echo "# ChowanDEQ" >> Untitled.md

git init

git add README.md

git commit -m "first commit"

git branch -M main

git remote add origin https://github.com/haleyplaas/ChowanDEQ.git

git push -u origin main

install.packages(usethis)

rm(list=ls())

setwd("/Users/haleyplaas/OneDrive - University of North Carolina at Chapel Hill/Coding/R/Chowan Data/multi-year")

library(dplyr);library(tidyr);library(ggplot2);library(gam);library(gamm4);library(cowplot);library(mgcv);library(reshape2);library(MuMIn); library(stringr); library(ISLR); library(voxel); library(gridExtra); library(purrr)

#Each chunk of years must be downloaded from the Water Quality Portal separately due to the size of the data set -- load them and combine

#make sure when loading the csv. file has date listed as YYYY-MM-DD or it won't important properly and yield NAs

NC01WQ <- read.csv("21NC01WQ.2.csv", header = TRUE, na.strings = c(""," ", ".", "NA"))

NC01WQ.1 <- NC01WQ %>% mutate(ActivityStartDate = as.Date(ActivityStartDate, "%Y-%m-%d"))

NC02WQ <- read.csv("21NC02WQ.2.csv", header = TRUE, na.strings = c(""," ", ".", "NA"))

NC02WQ.1 <- NC02WQ %>% mutate(ActivityStartDate = as.Date(ActivityStartDate, "%Y-%m-%d"))

NC03WQ <- read.csv("21NC03WQ.2.csv", header = TRUE, na.strings = c(""," ", ".", "NA"))

NC03WQ.1 <- NC03WQ %>% mutate(ActivityStartDate = as.Date(ActivityStartDate, "%Y-%m-%d"))

dirty.df <- rbind(NC01WQ.1, NC02WQ.1, NC03WQ.1)

#selecting specific parameters of interest

clean.0 <- dirty.df %>% select(Date = ActivityStartDate,

Depth = ActivityDepthHeightMeasure.MeasureValue,

Depth.Unit = ActivityDepthHeightMeasure.MeasureUnitCode,

Identifier = MonitoringLocationIdentifier,

Parameter = CharacteristicName,

Parameter.Value = ResultMeasureValue,

Parameter.Unit = ResultMeasure.MeasureUnitCode,

Qualifier.Code = ResultCommentText,

LOQ = DetectionQuantitationLimitMeasure.MeasureValue,

LOQ.Unit = DetectionQuantitationLimitMeasure.MeasureUnitCode)

#Labeling each site by its Station (separating identifier 21NCXXWQ)

clean.1 <- tidyr::separate(clean.0, Identifier, c("Identifier", "Station"))

#relabeling stations which were renamed over time

clean.1 <- clean.1 %>% mutate(Station = recode(clean.1$Station, `M6100000` = "M610000C", `M3900000` = "M390000C", `D9995000` = "D999500C"))

#assign data types for later manipulation

clean.2 <- clean.1 %>% mutate(Depth = as.numeric(Depth),

Depth.Unit = as.character(Depth.Unit),

Station = as.character(Station),

Identifier = as.character(Identifier),

Parameter = as.character(Parameter),

Parameter.Value = as.numeric(Parameter.Value),

Parameter.Unit = as.character(Parameter.Unit),

Qualifier.Code = as.character(Qualifier.Code),

LOQ = as.numeric(LOQ),

LOQ.Unit = as.character(LOQ.Unit))

#the NAs introduced by coercion is specifically for the Flow, which was categorized into severity values (character strings) but this is fine because later I import flow rates from USGS files -- so these will later be filtered out

str(clean.2) #use this command to confirm assigned data types which is essential for later manipulation

#CONVERT ALL VALUES INPUT WITH DIFFERENT UNITS

unique(clean.2$Parameter.Unit) #to see how many different units are there (to determine which need to be converted)

clean.2 %>% count(clean.2$Parameter.Unit== c("deg F")) # for baseline to see how many values are in Fahrenheit

clean.3 <- clean.2 %>% mutate(Parameter.Value = case\_when(clean.2$Parameter.Unit == "deg C" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "mg/l" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "uS/cm" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "None" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "ppth" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "ug/l" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == NA ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "NTU" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "cfu/100ml" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "umho/cm" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "%" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "FTU" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "ft" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "#/100ml" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "PCU" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "g/l" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "ml/l" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "cfs" ~ clean.2$Parameter.Value,

clean.2$Parameter.Unit == "deg F" ~ (clean.2$Parameter.Value-32)\*5/9)) #Fahrenheit to Celsius

clean.3 <- clean.3 %>% mutate(Parameter.Unit = recode(clean.3$Parameter.Unit, `deg F` = "deg C", `g/l` = "ppth", `FTU` = "NTU")) #Salinity from g/L to ppth and FTU to NTU (same values but different unit label)

clean.3 %>% count(clean.3$Parameter.Unit== c("deg F")) #to check if recoding it worked, TRUE should be 0

#CONVERTING ALL VALUES IN FEET TO METERS

unique(clean.3$Depth.Unit) # to see how many different units are there

clean.3 %>% count(clean.3$Depth.Unit=="ft") #baseline

clean.4 <- clean.3 %>% mutate(Depth = case\_when(clean.3$Depth.Unit == "m" ~ clean.3$Depth,

clean.3$Depth.Unit == NA ~ clean.3$Depth,

clean.3$Depth.Unit == "ft" ~ (clean.3$Depth\*0.3048))) %>% mutate(Depth.Unit = recode(clean.3$Depth.Unit, "ft" = "m"))

clean.4 %>% count(clean.4$Depth.Unit=="ft") #to check if it worked

#Next step is to see what other less obvious values might need to be converted so that the units match --> exporting to excel for examination

unique(clean.4$Parameter.Unit)

grouped.units <- clean.4 %>% mutate(together = paste(Parameter, Parameter.Unit, sep = "."))

df <- as.data.frame(unique(grouped.units$together))

#the below function to save csv. (rename before re-running script so as not to overwrite)

#write.csv(df,"/Users/haleyplaas/OneDrive - University of North Carolina at Chapel Hill/Coding/R/Chowan Data/multi-year\\Parameters.with.units.1.csv", row.names = FALSE)

#see above CSV for list of parameters with justification for selections / conversions determined at this point, and see email from Tammy Hill (in READ.ME for GitHub users) with specifications on labeling schemes and quantification methods for various parameters

#First thing to address from NC-DEQ's specifications. Qualifier codes, which are actually stored in the "ResultCommentText" column. These codes mainly reveal detection limits for the NDs which are listed as blank cells

#What I need to do is change blank cells which were actually NDs to ND with detection threshold in another column. This is the case with several of the nutrient analytes. Label them ND or <X detection value instead of NA or blank

unique(clean.4$Qualifier.Code)

#replacing empty cells in the Parameter values with ND (not detected) when the qualifier code is or contains U or "Reported".

clean.5 <- clean.4 %>% mutate(AA = case\_when(str\_detect(Qualifier.Code, 'U|Reported') ~ 'ND')) %>%

mutate(Parameter.value.new = Parameter.Value %>% is.na %>% ifelse(AA, Parameter.Value)) %>%

select(-Parameter.Value, -AA) %>%

rename(Parameter.Value = Parameter.value.new)

#trying this but instead of choosing the specific qualifier codes make it so it's just replacing the 0's with NDs for rows with ANY VALUE != NA in qualifier code.. hopefully this clears up the ortho.P and CHLA issue

clean.5 <- clean.4 %>% mutate(AA = case\_when(Qualifier.Code != "NA" ~ 'ND')) %>%

mutate(Parameter.value.new = Parameter.Value %>% is.na %>% ifelse(AA, Parameter.Value)) %>%

select(-Parameter.Value, -AA) %>%

rename(Parameter.Value = Parameter.value.new)

col\_order <- c("Date", "Depth", "Depth.Unit", "Identifier", "Station", "Parameter", "Parameter.Value", "Parameter.Unit", "Qualifier.Code", "LOQ", "LOQ.Unit")

clean.6 <- clean.5[,col\_order]

#so now the blank cells necessary (mostly nutrient analytes) are assigned ND, time to merge parameters that have been re-labeled throughout time

#2. Ammonia in 21NC01WQ data is calculated. The actual result is Ammonia-nitrogen as N. Need to delete all of the Ammonia calculated data.

count(clean.6, clean.6$Identifier == "21NC01WQ" & clean.6$Parameter=="Ammonia") #baseline

rid.of.calc.ammonia <- clean.6 %>% filter(Identifier == "21NC01WQ")

no.calc.ammonia <- rid.of.calc.ammonia %>% filter(Parameter != "Ammonia") #removing all calculated Ammonias so they don't get confused with the later labeled Ammonia in 21NC03WQ

recent.years <- clean.6 %>% filter(Identifier == "21NC02WQ" | Identifier == "21NC03WQ")

clean.7 <- rbind(no.calc.ammonia, recent.years)

count(clean.7, clean.7$Identifier == "21NC01WQ" & clean.7$Parameter=="Ammonia") #check, successful

# Ammonia in the 21NC03WQ data, on 1/1/2017, Ammonia-nitrogen became Ammonia (let's relabel all Ammonia-X to just plain Ammonia)

count(clean.7, Parameter=="Ammonia") #baseline

clean.8 <- clean.7 %>% mutate(Parameter = recode(clean.7$Parameter, `Ammonia-nitrogen as N` = "Ammonia", `Ammonia-nitrogen` = "Ammonia"))

count(clean.8, Parameter=="Ammonia") #check, successful. Now all AMMONIA is the same

#Inorganic Nitrogen (nitrate and nitrite) as N and Inorganic Nitrogen (nitrate and nitrite) are the same -- just a labeling change

clean.8 %>% count(Parameter=="Inorganic nitrogen (nitrate and nitrite) as N") #baseline

clean.8 <- clean.8 %>% mutate(Parameter = recode(Parameter, "Inorganic nitrogen (nitrate and nitrite) as N" = "Inorganic nitrogen (nitrate and nitrite)"))

clean.8 %>% count(Parameter=="Inorganic nitrogen (nitrate and nitrite)") #check, success

#Phosphate-phosphorus as P and Phosphorus are the same analyte

clean.8 %>% count(Parameter=="Phosphate-phosphorus as P") #baseline

clean.8 <- clean.8 %>% mutate(Parameter = recode(Parameter, "Phosphate-phosphorus as P" = "Phosphorus"))

clean.8 %>% count(Parameter=="Phosphorus") #check, success

#Orthophosphate as PO4 and Orthophosphate as P are comparable

clean.8 %>% count(Parameter=="Orthophosphate as P04") #baseline

clean.8 <- clean.8 %>% mutate(Parameter = recode(Parameter, "Orthophosphate as PO4" = "Orthophosphate as P"))

clean.8 %>% count(Parameter=="Orthophosphate as P") #check, success

#need to merge Chlorophyll a and Chlorophyll a, free of pheophytin

clean.8 %>% count(str\_detect(Parameter, "^Chlorophyll a")) #baseline

clean.8 <- clean.8 %>% mutate(Parameter = recode(Parameter, "Chlorophyll a, free of pheophytin" = "Chlorophyll a"))

clean.8 %>% count(Parameter=="Chlorophyll a") #check, success

#Units unimportant, now that all parameters of my interest are in equivalent units

clean.10 <- select(clean.8, -"Depth.Unit", -"Parameter.Unit")

#rounding depth units so more are uniform when they are similar (the .3 m = surface, the 1 = subsurface, etc.)

clean.11 <- clean.10 %>% mutate(Depth = signif(Depth, digits = 1))

unique(clean.11$Depth)

#sorting parameters into new columns

clean.12 <- clean.11 %>% select(Date, Identifier, Station, Depth, Parameter, Parameter.Value) %>%

group\_by(Parameter) %>%

mutate(row = row\_number()) %>%

tidyr::pivot\_wider(names\_from = Parameter, values\_from = Parameter.Value) %>%

select(Date, Identifier, Station, Depth, -row, Phosphorus,

Ammonia,

pH,

Turbidity,

CHLA = `Chlorophyll a`,

DO = `Dissolved oxygen (DO)`,

Inorganic.N = `Inorganic nitrogen (nitrate and nitrite)`,

Kjedahl.N = `Kjeldahl nitrogen`,

Ortho.P = `Orthophosphate as P`,

Salinity,

Specific.Conductance = `Specific conductance`,

Temperature = `Temperature, water`)

#Skip clean.13 because at first I tried to select specific depths, but this filtered out too much important data, so I categorized the depths into surface, sub-surface, deep (bottom?), and will make note of this as a limitation when treating all the parameter values as if they were collected and quantified at the same depth, although the measurements could vary greatly due to depth of reading.

#Recoding all "not detected values" to halfway point between 0 and the LOD for later transformation if necessary to reduce # of zeroes in data

# Links to methods for quantification of each nutrient analyte:

# Ammonia https://www.nemi.gov/methods/method\_summary/5405/ Nitrogen, Ammonia (Colorimetric, Automated Phenate)

# Inorganic N https://www.nemi.gov/methods/method\_summary/4702/ Nitrogen, Nitrate-Nitrite (Colorimetric, Automated, Cadmium Reduction)

# Kjedahl N https://www.nemi.gov/methods/method\_summary/9626/ TKN by semi-automated colorimetry

# Ortho.P

# Chlorophyll A https://www.nemi.gov/methods/method\_summary/7222/

# Phosphorus https://www.nemi.gov/methods/method\_summary/4823/ Phosphorus, All Forms (Colorimetric, Automated, Ascorbic Acid)

clean.14 <- clean.12 %>% mutate(Ammonia = recode(Ammonia, 'ND' = '0.01'),

Inorganic.N = recode(Inorganic.N, 'ND' = '0.01'),

Kjedahl.N = recode(Kjedahl.N, 'ND' = '0.05'),

Ortho.P = recode(Ortho.P, 'ND' = '0.025'),

CHLA = recode(CHLA, 'ND' = "0.5"),

Salinity = recode(Salinity, 'ND' = "0.0"),

Phosphorus = recode(Phosphorus, 'ND' = "0.005"))

#assigning data types for all newly established columns (parameters)

clean.15 <- clean.14 %>% mutate(Date = as.Date(Date, "%Y-%m-%d"),

Identifier = as.character(Identifier),

Station = as.factor(Station),

Ammonia = as.numeric(Ammonia),

pH = as.numeric(pH),

Turbidity = as.numeric(Turbidity),

CHLA = as.numeric(CHLA),

DO = as.numeric(DO),

Phosphorus = as.numeric(Phosphorus),

Inorganic.N = as.numeric(Inorganic.N),

Kjedahl.N = as.numeric(Kjedahl.N),

Ortho.P = as.numeric(Ortho.P),

Salinity = as.numeric(Salinity),

Specific.Conductance = as.numeric(Specific.Conductance),

Temperature = as.numeric(Temperature),

Depth = as.numeric(Depth))

clean.15 <- clean.15 %>% filter(Depth < 200) #remove the outlier Depth of 200 (this must be an incorrect entry)

#merging all the lines which are overlapping by Date and Station

attempt.1 <- clean.15 %>% group\_by(Date, Station) %>% summarise\_all(~first(na.omit(.))) #row with first mention of value

attempt.2 <- clean.15 %>% group\_by(Date, Station) %>% summarise\_all(~max(na.omit(.))) #taking maximum value

attempt.3 <- clean.15 %>% group\_by(Date, Station) %>% summarise\_all(~mean(na.omit(.))) #taking mean of values

attempt.4 <- clean.15 %>% group\_by(Date, Station) %>% summarise\_at(vars(c("Phosphorus","Ammonia", "pH", "Turbidity", "CHLA", "DO", "Inorganic.N", "Kjedahl.N", "Ortho.P", "Salinity", "Specific.Conductance","Temperature", "Depth")), ~median(na.exclude(.))) #taking median of values per recommendation to use median rather than averages for non-parametric data-sets

#now assigning the groupings for depth

attempt.5 <- attempt.4 %>% mutate('Depth.Label' = case\_when(Depth <= 0.3 ~ 'Surface',

Depth <= 1.0 ~ 'Sub-Surface',

Depth > 1.0 ~ 'Depth'))

#-----------------------------------------------------------------------------------------------

#Adding in additional environmental parameters of interest (the Flow Rates per date (USGS) and also the main phytoplankton groups (NC-DEQ)).

# ADDING THE PHYTO GROUPS -- only from 1999 and beyond

chowan.phytos <- read.csv("PHYTOS.csv", header = TRUE, na.strings = c(""," ", ".", "NA"))

albemarle.phytos <- read.csv("PHYTOS.2.csv", header = TRUE, na.strings = c(""," ", ".", "NA"))

CR <- chowan.phytos %>% select(Station = Storet,

Date = DateCollected,

Group = AlgalGroup,

Genus,

Species,

Cell.Density,

Unit.Density,

Biovolume)

AS <- albemarle.phytos %>% select(Station = Storet,

Date = DateCollected,

Group = AlgalGroup,

Genus,

Species,

Cell.Density,

Unit.Density,

Biovolume)

#need to assign variable types

CR.AS <- rbind(CR,AS)

CR.AS.2 <- CR.AS %>% mutate(Station = as.factor(Station),

Date = as.Date(Date, "%Y-%m-%d"),

Group = as.factor(Group),

Genus = as.factor(Genus),

Species = as.factor(Species),

Cell.Density = as.numeric(Cell.Density),

Biovolume = as.numeric(Biovolume))

#selecting the 10 stations of interest (which overlap between availability of phytoplankton data and WQP data)

CR.AS.3 <- CR.AS.2 %>% filter(Station == "D6250000" | Station == "D8356200" | Station == "D8950000" | Station == "D9490000" | Station == "D999500C" | Station == "D999500N" | Station == "D999500S" | Station == "M390000C" | Station == "M610000C" | Station == "N9700000")

#this is where the current problem is happening

#merging CR.AS phyto data and physical water quality data based on the date and station

combo <- dplyr::left\_join(CR.AS.3, attempt.5, by = c("Date","Station"), keep = FALSE)

#READING IN THE FLOW DATA

USGS <- read.csv("Chowan\_Flow.csv", header = TRUE, na.strings = c(""," ", ".", "NA"))

flow <- USGS %>% select(Date = date, Flow.rate = Chowan.Flow..cfs.) %>% mutate(Date = as.Date(Date, "%Y-%m-%d"), Flow.rate = as.numeric(Flow.rate)) #This will need to be changed eventually because Nathan's flow rate data is approximate. Need to get it from USGS at some point.

#Keep in mind that the flow rate is reported in cfs ...

#now the goal is to merge this flow data into the bigger df by date and drop the values that don't match the dates of the DEQs sampling

combo.2 <- dplyr::left\_join(combo, flow, by = "Date", keep = FALSE, na\_matches = "never")

#I don't have most of the physical data from 2019 or the flow data from 2019 -- ideal to get this from NCDEQ and USGS

count(combo.2, complete.cases(combo.2)) #lots of incomplete cases due to Ortho.P

combo.3 <- combo.2 %>% select(c(-Ortho.P, -Depth.Label))

count(combo.3, complete.cases(combo.3))

combo.3 <- combo.3 %>% mutate(Year = as.numeric(format(Date,"%Y")))

#correcting any double entries

combo.3.1 <- combo.3 %>% group\_by(Station, Group, Genus, Species, Date) %>% summarise\_at(vars(c("Biovolume")), ~sum(na.exclude(.)))

combo.3.2 <- combo.3 %>% group\_by(Station, Group, Genus, Species, Date) %>% summarise\_at(vars(c("Phosphorus","Ammonia", "pH", "Turbidity", "CHLA", "DO", "Inorganic.N", "Kjedahl.N","Salinity", "Specific.Conductance","Temperature", "Depth","Flow.rate", "Year")), ~first(na.exclude(.)))

combo.3.3 <- dplyr::left\_join(combo.3.1, combo.3.2, by = c("Station","Group","Genus","Species","Date"), keep = FALSE, na\_matches = "never")

#Starting point data frame with all phytoplankton to species level

phyto.by.species <- combo.3.3

phyto.by.species <- phyto.by.species %>% mutate(exceeded.CHLA.std = case\_when(`CHLA` >= 40 ~ 1,`CHLA` < 40 ~ 0))

phyto.by.species <- phyto.by.species %>% mutate(log.biovolume = log(Biovolume+1))

#data frame with phytoplankton biovolume sorted by phylum

phyto.by.phylum.0 <- combo.3.3 %>% group\_by(Station,Group,Date) %>% summarise\_at(vars(c("Biovolume")), ~sum(na.exclude(.)))

phyto.by.phylum.1 <- combo.3.3 %>% group\_by(Station, Group, Date) %>% summarise\_at(vars(c("Phosphorus","Ammonia", "pH", "Turbidity", "CHLA", "DO", "Inorganic.N", "Kjedahl.N","Salinity", "Specific.Conductance","Temperature", "Depth","Flow.rate", "Year")), ~first(na.exclude(.)))

phyto.by.phylum <- dplyr::left\_join(phyto.by.phylum.0, phyto.by.phylum.1, by = c("Station","Group","Date"), keep = FALSE, na\_matches = "never")

phyto.by.phylum <- phyto.by.phylum %>% mutate(exceeded.CHLA.std = case\_when(`CHLA` >= 40 ~ 1, `CHLA` < 40 ~ 0) )

phyto.by.phylum <- phyto.by.phylum %>% mutate(log.biovolume = log(Biovolume+1))

#data frame with phytoplankton biovolume sorted by genus

phyto.by.genus.0 <- combo.3.3 %>% group\_by(Station,Group, Genus, Date) %>% summarise\_at(vars(c("Biovolume")), ~sum(na.exclude(.)))

phyto.by.genus.1 <- combo.3.3 %>% group\_by(Station, Group, Genus, Date) %>% summarise\_at(vars(c("Phosphorus","Ammonia", "pH", "Turbidity", "CHLA", "DO", "Inorganic.N", "Kjedahl.N","Salinity", "Specific.Conductance","Temperature", "Depth","Flow.rate", "Year")), ~first(na.exclude(.)))

phyto.by.genus <- dplyr::left\_join(phyto.by.genus.0, phyto.by.genus.1, by = c("Station","Group","Date"), keep = FALSE, na\_matches = "never")

phyto.by.genus <- phyto.by.genus %>% mutate(exceeded.CHLA.std = case\_when(`CHLA` >= 40 ~ 1, `CHLA` < 40 ~ 0) )

phyto.by.genus <- phyto.by.genus %>% mutate(log.biovolume = log(Biovolume+1))

#data frame with only cyanobacteria

cyanos.by.species <- combo.3.3 %>% filter(Group == "Cyanobacteria")

cyanos.by.species <- cyanos.by.species %>% mutate(exceeded.CHLA.std = case\_when(`CHLA` >= 40 ~ 1, `CHLA` < 40 ~ 0) )

cyanos.by.species <- cyanos.by.species %>% mutate(log.biovolume = log(Biovolume+1))

#data frame with only cyanobacteria by phylum

cyanos.by.phylum.0 <- combo.3.3 %>% filter(Group == "Cyanobacteria") %>% group\_by(Station,Group,Date) %>% summarise\_at(vars(c("Biovolume")), ~sum(na.exclude(.)))

cyanos.by.phylum.1 <- combo.3.3 %>% filter(Group == "Cyanobacteria") %>% group\_by(Station, Group, Date) %>% summarise\_at(vars(c("Phosphorus","Ammonia", "pH", "Turbidity", "CHLA", "DO", "Inorganic.N", "Kjedahl.N","Salinity", "Specific.Conductance","Temperature", "Depth","Flow.rate", "Year")), ~first(na.exclude(.)))

cyanos.by.phylum <- dplyr::left\_join(cyanos.by.phylum.0, cyanos.by.phylum.1, by = c("Station","Group","Date"), keep = FALSE, na\_matches = "never")

cyanos.by.phylum <- cyanos.by.phylum %>% mutate(exceeded.CHLA.std = case\_when(`CHLA` >= 40 ~ 1,`CHLA` < 40 ~ 0) )

cyanos.by.phylum <- cyanos.by.phylum %>% mutate(log.biovolume = log(Biovolume+1))

#data frame with only cyanobacteria by genus

cyanos.by.genus.0 <- combo.3.3 %>% filter(Group == "Cyanobacteria") %>% group\_by(Station,Group, Genus, Date) %>% summarise\_at(vars(c("Biovolume")), ~sum(na.exclude(.)))

cyanos.by.genus.1 <- combo.3.3 %>% filter(Group == "Cyanobacteria") %>% group\_by(Station, Group, Genus, Date) %>% summarise\_at(vars(c("Phosphorus","Ammonia", "pH", "Turbidity", "CHLA", "DO", "Inorganic.N", "Kjedahl.N","Salinity", "Specific.Conductance","Temperature", "Depth","Flow.rate", "Year")), ~first(na.exclude(.)))

cyanos.by.genus <- dplyr::left\_join(cyanos.by.genus.0, cyanos.by.genus.1, by = c("Station","Group","Genus","Date"), keep = FALSE, na\_matches = "never")

cyanos.by.genus <- cyanos.by.genus %>% mutate(exceeded.CHLA.std = case\_when(`CHLA` >= 40 ~ 1, `CHLA` < 40 ~ 0) )

cyanos.by.genus <- cyanos.by.genus %>% mutate(log.biovolume = log(Biovolume+1))

#data frame with cyanos grouped by genus characteristics (N.fix, Non.fix, Filamentous, MC.producers)

#adding N fixer / Non.Fixer as a category

cyanos.by.N.Fix.0 <- combo.3.3 %>% filter(Group == "Cyanobacteria") %>% mutate(N.Fixation = case\_when(

`Genus` == "Anabaena" ~ "N.fixer",

`Genus` == "Anabaenopsis" ~ "N.fixer",

`Genus` == "Aphanizomenon" ~ "N.fixer",

`Genus` == "Cylindrospermopsis" ~ "N.fixer",

`Genus` == "Raphidiopsis" ~ "N.fixer"))

cyanos.by.N.Fix.1 <- cyanos.by.N.Fix.0$N.Fixation %>% replace\_na("Non.fixer")

cyanos.by.N.Fix.2 <- cbind(cyanos.by.N.Fix.0, cyanos.by.N.Fix.1)

cyanos.by.N.Fix.3 <- cyanos.by.N.Fix.2 %>% select(-N.Fixation)

cyanos.by.N.Fix.3 <- rename(cyanos.by.N.Fix.3, "N.Fixation" = "...22")

cyanos.by.N.Fix.4 <- cyanos.by.N.Fix.3 %>% filter(N.Fixation == "N.fixer") %>% group\_by(Station, Group, Genus, Date) %>% summarise\_at(vars(c("Biovolume")), ~sum(na.exclude(.)))

cyanos.by.N.Fix.5 <- cyanos.by.N.Fix.3 %>% filter(N.Fixation == "N.fixer") %>% group\_by(Station, Group, Genus, Date) %>% summarise\_at(vars(c("Phosphorus","Ammonia", "pH", "Turbidity", "CHLA", "DO", "Inorganic.N", "Kjedahl.N","Salinity", "Specific.Conductance","Temperature", "Depth","Flow.rate", "Year")), ~first(na.exclude(.)))

cyanos.by.N.Fix <- dplyr::left\_join(cyanos.by.N.Fix.4, cyanos.by.N.Fix.5, by = c("Station","Group","Genus","Date"), keep = FALSE, na\_matches = "never")

cyanos.by.N.Fix <- cyanos.by.N.Fix %>% mutate(exceeded.CHLA.std = case\_when(`CHLA` >= 40 ~ 1,`CHLA` < 40 ~ 0) )

cyanos.by.N.Fix <- cyanos.by.N.Fix %>% mutate(log.biovolume = log(Biovolume+1))

#Non.Fixers

cyanos.by.Non.Fix.0 <- cyanos.by.N.Fix.3 %>% filter(N.Fixation == "Non.fixer") %>% group\_by(Station, Group, Genus, Date) %>% summarise\_at(vars(c("Biovolume")), ~sum(na.exclude(.)))

cyanos.by.Non.Fix.1 <- cyanos.by.N.Fix.3 %>% filter(N.Fixation == "Non.fixer") %>% group\_by(Station, Group, Genus, Date) %>% summarise\_at(vars(c("Phosphorus","Ammonia", "pH", "Turbidity", "CHLA", "DO", "Inorganic.N", "Kjedahl.N","Salinity", "Specific.Conductance","Temperature", "Depth","Flow.rate", "Year")), ~first(na.exclude(.)))

cyanos.by.Non.Fix <- dplyr::left\_join(cyanos.by.Non.Fix.0, cyanos.by.Non.Fix.1, by = c("Station","Group","Genus","Date"), keep = FALSE, na\_matches = "never")

cyanos.by.Non.Fix <- cyanos.by.Non.Fix %>% mutate(exceeded.CHLA.std = case\_when(`CHLA` >= 40 ~ 1, `CHLA` < 40 ~ 0) )

cyanos.by.Non.Fix <- cyanos.by.Non.Fix %>% mutate(log.biovolume = log(Biovolume+1))

#microcystin producers only

cyanos.by.MC.0 <- combo.3.3 %>% filter(Group == "Cyanobacteria") %>% mutate(MC = case\_when(`Genus` == "Aphanizomenon" ~ "MC\_producer",

`Genus` == "Anabaena" ~ "MC\_producer",

`Genus` == "Cylindrospermum" ~ "MC\_producer",

`Genus` == "Fischerella" ~ "MC\_producer",

`Genus` == "Gloeotrichia" ~ "MC\_producer",

`Genus` == "Haplosiphon" ~ "MC\_producer",

`Genus` == "Microcystis" ~ "MC\_producer",

`Genus` == "Anabaenopsis" ~ "MC\_producer",

`Genus` == "Nostoc" ~ "MC\_producer",

`Genus` == "Oscillatoria" ~ "MC\_producer",

`Genus` == "Phormidium" ~ "MC\_producer",

`Genus` == "Plantothrix" ~ "MC\_producer",

`Genus` == "Radiocystis" ~ "MC\_producer",

`Genus` == "Raphidiopsis" ~ "MC\_producer",

`Genus` == "Scytonema" ~ "MC\_producer",

`Genus` == "Umezakia" ~ "MC\_producer"))

cyanos.by.MC.1 <- cyanos.by.MC.0$MC %>% replace\_na("Non.MC")

cyanos.by.MC.2 <- cbind(cyanos.by.MC.0, cyanos.by.MC.1)

cyanos.by.MC.3 <- cyanos.by.MC.2 %>% select(-MC)

cyanos.by.MC.3 <- rename(cyanos.by.MC.3, "MC" = "...22")

cyanos.by.MC.4 <- cyanos.by.MC.3 %>% filter(MC == "MC\_producer") %>% group\_by(Station, Group, Genus, Date) %>% summarise\_at(vars(c("Biovolume")), ~sum(na.exclude(.)))

cyanos.by.MC.5 <- cyanos.by.MC.3 %>% filter(MC == "MC\_producer") %>% group\_by(Station, Group, Genus, Date) %>% summarise\_at(vars(c("Phosphorus","Ammonia", "pH", "Turbidity", "CHLA", "DO", "Inorganic.N", "Kjedahl.N","Salinity", "Specific.Conductance","Temperature", "Depth","Flow.rate", "Year")), ~first(na.exclude(.)))

cyanos.by.MC <- dplyr::left\_join(cyanos.by.MC.4, cyanos.by.MC.5, by = c("Station","Group","Genus","Date"), keep = FALSE, na\_matches = "never")

cyanos.by.MC <- cyanos.by.MC %>% mutate(exceeded.CHLA.std = case\_when(`CHLA` >= 40 ~ 1, `CHLA` < 40 ~ 0) )

cyanos.by.MC <- cyanos.by.MC %>% mutate(log.biovolume = log(Biovolume+1))

#Filamentous

filaments <- unique(cyanos.by.genus$Genus)

cyanos.by.Filament.0 <- combo.3.3 %>% filter(Group == "Cyanobacteria") %>% mutate(Filament = case\_when(`Genus` == "Pseudanabaena" ~ "Filamentous", `Genus` == "Anabaena" ~ "Filamentous",

`Genus` == "Raphidiopsis" ~ "Filamentous",

`Genus` == "Aphanizomenon" ~ "Filamentous",

`Genus` == "Cylindrospermopsis" ~ "Filamentous",

`Genus` == "Planktolyngbya" ~ "Filamentous",

`Genus` == "Anabaenopsis" ~ "Filamentous",

`Genus` == "Anabaenopsis" ~ "Filamentous",

`Genus` == "Leptolyngbya " ~ "Filamentous",))

cyanos.by.Filament.1 <- cyanos.by.Filament.0$Filament %>% replace\_na("Non.Filament")

cyanos.by.Filament.2 <- cbind(cyanos.by.Filament.0, cyanos.by.Filament.1)

cyanos.by.Filament.3 <- cyanos.by.Filament.2 %>% select(-Filament)

cyanos.by.Filament.3 <- rename(cyanos.by.Filament.3, "Filamentous" = "...22")

cyanos.by.Filament.4 <- cyanos.by.Filament.3 %>% filter(Filamentous == "Filamentous") %>% group\_by(Station, Group, Genus, Date) %>% summarise\_at(vars(c("Biovolume")), ~sum(na.exclude(.)))

cyanos.by.Filament.5 <- cyanos.by.Filament.3 %>% filter(Filamentous == "Filamentous") %>% group\_by(Station, Group, Genus, Date) %>% summarise\_at(vars(c("Phosphorus","Ammonia", "pH", "Turbidity", "CHLA", "DO", "Inorganic.N", "Kjedahl.N","Salinity", "Specific.Conductance","Temperature", "Depth","Flow.rate", "Year")), ~first(na.exclude(.)))

cyanos.by.Filament <- dplyr::left\_join(cyanos.by.Filament.4, cyanos.by.Filament.5, by = c("Station","Group","Genus","Date"), keep = FALSE, na\_matches = "never")

cyanos.by.Filament <- cyanos.by.N.Fix %>% mutate(exceeded.CHLA.std = case\_when(`CHLA` >= 40 ~ 1,`CHLA` < 40 ~ 0) )

cyanos.by.Filament <- cyanos.by.N.Fix %>% mutate(log.biovolume = log(Biovolume+1))

# -------- APPLYING ACTUAL GAMS -------------------------------------------------------------------------------------- #

# 9/28/2021 #

#GAM SERIES 1 -- ADDRESSING Q3 and Q4 from original NCSG/APNEP proposal: How does Salinity and Nitrogen enrichment impact microcystin production?

gam\_MC.0 <- gam(log.biovolume ~ s(Ammonia, k=9) + s(Kjedahl.N) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = cyanos.by.MC)

summary(gam\_MC.0)

gam.check(gam\_MC.0)

plot.gam(gam\_MC.0, residuals = TRUE, shift = coef(gam\_MC.0)[1], se = TRUE, shade = TRUE, pages=1)

#removing Kjeldahl N because association is due to cellular N, not anything causation related

gam\_MC.1 <- gam(log.biovolume ~ s(Ammonia, k=9) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = cyanos.by.MC)

summary(gam\_MC.1)

gam.check(gam\_MC.1)

plot.gam(gam\_MC.1, residuals = TRUE, shift = coef(gam\_MC.1)[1], se = TRUE, shade = TRUE, pages=1)

#but how does Kjeldahl N look with all phytoplankton biovolume?

gam\_all.1 <- gam(log.biovolume ~ s(Ammonia) + s(Kjedahl.N) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = phyto.by.genus)

summary(gam\_all.1)

gam.check(gam\_all.1)

plot.gam(gam\_all.1, residuals = TRUE, shift = coef(gam\_all.1)[1], se = TRUE, shade = TRUE, pages=1)

#now only using complete cases -- stick with this for ALL models. Discuss missing data with committee and Odum people

complete.MC <- cyanos.by.MC %>% drop\_na()

gam\_MC.2 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Kjedahl.N) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.MC)

summary(gam\_MC.2)

gam.check(gam\_MC.2)

plot.gam(gam\_MC.2, residuals = TRUE, shift = coef(gam\_MC.2)[1], se = TRUE, shade = TRUE, pages=1)

#without TKN

gam\_MC.3 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.MC)

summary(gam\_MC.3)

gam.check(gam\_MC.3)

plot.gam(gam\_MC.3, residuals = TRUE, shift = coef(gam\_MC.3)[1], se = TRUE, shade = TRUE, pages=1)

#the phosphorus relationship changes markedly... examine the phosphorus and kjeldahl N relationship? for concurvity

concurv <- concurvity(gam\_MC.2, full = TRUE)

options(digits = 2, scientific = F)

concurv

write.csv(concurv, "/Users/haleyplaas/OneDrive - University of North Carolina at Chapel Hill/Coding/R/Chowan Data/multi-year/excel//concurvity.output.csv") # <- more reason to remove Kjedahl.N from all of the models

gam\_MC.P <- gam(log.biovolume ~ s(Phosphorus), method = "REML", data = complete.MC)

summary(gam\_MC.P)

gam.check(gam\_MC.P)

plot.gam(gam\_MC.P, residuals = TRUE, shift = coef(gam\_MC.P)[1], se = TRUE, shade = TRUE, pages=1)

#TKN alone

gam\_MC.TKN <- gam(log.biovolume ~ s(Kjedahl.N), method = "REML", data = complete.MC)

summary(gam\_MC.TKN)

gam.check(gam\_MC.TKN)

plot.gam(gam\_MC.TKN, residuals = TRUE, shift = coef(gam\_MC.TKN)[1], se = TRUE, shade = TRUE, pages=1)

#TP and TKN together

gam\_MC.both <- gam(log.biovolume ~ s(Kjedahl.N) + s(Phosphorus), method = "REML", data = complete.MC)

summary(gam\_MC.both)

gam.check(gam\_MC.both)

plot.gam(gam\_MC.both, residuals = TRUE, shift = coef(gam\_MC.both)[1], se = TRUE, shade = TRUE, pages=1)

TKN.P.plot <- ggplot(complete.MC) + geom\_point(aes(y=Kjedahl.N, x=Phosphorus))

TKN.P.plot #more evidence of the clearly some interplay here between TKN and P measurements --> linear relationship between TKN and P, concurvity shows .999 relationship between the two -- extremely high relationship due to both methods quantifying the toxin within the cells

#but how does Kjeldahl N look with all cyanobacteria biovolume (complete cases only)?

complete.cyanos <- cyanos.by.genus %>% drop\_na()

gam\_all.2 <- gam(log.biovolume ~ s(Ammonia) + s(Kjedahl.N) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.cyanos)

summary(gam\_all.2)

gam.check(gam\_all.2)

plot.gam(gam\_all.2, residuals = TRUE, shift = coef(gam\_all.2)[1], se = TRUE, shade = TRUE, pages=1)

#removing TKN again

gam\_all.3 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.cyanos)

summary(gam\_all.3)

gam.check(gam\_all.3)

plot.gam(gam\_all.3, residuals = TRUE, shift = coef(gam\_all.3)[1], se = TRUE, shade = TRUE, pages=1)

#but how does Kjeldahl N look with all phyto biovolume (complete cases only)?

complete.phytos <- phyto.by.genus %>% drop\_na()

gam\_all.phytos.2 <- gam(log.biovolume ~ s(Ammonia) + s(Kjedahl.N) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.phytos)

summary(gam\_all.phytos.2)

gam.check(gam\_all.phytos.2)

plot.gam(gam\_all.phytos.2, residuals = TRUE, shift = coef(gam\_all.phytos.2)[1], se = TRUE, shade = TRUE, pages=1)

#removing TKN again

gam\_all.phytos.3 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.phytos)

summary(gam\_all.phytos.3)

gam.check(gam\_all.phytos.3)

plot.gam(gam\_all.phytos.3, residuals = TRUE, shift = coef(gam\_all.phytos.3)[1], se = TRUE, shade = TRUE, pages=1)

#Now moving onto playing with the cyano model with the nuts analytes and salinity predictors

#by phylum

complete.cyanos.p <- cyanos.by.phylum %>% drop\_na()

gam\_cyanos.p <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.cyanos.p)

summary(gam\_cyanos.p)

gam.check(gam\_cyanos.p)

plot.gam(gam\_cyanos.p, residuals = TRUE, shift = coef(gam\_cyanos.p)[1], se = TRUE, shade = TRUE, pages=1)

#by genus

gam\_all.3 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.cyanos)

summary(gam\_all.3)

gam.check(gam\_all.3)

plot.gam(gam\_all.3, residuals = TRUE, shift = coef(gam\_all.3)[1], se = TRUE, shade = TRUE, pages=1)

#by species

complete.cyanos.s <- cyanos.by.species %>% drop\_na()

gam\_cyanos.s <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.cyanos.s)

summary(gam\_cyanos.s)

gam.check(gam\_cyanos.s)

plot.gam(gam\_cyanos.s, residuals = TRUE, shift = coef(gam\_cyanos.s)[1], se = TRUE, shade = TRUE, pages=1)

#ultimately --trends follow the same patterns between taxonomic groupings -- so stick with genus in order to compare to models grouping by microcystin producing and N fixing genera

AICc(gam\_cyanos.s, gam\_cyanos.p, gam\_all.3)

#although when you compare the phylum vs genus vs species levels -- the phylum model is best

#Looking at specific interaction between Inorganic N and the genus due to Inorganic N being the most important predictor of cyanobacterial biomass

gam.within.g <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N, by = Genus) + s(Salinity), method = "REML", data = complete.cyanos)

summary(gam.within.g)

gam.check(gam.within.g)

plot.gam(gam.within.g, residuals = TRUE, shift = coef(gam.within.g)[1], se = TRUE, shade = TRUE, pages=1)

#do these trends in associations hold up for N fixers?

complete.cyanos.N <- cyanos.by.N.Fix %>% drop\_na()

#with TKN

gam\_cyanos.N.1 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Kjedahl.N) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.cyanos.N)

summary(gam\_cyanos.N.1)

gam.check(gam\_cyanos.N.1)

plot.gam(gam\_cyanos.N.1, residuals = TRUE, shift = coef(gam\_cyanos.N.1)[1], se = TRUE, shade = TRUE, pages=1)

#without TKN

gam\_cyanos.N.2 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.cyanos.N)

summary(gam\_cyanos.N.2)

gam.check(gam\_cyanos.N.2)

plot.gam(gam\_cyanos.N.2, residuals = TRUE, shift = coef(gam\_cyanos.N.2)[1], se = TRUE, shade = TRUE, pages=1)

#do these trends in associations hold up for Non N fixers?

complete.cyanos.Non <- cyanos.by.Non.Fix %>% drop\_na()

#with TKN

gam\_cyanos.Non.1 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Kjedahl.N) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.cyanos.Non)

summary(gam\_cyanos.Non.1)

gam.check(gam\_cyanos.Non.1)

plot.gam(gam\_cyanos.Non.1, residuals = TRUE, shift = coef(gam\_cyanos.Non.1)[1], se = TRUE, shade = TRUE, pages=1)

#without TKN

gam\_cyanos.Non.2 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Salinity), method = "REML", data = complete.cyanos.Non)

summary(gam\_cyanos.Non.2)

gam.check(gam\_cyanos.Non.2)

plot.gam(gam\_cyanos.Non.2, residuals = TRUE, shift = coef(gam\_cyanos.Non.2)[1], se = TRUE, shade = TRUE, pages=1)

# GAM series #2 Looking at ALL predictors (not just nutrient analytes and salinity) with all phyto plankton

#phosphorus

gam\_phos <- gam(log.biovolume ~ s(Phosphorus), method = "REML", data = phyto.by.phylum)

summary(gam\_phos)

plot.gam(gam\_phos, residuals = TRUE, pch=1, cex=1, shift = coef(gam\_phos)[1], se = TRUE, shade = TRUE)

#ammonia

gam\_amm <- gam(log.biovolume ~ s(Ammonia), method = "REML", data = phyto.by.phylum)

summary(gam\_amm)

plot.gam(gam\_amm, residuals = TRUE, pch=1, cex=1, shift = coef(gam\_phos)[1], se = TRUE, shade = TRUE) #the ammonia data is super zero heavy

#inorganic.N

gam\_N <- gam(log.biovolume ~ s(Inorganic.N), method = "REML", data = phyto.by.phylum)

summary(gam\_N)

plot.gam(gam\_N, residuals = TRUE, pch=1, cex=1, shift = coef(gam\_phos)[1], se = TRUE, shade = TRUE)

#turbidity

gam\_turb <- gam(log.biovolume ~ s(Turbidity), method = "REML", data = phyto.by.phylum)

summary(gam\_turb)

plot.gam(gam\_turb, residuals = TRUE, shift = coef(gam\_turb)[1], se = TRUE, shade = TRUE, pages=1)

#all nuts

gam\_nuts <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Turbidity), method = "REML", data = phyto.by.phylum)

summary(gam\_nuts)

plot.gam(gam\_nuts, residuals = TRUE, pch=1, cex=1, shift = coef(gam\_phos)[1], se = TRUE, shade = TRUE, pages=1)

plot.gam(gam\_nuts, shift = coef(gam\_phos)[1], se = TRUE, shade = TRUE, pages=1)

#chem GAM

gam\_chem <- gam(log.biovolume ~ s(pH) + s(Salinity, Station, bs = "fs") + s(DO), method = "REML", data = phyto.by.phylum)

summary(gam\_chem)

plot.gam(gam\_chem, shift = coef(gam\_chem)[1], se = TRUE, shade = TRUE, pages=1)

#physical GAM

gam\_phys <- gam(log.biovolume ~ s(Flow.rate), method = "REML", data = phyto.by.phylum)

summary(gam\_phys)

plot.gam(gam\_phys, shift = coef(gam\_phys)[1], se = TRUE, shade = TRUE, pages=1, xlim = c(0,10000))

#CHLA

gam\_chla <- gam(log.biovolume ~ s(CHLA), method = "REML", data = phyto.by.phylum)

summary(gam\_chla)

plot.gam(gam\_chla, shift = coef(gam\_chla)[1], se = TRUE, shade = TRUE, pages=1, xlim = c(0,150))

#looking at individual factors with cyanobacterial biomass

#making sure to use the complete.cyanos df

cgam\_phos <- gam(log.biovolume ~ s(Phosphorus), method = "REML", data = complete.cyanos)

summary(cgam\_phos)

plot(complete.cyanos$Phosphorus, complete.cyanos$log.biovolume)

#AMMONIA

cgam\_amm <- gam(log.biovolume ~ s(Ammonia), method = "REML", data = complete.cyanos)

summary(cgam\_amm)

plot(complete.cyanos$Ammonia, complete.cyanos$log.biovolume)

#INORGANIC.N

cgam\_N <- gam(log.biovolume ~ s(Inorganic.N), method = "REML", data = complete.cyanos)

summary(cgam\_N)

plot(complete.cyanos$Inorganic.N, complete.cyanos$log.biovolume)

#KJELDAHL

cgam\_Kjeldahl <- gam(log.biovolume ~ s(Kjedahl.N), method = "REML", data = complete.cyanos)

summary(cgam\_Kjeldahl)

plot(complete.cyanos$Kjedahl.N, complete.cyanos$log.biovolume)

#TURBIDITY

cgam\_turb <- gam(log.biovolume ~ s(Turbidity), method = "REML", data = complete.cyanos)

summary(cgam\_turb)

plot(complete.cyanos$Turbidity, complete.cyanos$log.biovolume)

#CHLA

cgam\_chla <- gam(log.biovolume ~ s(CHLA), method = "REML", data = complete.cyanos)

summary(cgam\_chla)

plot(complete.cyanos$CHLA, complete.cyanos$log.biovolume)

plot.gam(cgam\_chla, residuals = TRUE, shift = coef(cgam\_chla)[1], se = TRUE, shade = TRUE, pages=1)

#pH

cgam\_pH <- gam(log.biovolume ~ s(pH), method = "REML", data = complete.cyanos)

summary(cgam\_pH)

plot(complete.cyanos$pH, complete.cyanos$log.biovolume)

plot.gam(cgam\_pH, residuals = TRUE, shift = coef(cgam\_pH)[1], se = TRUE, shade = TRUE, pages=1)

#DO

cgam\_DO <- gam(log.biovolume ~ s(DO), method = "REML", data = complete.cyanos)

summary(cgam\_DO)

plot(complete.cyanos$DO, complete.cyanos$log.biovolume)

#Salinity

cgam\_salt <- gam(log.biovolume ~ s(Salinity), method = "REML", data = complete.cyanos)

summary(cgam\_salt)

plot(complete.cyanos$Salinity, complete.cyanos$log.biovolume)

#FLOW

cgam\_flow <- gam(log.biovolume ~ s(Flow.rate), method = "REML", data = complete.cyanos)

summary(cgam\_flow)

plot(complete.cyanos$Flow.rate, complete.cyanos$log.biovolume, xlim = c(0,10000))

#TEMP

cgam\_temp <- gam(log.biovolume ~ s(Temperature), method = "REML", data = complete.cyanos)

summary(cgam\_temp)

plot(complete.cyanos$Temperature, complete.cyanos$log.biovolume)

#model selection

AICc(cgam\_amm, cgam\_chla, cgam\_DO, cgam\_flow, cgam\_Kjeldahl, cgam\_N, cgam\_pH, cgam\_phos, cgam\_salt, cgam\_temp, cgam\_turb)

#looking at individual factors with microcystin producer biomass

#making sure to use complete.MC

mcgam\_phos <- gam(log.biovolume ~ s(Phosphorus), method = "REML", data = complete.MC)

summary(mcgam\_phos)

plot(complete.MC$Phosphorus, complete.MC$log.biovolume)

#AMMONIA

mcgam\_amm <- gam(log.biovolume ~ s(Ammonia, k = 9), method = "REML", data = complete.MC)

summary(mcgam\_amm)

plot(complete.MC$Ammonia, complete.MC$log.biovolume, xlim = c(0,0.15)) #most ammonia values are <BLD...

#INORGANIC.N

mcgam\_N <- gam(log.biovolume ~ s(Inorganic.N), method = "REML", data = complete.MC)

summary(mcgam\_N)

plot(complete.MC$Inorganic.N, complete.MC$log.biovolume)

#KJELDAHL

mcgam\_Kjeldahl <- gam(log.biovolume ~ s(Kjedahl.N), method = "REML", data = complete.MC)

summary(mcgam\_Kjeldahl)

plot(complete.MC$Kjedahl.N, complete.MC$log.biovolume)

plot.gam(mcgam\_Kjeldahl, residuals = TRUE, shift = coef(mcgam\_Kjeldahl)[1], se = TRUE, shade = TRUE, pages=1)

#TURBIDITY

mcgam\_turb <- gam(log.biovolume ~ s(Turbidity), method = "REML", data = complete.MC)

summary(mcgam\_turb)

plot(complete.MC$Turbidity, complete.MC$log.biovolume)

#CHLA

mcgam\_chla <- gam(log.biovolume ~ s(CHLA), method = "REML", data = complete.MC)

summary(mcgam\_chla)

plot(complete.MC$CHLA, complete.MC$log.biovolume)

plot.gam(mcgam\_chla, residuals = TRUE, shift = coef(mcgam\_chla)[1], se = TRUE, shade = TRUE, pages=1)

#pH

mcgam\_pH <- gam(log.biovolume ~ s(pH), method = "REML", data = complete.MC)

summary(mcgam\_pH)

plot(complete.MC$pH, complete.MC$log.biovolume)

#DO

mcgam\_DO <- gam(log.biovolume ~ s(DO), method = "REML", data = complete.MC)

summary(mcgam\_DO)

plot(complete.MC$DO, complete.MC$log.biovolume)

#Salinity

mcgam\_salt <- gam(log.biovolume ~ s(Salinity), method = "REML", data = complete.MC)

summary(mcgam\_salt)

plot(complete.MC$Salinity, complete.MC$log.biovolume)

#FLOW

mcgam\_flow <- gam(log.biovolume ~ s(Flow.rate), method = "REML", data = complete.MC)

summary(mcgam\_flow)

plot(complete.MC$Flow.rate, complete.MC$log.biovolume, xlim = c(0,10000))

#TEMP

mcgam\_temp <- gam(log.biovolume ~ s(Temperature), method = "REML", data = complete.MC)

summary(mcgam\_temp)

plot(complete.MC$Temperature, complete.MC$log.biovolume)

plot.gam(mcgam\_temp, residuals = TRUE, shift = coef(mcgam\_temp)[1], se = TRUE, shade = TRUE, pages=1)

#model selection

AICc(mcgam\_amm, mcgam\_chla, mcgam\_DO, mcgam\_flow, mcgam\_Kjeldahl, mcgam\_N, mcgam\_pH, mcgam\_phos, mcgam\_salt, mcgam\_temp, mcgam\_turb)

# --- MORE COMPLEX GAMS, ADDING MORE THAN ONE PREDICTOR VARIABLE ---

# REMOVING TKN AND CHLA FROM ALL MODELS -- DUE TO CONCURVITIES TO EXAMINE WEAKER ASSOCIATIONS WITH OTHER PREDICTORS -- NEED TO DO THIS

#combined at the phylum level

gam\_phytos <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = phyto.by.phylum)

summary(gam\_phytos)

gam.check(gam\_phytos)

plot.gam(gam\_phytos, residuals = TRUE, shift = coef(gam\_phytos)[1], se = TRUE, shade = TRUE, pages=1)

AICc(gam\_phytos, gam\_phos, gam\_amm, gam\_N, gam\_turb, gam\_nuts, gam\_chem, gam\_phys, gam\_chla)

#now trying with genus level

gam\_phyto\_genus <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate) , method = "REML", data = phyto.by.genus)

summary(gam\_phyto\_genus)

gam.check(gam\_phyto\_genus)

plot.gam(gam\_phyto\_genus, residuals = TRUE, shift = coef(gam\_phyto\_genus)[1], se = TRUE, shade = TRUE, pages=1)

AICc(gam\_phyto\_genus, gam\_phytos)

#now trying with species level

gam\_phyto\_species <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N)+ s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = phyto.by.species)

summary(gam\_phyto\_species)

gam.check(gam\_phyto\_species)

plot.gam(gam\_phyto\_species, residuals = TRUE, shift = coef(gam\_phyto\_species)[1], se = TRUE, shade = TRUE, pages=1)

AICc(gam\_phyto\_species, gam\_phyto\_genus, gam\_phytos)

#cyanos only at phylum level

gam\_cyano\_phylum <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.phylum)

summary(gam\_cyano\_phylum)

gam.check(gam\_cyano\_phylum)

plot.gam(gam\_cyano\_phylum, residuals = TRUE, shift = coef(gam\_cyano\_phylum)[1], se = TRUE, shade = TRUE, pages=1)

AICc(gam\_cyano\_phylum, gam\_phyto\_species, gam\_phyto\_genus, gam\_phytos)

#cyanos only at genus level -- sort by at some point to see if genus is fixed factor? some genera may specifically be influenced and there is a way to test this

gam\_cyano\_genus <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.genus)

summary(gam\_cyano\_genus)

gam.check(gam\_cyano\_genus)

plot.gam(gam\_cyano\_genus, residuals = TRUE, shift = coef(gam\_cyano\_genus)[1], se = TRUE, shade = TRUE, pages=1)

AICc(gam\_cyano\_phylum, gam\_cyano\_genus)

#cyanos only at species level

gam\_cyano\_species <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.species)

summary(gam\_cyano\_species)

gam.check(gam\_cyano\_species)

plot.gam(gam\_cyano\_species, residuals = TRUE, shift = coef(gam\_cyano\_species)[1], se = TRUE, shade = TRUE, pages=1)

AICc(gam\_cyano\_phylum, gam\_cyano\_genus, gam\_cyano\_species)

#cyanos only with N fixers

gam\_cyano\_Nfix <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.N.Fix)

summary(gam\_cyano\_Nfix)

gam.check(gam\_cyano\_Nfix)

plot.gam(gam\_cyano\_Nfix, residuals = TRUE, shift = coef(gam\_cyano\_Nfix)[1], se = TRUE, shade = TRUE, pages=1)

AICc(gam\_cyano\_phylum, gam\_cyano\_genus, gam\_cyano\_species, gam\_cyano\_Nfix)

#cyanos only with NON N fixers

gam\_cyano\_Nonfix <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.Non.Fix)

summary(gam\_cyano\_Nonfix)

gam.check(gam\_cyano\_Nonfix)

plot.gam(gam\_cyano\_Nonfix, residuals = TRUE, shift = coef(gam\_cyano\_Nonfix)[1], se = TRUE, shade = TRUE, pages=1)

AICc(gam\_cyano\_phylum, gam\_cyano\_genus, gam\_cyano\_species, gam\_cyano\_Nfix, gam\_cyano\_Nonfix)

#cyanos only with Filamentous (these are basically the same as the N fixers in this system)

gam\_cyano\_filament <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.Filament)

summary(gam\_cyano\_filament)

gam.check(gam\_cyano\_filament)

plot.gam(gam\_cyano\_filament, residuals = TRUE, shift = coef(gam\_cyano\_filament)[1], se = TRUE, shade = TRUE, pages=1)

AICc(gam\_cyano\_filament, gam\_cyano\_phylum, gam\_cyano\_genus, gam\_cyano\_species, gam\_cyano\_Nfix, gam\_cyano\_Nonfix)

#cyanos but only microcystin producers

gam\_cyano\_MC <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.MC)

summary(gam\_cyano\_MC)

gam.check(gam\_cyano\_MC)

plot.gam(gam\_cyano\_MC, residuals = TRUE, shift = coef(gam\_cyano\_MC)[1], se = TRUE, shade = TRUE, pages=1)

AICc(gam\_cyano\_MC, gam\_cyano\_filament, gam\_cyano\_phylum, gam\_cyano\_genus, gam\_cyano\_species, gam\_cyano\_Nfix, gam\_cyano\_Nonfix)

#Plotting GAMS for difference in trends

#first thing to do is improve the 5 selected models

## all phytos by genera, all cyanos by genera, n fixers, non n fixers, and microcystin producers

#1. phyto by genera

gam\_phyto\_genus.0 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = phyto.by.genus)

summary(gam\_phyto\_genus.0)

gam.check(gam\_phyto\_genus.0)

plot.gam(gam\_phyto\_genus.0, residuals = TRUE, shift = coef(gam\_phyto\_genus.0)[1], se = TRUE, shade = TRUE, pages=1)

concurv <- concurvity(gam\_phyto\_genus.0, full = FALSE)

options(digits = 2, scientific = F)

concurv

write.csv(concurv, "/Users/haleyplaas/OneDrive - University of North Carolina at Chapel Hill/Coding/R/Chowan Data/multi-year/excel//concurvity.output.csv")

#with all phytos, it appears that

##1.turbidity is concurve with flow rate (this makes ecological sense, more water flow leads to resuspension of sediments)

##2.turbidity is concurve with phosphorus (TP) -- ask the group about this ecological significance. Could this be due to resuspension of phosphorus adsorbed to sediments? or because TP is tied up in algae (which blooms elevate turbidity).

##3. CHLA is concurve with TP. this could be due to the method measuring the TP tied up in algae?

##4. DO and T are concurve. This is due to colder water being able to hold more DO -- do a quick plot showing them?

plot(x = phyto.by.genus$Temperature, y = phyto.by.genus$DO) #confirmed, negative linear relationship. Need to show interactions between these in the model.

gam\_phyto\_genus.1 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(pH) + s(Salinity)+ s(CHLA) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = phyto.by.genus) #do away with Turbidity, show interactions between DO and Temp.

summary(gam\_phyto\_genus.1)

gam.check(gam\_phyto\_genus.1)

plot.gam(gam\_phyto\_genus.1, residuals = TRUE, shift = coef(gam\_phyto\_genus.1)[1], se = TRUE, shade = TRUE, pages=1)

AIC(gam\_phyto\_genus.0, gam\_phyto\_genus.1)

#take away interaction (just show T)

gam\_phyto\_genus.2 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(pH) + s(Salinity)+ s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = phyto.by.genus) #do away with Turbidity, show interactions between DO and Temp.

summary(gam\_phyto\_genus.2)

gam.check(gam\_phyto\_genus.2)

plot.gam(gam\_phyto\_genus.2, residuals = TRUE, shift = coef(gam\_phyto\_genus.2)[1], se = TRUE, shade = TRUE, pages=1)

AIC(gam\_phyto\_genus.0, gam\_phyto\_genus.1, gam\_phyto\_genus.2)

#showing that the over-fit is even more prominent -- try just adding the interaction between DO and Temperature

gam\_phyto\_genus.3 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(pH) + s(CHLA) + s(Turbidity) + s(Salinity) + s(DO, Temperature) + s(Flow.rate), method = "REML", data = phyto.by.genus)

summary(gam\_phyto\_genus.3)

gam.check(gam\_phyto\_genus.3)

plot.gam(gam\_phyto\_genus.3, residuals = TRUE, shift = coef(gam\_phyto\_genus.3)[1], se = TRUE, shade = TRUE, pages=1) #this is the best fit so far -- but need to discuss other limitations with the concurvities in the meeting

AIC(gam\_phyto\_genus.0, gam\_phyto\_genus.1, gam\_phyto\_genus.2, gam\_phyto\_genus.3)

#gam\_phyto\_genus.3

#1.cyanos by genera

gam\_cyano\_genus.0 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.genus)

summary(gam\_cyano\_genus.0)

gam.check(gam\_cyano\_genus.0)

plot.gam(gam\_cyano\_genus.0, residuals = TRUE, shift = coef(gam\_cyano\_genus.0)[1], se = TRUE, shade = TRUE, pages=1)

concurv <- concurvity(gam\_cyano\_genus.0, full = FALSE)

options(digits = 2, scientific = F)

concurv

write.csv(concurv, "/Users/haleyplaas/OneDrive - University of North Carolina at Chapel Hill/Coding/R/Chowan Data/multi-year/excel//concurvity.output.csv")

#this shows concurvity between P and CHLA, Turbidity and flow rate, DO and T (same as phyto model)

gam\_cyano\_genus.1 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity)+ s(CHLA) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.genus)

summary(gam\_cyano\_genus.1)

gam.check(gam\_cyano\_genus.1)

plot.gam(gam\_cyano\_genus.1, residuals = TRUE, shift = coef(gam\_cyano\_genus.1)[1], se = TRUE, shade = TRUE, pages=1)

AIC(gam\_cyano\_genus.0, gam\_cyano\_genus.1)

gam\_cyano\_genus.2 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(pH) + s(Salinity) + s(Temperature, DO) + s(CHLA) + s(Flow.rate), method = "REML", data = cyanos.by.genus)

summary(gam\_cyano\_genus.2)

gam.check(gam\_cyano\_genus.2)

plot.gam(gam\_cyano\_genus.2, residuals = TRUE, shift = coef(gam\_cyano\_genus.2)[1], se = TRUE, shade = TRUE, pages=1)

AIC(gam\_cyano\_genus.0, gam\_cyano\_genus.1, gam\_cyano\_genus.2) #getting rid of turbidity but keeping CHLA seems to be working

#gam\_cyano\_genus.2

#now moving onto N fixers

gam\_cyano\_Nfix.0 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.N.Fix)

summary(gam\_cyano\_Nfix.0)

gam.check(gam\_cyano\_Nfix.0)

plot.gam(gam\_cyano\_Nfix.0, residuals = TRUE, shift = coef(gam\_cyano\_Nfix.0)[1], se = TRUE, shade = TRUE, pages=1)

concurv <- concurvity(gam\_cyano\_Nfix.0, full = FALSE)

options(digits = 2, scientific = F)

concurv

write.csv(concurv, "/Users/haleyplaas/OneDrive - University of North Carolina at Chapel Hill/Coding/R/Chowan Data/multi-year/excel//concurvity.output.csv")

#this shows concurvity between P and CHLA, Turbidity and flow rate, DO and T (same as phyto model)

#concurvity between temperature, DO and Inorganic N, as well as CHLA and P

gam\_cyano\_Nfix.1 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(CHLA) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.N.Fix)

summary(gam\_cyano\_Nfix.1)

gam.check(gam\_cyano\_Nfix.1)

plot.gam(gam\_cyano\_Nfix.1, residuals = TRUE, shift = coef(gam\_cyano\_Nfix.1)[1], se = TRUE, shade = TRUE, pages=1)

AIC(gam\_cyano\_Nfix.0, gam\_cyano\_Nfix.1)

#removing CHLA

gam\_cyano\_Nfix.2 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.N.Fix)

summary(gam\_cyano\_Nfix.2)

gam.check(gam\_cyano\_Nfix.2)

plot.gam(gam\_cyano\_Nfix.2, residuals = TRUE, shift = coef(gam\_cyano\_Nfix.2)[1], se = TRUE, shade = TRUE, pages=1)

AIC(gam\_cyano\_Nfix.0, gam\_cyano\_Nfix.1, gam\_cyano\_Nfix.2)

#gam\_cyano\_Nfix.1

#now moving onto NON N fixers

gam\_cyano\_Non.fix.0 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.Non.Fix)

summary(gam\_cyano\_Non.fix.0)

gam.check(gam\_cyano\_Non.fix.0)

plot.gam(gam\_cyano\_Non.fix.0, residuals = TRUE, shift = coef(gam\_cyano\_Non.fix.0)[1], se = TRUE, shade = TRUE, pages=1)

concurv <- concurvity(gam\_cyano\_Non.fix.0, full = FALSE)

options(digits = 2, scientific = F)

concurv

write.csv(concurv, "/Users/haleyplaas/OneDrive - University of North Carolina at Chapel Hill/Coding/R/Chowan Data/multi-year/excel//concurvity.output.csv")

#this shows concurvity between P and CHLA, Turbidity and flow rate, flow rate and pH, T and DO

plot(x = cyanos.by.Non.Fix$Phosphorus, y = cyanos.by.Non.Fix$CHLA) #confirmed, positive linear relationship

gam\_cyano\_Non.fix.1 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(pH) + s(Salinity) + s(CHLA) + s(Temperature, DO) + s(Flow.rate, Turbidity), method = "REML", data = cyanos.by.Non.Fix)

summary(gam\_cyano\_Non.fix.1)

gam.check(gam\_cyano\_Non.fix.1)

plot.gam(gam\_cyano\_Non.fix.1, residuals = TRUE, shift = coef(gam\_cyano\_Non.fix.1)[1], se = TRUE, shade = TRUE, pages=1)

AIC(gam\_cyano\_Non.fix.0, gam\_cyano\_Non.fix.1)

#gam\_cyano\_Non.fix.1

#now moving onto microcystin producers

gam\_cyano\_MC.0 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate), method = "REML", data = cyanos.by.MC)

summary(gam\_cyano\_MC.0)

gam.check(gam\_cyano\_MC.0)

plot.gam(gam\_cyano\_MC.0, residuals = TRUE, shift = coef(gam\_cyano\_MC.0)[1], se = TRUE, shade = TRUE, pages=1)

concurv <- concurvity(gam\_cyano\_MC.0, full = FALSE)

options(digits = 2, scientific = F)

concurv

write.csv(concurv, "/Users/haleyplaas/OneDrive - University of North Carolina at Chapel Hill/Coding/R/Chowan Data/multi-year/excel//concurvity.output.csv")

#this shows concurvity between DO and N and T, and CHLA and P

gam\_cyano\_MC.1 <- gam(log.biovolume ~ s(Ammonia, k=8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(CHLA) + s(Temperature, DO, k=8) + s(Flow.rate), method = "REML", data = cyanos.by.MC)

summary(gam\_cyano\_MC.1)

gam.check(gam\_cyano\_MC.1)

plot.gam(gam\_cyano\_MC.1, residuals = TRUE, shift = coef(gam\_cyano\_MC.1)[1], se = TRUE, shade = TRUE, pages=1)

AIC(gam\_cyano\_MC.1, gam\_cyano\_MC.0)

#gam\_cyano\_MC.1

#plotting the 5 models:

Phyto\_genus <- plot.gam(gam\_phyto\_genus.3, shift = coef(gam\_phyto\_genus.3)[1], se = TRUE, shade = TRUE, shade.col = "palegreen1", pages=1)

cyano\_genus <- plot.gam(gam\_cyano\_genus.2, shift = coef(gam\_cyano\_genus.2)[1], se = TRUE, shade = TRUE, shade.col = "paleturquoise", pages=1)

cyano\_N\_fixer <- plot.gam(gam\_cyano\_Nfix.1, shift = coef(gam\_cyano\_Nfix.1)[1], se = TRUE, shade = TRUE, shade.col = "seashell1", pages=1)

cyano\_Non\_fixer <- plot.gam(gam\_cyano\_Non.fix.1, shift = coef(gam\_cyano\_Non.fix.1)[1], se = TRUE, shade = TRUE, shade.col = "thistle1", pages=1)

cyano\_MC <- plot.gam(gam\_cyano\_MC.1, shift = coef(gam\_cyano\_MC)[1], shade.col = "lightsteelblue1", se = TRUE, shade = TRUE, pages=1)

#showing co-linearities / relationships between predictors

ggplot(phyto.by.species, aes(x = Phosphorus, y = CHLA)) + geom\_point() + ylim(0,100)

#models without CHLA -- better fit, but overfit according to AICc? but also shows the association between the other environmental parameters and biovolume better

#phytos by genus without CHLA

gam\_phyto\_genus.no.chla <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(pH) + s(Turbidity) + s(Salinity) + s(DO, Temperature) + s(Flow.rate), method = "REML", data = phyto.by.genus)

summary(gam\_phyto\_genus.no.chla)

gam.check(gam\_phyto\_genus.no.chla)

plot.gam(gam\_phyto\_genus.no.chla, residuals = TRUE, shift = coef(gam\_phyto\_genus.no.chla)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus without CHLA

gam\_cyano\_genus.no.chla <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.genus)

summary(gam\_cyano\_genus.no.chla)

gam.check(gam\_cyano\_genus.no.chla)

plot.gam(gam\_cyano\_genus.no.chla, residuals = TRUE, shift = coef(cyano\_genus.no.chla)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus N fixers only without CHLA

gam\_cyano\_N.Fix.no.chla <- gam(log.biovolume ~ s(Ammonia, k = 8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.N.Fix)

summary(gam\_cyano\_N.Fix.no.chla)

gam.check(gam\_cyano\_N.Fix.no.chla)

plot.gam(gam\_cyano\_N.Fix.no.chla, residuals = TRUE, shift = coef(gam\_cyano\_N.Fix.no.chla)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus Non N fixers only without CHLA

gam\_cyano\_Non.Fix.no.chla <- gam(log.biovolume ~ s(Ammonia, k = 8) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.Non.Fix)

summary(gam\_cyano\_Non.Fix.no.chla)

gam.check(gam\_cyano\_Non.Fix.no.chla)

plot.gam(gam\_cyano\_Non.Fix.no.chla, residuals = TRUE, shift = coef(gam\_cyano\_Non.Fix.no.chla)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus MC producers only without CHLA

gam\_cyano\_MC.no.chla <- gam(log.biovolume ~ s(Ammonia, k = 7) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.MC)

summary(gam\_cyano\_MC.no.chla)

gam.check(gam\_cyano\_MC.no.chla)

plot.gam(gam\_cyano\_MC.no.chla, residuals = TRUE, shift = coef(gam\_cyano\_MC.no.chla)[1], se = TRUE, shade = TRUE, pages=1)

#Does using complete cases only change trends?

#the issue here is removing more rows leads to even less data which is problematic for the amount of predictors I have plugged into the model -- but trying all complete cases only

#phyto by genus complete cases 10531 -> 6623

complete.phyto.genus <- na.omit(phyto.by.genus)

gam\_phyto.complete <- gam(log.biovolume ~ s(Ammonia, k = 7) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = complete.phyto.genus)

summary(gam\_phyto.complete)

gam.check(gam\_phyto.complete)

#cyanos by genus complete cases 571 -> 363

complete.cyano.genus <- na.omit(cyanos.by.genus)

gam\_cyano.complete <- gam(log.biovolume ~ s(Ammonia, k = 7) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.genus)

summary(gam\_cyano.complete)

gam.check(gam\_cyano.complete) #even with complete cases only, the nutrient analytes seem to be the issue with the fit of the model

#cyanos by N fixers complete cases 167 -> 104

complete.cyano.N.fix <- na.omit(cyanos.by.N.Fix)

gam\_cyano.N.Fix.complete <- gam(log.biovolume ~ s(Ammonia, k = 7) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = complete.cyano.N.fix)

summary(gam\_cyano.N.Fix.complete)

gam.check(gam\_cyano.N.Fix.complete) #again nut analytes causing problems

#cyanos by NON fixers complete cases 404 -> 259

complete.cyano.Non.fix <- na.omit(cyanos.by.Non.Fix)

gam\_cyano.Non.Fix.complete <- gam(log.biovolume ~ s(Ammonia, k = 7) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = complete.cyano.Non.fix)

summary(gam\_cyano.Non.Fix.complete)

gam.check(gam\_cyano.Non.Fix.complete) #again nut analytes causing problems

#cyanos by MC complete cases 158 -> 96

complete.MC <- na.omit(cyanos.by.MC)

gam\_cyano\_MC.complete <- gam(log.biovolume ~ s(Ammonia, k = 7) + s(Phosphorus) + s(Inorganic.N) + s(Kjedahl.N) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = complete.MC)

summary(gam\_cyano\_MC.complete)

gam.check(gam\_cyano\_MC.complete)

#More indication that the nutrient data is not a good predictor of biomass! so let's take it out of the models and see what happens

#phytos by genus without NUTS

gam\_phyto\_genus.no.nuts <- gam(log.biovolume ~ s(pH) + s(Turbidity) + s(Salinity) + s(DO, Temperature) + s(Flow.rate) + s(CHLA), method = "REML", data = phyto.by.genus)

summary(gam\_phyto\_genus.no.nuts)

gam.check(gam\_phyto\_genus.no.nuts)

plot.gam(gam\_phyto\_genus.no.nuts, residuals = TRUE, shift = coef(gam\_phyto\_genus.no.nuts)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus without NUTS

gam\_cyano\_genus.no.NUTS <- gam(log.biovolume ~ s(CHLA) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.genus)

summary(gam\_cyano\_genus.no.NUTS)

gam.check(gam\_cyano\_genus.no.NUTS)

plot.gam(gam\_cyano\_genus.no.NUTS, residuals = TRUE, shift = coef(cyano\_genus.no.NUTS)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus N fixers only without NUTS

gam\_cyano\_N.Fix.no.NUTS <- gam(log.biovolume ~ s(CHLA)+ s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.N.Fix)

summary(gam\_cyano\_N.Fix.no.NUTS)

gam.check(gam\_cyano\_N.Fix.no.NUTS)

plot.gam(gam\_cyano\_N.Fix.no.NUTS, residuals = TRUE, shift = coef(gam\_cyano\_N.Fix.no.NUTS)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus Non N fixers only without NUTS

gam\_cyano\_Non.Fix.no.NUTS <- gam(log.biovolume ~ s(CHLA) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.Non.Fix)

summary(gam\_cyano\_Non.Fix.no.NUTS)

gam.check(gam\_cyano\_Non.Fix.no.NUTS)

plot.gam(gam\_cyano\_Non.Fix.no.NUTS, residuals = TRUE, shift = coef(gam\_cyano\_Non.Fix.no.NUTS)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus MC producers only without NUTS

gam\_cyano\_MC.no.NUTS <- gam(log.biovolume ~ s(CHLA) + s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.MC)

summary(gam\_cyano\_MC.no.NUTS)

gam.check(gam\_cyano\_MC.no.NUTS)

plot.gam(gam\_cyano\_MC.no.NUTS, residuals = TRUE, shift = coef(gam\_cyano\_MC.no.NUTS)[1], se = TRUE, shade = TRUE, pages=1)

#to reduce number of predictors so that there are not an excessive amount of basis functions per small data sets (esp with the specific cyano groups)

#phytos by genus without BOTH CHLA & NUTS

gam\_phyto\_genus.both <- gam(log.biovolume ~ s(pH) + s(Turbidity) + s(Salinity) + s(DO, Temperature) + s(Flow.rate), method = "REML", data = phyto.by.genus)

summary(gam\_phyto\_genus.both)

gam.check(gam\_phyto\_genus.both)

plot.gam(gam\_phyto\_genus.both, residuals = TRUE, shift = coef(gam\_phyto\_genus.both)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus without BOTH CHLA & NUTS

gam\_cyano\_genus.both <- gam(log.biovolume ~ s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.genus)

summary(gam\_cyano\_genus.both)

gam.check(gam\_cyano\_genus.both)

plot.gam(gam\_cyano\_genus.both, residuals = TRUE, shift = coef(cyano\_genus.both)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus N fixers only without BOTH CHLA & NUTS

gam\_cyano\_N.Fix.both <- gam(log.biovolume ~ s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.N.Fix)

summary(gam\_cyano\_N.Fix.both)

gam.check(gam\_cyano\_N.Fix.both)

plot.gam(gam\_cyano\_N.Fix.both, residuals = TRUE, shift = coef(gam\_cyano\_N.Fix.both)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus Non N fixers only without BOTH CHLA & NUTS

gam\_cyano\_Non.Fix.both <- gam(log.biovolume ~ s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.Non.Fix)

summary(gam\_cyano\_Non.Fix.both)

gam.check(gam\_cyano\_Non.Fix.both)

plot.gam(gam\_cyano\_Non.Fix.both, residuals = TRUE, shift = coef(gam\_cyano\_Non.Fix.both)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by genus MC producers only without BOTH CHLA & NUTS

gam\_cyano\_MC.both <- gam(log.biovolume ~ s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.MC)

summary(gam\_cyano\_MC.both)

gam.check(gam\_cyano\_MC.both)

plot.gam(gam\_cyano\_MC.both, residuals = TRUE, shift = coef(gam\_cyano\_MC.both)[1], se = TRUE, shade = TRUE, pages=1)

#COMPARING ALL THESE MODELS USING AICc rather than AIC because it corrects for the small number of data points for the large k value (number of predictors / basis functions)

AICc(gam\_phytos, gam\_phyto\_genus, gam\_phyto.complete, gam\_phyto\_genus.no.chla, gam\_phyto\_genus.no.nuts, gam\_phyto\_genus.both) #the first seems to be the best fit -- but reminder that when just sorting by phylum the model is probably not directly comparable to the ones all sorted by genus

summary(gam\_phytos)

AICc(gam\_cyano\_phylum, gam\_cyano\_genus, gam\_cyano\_genus.both, gam\_cyano.complete, gam\_cyano\_genus.no.chla, gam\_cyano\_genus.no.NUTS) #best fit here is original and then without NUTS

summary(gam\_cyano\_phylum)

AICc(gam\_cyano\_Nfix, gam\_cyano.N.Fix.complete, gam\_cyano\_N.Fix.both, gam\_cyano\_N.Fix.no.chla, gam\_cyano\_N.Fix.no.NUTS) #best fit here is with no NUTS

summary(gam\_cyano\_N.Fix.no.NUTS)

AICc(gam\_cyano\_Nonfix, gam\_cyano.Non.Fix.complete, gam\_cyano\_Non.Fix.no.NUTS, gam\_cyano\_Non.Fix.no.chla, gam\_cyano\_Non.Fix.both) #best fit here, original

summary(gam\_cyano\_Nonfix)

AICc(gam\_cyano\_MC, gam\_cyano\_MC.complete, gam\_cyano\_MC.no.chla, gam\_cyano\_MC.no.NUTS, gam\_cyano\_MC.both) #original and without NUTS have the same fit

summary(gam\_cyano\_MC.no.NUTS)

gam.check(gam\_cyano\_MC.no.NUTS)

#Nuts are one of the most important factors, trying model with suspected most important factors

gam\_phyto\_genus.both <- gam(log.biovolume ~ s(pH) + s(Turbidity) + s(Salinity) + s(DO, Temperature) + s(Flow.rate), method = "REML", data = phyto.by.genus)

summary(gam\_phyto\_genus.both)

gam.check(gam\_phyto\_genus.both)

plot.gam(gam\_phyto\_genus.both, residuals = TRUE, shift = coef(gam\_phyto\_genus.both)[1], se = TRUE, shade = TRUE, pages=1)

#cyano by phylum

gam\_cyano\_phylum <- gam(log.biovolume ~ s(Turbidity) + s(pH) + s(Salinity) + s(Temperature, DO) + s(Flow.rate), method = "REML", data = cyanos.by.phylum)

summary(gam\_cyano\_phylum)

gam.check(gam\_cyano\_genus.both)

plot.gam(gam\_cyano\_genus.both, residuals = TRUE, shift = coef(cyano\_genus.both)[1], se = TRUE, shade = TRUE, pages=1)

# Moving into some visualization of the data aside from the GAMs

# first task is to visualize concentration of the three nutrient analytes at each station

# will need to average each analyte over each year grouped by station and analyte

# then make a stacked bar chart showing each relative contribution

# then if feeling really spicy, place a pie chart overlayed an actual map showing the stations with pie chart size relative to mass of nuts

#data frame to work with is combo.3

nuts.viz.0 <- combo.3 %>% select(Station, Date, Phosphorus, Ammonia, Inorganic.N, Kjedahl.N, CHLA)

nuts.viz.1 <- nuts.viz.0 %>% group\_by(Date, Station) %>% summarise\_all(~first(na.omit(.)))

nuts.viz.2 <- na.omit(nuts.viz.1)

nuts.viz.3 <- nuts.viz.2 %>% group\_by(Station) %>% summarise\_all(~mean(na.omit(.)))

#there are stations missing -- should give 10

?pivot\_longer

nuts.viz.4 <- pivot\_longer(nuts.viz.3, cols = c(Phosphorus, Ammonia, Inorganic.N, Kjedahl.N), names\_to = "Nut.Analyte")

nuts.plot <- ggplot(nuts.viz.4, aes(fill = Nut.Analyte, y=value, x=Station)) +

geom\_bar(position="stack", stat="identity") + theme\_half\_open()

nuts.plot

#NOTE -- JUST WENT THROUGH EVERYTHING AND ADDED KJELDEHL NITROGEN. NEED TO RE-RUN AND RE-EVAL EVERYTHING WITH THIS ADDED!

#EVERYTHING BELOW HERE IS OLD !!! BUT HAS NOT BEEN EVALUATED FOR POTENTIAL USE YET -----------------------------------------------------

gam\_cyanos.1 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Turbidity) + s(pH) + s(Salinity, Station, bs = "fs") + s(DO) + s(CHLA) + s(Temperature) + s(Flow.rate) + cyano\_groups, method = "REML", data = cyanos.3)

gam.check(gam\_cyanos.1)

summary(gam\_cyanos.1)

rounded <- concurvity(gam\_cyanos.1, full = FALSE)

options(digits = 2, scientific = F)

rounded

write.csv(rounded, "/Users/haleyplaas/OneDrive - University of North Carolina at Chapel Hill/Coding/R/Chowan Data/multi-year/excel//concurvity.output.csv")

#salinity had many issues with concurvity -- taking out interactions with station #this improved the model the most

#chla and phosphorus were correlated (0.95) -- taking out CHLA as it is result of increased biomass not a contributing factor (this made the model slightly worse)

#all of these one at a time made the model worse -- best bet is simply removing the station interaction with salinity

#keeping chla, phosphorus, DO, and Temp in the model but noted their concurvity/co-linearity

gam\_cyanos.2 <- gam(log.biovolume ~ s(Ammonia) + s(Phosphorus) + s(Inorganic.N) + s(Turbidity) + s(pH) + s(Salinity) + s(DO) + s(Temperature) + s(CHLA) + s(Flow.rate) + cyano\_groups, method = "REML", data = cyanos.3)

gam.check(gam\_cyanos.2)

summary <- summary(gam\_cyanos.2)

AIC(gam\_cyanos.1, gam\_cyanos.2)

rounded <- concurvity(gam\_cyanos.2, full = FALSE)

options(digits = 2, scientific = F)

write.csv(rounded, "/Users/haleyplaas/OneDrive - University of North Carolina at Chapel Hill/Coding/R/Chowan Data/multi-year/excel//concurvity.output.2.csv")

plot.gam(gam\_cyanos.2, residuals = TRUE, shift = coef(gam\_all)[1], se = TRUE, shade = TRUE, pages=1)

#visualizing key parameters

ggplot(cyanos.2, aes(x = Year, y = Biovolume, color = Temperature))+ geom\_jitter()

hist(cyanos.1$Temperature)

#visualization

gam.collage <- gam(Biovolume~Year+s(Phosphorus)+s(Ammonia, k = 9)+s(Turbidity)+s(pH)+s(CHLA)+s(DO)+s(Inorganic.N)+s(Salinity)+s(Temperature)+s(Flow.rate), method = "REML", data=cyanos.2)

vars <- c("Phosphorus","Ammonia","Turbidity","pH","CHLA","DO","Inorganic.N","Salinity","Temperature","Flow.rate")

map(vars, function(x){

p <- plotGAM(gam.collage, smooth.cov = x)

summary(gam.collage)

g <- ggplotGrob(p)

}) %>%

{grid.arrange(grobs = (.), ncol = 4, nrow = 4)}

#N-fixers only

N.fixer.gam <- gam(Biovolume~Year+s(Phosphorus)+s(Ammonia, k=5)+s(Turbidity)+s(pH)+s(CHLA)+s(DO)+s(Inorganic.N)+s(Salinity)+s(Temperature)+s(Flow.rate), method = "REML", data=N2.fixers)

vars <- c("Phosphorus","Ammonia","Turbidity","pH","CHLA","DO","Inorganic.N","Salinity","Temperature","Flow.rate")

map(vars, function(x){p <- plotGAM(N.fixer.gam, smooth.cov = x)

summary(N.fixer.gam)

g <- ggplotGrob(p)}) %>% {

grid.arrange(grobs = (.), ncol = 4, nrow = 4)}

#Non-N fixers only

Non.N2.fixer.gam <- gam(Biovolume~Year+s(Phosphorus)+s(Ammonia, k=5)+s(Turbidity)+s(pH)+s(CHLA)+s(DO)+s(Inorganic.N)+s(Salinity)+s(Temperature)+s(Flow.rate), method = "REML", data=Non.N2.fixers)

vars <- c("Phosphorus","Ammonia","Turbidity","pH","CHLA","DO","Inorganic.N","Salinity","Temperature","Flow.rate")

map(vars, function(x){p <- plotGAM(Non.N2.fixer.gam, smooth.cov = x)

summary(Non.N2.fixer.gam)

g <- ggplotGrob(p)}) %>% {

grid.arrange(grobs = (.), ncol = 4, nrow = 4)}

#All phytos

phyto.gam <- gam(Biovolume~Year+s(Phosphorus)+s(Ammonia, k=5)+s(Turbidity)+s(pH)+s(CHLA)+s(DO)+s(Inorganic.N)+s(Salinity)+s(Temperature)+s(Flow.rate), method = "REML", data=combo.3)

vars <- c("Phosphorus","Ammonia","Turbidity","pH","CHLA","DO","Inorganic.N","Salinity","Temperature","Flow.rate")

map(vars, function(x){p <- plotGAM(phyto.gam, smooth.cov = x)

summary(phyto.gam)

g <- ggplotGrob(p)}) %>% {

grid.arrange(grobs = (.), ncol = 4, nrow = 4)}

>>>>>>> Stashed changes