotpink","tan","saddlebrown","purple4"),
                        pch=19, cex=1.2)
legend(locator(1),legend= c("1","2","3","4","5","6","7","8","9","10"),
       title = "Month",ncol = 2,
       col=c("firebrick2","darkorange1","gray34","goldenrod2","green3",
                        "cadetblue2","dodgerblue2","mediumpurple2","hotpink","tan"),
                        pch=19, cex=1.2)
title(main="Years 1 - 3 (2019 - 2021)")
title(main="Year 1 - 2019")
title(main="Year 2 - 2020")
title(main="Year 3 - 2021")
#Station
#Adding the points
points(cca.p,"sites", pch=19, col= "#A6CEE3", select = met3$Station == "CLV10A")
points(cca.p,"sites", pch=19, col= "#579CC7", select = met3$Station == "KISSR0.0")
points(cca.p,"sites", pch=19, col= "#3688AD", select = met3$Station == "L001")
points(cca.p,"sites", pch=19, col= "#8BC395", select = met3$Station == "L004")
points(cca.p,"sites", pch=19, col= "#89CB6C", select = met3$Station == "L005")
points(cca.p,"sites", pch=19, col= "#40A635", select = met3$Station == "L006")
points(cca.p,"sites", pch=19, col= "#919D5F", select = met3$Station == "L007")
points(cca.p,"sites", pch=19, col= "#F99392", select = met3$Station == "L008")
points(cca.p,"sites", pch=19, col= "#EB444A", select = met3$Station == "LZ2")
```

```
points(cca.p,"sites", pch=19, col= "red", select = met3$Station == "LZ25A")
points(cca.p,"sites", pch=19, col= "#F79C5D", select = met3$Station == "LZ30")
points(cca.p,"sites", pch=19, col= "#FDA746", select = met3$Station == "LZ40")
points(cca.p,"sites", pch=19, col= "#FE8205", select = met3$Station == "PALMOUT")
points(cca.p,"sites", pch=19, col= "#E39970", select = met3$Station == "PELBAY3")
points(cca.p,"sites", pch=19, col= "#BFA5CF", select = met3$Station == "POLE3S")
points(cca.p,"sites", pch=19, col= "#8861AC", select = met3$Station == "POLESOUT")
points(cca.p,"sites", pch=19, col= "violet", select = met3$Station == "RITTAE2")
points(cca.p,"sites", pch=19, col= "#E7E099", select = met3$Station == "S308")
points(cca.p,"sites", pch=19, col= "#DEB969", select = met3$Station == "S77")
points(cca.p,"sites", pch=19, col= "#B15928", select = met3$Station == "S79")
#Plotting envfit vectors
plot(ef.cca, col = "black", p.max=0.05)
#Add legend (click to place legend on the outside of the plot) & Title
legend(locator(1),legend= c("CLV10A","KISSR0.0","L001","L004","L005","L006","L007",
                            "L008","LZ2","LZ25A","LZ30","LZ40","PALMOUT","PELBAY3",
                            "POLE3S","POLESOUT","RITTAE2","S308","S77","S79"),
       title = "Station",ncol=2,
       col=c("#A6CEE3","#579CC7","#3688AD","#8BC395","#89CB6C","#40A635","#919D5F",
                      "#F99392","#EB444A","red","#F79C5D","#FDA746","#FE8205","#E39970",
                      "#BFA5CF","#8861AC","violet","#E7E099","#DEB969","#B15928"),
                      pch=19, cex=0.9)
title(main="Years 1 - 3 (2019 - 2021)")
title(main="Year 1 - 2019")
title(main="Year 2 - 2020")
title(main="Year 3 - 2021")


###### Differential Abundance Analysis - DESEQ2 ######
## USING DESEQ2 (following lashlock github tutorial)
library(DESeq2)

##Differences between years
#load in data WITHOUT rownames
years <- read.csv("feature_Y123_0.01per.csv")
met <- read.csv("Metadata-Diversity_BATCH.csv")
#turning Year into a factor (since it may be read as a number)
met$Year <- as.factor(met$Year)

##Constructing Deseq2 object from data frame
dds <- DESeqDataSetFromMatrix(countData=years,
                              colData=met,
                              design=~Year, tidy = TRUE)
#Design specifies how the counts from each gene depend on our variables in the metadata
#For this dataset the factor we care about is the Zone
#tidy=TRUE argument = tells DESeq2 to output the results table with row names as a first #column called 'row.


#let's see what this object looks like
dds
# class: DESeqDataSet
# dim: 8340 541
# metadata(1): version
# assays(1): counts
# rownames(8340): 0885965c051f3034c0e28043193bc5d2 51e00e866016fba8a19581249b811ec4 ...
# f9fe4768ad3ef514b97950516e4af5b2 fe2896a859ec05fd0b600b2f633a3bc7
# rowData names(0):
#   colnames(541): KISSR0.0_3_19 L001_3_19 ... S77_10_21 S79_10_21
# colData names(43): Sample Month ... J inv.D


##Running the DESeq function
dds <- DESeq(dds)
#Error in estimateSizeFactorsForMatrix(counts(object),locfunc =
#locfunc,: every gene contains at least one zero, cannot compute log geometric
#means -> got this error so going to add a pseudocount of 1 to eliminate zeroes
#         (may add bias to the data according to vegan HELP)

##Adding pseudocount of 1 to feature table
#looking at the structure of the data frame
str(years)
#first column is a character so don't include in the transformation

#Adding 1 excluding the first column (ASV column)
years[-1] <- years[-1] + 1
```

```
##Retrying the constructing DESeq object and running the DESeq function
dds <- DESeqDataSetFromMatrix(countData=years,
                              colData=met,
                              design=~Year, tidy = TRUE)
dds <- DESeq(dds)

##What just happen?
#estimateSizeFactors
#This calculates the relative library depth of each sample

#estimateDispersions
#estimates the dispersion of counts for each gene

#nbinomWaldTest
#calculates the significance of coefficients in a Negative Binomial GLM using the size and dispersion outputs


##Looking at the results table
res31 <- results(dds)
res31 #looking at the results table
# log2 fold change (MLE): Year 3 vs 1
# Wald test p-value: Year 3 vs 1
# DataFrame with 8340 rows and 6 columns
#                                  baseMean log2FoldChange     lfcSE      stat     pvalue      padj
#                                 <numeric>      <numeric> <numeric> <numeric>  <numeric> <numeric>
# 0885965c051f3034c0e28043193bc5d2   1.17377      0.2149733  0.152042  1.413911  0.1573881        NA
# 51e00e866016fba8a19581249b811ec4   1.14815      0.0699605  0.157170  0.445127  0.6562281        NA
# dfd3874c0e70ae177e8cdc4fb6961e7d   1.22257      0.0762662  0.152455  0.500255  0.6168956 0.7984313
# ac879ef0bc703ee2637bc55f0ef97afc   1.24454      0.3709705  0.155215  2.390037  0.0168467 0.0805026
# 41714fa1a258e8098d51d03a1e1b3304   1.20327     -0.3581498  0.149734 -2.391912  0.0167608 0.0802964


##NOTE: If there are more than 2 levels for the variable - as is the case
##for Year w/ 3 levels - results will extract the results table for a comparison
##of the last level over the first level (so year 3 vs year 1)


##Other comparisons
res23 <- results(dds, contrast = c("Year", "3", "2") )
res23
# log2 fold change (MLE): Year 3 vs 2
# Wald test p-value: Year 3 vs 2
# DataFrame with 8340 rows and 6 columns
#                                  baseMean log2FoldChange     lfcSE      stat      pvalue        padj
#                                 <numeric>      <numeric> <numeric> <numeric>   <numeric>   <numeric>
# 0885965c051f3034c0e28043193bc5d2   1.17377      0.2169855  0.140908  1.539912   0.1235819          NA
# 51e00e866016fba8a19581249b811ec4   1.14815     -0.0866411  0.142640 -0.607409   0.5435795          NA
# dfd3874c0e70ae177e8cdc4fb6961e7d   1.22257     -0.0207447  0.139635 -0.148564   0.8818978   0.9376288
# ac879ef0bc703ee2637bc55f0ef97afc   1.24454      0.3451806  0.142798  2.417259   0.0156379   0.0690231
# 41714fa1a258e8098d51d03a1e1b3304   1.20327     -0.0463689  0.146971 -0.315496   0.7523851          NA

res12 <- results(dds, contrast = c("Year", "1", "2") )
res12
# log2 fold change (MLE): Year 1 vs 2
# Wald test p-value: Year 1 vs 2
# DataFrame with 8340 rows and 6 columns
#                                  baseMean log2FoldChange     lfcSE       stat    pvalue      padj
#                                 <numeric>      <numeric> <numeric>  <numeric> <numeric> <numeric>
# 0885965c051f3034c0e28043193bc5d2   1.17377     0.00201214  0.150700  0.0133519 0.9893470        NA
# 51e00e866016fba8a19581249b811ec4   1.14815    -0.15660154  0.149012 -1.0509325 0.2932896        NA
# dfd3874c0e70ae177e8cdc4fb6961e7d   1.22257    -0.09701090  0.145892 -0.6649517 0.5060814  0.726901
# ac879ef0bc703ee2637bc55f0ef97afc   1.24454    -0.02578994  0.155971 -0.1653507 0.8686680  0.941353
# 41714fa1a258e8098d51d03a1e1b3304   1.20327     0.31178087  0.141655  2.2009921 0.0277366  0.116155

##Saving all comparisons as CSVs
write.csv(res31, "DESEQ-Y13_results.csv")
write.csv(res23, "DESEQ-Y23_results.csv")
write.csv(res12, "DESEQ-Y12_results.csv")


#Visualizing using Volcano plots
##Volcano Plot
par(mfrow=c(1,3))
#Year 3 vs Year 1
# Make a basic volcano plot
```

```
with(res31, plot(log2FoldChange, -log10(pvalue), pch=20, main="Year 3 vs. Year 1", xlim=c(-2,2)))
# Add colored points: red = padj<0.05 AND log2FC >1, black = pdj>0.05
with(subset(res31, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(res31, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Year 3 vs Year 2
# Make a basic volcano plot
with(res23, plot(log2FoldChange, -log10(pvalue), pch=20, main="Year 3 vs. Year 2", xlim=c(-3,3)))
# Add colored points: red = padj<0.05 AND log2FC >1, black = pdj>0.05
with(subset(res23, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(res23, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Year 1 vs Year 2
# Make a basic volcano plot
with(res12, plot(log2FoldChange, -log10(pvalue), pch=20, main="Year 1 vs. Year 2", xlim=c(-3,3)))
# Add colored points: red = padj<0.05 AND log2FC >1, black = pdj>0.05
with(subset(res12, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(res12, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

##PCA
#First we need to transform the raw count data
#vst function will perform variance stabilizing transformation
par(mfrow=c(1,1))
vsdata <- vst(dds, blind=FALSE) #using the DESEQ2 plotPCA function we can
#look at how our samples group by treatment
plotPCA(vsdata, intgroup="Year")+
  labs(title = "Years 1-3 (2019-2021)")+
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))


#### Differences in Zone for EACH YEAR
#loading in data
Y1 <- dat.01per[grep("_19$", rownames(dat.01per)),]
Y2 <- dat.01per[grep("_20$", rownames(dat.01per)),]
Y3 <- dat.01per[grep("_21$", rownames(dat.01per)),]
write.csv(t(Y1), "feature_Y1_0.01per.csv")
write.csv(t(Y2), "feature_Y2_0.01per.csv")
write.csv(t(Y3), "feature_Y3_0.01per.csv")


##Differences found in Zone of Year 1
Y1 <- read.csv("feature_Y1_0.01per.csv")
met1 <- read.csv("Metadata_BATCH_Y1.csv")

##Adding pseudocount of 1
Y1[-1] <- Y1[-1] + 1

##Constructing Deseq2 object and running DESeq function
dds <- DESeqDataSetFromMatrix(countData=Y1,
                              colData=met1,
                              design=~Zone, tidy = TRUE)
dds <- DESeq(dds)

##Retrieving results tables for each comparison
resIP <- results(dds, contrast = c("Zone", "Inflow", "Pelagic") )
resIN <- results(dds, contrast = c("Zone", "Inflow", "Nearshore") )
resNP <- results(dds, contrast = c("Zone", "Nearshore", "Pelagic") )
resNS <- results(dds, contrast = c("Zone", "Nearshore", "S79") )
resPS <- results(dds, contrast = c("Zone", "Pelagic", "S79") )
resSI <- results(dds)

##Saving all comparisons as CSVs
write.csv(resIP, "DESEQ-Y1IP_results.csv")
write.csv(resIN, "DESEQ-Y1IN_results.csv")
write.csv(resNP, "DESEQ-Y1NP_results.csv")
write.csv(resNS, "DESEQ-Y1NS_results.csv")
write.csv(resPS, "DESEQ-Y1PS_results.csv")
write.csv(resSI, "DESEQ-Y1SI_results.csv")

##Volcano Plots
par(mfrow=c(2,3))
#Inflow vs Pelagic
with(resIP, plot(log2FoldChange, -log10(pvalue), pch=20, main="Inflow vs. Pelagic", xlim=c(-6,6)))
with(subset(resIP, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resIP, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))
```

```
#Inflow vs Nearshore
with(resIN, plot(log2FoldChange, -log10(pvalue), pch=20, main="Inflow vs. Nearshore", xlim=c(-6,6)))
with(subset(resIN, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resIN, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Nearshore vs. Pelagic
with(resNP, plot(log2FoldChange, -log10(pvalue), pch=20, main="Nearshore vs. Pelagic", xlim=c(-4,4)))
with(subset(resNP, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resNP, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Nearshore vs. S79
with(resNS, plot(log2FoldChange, -log10(pvalue), pch=20, main="Nearshore vs. S79", xlim=c(-8,8)))
with(subset(resNS, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resNS, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Pelagic vs. S79
with(resPS, plot(log2FoldChange, -log10(pvalue), pch=20, main="Pelagic vs. S79", xlim=c(-8,8)))
with(subset(resPS, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resPS, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#S79 vs Inflow
with(resSI, plot(log2FoldChange, -log10(pvalue), pch=20, main="S79 vs. Inflow", xlim=c(-7,7)))
with(subset(resSI, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resSI, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

##PCA
par(mfrow=c(1,1))
vsdata <- vst(dds, blind=FALSE)
plotPCA(vsdata, intgroup="Zone")+
  labs(title = "Year 1 - Ecological zones")+
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

##Year 2 Zone (No significant differences found but doing it anyway)
Y2 <- read.csv("feature_Y2_0.01per.csv")
met2 <- read.csv("Metadata_BATCH_Y2.csv")

##Adding pseudocount of 1
Y2[-1] <- Y2[-1] + 1

##Constructing Deseq2 object and running DESeq function
dds <- DESeqDataSetFromMatrix(countData=Y2,
                              colData=met2,
                              design=~Zone, tidy = TRUE)
dds <- DESeq(dds)

##Retrieving results tables for each comparison
resIP <- results(dds, contrast = c("Zone", "Inflow", "Pelagic") )
resIN <- results(dds, contrast = c("Zone", "Inflow", "Nearshore") )
resNP <- results(dds, contrast = c("Zone", "Nearshore", "Pelagic") )
resNS <- results(dds, contrast = c("Zone", "Nearshore", "S79") )
resPS <- results(dds, contrast = c("Zone", "Pelagic", "S79") )
resSI <- results(dds)

##Saving all comparisons as CSVs
write.csv(resIP, "DESEQ-Y2IP_results.csv")
write.csv(resIN, "DESEQ-Y2IN_results.csv")
write.csv(resNP, "DESEQ-Y2NP_results.csv")
write.csv(resNS, "DESEQ-Y2NS_results.csv")
write.csv(resPS, "DESEQ-Y2PS_results.csv")
write.csv(resSI, "DESEQ-Y2SI_results.csv")

##Volcano Plots
par(mfrow=c(2,3))
#Inflow vs Pelagic
with(resIP, plot(log2FoldChange, -log10(pvalue), pch=20, main="Inflow vs. Pelagic", xlim=c(-5,5)))
with(subset(resIP, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resIP, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Inflow vs Nearshore
with(resIN, plot(log2FoldChange, -log10(pvalue), pch=20, main="Inflow vs. Nearshore", xlim=c(-6,6)))
with(subset(resIN, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resIN, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Nearshore vs. Pelagic
```

```
with(resNP, plot(log2FoldChange, -log10(pvalue), pch=20, main="Nearshore vs. Pelagic", xlim=c(-6,6)))
with(subset(resNP, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resNP, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Nearshore vs. S79
with(resNS, plot(log2FoldChange, -log10(pvalue), pch=20, main="Nearshore vs. S79", xlim=c(-7,7)))
with(subset(resNS, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resNS, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Pelagic vs. S79
with(resPS, plot(log2FoldChange, -log10(pvalue), pch=20, main="Pelagic vs. S79", xlim=c(-7,7)))
with(subset(resPS, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resPS, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#S79 vs Inflow
with(resSI, plot(log2FoldChange, -log10(pvalue), pch=20, main="S79 vs. Inflow", xlim=c(-7,7)))
with(subset(resSI, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resSI, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

##PCA
par(mfrow=c(1,1))
vsdata <- vst(dds, blind=FALSE)
plotPCA(vsdata, intgroup="Zone")+
  labs(title = "Year 2 - Ecological zones")+
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

##Differences found in Zone of Year 3
Y3 <- read.csv("feature_Y3_0.01per.csv")
met3 <- read.csv("Metadata_BATCH_Y3.csv")

##Adding pseudocount of 1
Y3[-1] <- Y3[-1] + 1

##Constructing Deseq2 object and running DESeq function
dds <- DESeqDataSetFromMatrix(countData=Y3,
                              colData=met3,
                              design=~Zone, tidy = TRUE)
dds <- DESeq(dds)

##Retrieving results tables for each comparison
resIP <- results(dds, contrast = c("Zone", "Inflow", "Pelagic") )
resIN <- results(dds, contrast = c("Zone", "Inflow", "Nearshore") )
resNP <- results(dds, contrast = c("Zone", "Nearshore", "Pelagic") )
resNS <- results(dds, contrast = c("Zone", "Nearshore", "S79") )
resPS <- results(dds, contrast = c("Zone", "Pelagic", "S79") )
resSI <- results(dds)

##Saving all comparisons as CSVs
write.csv(resIP, "DESEQ-Y3IP_results.csv")
write.csv(resIN, "DESEQ-Y3IN_results.csv")
write.csv(resNP, "DESEQ-Y3NP_results.csv")
write.csv(resNS, "DESEQ-Y3NS_results.csv")
write.csv(resPS, "DESEQ-Y3PS_results.csv")
write.csv(resSI, "DESEQ-Y3SI_results.csv")

##Volcano Plots
par(mfrow=c(2,3))
#Inflow vs Pelagic
with(resIP, plot(log2FoldChange, -log10(pvalue), pch=20, main="Inflow vs. Pelagic", xlim=c(-5,5)))
with(subset(resIP, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resIP, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Inflow vs Nearshore
with(resIN, plot(log2FoldChange, -log10(pvalue), pch=20, main="Inflow vs. Nearshore", xlim=c(-6,6)))
with(subset(resIN, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resIN, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Nearshore vs. Pelagic
with(resNP, plot(log2FoldChange, -log10(pvalue), pch=20, main="Nearshore vs. Pelagic", xlim=c(-7,7)))
with(subset(resNP, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resNP, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Nearshore vs. S79
with(resNS, plot(log2FoldChange, -log10(pvalue), pch=20, main="Nearshore vs. S79", xlim=c(-6,6)))
with(subset(resNS, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
```

```
with(subset(resNS, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#Pelagic vs. S79
with(resPS, plot(log2FoldChange, -log10(pvalue), pch=20, main="Pelagic vs. S79", xlim=c(-7,7)))
with(subset(resPS, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resPS, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

#S79 vs Inflow
with(resSI, plot(log2FoldChange, -log10(pvalue), pch=20, main="S79 vs. Inflow", xlim=c(-7,7)))
with(subset(resSI, padj<.05 & abs(log2FoldChange)>1), points(log2FoldChange, -log10(pvalue), pch=20, col="red"))
with(subset(resSI, padj>.05), points(log2FoldChange, -log10(pvalue), pch=20, col="black"))

##PCA
par(mfrow=c(1,1))
vsdata <- vst(dds, blind=FALSE)
plotPCA(vsdata, intgroup="Zone")+
  labs(title = "Year 3 - Ecological zones")+
  theme(plot.title.position = "panel")+
  theme(plot.title = element_text(size = rel(1.5), hjust = 0.5))

###### Species Co-occurrence (Correlations) ######
library(Hmisc)

#All Years
x<-read.csv("feature_Y123_0.01per.csv", header=TRUE, row.names=1)
x<-t(x)
y<-rcorr(as.matrix(x, type = c("pearson")))  ## or spearman (pearson may be best here)
yR<-y$r
yP<-y$P

flattenCorrMatrix <- function(cormat, pmat) {
  ut <- upper.tri(cormat)
  data.frame(
    row = rownames(cormat)[row(cormat)[ut]],
    column = rownames(cormat)[col(cormat)[ut]],
    corr =(cormat)[ut],
    p = pmat[ut]
  )
}
corr_data<-flattenCorrMatrix(y$r, y$P)

#Note:
# Sort in R or in excel... may want to only keep significant correlations that are
# to Microcystis specifically to keep it simple. then retain R2 values that are the
# highest (>0.9 or <-0.9 -- you can change that if you want.) <- cut off will have
# to be 0.3 since that's the highest

# Use these values to create network in Cytoscape to visualize the correlations of taxa
# to Microcystis.

#Excluding any non-significant correlations (including zeros) and exporting
corr_data <- corr_data[order(corr_data$p),] #sort from smallest to largest
corr_sig <- corr_data[corr_data$p < 0.05, ] #Subsetting data to ONLY include significant correlations
write.csv(corr_sig, "LakeOCorrelationsSigONLY.csv")

#Created network in Cytoscape, merging nodes with taxonomy
node <- read.csv("LakeOCorrelations_Nodes.csv")
tax <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE)
merged <- merge(node,tax, by="FeatureID")
write.csv(merged,"LakeOCorrelations_NodeTaxa.csv")
#Microcystis with corr = 0.7 and up, merging with taxonomy
node <- read.csv("Microcystis Network-0.7+_Node.csv")
tax <- read.csv("taxonomy_Y123_edited&cleaned.csv", header = TRUE)
merged <- merge(node,tax, by="FeatureID")
write.csv(merged,"LakeOCorrelations_Microcystis0.7NodeTaxa.csv")


#### ####
```