SEACAR Discrete Water Quality Analysis: Sample Surface Total Phosphorus

Last compiled on 24 June, 2022

Contents

Important Notes **1**

Libraries **2**

File Import **2**

Data Filtering and Data Impacted by Specific Value Qualifiers **3**

Managed Area Statistics **6**

Monitoring Location Statistics **8**

Seasonal Kendall Tau Analysis **8**

Appendix I: Scatter Plot of Entire Dataset **13**

Appendix II: Dataset Summary Box Plots **15**

Appendix III: Excluded Managed Areas **21**

Appendix IV: Managed Area Trendlines **23**

Appendix V: Managed Area Summary Box Plots **67**

Important Notes

All scripts and outputs can be found on the SEACAR GitHub repository:

https://github.com/FloridaSEACAR/SEACAR_Panzik

Note: The top 2% of data is excluded when computing mean and standard deviations in plotting sections solely for the purpose of getting y-axis scales. The exclusion of the top 2% is not used in any statistics that are exported.

Libraries

Loads libraries used in the script. The inclusion of `scipen` option limits how frequently R defaults to scientific notation.

```
library(knitr)
library(data.table)
library(plyr)
library(dplyr)
library(lubridate)
library(ggplot2)
library(ggpubr)
library(scales)
library(EnvStats)
library(tidyr)
library(stringr)
library(kableExtra)

windowsFonts(`Segoe UI` = windowsFont('Segoe UI'))
options(scipen=999)
opts_chunk$set(warning=FALSE, message=FALSE, dpi=200)
```

File Import

Imports file that is determined in the WC_Discrete_parameter_ReportCompile.R script.

The command `fread` is used because of its improved speed while handling large data files. Only columns that are used by the script are imported from the file, and are designated in the `select` input.

The script then gets the name of the parameter as it appears in the data file, units of the parameter, sets the `SampleDate` as a date object, and creates various scales of the date to be used by plotting functions.

```
#MA_All <- fread(here::here("WQ_Discrete/data/ManagedArea.csv"), sep = ",", header = TRUE, stringsAsFac
                #na.strings = "")

#file_in <- "C:/Users/steph/Dropbox/SEACAR_Panzik/SEACAR_Panzik/WQ_Discrete/data/Combined_WQ_WC_NUT_Wat
data <- fread(file_in, sep="|", header=TRUE, stringsAsFactors=FALSE,
              select=c("ManagedAreaName", "ProgramID", "ProgramName",
                       "ProgramLocationID", "SampleDate", "Year", "Month",
                       "RelativeDepth", "ActivityType", "ParameterName",
                       "ResultValue", "ParameterUnits", "ValueQualifier",
                       "SEACAR_QAQCFlagCode", "Include"), na.strings="")

activity <- activity
depth <- depth
parameter <- unique(data$ParameterName)
unit <- unique(data$ParameterUnits)
# activity <- unique(data$ActivityType)
# depth <- unique(data$RelativeDepth)
data$SampleDate <- as.Date(data$SampleDate)
data$YearMonth <- paste0(data$Month, "-", data$Year)
data$YearMonthDec <- data$Year + ((data$Month-0.5) / 12)
data$DecDate <- decimal_date(data$SampleDate)
```

```
data[, `:=` (relyear = Year - min(Year), relyear_dd = DecDate - min(DecDate)), by = "ManagedAreaName"]
data <- data[ParameterName == parameter & str_detect(ActivityType, activity) & RelativeDepth == depth &
```

Data Filtering and Data Impacted by Specific Value Qualifiers

Most data filtering is performed on export from the database, and is indicated by the `Include` variable. `Include` values of 1 indicate the data should be used for analysis, values of 0 indicate the data should not be used for analysis. Documentation on the database filtering is provided here: SEACAR Documentation-Analysis Filters and Calculations.docx

The filtering that is performed by the script at this point removes rows that are missing values for `ResultValue`, and only keeps data that is measured at the relative depth (surface, bottom, etc.) and activity type (field or sample) of interest. This is partly handled on export with the `RelativeDepth` variable, but there are some measurements that are considered both surface and bottom based on measurement depth and total depth. By default, these are marked as `Surface` for `RelativeDepth` and receive a `SEACAR_QAQCFlag` indicator of `12Q`. Data passes the filtering the process if it is from the correct depth and has an `Include` value of 1. The script also only looks at data of the desired `ActivityType` which indicates whether it was measured in the field (`Field`) or in the lab (`Sample`).

After the initial filtering, a second filter variable is created to determine whether enough time is represented in the managed area, which is that each managed area has 10 year or more of unique year entries for observation that pass the initial filter. If data passes the first set of filtering criteria and the time criteria, they are used in the analysis.

After filtering, the amount of data impacted by the H (for dissolved oxygen & pH in program 476), I, Q, S (for Secchi depth), and U value qualifiers. A variable is also created that determines if scatter plot points should be a different color based on value qualifiers of interest.

```
# param_name <- "Water_Temperature"
# out_dir <- here::here("WQ_Discrete/output/by_parameter/")
# APP_Plots <- TRUE

if(depth=="Bottom"){
   data$RelativeDepth[grep("12Q", data$SEACAR_QAQCFlagCode[
      data$RelativeDepth=="Surface"])] <- "Bottom"
}

data$Include <- as.logical(data$Include)
data$Include[grep("H", data$ValueQualifier[data$ProgramID==476])] <- TRUE
data <- data[!is.na(data$ResultValue),]

if(param_name!="Secchi_Depth"){
   data <- data[!is.na(data$RelativeDepth),]
   data <- data[data$RelativeDepth==depth,]
}

if(length(grep("Blank", data$ActivityType))>0){
   data <- data[-grep("Blank", data$ActivityType),]
}

if(param_name=="Chlorophyll_a_uncorrected_for_pheophytin" |
   param_name=="Salinity" | param_name=="Turbidity"){
   data <- data[grep(activity, data$ActivityType[!is.na(data$ActivityType)]),]
```

```
}

if(param_name=="Water_Temperature"){
   data <- data[data$ResultValue>=-2,]
} else{
   data <- data[data$ResultValue>=0,]
}


data <- merge.data.frame(MA_All[,c("AreaID", "ManagedAreaName")],
                         data, by="ManagedAreaName", all=TRUE)

MA_Summ <- data %>%
   group_by(AreaID, ManagedAreaName) %>%
   dplyr::summarize(ParameterName=parameter,
             RelativeDepth=depth,
             ActivityType=activity,
             N_Data=length(ResultValue[Include==TRUE & !is.na(ResultValue)]),
             N_Years=length(unique(Year[Include==TRUE & !is.na(Year)])),
             EarliestYear=min(Year[Include==TRUE]),
             LatestYear=max(Year[Include==TRUE]),
             SufficientData=ifelse(N_Data>0 & N_Years>=10, TRUE, FALSE))

data <- merge.data.frame(data, MA_Summ[,c("ManagedAreaName", "SufficientData")],
                         by="ManagedAreaName")

data$Use_In_Analysis <- ifelse(data$Include==TRUE & data$SufficientData==TRUE,
                               TRUE, FALSE)

MA_Summ <- MA_Summ %>%
   select(AreaID, ManagedAreaName, ParameterName, RelativeDepth, ActivityType,
          SufficientData, everything())
MA_Summ <- as.data.frame(MA_Summ[order(MA_Summ$ManagedAreaName), ])


total <- length(data$Include)
pass_filter <- length(data$Include[data$Include==TRUE])

count_H <- length(grep("H", data$ValueQualifier[data$ProgramID==476]))
perc_H <- 100*count_H/length(data$ValueQualifier)

count_I <- length(grep("I", data$ValueQualifier))
perc_I <- 100*count_I/length(data$ValueQualifier)

count_Q <- length(grep("Q", data$ValueQualifier))
perc_Q <- 100*count_Q/length(data$ValueQualifier)

count_S <- length(grep("S", data$ValueQualifier))
perc_S <- 100*count_S/length(data$ValueQualifier)

count_U <- length(grep("U", data$ValueQualifier))
perc_U <- 100*count_U/length(data$ValueQualifier)
```

```r
data$VQ_Plot <- data$ValueQualifier

inc_H <- ifelse(param_name=="pH" | param_name=="Dissolved_Oxygen" |
                   param_name=="Dissolved_Oxygen_Saturation", TRUE, FALSE)

if (inc_H==TRUE){
   data$VQ_Plot <- gsub("[^HU]+", "", data$VQ_Plot)
   data$VQ_Plot <- gsub("UH", "HU", data$VQ_Plot)
   data$VQ_Plot[na.omit(data$ProgramID!=476)] <- gsub("[^U]+", "",
                                          data$VQ_Plot[na.omit(data$ProgramID!=476)])
   data$VQ_Plot[data$VQ_Plot==""] <- NA


   cat(paste0("Number of Measurements: ", total,
              ", Number Passed Filter: ", pass_filter, "\n",
              "Program 476 H Codes: ", count_H, " (", round(perc_H, 6), "%)\n",
              "I Codes: ", count_I, " (", round(perc_I, 6), "%)\n",
              "Q Codes: ", count_Q, " (", round(perc_Q, 6), "%)\n",
              "U Codes: ", count_U, " (", round(perc_U, 6), "%)"))
} else if (param_name=="Secchi_Depth") {
   count_S <- length(grep("S", data$ValueQualifier))
   perc_S <- 100*count_S/length(data$ValueQualifier)
   data$VQ_Plot <- gsub("[^SU]+", "", data$VQ_Plot)
   data$VQ_Plot <- gsub("US", "SU", data$VQ_Plot)
   data$VQ_Plot[data$VQ_Plot==""] <- NA
   cat(paste0("Number of Measurements: ", total,
              ", Number Passed Filter: ", pass_filter, "\n",
              "I Codes: ", count_I, " (", round(perc_I, 6), "%)\n",
              "Q Codes: ", count_Q, " (", round(perc_Q, 6), "%)\n",
              "S Codes: ", count_S, " (", round(perc_S, 6), "%)\n",
              "U Codes: ", count_U, " (", round(perc_U, 6), "%)"))
} else{
   data$VQ_Plot <- gsub("[^U]+", "", data$VQ_Plot)
   data$VQ_Plot[data$VQ_Plot==""] <- NA
   cat(paste0("Number of Measurements: ", total,
              ", Number Passed Filter: ", pass_filter, "\n",
              "I Codes: ", count_I, " (", round(perc_I, 6), "%)\n",
              "Q Codes: ", count_Q, " (", round(perc_Q, 6), "%)\n",
              "U Codes: ", count_U, " (", round(perc_U, 6), "%)"))
}
```

```
## Number of Measurements: 95204, Number Passed Filter: 95204
## I Codes: 11403 (11.977438%)
## Q Codes: 1225 (1.286711%)
## U Codes: 2876 (3.020881%)
```

```r
data_summ <- data %>%
   group_by(AreaID, ManagedAreaName) %>%
   dplyr::summarize(ParameterName=parameter,
             RelativeDepth=depth,
             ActivityType=activity,
             N_Total=length(ResultValue),
             N_AnalysisUse=length(ResultValue[SufficientData==TRUE]),
```

```
            N_H=length(grep("H", data$ValueQualifier[data$ProgramID==476])),
            perc_H=100*N_H/length(data$ValueQualifier),
            N_I=length(grep("I", data$ValueQualifier)),
            perc_I=100*N_I/length(data$ValueQualifier),
            N_Q=length(grep("Q", data$ValueQualifier)),
            perc_Q=100*N_Q/length(data$ValueQualifier),
            N_S=length(grep("S", data$ValueQualifier)),
            perc_S=100*N_S/length(data$ValueQualifier),
            N_U=length(grep("U", data$ValueQualifier)),
            perc_U=100*N_U/length(data$ValueQualifier))

data_summ <- as.data.table(data_summ[order(data_summ$ManagedAreaName), ])
fwrite(data_summ, paste0(out_dir,"/", param_name, "_", activity, "_", depth,
                         "_DataSummary.csv"), sep=",")
rm(data_summ)
MA_Include <- MA_Summ$ManagedAreaName[MA_Summ$SufficientData==TRUE &
                                         MA_Summ$N_Data<2000000]
n <- length(MA_Include)
MA_Exclude <- MA_Summ[MA_Summ$N_Years<10 & MA_Summ$N_Years>0,]
MA_Exclude <- MA_Exclude[,c("ManagedAreaName", "N_Years")]
z <- nrow(MA_Exclude)
setDT(data)
```

Managed Area Statistics

Gets summary statistics for each managed area. Excluded managed areas are not included into whether the data should be used or not. Uses piping from dplyr package to feed into subsequent steps. The following steps are performed:

1. Take the `data` variable and only include rows that have a `SufficientData` value of TRUE
2. Group data that have the same `ManagedAreaName`, `Year`, and `Month`.
   - Second summary statistics do not use the `Month` grouping and are only for `ManagedAreaName` and `Year`.
   - Third summary statistics do not use `Year` grouping and are only for `ManagedAreaName` and `Month`
3. For each group, provide the following information: Number of Entries (N), Lowest Value (Min), Largest Value (Max), Median, Mean, Standard Deviation, and a list of all Program IDs included in these measurements.
4. Sort the data in ascending (A to Z and 0 to 9) order based on `ManagedAreaName` then `Year` then `Month`
5. Write summary stats to a pipe-delimited .txt file in the output directory
   - Click this text to open Git directory with output files

```
MA_YM_Stats <- data[data$Use_In_Analysis==TRUE, ] %>%
   group_by(AreaID, ManagedAreaName, Year, Month) %>%
   dplyr::summarize(ParameterName=parameter,
            RelativeDepth=depth,
            ActivityType=activity,
            N_Data=length(ResultValue),
            Min=min(ResultValue),
            Max=max(ResultValue),
            Median=median(ResultValue),
```

```r
             Mean=mean(ResultValue),
             StandardDeviation=sd(ResultValue),
             ProgramIDs=paste(sort(unique(ProgramID), decreasing=FALSE),
                              collapse=', '))
MA_YM_Stats <- as.data.table(MA_YM_Stats[order(MA_YM_Stats$ManagedAreaName,
                                               MA_YM_Stats$Year,
                                               MA_YM_Stats$Month), ])
fwrite(MA_YM_Stats, paste0(out_dir,"/", param_name, "_", activity, "_", depth,
                           "_ManagedArea_YearMonth_Stats.txt"), sep="|")
rm(MA_YM_Stats)

MA_Y_Stats <- data[data$Use_In_Analysis==TRUE, ] %>%
   group_by(AreaID, ManagedAreaName, Year) %>%
   dplyr::summarize(ParameterName=parameter,
             RelativeDepth=depth,
             ActivityType=activity,
             N=length(ResultValue),
             Min=min(ResultValue),
             Max=max(ResultValue),
             Median=median(ResultValue),
             Mean=mean(ResultValue),
             StandardDeviation=sd(ResultValue),
             ProgramIDs=paste(sort(unique(ProgramID), decreasing=FALSE),
                              collapse=', '))
MA_Y_Stats <- as.data.table(MA_Y_Stats[order(MA_Y_Stats$ManagedAreaName,
                                             MA_Y_Stats$Year), ])
fwrite(MA_Y_Stats, paste0(out_dir,"/", param_name, "_", activity, "_", depth,
                          "_ManagedArea_Year_Stats.txt"), sep="|")
rm(MA_Y_Stats)

MA_M_Stats <- data[data$Use_In_Analysis==TRUE, ] %>%
   group_by(AreaID, ManagedAreaName, Month) %>%
   dplyr::summarize(ParameterName=parameter,
             RelativeDepth=depth,
             ActivityType=activity,
             N=length(ResultValue),
             Min=min(ResultValue),
             Max=max(ResultValue),
             Median=median(ResultValue),
             Mean=mean(ResultValue),
             StandardDeviation=sd(ResultValue),
             ProgramIDs=paste(sort(unique(ProgramID), decreasing=FALSE),
                              collapse=', '))
MA_M_Stats <- as.data.table(MA_M_Stats[order(MA_M_Stats$ManagedAreaName,
                                             MA_M_Stats$Month), ])
fwrite(MA_M_Stats, paste0(out_dir,"/", param_name, "_", activity, "_", depth,
                          "_ManagedArea_Month_Stats.txt"), sep="|")
#rm(MA_M_Stats)
```

Monitoring Location Statistics

Gets monitoring location statistics, which is defined as a unique combination of `ManagedAreaName`, `ProgramID`, `ProgramAreaName`, and `ProgramLocationID`, using piping from dplyr package. The following steps are performed:

1. Take the `data` variable and only include rows that have a `SufficientData` value of `TRUE`
2. Group data that have the same `ManagedAreaName`, `ProgramID`, `ProgramName`, and `ProgramLocationID`.
3. For each group, provide the following information: Earliest Sample Date (EarliestSampleDate), Latest Sample Date (LastSampleDate), Number of Entries (N), Lowest Value (Min), Largest Value (Max), Median, Mean, and Standard Deviation.
4. Sort the data in ascending (A to Z and 0 to 9) order based on `ManagedAreaName` then `ProgramName` then `ProgramID` then `ProgramLocationID`
5. Write summary stats to a pipe-delimited .txt file in the output directory
    - Click this text to open Git directory with output files

```
Mon_Stats <- data[data$Use_In_Analysis==TRUE, ] %>%
   group_by(AreaID, ManagedAreaName, ProgramID, ProgramName, ProgramLocationID) %>%
   dplyr::summarize(ParameterName=parameter,
             RelativeDepth=depth,
             ActivityType=activity,
             EarliestSampleDate=min(SampleDate),
             LastSampleDate=max(SampleDate),
             N=length(ResultValue),
             Min=min(ResultValue),
             Max=max(ResultValue),
             Median=median(ResultValue),
             Mean=mean(ResultValue),
             StandardDeviation=sd(ResultValue))

Mon_Stats <- as.data.table(Mon_Stats[order(Mon_Stats$ManagedAreaName,
                                           Mon_Stats$ProgramName,
                                           Mon_Stats$ProgramID,
                                           Mon_Stats$ProgramLocationID), ])
fwrite(Mon_Stats, paste0(out_dir,"/", param_name, "_", activity, "_", depth,
                         "_MonitoringLoc_Stats.txt"), sep="|")
rm(Mon_Stats)
```

Seasonal Kendall Tau Analysis

Gets seasonal Kendall Tau statistics using the `kendallSeasonalTrendTest` from the `EnvStats` package. The `Trend` parameter is determined from a user-defined function based on the median, Senn slope, and p values from the data. Analysis modified from code created by Jason Scolaro that performed at The Water Atlas: https://sarasota.wateratlas.usf.edu/water-quality-trends/#analysis-overview

The following steps are performed:

1. Define the functions used in the analysis
2. Check to see if there are any groups to run analysis on.
3. Take the `data` variable and only include rows that have a `SufficientData` value of `TRUE`
4. Group data that have the same `ManagedAreaName`.

5. For each group, provides the following information: Earliest Sample Date (EarliestSampleDate), Latest Sample Date (LastSampleDate), Number of Entries (N), Lowest Value (Min), Largest Value (Max), Median, Mean, Standard Deviation, tau, Senn Slope (SennSlope), Senn Intercept (SennIntercept), and p.

    - The analysis is run with the `kendallSeasonalTrendTest` function using the `Year` values for year, and `Month` as the seasonal qualifier, and Trend.
    - An independent.obs value of `TRUE` indicates that the data should be treated as not being serially auto-correlated. An independent.obs value of `FALSE` indicates that it is treated as being serially auto-correlated, but also requires one observation per season per year for the full time of observation.

6. Reformat columns in the data frame from export.
7. Write summary stats to a pipe-delimited .txt file in the output directory

    - Click this text to open Git directory with output files

```
tauSeasonal <- function(data, independent, stats.median, stats.minYear,
                        stats.maxYear, seasondata = MA_M_Stats[MA_M_Stats$ManagedAreaName == MA_Include
  setDT(data)
  tau <- NULL
   tryCatch({ken <- kendallSeasonalTrendTest(
      y = data$ResultValue,
      season = data$Month,
      year = data$relyear,
      independent.obs = independent)

   tau <- ken$estimate[1]
   z <- ken$statistic[2]
   p_z <- ken$p.value[2]
   chi_sq <- ken$statistic[1]
   p_chi_sq <- ken$p.value[1]
   slope <- ken$estimate[2]
   intercept <- ken$estimate[3]
   trend <- trend_calculator(slope, stats.median, p_z)

   seasonresults <- as.data.table(ken$seasonal.estimates)
   rm(ken)
   }, warning = function(w) {
      print(w)
   }, error = function(e) {
      print(e)
   }, finally = {
      if (!exists("tau")) {
         tau <- NA
      }
     if (!exists("z")) {
         z <- NA
      }
      if (!exists("p_z")) {
         p_z <- NA
      }
      if (!exists("chi_sq")) {
         chi_sq <- NA
      }
```

```
    if (!exists("p_chi_sq")) {
      p_chi_sq <- NA
    }
    if (!exists("slope")) {
      slope <- NA
    }
    if (!exists("intercept")) {
      intercept <- NA
    }
    if (!exists("trend")) {
      trend <- NA
    }
})
KT <-data.table(AreaID = unique(data$AreaID),
                ManagedAreaName = unique(data$ManagedAreaName),
                season = "All",
                stats.median = stats.median,
                independent = independent,
                tau = tau,
                z = z,
                p_z = p_z,
                chi_sq = chi_sq,
                p_chi_sq = p_chi_sq,
                slope = slope,
                intercept = intercept,
                trend = trend)

seasonresults[, `:=` (AreaID = unique(data$AreaID),
                      ManagedAreaName = unique(data$ManagedAreaName),
                      season = unique(data$Month),
                      stats.median = as.numeric(NA),
                      independent = independent,
                      z = as.numeric(NA),
                      p_z = as.numeric(NA),
                      chi_sq = as.numeric(NA),
                      p_chi_sq = as.numeric(NA),
                      trend = as.integer(NA))]

for(s in as.integer(unique(seasonresults$season))){
  seasondat_s <- data[Month == s, ]

  if(nrow(seasondat_s) < 3 | length(unique(seasondat_s$Year)) < 3 | is.na(seasonresults[season == s,
    next

  } else{
    if(!is.na(unique(seasondat_s$Month))){
    trend_s <- trend_calculator(seasonresults[season == s, slope], seasondata[Month == s, Median], p_
    ken_s <- kendallTrendTest(ResultValue ~ relyear, data = seasondat_s)
    seasonresults[season == s, `:=` (stats.median = unique(seasondata[Month == s, Median]),
                                     z = ken_s$statistic,
                                     p_z = ken_s$p.value,
                                     chi_sq = NA,
                                     p_chi_sq = NA,
```

```
                                                  trend = trend_s)]
      } else{
        next
      }
     }
    }

    seasonresults[, season := as.character(season)]

    KT <- rbind(KT, seasonresults)
    KT[, season := factor(season, levels = c("All", seq(1:12)), ordered = TRUE)]

    return(KT)
}
runStats <- function(data, MA_M_Stats) {
    data$Index <- as.Date(data$SampleDate) # , "%Y-%m-%d")
    data$ResultValue <- as.numeric(data$ResultValue)
    # Calculate basic stats
    stats.median <- median(data$ResultValue, na.rm = TRUE)
    stats.minYear <- min(data$relyear, na.rm = TRUE)
    stats.maxYear <- max(data$relyear, na.rm = TRUE)
    # Calculate Kendall Tau and Slope stats, then update appropriate columns and table
    seasondata <- MA_M_Stats[MA_M_Stats$ManagedAreaName == MA_Include[i]]
    KT <- tauSeasonal(data, TRUE, stats.median,
                      stats.minYear, stats.maxYear, seasondata)
    # if (is.null(KT[9])) {
    if (is.na(KT[season == "All", trend])) {
       KT <- tauSeasonal(data, FALSE, stats.median,
                         stats.minYear, stats.maxYear, seasondata)
    }
    if (is.null(KT.Stats) == TRUE) {
       KT.Stats <- KT
    } else{
       KT.Stats <- rbind(KT.Stats, KT)
    }
    return(KT.Stats)
}
trend_calculator <- function(slope, median_value, p) {
    trend <-
       if (p < .05 & abs(slope) > abs(median_value) / 10.) {
          if (slope > 0) {
             2
          }
          else {
             -2
          }
       }
    else if (p < .05 & abs(slope) < abs(median_value) / 10.) {
       if (slope > 0) {
          1
       }
       else {
          -1
```

```r
      }
   }
   else
      0
   return(trend)
}
KT.Stats <- NULL
# Loop that goes through each managed area.
# List of managed areas stored in MA_Years$ManagedAreaName
c_names <- c("AreaID", "ManagedAreaName", "Season", "Median", "Independent",
             "tau", "z", "p_z", "chi_sq", "p_chi_sq", "SennSlope", "SennIntercept", "Trend")
if(n==0){
   KT.Stats <- data.frame(matrix(ncol=length(c_names),
                                 nrow=length(MA_Summ$ManagedAreaName)))
   colnames(KT.Stats) <- c_names
   # KT.Stats[, c("AreaID", "ManagedAreaName")] <-
      # MA_Summ[, c("AreaID", "ManagedAreaName")]
} else{
   for (i in 1:n) {
      x <- nrow(data[data$Use_In_Analysis == TRUE &
                        data$ManagedAreaName == MA_Include[i], ])
      if (x>0) {
         KT.Stats <- runStats(data[data$Use_In_Analysis == TRUE &
                                     data$ManagedAreaName ==
                                     MA_Include[i], ], MA_M_Stats)
      }
   }
   KT.Stats <- as.data.frame(KT.Stats)
   # c_names <- c("AreaID", "ManagedAreaName", "Season", "Median", "Independent",
   #              "tau", "z", "p_z", "chi_sq", "p_chi_sq", "SennSlope", "SennIntercept", "Trend")
   if(dim(KT.Stats)[2]==1){
      KT.Stats <- as.data.frame(t(KT.Stats))
   }
   colnames(KT.Stats) <- c_names
   rownames(KT.Stats) <- seq(1:nrow(KT.Stats))
   KT.Stats$tau <- round(as.numeric(KT.Stats$tau), digits=4)
   KT.Stats$z <- round(as.numeric(KT.Stats$z), digits=4)
   KT.Stats$p_z <- round(as.numeric(KT.Stats$p_z), digits=4)
   KT.Stats$chi_sq <- round(as.numeric(KT.Stats$chi_sq), digits=4)
   KT.Stats$p_chi_sq <- round(as.numeric(KT.Stats$p_chi_sq), digits=4)
   KT.Stats$SennSlope <- as.numeric(KT.Stats$SennSlope)
   KT.Stats$SennIntercept <- as.numeric(KT.Stats$SennIntercept)
   KT.Stats$Trend <- as.integer(KT.Stats$Trend)
}

KT.Stats <-  merge.data.frame(MA_Summ, KT.Stats,
                              by=c("AreaID", "ManagedAreaName"), all=TRUE)

KT.Stats <- as.data.table(KT.Stats[order(KT.Stats$ManagedAreaName, KT.Stats$Season), ])
KT.Stats2 <- copy(KT.Stats)
KT.Stats[, `:=` (RelativeDepth = depth, Units = unit)]
KT.Stats_all <- rbind(KT.Stats_all, KT.Stats)
```

```
KT.Stats2$MonitoringID <- NULL
fwrite(KT.Stats2, paste0(out_dir,"/", param_name, "_", activity, "_", depth,
                        "_KendallTau_Stats.txt"), sep="|")
rm(KT.Stats2)
data <- data[!is.na(data$ResultValue),]
```

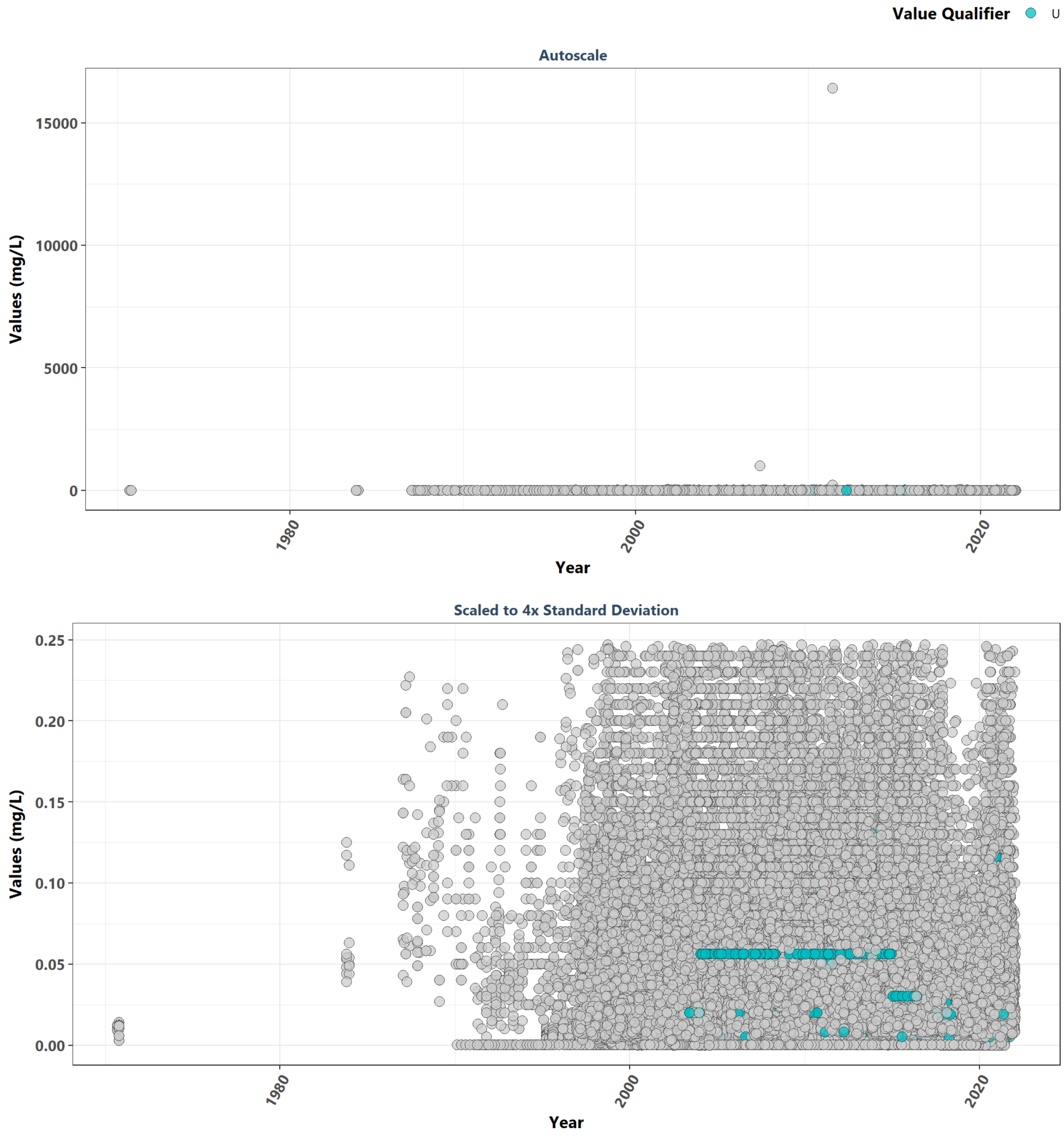
Appendix I: Scatter Plot of Entire Dataset

This part will create a scatter plot of the all data that passed initial filtering criteria with points colored based on specific value qualifiers. The values determined at the beginning (`year_lower`, `year_upper`, `min_RV`, `mn_RV`, `x_scale`, and `y_scale`) are solely for use by the plotting functions and are not output as part of the computed statistics.

```
plot_theme <- theme_bw() +
   theme(text=element_text(family="Segoe UI"),
         title=element_text(face="bold"),
         plot.title=element_text(hjust=0.5, size=14, color="#314963"),
         plot.subtitle=element_text(hjust=0.5, size=10, color="#314963"),
         axis.title.x = element_text(margin = margin(t = 5, r = 0,
                                                     b = 10, l = 0)),
         axis.title.y = element_text(margin = margin(t = 0, r = 10,
                                                     b = 0, l = 0)),
         axis.text=element_text(size=10),
         axis.text.x=element_text(face="bold", angle = 60, hjust = 1),
         axis.text.y=element_text(face="bold"))

year_lower <- min(data$Year)
year_upper <- max(data$Year)
min_RV <- min(data$ResultValue)
mn_RV <- mean(data$ResultValue[data$ResultValue <
                                  quantile(data$ResultValue, 0.98)])
sd_RV <- sd(data$ResultValue[data$ResultValue <
                                quantile(data$ResultValue, 0.98)])
x_scale <- ifelse(year_upper - year_lower > 30, 10, 5)
y_scale <- mn_RV + 4 * sd_RV


p1 <- ggplot(data=data[data$Include==TRUE,],
             aes(x=SampleDate, y=ResultValue, fill=VQ_Plot)) +
   geom_point(shape=21, size=3, color="#333333", alpha=0.75) +
   labs(subtitle="Autoscale",
        x="Year", y=paste0("Values (", unit, ")"),
        fill="Value Qualifier") +
   plot_theme +
   theme(legend.position="top", legend.box="horizontal",
         legend.justification="right") +
   scale_x_date(labels=date_format("%Y")) +
   {if(inc_H==TRUE){
      scale_fill_manual(values=c("H"= "#F8766D", "U"= "#00BFC4",
                                 "HU"="#7CAE00"), na.value="#cccccc")
   } else if(param_name=="Secchi_Depth"){
      scale_fill_manual(values=c("S"= "#F8766D", "U"= "#00BFC4",
```

```
                                  "SU"="#7CAE00"), na.value="#cccccc")
   } else {
      scale_fill_manual(values=c("U"= "#00BFC4"), na.value="#cccccc")
   }}


p2 <- ggplot(data=data[data$Include==TRUE,],
             aes(x=SampleDate, y=ResultValue, fill=VQ_Plot)) +
   geom_point(shape=21, size=3, color="#333333", alpha=0.75) +
   ylim(min_RV, y_scale) +
   labs(subtitle="Scaled to 4x Standard Deviation",
        x="Year", y=paste0("Values (", unit, ")")) +
   plot_theme +
   theme(legend.position="none") +
   scale_x_date(labels=date_format("%Y")) +
   {if(inc_H==TRUE){
      scale_fill_manual(values=c("H"= "#F8766D", "U"= "#00BFC4",
                                  "HU"="#7CAE00"), na.value="#cccccc")
   } else if(param_name=="Secchi_Depth"){
      scale_fill_manual(values=c("S"= "#F8766D", "U"= "#00BFC4",
                                  "SU"="#7CAE00"), na.value="#cccccc")
   } else {
      scale_fill_manual(values=c("U"= "#00BFC4"), na.value="#cccccc")
   }}

leg <- get_legend(p1)
pset <- ggarrange(leg, p1 + theme(legend.position="none"), p2,
                  ncol=1, heights=c(0.1, 1, 1))

p0 <- ggplot() + labs(title="Scatter Plot for Entire Dataset") +
   plot_theme + theme(panel.border=element_blank(),
                      panel.grid.major=element_blank(),
                      panel.grid.minor=element_blank(),
                      axis.line=element_blank())

ggarrange(p0, pset, ncol=1, heights=c(0.1, 1))
```

Scatter Plot for Entire Dataset



Appendix II: Dataset Summary Box Plots

Box plots are created by using the entire data set and excludes any data that has been previously filtered out. The scripts that create plots follow this format

1. Use the data set that only has `SufficientData` of `TRUE`
2. Set what values are to be used for the x-axis, y-axis, and the variable that should determine groups for the box plots
3. Set the plot type as a box plot with the size of the outlier points
4. Create the title, x-axis, y-axis, and color fill labels
5. Set the y and x limits
6. Make the axis labels bold
7. Plot the arrangement as a set of panels

This set of box plots are grouped by year.

```
min_RV <- min(data$ResultValue[data$Include==TRUE])
mn_RV <- mean(data$ResultValue[data$Include==TRUE &
                                  data$ResultValue <
                                  quantile(data$ResultValue, 0.98)])
sd_RV <- sd(data$ResultValue[data$Include==TRUE &
                                data$ResultValue <
                                quantile(data$ResultValue, 0.98)])
y_scale <- mn_RV + 4 * sd_RV

p1 <- ggplot(data=data[data$Include==TRUE, ],
             aes(x=Year, y=ResultValue, group=Year)) +
   geom_boxplot(color="#333333", fill="#cccccc", outlier.shape=21,
                outlier.size=3, outlier.color="#333333",
                outlier.fill="#cccccc", outlier.alpha=0.75) +
   labs(subtitle="Autoscale", x="Year",
        y=paste0("Values (", unit, ")")) +
   plot_theme

p2 <- ggplot(data=data[data$Include==TRUE, ],
             aes(x=Year, y=ResultValue, group=Year)) +
   geom_boxplot(color="#333333", fill="#cccccc", outlier.shape=21,
                outlier.size=3, outlier.color="#333333",
                outlier.fill="#cccccc", outlier.alpha=0.75) +
   labs(subtitle="Scaled to 4x Standard Deviation", x="Year",
        y=paste0("Values (", unit, ")")) +
   ylim(min_RV, y_scale) +
   plot_theme

p3 <- ggplot(data=data[data$Include==TRUE, ],
             aes(x=as.integer(Year), y=ResultValue, group=Year)) +
   geom_boxplot(color="#333333", fill="#cccccc", outlier.shape=21,
                outlier.size=3, outlier.color="#333333",
                outlier.fill="#cccccc", outlier.alpha=0.75) +
   labs(subtitle="Scaled to 4x Standard Deviation, Last 10 Years",
        x="Year", y=paste0("Values (", unit, ")")) +
   ylim(min_RV, y_scale) +
   scale_x_continuous(limits=c(max(data$Year) - 10.5, max(data$Year)+1),
                      breaks=seq(max(data$Year) - 10, max(data$Year), 2)) +
   plot_theme

set <- ggarrange(p1, p2, p3, ncol=1)

p0 <- ggplot() + labs(title="Summary Box Plots for Entire Data",
```

```
                       subtitle="By Year") + plot_theme +
   theme(panel.border=element_blank(), panel.grid.major=element_blank(),
         panel.grid.minor=element_blank(), axis.line=element_blank())

Yset <- ggarrange(p0, set, ncol=1, heights=c(0.07, 1))
```

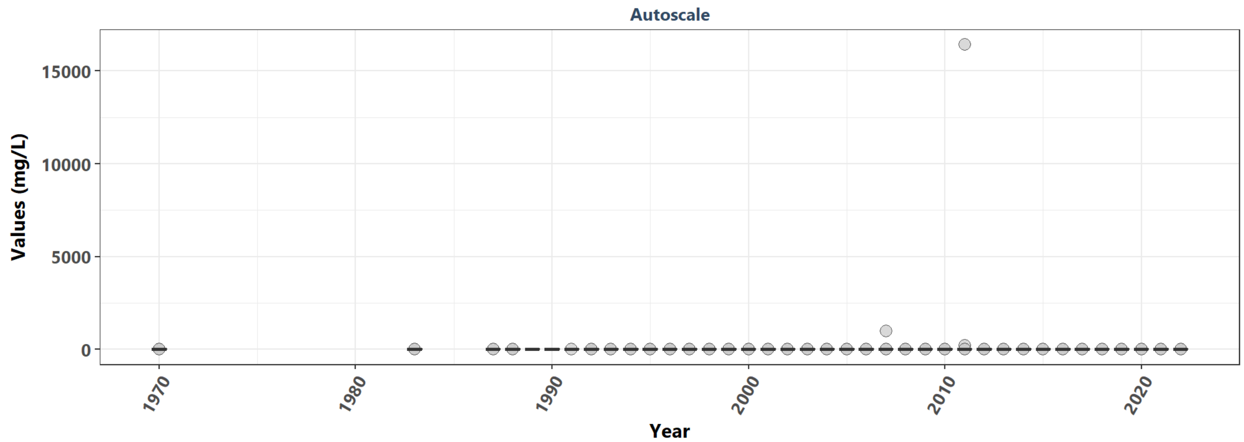
This set of box plots are grouped by year and month with the color being related to the month.

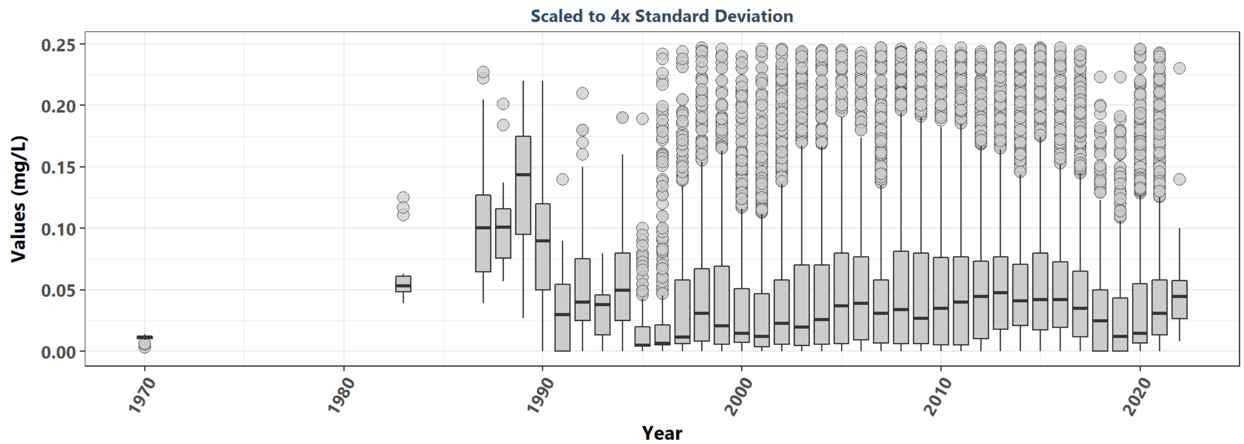
```
p1 <- ggplot(data=data[data$Include==TRUE, ],
             aes(x=YearMonthDec, y=ResultValue,
                 group=YearMonth, color=as.factor(Month))) +
   geom_boxplot(fill="#cccccc", outlier.size=1.5, outlier.alpha=0.75) +
   labs(subtitle="Autoscale", x="Year",
        y=paste0("Values (", unit, ")"), color="Month") +
   plot_theme +
   theme(legend.position="top", legend.box="horizontal") +
   guides(color=guide_legend(nrow=1))

p2 <- ggplot(data=data[data$Include==TRUE, ],
             aes(x=YearMonthDec, y=ResultValue,
                 group=YearMonth, color=as.factor(Month))) +
   geom_boxplot(fill="#cccccc", outlier.size=1.5, outlier.alpha=0.75) +
   labs(subtitle="Scaled to 4x Standard Deviation",
        x="Year", y=paste0("Values (", unit, ")")) +
   ylim(min_RV, y_scale) +
   plot_theme +
   theme(legend.position="none", axis.text.x=element_text(face="bold"),
         axis.text.y=element_text(face="bold"))

p3 <- ggplot(data=data[data$Include==TRUE, ],
             aes(x=YearMonthDec, y=ResultValue,
                 group=YearMonth, color=as.factor(Month))) +
   geom_boxplot(fill="#cccccc", outlier.size=1.5, outlier.alpha=0.75) +
   labs(subtitle="Scaled to 4x Standard Deviation, Last 10 Years",
        x="Year", y=paste0("Values (", unit, ")")) +
   ylim(min_RV, y_scale) +
   scale_x_continuous(limits=c(max(data$Year) - 10.5, max(data$Year)+1),
                      breaks=seq(max(data$Year) - 10, max(data$Year), 2)) +
   plot_theme +
   theme(legend.position="none")
leg <- get_legend(p1)
set <- ggarrange(leg, p1 + theme(legend.position="none"), p2, p3, ncol=1,
                 heights=c(0.1, 1, 1, 1))

p0 <- ggplot() + labs(title="Summary Box Plots for Entire Data",
                      subtitle="By Year & Month") + plot_theme +
   theme(panel.border=element_blank(), panel.grid.major=element_blank(),
         panel.grid.minor=element_blank(), axis.line=element_blank())

YMset <- ggarrange(p0, set, ncol=1, heights=c(0.07, 1))
```

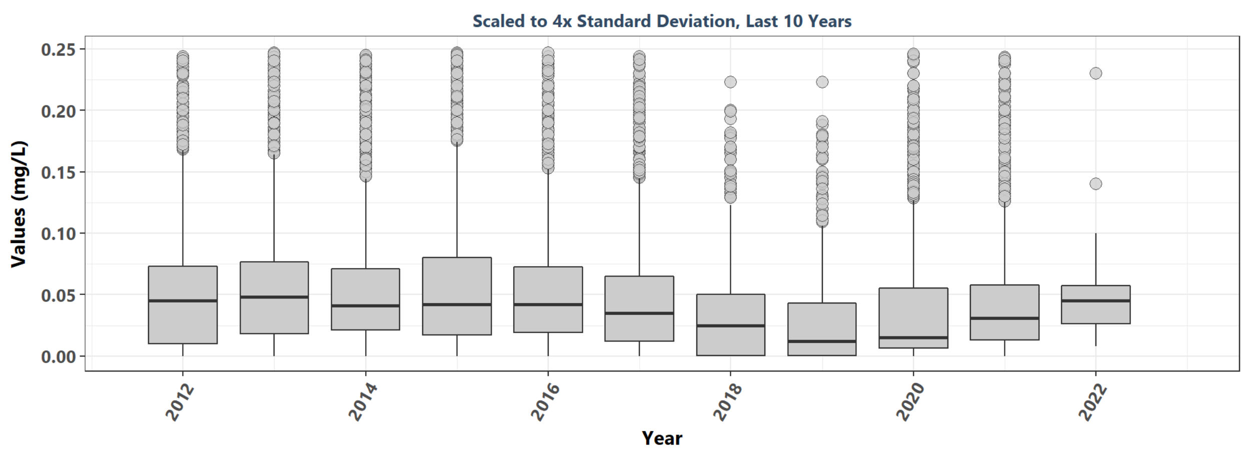
The following box plots are grouped by month with fill color being related to the month. This is designed to view potential seasonal trends.

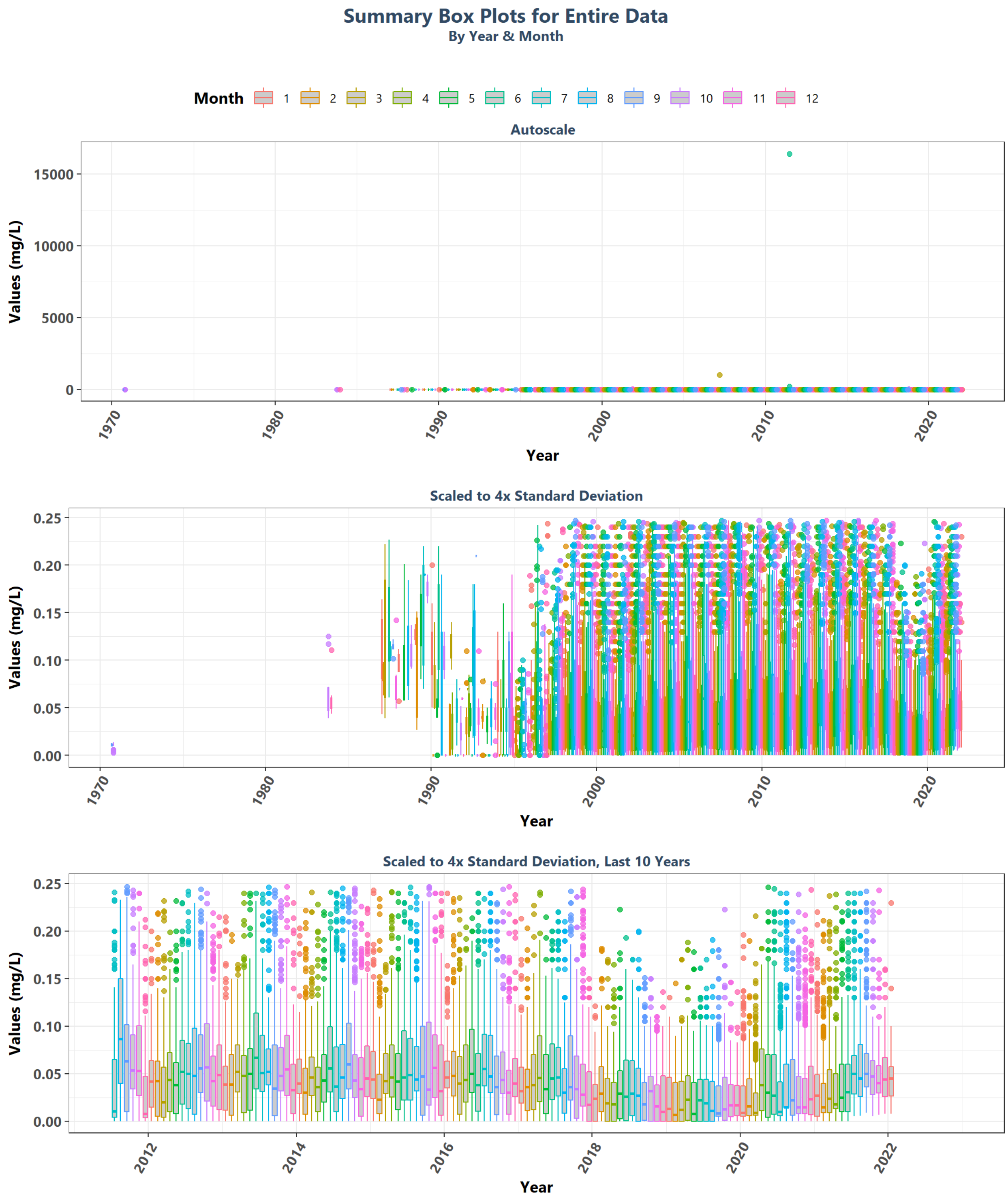
```r
p1 <- ggplot(data=data[data$Include==TRUE, ],
             aes(x=Month, y=ResultValue,
                 group=Month, fill=as.factor(Month))) +
   geom_boxplot(color="#333333", outlier.shape=21, outlier.size=3,
                outlier.color="#333333", outlier.alpha=0.75) +
   labs(subtitle="Autoscale", x="Month",
        y=paste0("Values (", unit, ")"), fill="Month") +
   scale_x_continuous(limits=c(0, 13), breaks=seq(3, 12, 3)) +
   plot_theme +
   theme(legend.position="top", legend.box="horizontal") +
   guides(fill=guide_legend(nrow=1))

p2 <- ggplot(data=data[data$Include==TRUE, ],
             aes(x=Month, y=ResultValue,
                 group=Month, fill=as.factor(Month))) +
   geom_boxplot(color="#333333", outlier.shape=21, outlier.size=3,
                outlier.color="#333333", outlier.alpha=0.75) +
   labs(subtitle="Scaled to 4x Standard Deviation",
        x="Month", y=paste0("Values (", unit, ")")) +
   ylim(min_RV, y_scale) +
   scale_x_continuous(limits=c(0, 13), breaks=seq(3, 12, 3)) +
   plot_theme +
   theme(legend.position="none")

p3 <- ggplot(data=data[data$Include==TRUE &
                          data$Year >= max(data$Year) - 10, ],
             aes(x=Month, y=ResultValue,
                 group=Month, fill=as.factor(Month))) +
   geom_boxplot(color="#333333", outlier.shape=21, outlier.size=3,
                outlier.color="#333333", outlier.alpha=0.75) +
   labs(subtitle="Scaled to 4x Standard Deviation, Last 10 Years",
        x="Month", y=paste0("Values (", unit, ")")) +
   ylim(min_RV, y_scale) +
   scale_x_continuous(limits=c(0, 13), breaks=seq(3, 12, 3)) +
   plot_theme +
   theme(legend.position="none")
leg <- get_legend(p1)
set <- ggarrange(leg, p1 + theme(legend.position="none"), p2, p3, ncol=1,
                 heights=c(0.1, 1, 1, 1))

p0 <- ggplot() + labs(title="Summary Box Plots for Entire Data",
                      subtitle="By Month") + plot_theme +
   theme(panel.border=element_blank(), panel.grid.major=element_blank(),
         panel.grid.minor=element_blank(), axis.line=element_blank())

Mset <- ggarrange(p0, set, ncol=1, heights=c(0.07, 1))
```

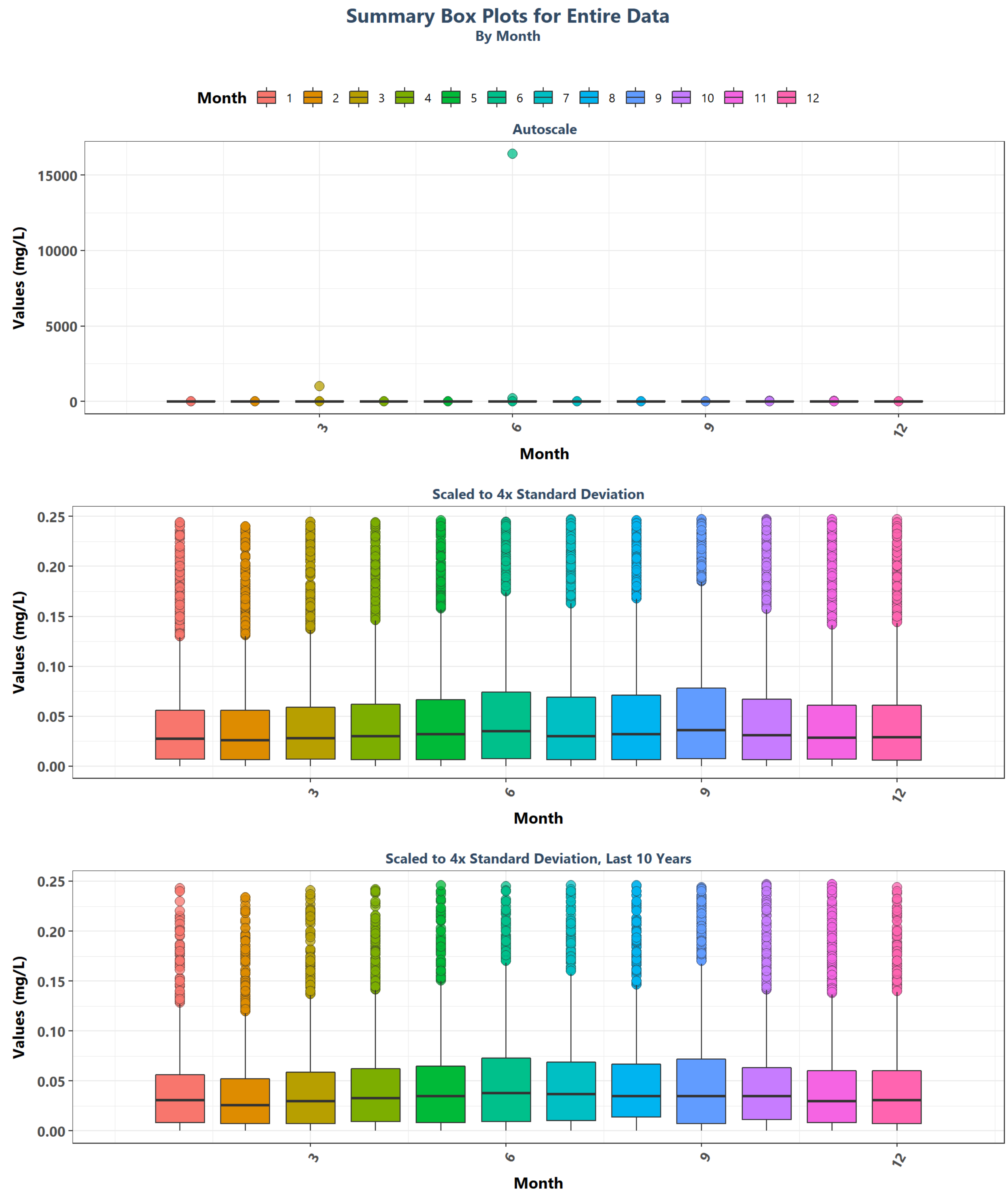
Summary Box Plots for Entire Data
By Year







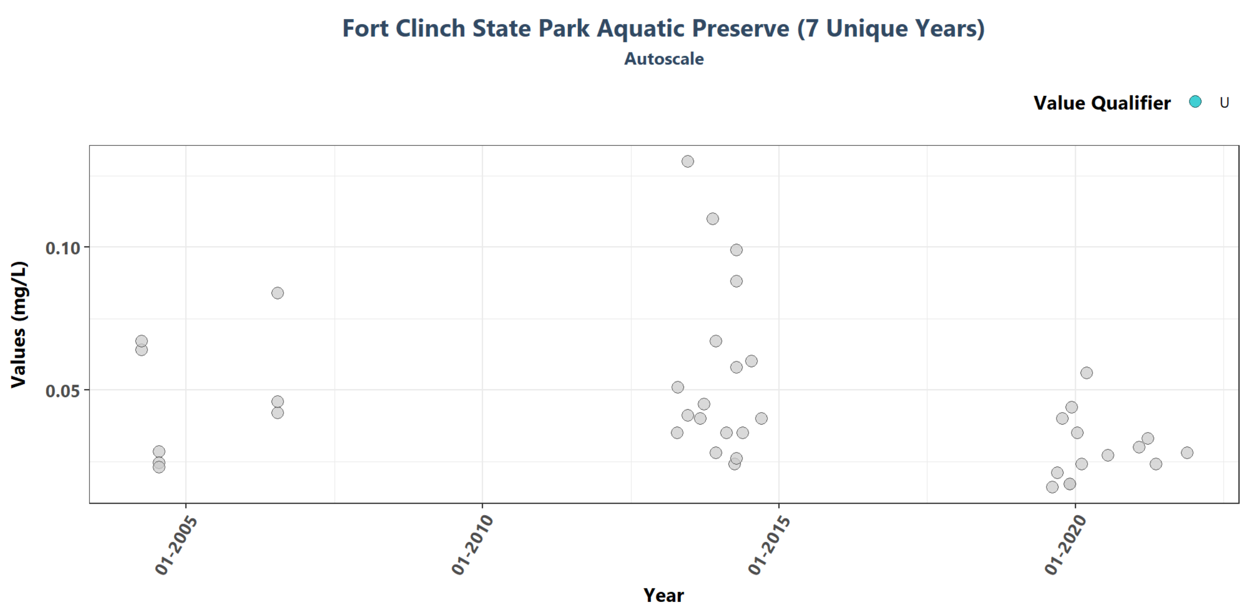

Summary Box Plots for Entire Data
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
15000
10000
5000
0
1970
1980
1990
2000
2010
2020
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.25
0.20
0.15
0.10
0.05
0.00
1970
1980
1990
2000
2010
2020
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.25
0.20
0.15
0.10
0.05
0.00
2012
2014
2016
2018
2020
2022
Year

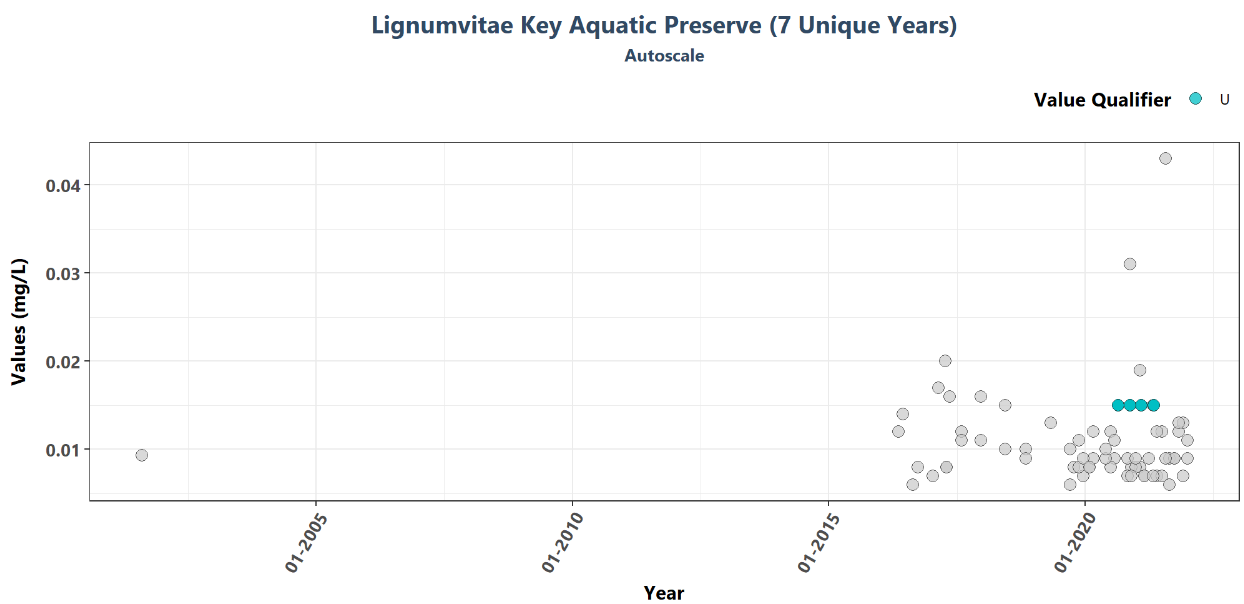


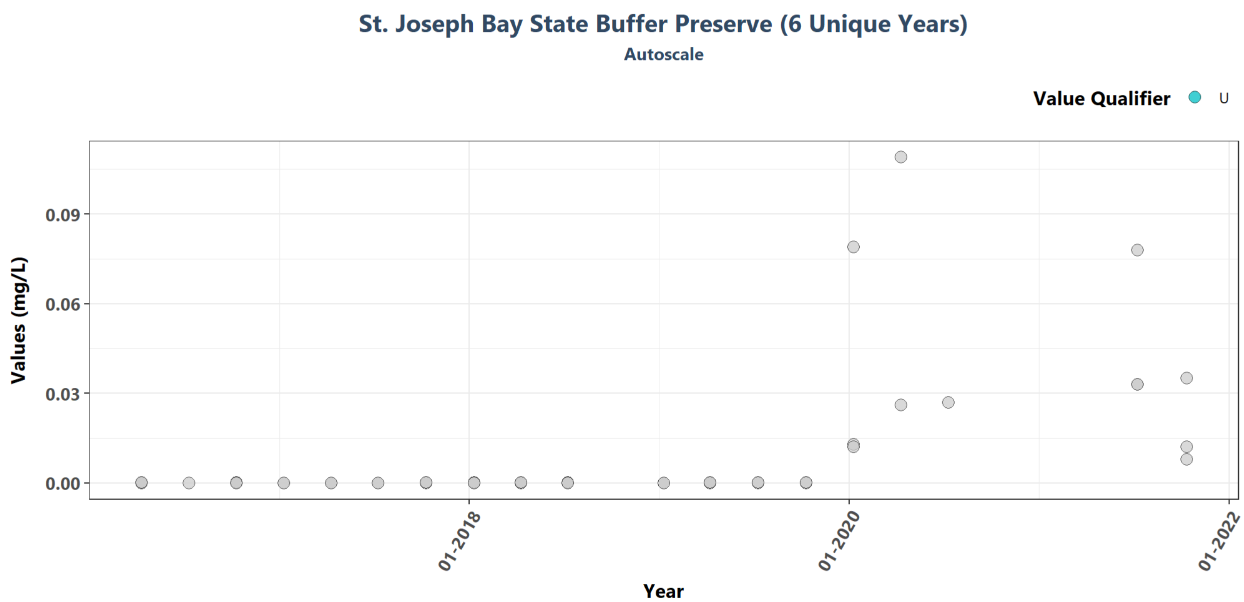
Appendix III: Excluded Managed Areas

Scatter plots of data values are created for managed areas that have fewer than 10 separate years of data entries. Data points are colored based on specific value qualifiers of interest.

```
if(z==0){
  print("There are no managed areas that qualify.")
} else {
  for(i in 1:z){
    p1<-ggplot(data=data[data$ManagedAreaName==MA_Exclude$ManagedAreaName[i]&
                          data$Include==TRUE, ],
               aes(x=SampleDate, y=ResultValue, fill=VQ_Plot)) +
      geom_point(shape=21, size=3, color="#333333", alpha=0.75) +
      labs(title=paste0(MA_Exclude$ManagedAreaName[i], " (",
                        MA_Exclude$N_Years[i], " Unique Years)"),
           subtitle="Autoscale", x="Year",
           y=paste0("Values (", unit, ")"), fill="Value Qualifier") +
      plot_theme +
      theme(legend.position="top", legend.box="horizontal",
            legend.justification="right") +
      scale_x_date(labels=date_format("%m-%Y")) +
      {if(inc_H==TRUE){
        scale_fill_manual(values=c("H"= "#F8766D", "U"= "#00BFC4",
                                   "HU"="#7CAE00"), na.value="#cccccc")
      } else if(param_name=="Secchi_Depth"){
        scale_fill_manual(values=c("S"= "#F8766D", "U"= "#00BFC4",
                                   "SU"="#7CAE00"), na.value="#cccccc")
      } else {
        scale_fill_manual(values=c("U"= "#00BFC4"), na.value="#cccccc")
      }}
    print(p1)
  }
}
```







Appendix IV: Managed Area Trendlines

The plots created in this section are designed to show the general trend of the data. Data is taken and grouped by `ManagedAreaName`. The trendlines on the plots are created using the Senn slope and intercept from the seasonal Kendall Tau analysis. The scripts that create plots follow this format

1. Use the data set that only has `SufficientData` of `TRUE` for the desired managed area
2. Determine the earliest and latest year of the data to create x-axis scale and intervals
3. Determine the minimum, mean, and standard deviation for the data to be used for y-axis scales
   - Excludes the top 2% of values to reduce the impact of extreme outliers on the y-axis scale
4. Set what values are to be used for the x-axis, y-axis, and the variable that should determine groups for the plots

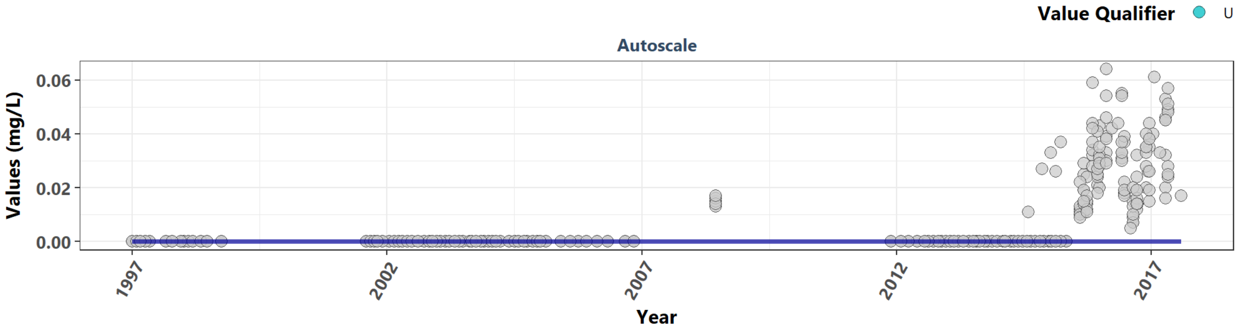
5. Set the plot type as a point plot with the size of the points
6. Add the linear trend
7. Create the title, x-axis, y-axis, and color fill labels
8. Set the y and x limits
9. Make the axis labels bold
10. Plot the arrangement as a set of panels

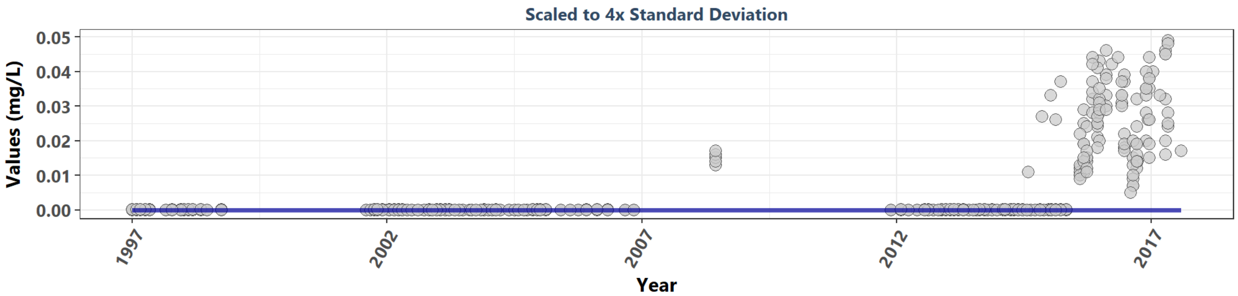
```
if(n==0){
   print("There are no managed areas that qualify.")
} else {
   for (i in 1:n) {
      plot_data <- data[data$SufficientData==TRUE &
                           data$ManagedAreaName==MA_Include[i],]
      plot_data$Season <- factor(plot_data$Month, levels = c("All", seq(1, 12)), ordered = TRUE)
      year_lower <- min(plot_data$relyear)
      year_upper <- max(plot_data$relyear)
      min_RV <- min(plot_data$ResultValue)
      mn_RV <- mean(plot_data$ResultValue[plot_data$ResultValue <
                                             quantile(data$ResultValue, 0.98)])
      sd_RV <- sd(plot_data$ResultValue[plot_data$ResultValue <
                                           quantile(data$ResultValue, 0.98)])
      x_scale <- ifelse(year_upper - year_lower > 30, 10, 5)
      y_scale <- mn_RV + 4 * sd_RV

      tau <- KT.Stats$tau[KT.Stats$ManagedAreaName==MA_Include[i]]
      s_slope <- KT.Stats$SennSlope[KT.Stats$ManagedAreaName==MA_Include[i]]
      s_int <- KT.Stats$SennIntercept[KT.Stats$ManagedAreaName==MA_Include[i]]
      trend <- KT.Stats$Trend[KT.Stats$ManagedAreaName==MA_Include[i]]
      z <- KT.Stats$z[KT.Stats$ManagedAreaName==MA_Include[i]]
      p_z <- KT.Stats$p_z[KT.Stats$ManagedAreaName==MA_Include[i]]
      chi_sq <- KT.Stats$chi_sq[KT.Stats$ManagedAreaName==MA_Include[i]]
      p_chi_sq <- KT.Stats$p_chi_sq[KT.Stats$ManagedAreaName==MA_Include[i]]

      # model <- lm(ResultValue ~ relyear_dd,
      #             data=plot_data)
      # m_int <- coef(model)[[1]]
      # m_slope <- coef(model)[[2]]
      # rm(model)

      xbrks <- seq(round_any(min(plot_data$relyear_dd), 5, floor), round_any(max(plot_data$relyear_dd),
                   by = (round_any(max(plot_data$relyear_dd), 5, ceiling) - round_any(min(plot_data$rel

      xlabs <- seq(max(plot_data$Year) - round_any(max(plot_data$relyear_dd), 5, ceiling),
                   max(plot_data$Year),
                   by = (max(plot_data$Year) - (max(plot_data$Year) - round_any(max(plot_data$relyear_d

      KT.Stats[, season := Season]
      KT.Stats[ManagedAreaName==MA_Include[i] & season != "All", `:=` (N_Data = nrow(plot_data[Season =
      KT.Stats[ManagedAreaName==MA_Include[i] & season == "All", `:=` (relyear_dd_lower = min(plot_data
      KT.Stats[, season := NULL]

      # plot_data[is.na(VQ_Plot), VQ_Plot := "None"]
      p1 <- ggplot(data=plot_data,
                   aes(x=relyear_dd, y=ResultValue, fill = VQ_Plot)) +
```

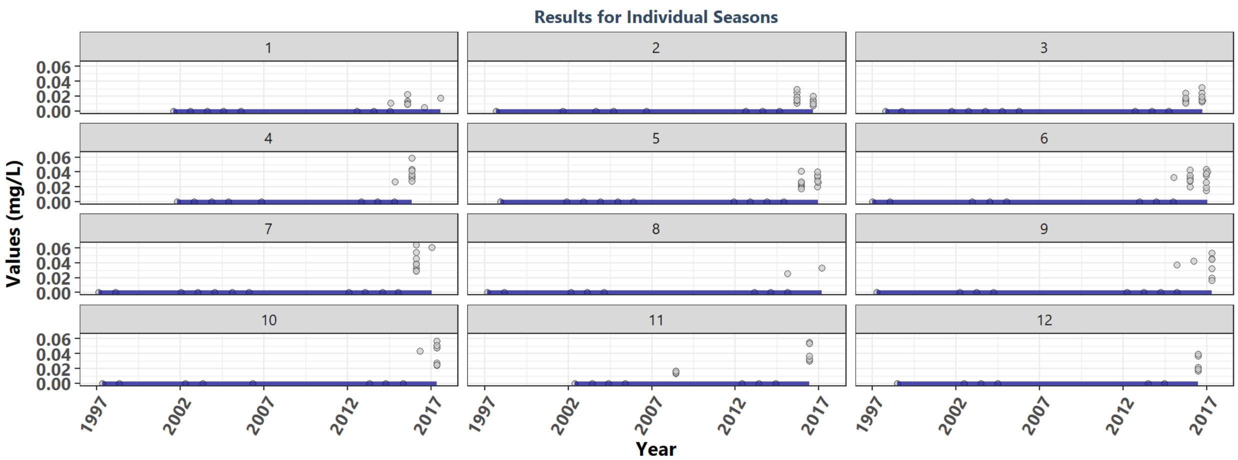
```
    geom_point(shape=21, size=3, color="#333333", alpha=0.75) +
    # geom_abline(aes(slope=s_slope, intercept=s_int),
    #             color="#000099", size=1.2, alpha=0.7) +
    geom_segment(data = KT.Stats[ManagedAreaName==MA_Include[i] & Season == "All", ], aes(x = relye
                                                                                         y = relyear_
                                                                                         xend = relyea
                                                                                         yend = relyea
                 color="#000099", size=1.2, alpha=0.7, inherit.aes = FALSE) +
    labs(subtitle="Autoscale",
         x="Year", y=paste0("Values (", unit, ")"),
         fill="Value Qualifier") +
    plot_theme +
    theme(legend.position="top", legend.box="horizontal",
          legend.justification="right") +
    {if(inc_H==TRUE){
       scale_fill_manual(values=c("H"= "#F8766D", "U"= "#00BFC4",
                                  "HU"="#7CAE00"), na.value="#cccccc")
    } else if(param_name=="Secchi_Depth"){
       scale_fill_manual(values=c("S"= "#F8766D", "U"= "#00BFC4",
                                  "SU"="#7CAE00"), na.value="#cccccc")
    } else {
       scale_fill_manual(values=c("U"= "#00BFC4"), na.value="#cccccc")
    }} +
    scale_x_continuous(breaks = xbrks,
                       labels = xlabs)

 p2 <- ggplot(data=plot_data,
              aes(x=relyear_dd, y=ResultValue, fill=VQ_Plot)) +
    geom_point(shape=21, size=3, color="#333333", alpha=0.75) +
    # geom_abline(aes(slope=s_slope, intercept=s_int),
    #             color="#000099", size=1.2, alpha=0.7) +
    geom_segment(data = KT.Stats[ManagedAreaName==MA_Include[i] & Season == "All", ], aes(x = relye
                                                                                         y = relyear_
                                                                                         xend = relyea
                                                                                         yend = relyea
                 color="#000099", size=1.2, alpha=0.7, inherit.aes = FALSE) +
    ylim(min_RV, y_scale) +
    labs(subtitle="Scaled to 4x Standard Deviation",
         x="Year", y=paste0("Values (", unit, ")")) +
    plot_theme +
    theme(legend.position="none") +
    {if(inc_H==TRUE){
       scale_fill_manual(values=c("H"= "#F8766D", "U"= "#00BFC4",
                                  "HU"="#7CAE00"), na.value="#cccccc")
    } else if(param_name=="Secchi_Depth"){
       scale_fill_manual(values=c("S"= "#F8766D", "U"= "#00BFC4",
                                  "SU"="#7CAE00"), na.value="#cccccc")
    } else {
       scale_fill_manual(values=c("U"= "#00BFC4"), na.value="#cccccc")
    }} +
    scale_x_continuous(breaks = xbrks,
                       labels = xlabs)
```

```
      splot <- ggplot(plot_data, aes(x = relyear_dd, y = ResultValue)) +
         geom_point(shape = 21, size = 1.5, color="#333333", fill="#cccccc", alpha=0.75) +
         geom_segment(data = KT.Stats[ManagedAreaName==MA_Include[i] & Season != "All", ], aes(x = rely
                                                                                                 y = relyear_
                                                                                                 xend = relyea
                                                                                                 yend = relyea
                      color="#000099", size=1.2, alpha=0.7) +
         #ylim(min_RV-0.1*y_scale, y_scale) +
         scale_x_continuous(breaks = xbrks,
                            labels = xlabs) +
         labs(y = paste0("Values (", unit, ")"), x = "Year", subtitle = "Results for Individual Seasons
         facet_wrap(~Season, ncol = 3) +
         plot_theme

      leg <- get_legend(p1)
      KTset <- ggarrange(leg, p1 + theme(legend.position="none"), p2,
                         splot, ncol=1, heights=c(0.1, 1, 1, 1.5))

      p0 <- ggplot() + labs(title=paste0(MA_Include[i])) +
         plot_theme + theme(panel.border=element_blank(),
                            panel.grid.major=element_blank(),
                            panel.grid.minor=element_blank(),
                            axis.line=element_blank())

      KT.Stats[ManagedAreaName==MA_Include[i], `:=` (N = N_Data,
                                                      Median = round(Median, 2),
                                                      Slope = round(SennSlope, 4),
                                                      Int. = round(SennIntercept, 4),
                                                      z = round(z, 1),
                                                      chi_sq = round(chi_sq, 1))]

      print(ggarrange(p0, KTset, ncol=1, heights=c(0.1, 1.25)))
      cat('\n')
      print(KT.Stats[KT.Stats$ManagedAreaName==MA_Include[i], ] %>%
         select(Season, N, Median, tau, Slope, Int., z, p_z, chi_sq, p_chi_sq, Trend) %>%
         kable(format="latex") %>%
            row_spec(0,bold=TRUE) %>%
            kable_styling(latex_options = "HOLD_position",
                          font_size = 7) %>%
            add_footnote(
            "p < 0.00005 appear as 0 due to rounding"))
      cat('\n')
      rm(plot_data)
      rm(KTset, leg)
   }
}
```

Alligator Harbor Aquatic Preserve



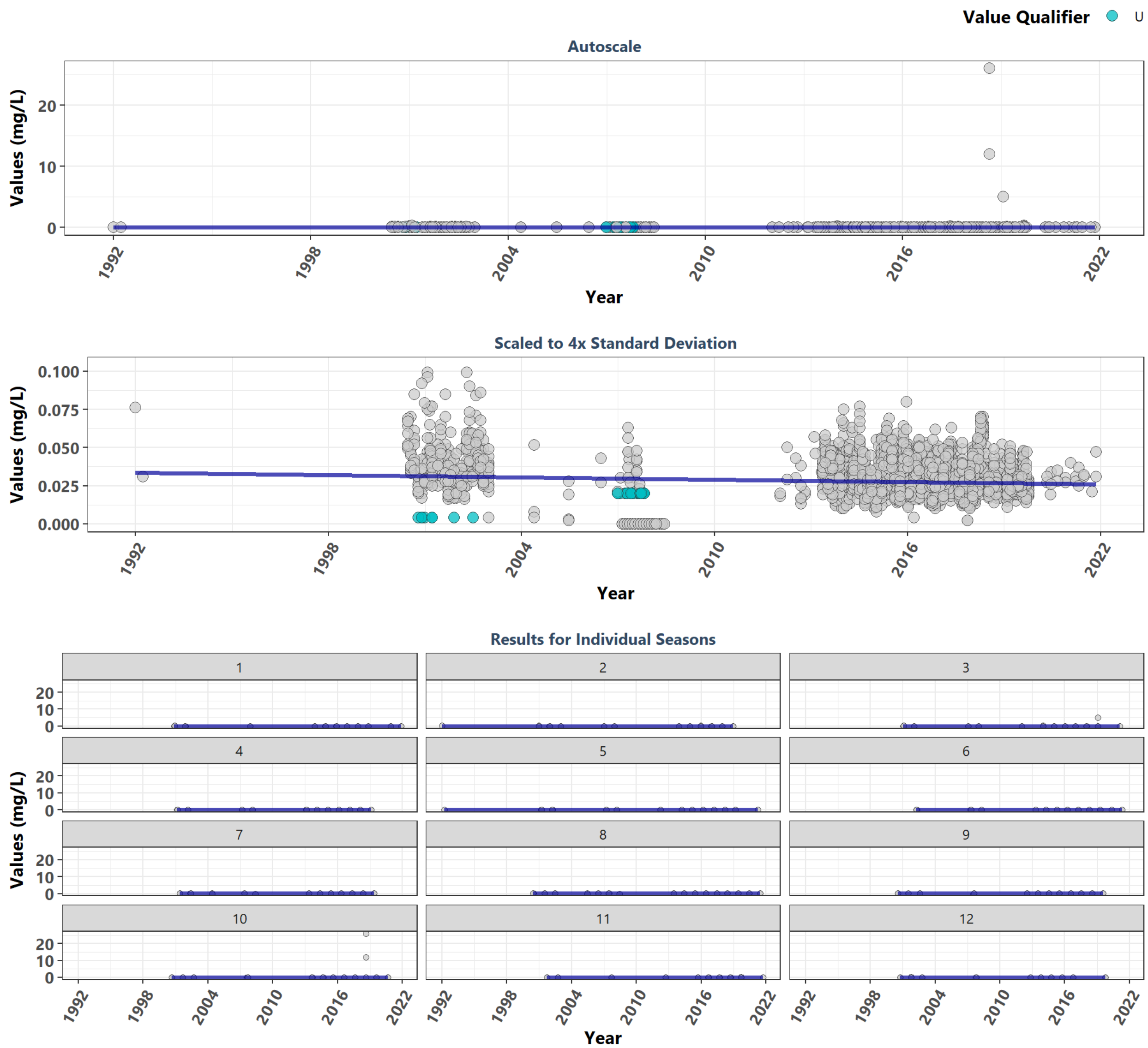




Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	807	0	0.2947	0	0	12.3	0.0000	28	0.0033	1
1	65	0	0.3547	0	0	5.3	0.0000	NA	NA	2
2	69	0	0.2807	0	0	5.2	0.0000	NA	NA	2
3	82	0	0.2278	0	0	3.6	0.0004	NA	NA	1
4	63	0	-0.0636	0	0	3.8	0.0001	NA	NA	-1
5	83	0	0.2922	0	0	4.8	0.0000	NA	NA	1
6	70	0	0.2522	0	0	3.5	0.0006	NA	NA	1
7	84	0	0.4077	0	0	3.1	0.0021	NA	NA	1
8	56	0	0.2722	0	0	-0.7	0.4891	NA	NA	1
9	64	0	0.4481	0	0	3.4	0.0006	NA	NA	1
10	62	0	0.4241	0	0	2.9	0.0036	NA	NA	1
11	61	0	0.2665	0	0	4.7	0.0000	NA	NA	1
12	48	0	0.3272	0	0	2.8	0.0058	NA	NA	2

[a] $p < 0.00005$ appear as 0 due to rounding

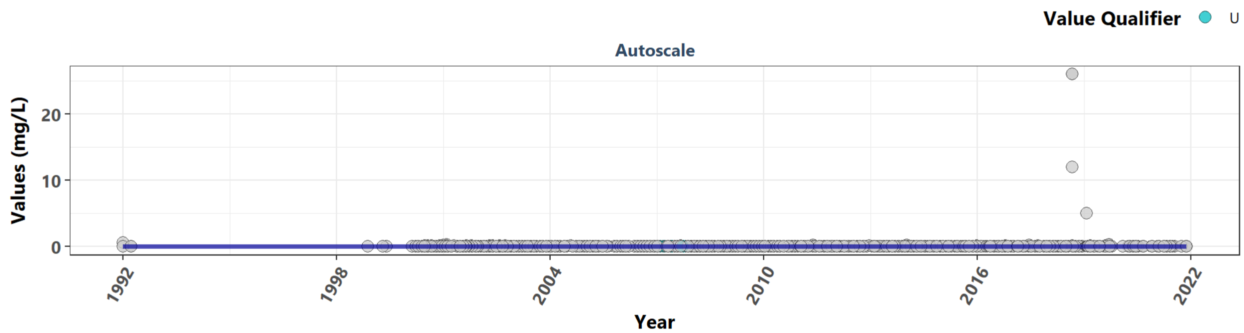
Apalachicola Bay Aquatic Preserve

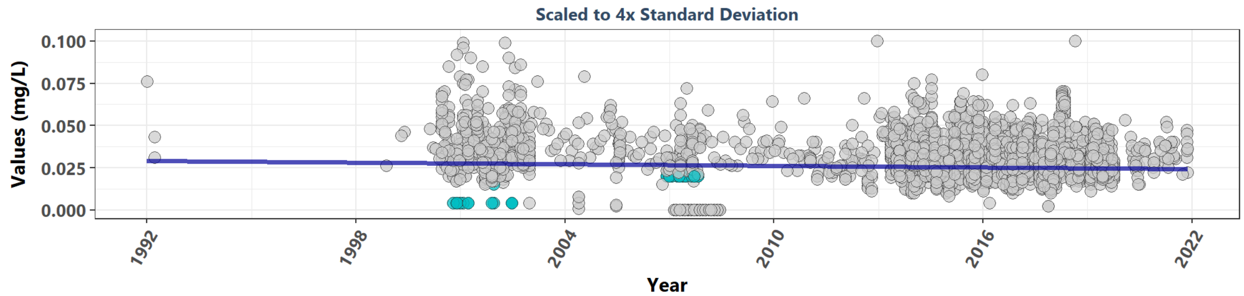


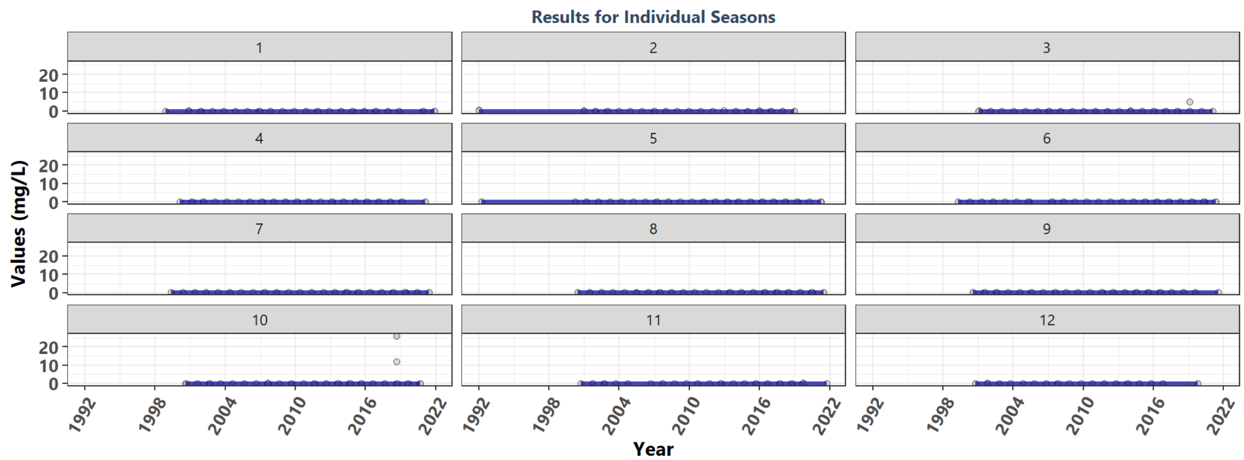
Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	1898	0.03	-0.0568	-0.0002	0.0333	-4.0	0.0001	68.4	0	-1
1	143	0.03	-0.1752	-0.0007	0.0509	0.1	0.8962	NA	NA	-1
2	156	0.03	0.0257	0.0001	0.0322	1.4	0.1596	NA	NA	1
3	136	0.03	0.1764	0.0009	0.0128	-1.4	0.1746	NA	NA	1
4	143	0.03	0.0168	0.0001	0.0345	-2.8	0.0051	NA	NA	1
5	157	0.03	0.0074	0.0000	0.0270	3.3	0.0009	NA	NA	-1
6	145	0.04	-0.3132	-0.0012	0.0606	0.5	0.6453	NA	NA	-1
7	173	0.04	0.0753	0.0003	0.0212	0.3	0.7422	NA	NA	1
8	182	0.03	-0.1568	-0.0006	0.0437	-6.3	0.0000	NA	NA	-1
9	166	0.03	-0.0882	-0.0002	0.0287	-3.4	0.0007	NA	NA	-1
10	190	0.04	-0.0993	-0.0004	0.0442	-2.1	0.0398	NA	NA	-1
11	167	0.03	-0.0779	-0.0003	0.0377	-0.5	0.5980	NA	NA	-1
12	140	0.03	-0.0272	-0.0001	0.0278	-1.6	0.1192	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

Apalachicola National Estuarine Research Reserve



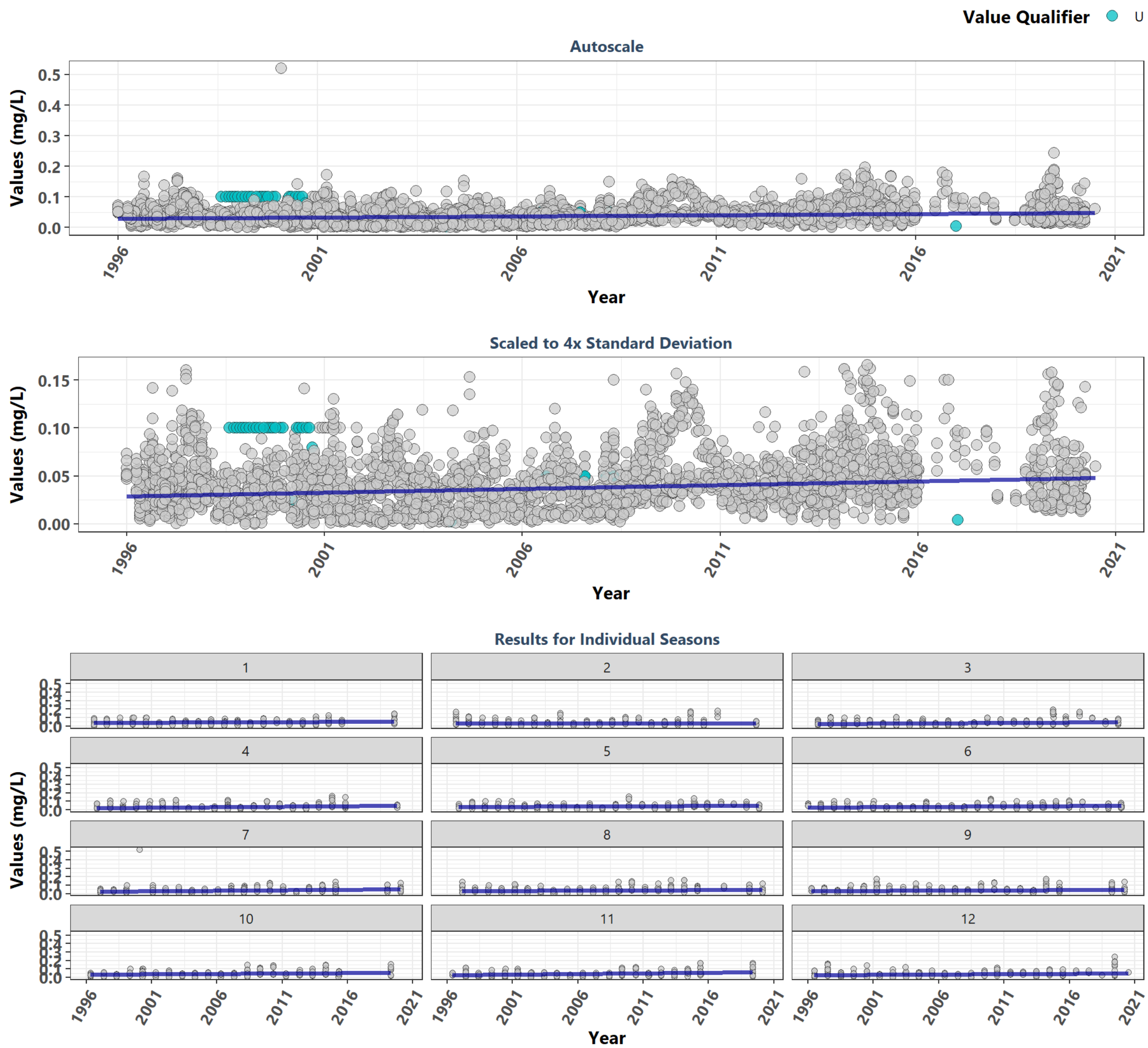




Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	2660	0.03	-0.0472	-0.0002	0.0290	-3.7	0.0002	68.4	0	-1
1	208	0.03	-0.1823	-0.0005	0.0443	-0.1	0.9553	NA	NA	-1
2	216	0.03	-0.2685	-0.0010	0.0550	1.4	0.1485	NA	NA	-1
3	182	0.03	0.0692	0.0003	0.0274	-1.4	0.1714	NA	NA	1
4	197	0.03	-0.1360	-0.0005	0.0425	-2.9	0.0043	NA	NA	-1
5	221	0.04	-0.0026	0.0000	0.0285	1.6	0.0994	NA	NA	-1
6	221	0.03	-0.0218	0.0000	0.0260	1.5	0.1234	NA	NA	-1
7	241	0.03	-0.0678	-0.0003	0.0358	0.9	0.3562	NA	NA	-1
8	252	0.03	0.0657	0.0002	0.0227	-6.4	0.0000	NA	NA	1
9	236	0.03	0.0741	0.0003	0.0286	-4.2	0.0000	NA	NA	1
10	277	0.03	0.0398	0.0002	0.0292	-0.6	0.5741	NA	NA	1
11	222	0.03	-0.0225	-0.0001	0.0342	-0.5	0.6254	NA	NA	-1
12	187	0.03	-0.1048	-0.0002	0.0288	-2.1	0.0322	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

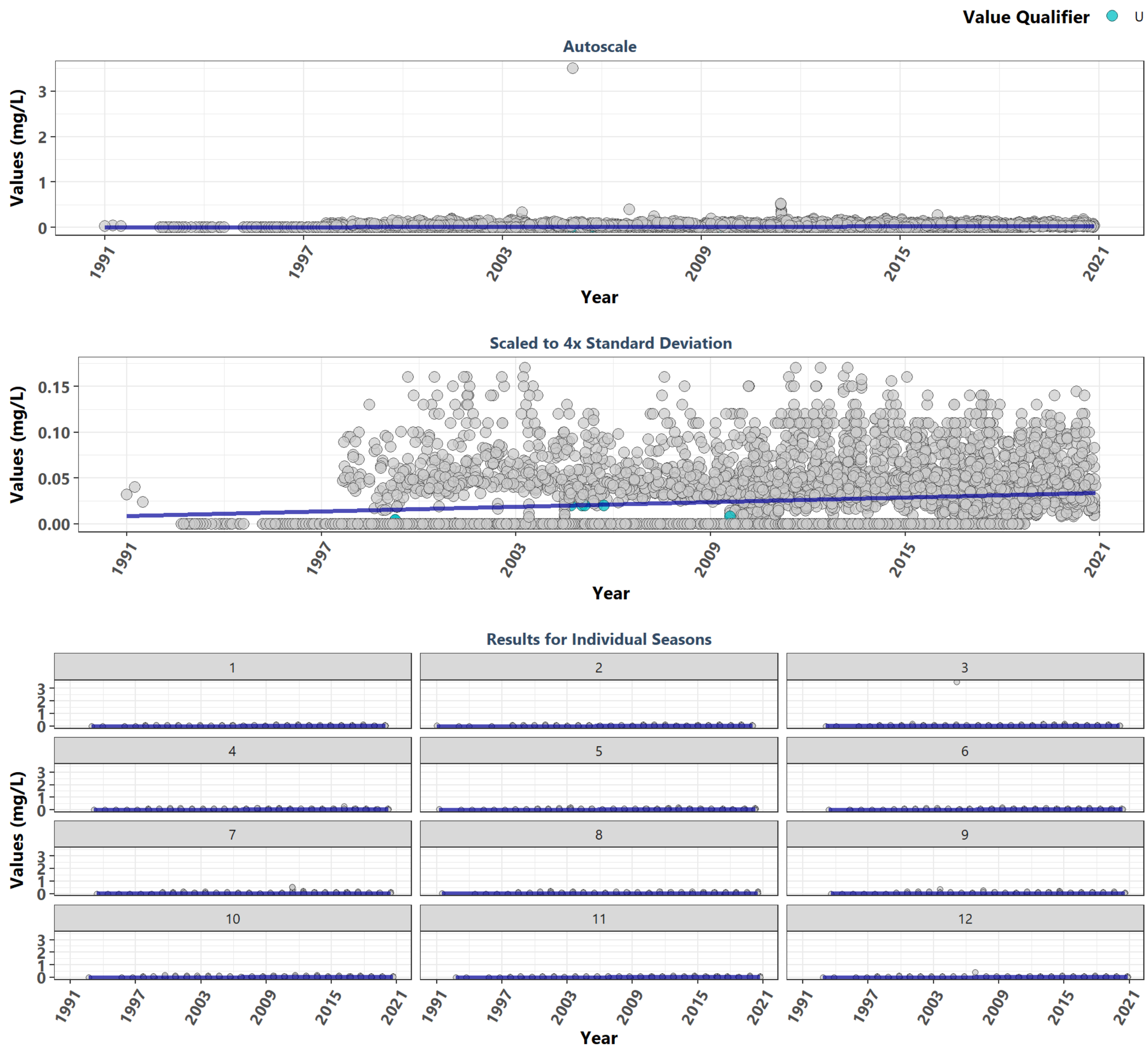
Banana River Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	3412	0.04	0.1314	0.0008	0.0286	11.5	0.0000	27.5	0.0039	1
1	293	0.04	0.0925	0.0006	0.0367	2.4	0.0181	NA	NA	1
2	277	0.03	-0.0030	0.0000	0.0330	-0.1	0.9414	NA	NA	-1
3	312	0.04	0.1410	0.0007	0.0278	4.2	0.0000	NA	NA	1
4	270	0.03	0.2057	0.0012	0.0214	2.8	0.0059	NA	NA	1
5	286	0.04	0.0598	0.0004	0.0375	2.9	0.0038	NA	NA	1
6	325	0.03	0.1579	0.0010	0.0263	3.8	0.0001	NA	NA	1
7	275	0.04	0.1838	0.0012	0.0257	5.1	0.0000	NA	NA	1
8	254	0.04	0.1148	0.0006	0.0299	4.4	0.0000	NA	NA	1
9	264	0.04	0.1124	0.0007	0.0277	1.4	0.1475	NA	NA	1
10	294	0.04	0.1592	0.0010	0.0306	5.6	0.0000	NA	NA	1
11	278	0.04	0.2200	0.0014	0.0258	3.1	0.0017	NA	NA	1
12	284	0.04	0.1258	0.0008	0.0294	4.0	0.0001	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

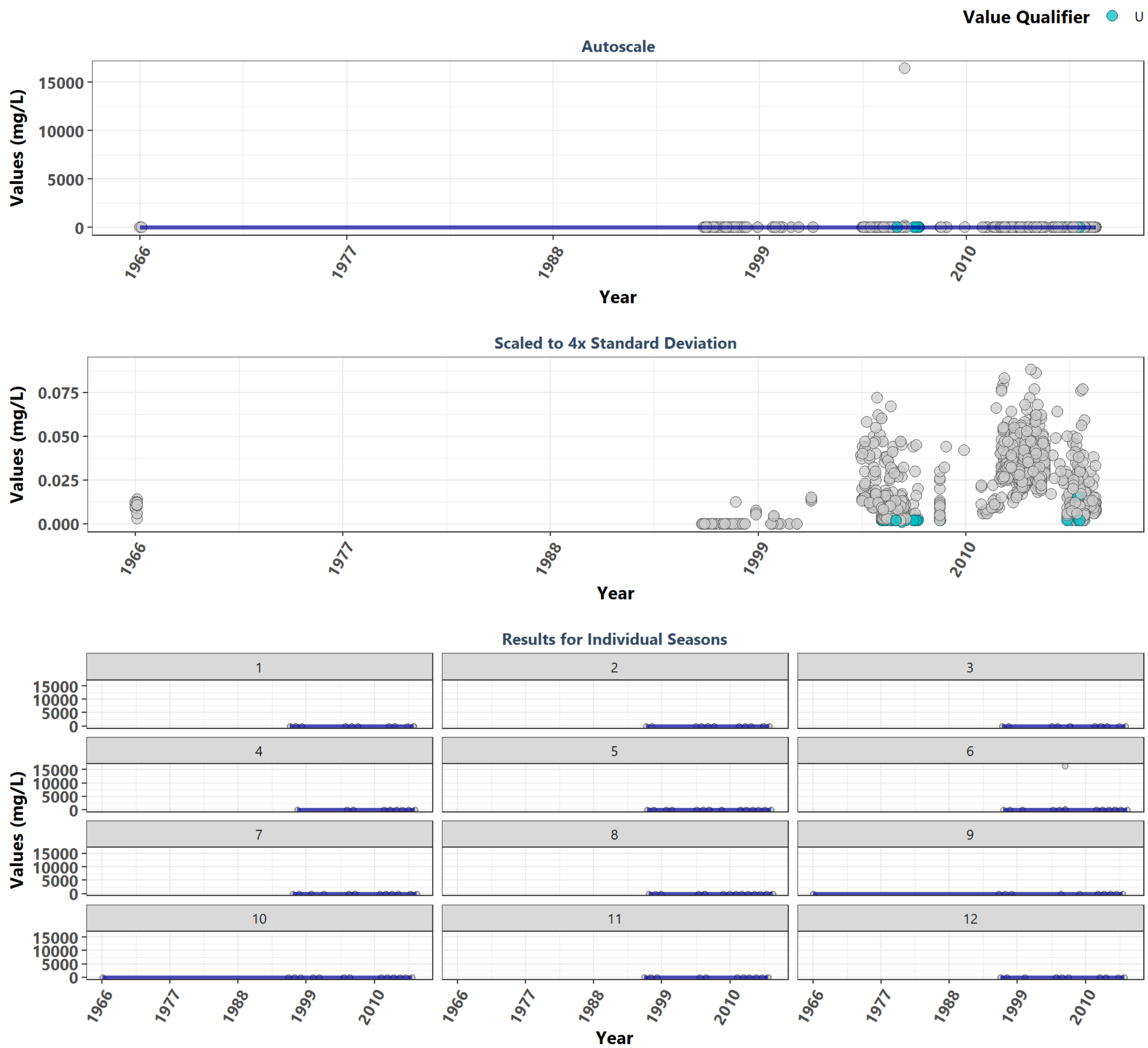
Big Bend Seagrasses Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	5052	0.03	0.1867	0.0008	0.0087	19.9	0.0000	13.4	0.268	1
1	352	0.01	0.1915	0.0007	0.0079	3.9	0.0001	NA	NA	1
2	434	0.02	0.1972	0.0011	0.0075	6.0	0.0000	NA	NA	1
3	452	0.03	0.1398	0.0000	0.0100	7.5	0.0000	NA	NA	1
4	398	0.02	0.1560	0.0007	0.0105	4.2	0.0000	NA	NA	1
5	457	0.03	0.1616	0.0007	0.0094	5.9	0.0000	NA	NA	1
6	430	0.03	0.1842	0.0010	0.0080	6.1	0.0000	NA	NA	1
7	405	0.03	0.2323	0.0016	0.0009	6.5	0.0000	NA	NA	1
8	456	0.03	0.1414	0.0000	0.0209	5.5	0.0000	NA	NA	1
9	404	0.03	0.2013	0.0010	0.0130	7.0	0.0000	NA	NA	1
10	445	0.03	0.1722	0.0011	0.0103	6.4	0.0000	NA	NA	1
11	373	0.02	0.2144	0.0013	0.0033	4.7	0.0000	NA	NA	1
12	446	0.02	0.2371	0.0012	0.0038	4.9	0.0000	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

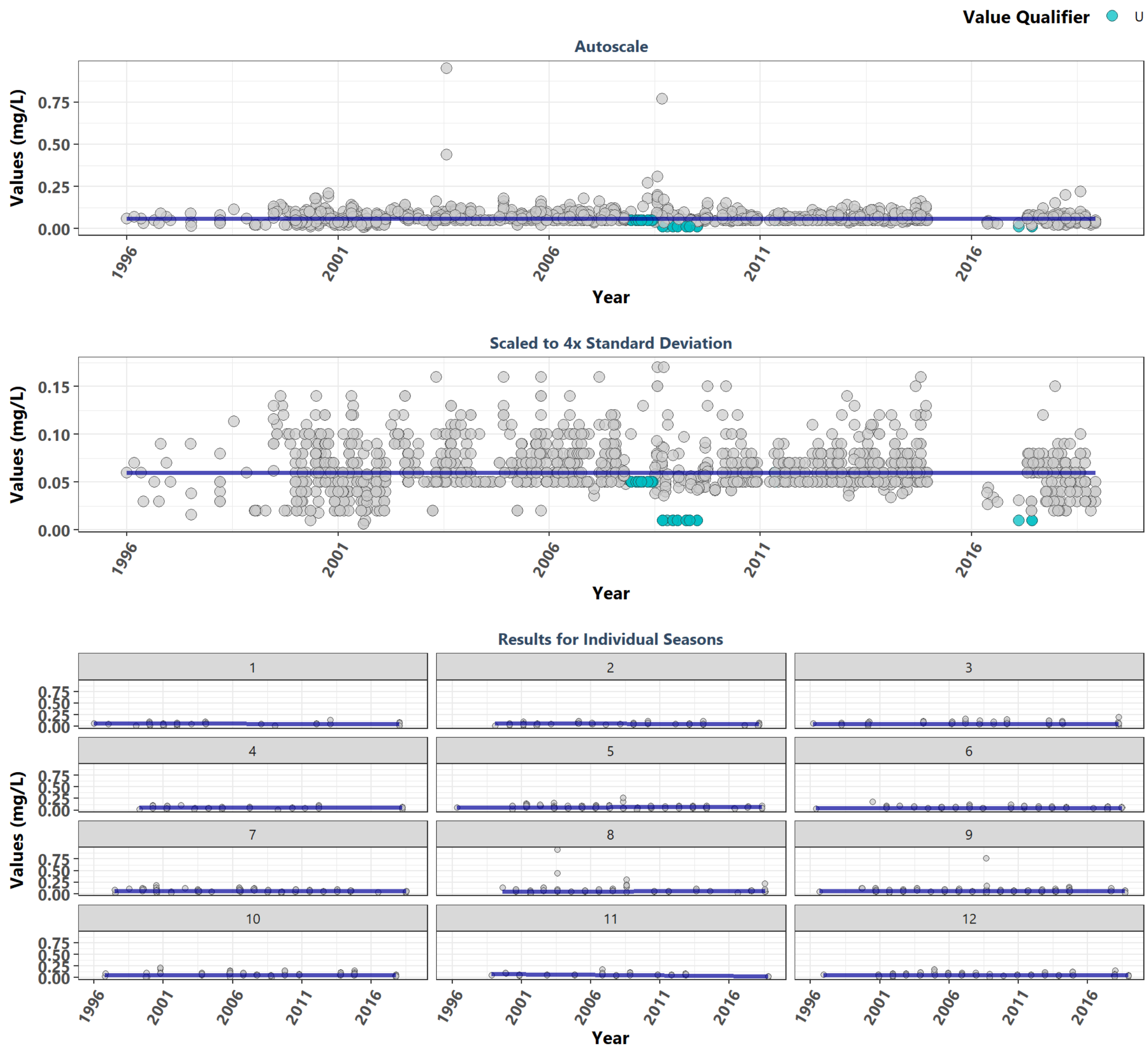
Biscayne Bay Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	2263	0.02	0.1566	0.0006	-0.0104	11.3	0.0000	86.1	0	1
1	188	0.01	0.2986	0.0017	-0.0669	4.5	0.0000	NA	NA	2
2	130	0.03	-0.0166	-0.0001	0.0297	-0.5	0.6519	NA	NA	-1
3	124	0.03	0.1959	0.0006	-0.0108	2.4	0.0173	NA	NA	1
4	138	0.01	0.2168	0.0004	-0.0104	5.3	0.0000	NA	NA	1
5	233	0.01	0.3399	0.0013	-0.0503	7.8	0.0000	NA	NA	2
6	193	0.02	-0.0614	-0.0003	0.0364	5.9	0.0000	NA	NA	-1
7	201	0.01	0.1263	0.0005	-0.0105	2.7	0.0068	NA	NA	1
8	228	0.03	0.2810	0.0017	-0.0606	-0.4	0.7046	NA	NA	1
9	196	0.02	0.1389	0.0016	-0.0484	-1.3	0.1921	NA	NA	1
10	229	0.03	0.1352	0.0006	-0.0002	4.9	0.0000	NA	NA	1
11	211	0.03	0.2157	0.0006	0.0000	3.0	0.0030	NA	NA	1
12	192	0.02	-0.0261	-0.0003	0.0470	4.1	0.0000	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

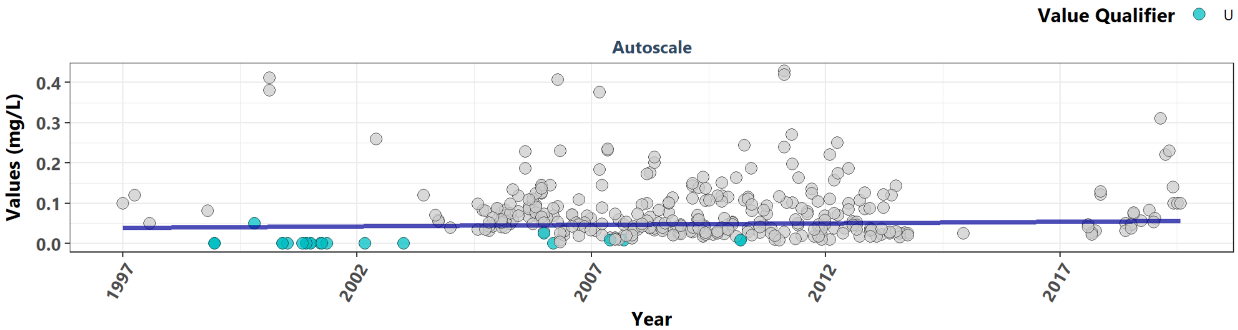
Boca Ciega Bay Aquatic Preserve

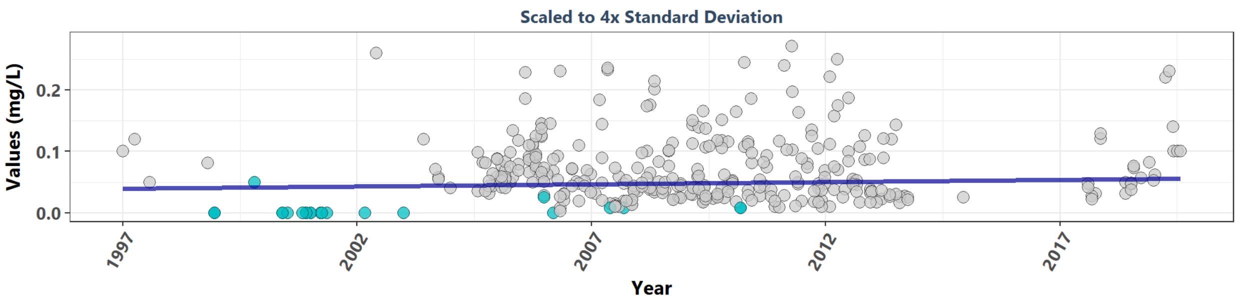


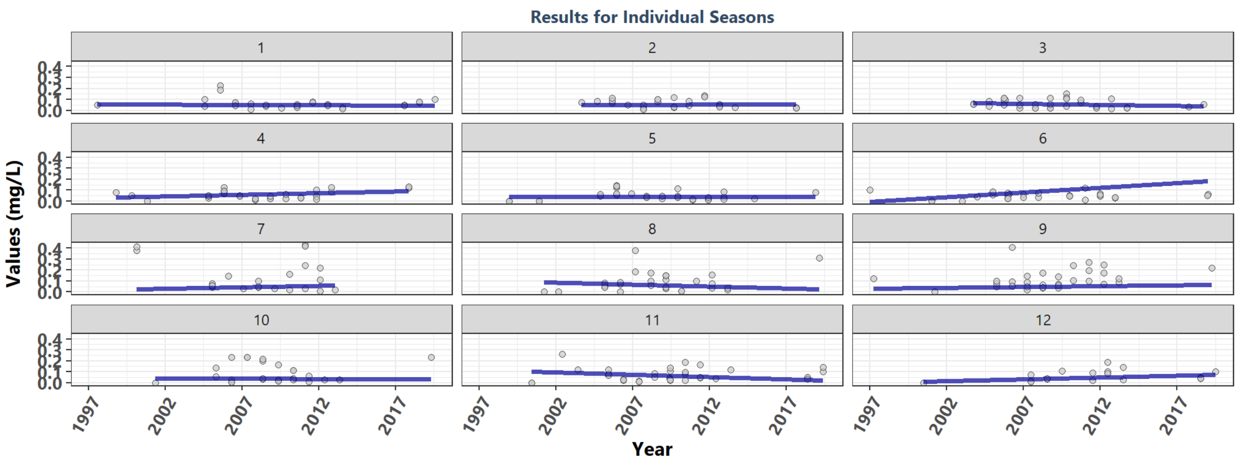
Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	1581	0.06	-0.0670	0.0000	0.0600	-4.7	0.0000	52.9	0	-1
1	82	0.05	-0.1673	-0.0008	0.0700	1.7	0.0804	NA	NA	-1
2	101	0.05	-0.1716	-0.0009	0.0691	0.5	0.6182	NA	NA	-1
3	97	0.06	-0.0148	0.0000	0.0500	0.0	0.9873	NA	NA	-1
4	86	0.05	0.0013	0.0000	0.0600	-0.2	0.8357	NA	NA	-1
5	221	0.06	0.1172	0.0006	0.0531	-2.0	0.0481	NA	NA	1
6	132	0.06	0.0333	0.0000	0.0500	-0.5	0.6353	NA	NA	-1
7	161	0.06	-0.0543	0.0000	0.0600	-3.3	0.0011	NA	NA	-1
8	134	0.06	0.1280	0.0005	0.0464	2.0	0.0414	NA	NA	1
9	211	0.06	-0.0873	0.0000	0.0600	-3.7	0.0003	NA	NA	-1
10	125	0.06	-0.0272	0.0000	0.0600	-5.3	0.0000	NA	NA	-1
11	63	0.06	-0.3172	-0.0023	0.0897	-0.6	0.5196	NA	NA	-1
12	168	0.06	-0.0325	0.0000	0.0600	-0.6	0.5230	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

Cape Haze Aquatic Preserve



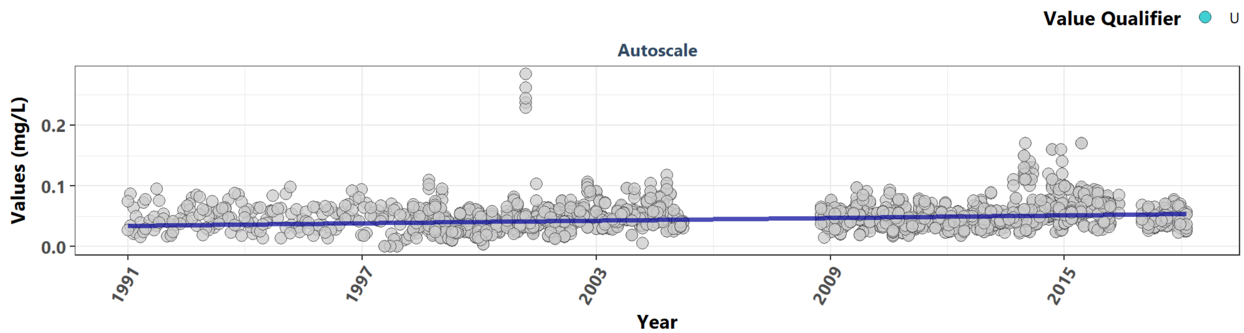


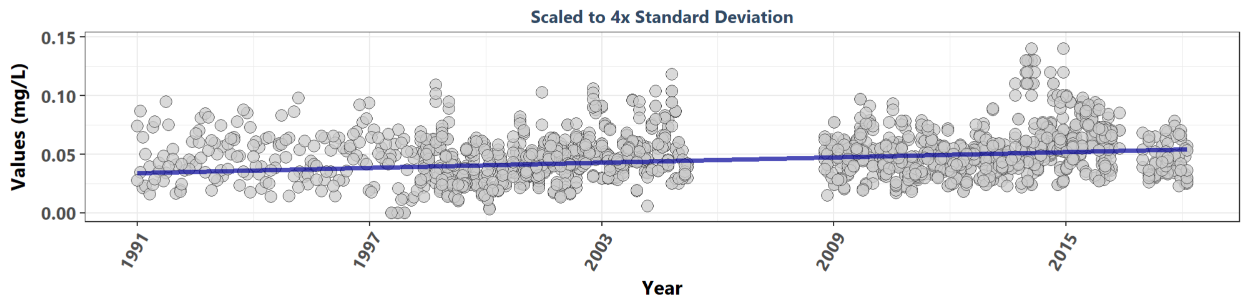


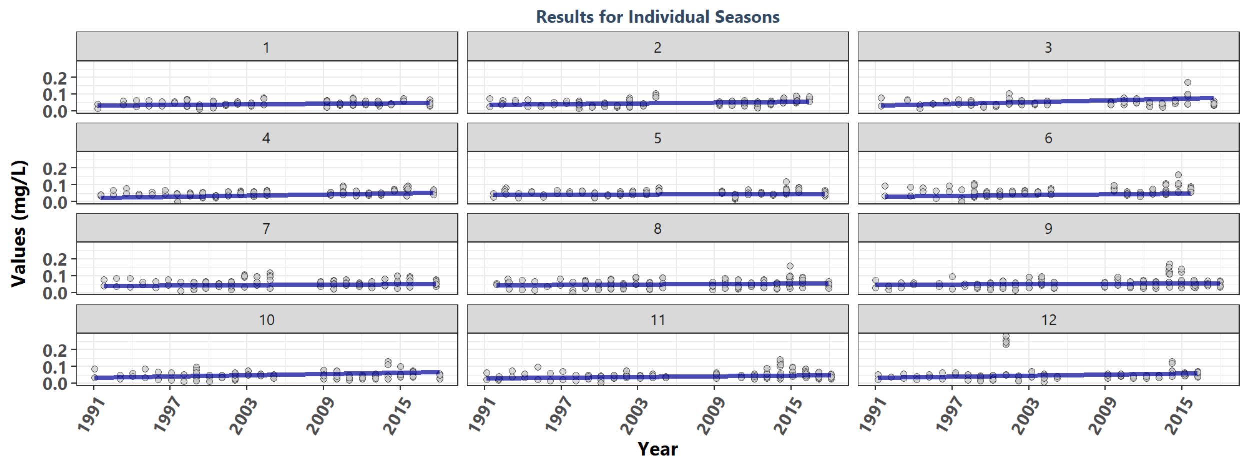
Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	357	0.05	0.0397	0.0007	0.0394	1.0	0.3075	18.3	0.075	0
1	28	0.05	-0.0450	-0.0005	0.0563	-0.3	0.7508	NA	NA	0
2	28	0.06	0.0230	0.0002	0.0490	-1.4	0.1570	NA	NA	0
3	34	0.06	-0.1230	-0.0017	0.0761	-1.0	0.3097	NA	NA	0
4	33	0.05	0.0851	0.0028	0.0312	0.7	0.5019	NA	NA	0
5	38	0.04	-0.0199	-0.0001	0.0388	-0.2	0.8690	NA	NA	0
6	30	0.05	0.2841	0.0087	-0.0063	0.2	0.8716	NA	NA	0
7	22	0.07	0.1686	0.0025	0.0200	-0.9	0.3617	NA	NA	0
8	30	0.06	-0.1905	-0.0033	0.1012	0.6	0.5162	NA	NA	0
9	33	0.09	0.0833	0.0015	0.0305	2.3	0.0201	NA	NA	0
10	27	0.04	-0.0256	-0.0003	0.0400	-0.2	0.8663	NA	NA	0
11	33	0.05	-0.1429	-0.0044	0.1183	1.4	0.1697	NA	NA	0
12	21	0.05	0.4095	0.0033	0.0002	2.6	0.0092	NA	NA	0

[a] $p < 0.00005$ appear as 0 due to rounding

Cape Romano-Ten Thousand Islands Aquatic Preserve



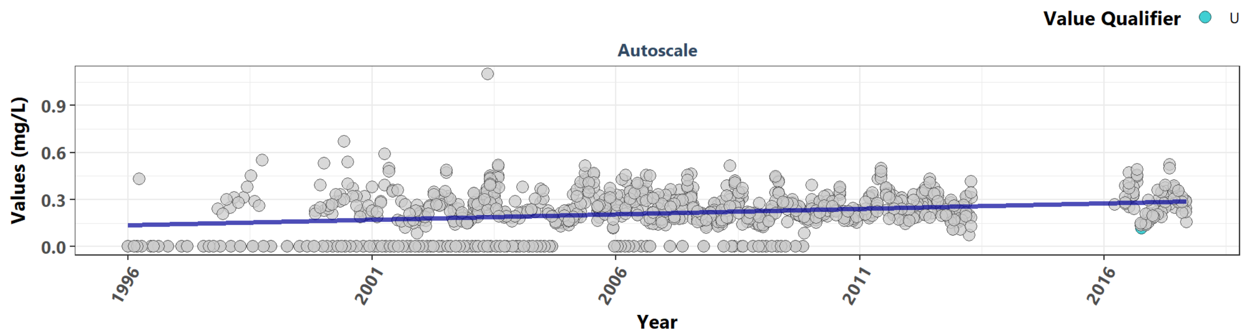


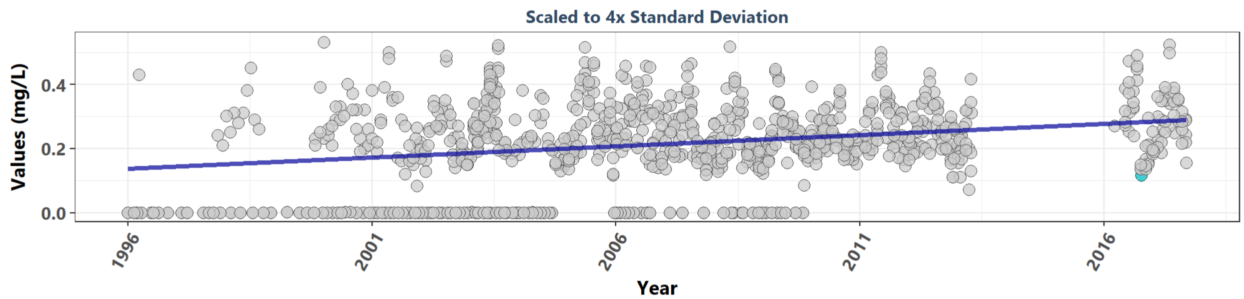


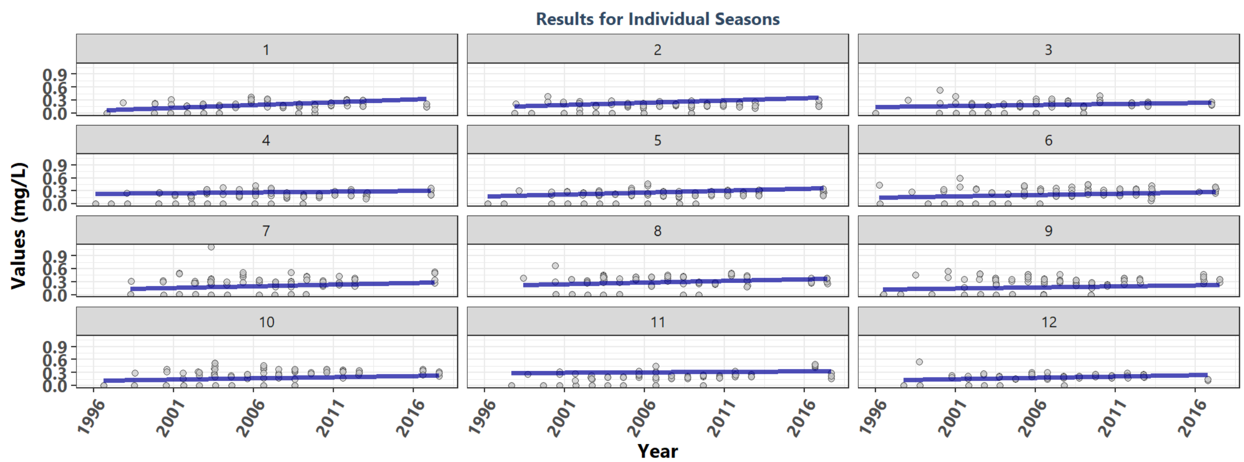
Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	1682	0.05	0.1807	0.0007	0.0339	11.0	0.0000	28.6	0.0026	1
1	132	0.04	0.1658	0.0006	0.0334	3.4	0.0007	NA	NA	1
2	139	0.04	0.1628	0.0007	0.0357	5.6	0.0000	NA	NA	1
3	133	0.04	0.2773	0.0016	0.0330	2.2	0.0248	NA	NA	1
4	141	0.05	0.3222	0.0012	0.0219	5.6	0.0000	NA	NA	1
5	143	0.05	0.0352	0.0002	0.0422	1.4	0.1555	NA	NA	1
6	138	0.05	0.2306	0.0008	0.0280	4.8	0.0000	NA	NA	1
7	142	0.05	0.1312	0.0004	0.0383	1.5	0.1299	NA	NA	1
8	152	0.05	0.0857	0.0004	0.0430	3.0	0.0026	NA	NA	1
9	152	0.05	0.0801	0.0003	0.0459	3.0	0.0029	NA	NA	1
10	132	0.04	0.3172	0.0012	0.0318	0.6	0.5503	NA	NA	1
11	149	0.04	0.1984	0.0007	0.0310	3.0	0.0027	NA	NA	1
12	129	0.04	0.1647	0.0010	0.0344	3.9	0.0001	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

Cockroach Bay Aquatic Preserve



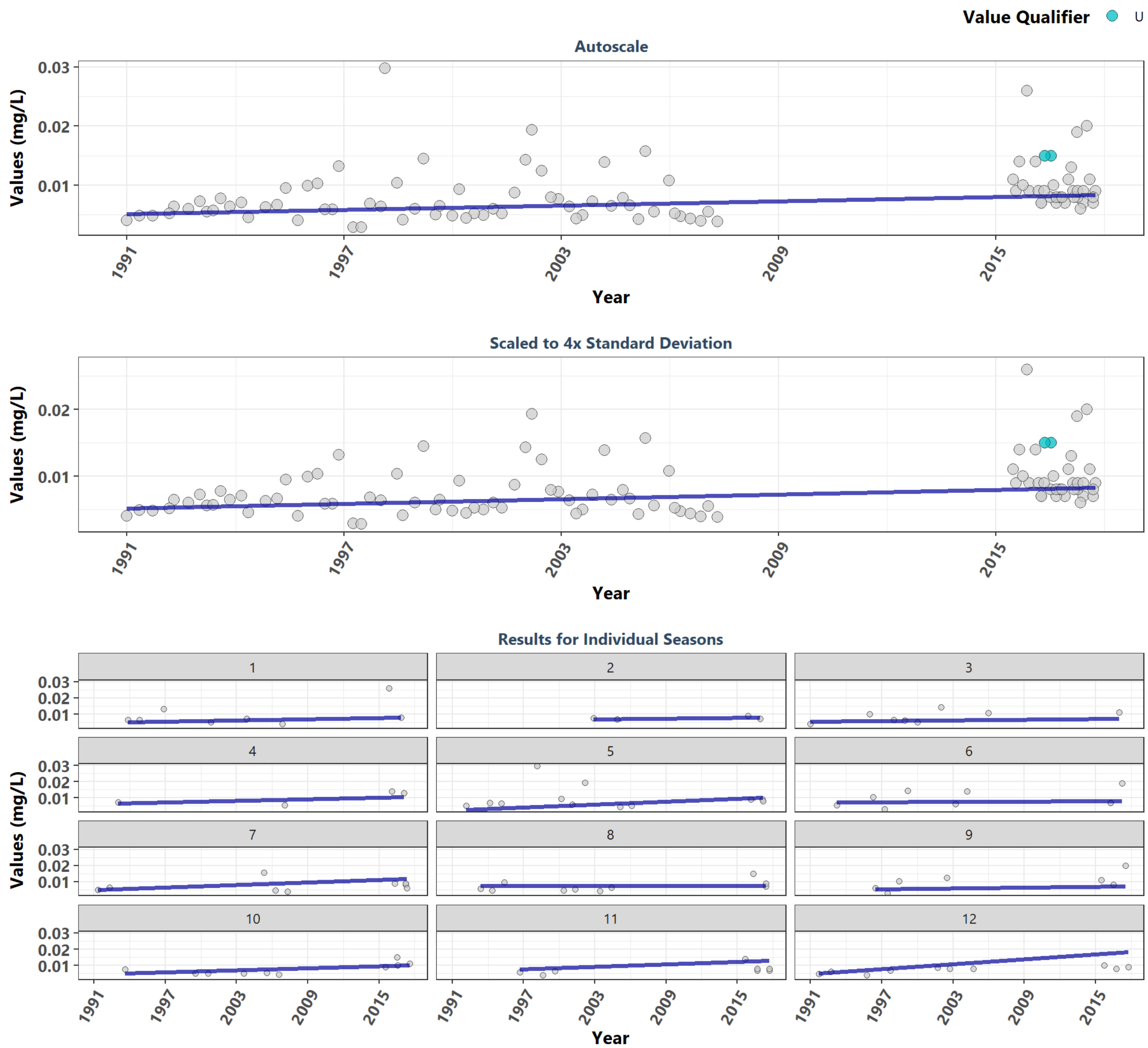




Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	1090	0.21	0.2032	0.0070	0.1376	9.9	0.0000	10.9	0.4557	1
1	88	0.17	0.3524	0.0123	0.0643	2.4	0.0160	NA	NA	1
2	92	0.17	0.2390	0.0105	0.1355	2.8	0.0048	NA	NA	1
3	79	0.18	0.1577	0.0051	0.1397	2.7	0.0060	NA	NA	1
4	99	0.19	0.1294	0.0035	0.2349	3.4	0.0006	NA	NA	1
5	102	0.21	0.2416	0.0090	0.1670	3.7	0.0003	NA	NA	1
6	88	0.24	0.2448	0.0062	0.1503	3.3	0.0010	NA	NA	1
7	96	0.27	0.2334	0.0075	0.1146	1.9	0.0613	NA	NA	1
8	90	0.30	0.1763	0.0075	0.2020	1.0	0.3017	NA	NA	1
9	96	0.28	0.1742	0.0050	0.1170	2.5	0.0108	NA	NA	1
10	100	0.26	0.1992	0.0056	0.1128	3.6	0.0004	NA	NA	1
11	86	0.19	0.0739	0.0027	0.2767	4.8	0.0000	NA	NA	1
12	74	0.19	0.2103	0.0059	0.1222	2.0	0.0466	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

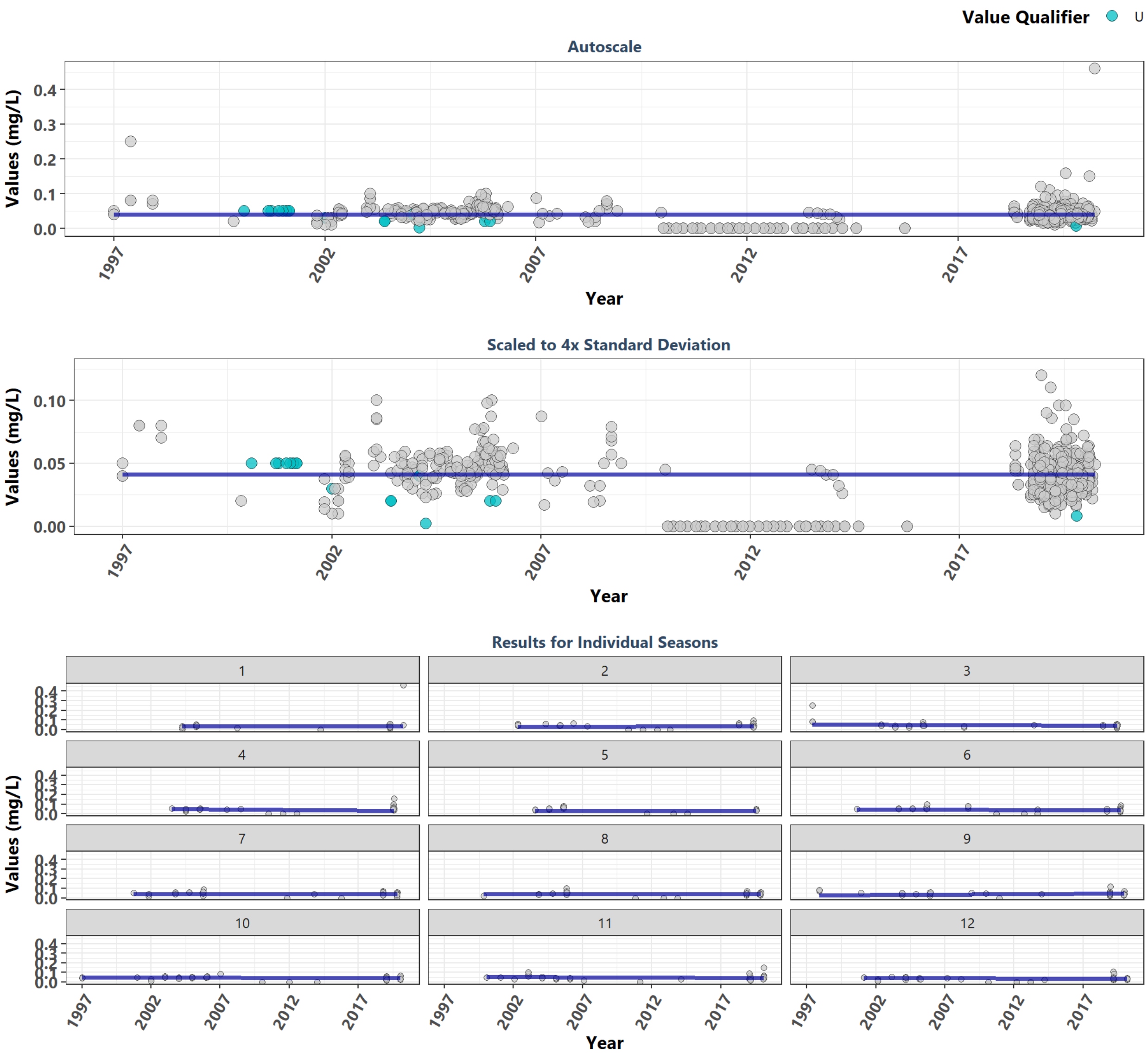
Coupon Bight Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	99	0.01	0.3185	0.0001	0.0051	3.8	0.0001	6.3	0.8519	1
1	8	0.01	0.2667	0.0001	0.0048	0.4	0.7105	NA	NA	1
2	4	0.01	0.1429	0.0001	0.0060	0.0	1.0000	NA	NA	1
3	8	0.01	0.1111	0.0001	0.0054	1.4	0.1735	NA	NA	1
4	4	0.01	0.6444	0.0002	0.0062	0.3	0.7341	NA	NA	1
5	12	0.01	0.4444	0.0003	0.0023	0.2	0.8363	NA	NA	1
6	9	0.01	0.0606	0.0000	0.0071	1.3	0.2060	NA	NA	1
7	9	0.01	0.3333	0.0003	0.0047	0.3	0.7483	NA	NA	1
8	10	0.01	0.0000	0.0000	0.0075	1.0	0.3232	NA	NA	-1
9	7	0.01	0.4286	0.0001	0.0048	1.5	0.1331	NA	NA	1
10	10	0.01	0.3611	0.0002	0.0044	1.7	0.0880	NA	NA	1
11	8	0.01	0.4286	0.0003	0.0060	1.4	0.1606	NA	NA	1
12	10	0.01	0.5238	0.0005	0.0049	2.5	0.0123	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

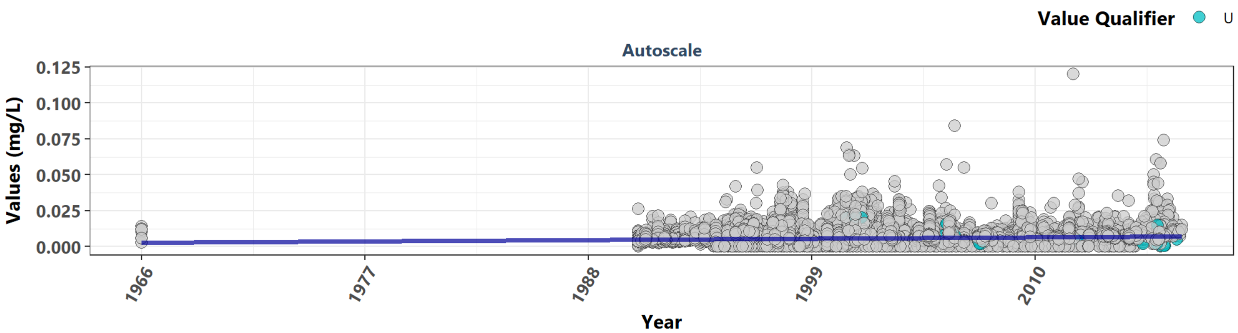
Estero Bay Aquatic Preserve

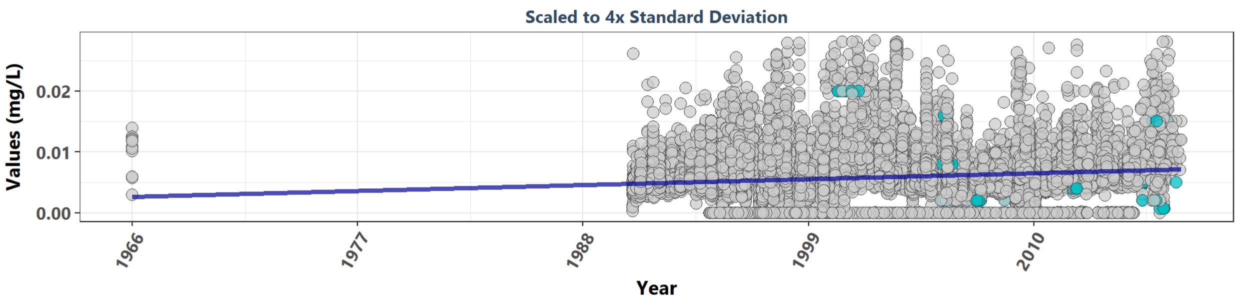


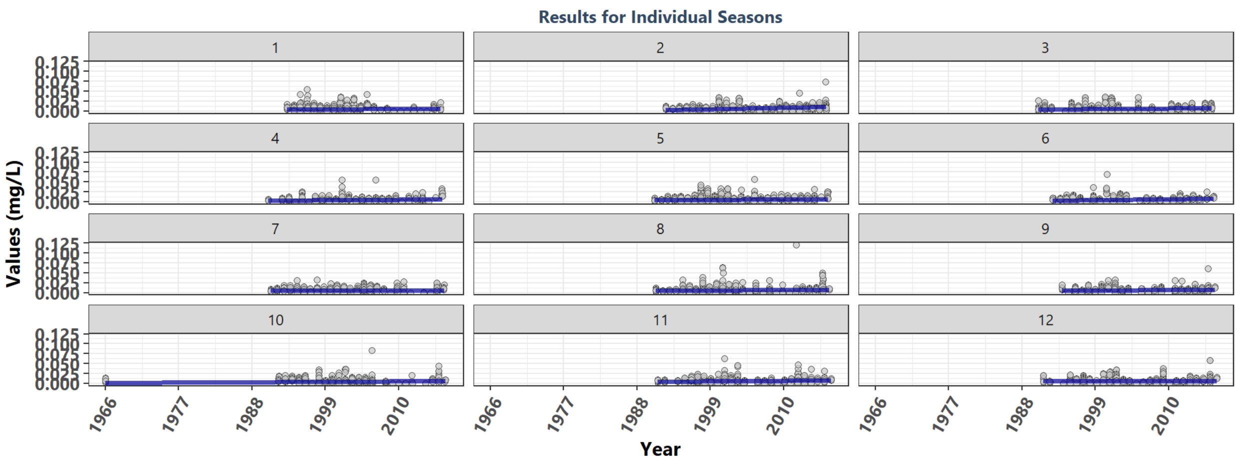
Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	678	0.04	-0.0151	0.0000	0.0413	-0.7	0.5053	17.8	0.0856	0
1	41	0.03	0.0573	0.0002	0.0318	0.8	0.3959	NA	NA	0
2	60	0.04	0.0829	0.0003	0.0253	0.4	0.6885	NA	NA	0
3	50	0.04	-0.0885	-0.0005	0.0520	-0.8	0.4341	NA	NA	0
4	45	0.05	-0.1232	-0.0010	0.0565	2.5	0.0119	NA	NA	0
5	24	0.04	0.0000	0.0000	0.0300	-0.8	0.4063	NA	NA	0
6	53	0.04	-0.0732	-0.0004	0.0446	-1.0	0.3246	NA	NA	0
7	63	0.04	0.0339	0.0000	0.0412	-0.9	0.3802	NA	NA	0
8	73	0.04	0.0407	0.0001	0.0363	0.5	0.5993	NA	NA	0
9	64	0.04	0.2364	0.0010	0.0250	-2.5	0.0135	NA	NA	0
10	76	0.04	-0.0718	-0.0002	0.0468	-0.8	0.4103	NA	NA	0
11	75	0.04	-0.2019	-0.0007	0.0562	0.8	0.4514	NA	NA	0
12	54	0.03	-0.0628	-0.0002	0.0413	0.0	1.0000	NA	NA	0

[a] $p < 0.00005$ appear as 0 due to rounding

Florida Keys National Marine Sanctuary



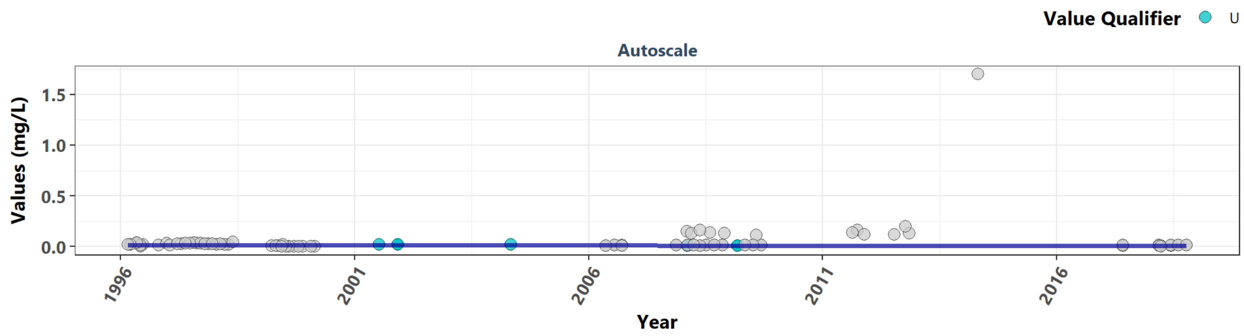


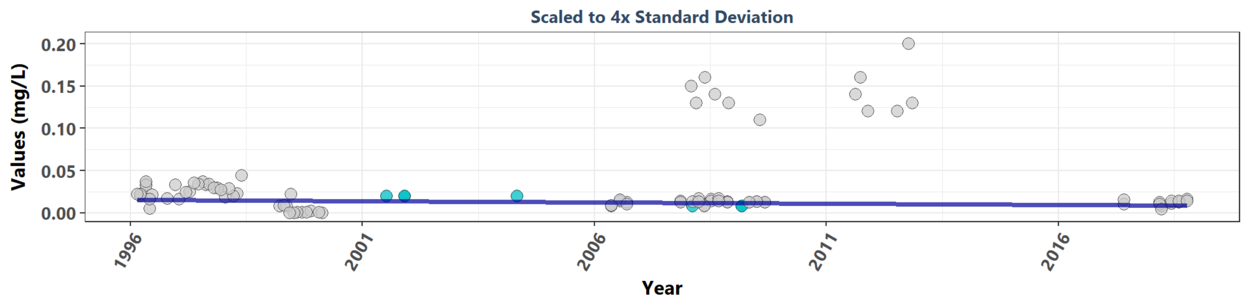


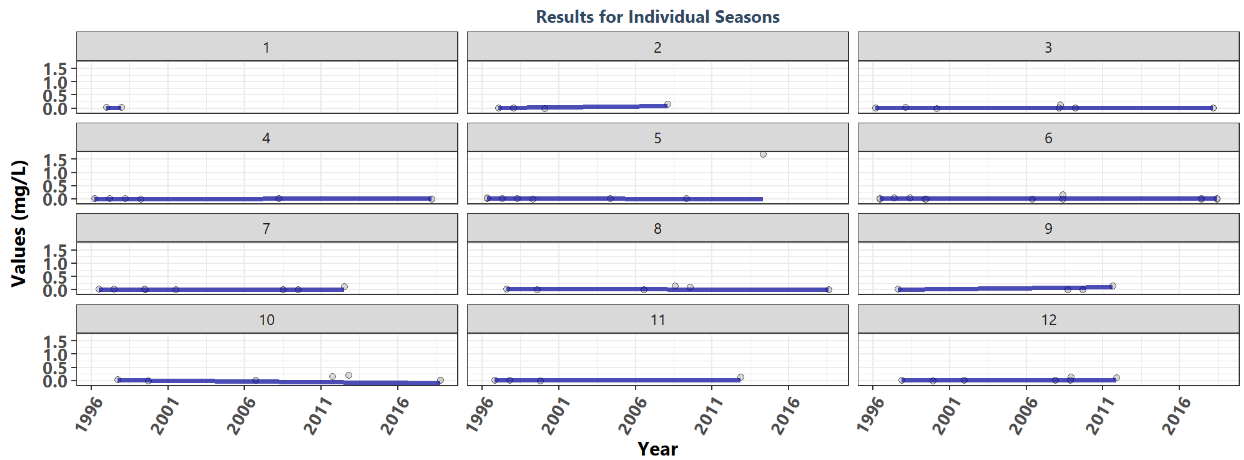
Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	18008	0.01	0.1281	0.0001	0.0026	25.9	0.0000	260.9	0	1
1	1304	0.01	0.1694	0.0001	0.0017	-2.8	0.0056	NA	NA	1
2	1590	0.01	0.2805	0.0003	-0.0039	12.5	0.0000	NA	NA	1
3	1584	0.01	0.1273	0.0001	0.0024	9.3	0.0000	NA	NA	1
4	1239	0.01	0.1365	0.0001	0.0014	7.2	0.0000	NA	NA	1
5	1809	0.01	0.0790	0.0000	0.0042	8.1	0.0000	NA	NA	1
6	1201	0.01	0.2085	0.0002	-0.0016	4.5	0.0000	NA	NA	1
7	1816	0.01	0.0528	0.0000	0.0045	5.9	0.0000	NA	NA	1
8	1475	0.01	0.0927	0.0001	0.0035	16.2	0.0000	NA	NA	1
9	1478	0.01	0.0870	0.0000	0.0050	10.2	0.0000	NA	NA	1
10	1551	0.01	0.1772	0.0001	0.0011	3.1	0.0017	NA	NA	1
11	1512	0.01	0.1560	0.0001	0.0028	4.6	0.0000	NA	NA	1
12	1449	0.01	-0.0511	0.0000	0.0074	9.7	0.0000	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

Fort Pickens State Park Aquatic Preserve



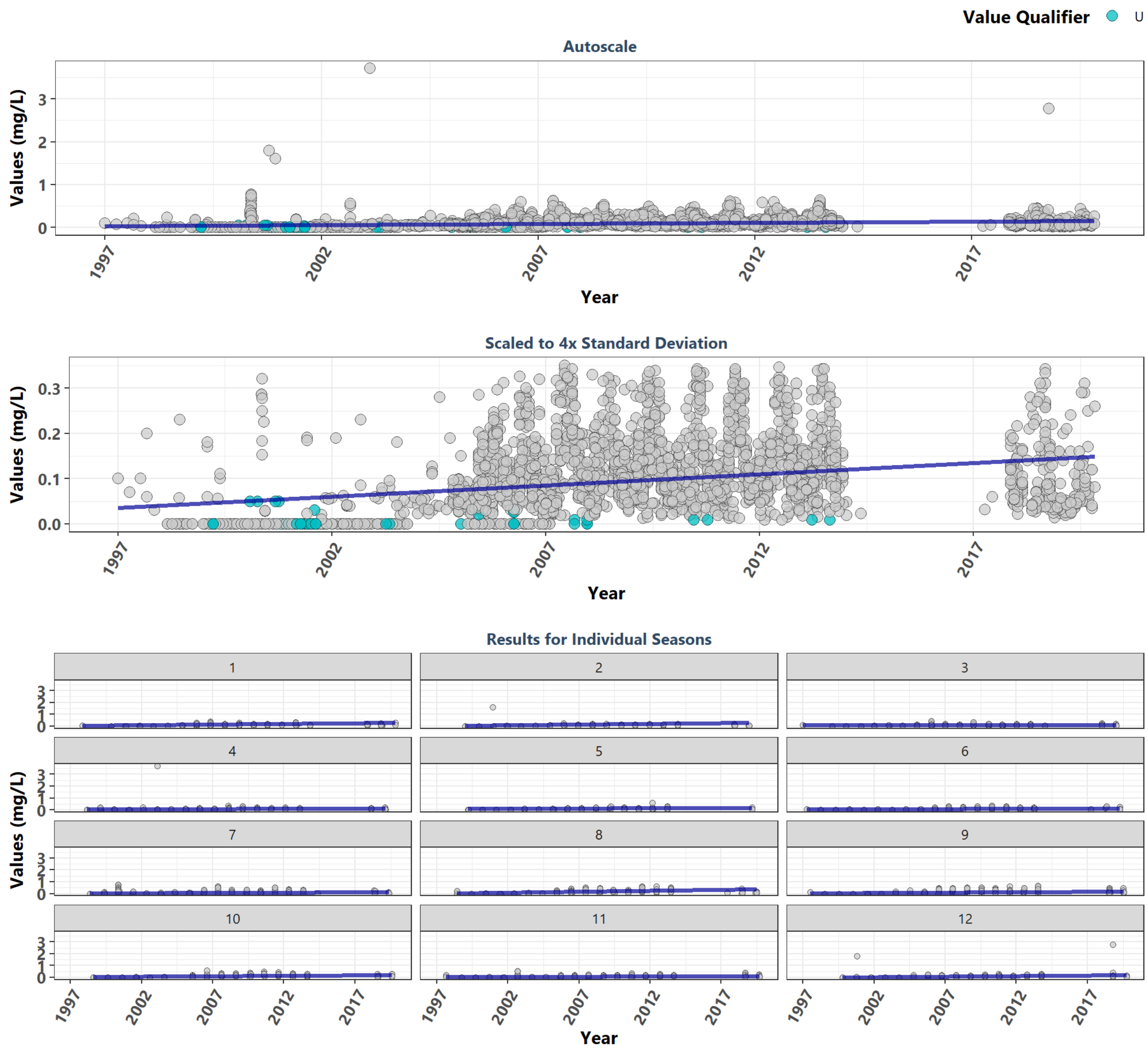




Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	98	0.02	-0.1034	-0.0003	0.0152	-1.2	0.2444	11.8	0.3803	0
1	2	NA	0.0667	0.0011	0.0052	NA	NA	NA	NA	NA
2	4	0.02	0.3333	0.0069	0.0092	0.3	0.7341	NA	NA	0
3	10	0.01	0.2857	0.0006	0.0073	-0.5	0.6425	NA	NA	0
4	10	0.02	0.0667	0.0001	0.0095	-2.4	0.0165	NA	NA	0
5	10	0.02	-0.3778	-0.0015	0.0278	-1.5	0.1432	NA	NA	0
6	16	0.01	-0.0357	0.0000	0.0150	0.3	0.7492	NA	NA	0
7	10	0.02	-0.1333	-0.0004	0.0155	-0.8	0.4134	NA	NA	0
8	8	0.01	-0.2000	-0.0017	0.0371	0.0	1.0000	NA	NA	0
9	6	0.02	0.3333	0.0071	-0.0043	-0.4	0.6919	NA	NA	0
10	8	0.01	-1.0000	-0.0060	0.0390	0.9	0.3791	NA	NA	0
11	4	0.02	-0.2222	-0.0006	0.0232	0.3	0.7341	NA	NA	0
12	10	0.02	-0.6000	-0.0007	0.0203	0.2	0.8537	NA	NA	0

[a] p < 0.00005 appear as 0 due to rounding

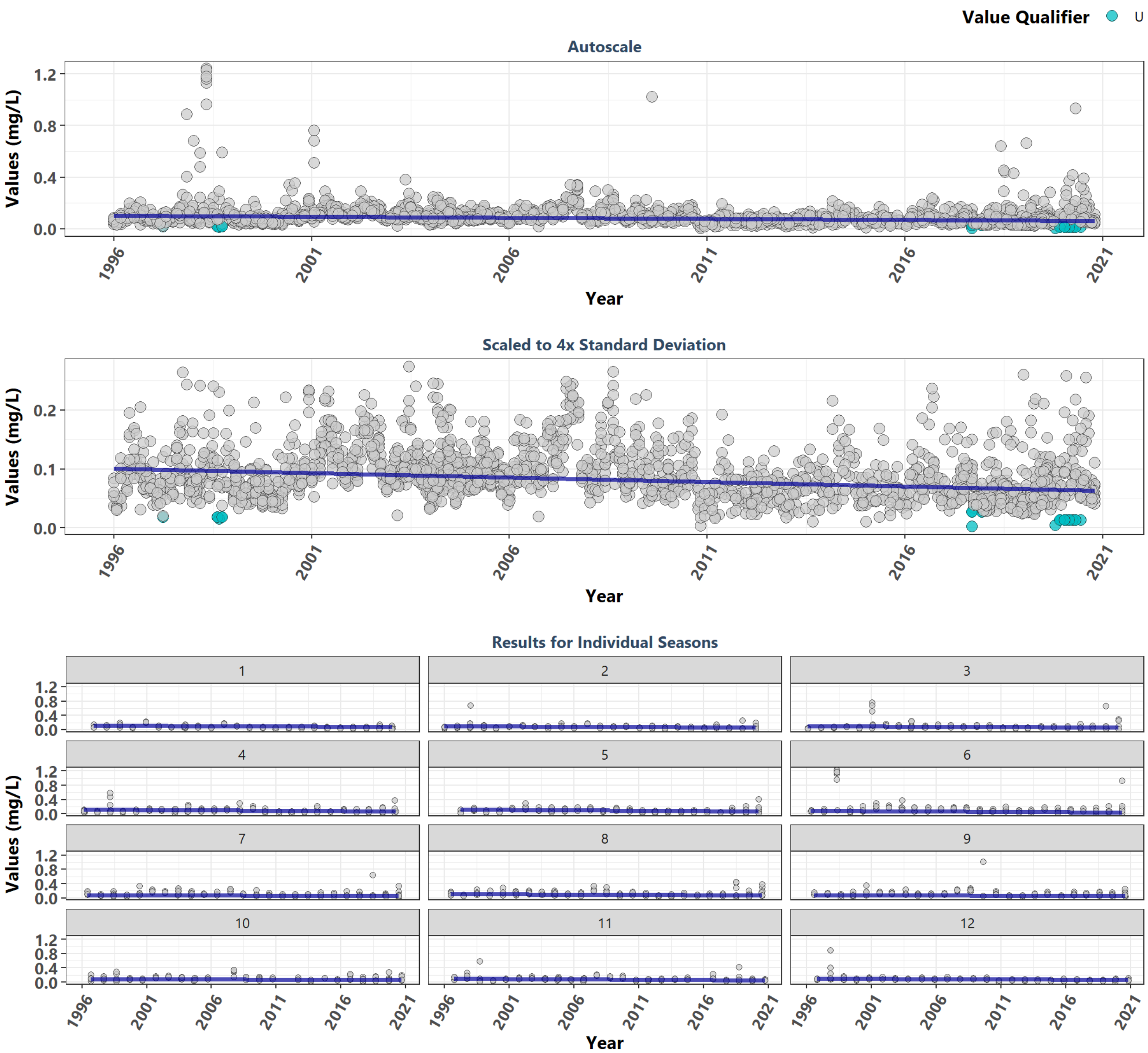
Gasparilla Sound-Charlotte Harbor Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	2957	0.09	0.1904	0.0050	0.0350	15.6	0.0000	73.2	0	1
1	228	0.08	0.2264	0.0117	0.0350	1.6	0.1055	NA	NA	2
2	243	0.07	0.2365	0.0103	0.0330	3.2	0.0012	NA	NA	2
3	230	0.07	0.0504	0.0010	0.0805	3.1	0.0020	NA	NA	1
4	253	0.07	0.0718	0.0014	0.0624	4.2	0.0000	NA	NA	1
5	218	0.08	0.0292	0.0010	0.1146	6.8	0.0000	NA	NA	1
6	268	0.08	0.1361	0.0024	0.0427	9.2	0.0000	NA	NA	1
7	247	0.13	0.1780	0.0030	0.0350	0.7	0.4929	NA	NA	1
8	251	0.17	0.2455	0.0150	0.0200	5.4	0.0000	NA	NA	1
9	250	0.20	0.3063	0.0067	0.0007	5.8	0.0000	NA	NA	1
10	237	0.16	0.2689	0.0070	0.0240	5.4	0.0000	NA	NA	1
11	278	0.11	0.1385	0.0023	0.0410	6.7	0.0000	NA	NA	1
12	254	0.09	0.3768	0.0092	-0.0230	1.2	0.2301	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

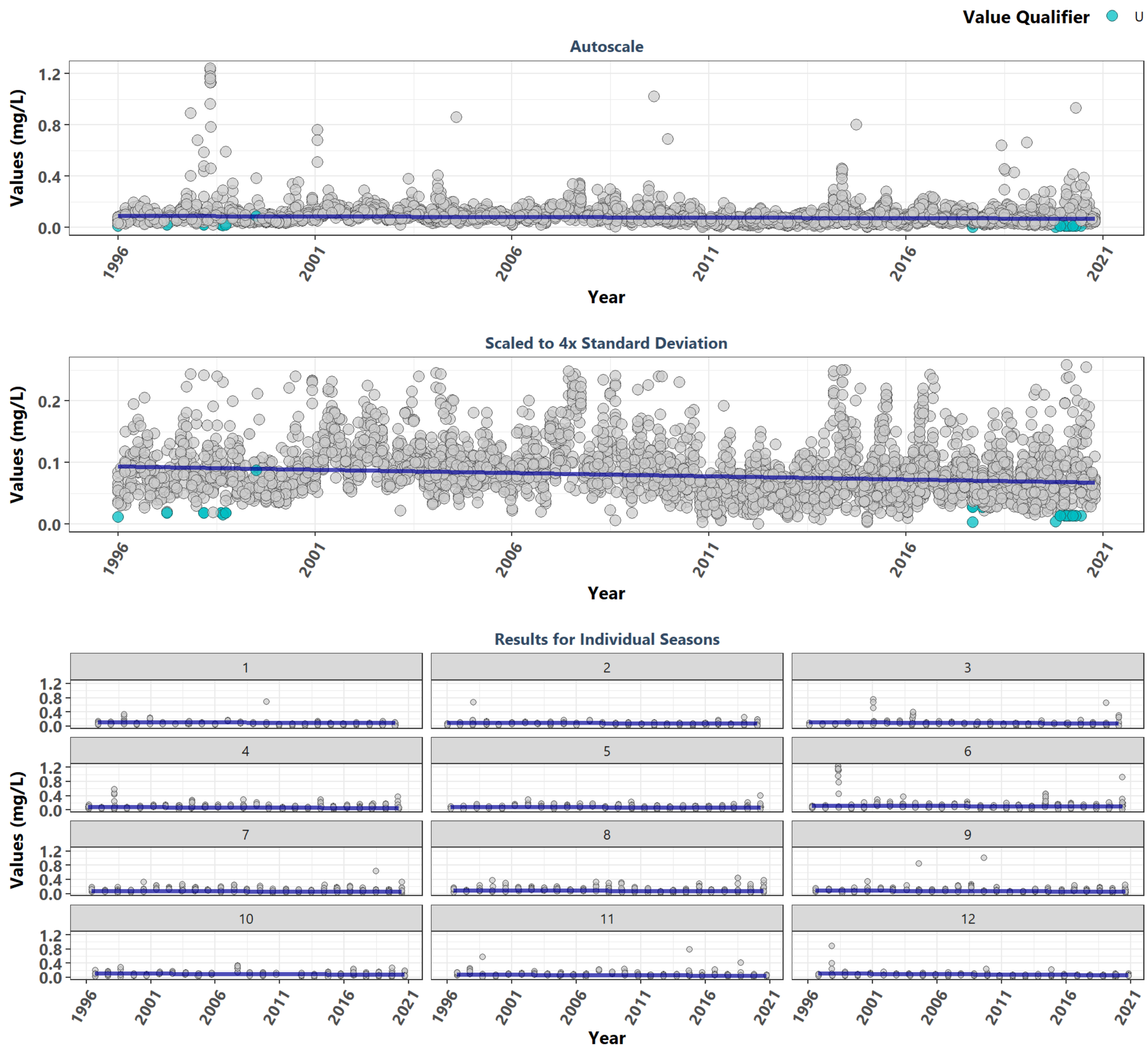
Guana River Marsh Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	1951	0.08	-0.1957	-0.0015	0.1009	-12.7	0.0000	26.7	0.0052	-1
1	160	0.07	-0.1469	-0.0013	0.1170	-6.9	0.0000	NA	NA	-1
2	166	0.07	-0.1660	-0.0017	0.1035	-3.2	0.0013	NA	NA	-1
3	163	0.07	-0.1904	-0.0015	0.0983	-2.3	0.0213	NA	NA	-1
4	175	0.08	-0.1692	-0.0018	0.1163	-2.4	0.0150	NA	NA	-1
5	155	0.08	-0.2407	-0.0021	0.1188	-3.1	0.0022	NA	NA	-1
6	164	0.10	-0.3680	-0.0022	0.0923	-3.2	0.0013	NA	NA	-1
7	162	0.10	-0.1684	-0.0010	0.0775	-4.6	0.0000	NA	NA	-1
8	189	0.10	-0.1496	-0.0013	0.1139	-3.1	0.0022	NA	NA	-1
9	153	0.10	-0.1214	-0.0010	0.0818	-2.7	0.0070	NA	NA	-1
10	148	0.09	-0.1237	-0.0010	0.0873	-2.9	0.0042	NA	NA	-1
11	167	0.08	-0.3704	-0.0019	0.0984	-3.7	0.0003	NA	NA	-1
12	149	0.08	-0.1584	-0.0014	0.1060	-6.7	0.0000	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

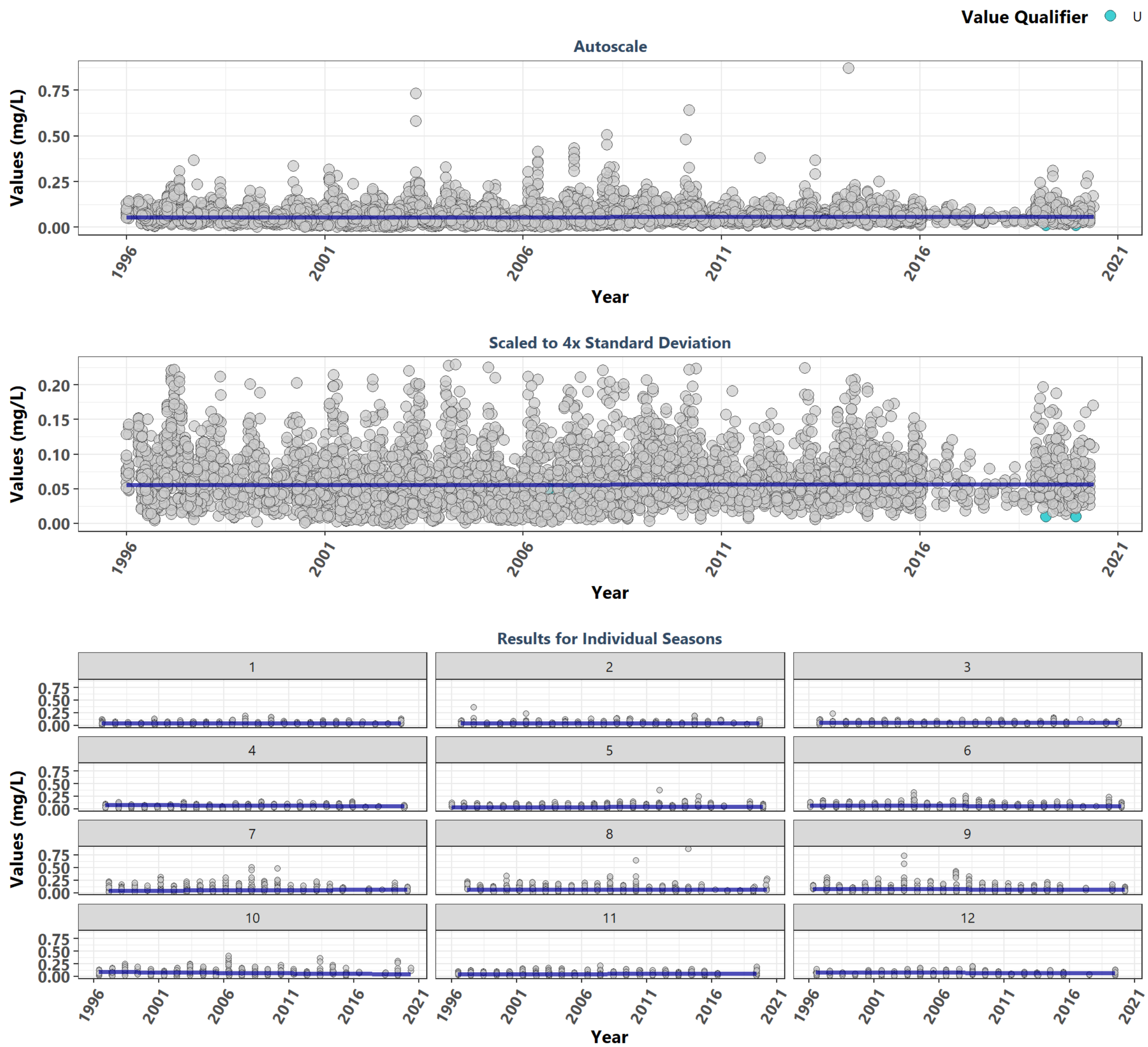
Guana Tolomato Matanzas National Estuarine Research Reserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	3928	0.08	-0.1498	-0.0011	0.0938	-13.8	0.0000	39.9	0	-1
1	335	0.06	-0.0609	-0.0007	0.1111	-6.9	0.0000	NA	NA	-1
2	335	0.06	-0.0990	-0.0007	0.0892	-3.7	0.0002	NA	NA	-1
3	336	0.07	-0.1218	-0.0010	0.1092	-4.1	0.0000	NA	NA	-1
4	365	0.08	-0.2517	-0.0015	0.0897	-2.8	0.0047	NA	NA	-1
5	333	0.08	-0.1499	-0.0010	0.0833	-2.0	0.0480	NA	NA	-1
6	330	0.10	-0.0864	-0.0008	0.1127	-1.7	0.0983	NA	NA	-1
7	326	0.10	-0.1356	-0.0008	0.0758	-2.3	0.0197	NA	NA	-1
8	355	0.09	-0.0725	-0.0007	0.0947	-3.4	0.0006	NA	NA	-1
9	304	0.09	-0.2511	-0.0016	0.0929	-3.3	0.0011	NA	NA	-1
10	283	0.08	-0.1256	-0.0010	0.1065	-5.1	0.0000	NA	NA	-1
11	321	0.07	-0.2644	-0.0013	0.0848	-6.7	0.0000	NA	NA	-1
12	305	0.06	-0.2016	-0.0016	0.1112	-6.9	0.0000	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

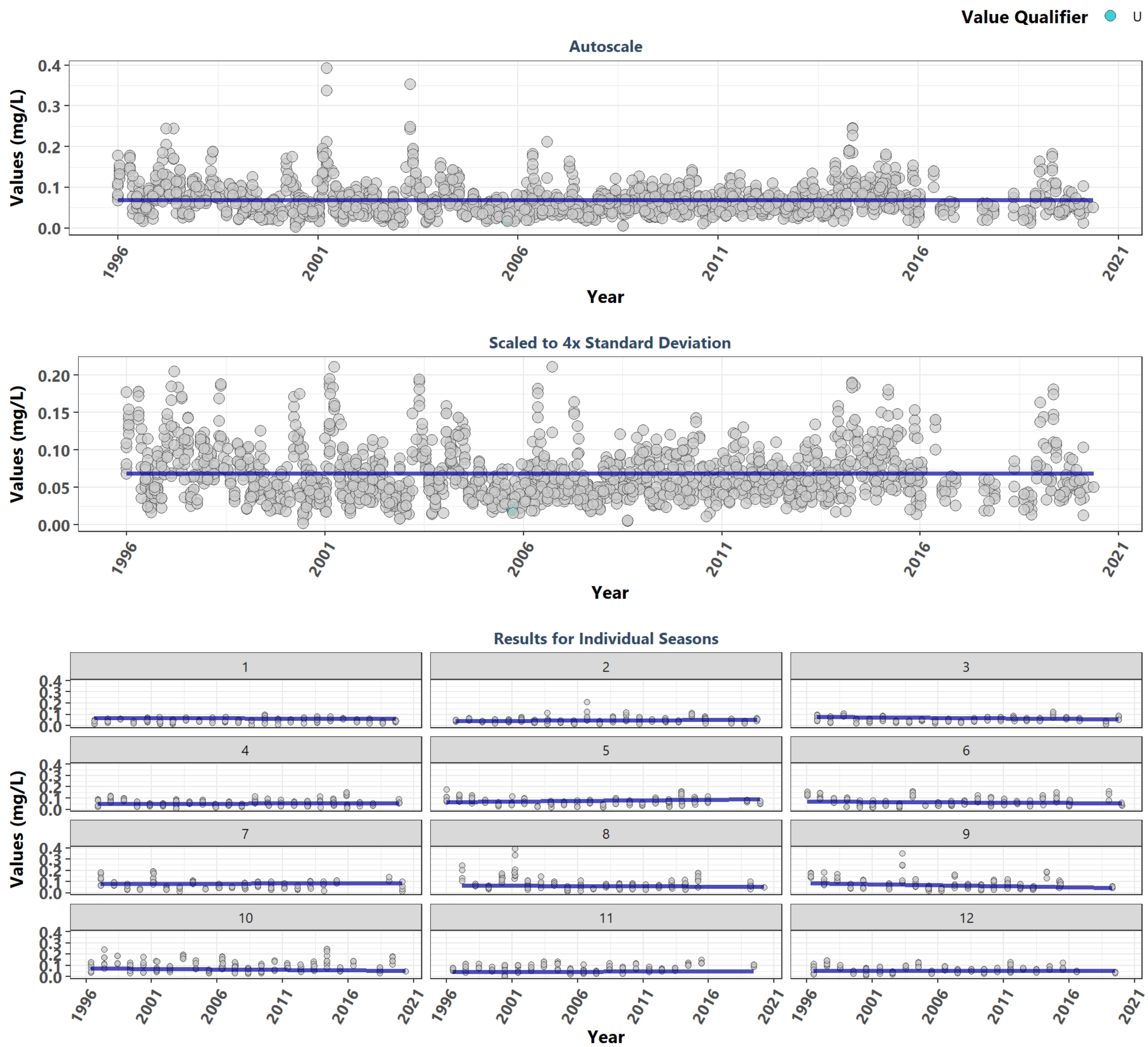
Indian River-Malabar to Vero Beach Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	5224	0.05	0.0044	0.0000	0.0557	0.8	0.4474	64.9	0	0
1	477	0.04	0.0461	0.0003	0.0385	2.0	0.0459	NA	NA	0
2	457	0.04	0.0611	0.0003	0.0367	2.9	0.0034	NA	NA	0
3	449	0.04	0.0087	0.0001	0.0534	1.5	0.1441	NA	NA	0
4	438	0.05	-0.0579	-0.0007	0.0797	2.9	0.0041	NA	NA	0
5	450	0.06	0.0917	0.0005	0.0323	1.3	0.1794	NA	NA	0
6	456	0.07	-0.0307	-0.0003	0.0715	-2.7	0.0072	NA	NA	0
7	399	0.07	0.0917	0.0006	0.0447	-1.7	0.0837	NA	NA	0
8	402	0.07	0.0423	0.0003	0.0579	-4.7	0.0000	NA	NA	0
9	396	0.07	-0.0841	-0.0007	0.0773	-1.7	0.0837	NA	NA	0
10	430	0.07	-0.1560	-0.0018	0.0918	-1.0	0.3409	NA	NA	0
11	427	0.05	0.0646	0.0004	0.0442	0.3	0.7875	NA	NA	0
12	443	0.05	-0.0581	-0.0007	0.0805	2.0	0.0419	NA	NA	0

[a] p < 0.00005 appear as 0 due to rounding

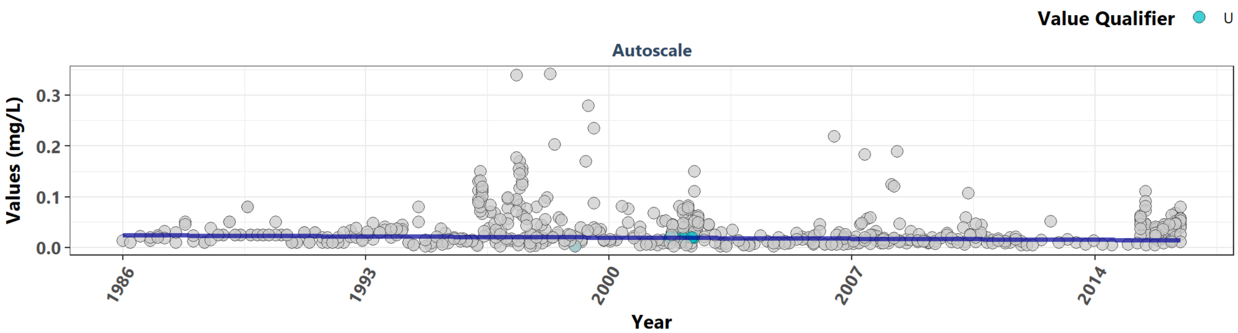
Indian River-Vero Beach to Ft. Pierce Aquatic Preserve

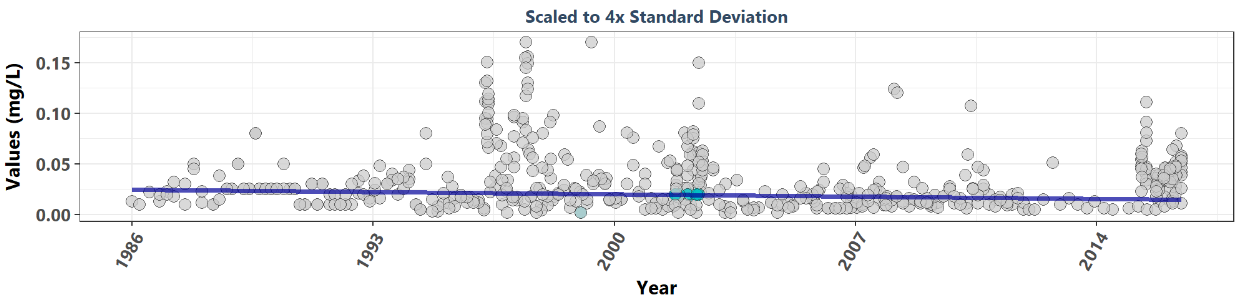


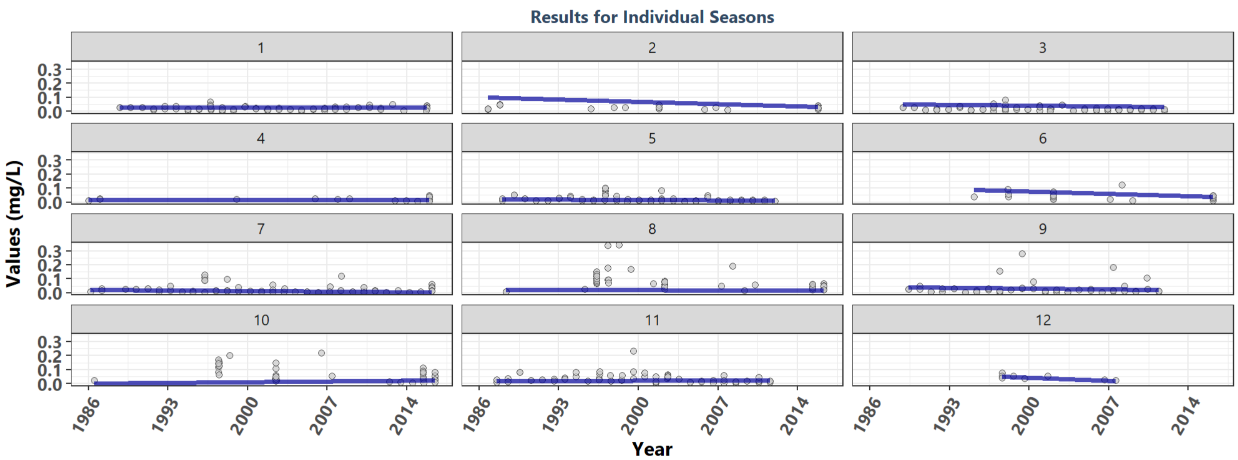
Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	1666	0.06	-0.0068	0.0000	0.0688	-0.2	0.8537	30	0.0016	0
1	158	0.04	-0.0466	-0.0003	0.0692	2.4	0.0164	NA	NA	0
2	150	0.04	0.1285	0.0005	0.0396	1.6	0.1155	NA	NA	0
3	131	0.05	-0.0820	-0.0008	0.0763	1.1	0.2730	NA	NA	0
4	167	0.05	0.0647	0.0003	0.0462	-0.2	0.8472	NA	NA	0
5	137	0.07	0.1238	0.0010	0.0646	-0.8	0.4197	NA	NA	0
6	147	0.06	-0.0675	-0.0005	0.0688	-1.0	0.3373	NA	NA	0
7	118	0.07	0.0167	0.0002	0.0832	-1.3	0.1879	NA	NA	0
8	138	0.07	-0.0534	-0.0005	0.0688	-2.0	0.0499	NA	NA	0
9	121	0.07	-0.1758	-0.0018	0.0859	-2.9	0.0042	NA	NA	0
10	152	0.09	-0.1126	-0.0010	0.0761	0.3	0.7606	NA	NA	0
11	122	0.07	0.0866	0.0003	0.0403	2.0	0.0429	NA	NA	0
12	125	0.06	-0.0101	-0.0001	0.0547	-1.1	0.2645	NA	NA	0

[a] $p < 0.00005$ appear as 0 due to rounding

Jensen Beach to Jupiter Inlet Aquatic Preserve



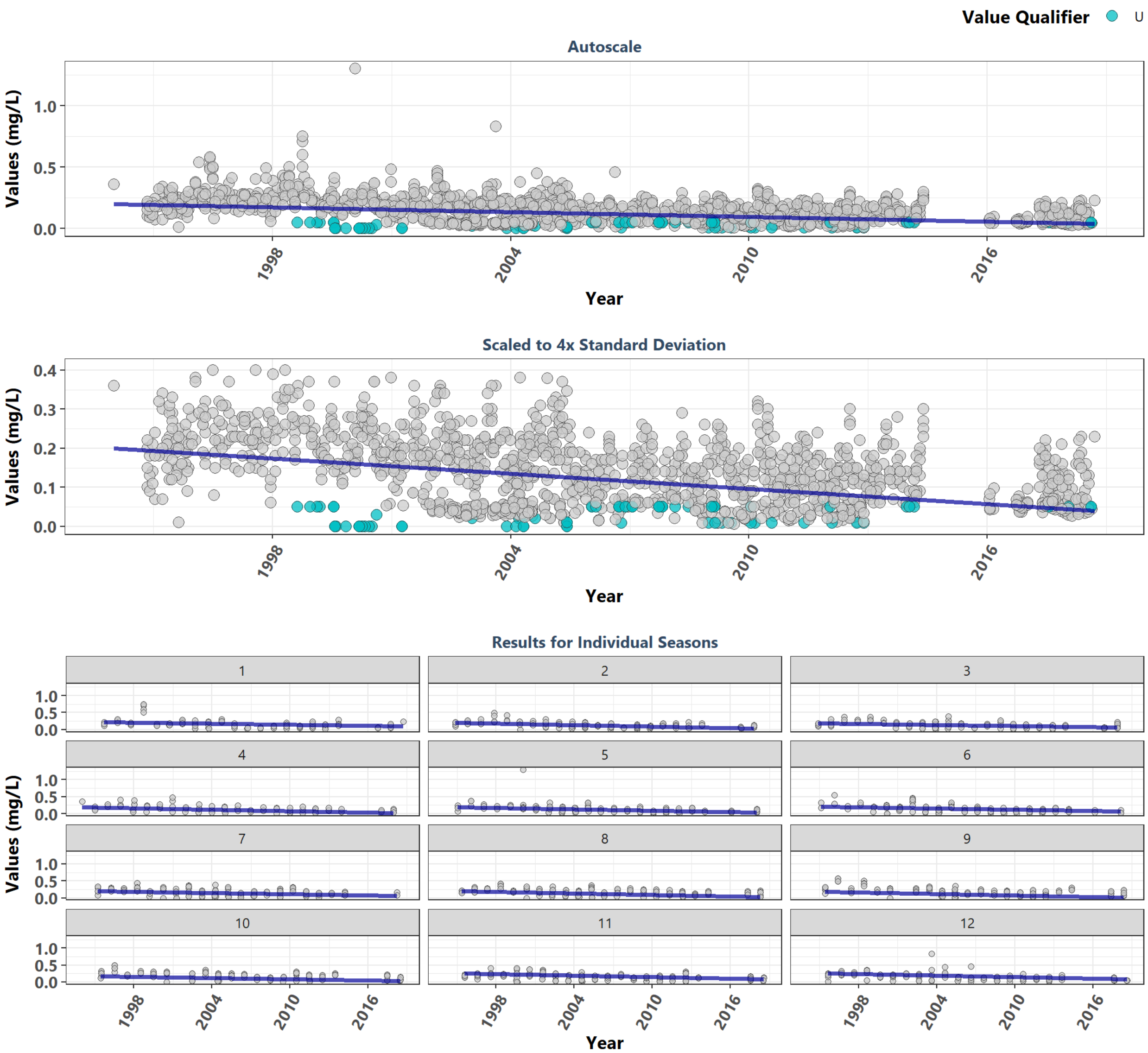




Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	552	0.03	-0.1253	-0.0003	0.0248	-3.9	0.0001	28.2	0.003	-1
1	70	0.02	-0.0123	0.0000	0.0265	1.0	0.3087	NA	NA	-1
2	26	0.03	-0.3462	-0.0024	0.1007	-0.1	0.9453	NA	NA	-1
3	60	0.01	-0.1429	-0.0006	0.0499	-1.7	0.0924	NA	NA	-1
4	19	0.02	-0.0589	-0.0001	0.0174	1.8	0.0681	NA	NA	-1
5	59	0.01	-0.1492	-0.0003	0.0198	-2.6	0.0105	NA	NA	-1
6	22	0.04	-0.2454	-0.0024	0.1089	-0.9	0.3437	NA	NA	-1
7	74	0.02	-0.2285	-0.0006	0.0226	-0.7	0.4592	NA	NA	-1
8	52	0.06	-0.0739	-0.0003	0.0231	-3.7	0.0002	NA	NA	-1
9	52	0.02	-0.1835	-0.0008	0.0412	-0.8	0.4427	NA	NA	-1
10	46	0.05	0.2924	0.0007	0.0014	-2.4	0.0144	NA	NA	1
11	64	0.03	0.0832	0.0002	0.0183	-2.1	0.0320	NA	NA	1
12	8	0.05	-0.6786	-0.0031	0.0871	-2.3	0.0219	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

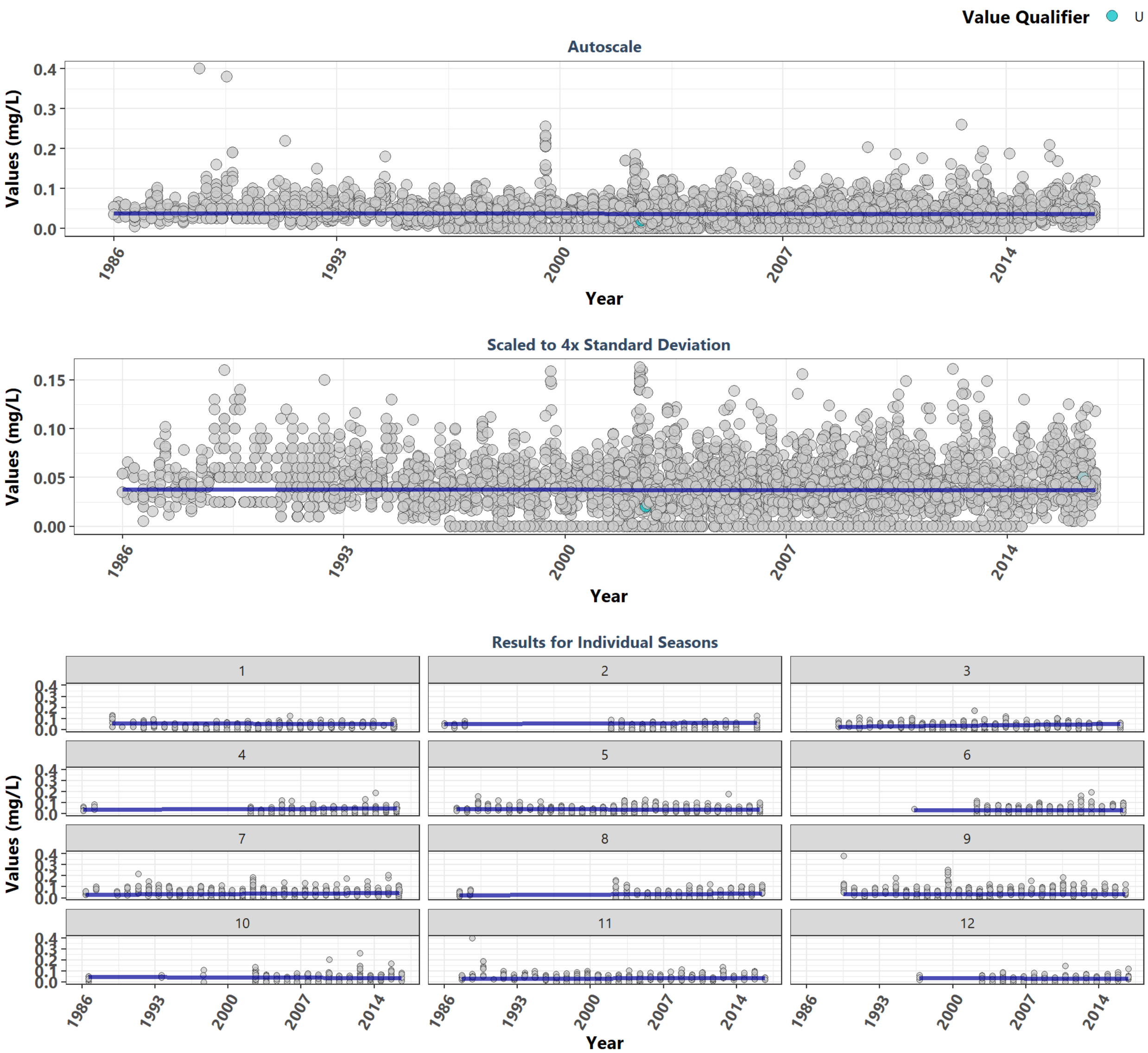
Lemon Bay Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	1543	0.14	-0.2986	-0.0065	0.2129	-17.2	0.0000	11.8	0.3789	-1
1	110	0.12	-0.2265	-0.0050	0.2350	-3.7	0.0002	NA	NA	-1
2	128	0.12	-0.3116	-0.0075	0.2250	-5.5	0.0000	NA	NA	-1
3	110	0.11	-0.2364	-0.0054	0.2062	-4.6	0.0000	NA	NA	-1
4	96	0.11	-0.3882	-0.0071	0.2100	-5.1	0.0000	NA	NA	-1
5	157	0.11	-0.3300	-0.0064	0.2090	-7.2	0.0000	NA	NA	-1
6	146	0.12	-0.1939	-0.0055	0.2155	-5.6	0.0000	NA	NA	-1
7	141	0.15	-0.2850	-0.0056	0.2103	-3.4	0.0006	NA	NA	-1
8	145	0.17	-0.3562	-0.0077	0.2288	-4.1	0.0001	NA	NA	-1
9	116	0.17	-0.3533	-0.0074	0.2096	-4.7	0.0000	NA	NA	-1
10	128	0.18	-0.2952	-0.0058	0.1898	-5.4	0.0000	NA	NA	-1
11	150	0.14	-0.2933	-0.0075	0.2750	-5.2	0.0000	NA	NA	-1
12	116	0.12	-0.3238	-0.0071	0.2814	-5.7	0.0000	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

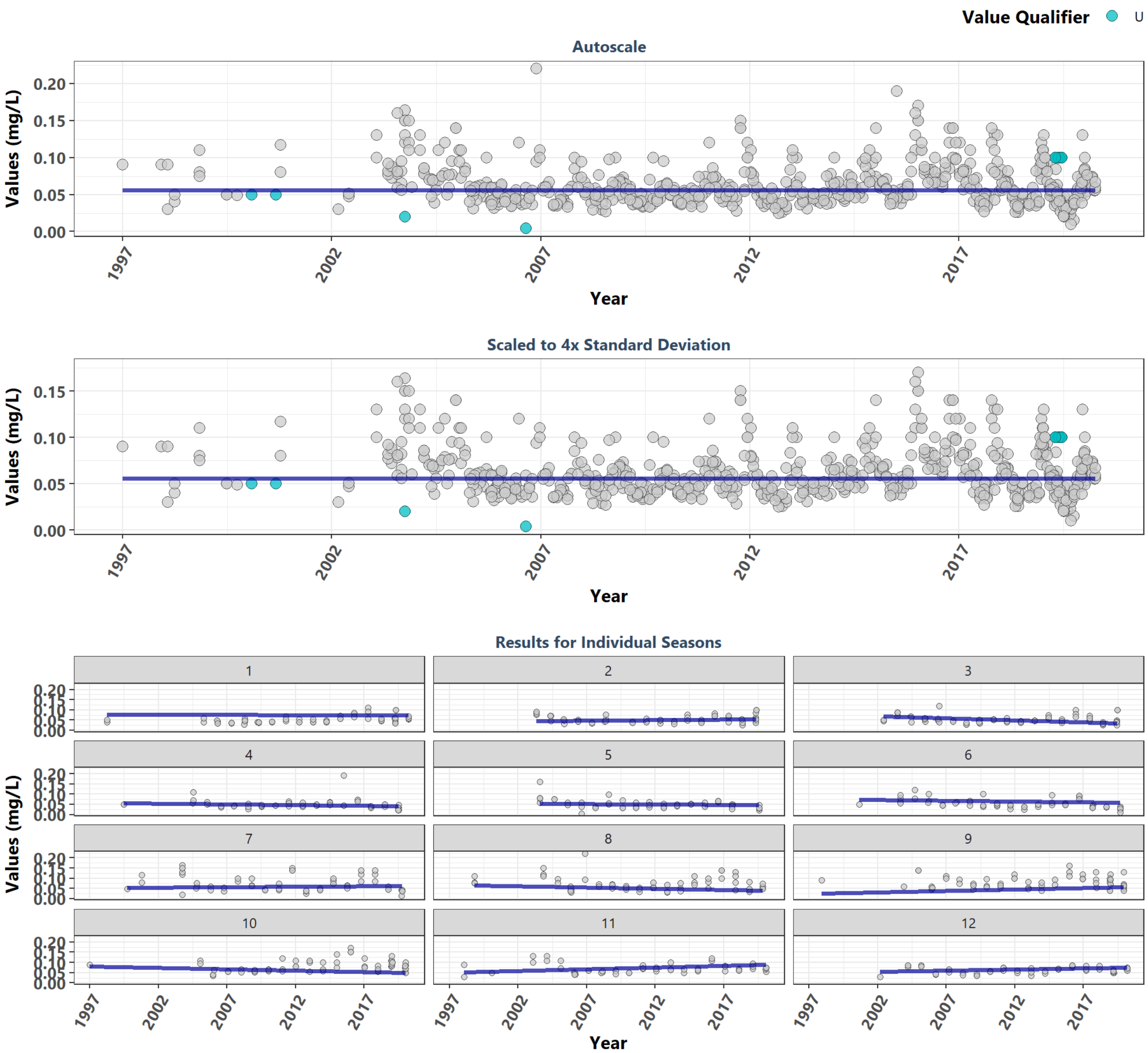
Loxahatchee River-Lake Worth Creek Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	3921	0.04	-0.0028	0.0000	0.0376	-1.4	0.1624	29.7	0.0018	0
1	479	0.03	-0.0467	-0.0002	0.0569	0.8	0.4254	NA	NA	0
2	157	0.03	0.0398	0.0003	0.0530	-0.9	0.3849	NA	NA	0
3	464	0.04	0.1058	0.0010	0.0225	-2.1	0.0400	NA	NA	0
4	189	0.03	0.0716	0.0003	0.0389	0.8	0.4215	NA	NA	0
5	480	0.04	-0.0348	-0.0001	0.0443	-1.1	0.2536	NA	NA	0
6	162	0.04	0.0243	0.0001	0.0299	2.0	0.0452	NA	NA	0
7	534	0.05	0.0958	0.0006	0.0302	-1.6	0.1060	NA	NA	0
8	159	0.05	0.0930	0.0006	0.0216	1.3	0.1801	NA	NA	0
9	471	0.06	-0.0467	-0.0001	0.0362	1.3	0.1962	NA	NA	0
10	211	0.04	-0.0707	-0.0003	0.0461	2.1	0.0380	NA	NA	0
11	466	0.04	0.0393	0.0001	0.0310	-2.3	0.0225	NA	NA	0
12	149	0.04	-0.0637	-0.0003	0.0393	1.7	0.0919	NA	NA	0

[a] $p < 0.00005$ appear as 0 due to rounding

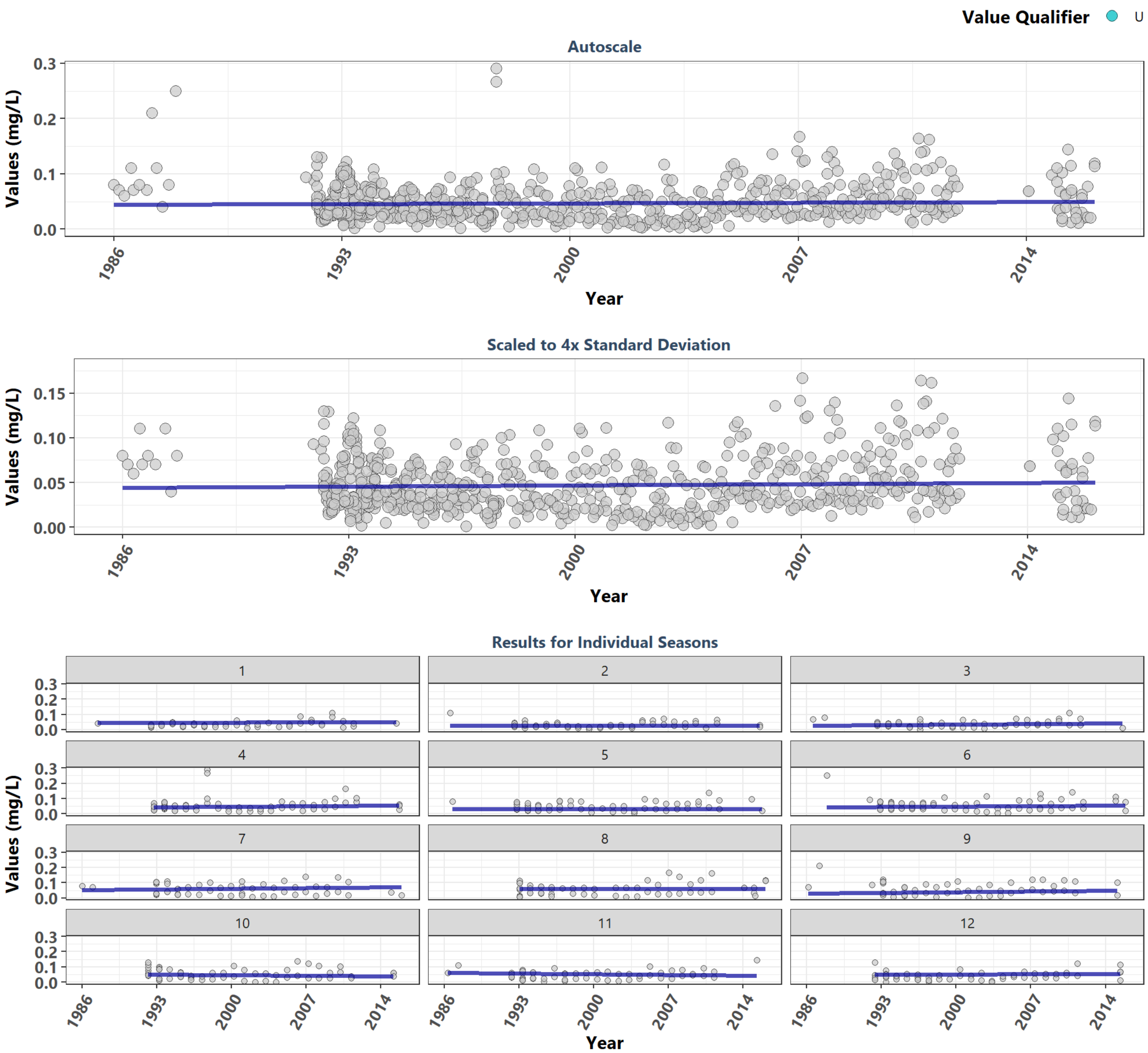
Matlacha Pass Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	701	0.06	-0.0129	0.0000	0.0558	-0.2	0.8637	73.2	0	0
1	60	0.05	-0.0236	-0.0001	0.0756	4.6	0.0000	NA	NA	0
2	60	0.05	0.1011	0.0005	0.0422	1.1	0.2530	NA	NA	0
3	68	0.05	-0.4377	-0.0020	0.0795	-0.9	0.3435	NA	NA	0
4	53	0.04	-0.2039	-0.0008	0.0576	-2.2	0.0310	NA	NA	0
5	49	0.05	-0.0786	-0.0004	0.0539	-2.4	0.0154	NA	NA	0
6	56	0.05	-0.0907	-0.0007	0.0756	-4.8	0.0000	NA	NA	0
7	57	0.06	0.0738	0.0004	0.0533	-1.9	0.0523	NA	NA	0
8	58	0.06	-0.2389	-0.0012	0.0670	-1.0	0.3161	NA	NA	0
9	58	0.07	0.4040	0.0014	0.0248	-0.3	0.7979	NA	NA	0
10	65	0.08	-0.1767	-0.0014	0.0808	2.6	0.0104	NA	NA	0
11	59	0.07	0.2154	0.0017	0.0494	2.2	0.0304	NA	NA	0
12	58	0.06	0.1935	0.0011	0.0505	0.8	0.4149	NA	NA	0

[a] $p < 0.00005$ appear as 0 due to rounding

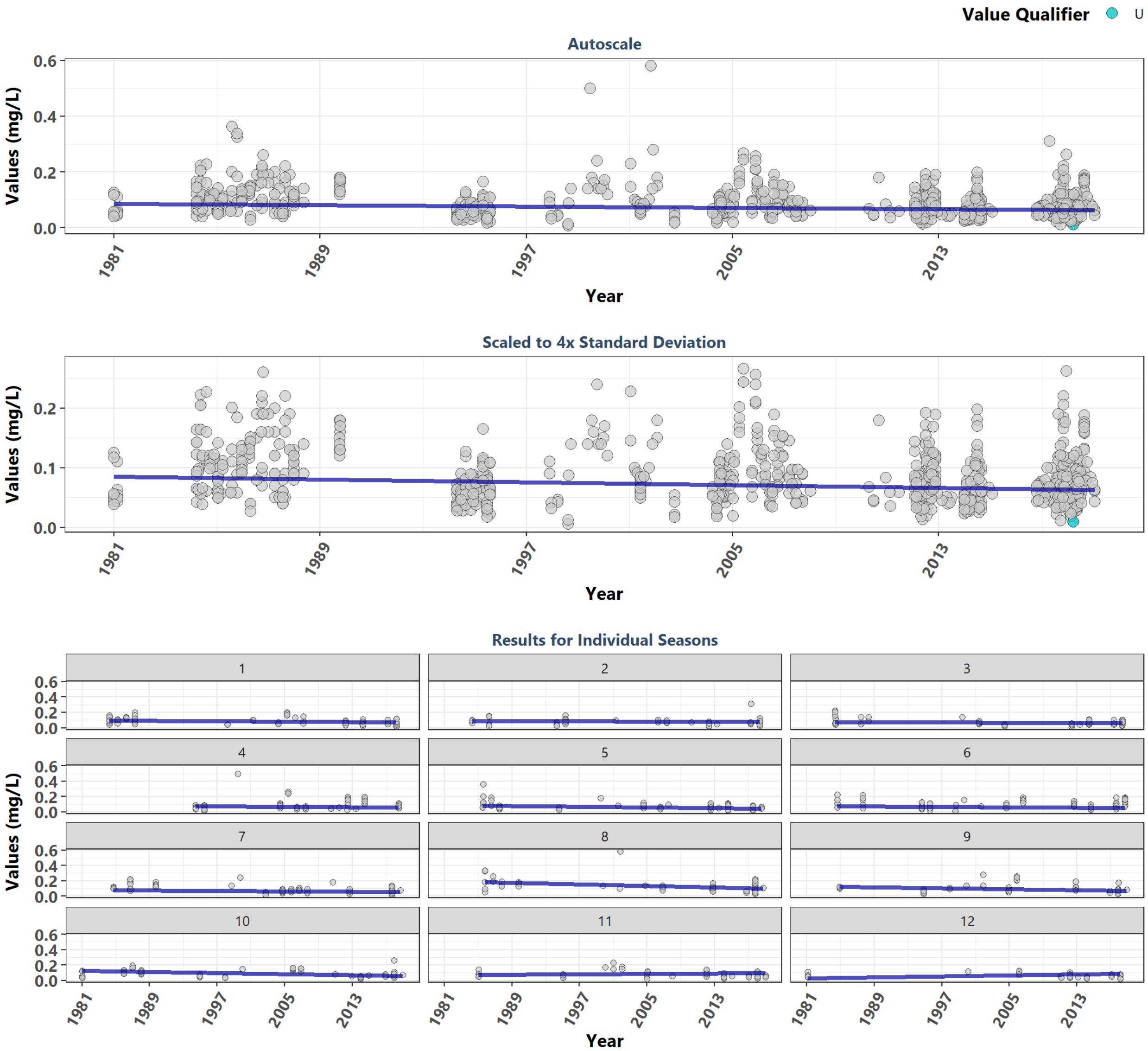
Mosquito Lagoon Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	698	0.04	0.0387	0.0002	0.0439	1.5	0.1340	7.9	0.7235	0
1	59	0.03	0.0224	0.0001	0.0462	1.8	0.0701	NA	NA	0
2	59	0.03	0.0076	0.0000	0.0279	0.1	0.9372	NA	NA	0
3	61	0.03	0.1619	0.0006	0.0261	0.0	0.9651	NA	NA	0
4	62	0.05	0.0952	0.0006	0.0394	1.1	0.2752	NA	NA	0
5	58	0.05	-0.0044	0.0000	0.0328	0.2	0.8085	NA	NA	0
6	64	0.06	0.0549	0.0004	0.0416	0.0	0.9768	NA	NA	0
7	54	0.05	0.1209	0.0007	0.0515	-0.9	0.3772	NA	NA	0
8	54	0.06	-0.0030	0.0000	0.0580	1.3	0.1979	NA	NA	0
9	50	0.05	0.1286	0.0007	0.0294	0.2	0.8142	NA	NA	0
10	59	0.05	-0.0643	-0.0005	0.0514	-0.7	0.4743	NA	NA	0
11	59	0.05	-0.0832	-0.0006	0.0610	0.6	0.5416	NA	NA	0
12	59	0.04	0.0237	0.0001	0.0501	1.4	0.1508	NA	NA	0

[a] $p < 0.00005$ appear as 0 due to rounding

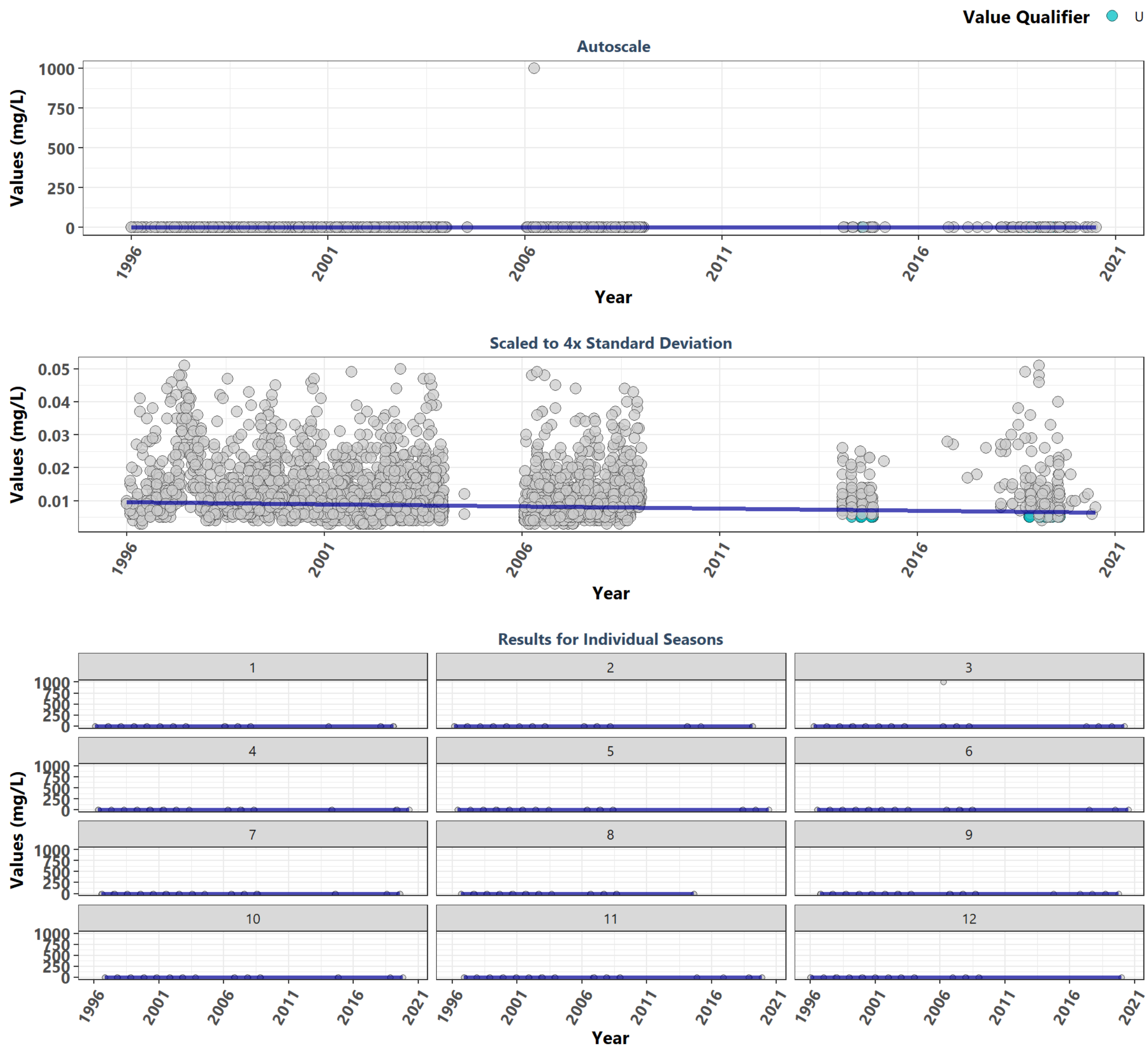
Nassau River-St. Johns River Marshes Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	764	0.07	-0.1210	-0.0006	0.0852	-4.5	0.0000	43.8	0	-1
1	72	0.08	-0.1317	-0.0007	0.0992	-4.2	0.0000	NA	NA	-1
2	66	0.07	-0.0544	-0.0003	0.0913	-0.7	0.4776	NA	NA	-1
3	72	0.06	-0.0597	-0.0003	0.0745	-1.9	0.0588	NA	NA	-1
4	72	0.07	-0.1495	-0.0007	0.0818	2.9	0.0039	NA	NA	-1
5	66	0.06	-0.2732	-0.0012	0.0887	-2.3	0.0211	NA	NA	-1
6	79	0.08	-0.1935	-0.0007	0.0798	1.3	0.2071	NA	NA	-1
7	62	0.09	-0.1407	-0.0005	0.0777	-3.2	0.0013	NA	NA	-1
8	54	0.11	-0.2956	-0.0024	0.1928	-3.2	0.0013	NA	NA	-1
9	49	0.08	-0.2787	-0.0016	0.1290	-0.6	0.5821	NA	NA	-1
10	69	0.09	-0.3372	-0.0016	0.1240	-1.6	0.1080	NA	NA	-1
11	62	0.07	0.0964	0.0006	0.0687	-1.6	0.1044	NA	NA	1
12	41	0.05	0.2308	0.0016	0.0270	-2.5	0.0110	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

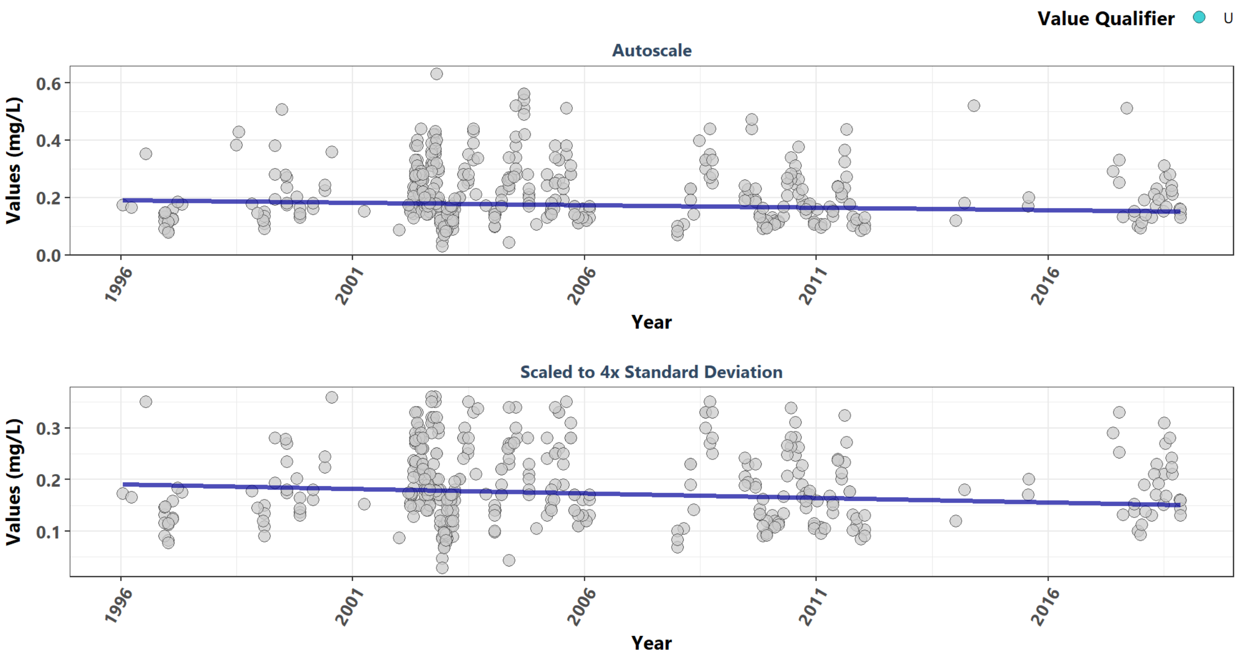
Nature Coast Aquatic Preserve

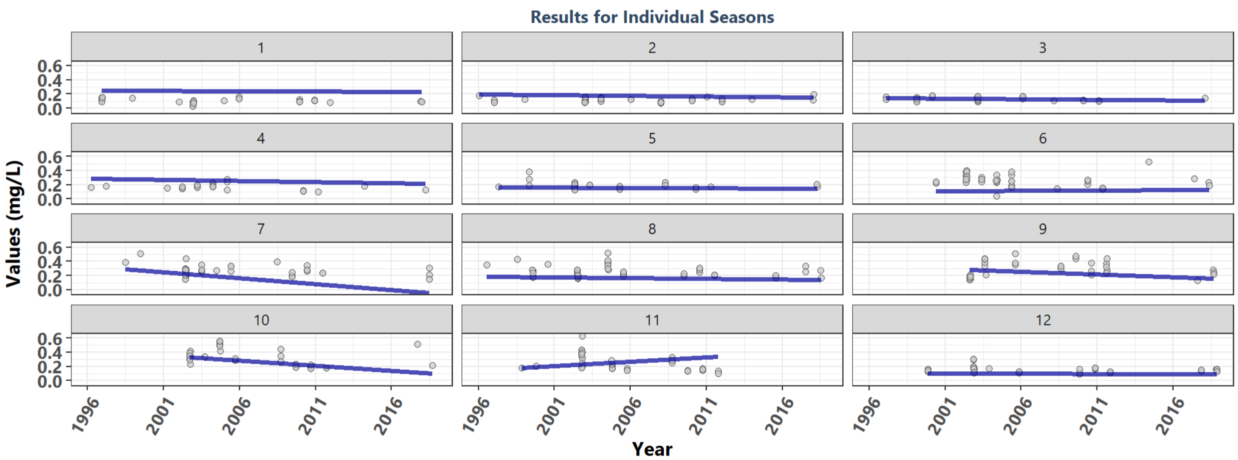


Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	4257	0.01	-0.1024	-0.0001	0.0096	-10.1	0.0000	15.2	0.1749	-1
1	390	0.01	-0.1214	-0.0001	0.0111	-3.0	0.0024	NA	NA	-1
2	338	0.01	-0.0844	-0.0001	0.0076	-2.8	0.0052	NA	NA	-1
3	335	0.01	-0.0285	0.0000	0.0080	-2.3	0.0198	NA	NA	-1
4	381	0.01	-0.1007	-0.0001	0.0077	-3.6	0.0003	NA	NA	-1
5	334	0.01	-0.1567	-0.0003	0.0133	-3.6	0.0003	NA	NA	-1
6	338	0.01	-0.1383	-0.0003	0.0120	-3.8	0.0001	NA	NA	-1
7	409	0.01	-0.1231	-0.0001	0.0090	-3.7	0.0002	NA	NA	-1
8	337	0.01	-0.1167	-0.0001	0.0109	-4.3	0.0000	NA	NA	-1
9	333	0.01	-0.0932	-0.0002	0.0124	-2.6	0.0106	NA	NA	-1
10	390	0.01	-0.1022	-0.0001	0.0085	-3.5	0.0005	NA	NA	-1
11	341	0.01	-0.1326	-0.0002	0.0102	-0.6	0.5520	NA	NA	-1
12	331	0.01	-0.0214	0.0000	0.0080	-0.8	0.4352	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

North Fork St. Lucie Aquatic Preserve

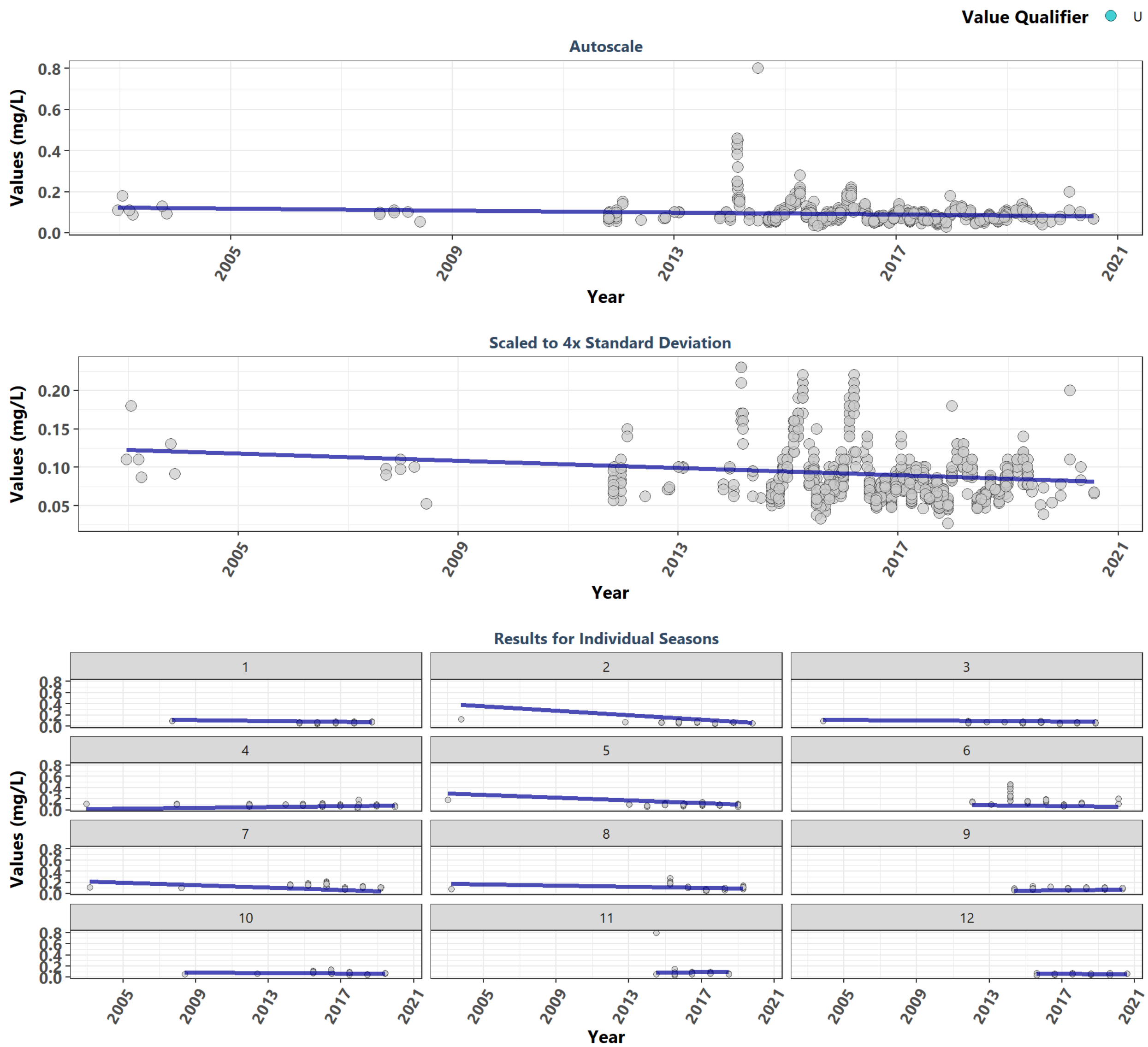




Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	428	0.18	-0.1110	-0.0018	0.1912	-3.2	0.0016	38.1	0.0001	-1
1	30	0.10	-0.0546	-0.0008	0.2462	-0.5	0.6341	NA	NA	-1
2	39	0.11	-0.1552	-0.0020	0.1980	0.9	0.3797	NA	NA	-1
3	37	0.13	-0.1486	-0.0013	0.1389	-1.4	0.1704	NA	NA	-1
4	25	0.17	-0.1092	-0.0030	0.2842	-0.7	0.5078	NA	NA	-1
5	29	0.18	-0.1408	-0.0010	0.1637	-1.2	0.2340	NA	NA	-1
6	46	0.27	0.0972	0.0009	0.1049	-2.7	0.0074	NA	NA	1
7	35	0.26	-0.4730	-0.0167	0.3343	-0.9	0.3482	NA	NA	-1
8	42	0.24	-0.0967	-0.0018	0.1844	-0.5	0.6122	NA	NA	-1
9	42	0.23	-0.2676	-0.0075	0.3270	3.2	0.0014	NA	NA	-1
10	33	0.31	-0.3239	-0.0141	0.4231	-2.7	0.0071	NA	NA	-1
11	36	0.20	0.3310	0.0126	0.1401	-4.2	0.0000	NA	NA	1
12	34	0.16	-0.0621	-0.0005	0.1033	-1.2	0.2335	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

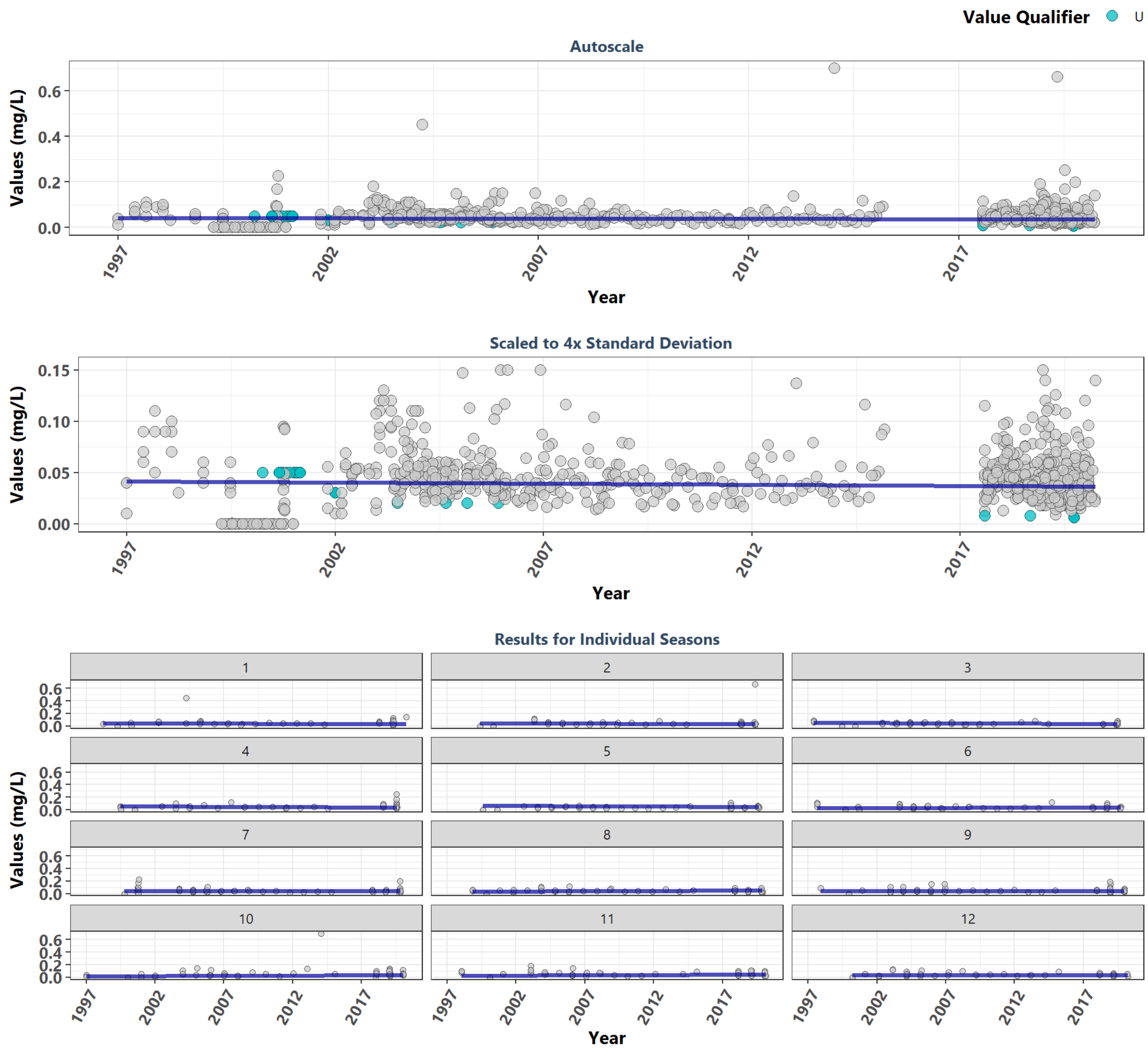
Pellicer Creek Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	748	0.08	-0.1456	-0.0023	0.1273	-6.7	0.0000	94.1	0	-1
1	66	0.07	-0.3041	-0.0030	0.1320	3.4	0.0006	NA	NA	-1
2	61	0.07	-0.3955	-0.0200	0.4400	-2.8	0.0053	NA	NA	-2
3	76	0.07	-0.1696	-0.0020	0.1250	-3.7	0.0002	NA	NA	-1
4	88	0.08	0.2844	0.0037	0.0083	-4.3	0.0000	NA	NA	1
5	67	0.10	-0.2673	-0.0125	0.3200	-2.1	0.0374	NA	NA	-2
6	72	0.14	-0.2393	-0.0037	0.1297	-5.0	0.0000	NA	NA	-1
7	68	0.12	-0.3695	-0.0110	0.2445	-3.3	0.0010	NA	NA	-1
8	54	0.10	-0.0622	-0.0053	0.1840	-0.7	0.4980	NA	NA	-1
9	55	0.09	0.3374	0.0038	-0.0008	1.5	0.1369	NA	NA	1
10	52	0.08	-0.2842	-0.0018	0.0948	-4.0	0.0001	NA	NA	-1
11	44	0.06	0.1347	0.0010	0.0750	-1.5	0.1345	NA	NA	1
12	45	0.06	-0.1522	-0.0027	0.1067	3.4	0.0007	NA	NA	-1

[a] p < 0.00005 appear as 0 due to rounding

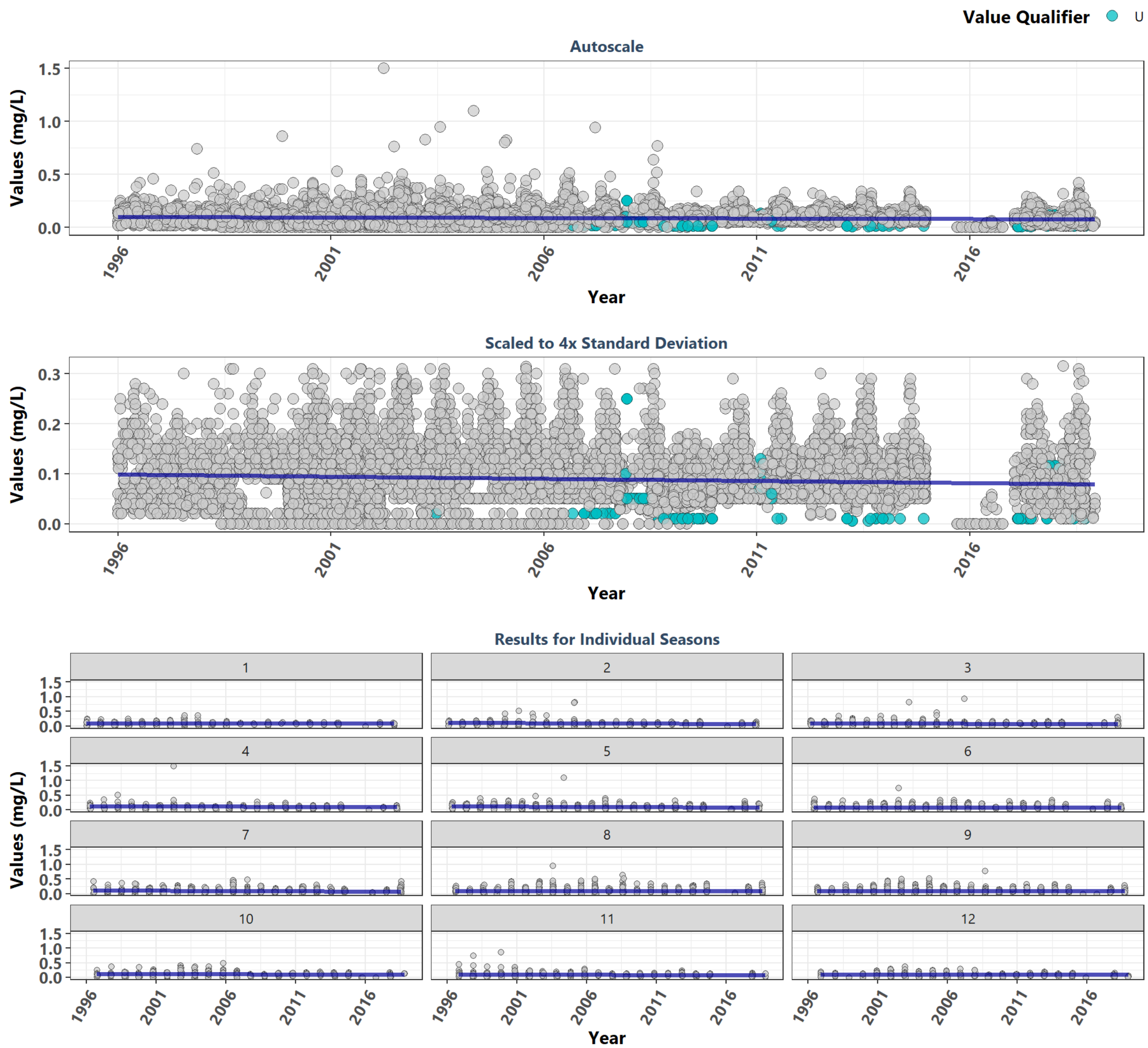
Pine Island Sound Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	1002	0.04	-0.0396	-0.0002	0.0414	-2.2	0.0256	41.3	0	-1
1	67	0.04	-0.0716	-0.0003	0.0445	1.3	0.1845	NA	NA	-1
2	64	0.03	-0.0813	-0.0005	0.0413	-1.0	0.3360	NA	NA	-1
3	75	0.04	-0.2119	-0.0008	0.0495	-1.0	0.2948	NA	NA	-1
4	72	0.04	-0.2723	-0.0011	0.0582	1.7	0.0980	NA	NA	-1
5	67	0.04	-0.1045	-0.0007	0.0658	2.3	0.0195	NA	NA	-1
6	94	0.03	0.1299	0.0006	0.0266	-3.1	0.0021	NA	NA	1
7	99	0.04	-0.0623	-0.0003	0.0454	-0.9	0.3561	NA	NA	-1
8	91	0.04	0.1175	0.0007	0.0310	-3.9	0.0001	NA	NA	1
9	98	0.04	-0.0303	-0.0001	0.0415	-0.4	0.6544	NA	NA	-1
10	97	0.05	0.1940	0.0008	0.0214	1.7	0.0824	NA	NA	1
11	87	0.05	0.1076	0.0006	0.0298	-1.4	0.1481	NA	NA	1
12	91	0.04	-0.0807	-0.0003	0.0383	-1.0	0.3042	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

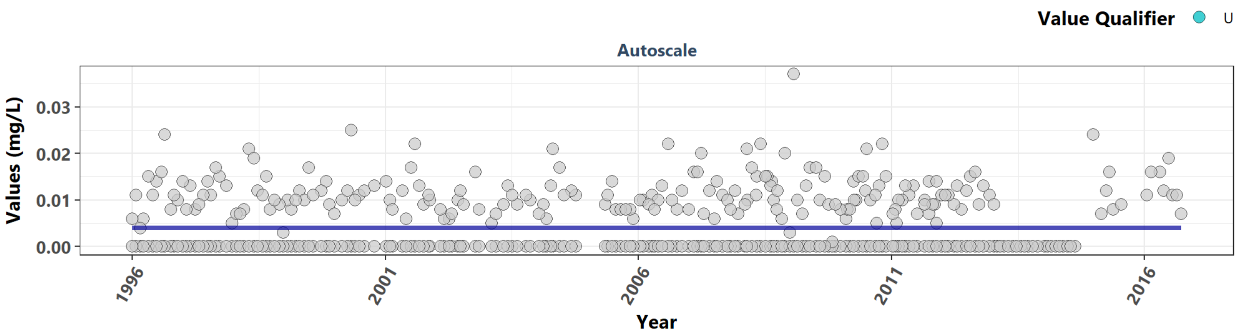
Pinellas County Aquatic Preserve

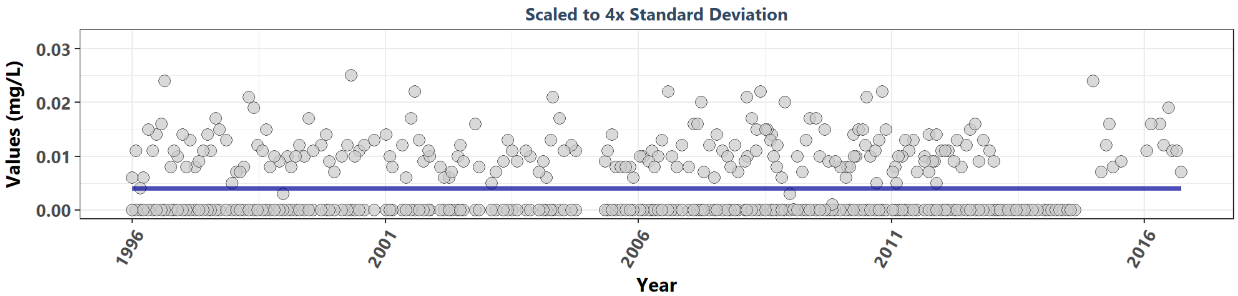


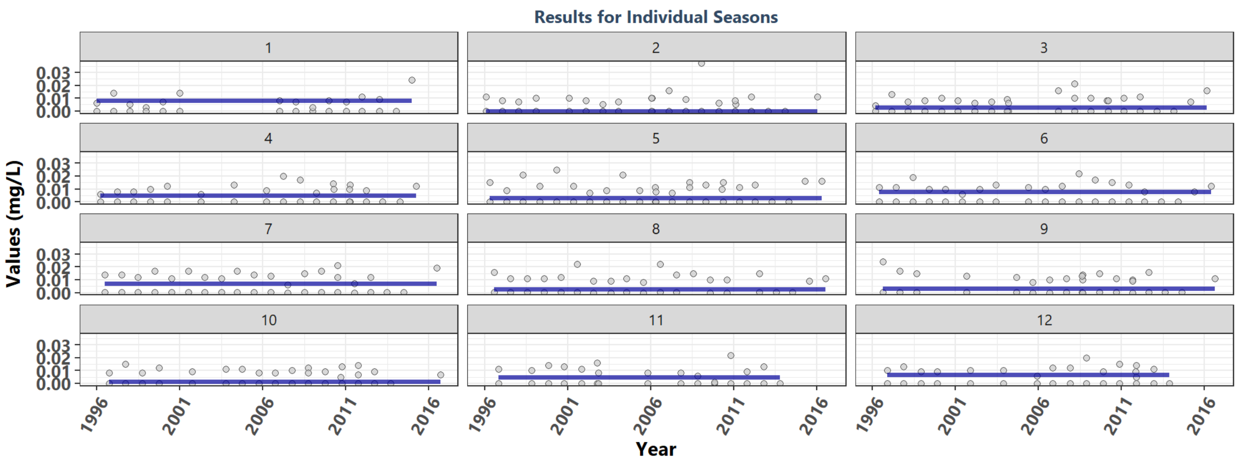
Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	10133	0.09	-0.0573	-0.0008	0.0988	-9.2	0.0000	55	0	-1
1	654	0.09	-0.0201	0.0000	0.0900	-0.6	0.5236	NA	NA	-1
2	780	0.08	-0.1179	-0.0020	0.1100	-1.6	0.1182	NA	NA	-1
3	761	0.09	-0.0568	-0.0005	0.0849	-3.0	0.0026	NA	NA	-1
4	658	0.08	-0.0793	-0.0014	0.1257	-2.2	0.0286	NA	NA	-1
5	1161	0.10	-0.1142	-0.0017	0.1183	-5.8	0.0000	NA	NA	-1
6	934	0.10	-0.0166	0.0000	0.0880	-2.4	0.0153	NA	NA	-1
7	803	0.09	-0.0929	-0.0013	0.1008	2.0	0.0472	NA	NA	-1
8	841	0.10	-0.0373	0.0000	0.0800	-0.6	0.5410	NA	NA	-1
9	1164	0.11	0.0467	0.0006	0.0844	-4.1	0.0000	NA	NA	1
10	801	0.09	-0.0529	-0.0007	0.1094	-5.0	0.0000	NA	NA	-1
11	571	0.09	-0.0729	-0.0008	0.0976	-0.7	0.4712	NA	NA	-1
12	1005	0.09	-0.0141	0.0000	0.1000	-4.4	0.0000	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

Rocky Bayou State Park Aquatic Preserve



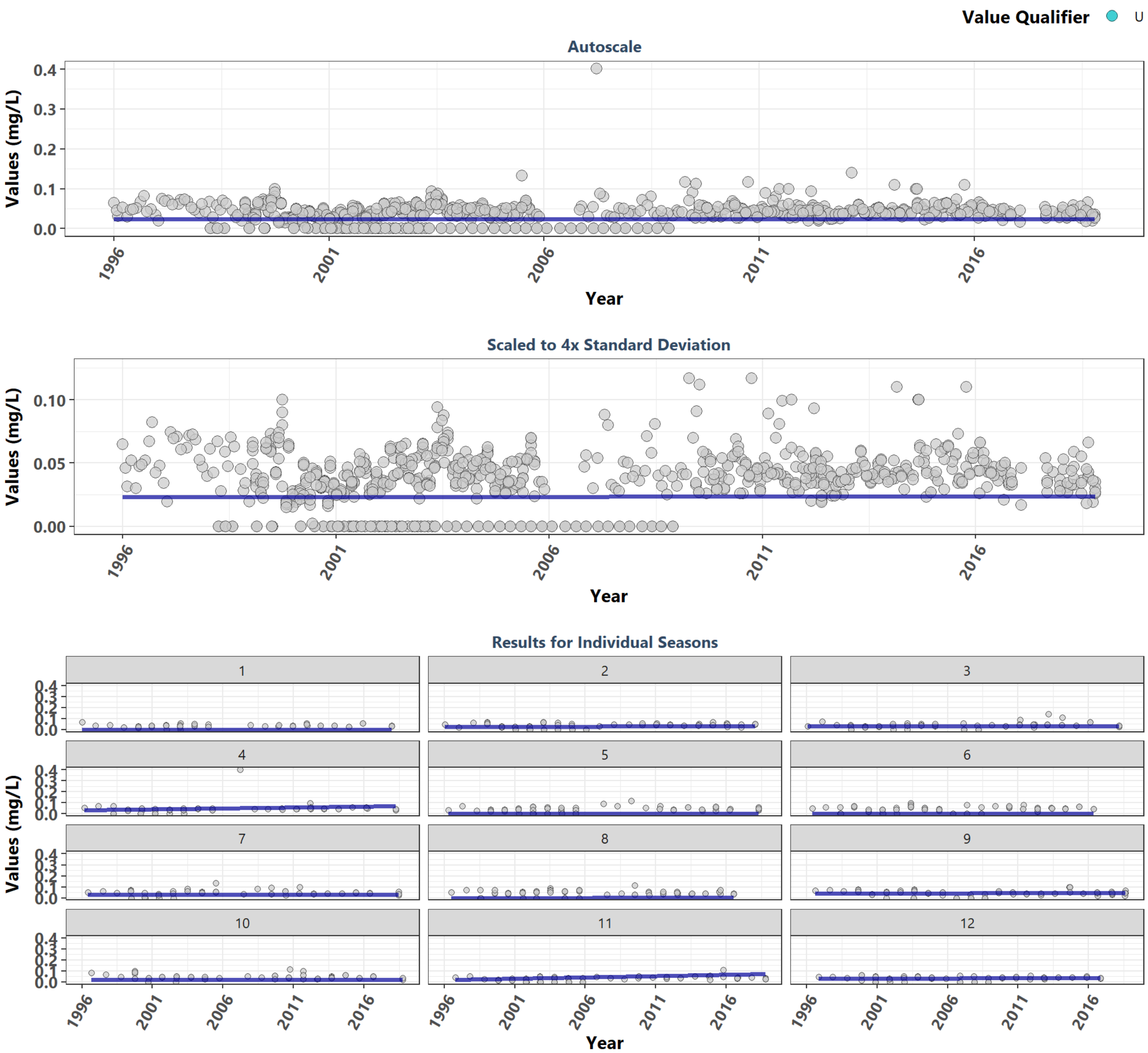




Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	415	0.00	0.0453	0	0.0040	1.3	0.1777	4.8	0.9402	0
1	28	0.00	-0.0985	0	0.0080	1.0	0.3405	NA	NA	0
2	36	0.00	-0.0739	0	0.0000	0.4	0.7115	NA	NA	0
3	38	0.00	-0.0085	0	0.0030	1.4	0.1570	NA	NA	0
4	34	0.00	0.1607	0	0.0050	1.2	0.2277	NA	NA	0
5	42	0.01	0.0635	0	0.0030	0.4	0.6715	NA	NA	0
6	36	0.00	0.0101	0	0.0080	0.5	0.5937	NA	NA	0
7	38	0.00	0.0465	0	0.0070	-0.1	0.9496	NA	NA	0
8	35	0.01	0.0444	0	0.0025	0.1	0.9431	NA	NA	0
9	33	0.01	0.1462	0	0.0030	-0.8	0.4268	NA	NA	0
10	35	0.01	0.1296	0	0.0015	0.5	0.6475	NA	NA	0
11	29	0.00	0.0538	0	0.0050	-0.5	0.5844	NA	NA	0
12	31	0.00	0.0555	0	0.0070	0.4	0.6815	NA	NA	0

[a] p < 0.00005 appear as 0 due to rounding

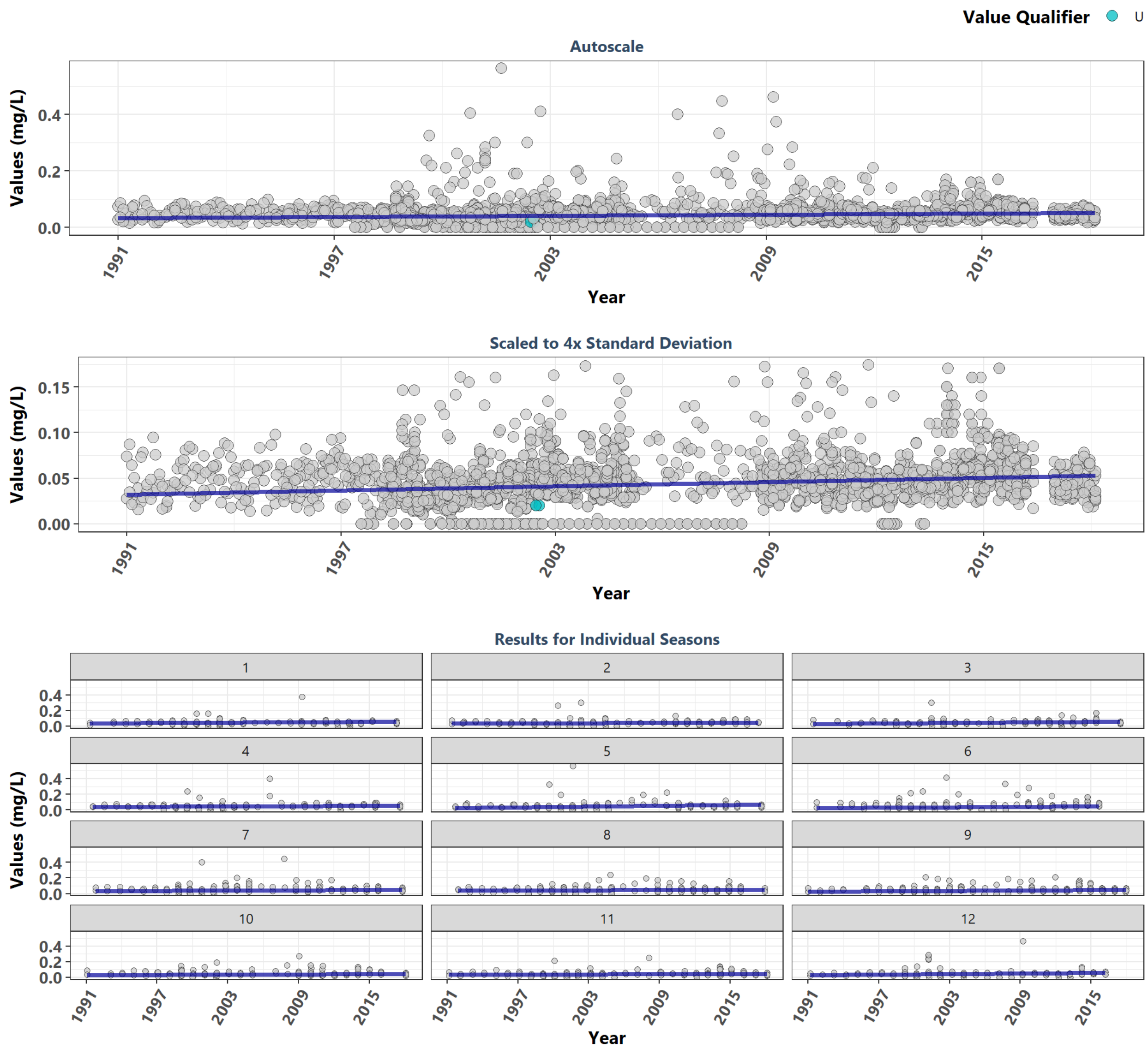
Rookery Bay Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	1235	0.03	0.1202	0.0000	0.0234	5.8	0.0000	28.9	0.0023	1
1	54	0.03	0.0270	0.0000	0.0000	1.2	0.2458	NA	NA	1
2	141	0.00	0.1572	0.0003	0.0257	3.0	0.0023	NA	NA	2
3	97	0.03	0.0173	0.0000	0.0290	2.3	0.0220	NA	NA	1
4	90	0.03	0.2112	0.0018	0.0293	6.1	0.0000	NA	NA	1
5	106	0.03	0.1127	0.0000	0.0000	1.3	0.2025	NA	NA	1
6	126	0.00	0.1722	0.0000	0.0000	0.4	0.6530	NA	NA	1
7	77	0.04	0.0834	0.0000	0.0289	2.7	0.0063	NA	NA	1
8	128	0.00	0.0785	0.0000	0.0000	1.9	0.0576	NA	NA	2
9	122	0.03	0.0584	0.0003	0.0377	0.3	0.7776	NA	NA	1
10	63	0.04	0.0483	0.0000	0.0211	0.7	0.5007	NA	NA	1
11	125	0.00	0.4370	0.0025	0.0198	1.3	0.1919	NA	NA	2
12	106	0.02	0.1090	0.0003	0.0309	0.7	0.4612	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

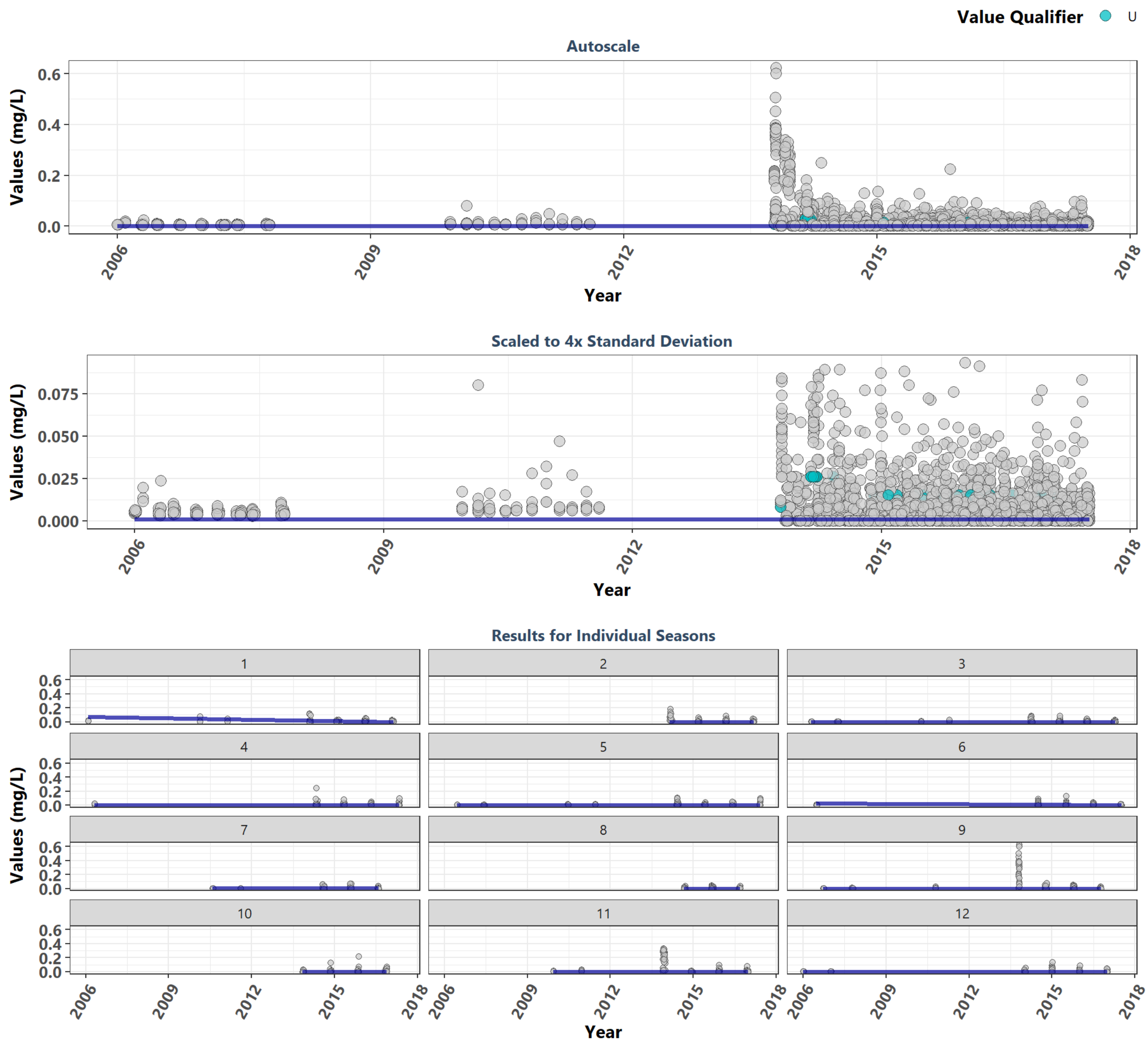
Rookery Bay National Estuarine Research Reserve



Season	N	Median	tau	Slope	Int.	z	p__z	chi__sq	p__chi__sq	Trend
All	3116	0.04	0.1394	0.0008	0.0323	11.8	0.0000	38.5	0.0001	1
1	208	0.04	0.1176	0.0007	0.0376	2.1	0.0385	NA	NA	1
2	295	0.03	0.0992	0.0004	0.0327	5.5	0.0000	NA	NA	1
3	249	0.04	0.1617	0.0013	0.0237	2.8	0.0047	NA	NA	1
4	248	0.04	0.1005	0.0005	0.0385	7.5	0.0000	NA	NA	1
5	265	0.04	0.3210	0.0017	0.0213	2.4	0.0145	NA	NA	1
6	280	0.04	0.2159	0.0011	0.0200	3.7	0.0002	NA	NA	1
7	234	0.05	0.0963	0.0004	0.0349	2.7	0.0073	NA	NA	1
8	297	0.04	0.0327	0.0002	0.0428	4.2	0.0000	NA	NA	1
9	287	0.04	0.1661	0.0009	0.0224	1.5	0.1267	NA	NA	1
10	208	0.04	0.1200	0.0005	0.0320	0.7	0.4827	NA	NA	1
11	292	0.03	0.0604	0.0003	0.0380	4.2	0.0000	NA	NA	1
12	253	0.04	0.1473	0.0012	0.0293	2.4	0.0186	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

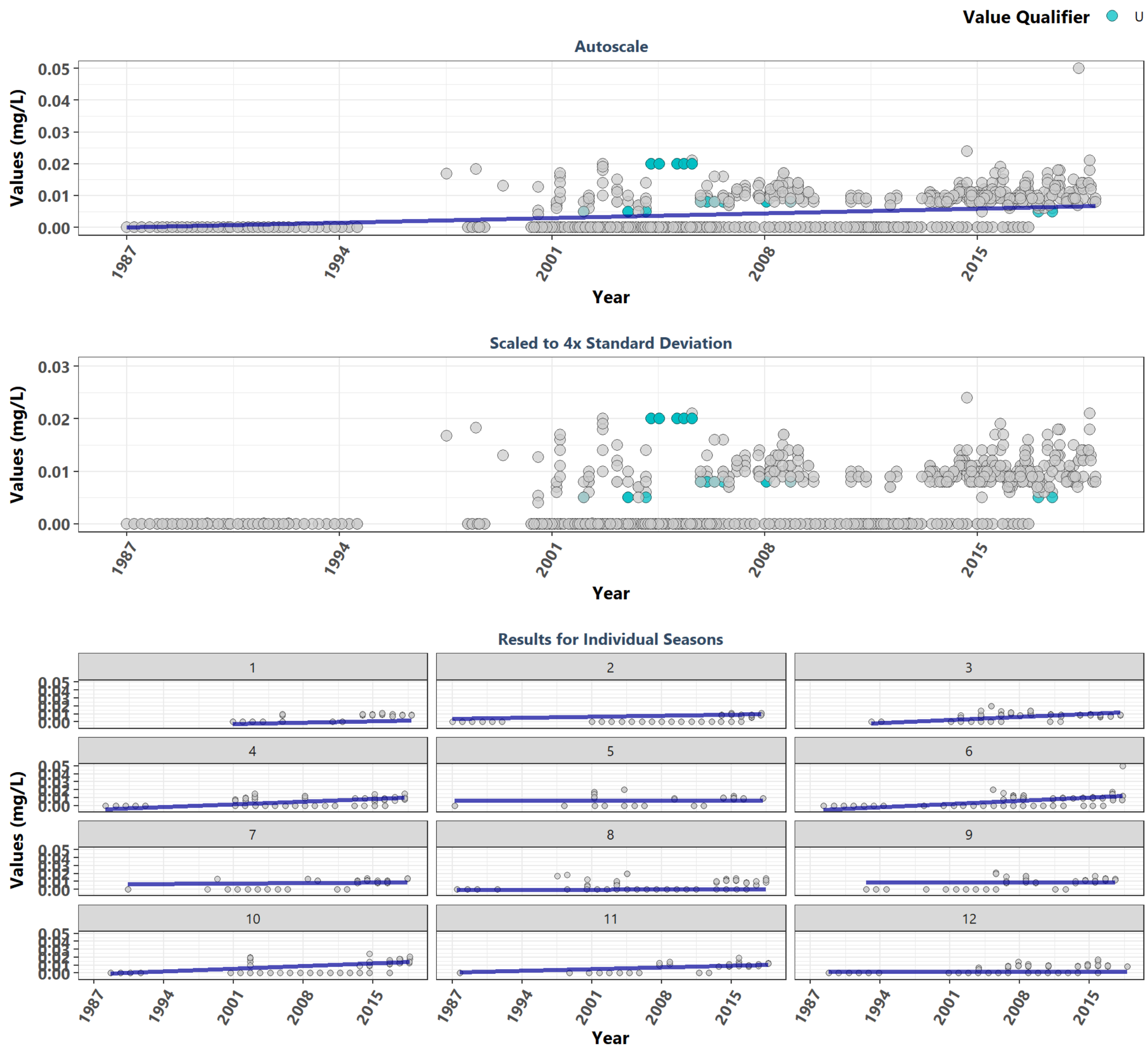
Southeast Florida Coral Reef Ecosystem Conservation Area



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	4801	0	-0.0583	0.0000	0.0010	-7.1	0.0000	221	0	0
1	436	0	-0.3215	-0.0067	0.0753	2.5	0.0120	NA	NA	-2
2	414	0	0.0817	0.0000	0.0000	-10.6	0.0000	NA	NA	-1
3	460	0	0.0714	0.0000	0.0010	-3.8	0.0002	NA	NA	-1
4	337	0	-0.1075	0.0000	0.0010	2.1	0.0317	NA	NA	-1
5	462	0	0.0334	0.0000	0.0000	-1.8	0.0676	NA	NA	0
6	424	0	-0.1795	-0.0020	0.0240	1.8	0.0701	NA	NA	-2
7	319	0	-0.2461	-0.0006	0.0080	-2.3	0.0216	NA	NA	-2
8	300	0	-0.0511	0.0000	0.0000	2.7	0.0062	NA	NA	0
9	426	0	0.0724	0.0000	0.0000	-8.1	0.0000	NA	NA	-1
10	395	0	0.0537	0.0000	0.0010	1.2	0.2189	NA	NA	0
11	406	0	-0.0773	0.0000	0.0010	-5.7	0.0000	NA	NA	-1
12	422	0	0.0296	0.0000	0.0000	1.0	0.3179	NA	NA	0

[a] $p < 0.00005$ appear as 0 due to rounding

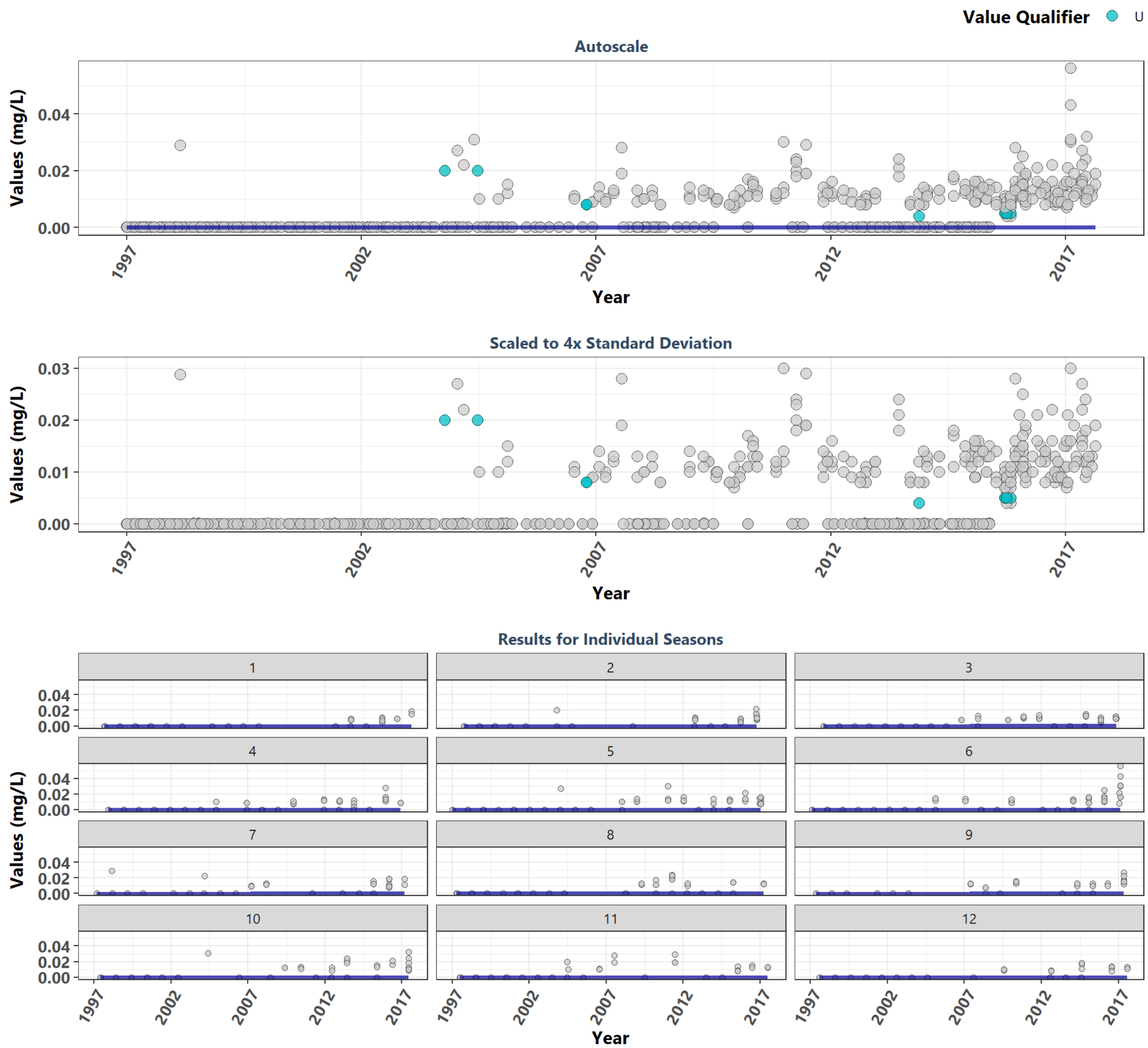
St. Andrews State Park Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	804	0.01	0.3210	0.0002	0.0000	12.6	0.0000	17.6	0.0915	1
1	41	0.01	0.4864	0.0003	-0.0062	3.3	0.0009	NA	NA	1
2	69	0.00	0.2779	0.0002	0.0038	5.9	0.0000	NA	NA	2
3	82	0.01	0.4657	0.0006	-0.0058	1.5	0.1225	NA	NA	1
4	79	0.01	0.3537	0.0005	-0.0055	3.1	0.0020	NA	NA	1
5	49	0.01	0.2356	0.0000	0.0060	2.0	0.0422	NA	NA	1
6	103	0.01	0.4220	0.0006	-0.0062	4.2	0.0000	NA	NA	1
7	48	0.01	0.1153	0.0001	0.0063	4.3	0.0000	NA	NA	1
8	80	0.00	0.3957	0.0000	-0.0009	3.5	0.0005	NA	NA	1
9	61	0.01	0.1981	0.0000	0.0090	4.4	0.0000	NA	NA	1
10	65	0.00	0.3820	0.0005	-0.0015	4.7	0.0000	NA	NA	2
11	45	0.01	0.3908	0.0003	0.0008	4.5	0.0000	NA	NA	1
12	82	0.01	0.2665	0.0000	0.0020	5.2	0.0000	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

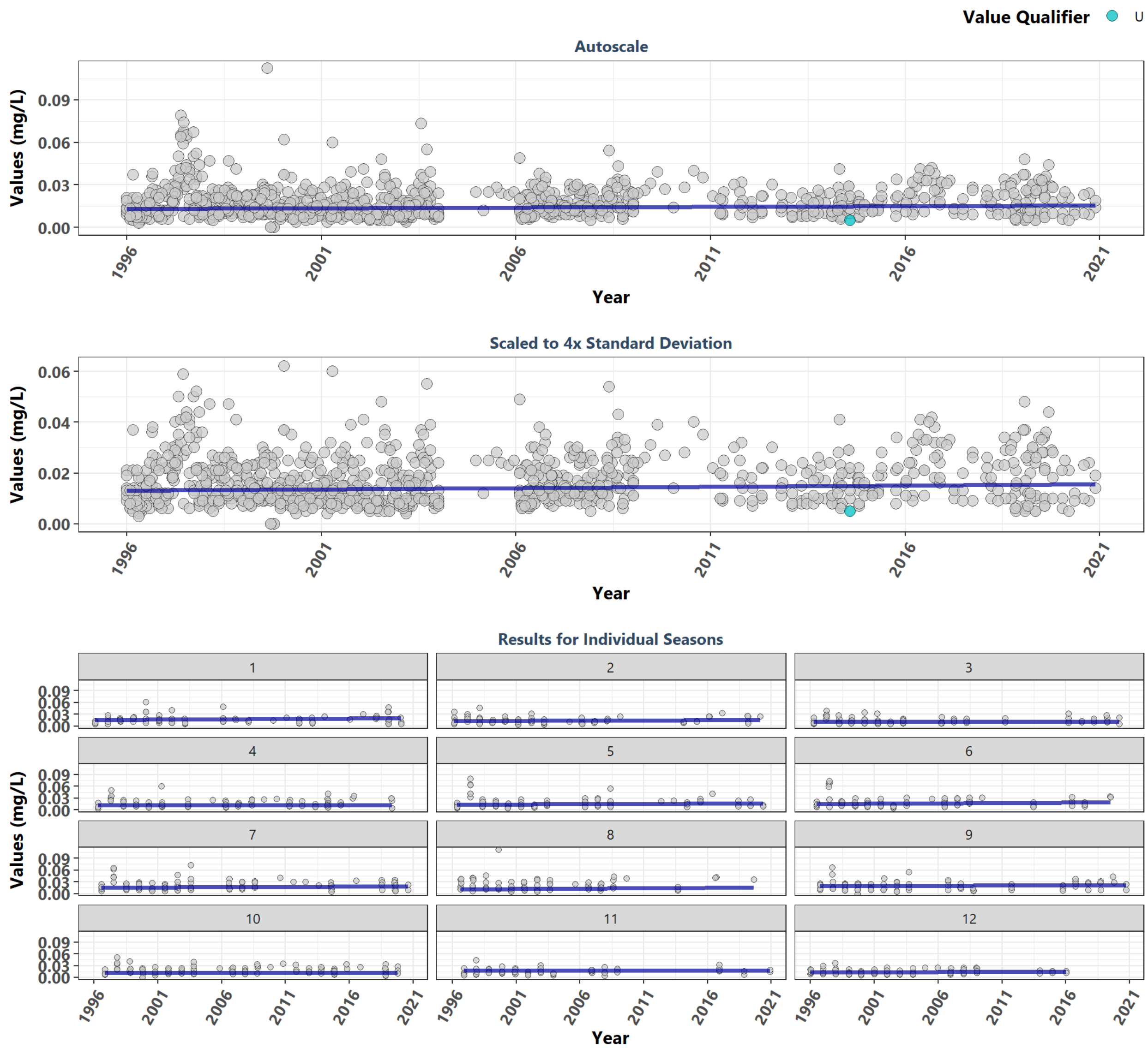
St. Joseph Bay Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p__z	chi__sq	p__chi__sq	Trend
All	934	0	0.3566	0	0.0000	15.9	0.0000	23	0.0178	1
1	74	0	0.0896	0	0.0000	6.1	0.0000	NA	NA	1
2	65	0	0.3395	0	0.0000	5.7	0.0000	NA	NA	1
3	86	0	0.3942	0	-0.0001	5.9	0.0000	NA	NA	2
4	89	0	0.4798	0	0.0000	4.4	0.0000	NA	NA	1
5	88	0	0.3045	0	0.0000	5.5	0.0000	NA	NA	1
6	101	0	0.2618	0	0.0000	6.3	0.0000	NA	NA	1
7	73	0	0.4204	0	-0.0001	3.3	0.0010	NA	NA	2
8	80	0	0.4216	0	0.0000	1.2	0.2388	NA	NA	1
9	66	0	0.4323	0	-0.0001	4.1	0.0000	NA	NA	2
10	77	0	0.4784	0	-0.0001	3.9	0.0001	NA	NA	2
11	65	0	0.3473	0	0.0000	5.0	0.0000	NA	NA	2
12	70	0	0.3177	0	0.0000	4.2	0.0000	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

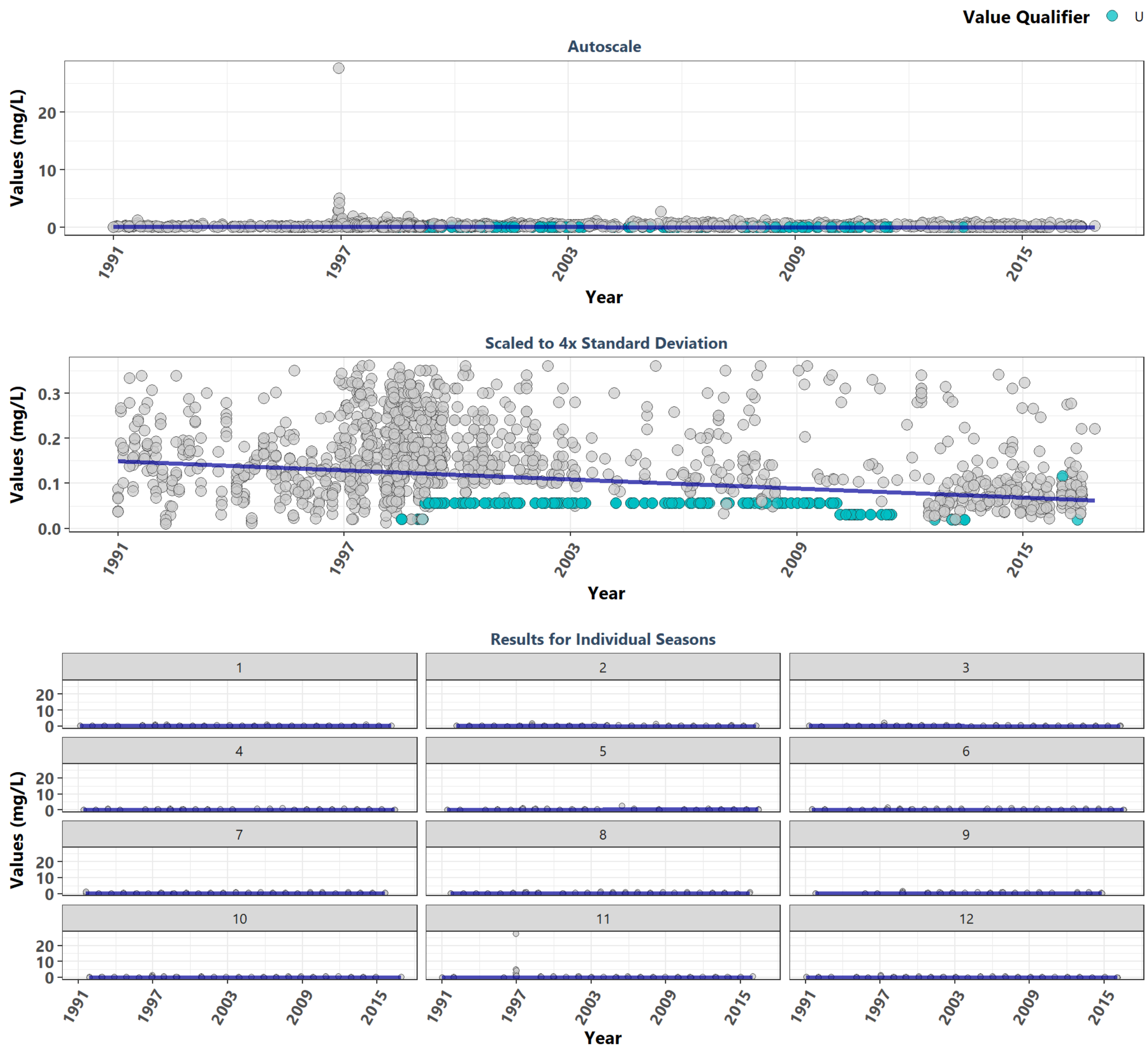
St. Martins Marsh Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	991	0.01	0.0596	0.0001	0.0131	2.8	0.0053	2.1	0.9981	1
1	85	0.01	0.0810	0.0002	0.0159	0.9	0.3912	NA	NA	1
2	80	0.01	0.0633	0.0001	0.0132	-0.1	0.9134	NA	NA	1
3	82	0.01	0.0500	0.0000	0.0120	0.6	0.5621	NA	NA	-1
4	91	0.01	-0.0085	0.0000	0.0120	1.3	0.1885	NA	NA	-1
5	80	0.01	0.0997	0.0002	0.0130	0.8	0.4159	NA	NA	1
6	77	0.02	0.0903	0.0002	0.0155	1.0	0.2971	NA	NA	1
7	91	0.02	0.0741	0.0002	0.0150	0.5	0.6226	NA	NA	1
8	71	0.02	0.0935	0.0002	0.0112	0.3	0.7838	NA	NA	1
9	83	0.02	0.0352	0.0001	0.0196	1.0	0.3215	NA	NA	1
10	102	0.02	0.0437	0.0000	0.0115	1.3	0.1783	NA	NA	-1
11	69	0.01	0.0225	0.0000	0.0180	1.2	0.2247	NA	NA	-1
12	80	0.01	0.0620	0.0001	0.0128	0.7	0.5114	NA	NA	1

[a] $p < 0.00005$ appear as 0 due to rounding

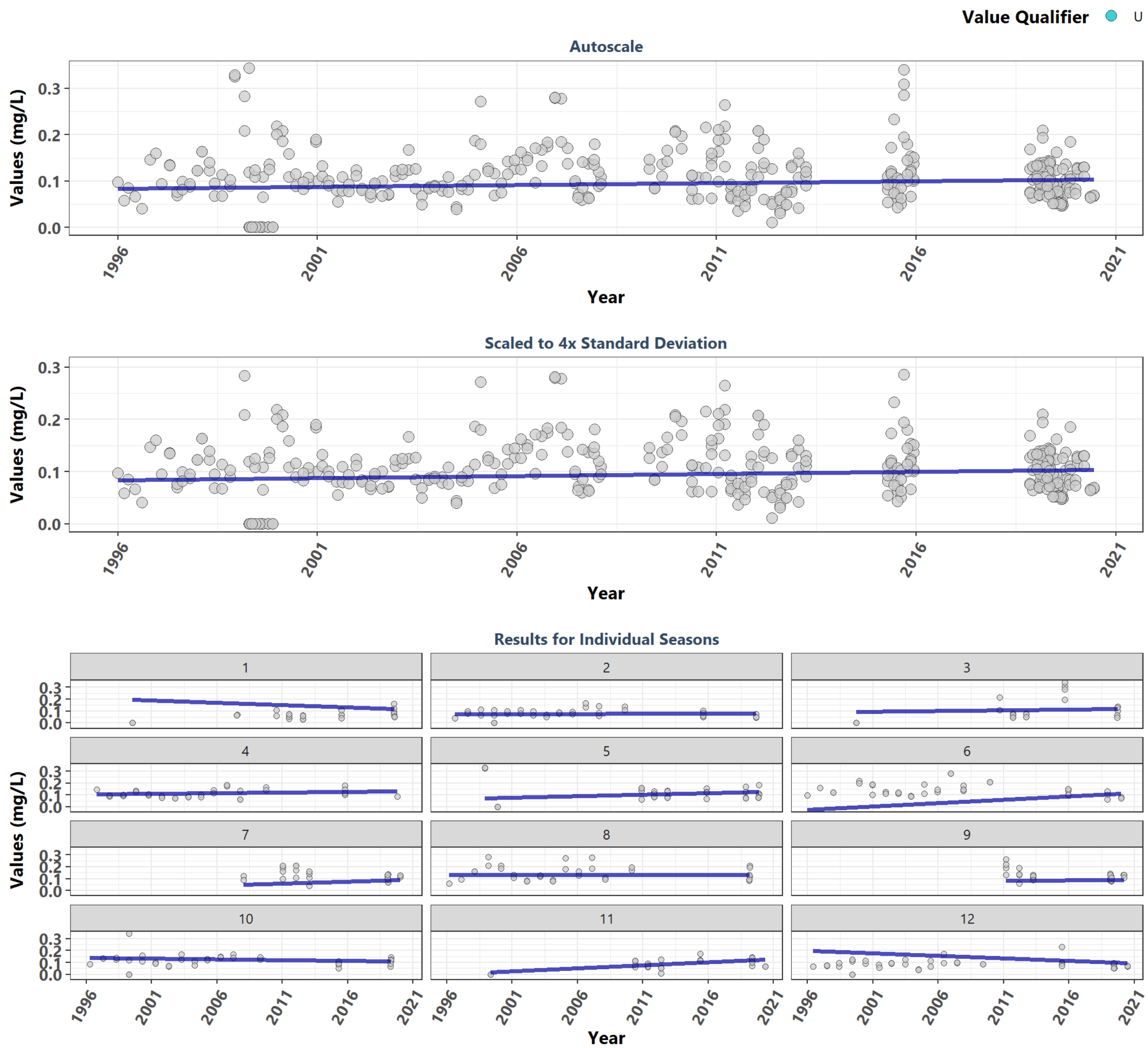
Terra Ceia Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	2075	0.12	-0.1792	-0.0034	0.1498	-12.7	0.0000	44.4	0	-1
1	181	0.10	-0.0669	-0.0012	0.1317	-4.3	0.0000	NA	NA	-1
2	178	0.08	-0.1466	-0.0030	0.1470	-4.0	0.0001	NA	NA	-1
3	185	0.12	-0.2570	-0.0064	0.2104	-1.4	0.1742	NA	NA	-1
4	202	0.11	-0.1717	-0.0046	0.2090	-4.0	0.0001	NA	NA	-1
5	203	0.11	0.1127	0.0051	0.1254	-6.2	0.0000	NA	NA	1
6	176	0.12	-0.2138	-0.0029	0.1275	-2.9	0.0036	NA	NA	-1
7	181	0.14	-0.1704	-0.0040	0.2030	-5.2	0.0000	NA	NA	-1
8	153	0.16	-0.1869	-0.0030	0.1384	-3.1	0.0017	NA	NA	-1
9	103	0.18	-0.2277	-0.0052	0.1774	1.7	0.0882	NA	NA	-1
10	168	0.17	-0.2900	-0.0047	0.1527	-3.3	0.0009	NA	NA	-1
11	165	0.12	-0.1989	-0.0044	0.1576	-4.4	0.0000	NA	NA	-1
12	180	0.12	-0.1972	-0.0026	0.1121	-4.0	0.0001	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

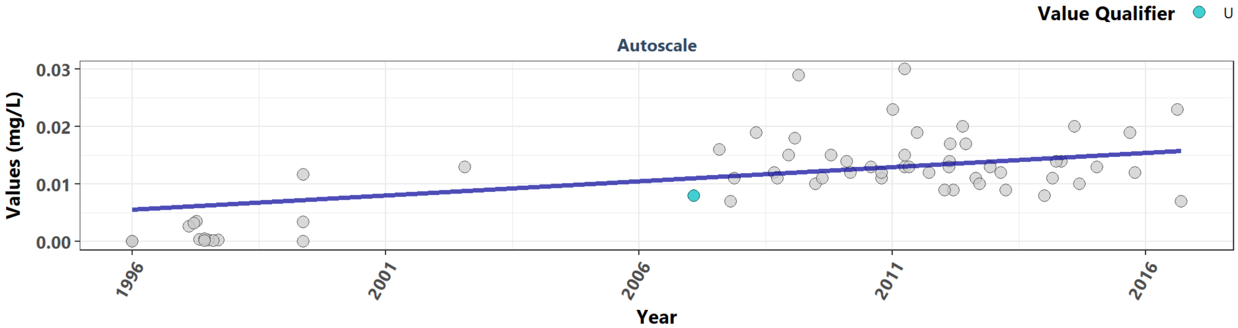
Tomoka Marsh Aquatic Preserve

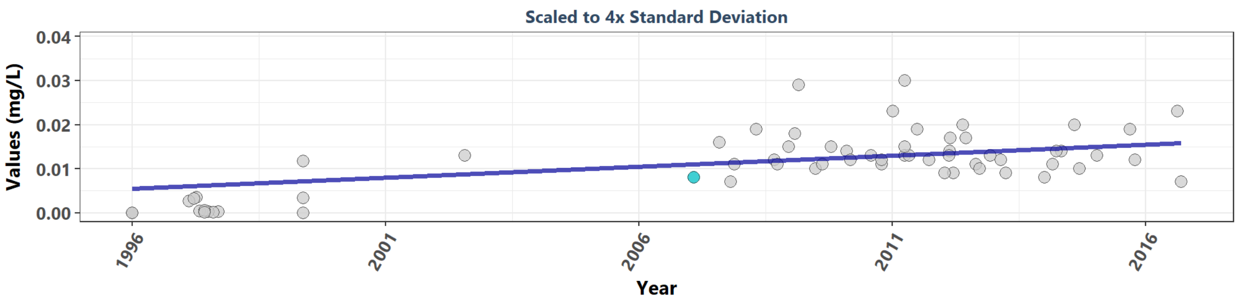


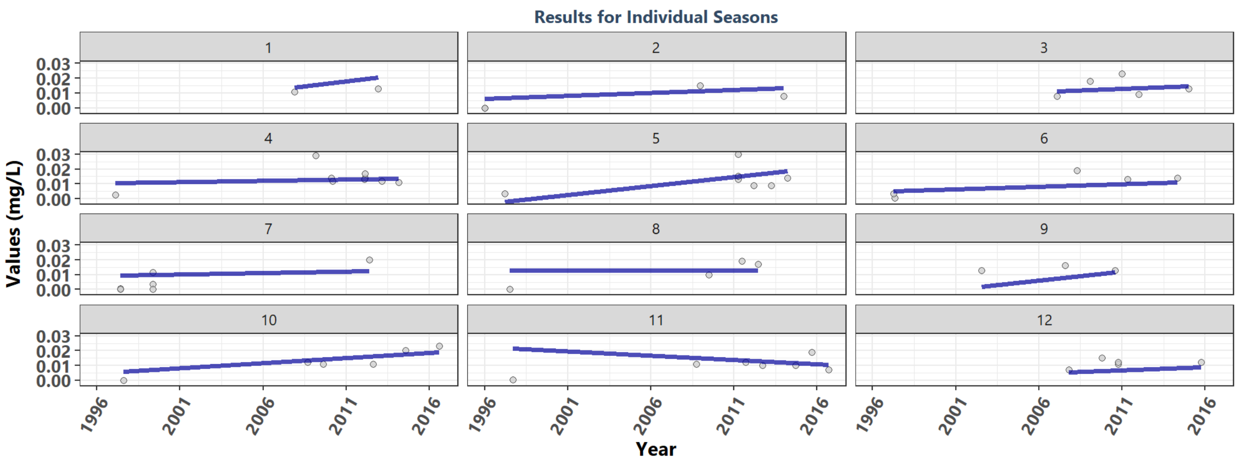
Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	391	0.10	0.0794	0.0008	0.0832	2.1	0.0326	32.1	0.0007	1
1	29	0.06	-0.2892	-0.0038	0.2072	3.0	0.0030	NA	NA	-1
2	41	0.07	0.0305	0.0002	0.0732	0.3	0.7862	NA	NA	1
3	24	0.07	0.1204	0.0013	0.0887	2.7	0.0063	NA	NA	1
4	31	0.10	0.0929	0.0009	0.1070	0.9	0.3482	NA	NA	1
5	32	0.11	0.0766	0.0025	0.0649	0.6	0.5443	NA	NA	1
6	35	0.13	0.3949	0.0057	-0.0252	-1.0	0.3118	NA	NA	1
7	26	0.13	0.3842	0.0032	0.0098	-1.5	0.1335	NA	NA	1
8	33	0.13	0.0227	0.0000	0.1309	0.2	0.8637	NA	NA	1
9	26	0.12	0.0836	0.0005	0.0778	-2.2	0.0311	NA	NA	1
10	42	0.11	-0.1210	-0.0012	0.1384	0.9	0.3882	NA	NA	-1
11	30	0.08	0.3977	0.0050	0.0007	3.1	0.0018	NA	NA	1
12	42	0.08	-0.2062	-0.0042	0.1978	0.8	0.4391	NA	NA	-1

[a] $p < 0.00005$ appear as 0 due to rounding

Yellow River Marsh Aquatic Preserve







Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	64	0.01	0.2807	0.0005	0.0055	1.9	0.0609	6.2	0.8622	0
1	2	NA	0.4667	0.0013	-0.0020	NA	NA	NA	NA	NA
2	5	0.00	1.0000	0.0004	0.0062	1.1	0.2541	NA	NA	0
3	5	0.01	0.2000	0.0004	0.0068	0.2	0.8065	NA	NA	0
4	9	0.01	0.0476	0.0002	0.0105	-0.5	0.5888	NA	NA	0
5	7	0.01	0.6667	0.0012	-0.0036	0.0	1.0000	NA	NA	0
6	5	0.01	0.0952	0.0003	0.0047	1.0	0.3122	NA	NA	0
7	6	0.00	0.2000	0.0002	0.0092	1.2	0.2174	NA	NA	0
8	4	0.01	0.0000	0.0000	0.0130	1.0	0.3082	NA	NA	0
9	3	0.01	0.6667	0.0012	-0.0060	0.0	1.0000	NA	NA	0
10	6	0.01	0.5000	0.0007	0.0049	1.7	0.0852	NA	NA	0
11	7	0.01	-0.1667	-0.0006	0.0223	0.2	0.8793	NA	NA	0
12	5	0.01	0.5000	0.0004	0.0000	0.3	0.7947	NA	NA	0

[a] $p < 0.00005$ appear as 0 due to rounding

Appendix V: Managed Area Summary Box Plots

Data is taken and grouped by `ManagedAreaName`. The scripts that create plots follow this format

1. Use the data set that only has `SufficientData` of `TRUE` for the desired managed area
2. Determine the earliest and latest year of the data to create x-axis scale and intervals
3. Determine the minimum, mean, and standard deviation for the data to be used for y-axis scales
    - Excludes the top 2% of values to reduce the impact of extreme outliers on the y-axis scale
4. Set what values are to be used for the x-axis, y-axis, and the variable that should determine groups for the box plots
5. Set the plot type as a box plot with the size of the outlier points
6. Create the title, x-axis, y-axis, and color fill labels
7. Set the y and x limits
8. Make the axis labels bold
9. Plot the arrangement as a set of panels

The following plots are arranged by `ManagedAreaName` with data grouped by `Year`, then `Year` and `Month`, then finally `Month` only. Each managed area will have 3 sets of plots, each with 3 panels in them. Each panel goes as follows:

1. Y-axis autoscaled
2. Y-axis set to be mean + 4 times the standard deviation
3. Y-axis set to be mean + 4 times the standard deviation for most recent 10 years of data

```
if(n==0){
   print("There are no managed areas that qualify.")
} else {
   for (i in 1:n) {
      plot_data <- data[data$SufficientData==TRUE &
                           data$ManagedAreaName==MA_Include[i],]
      year_lower <- min(plot_data$Year)
      year_upper <- max(plot_data$Year)
      min_RV <- min(plot_data$ResultValue)
      mn_RV <- mean(plot_data$ResultValue[plot_data$ResultValue <
                                          quantile(data$ResultValue, 0.98)])
      sd_RV <- sd(plot_data$ResultValue[plot_data$ResultValue <
                                          quantile(data$ResultValue, 0.98)])
      x_scale <- ifelse(year_upper - year_lower > 30, 10, 5)
      y_scale <- mn_RV + 4 * sd_RV

      ##Year plots
      p1 <- ggplot(data=plot_data,
                   aes(x=Year, y=ResultValue, group=Year)) +
         geom_boxplot(color="#333333", fill="#cccccc", outlier.shape=21,
                      outlier.size=3, outlier.color="#333333",
                      outlier.fill="#cccccc", outlier.alpha=0.75) +
         labs(subtitle="Autoscale",
              x="Year", y=paste0("Values (", unit, ")")) +
         scale_x_continuous(limits=c(year_lower - 1, year_upper + 1),
                            breaks=rev(seq(year_upper,
                                           year_lower, -x_scale))) +
         plot_theme
```

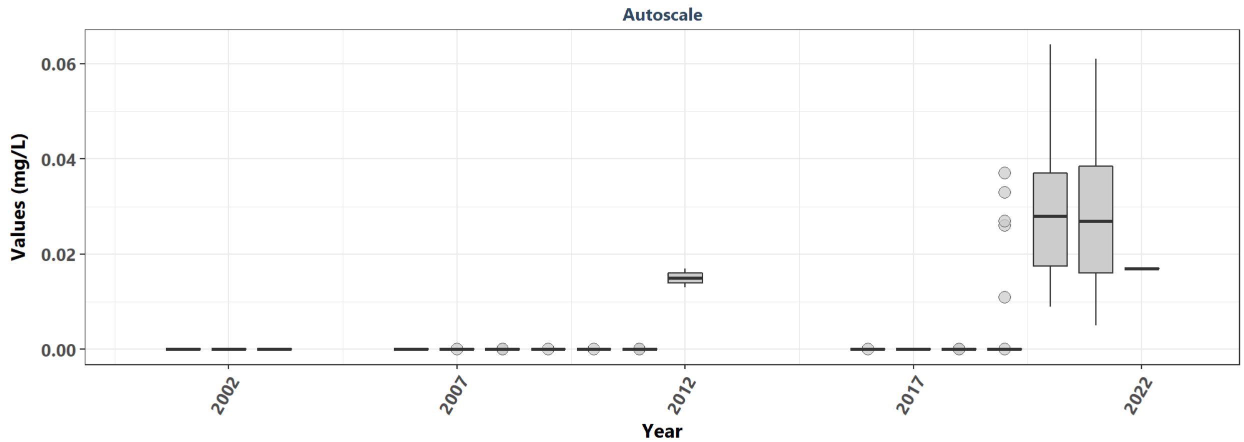
```r
p2 <- ggplot(data=plot_data,
             aes(x=Year, y=ResultValue, group=Year)) +
   geom_boxplot(color="#333333", fill="#cccccc", outlier.shape=21,
                outlier.size=3, outlier.color="#333333",
                outlier.fill="#cccccc", outlier.alpha=0.75) +
   labs(subtitle="Scaled to 4x Standard Deviation",
        x="Year", y=paste0("Values (", unit, ")")) +
   ylim(min_RV, y_scale) +
   scale_x_continuous(limits=c(year_lower - 1, year_upper + 1),
                      breaks=rev(seq(year_upper,
                                     year_lower, -x_scale))) +
   plot_theme

p3 <- ggplot(data=plot_data[plot_data$Year >= year_upper - 10, ],
             aes(x=Year, y=ResultValue, group=Year)) +
   geom_boxplot(color="#333333", fill="#cccccc", outlier.shape=21,
                outlier.size=3, outlier.color="#333333",
                outlier.fill="#cccccc", outlier.alpha=0.75) +
   labs(subtitle="Scaled to 4x Standard Deviation, Last 10 Years",
        x="Year", y=paste0("Values (", unit, ")")) +
   ylim(min_RV, y_scale) +
   scale_x_continuous(limits=c(year_upper - 10.5, year_upper + 1),
                      breaks=rev(seq(year_upper, year_upper - 10,-2))) +
   plot_theme

Yset <- ggarrange(p1, p2, p3, ncol=1)

p0 <- ggplot() + labs(title=paste0(MA_Include[i]),
                      subtitle="By Year") +
   plot_theme + theme(panel.border=element_blank(),
                      panel.grid.major=element_blank(),
                      panel.grid.minor=element_blank(),
                      axis.line=element_blank())

## Year & Month Plots
p4 <- ggplot(data=plot_data,
             aes(x=YearMonthDec, y=ResultValue,
                 group=YearMonth, color=as.factor(Month))) +
   geom_boxplot(fill="#cccccc", outlier.size=1.5, outlier.alpha=0.75) +
   labs(subtitle="Autoscale",
        x="Year", y=paste0("Values (", unit, ")"), color="Month") +
   scale_x_continuous(limits=c(year_lower - 1, year_upper + 1),
                      breaks=rev(seq(year_upper,
                                     year_lower, -x_scale))) +
   plot_theme +
   theme(legend.position="none")

p5 <- ggplot(data=plot_data,
             aes(x=YearMonthDec, y=ResultValue,
                 group=YearMonth, color=as.factor(Month))) +
   geom_boxplot(fill="#cccccc", outlier.size=1.5, outlier.alpha=0.75) +
   labs(subtitle="Scaled to 4x Standard Deviation",
        x="Year", y=paste0("Values (", unit, ")"), color="Month") +
```

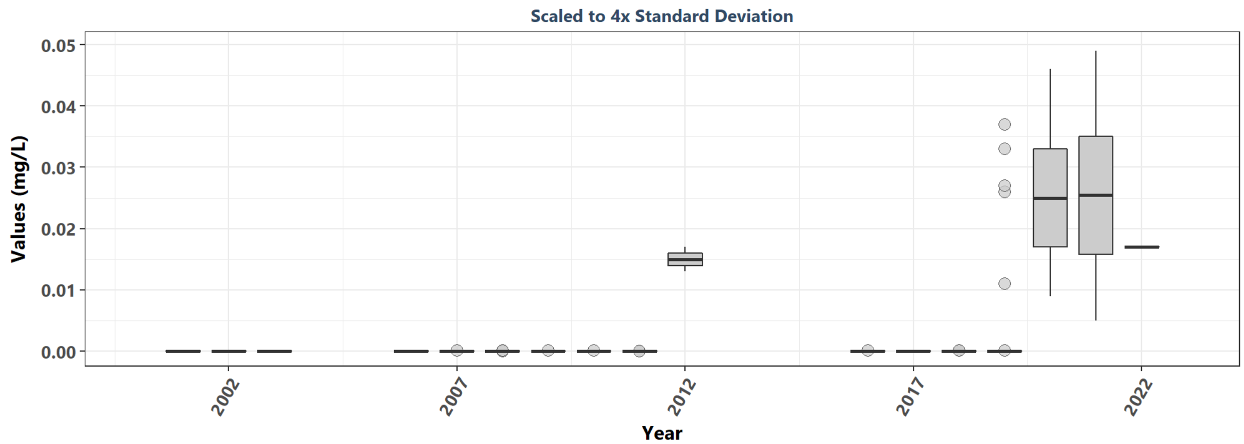
```r
    ylim(min_RV, y_scale) +
    scale_x_continuous(limits=c(year_lower - 1, year_upper + 1),
                       breaks=rev(seq(year_upper,
                                      year_lower, -x_scale))) +
    plot_theme +
    theme(legend.position="top", legend.box="horizontal") +
    guides(color=guide_legend(nrow=1))

p6 <- ggplot(data=plot_data[plot_data$Year >= year_upper - 10, ],
             aes(x=YearMonthDec, y=ResultValue,
                 group=YearMonth, color=as.factor(Month))) +
    geom_boxplot(fill="#cccccc", outlier.size=1.5, outlier.alpha=0.75) +
    labs(subtitle="Scaled to 4x Standard Deviation, Last 10 Years",
         x="Year", y=paste0("Values (", unit, ")"), color="Month") +
    ylim(min_RV, y_scale) +
    scale_x_continuous(limits=c(year_upper - 10.5, year_upper + 1),
                       breaks=rev(seq(year_upper, year_upper - 10,-2))) +
    plot_theme +
    theme(legend.position="none")

leg1 <- get_legend(p5)
YMset <- ggarrange(leg1, p4, p5 + theme(legend.position="none"), p6,
                   ncol=1, heights=c(0.1, 1, 1, 1))

p00 <- ggplot() + labs(title=paste0(MA_Include[i]),
                       subtitle="By Year & Month") + plot_theme +
    theme(panel.border=element_blank(),
          panel.grid.major=element_blank(),
          panel.grid.minor=element_blank(), axis.line=element_blank())

## Month Plots
p7 <- ggplot(data=plot_data,
             aes(x=Month, y=ResultValue,
                 group=Month, fill=as.factor(Month))) +
    geom_boxplot(color="#333333", outlier.shape=21, outlier.size=3,
                 outlier.color="#333333", outlier.alpha=0.75) +
    labs(subtitle="Autoscale",
         x="Month", y=paste0("Values (", unit, ")"), fill="Month") +
    scale_x_continuous(limits=c(0, 13), breaks=seq(3, 12, 3)) +
    plot_theme +
    theme(legend.position="none")

p8 <- ggplot(data=plot_data,
             aes(x=Month, y=ResultValue,
                 group=Month, fill=as.factor(Month))) +
    geom_boxplot(color="#333333", outlier.shape=21, outlier.size=3,
                 outlier.color="#333333", outlier.alpha=0.75) +
    labs(subtitle="Scaled to 4x Standard Deviation",
         x="Month", y=paste0("Values (", unit, ")"), fill="Month") +
    ylim(min_RV, y_scale) +
    scale_x_continuous(limits=c(0, 13), breaks=seq(3, 12, 3)) +
    plot_theme +
    theme(legend.position="top", legend.box="horizontal") +
```

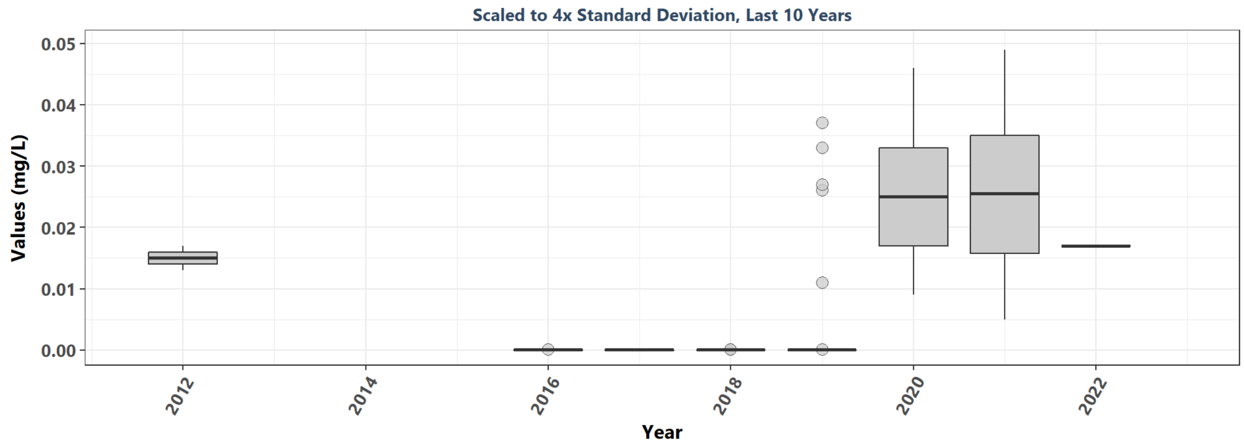
```
        guides(fill=guide_legend(nrow=1))

      p9 <- ggplot(data=plot_data[plot_data$Year >= year_upper - 10, ],
                   aes(x=Month, y=ResultValue,
                       group=Month, fill=as.factor(Month))) +
        geom_boxplot(color="#333333", outlier.shape=21, outlier.size=3,
                     outlier.color="#333333", outlier.alpha=0.75) +
        labs(subtitle="Scaled to 4x Standard Deviation, Last 10 Years",
             x="Month", y=paste0("Values (", unit, ")"), fill="Month") +
        ylim(min_RV, y_scale) +
        scale_x_continuous(limits=c(0, 13), breaks=seq(3, 12, 3)) +
        plot_theme +
        theme(legend.position="none")

      leg2 <- get_legend(p8)
      Mset <- ggarrange(leg2, p7, p8 + theme(legend.position="none"), p9,
                        ncol=1, heights=c(0.1, 1, 1, 1))

      p000 <- ggplot() + labs(title=paste0(MA_Include[i]),
                              subtitle="By Month") + plot_theme +
        theme(panel.border=element_blank(),
              panel.grid.major=element_blank(),
              panel.grid.minor=element_blank(), axis.line=element_blank())

      print(ggarrange(p0, Yset, ncol=1, heights=c(0.07, 1)))
      print(ggarrange(p00, YMset, ncol=1, heights=c(0.07, 1)))
      print(ggarrange(p000, Mset, ncol=1, heights=c(0.07, 1, 0.7)))

      rm(plot_data)
      rm(p1, p2, p3, p4, p5, p6, p7, p8, p9, p0, p00, p000, leg1, leg2,
        Yset, YMset, Mset)
   }
}
```

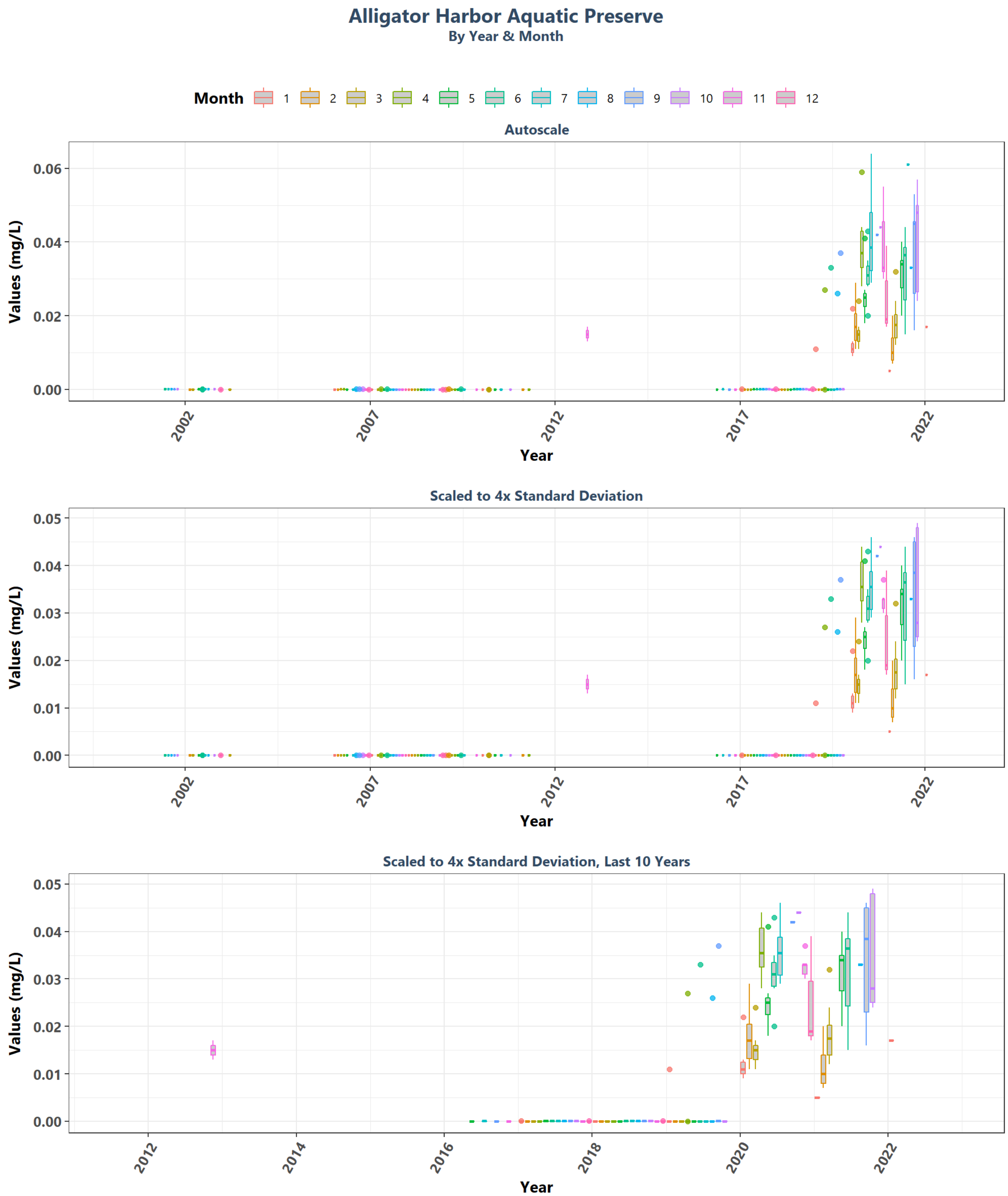
Alligator Harbor Aquatic Preserve

By Year





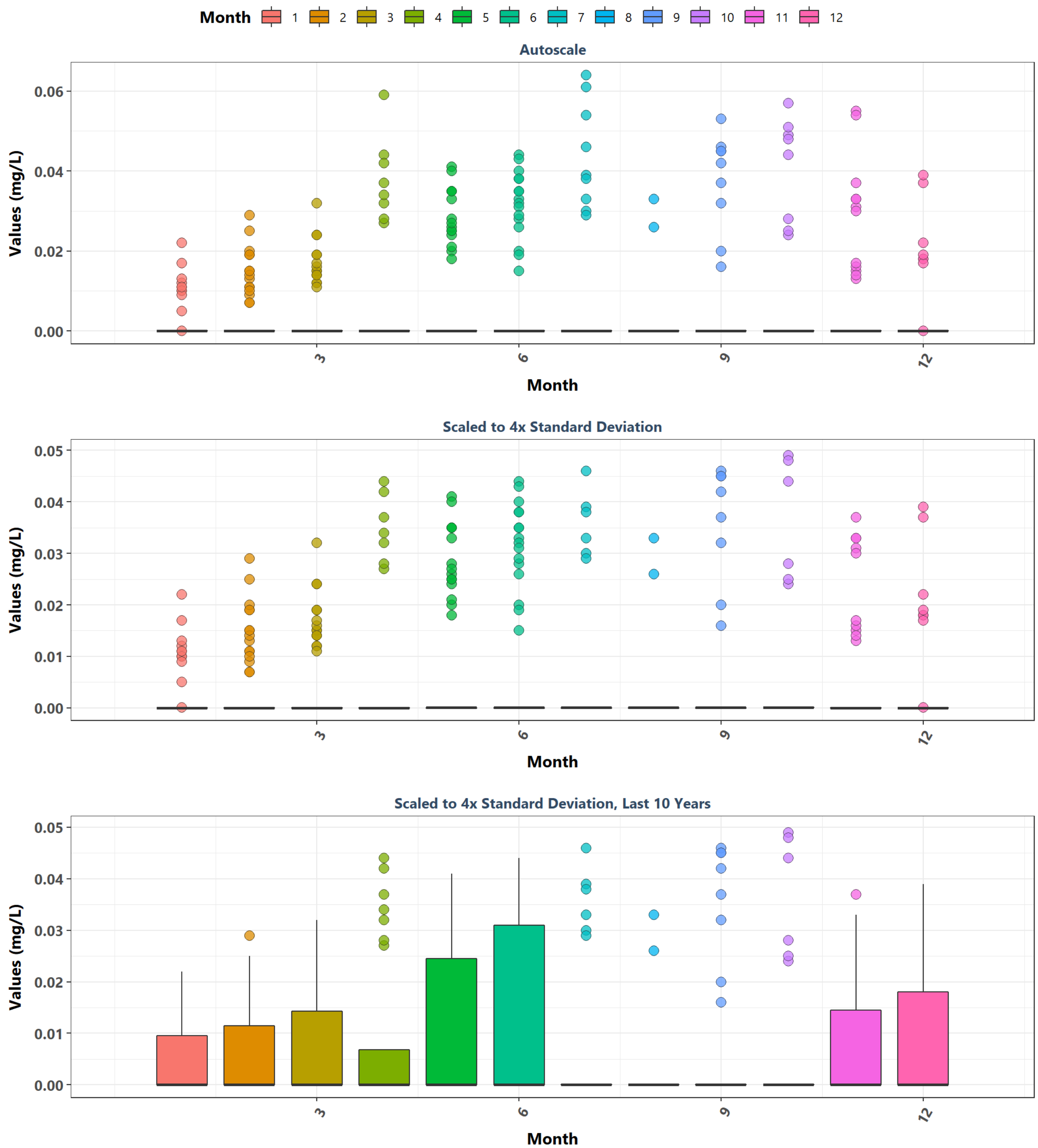




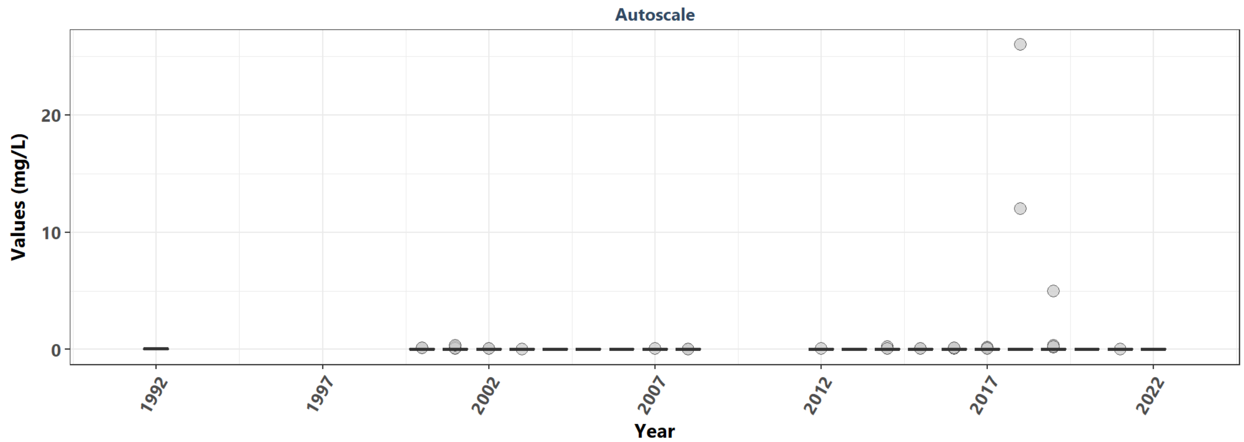
Alligator Harbor Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.00
0.02
0.04
0.06
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.01
0.02
0.03
0.04
0.05
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.01
0.02
0.03
0.04
0.05
2012
2014
2016
2018
2020
2022
Year

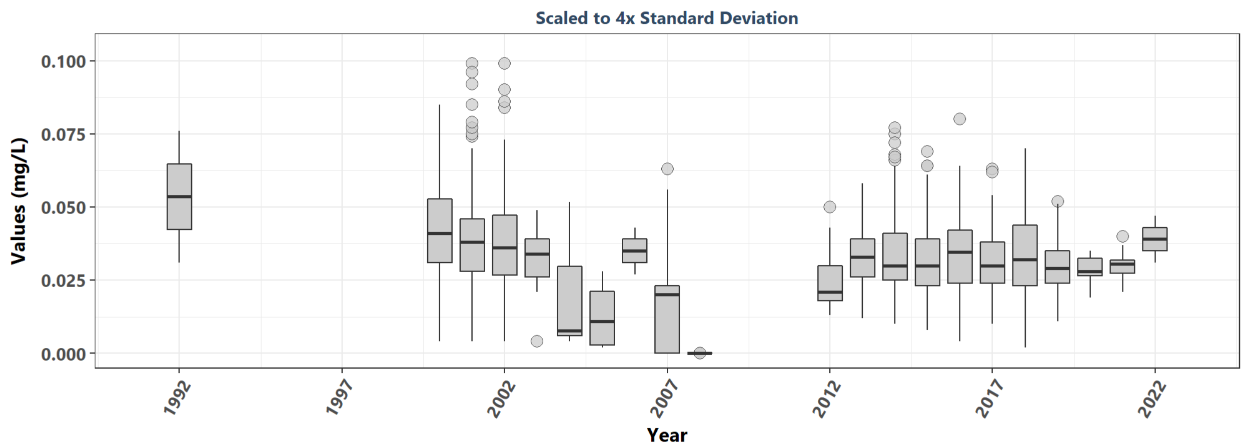
Alligator Harbor Aquatic Preserve

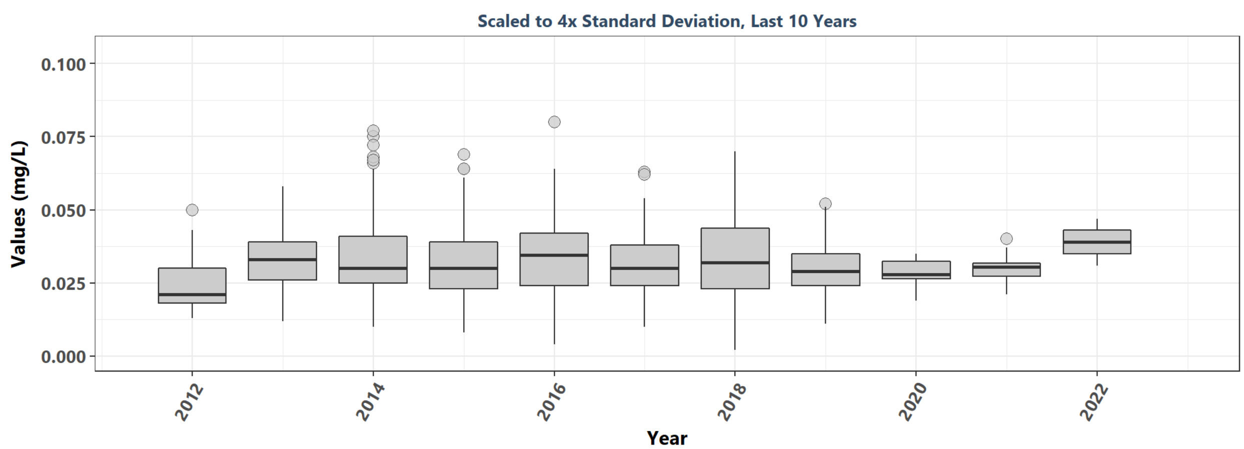
By Month

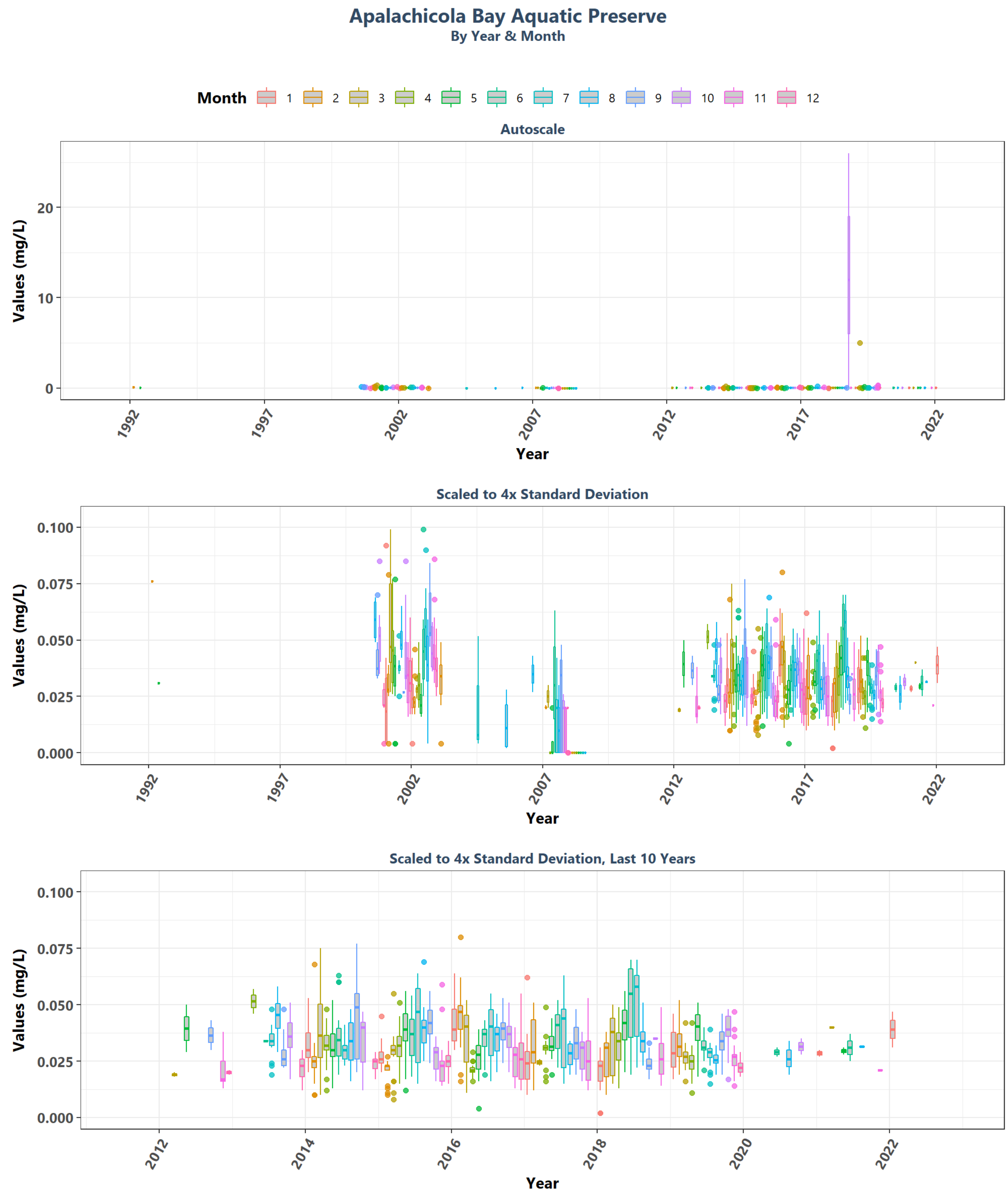


Apalachicola Bay Aquatic Preserve
By Year



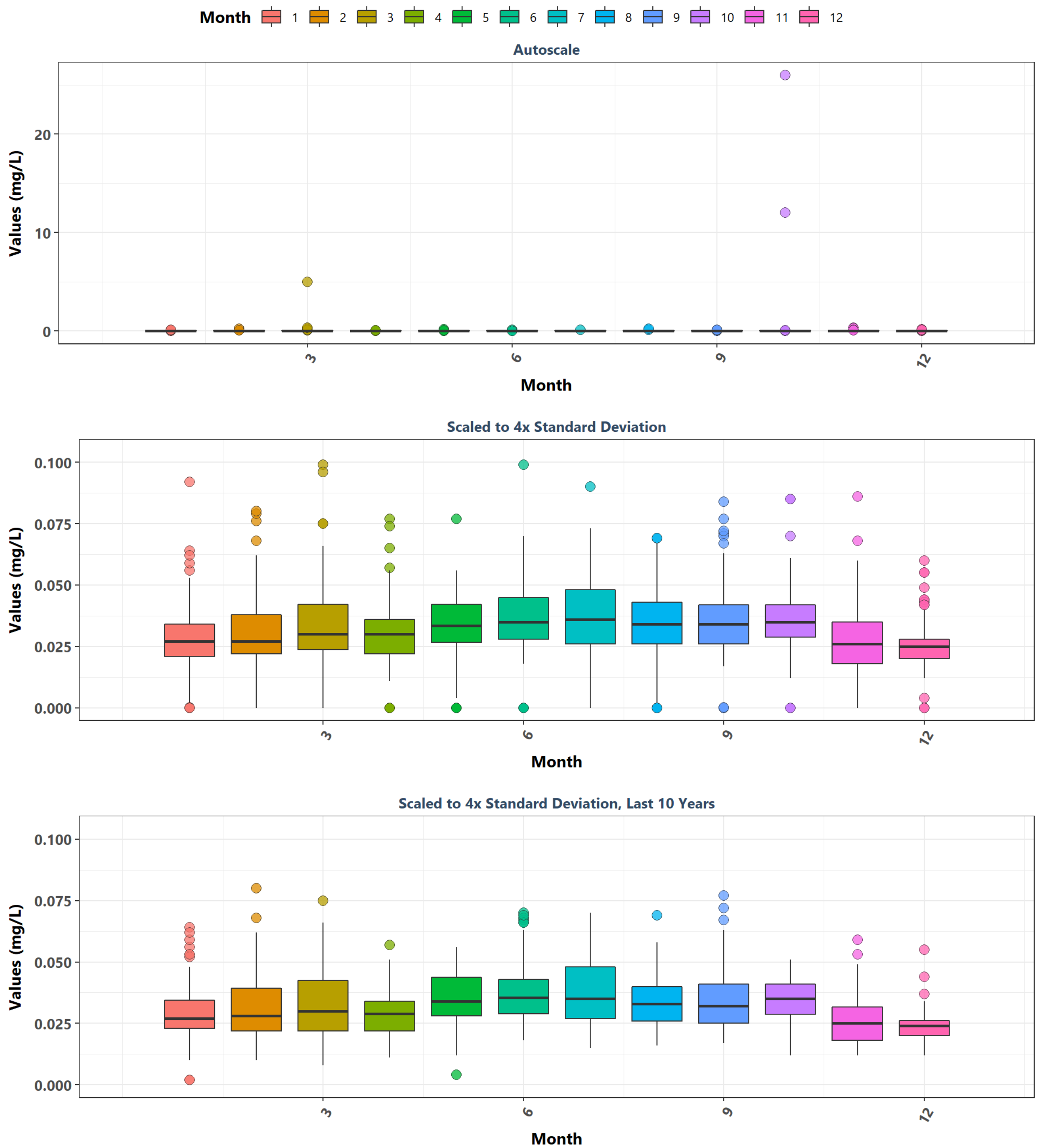






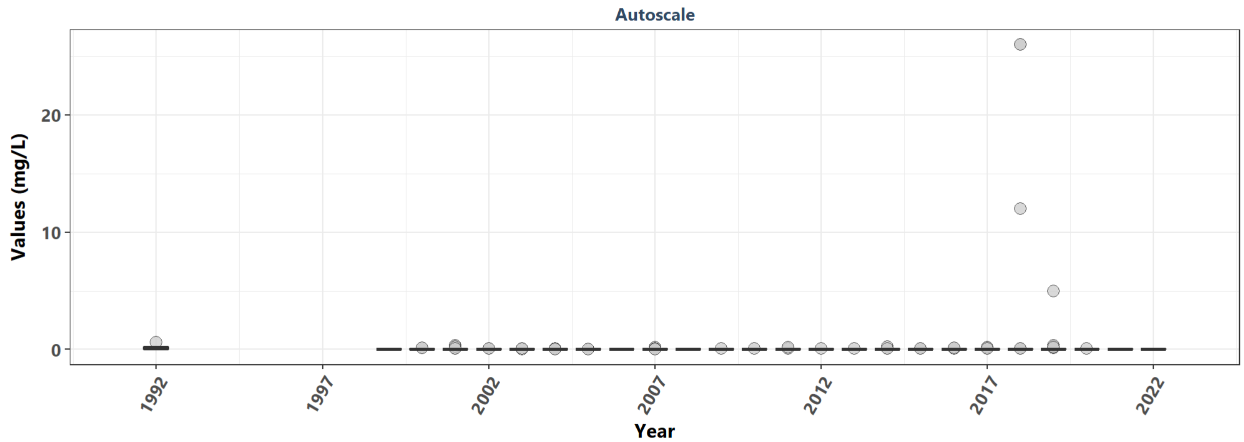
Apalachicola Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
Year
Scaled to 4x Standard Deviation
Values (mg/L)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
Year

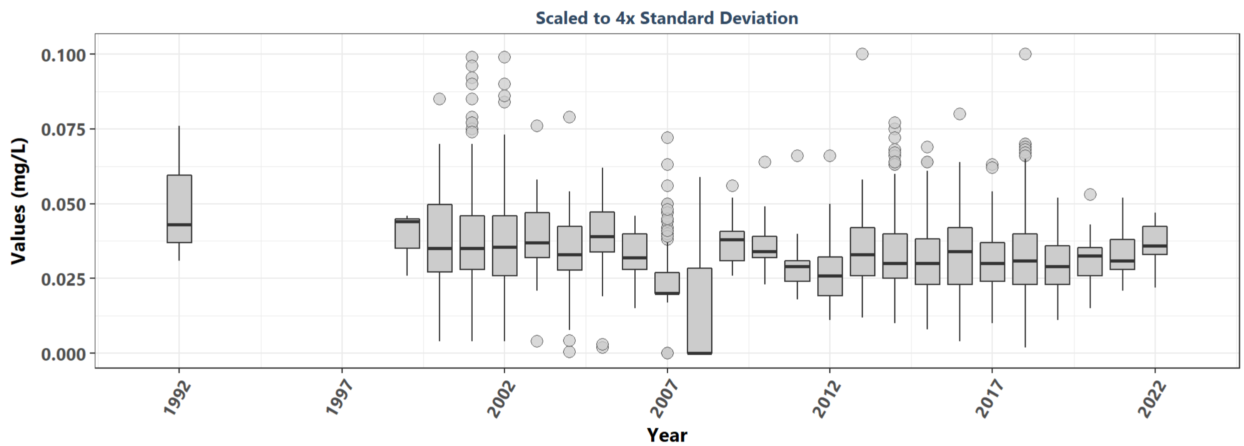
Apalachicola Bay Aquatic Preserve
By Month

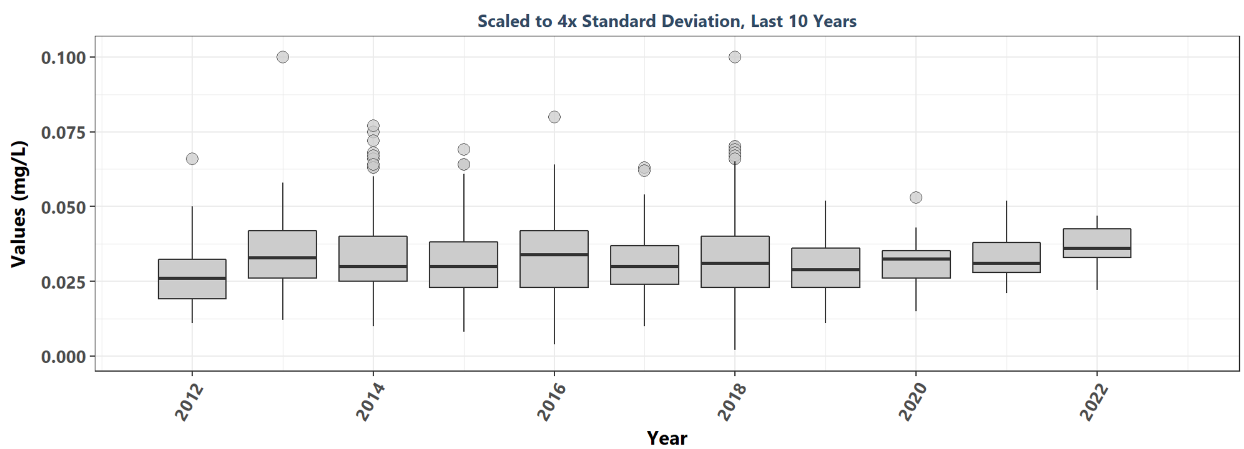


Apalachicola National Estuarine Research Reserve

By Year

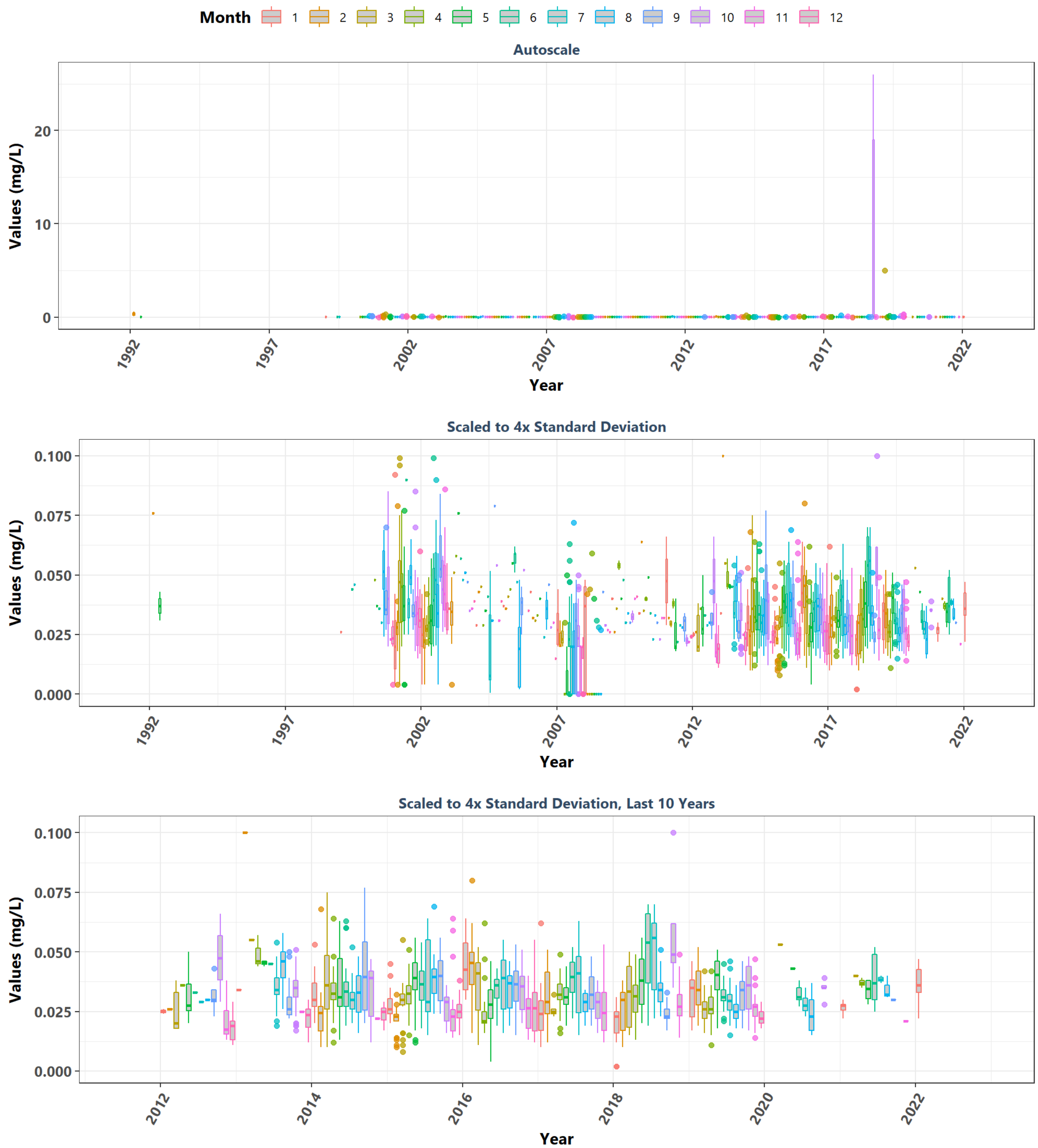






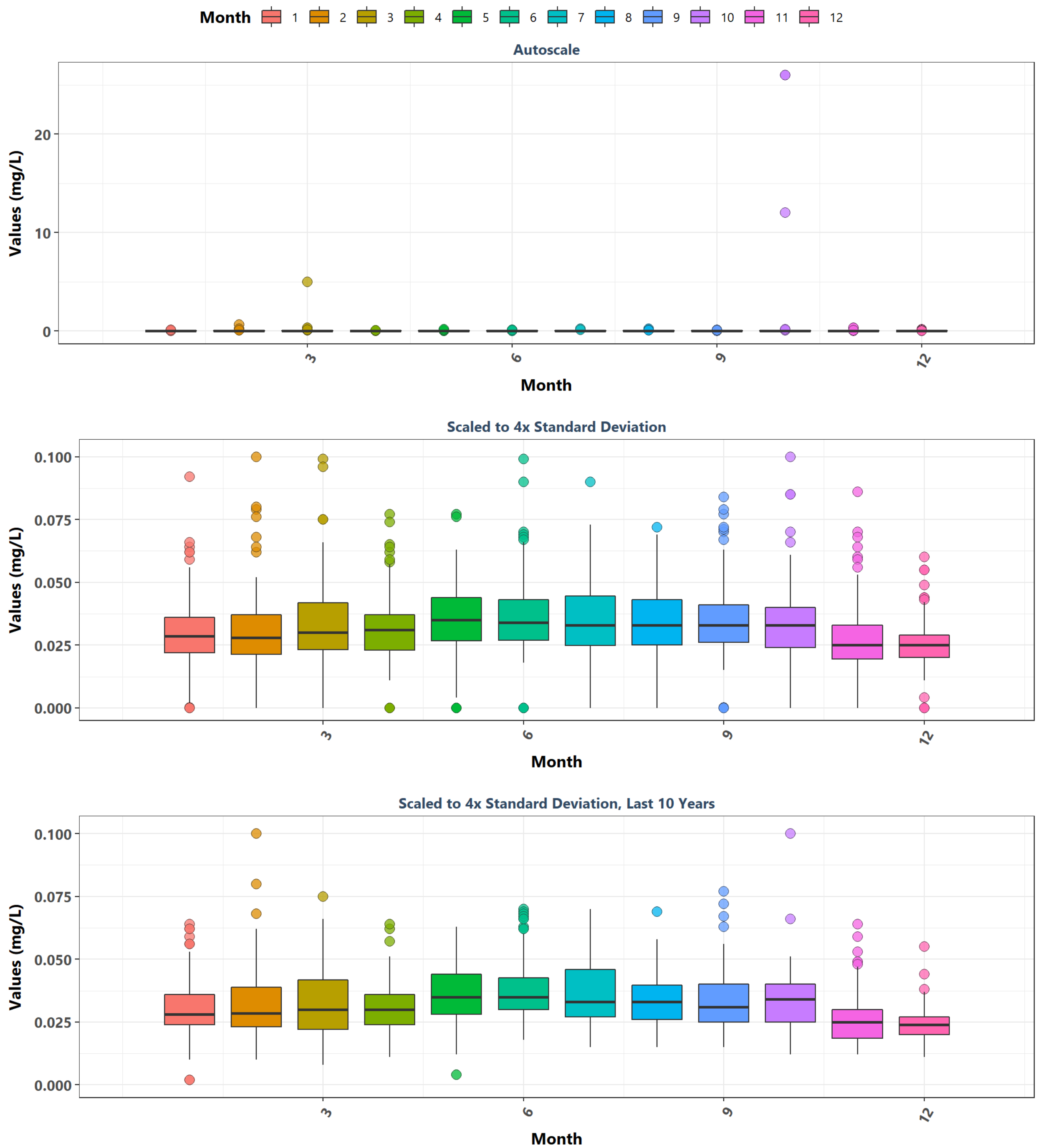
Apalachicola National Estuarine Research Reserve

By Year & Month



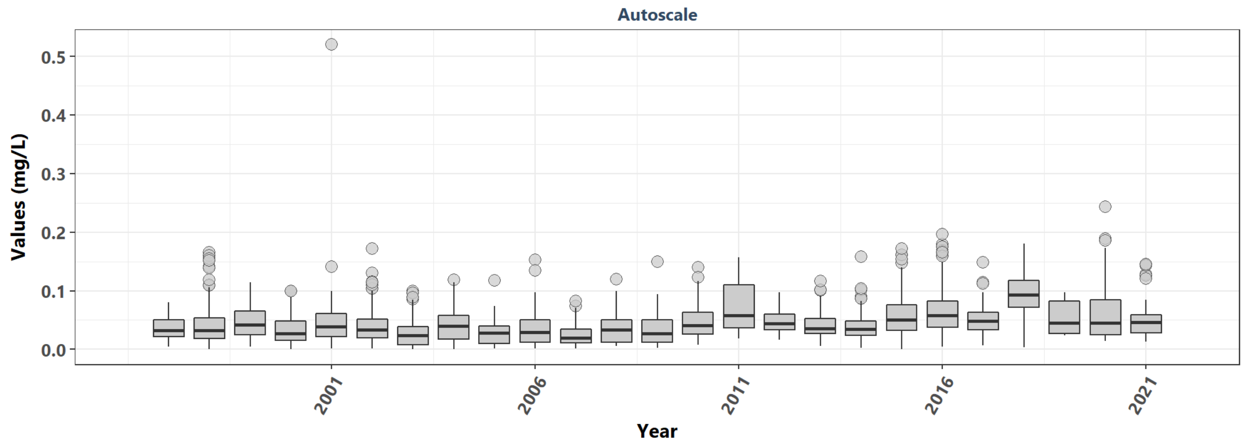
Apalachicola National Estuarine Research Reserve

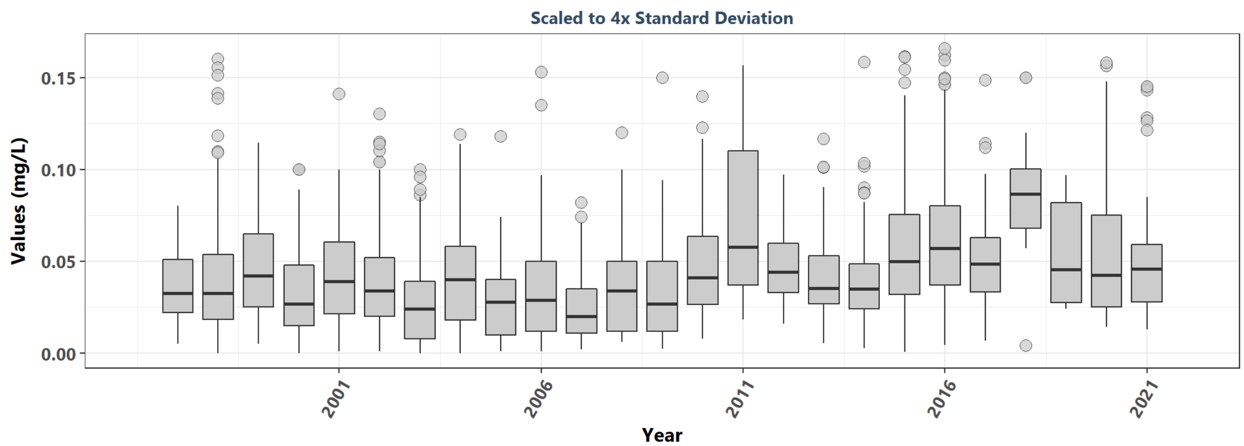
By Month

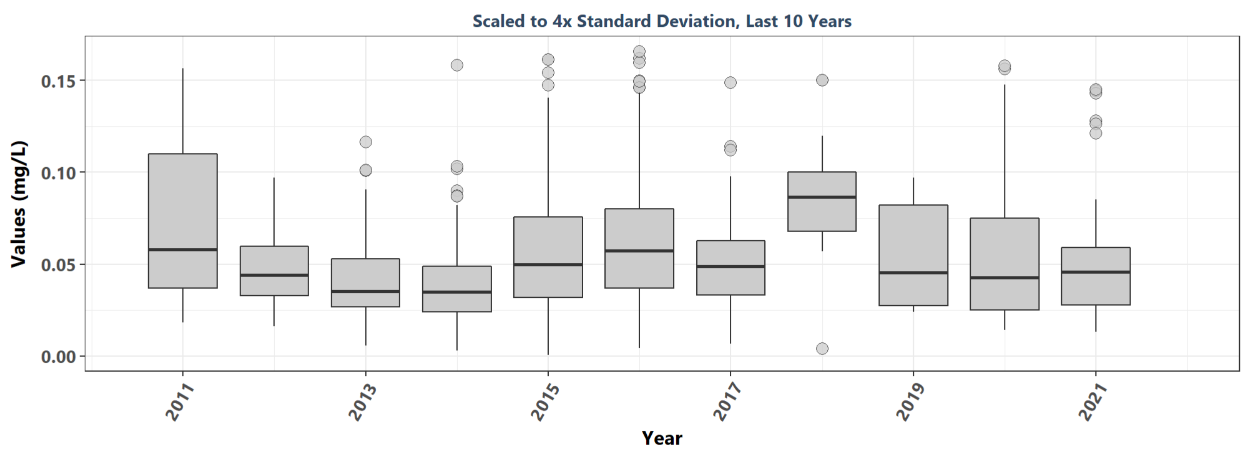


Banana River Aquatic Preserve

By Year

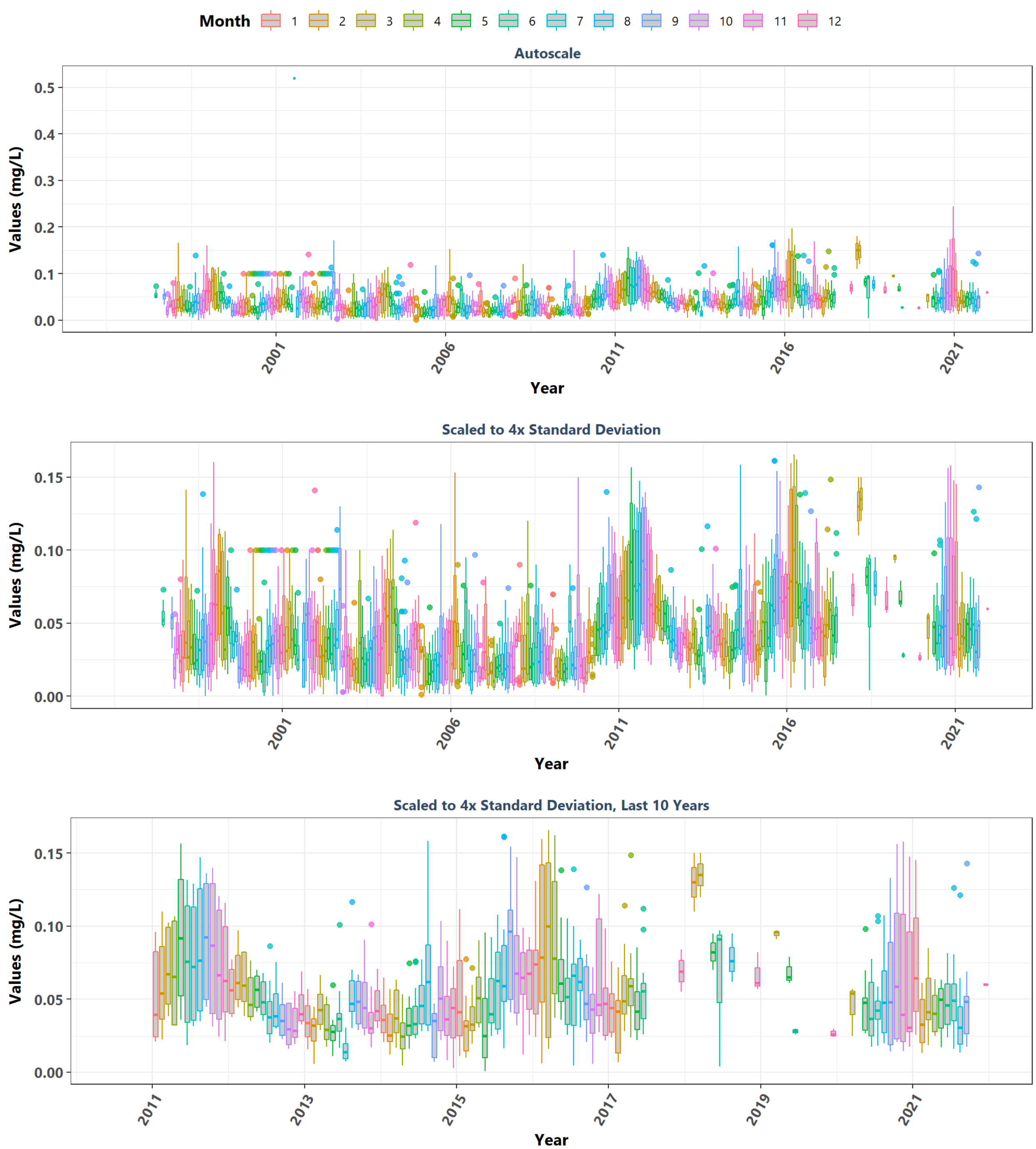




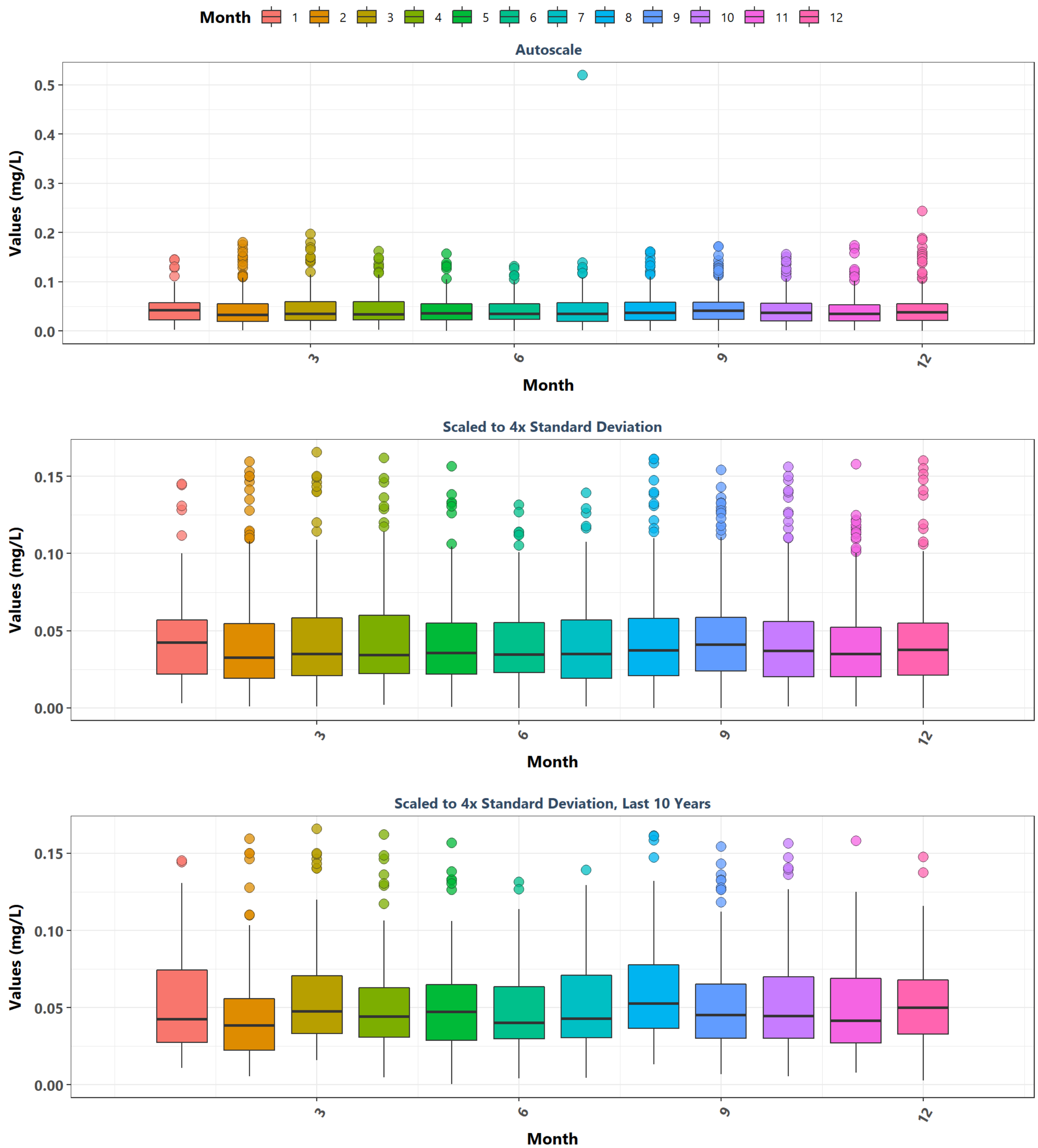


Banana River Aquatic Preserve

By Year & Month



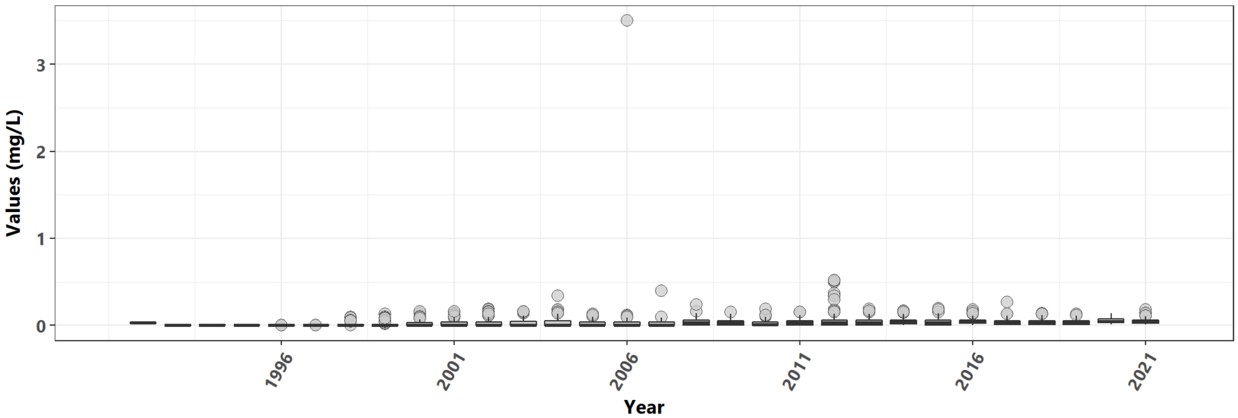
Banana River Aquatic Preserve
By Month



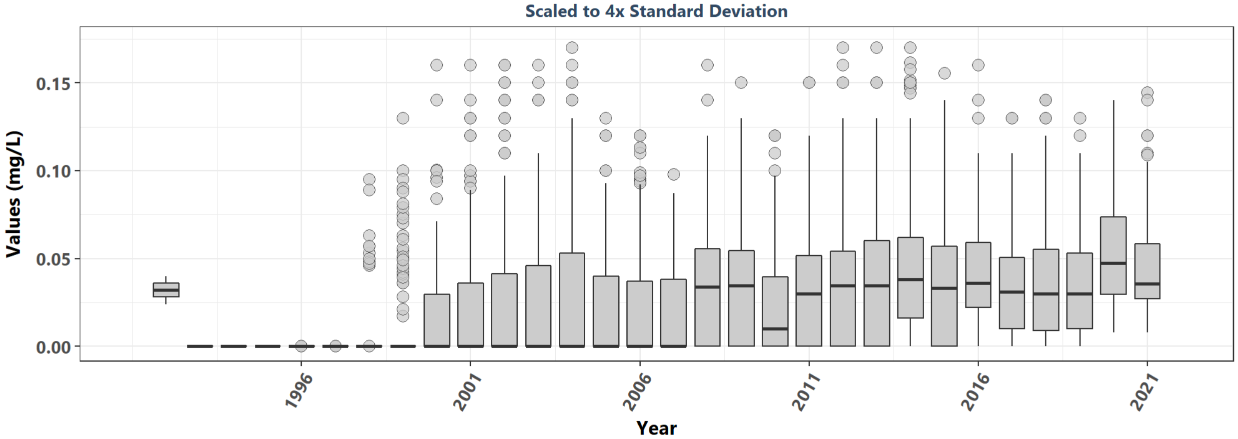
Big Bend Seagrasses Aquatic Preserve

By Year

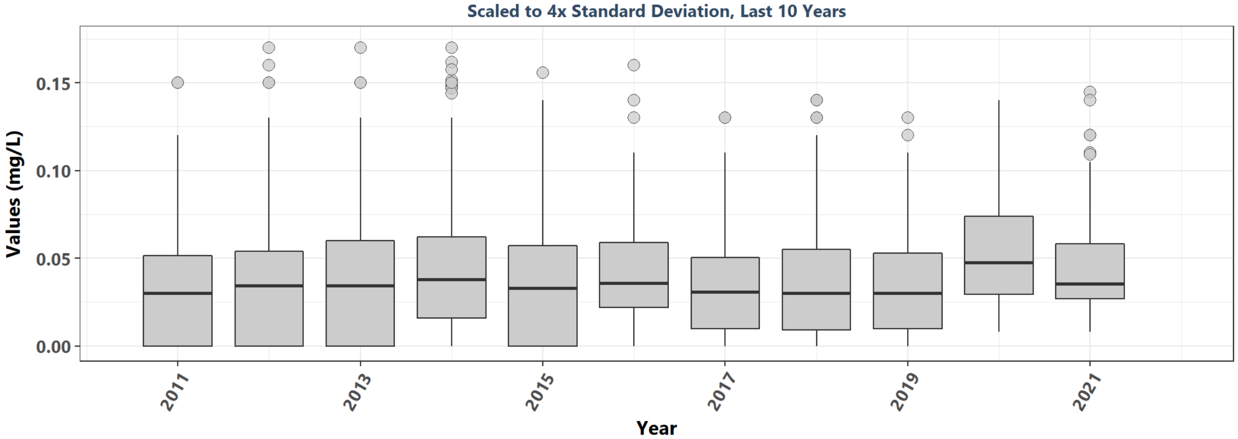
Autoscale

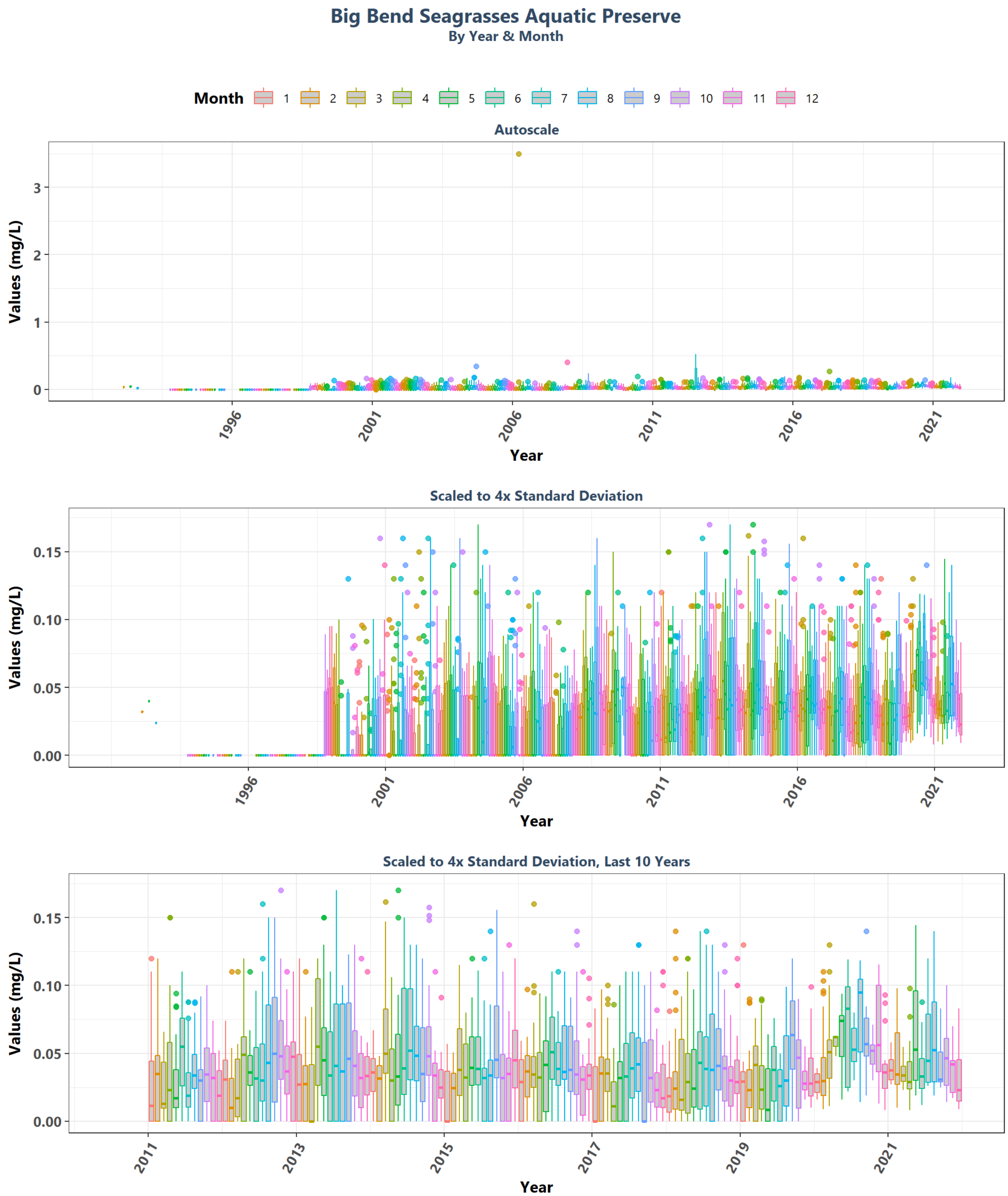


Scaled to 4x Standard Deviation

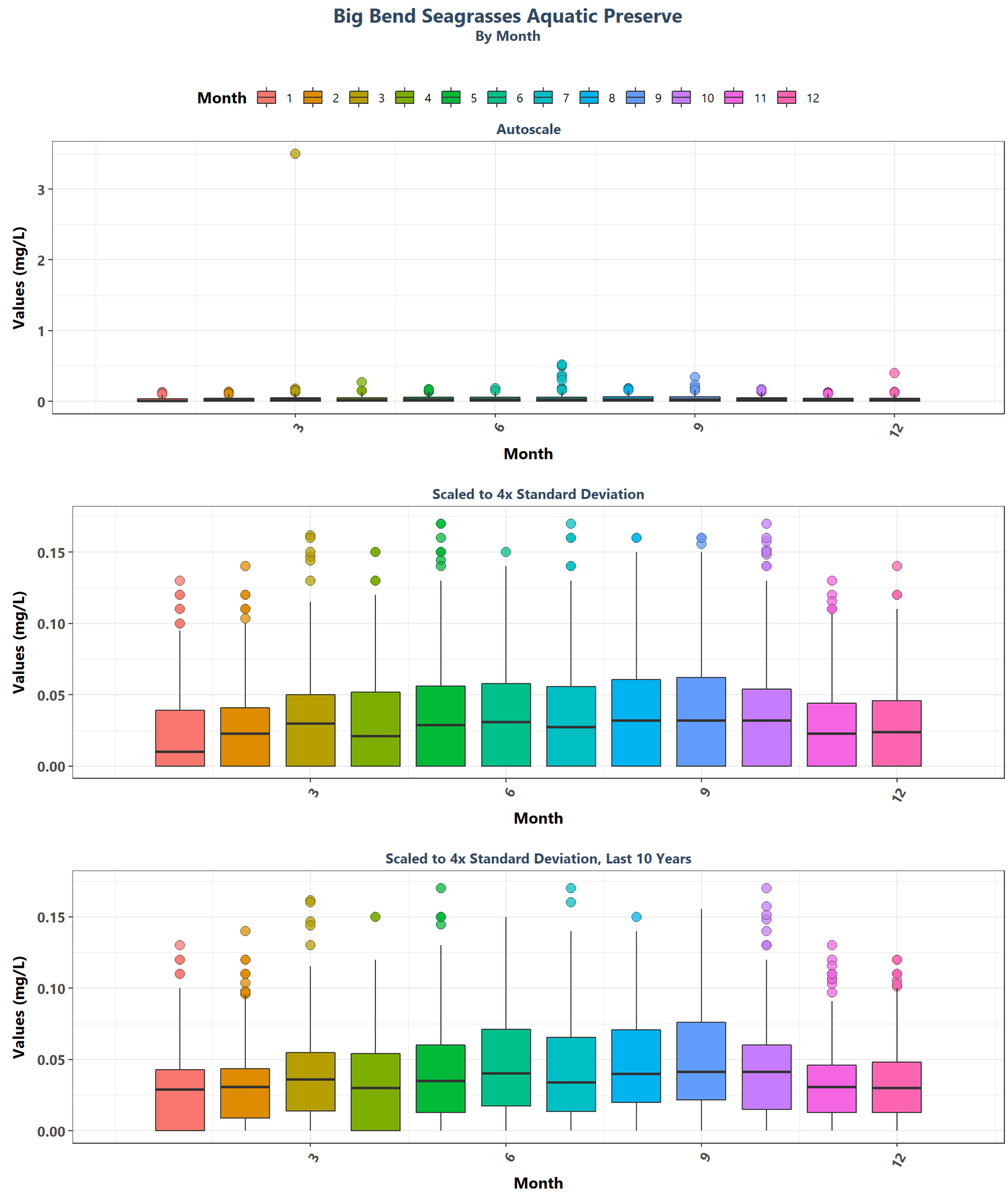


Scaled to 4x Standard Deviation, Last 10 Years

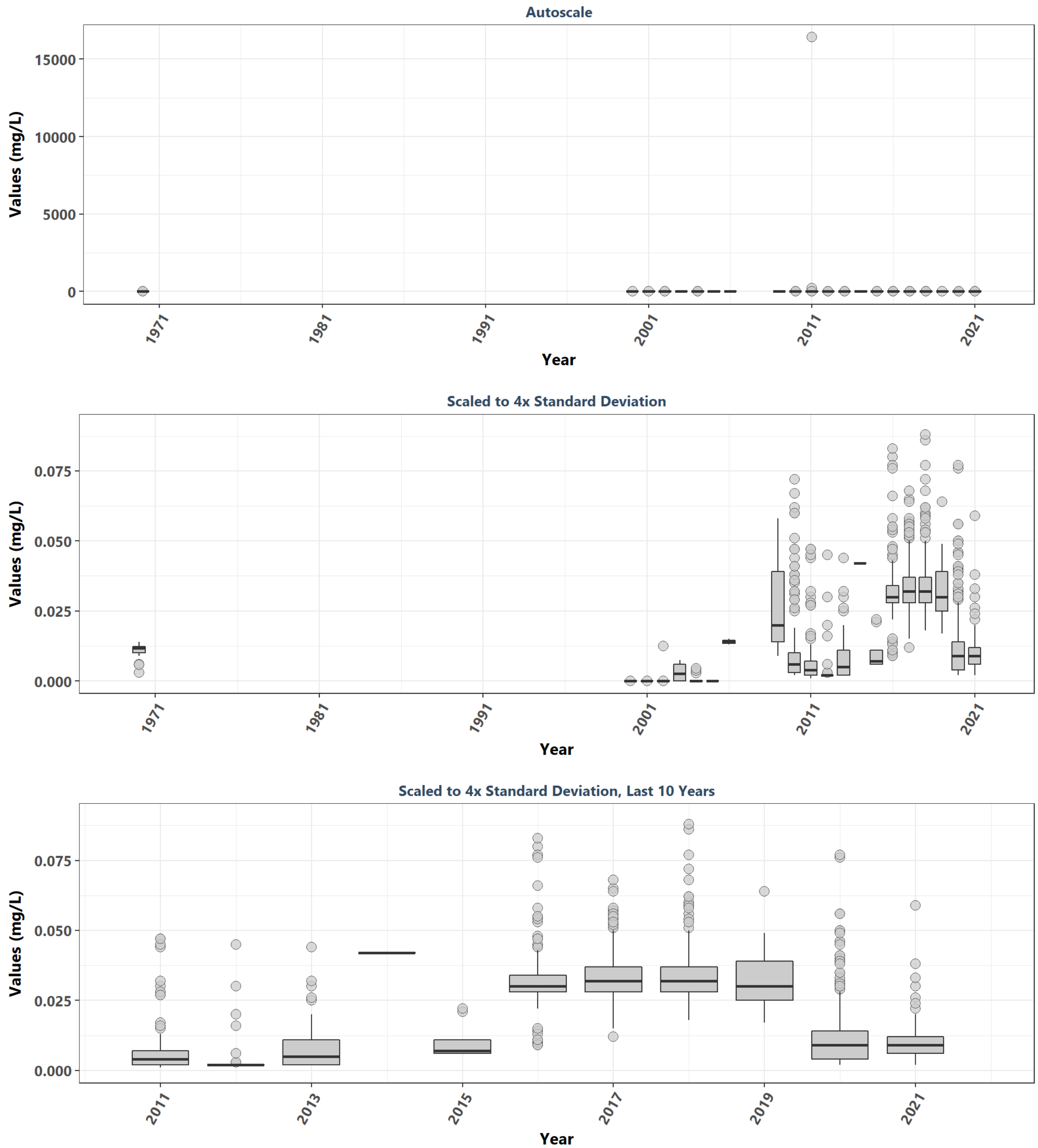




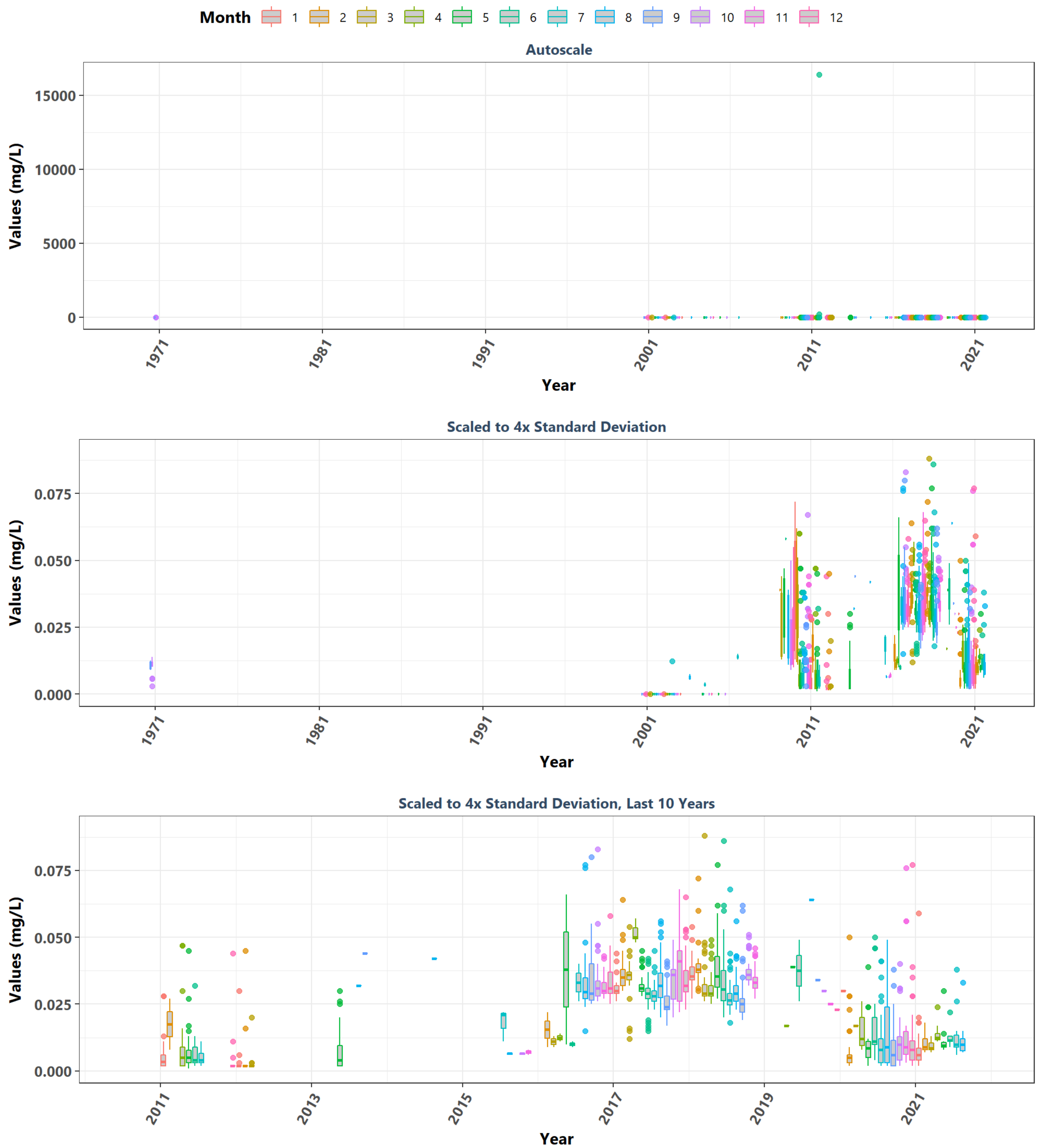
Big Bend Seagrasses Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0
1
2
3
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
2011
2013
2015
2017
2019
2021
Year

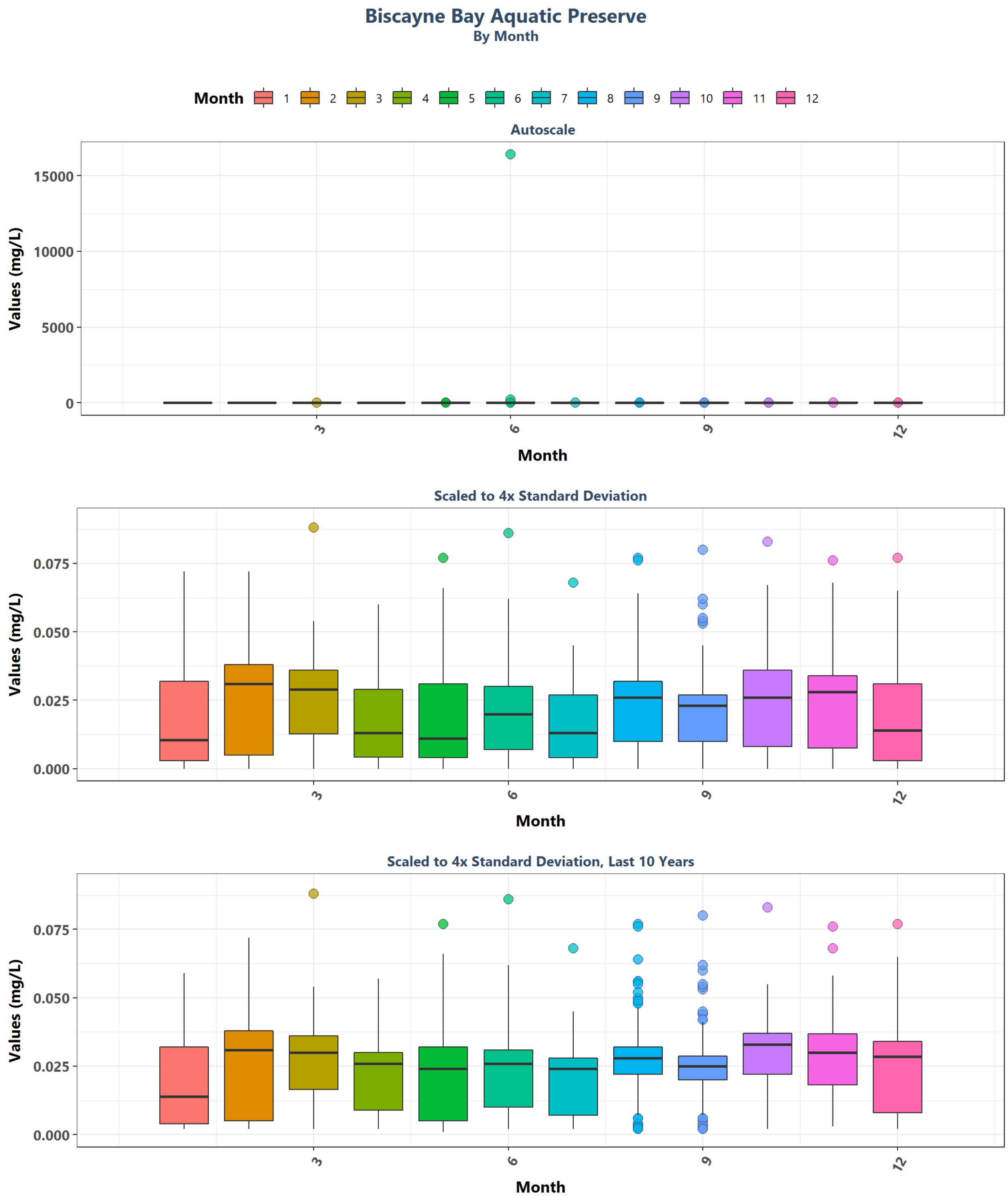

Big Bend Seagrasses Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0
1
2
3
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
3
6
9
12
Month

Biscayne Bay Aquatic Preserve
By Year



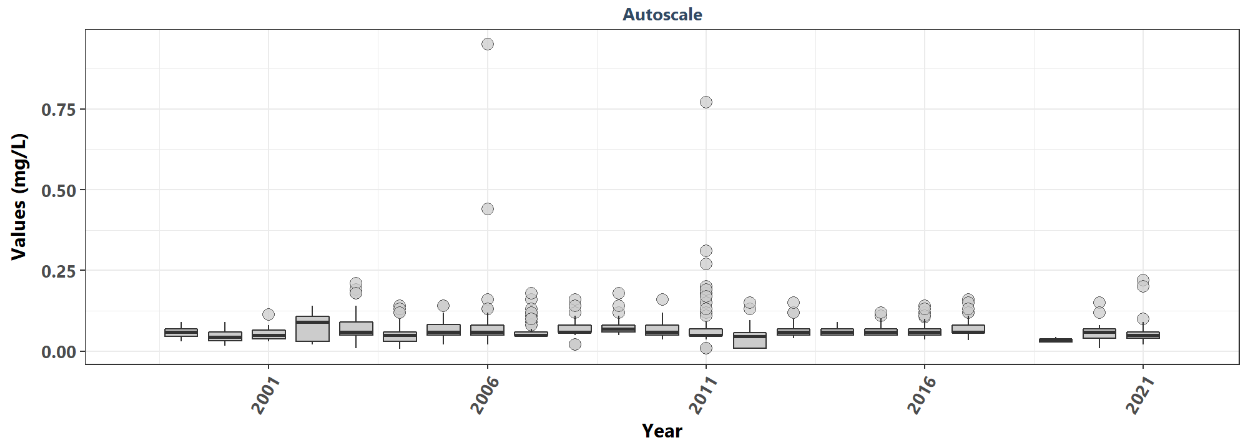
Biscayne Bay Aquatic Preserve
By Year & Month

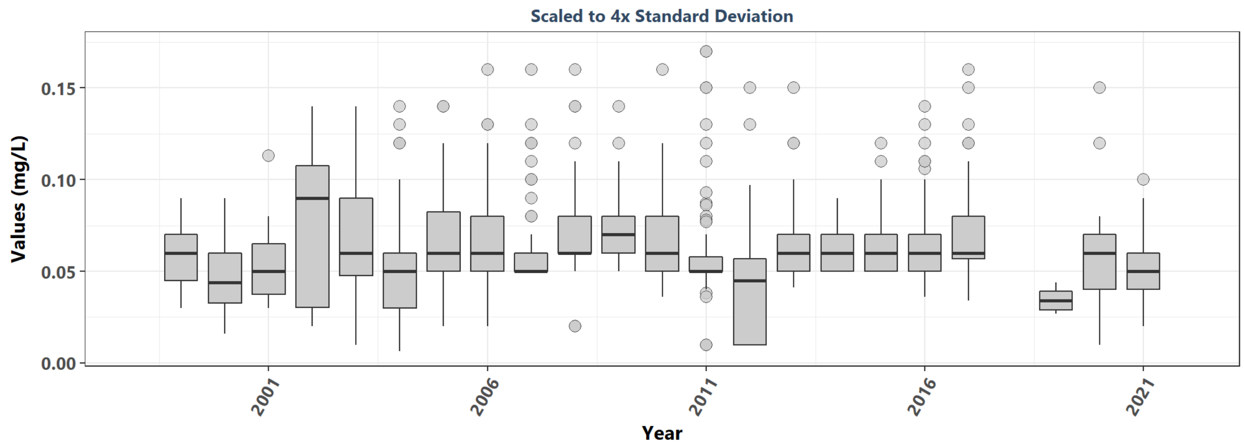


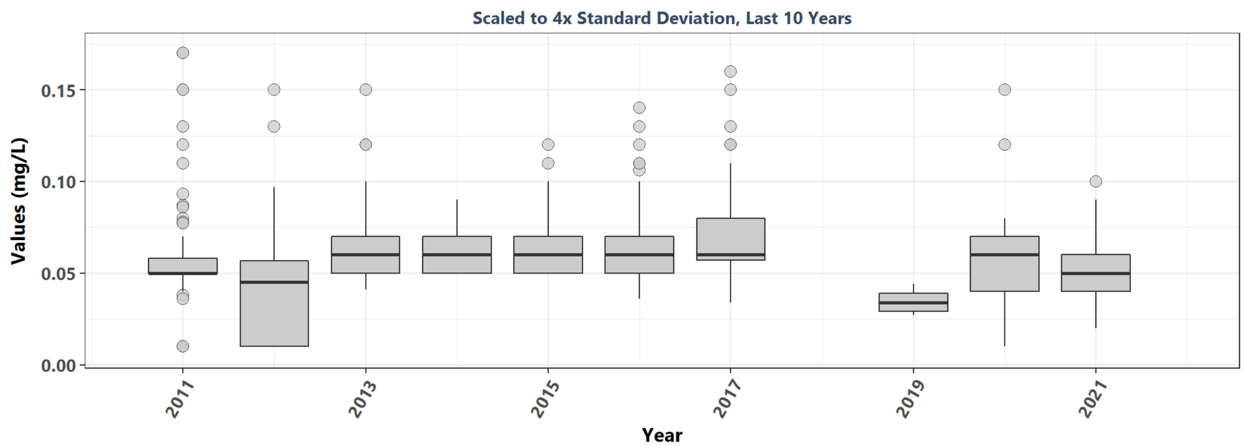


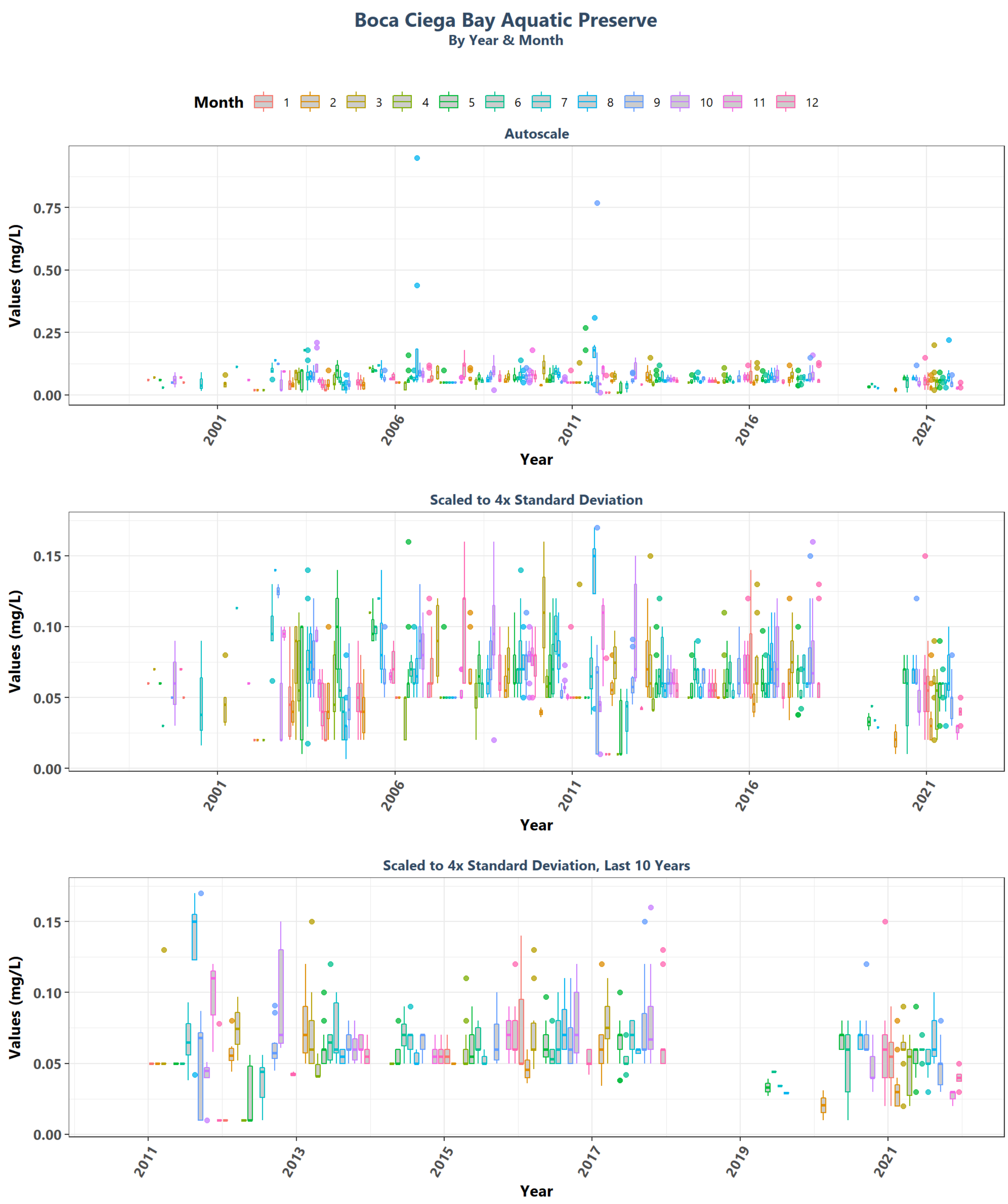
Biscayne Bay Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
15000
10000
5000
0
Values (mg/L)
3
6
9
12
Month
Scaled to 4x Standard Deviation
0.075
0.050
0.025
0.000
Values (mg/L)
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
0.075
0.050
0.025
0.000
Values (mg/L)
3
6
9
12
Month

Boca Ciega Bay Aquatic Preserve
By Year

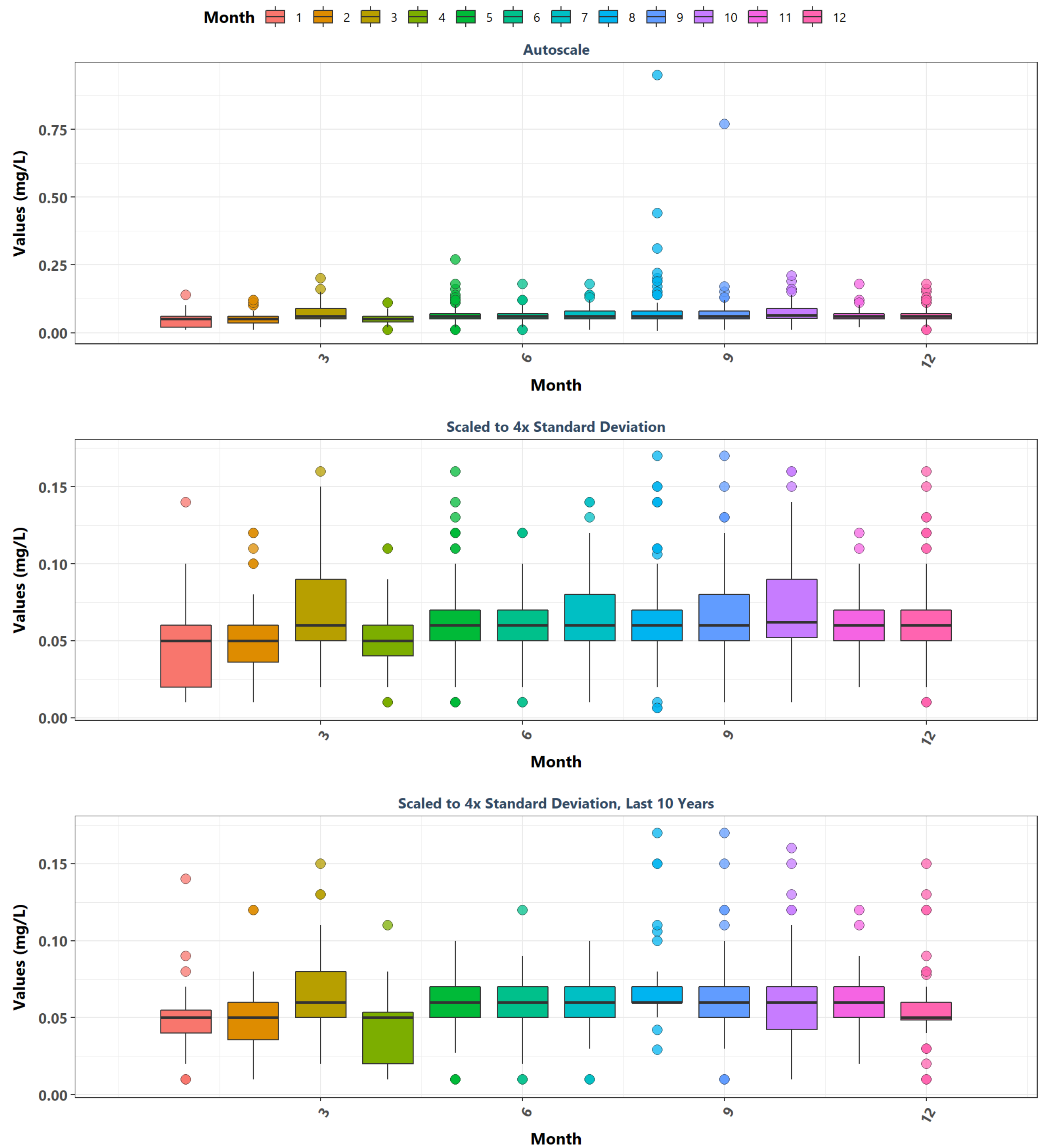






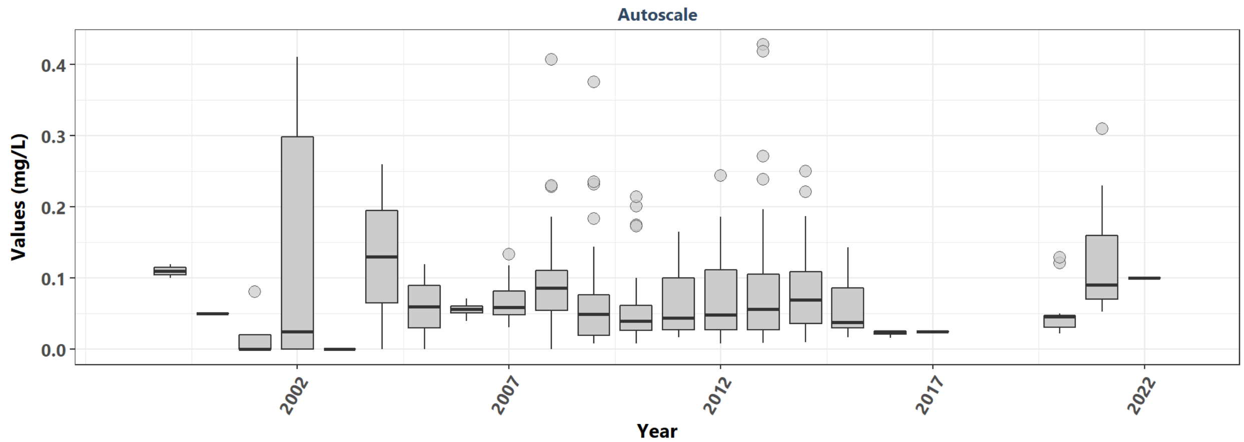

Boca Ciega Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.00
0.25
0.50
0.75
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
0.00
0.05
0.10
0.15
Scaled to 4x Standard Deviation, Last 10 Years
2011
2013
2015
2017
2019
2021

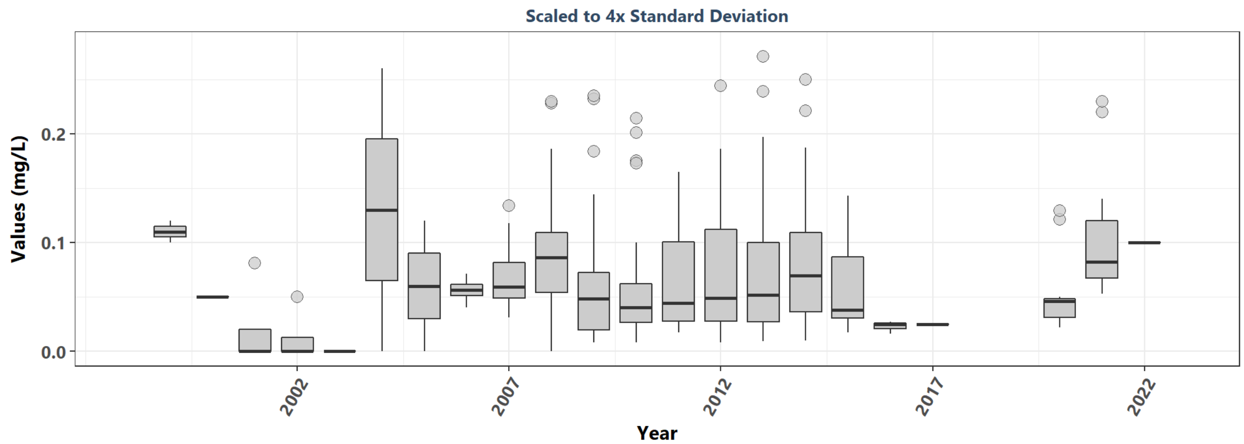
Boca Ciega Bay Aquatic Preserve
By Month

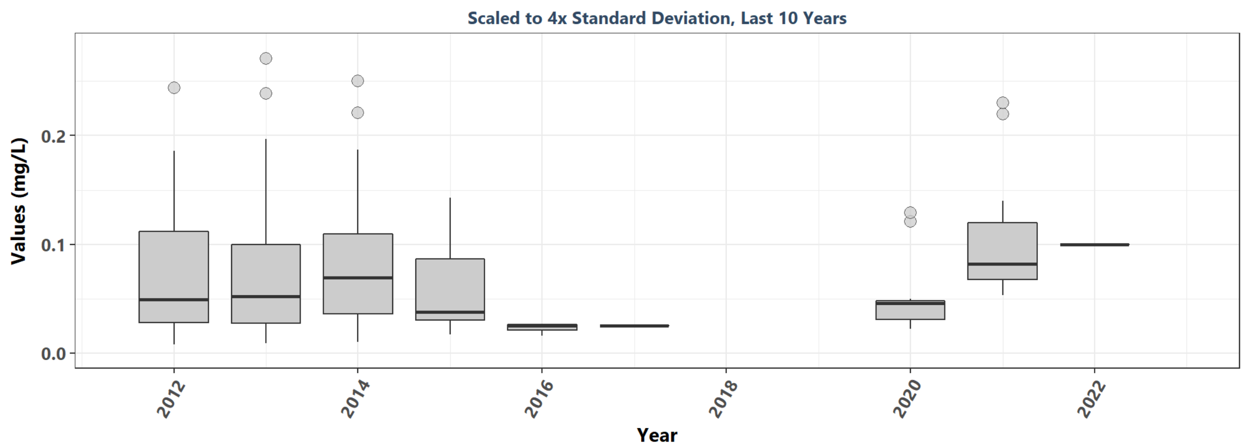


Cape Haze Aquatic Preserve

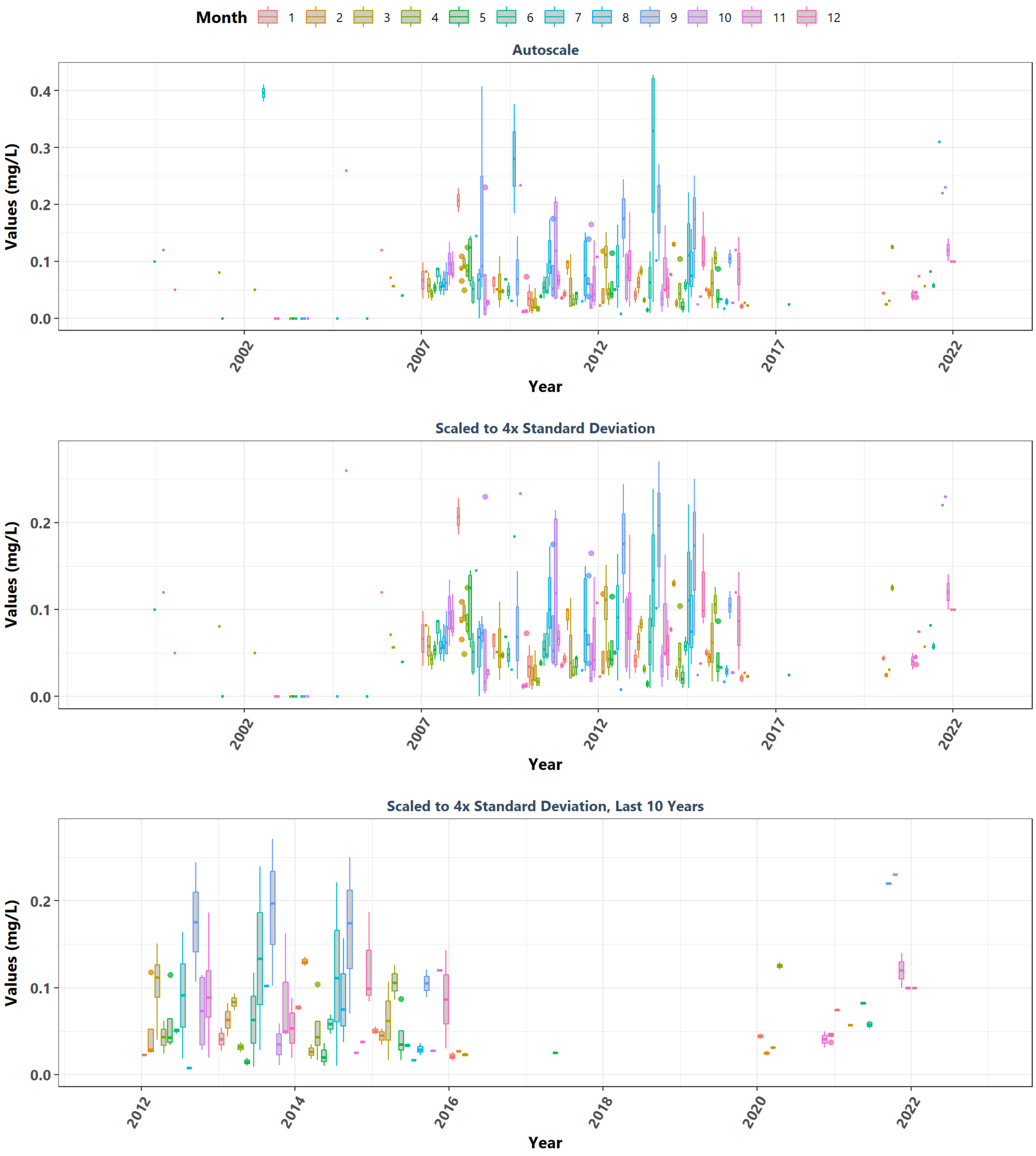
By Year

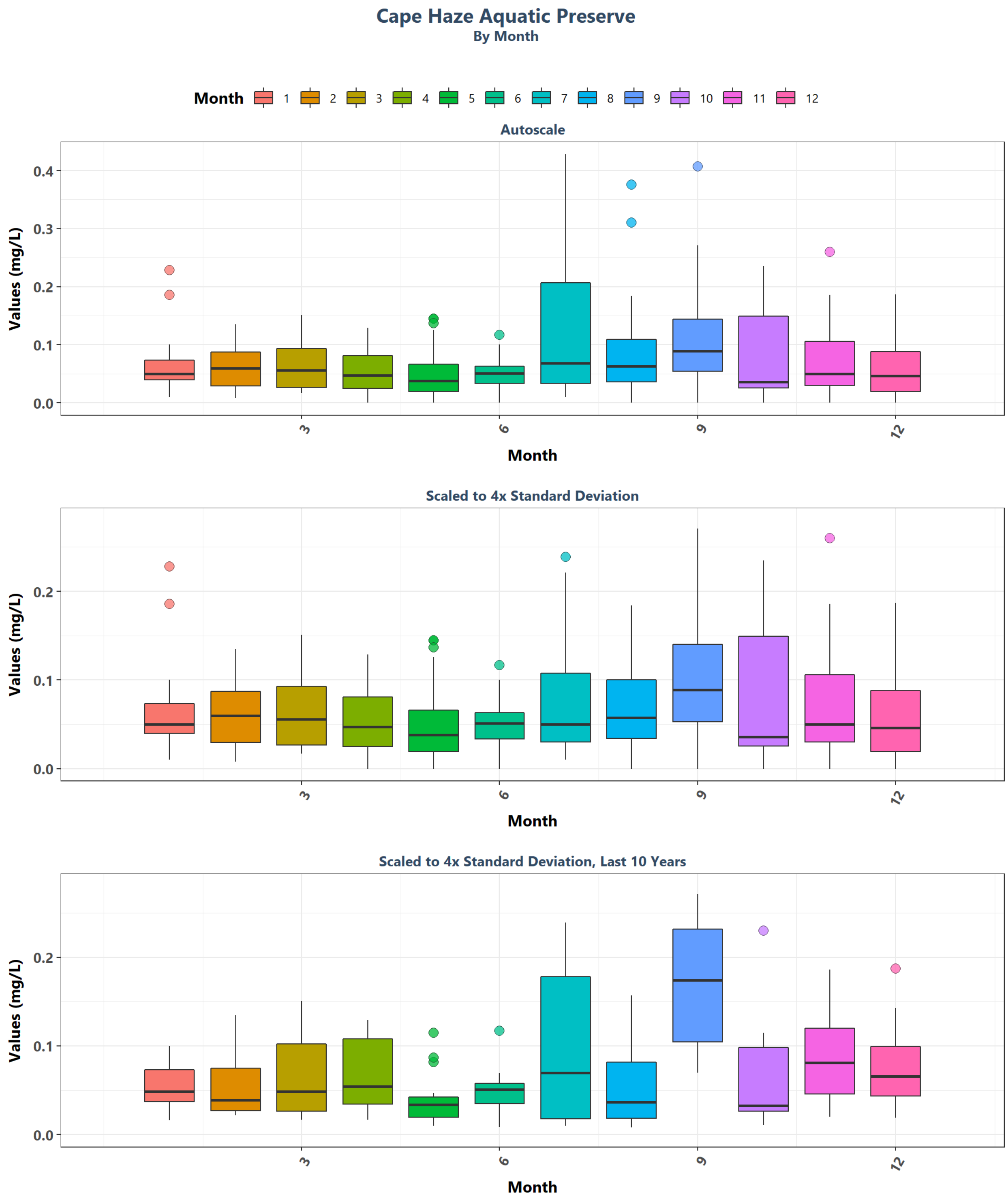






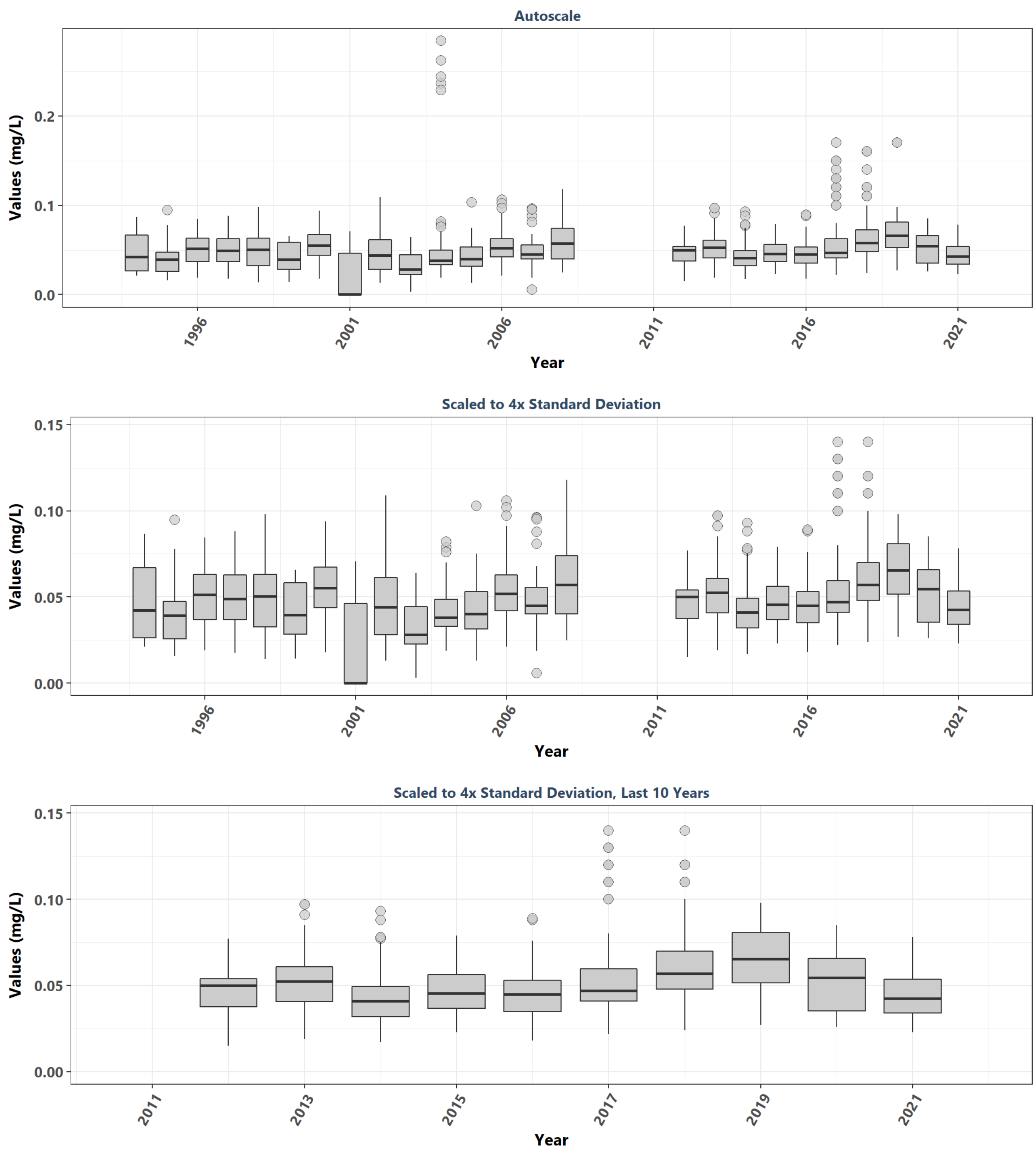
Cape Haze Aquatic Preserve
By Year & Month

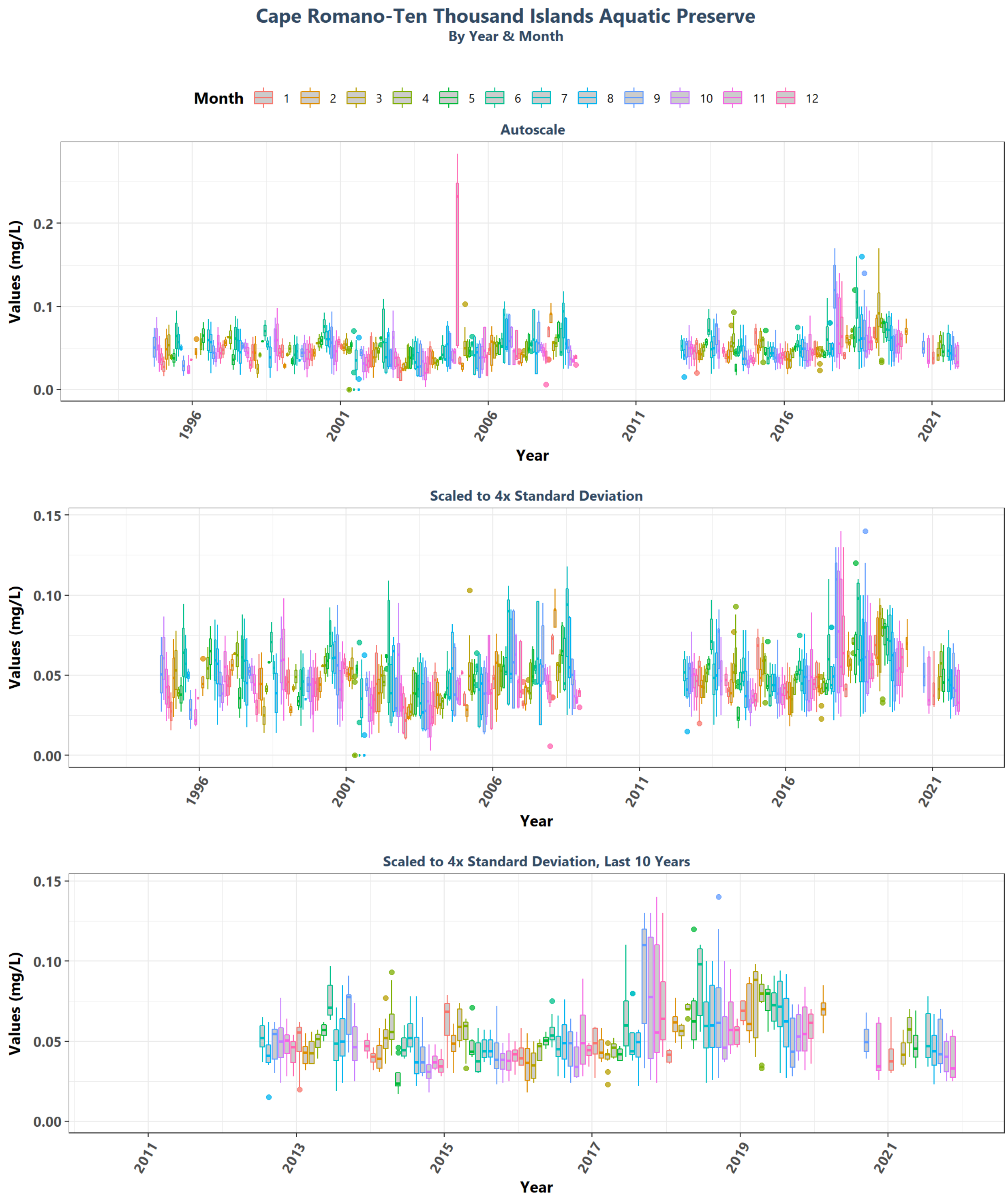



Cape Haze Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.1
0.2
0.3
0.4
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.0
0.1
0.2
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.0
0.1
0.2
3
6
9
12
Month

Cape Romano-Ten Thousand Islands Aquatic Preserve

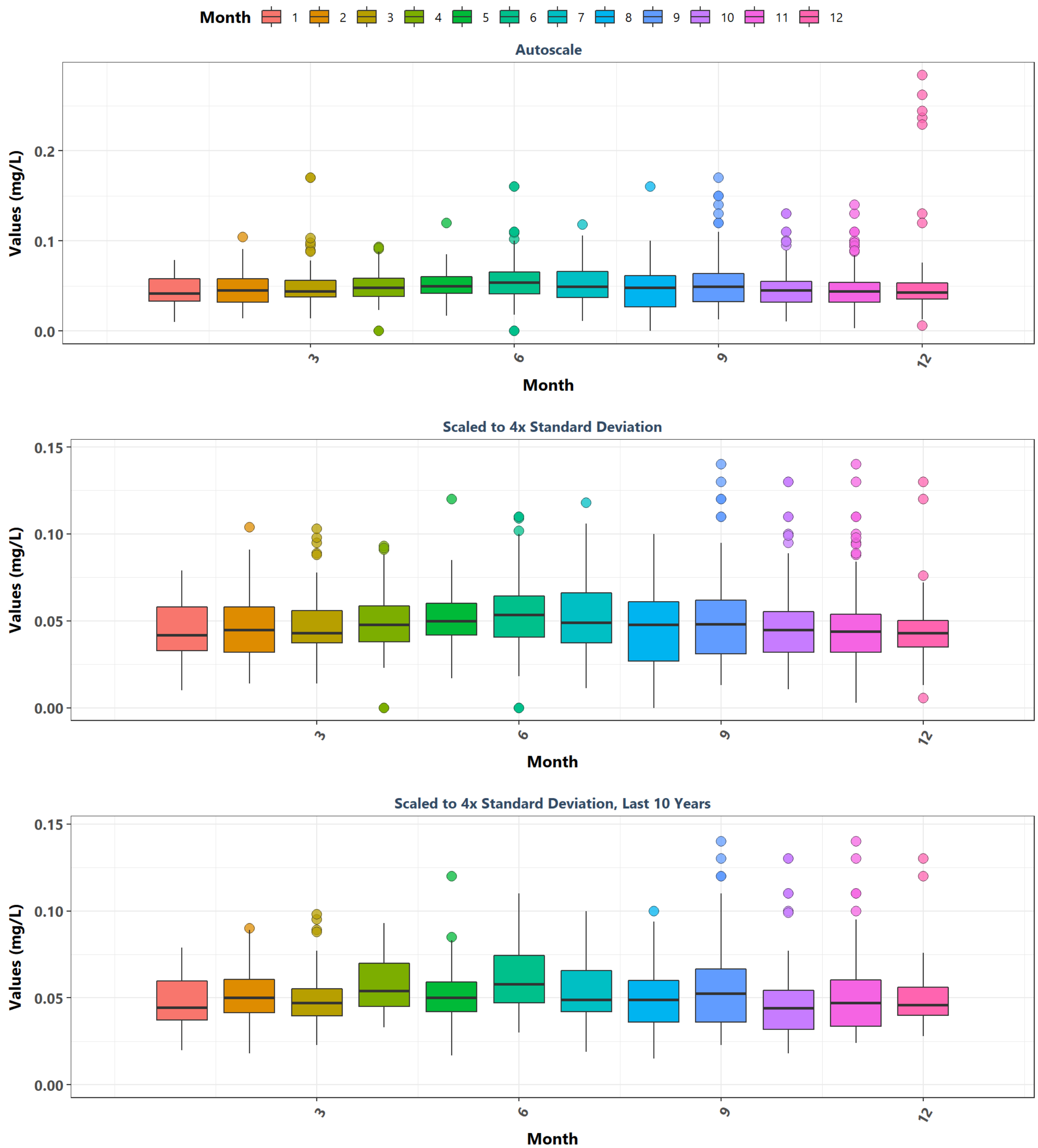
By Year




Cape Romano-Ten Thousand Islands Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.1
0.2
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
2011
2013
2015
2017
2019
2021
Year

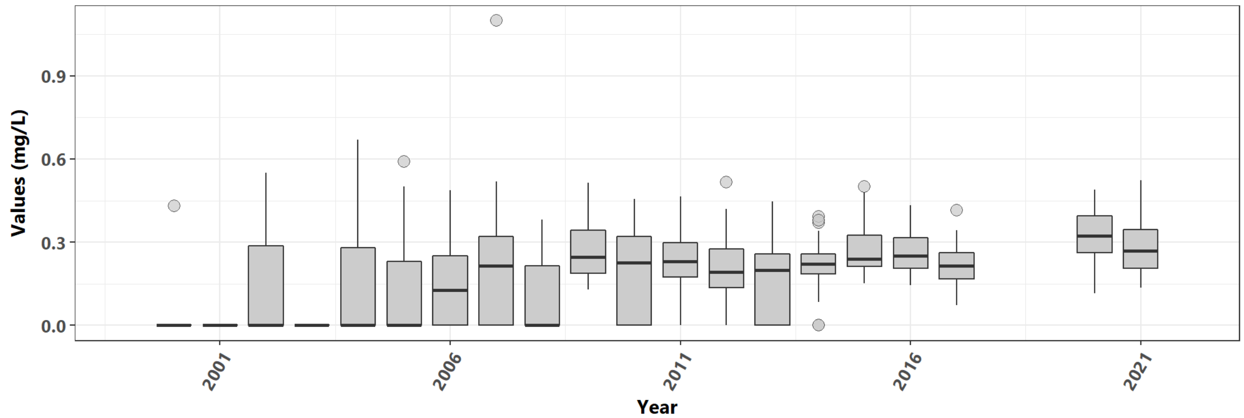
Cape Romano-Ten Thousand Islands Aquatic Preserve

By Month

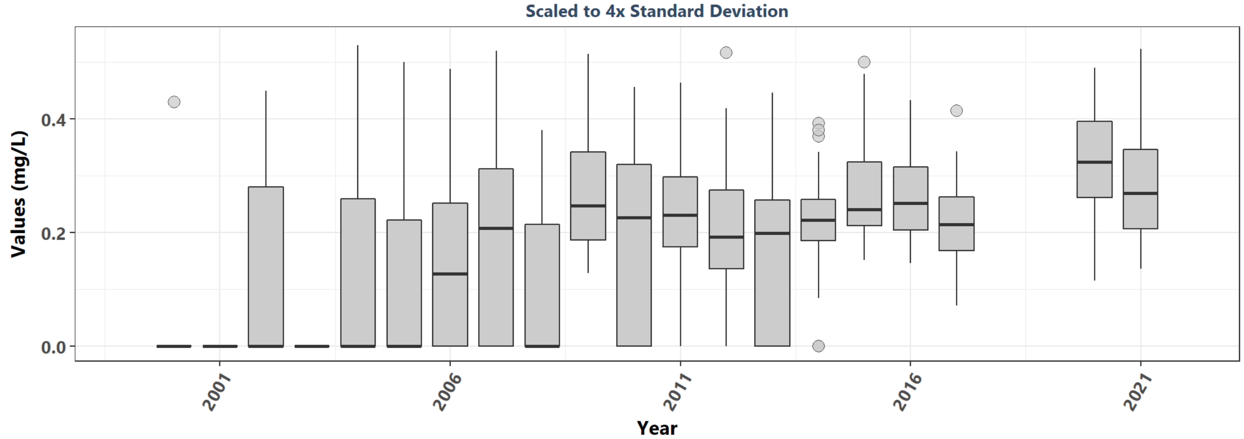


Cockroach Bay Aquatic Preserve
By Year

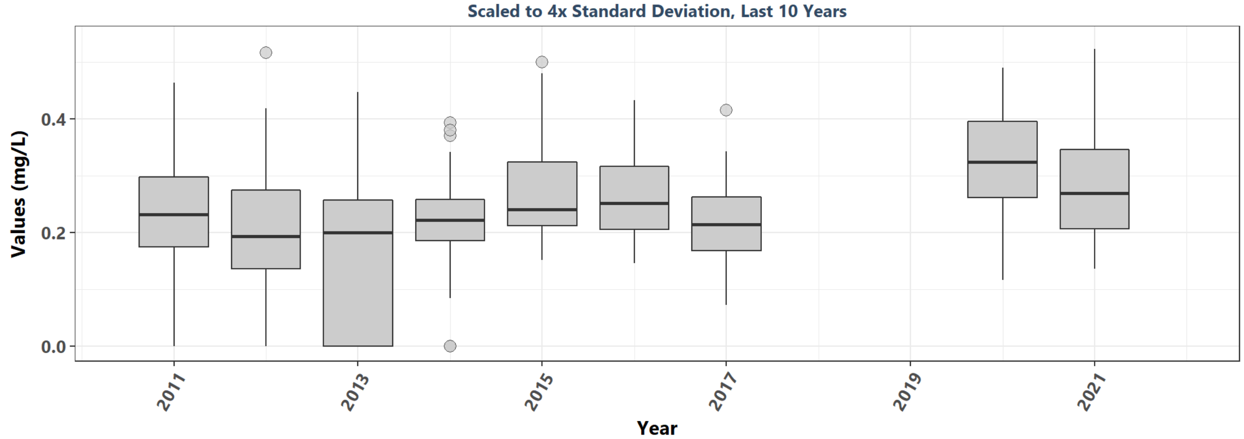
Autoscale

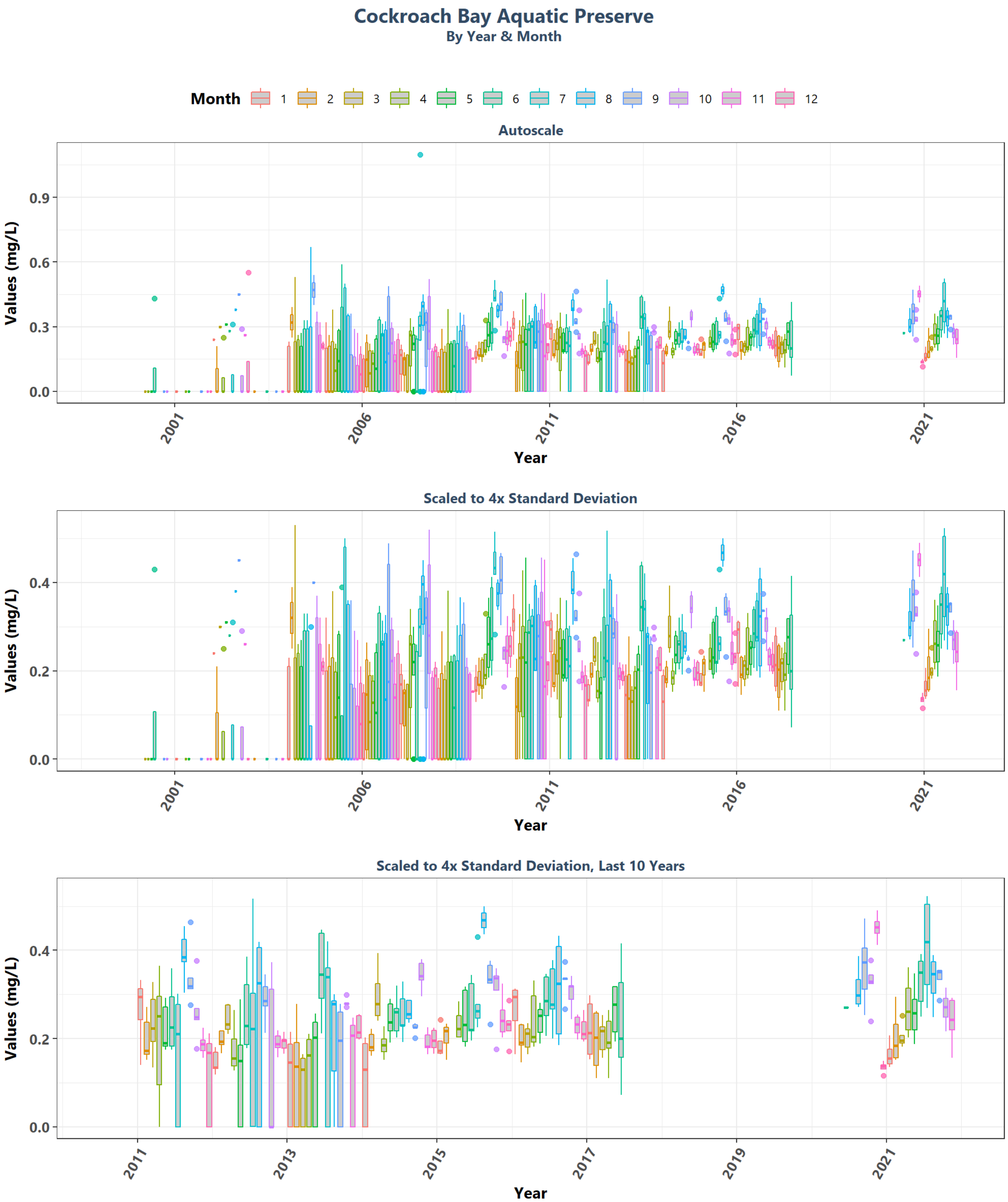


Scaled to 4x Standard Deviation



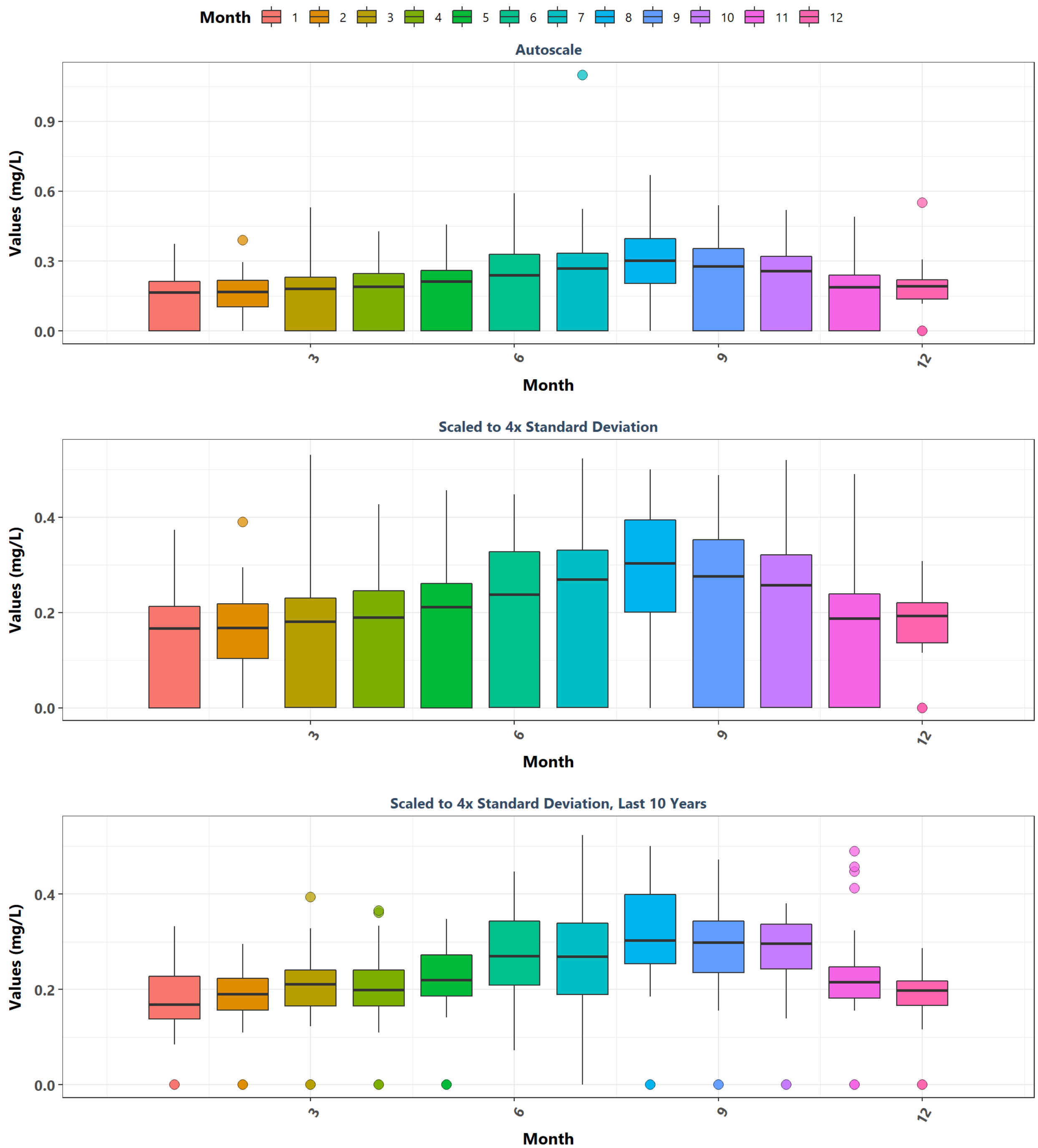
Scaled to 4x Standard Deviation, Last 10 Years



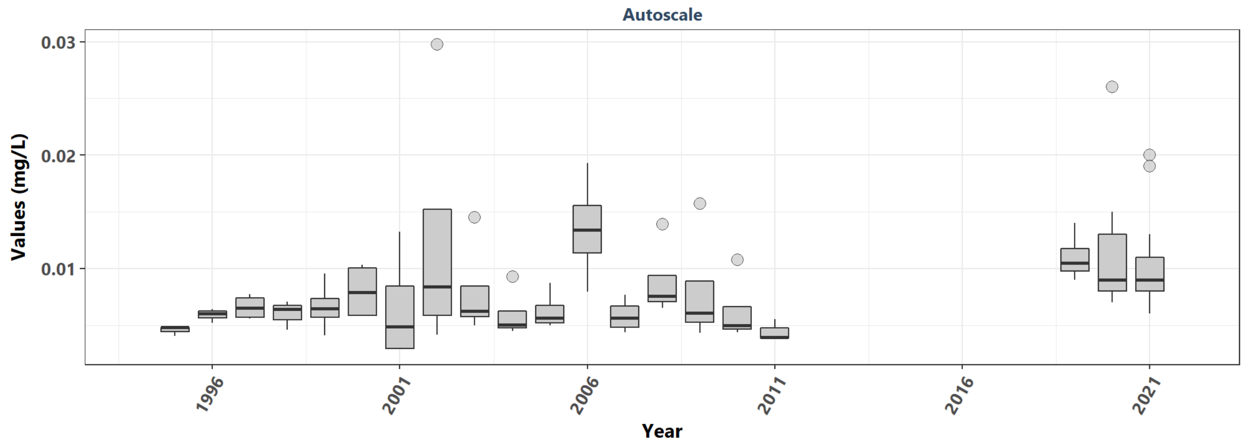

Cockroach Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.3
0.6
0.9
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.0
0.2
0.4
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.0
0.2
0.4
2011
2013
2015
2017
2019
2021
Year

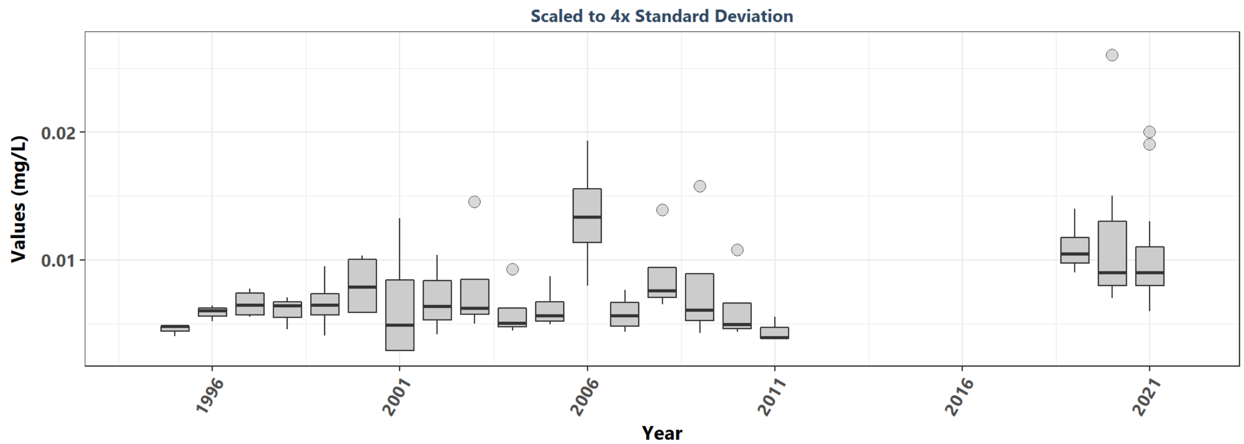
Cockroach Bay Aquatic Preserve

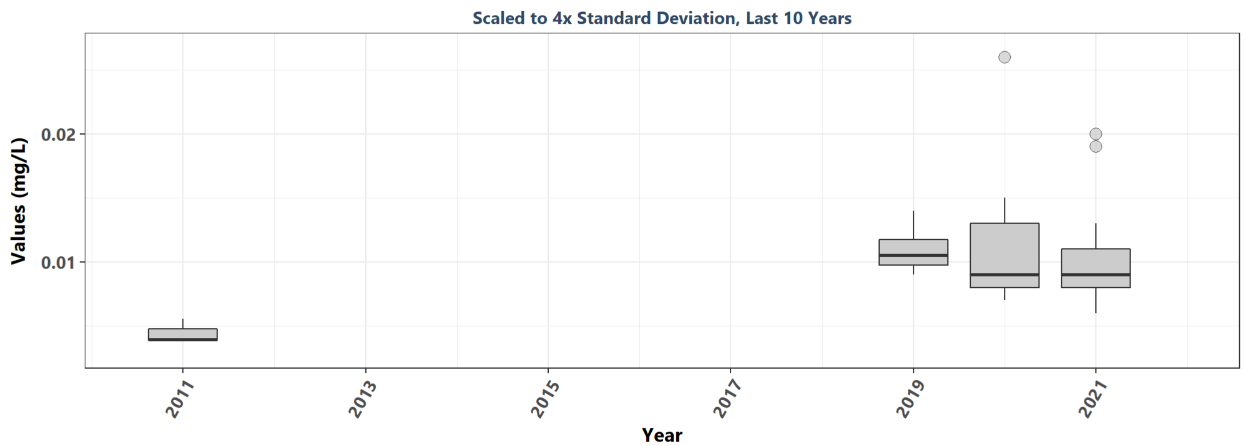
By Month



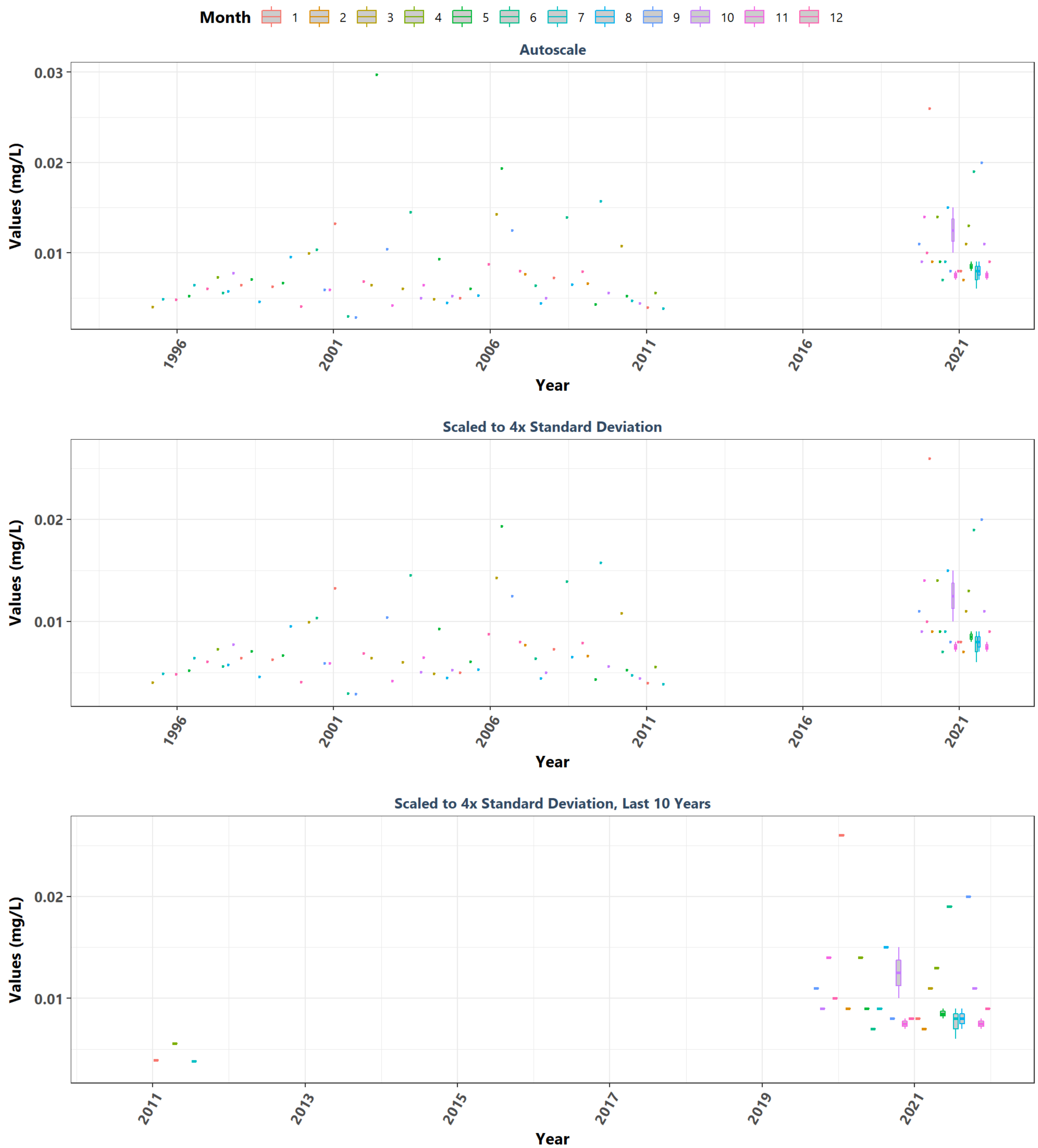
Coupon Bight Aquatic Preserve
By Year

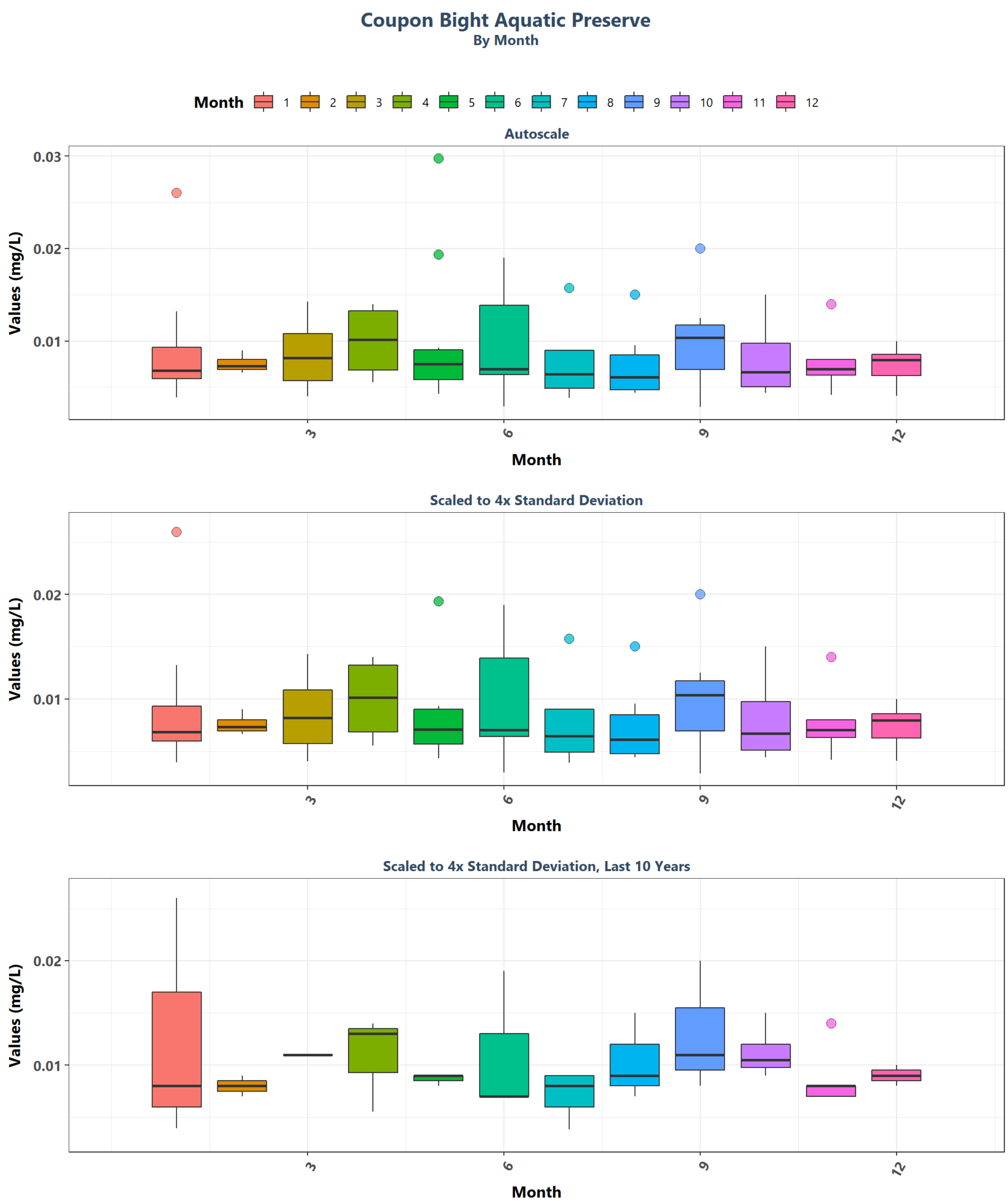






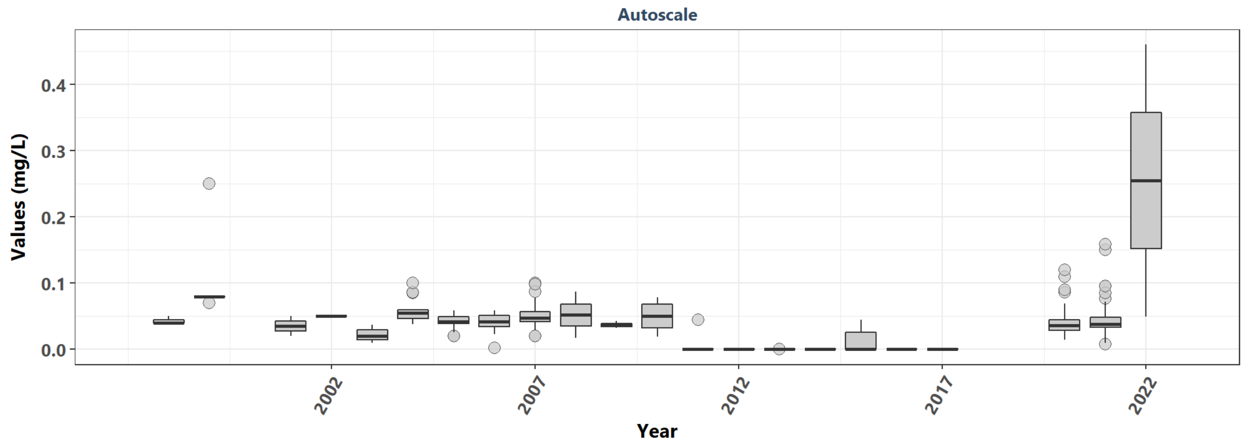
Coupon Bight Aquatic Preserve
By Year & Month

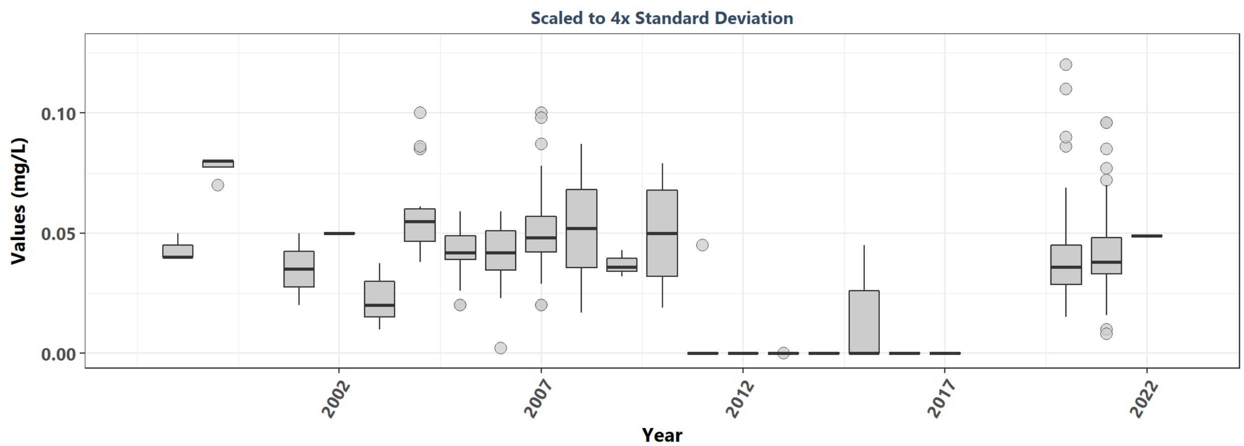


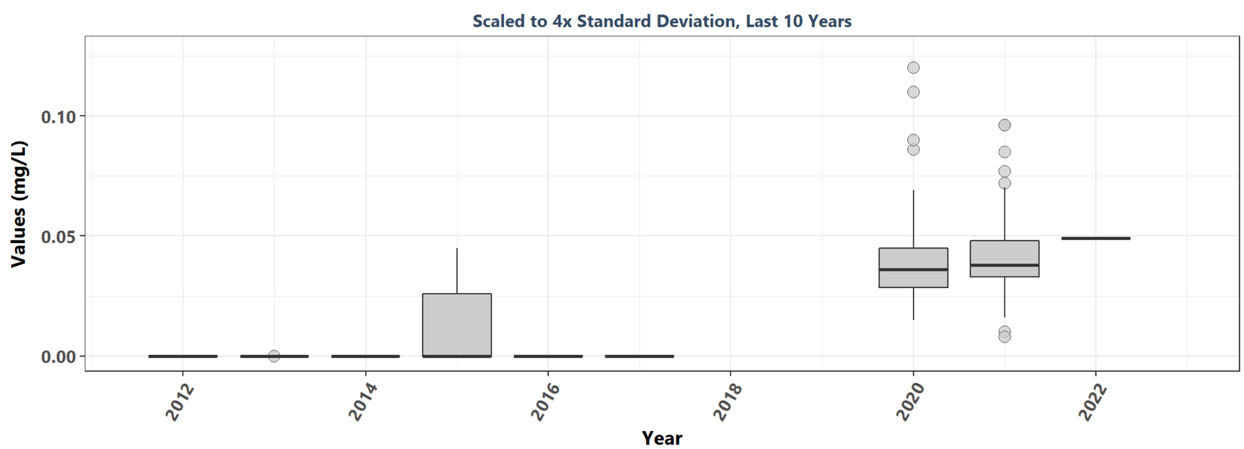


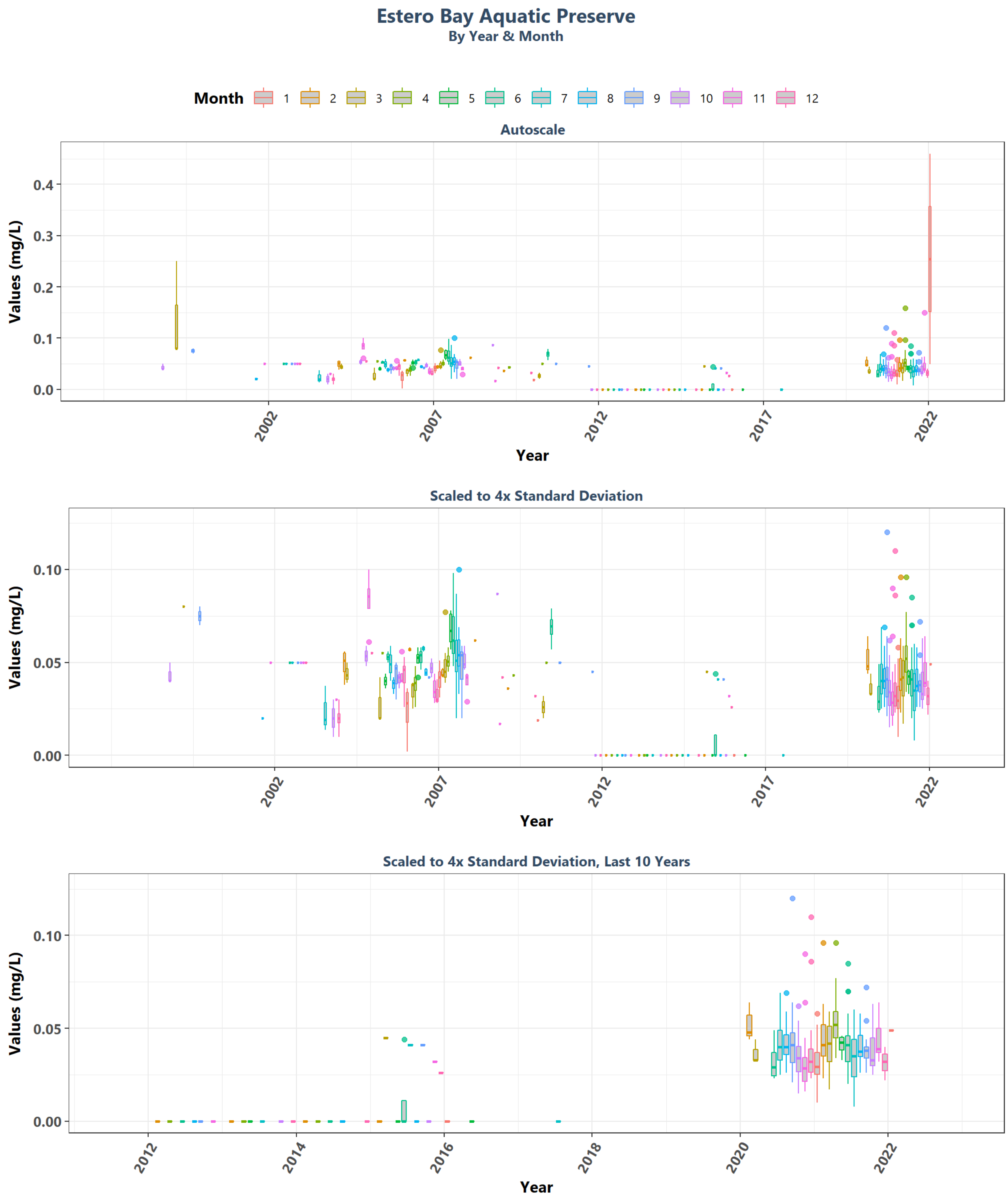
Coupon Bight Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.03
0.02
0.01
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.02
0.01
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.02
0.01
3
6
9
12
Month

Estero Bay Aquatic Preserve
By Year



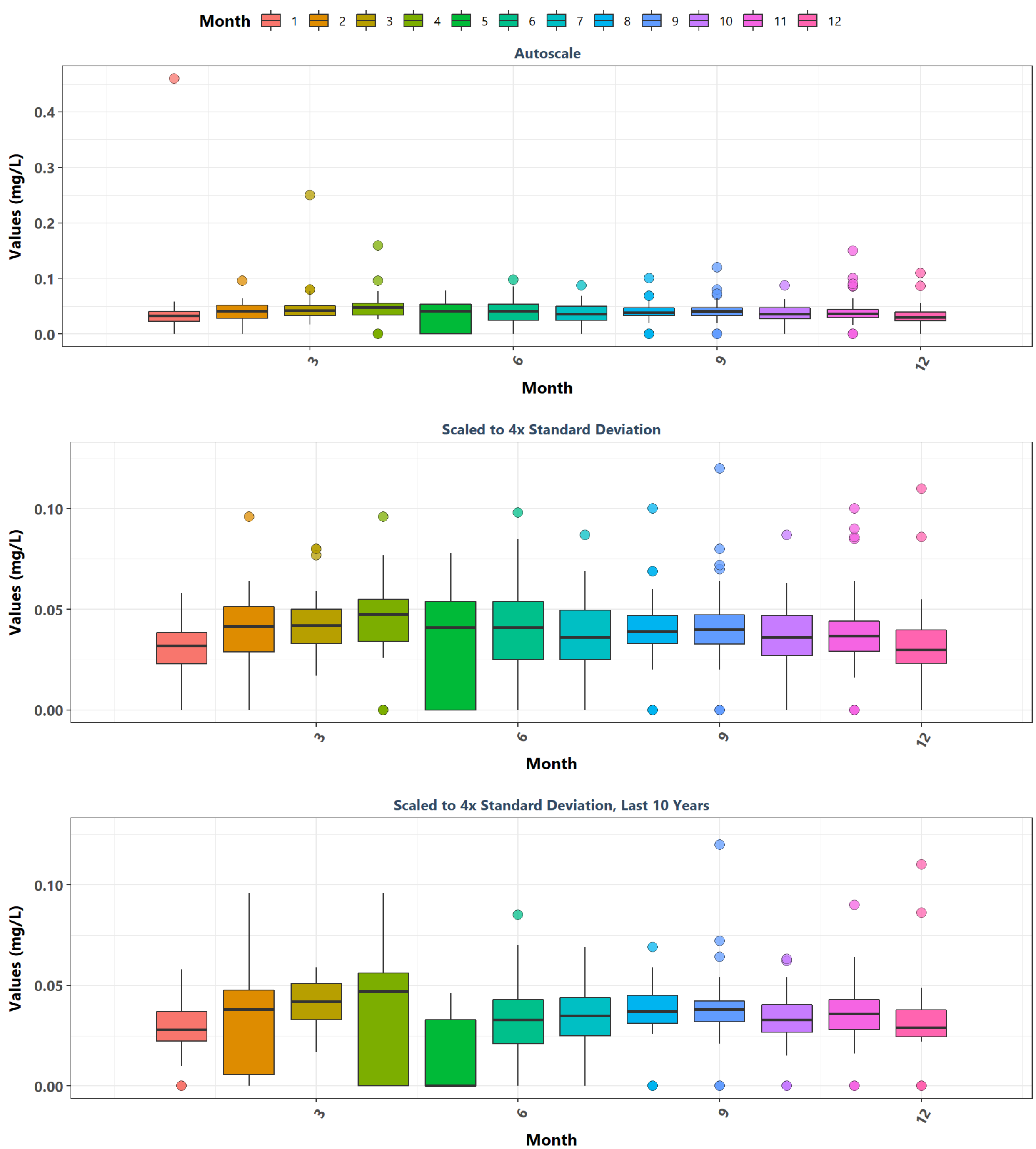






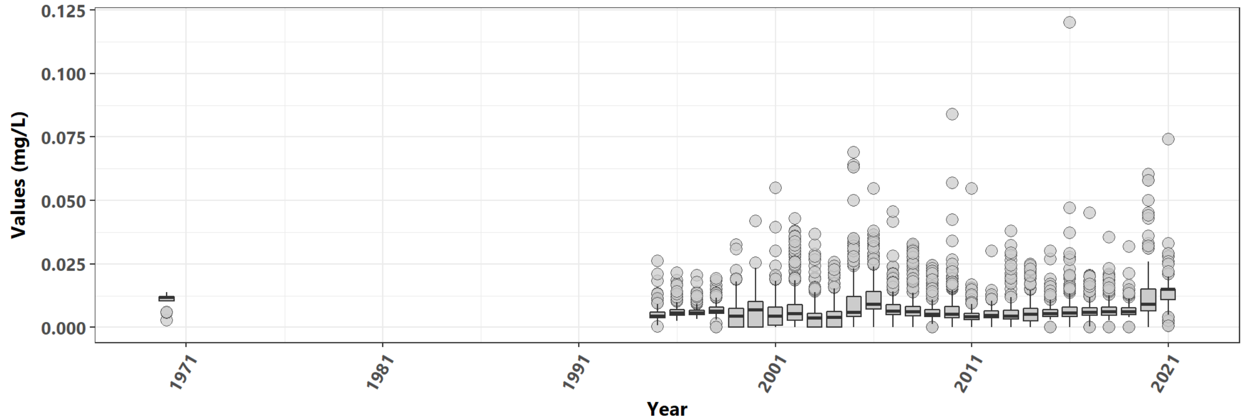
Estero Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
Year
Scaled to 4x Standard Deviation
Scaled to 4x Standard Deviation, Last 10 Years

Estero Bay Aquatic Preserve
By Month

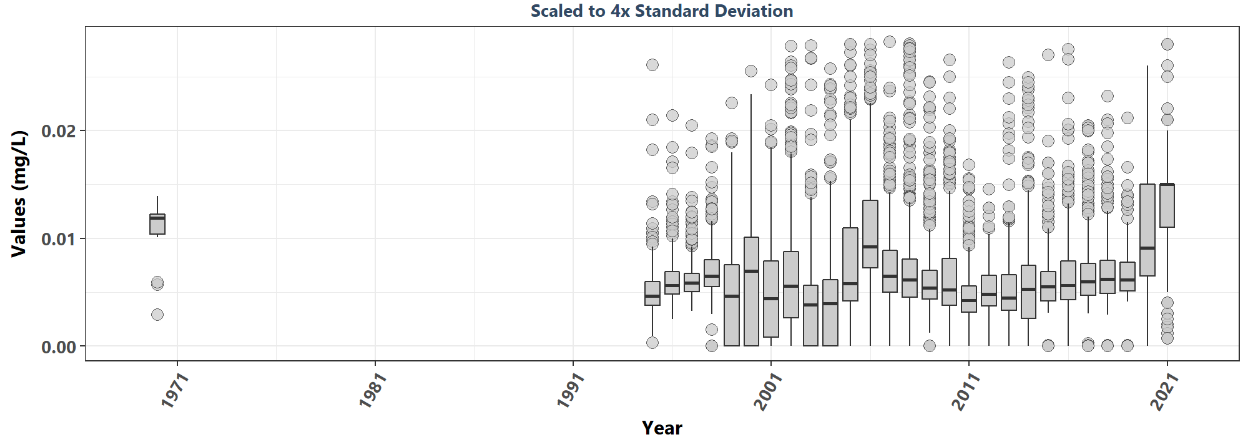


Florida Keys National Marine Sanctuary
By Year

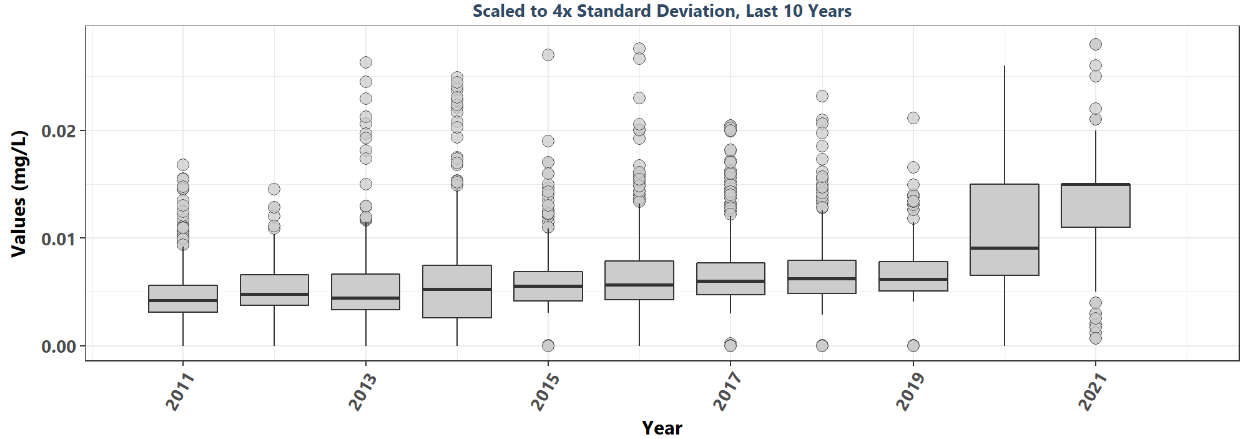
Autoscale

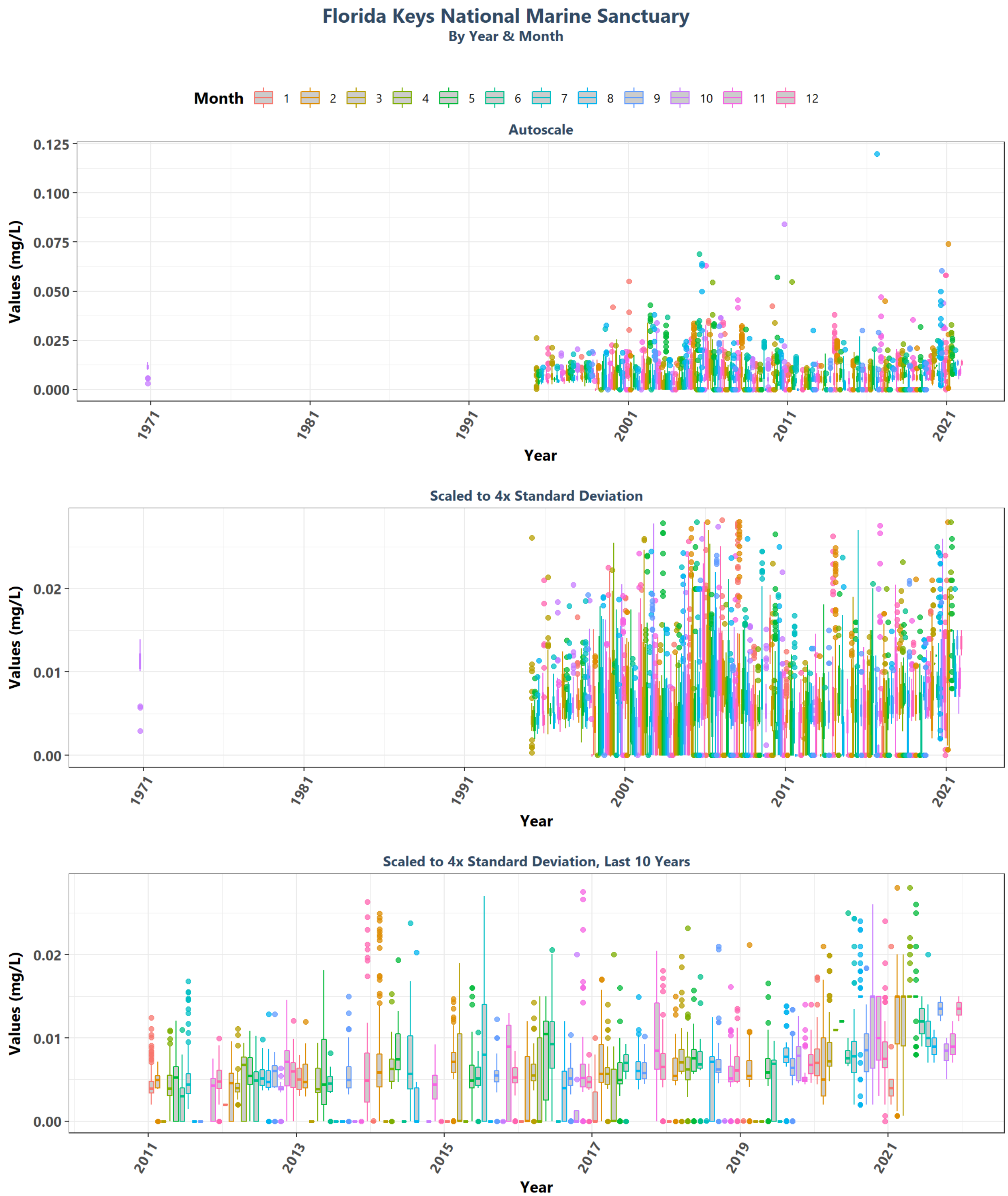


Scaled to 4x Standard Deviation

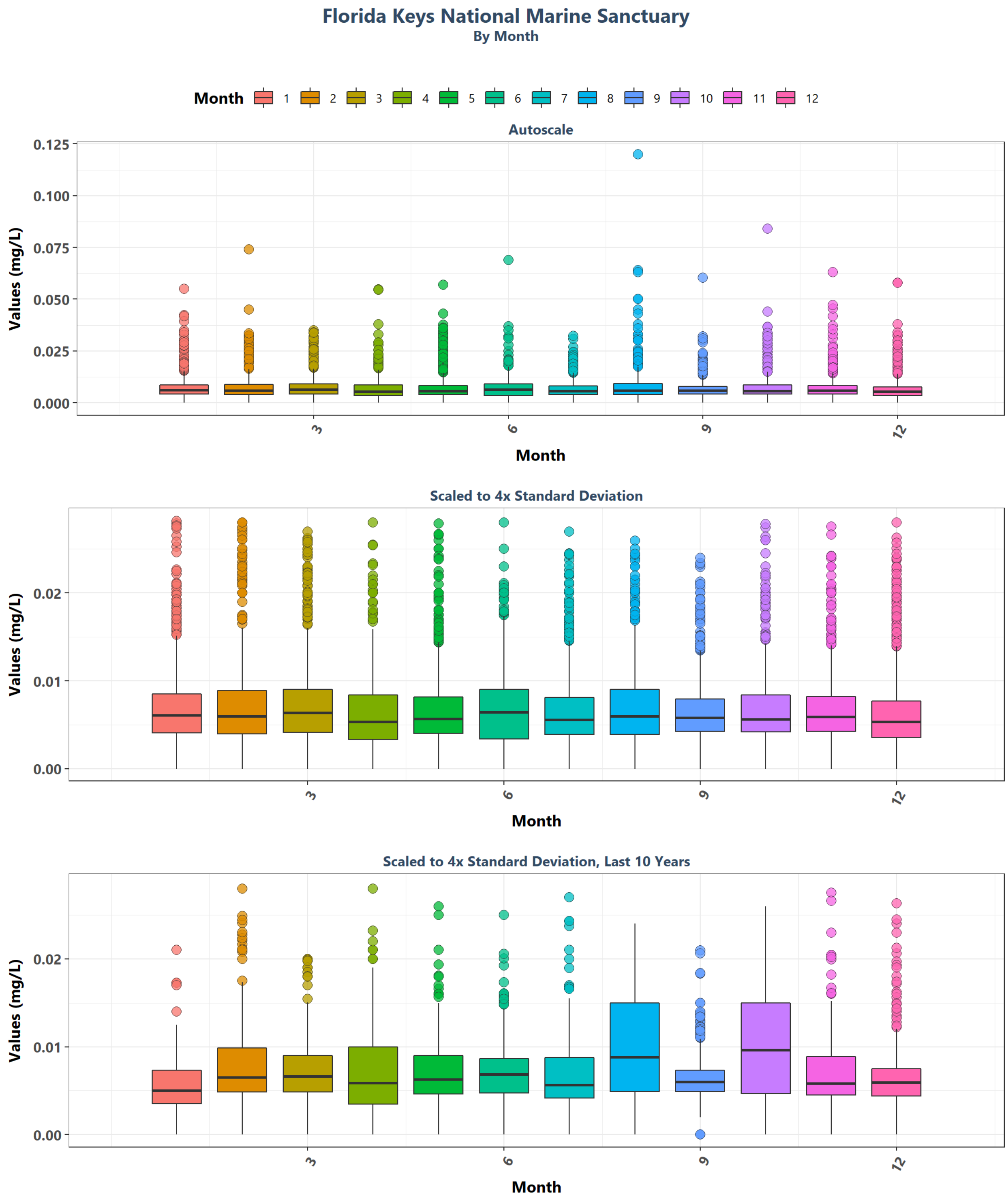


Scaled to 4x Standard Deviation, Last 10 Years



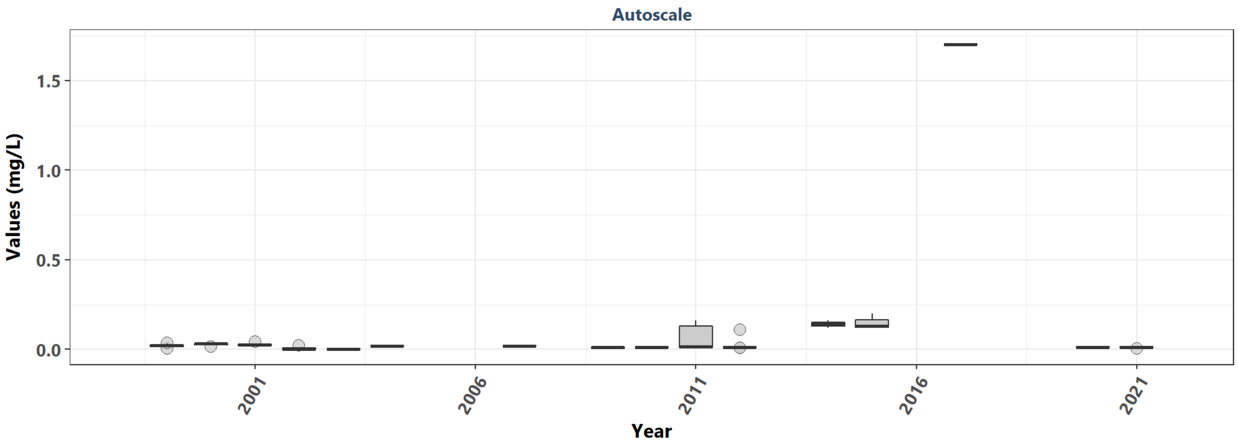


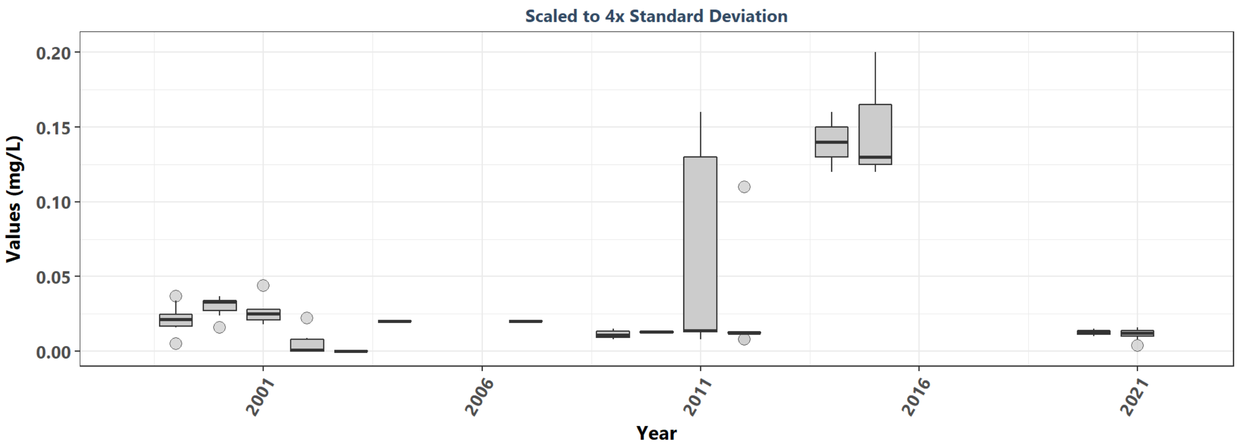
Florida Keys National Marine Sanctuary
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.125
0.100
0.075
0.050
0.025
0.000
1971 1981 1991 2001 2011 2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.02
0.01
0.00
1971 1981 1991 2001 2011 2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.02
0.01
0.00
2011 2013 2015 2017 2019 2021
Year

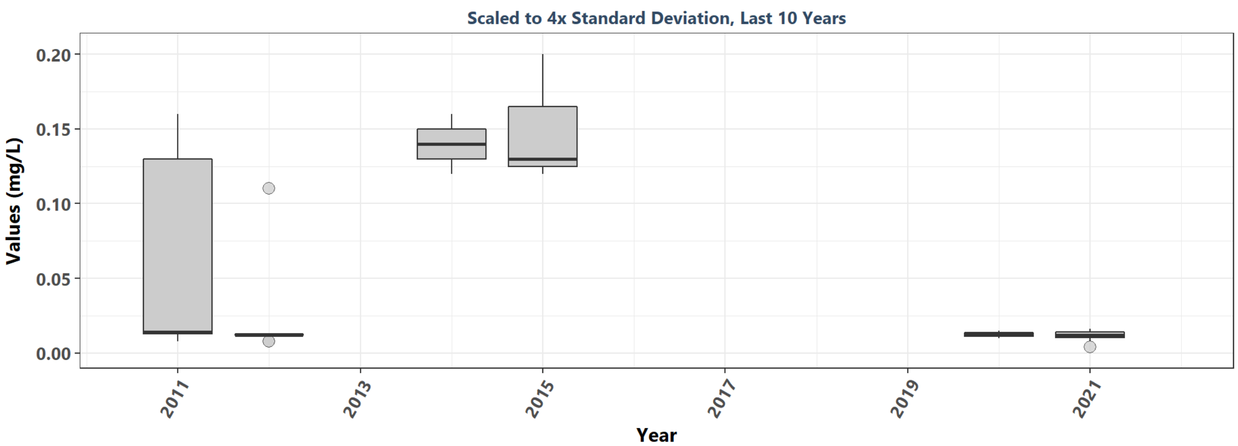

Florida Keys National Marine Sanctuary
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.125
0.100
0.075
0.050
0.025
0.000
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.02
0.01
0.00
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.02
0.01
0.00
3
6
9
12
Month

Fort Pickens State Park Aquatic Preserve

By Year

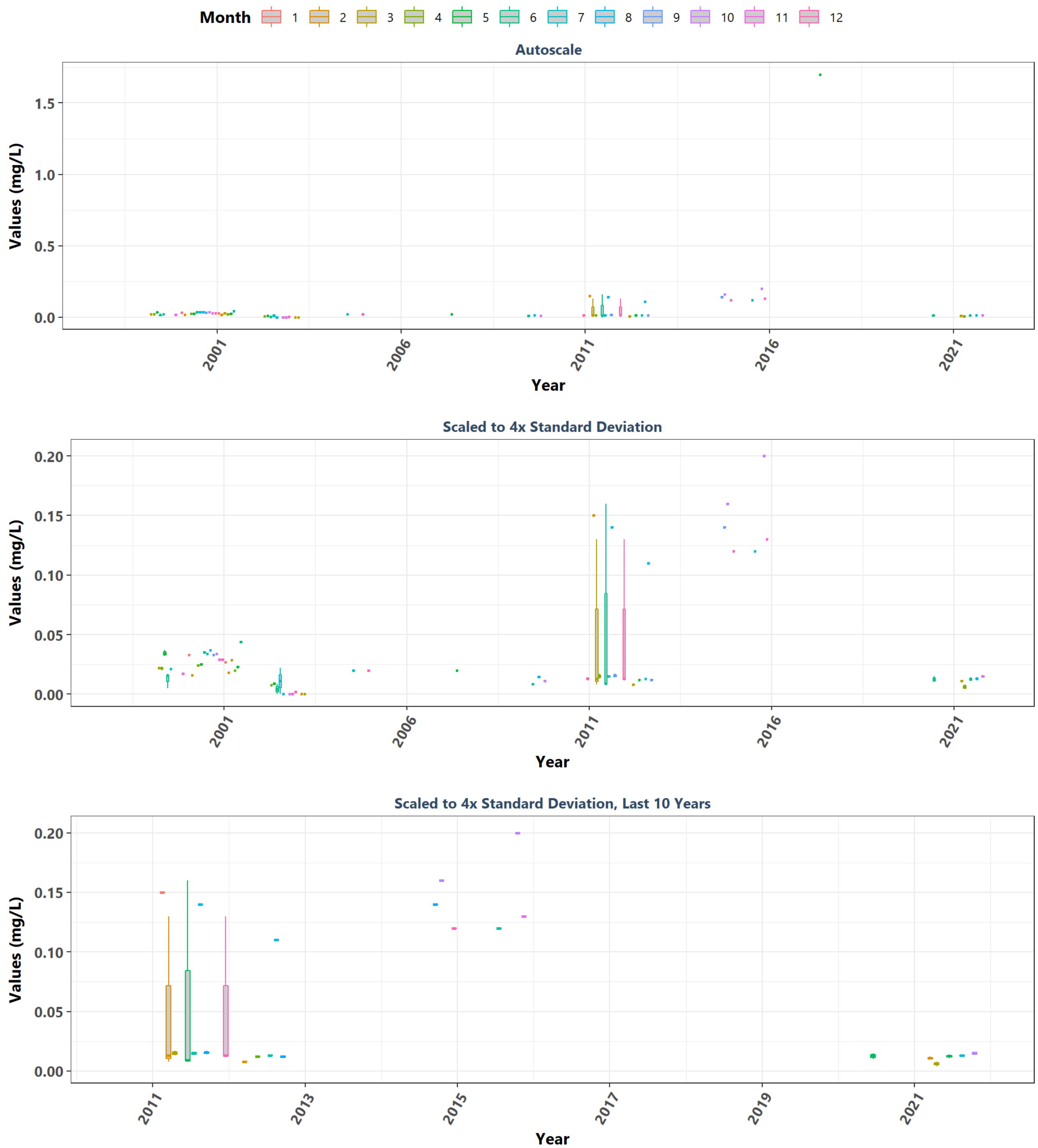




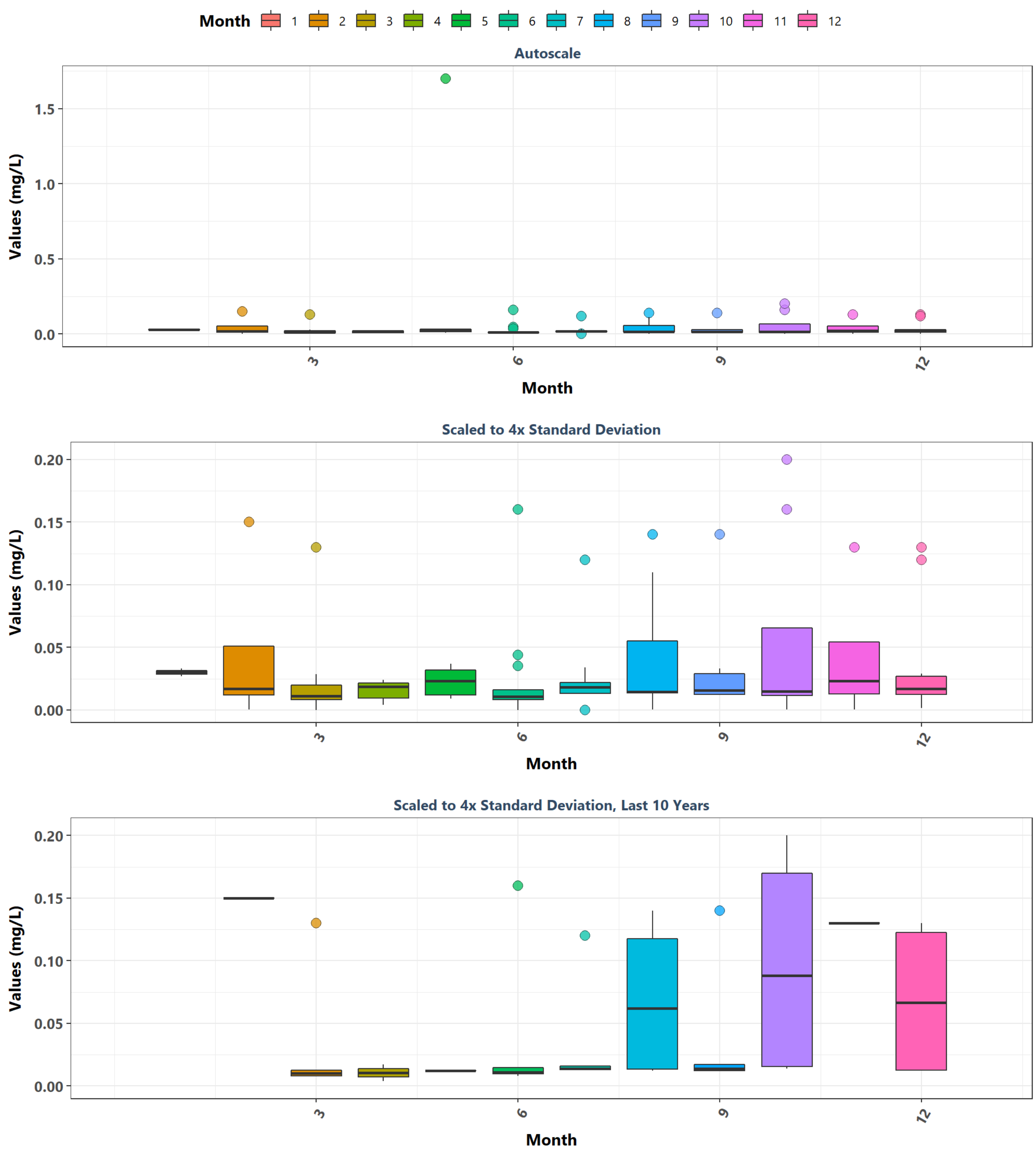


Fort Pickens State Park Aquatic Preserve

By Year & Month

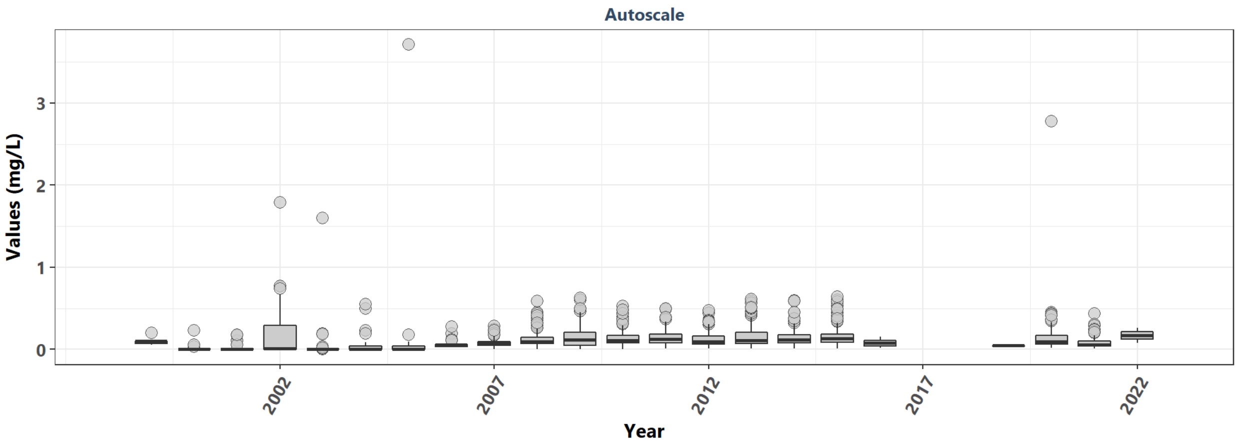


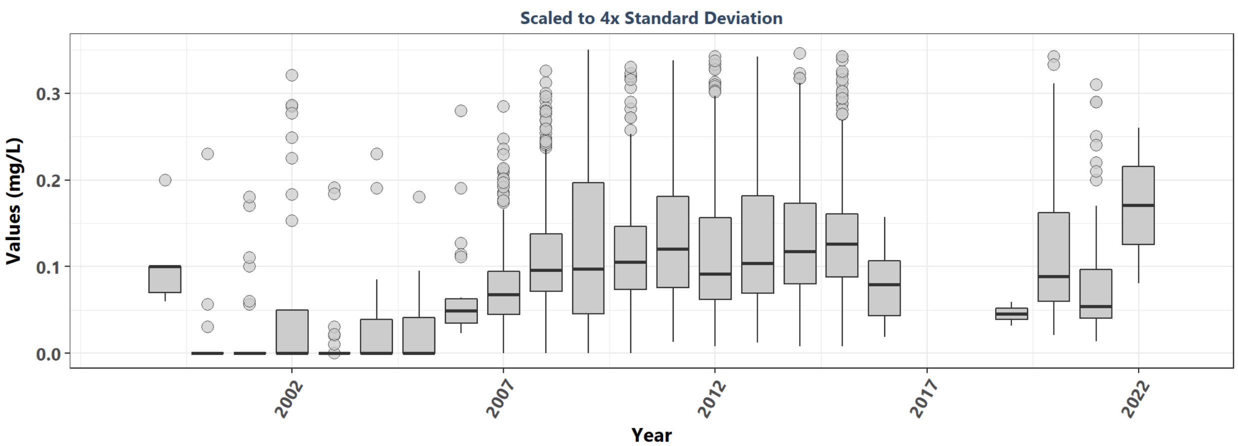
Fort Pickens State Park Aquatic Preserve
By Month

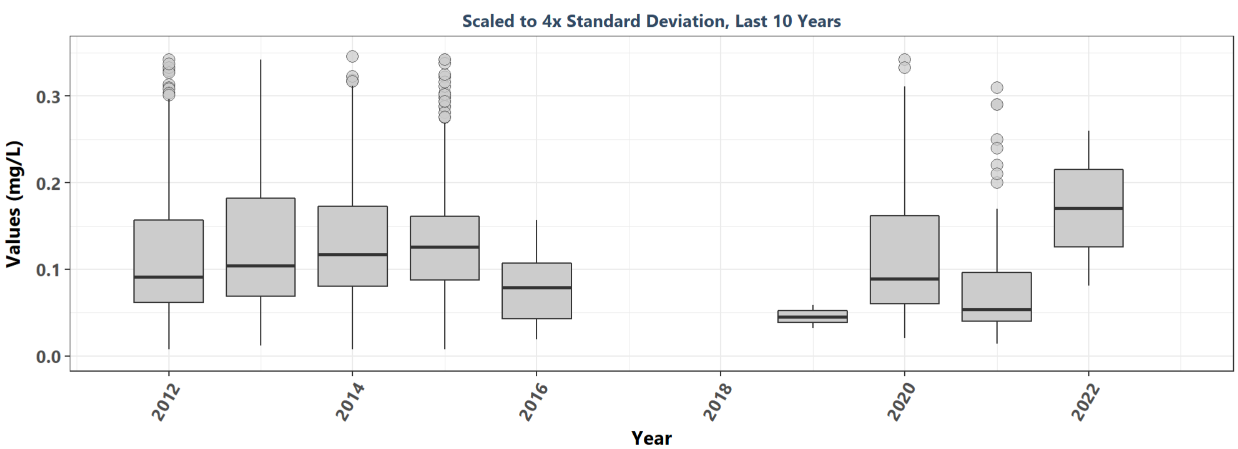


Gasparilla Sound-Charlotte Harbor Aquatic Preserve

By Year

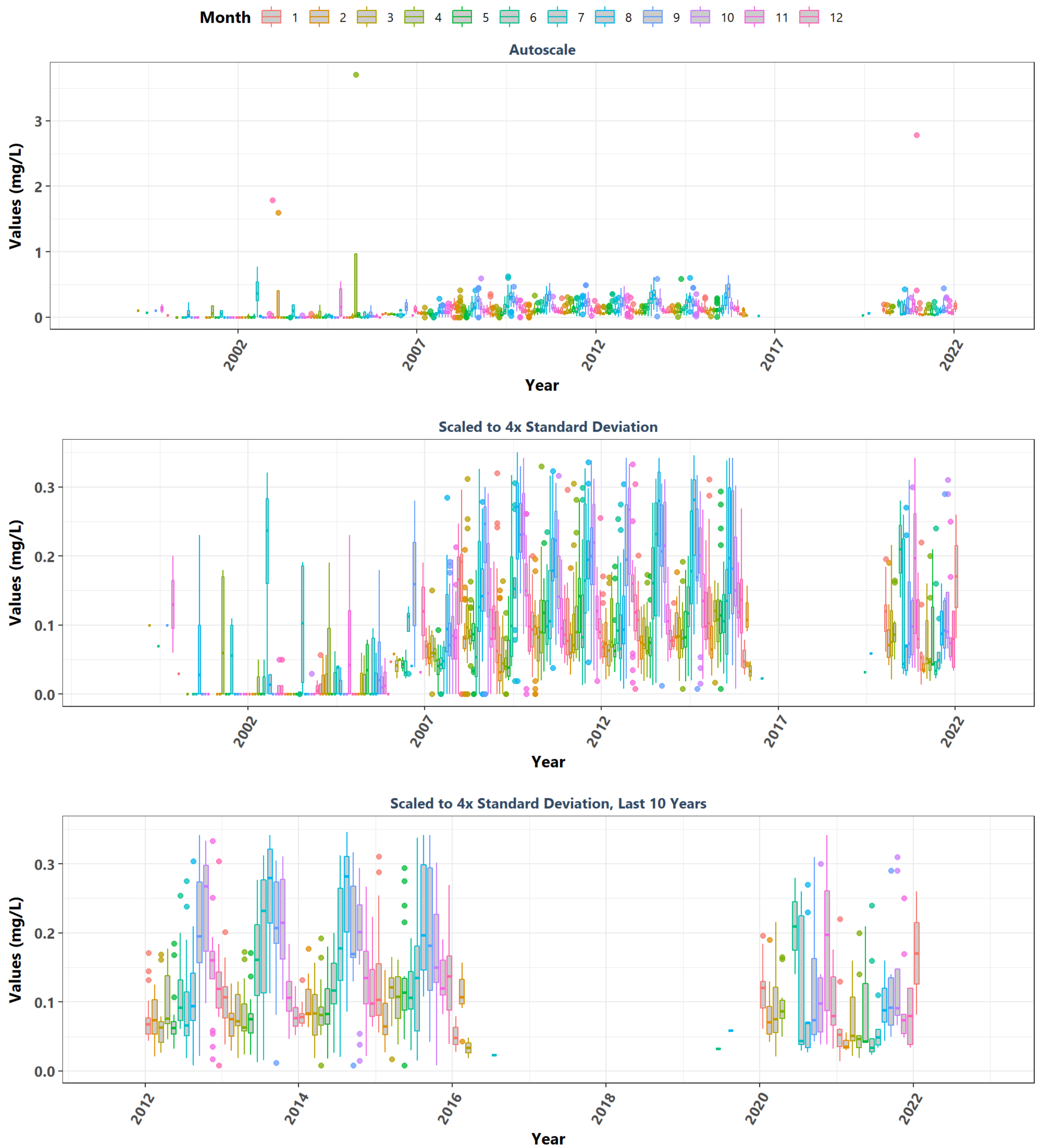






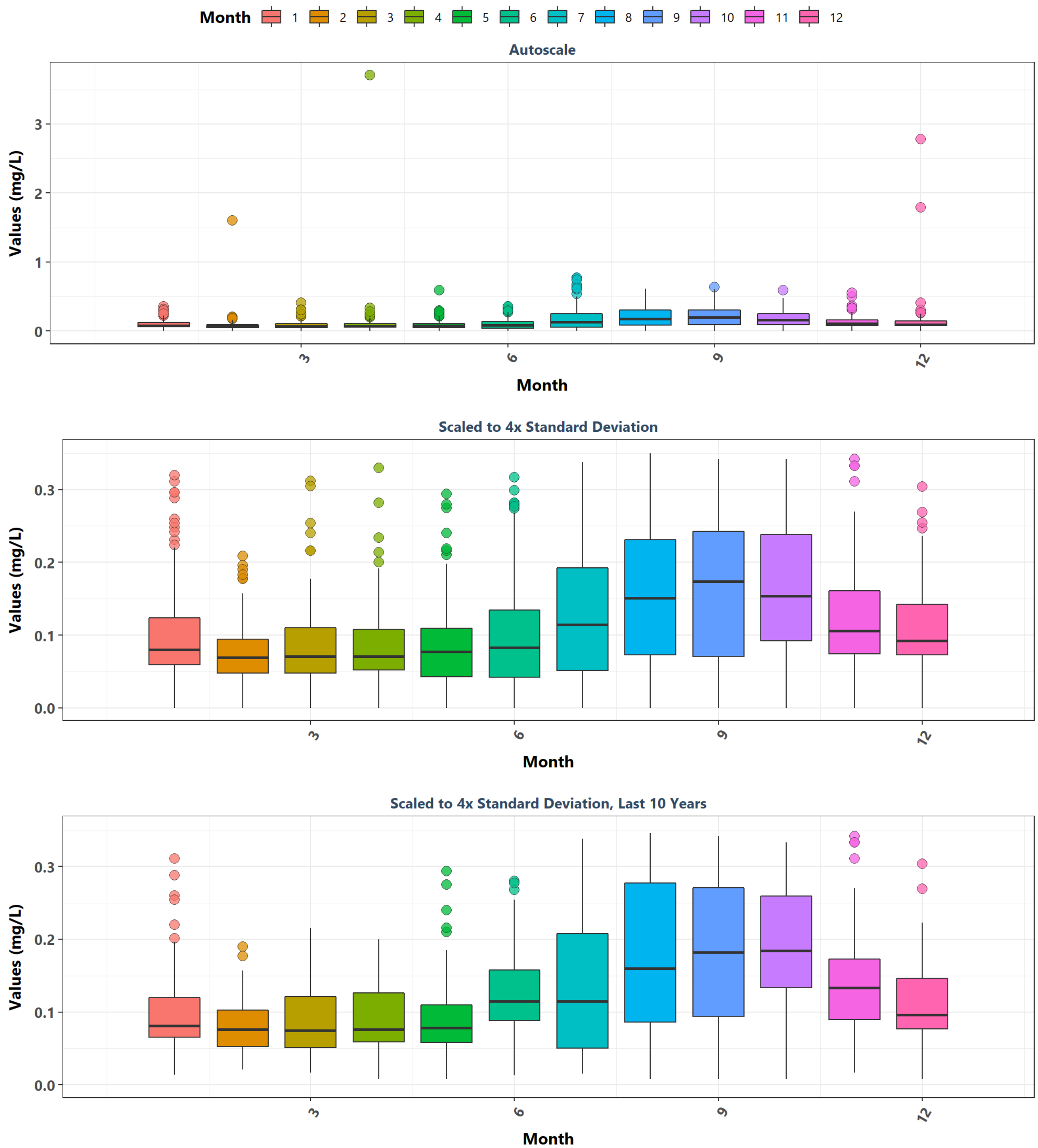
Gasparilla Sound-Charlotte Harbor Aquatic Preserve

By Year & Month



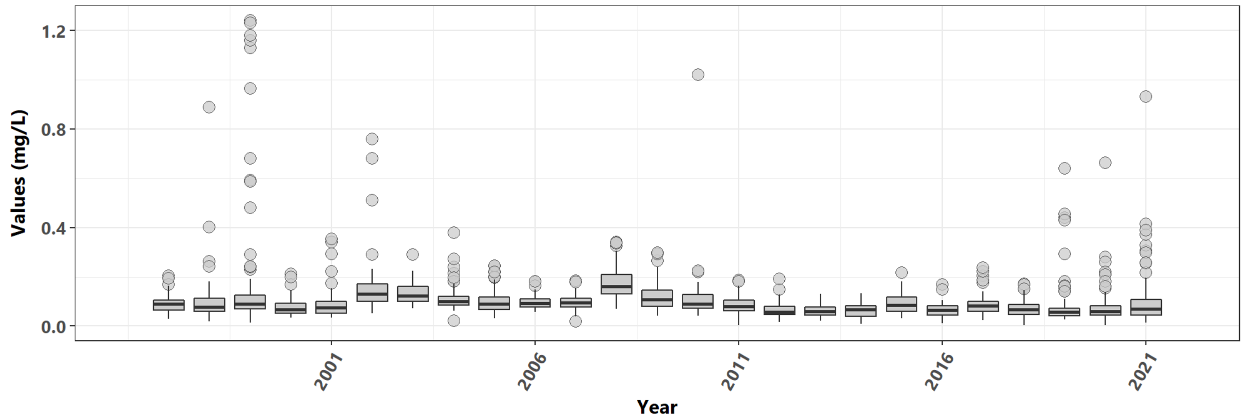
Gasparilla Sound-Charlotte Harbor Aquatic Preserve

By Month

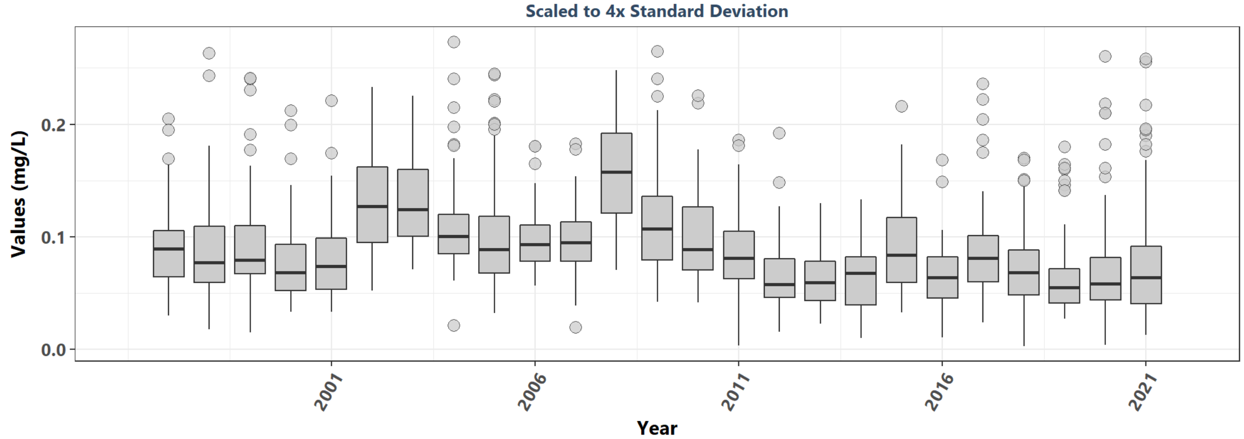


Guana River Marsh Aquatic Preserve
By Year

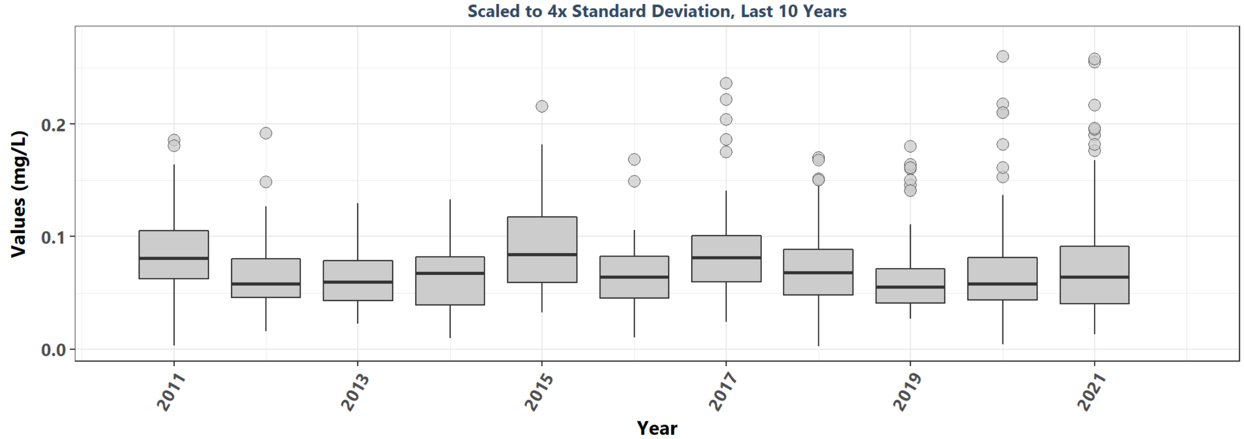
Autoscale

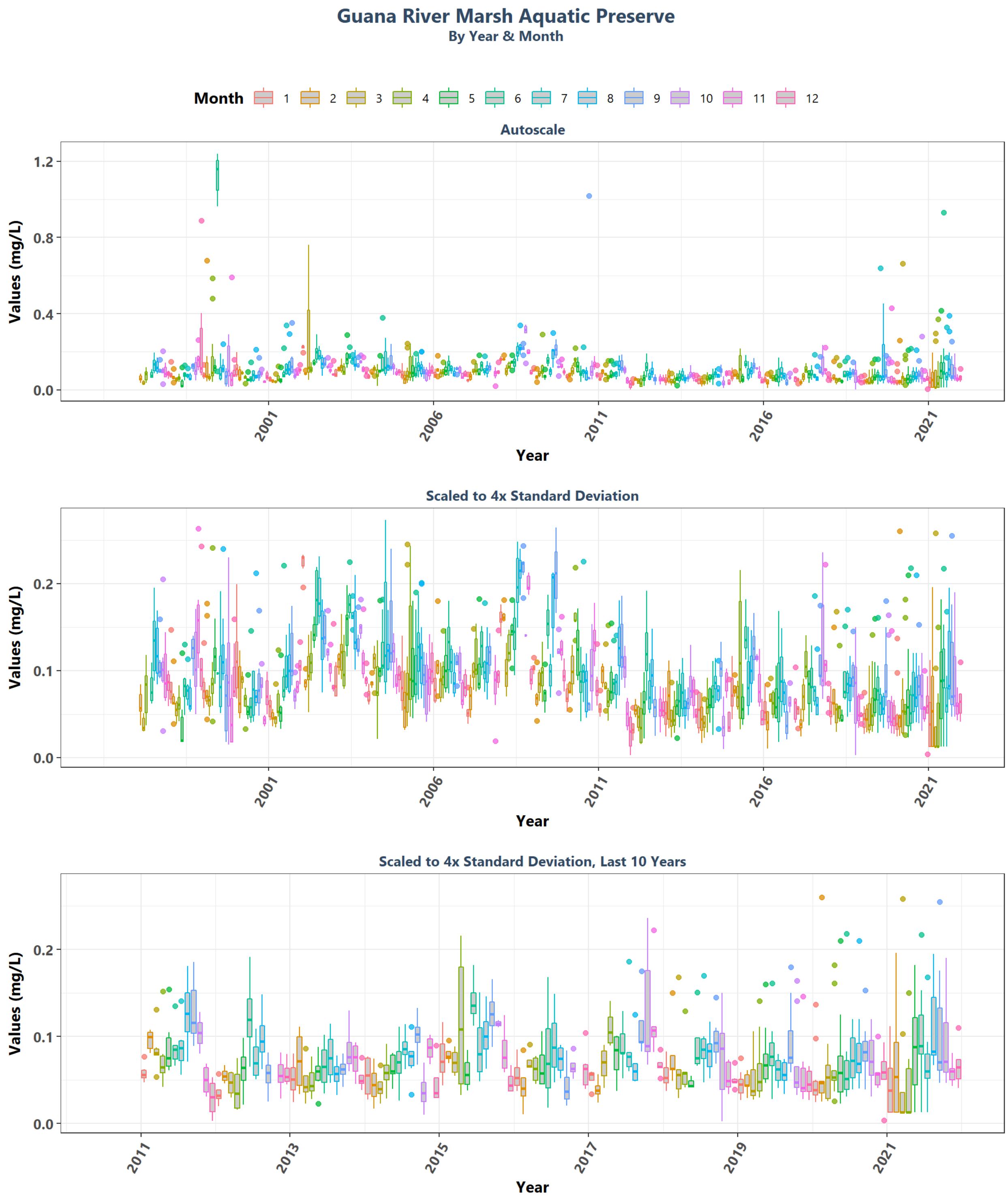


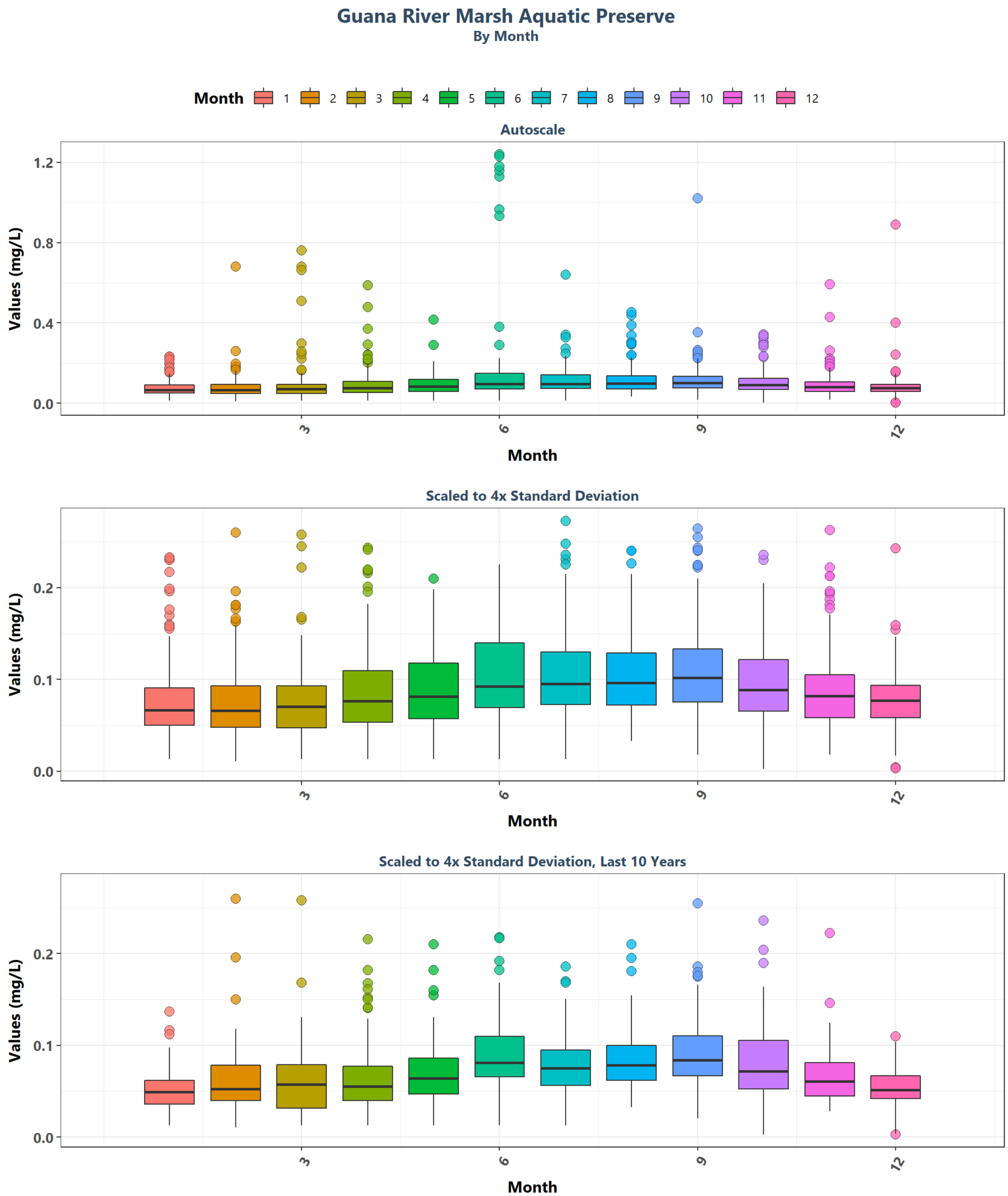
Scaled to 4x Standard Deviation



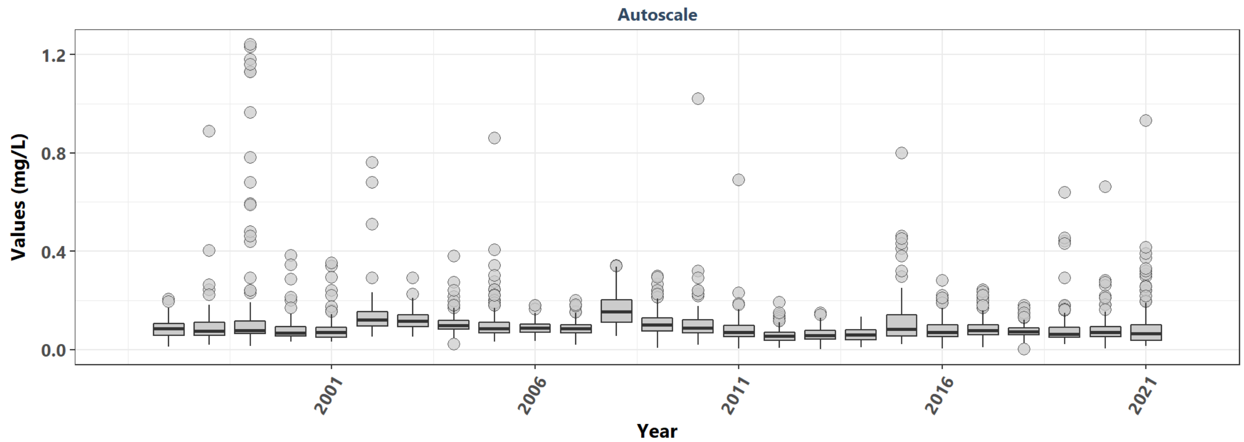
Scaled to 4x Standard Deviation, Last 10 Years

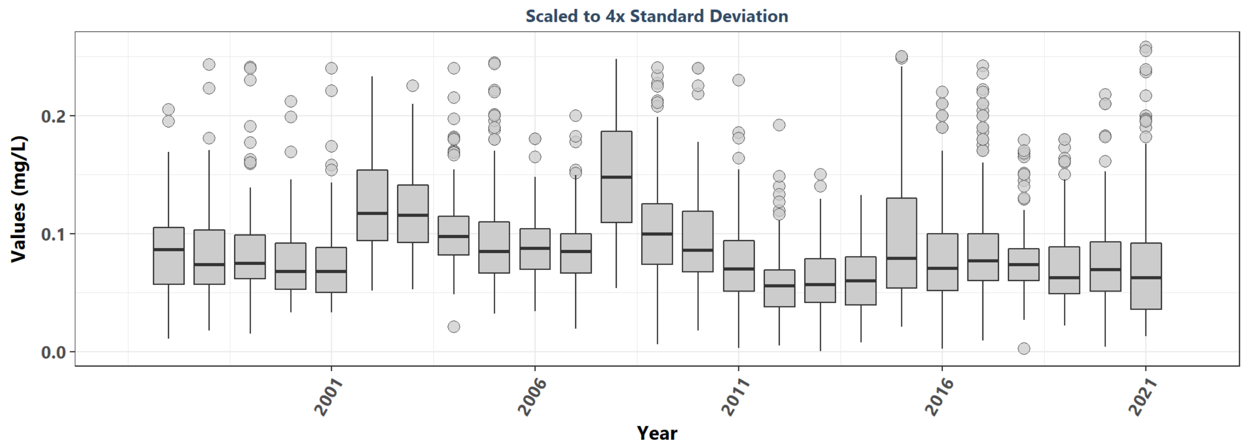


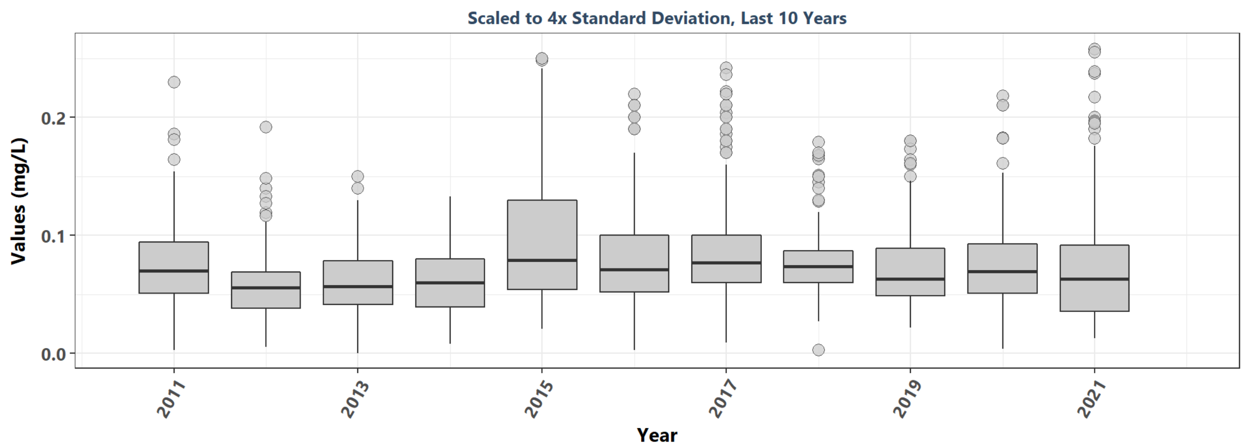

Guana River Marsh Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
1.2
0.8
0.4
0.0
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.2
0.1
0.0
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.2
0.1
0.0
2011
2013
2015
2017
2019
2021
Year

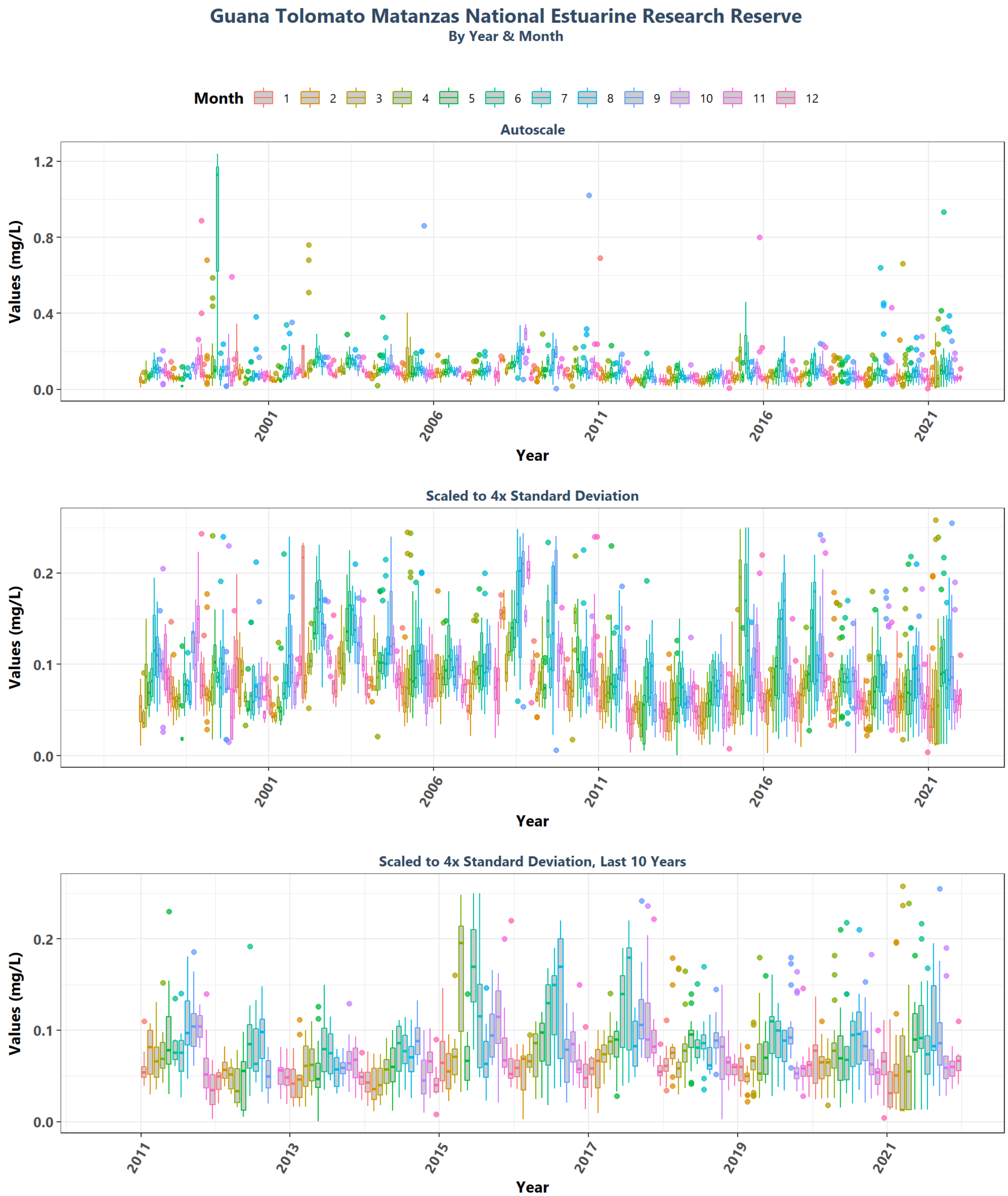

Guana River Marsh Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
1.2
0.8
0.4
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.2
0.1
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.2
0.1
0.0
3
6
9
12
Month

Guana Tolomato Matanzas National Estuarine Research Reserve
By Year

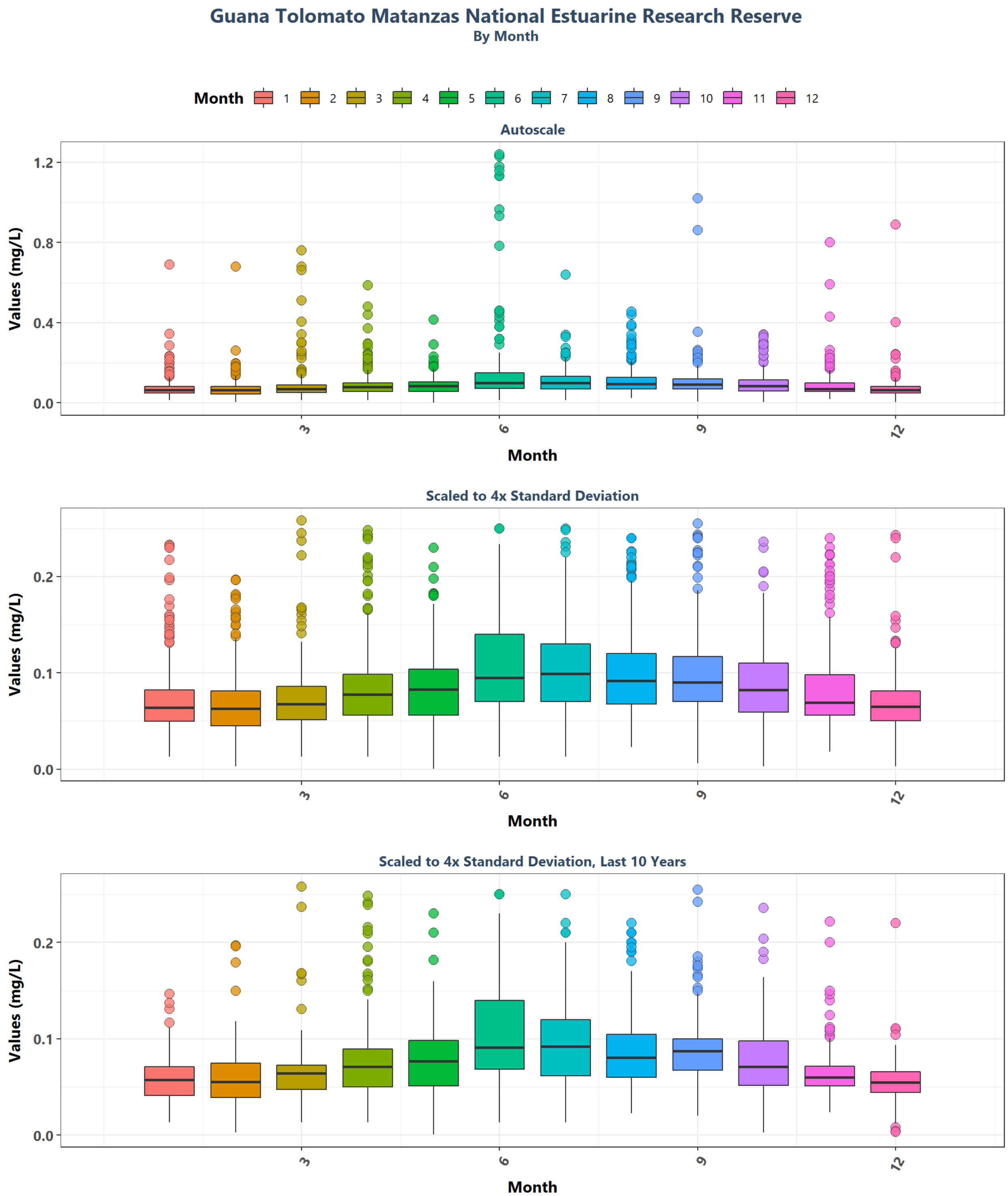






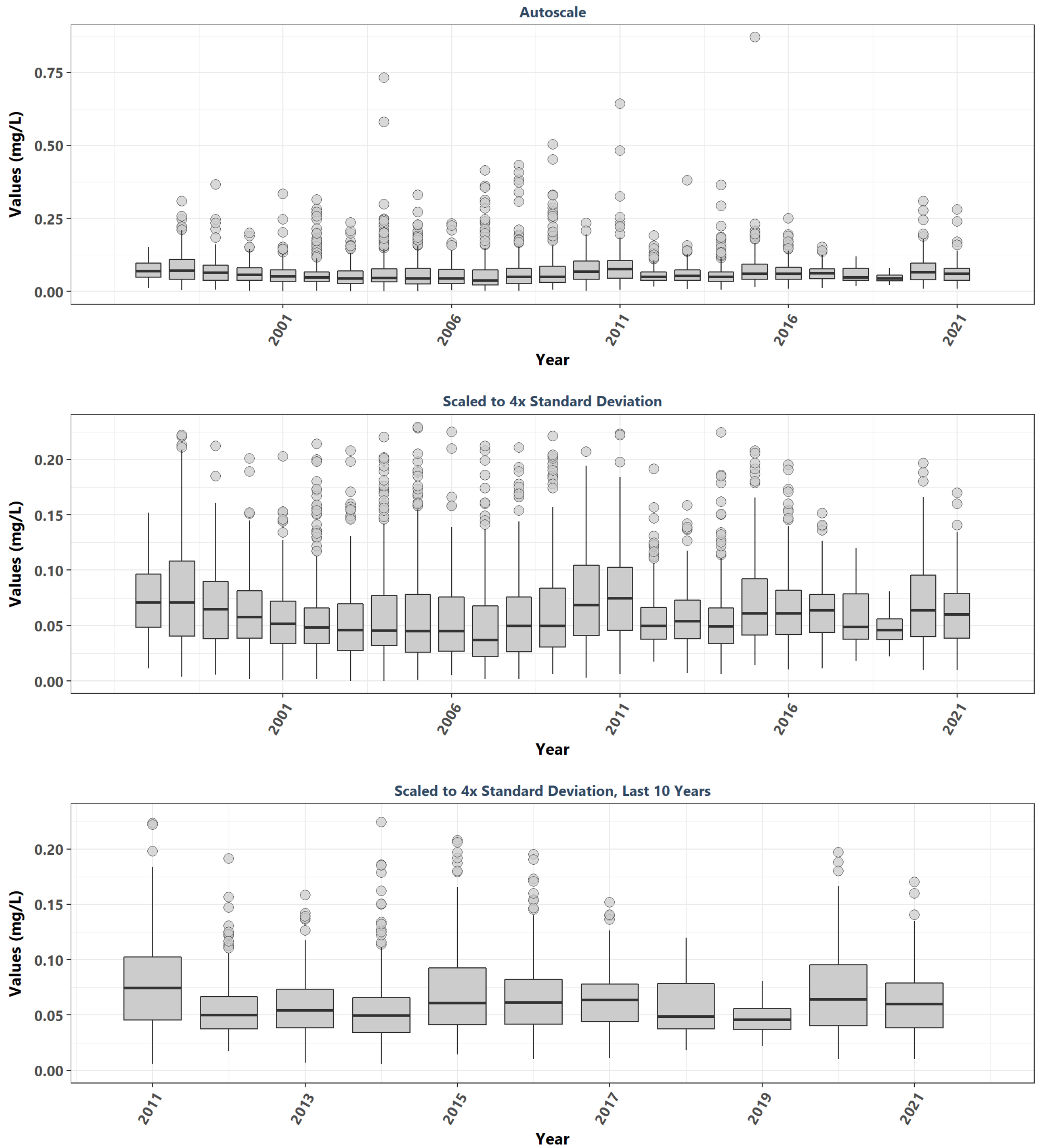


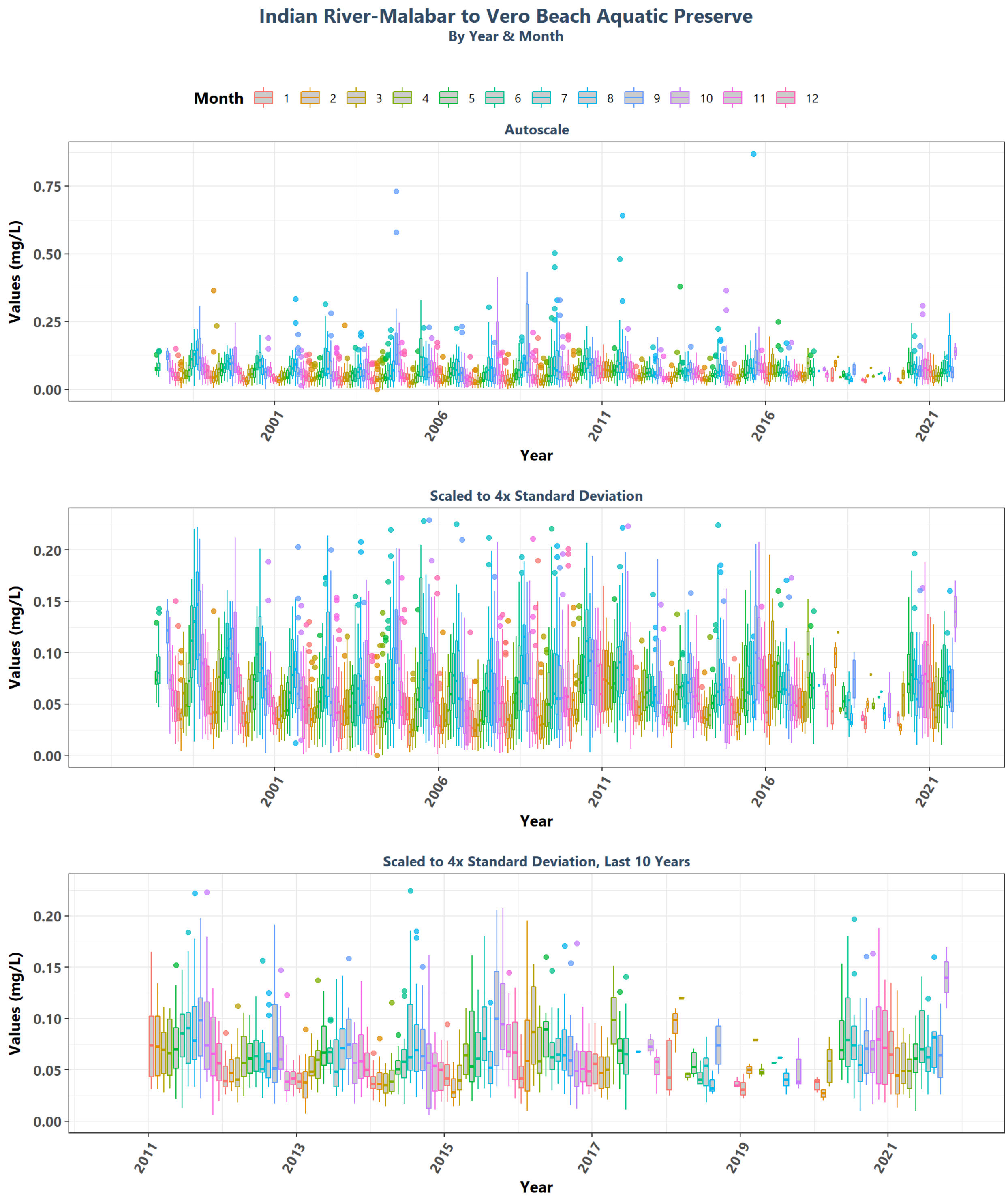
Guana Tolomato Matanzas National Estuarine Research Reserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
1.2
0.8
0.4
0.0
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.2
0.1
0.0
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.2
0.1
0.0
2011
2013
2015
2017
2019
2021
Year


Guana Tolomato Matanzas National Estuarine Research Reserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
1.2
0.8
0.4
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.2
0.1
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.2
0.1
0.0
3
6
9
12
Month

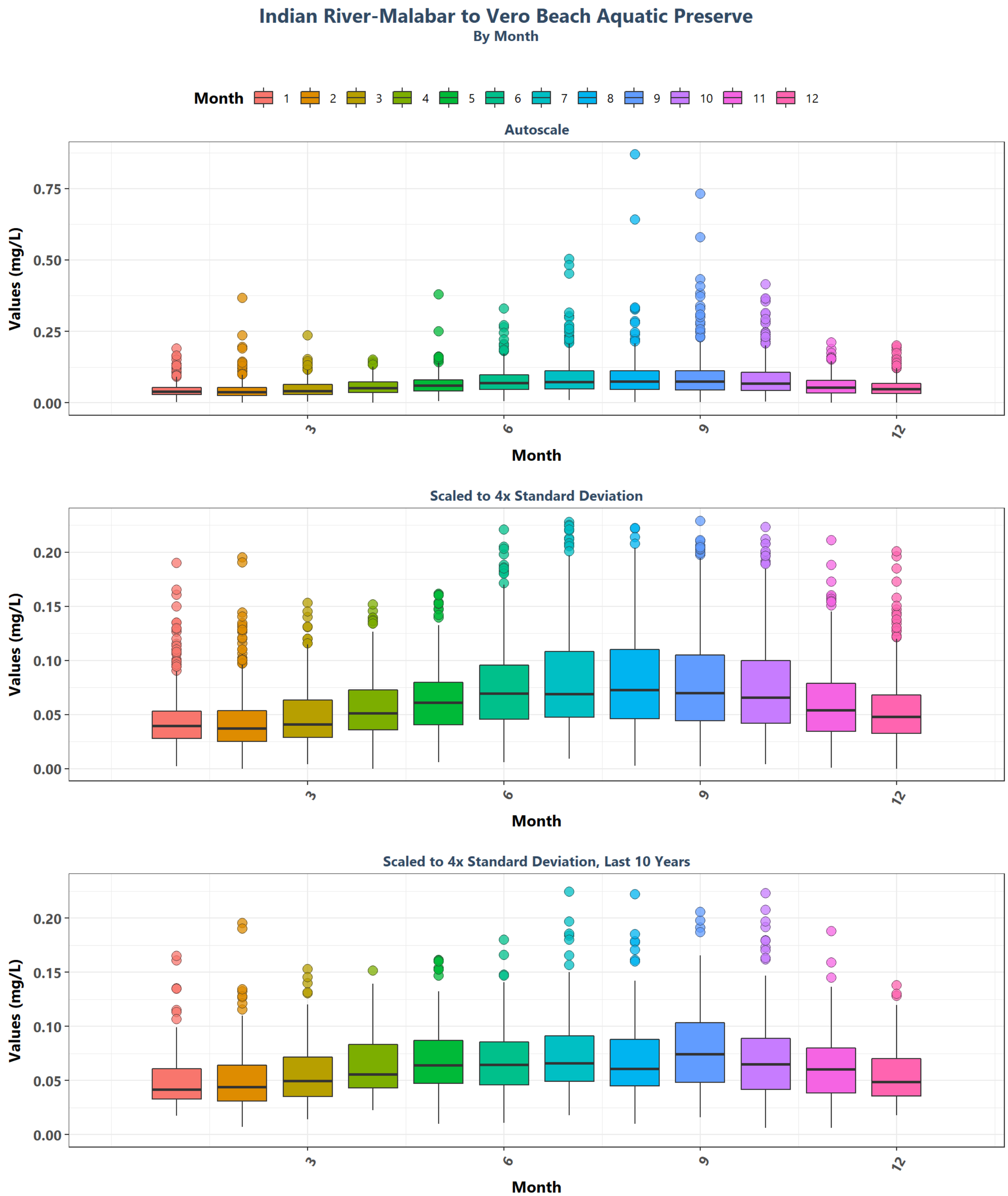
Indian River-Malabar to Vero Beach Aquatic Preserve

By Year





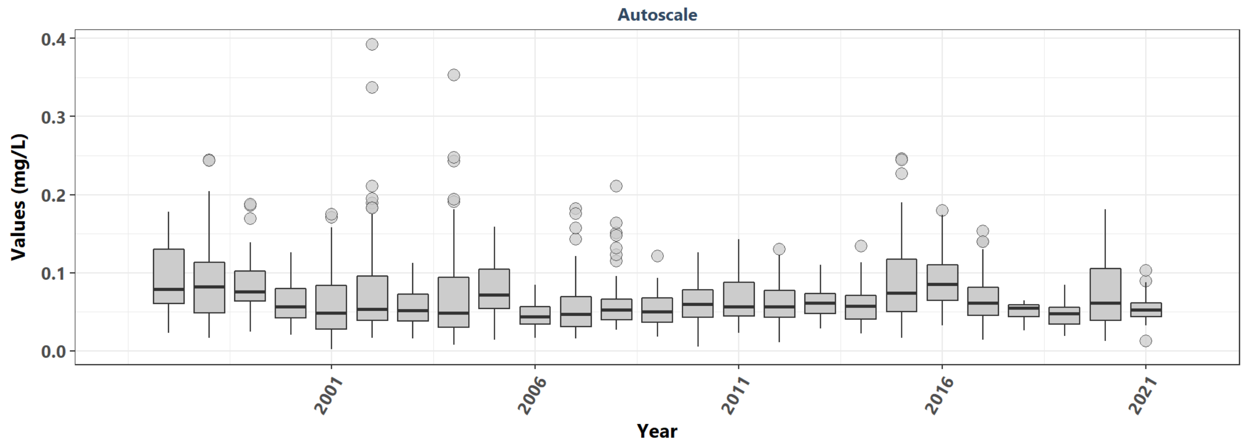
Indian River-Malabar to Vero Beach Aquatic Preserve
By Year & Month
Month
1
2
3
4
5
6
7
8
9
10
11
12
Autoscale
Values (mg/L)
0.00
0.25
0.50
0.75
Year
2001
2006
2011
2016
2021
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
0.20
Year
2001
2006
2011
2016
2021
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
0.20
Year
2011
2013
2015
2017
2019
2021

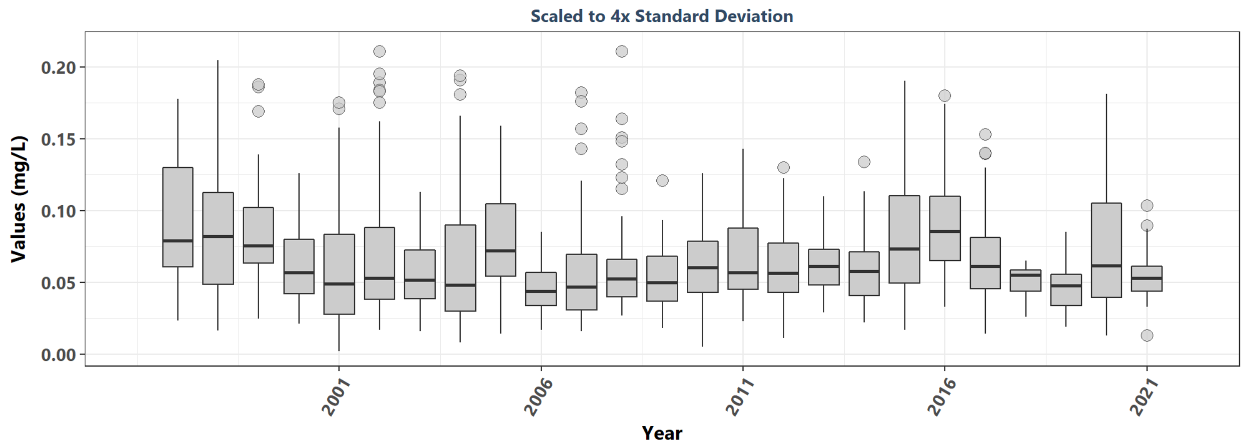


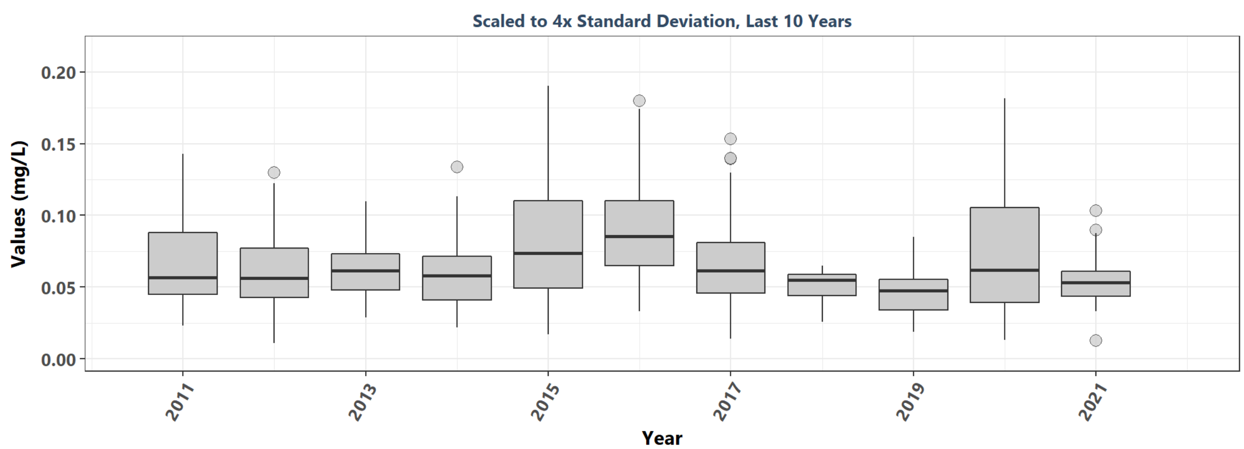
Indian River-Malabar to Vero Beach Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.00
0.25
0.50
0.75
Month
3
6
9
12
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
0.20
Month
3
6
9
12
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
0.20
Month
3
6
9
12

Indian River-Vero Beach to Ft. Pierce Aquatic Preserve

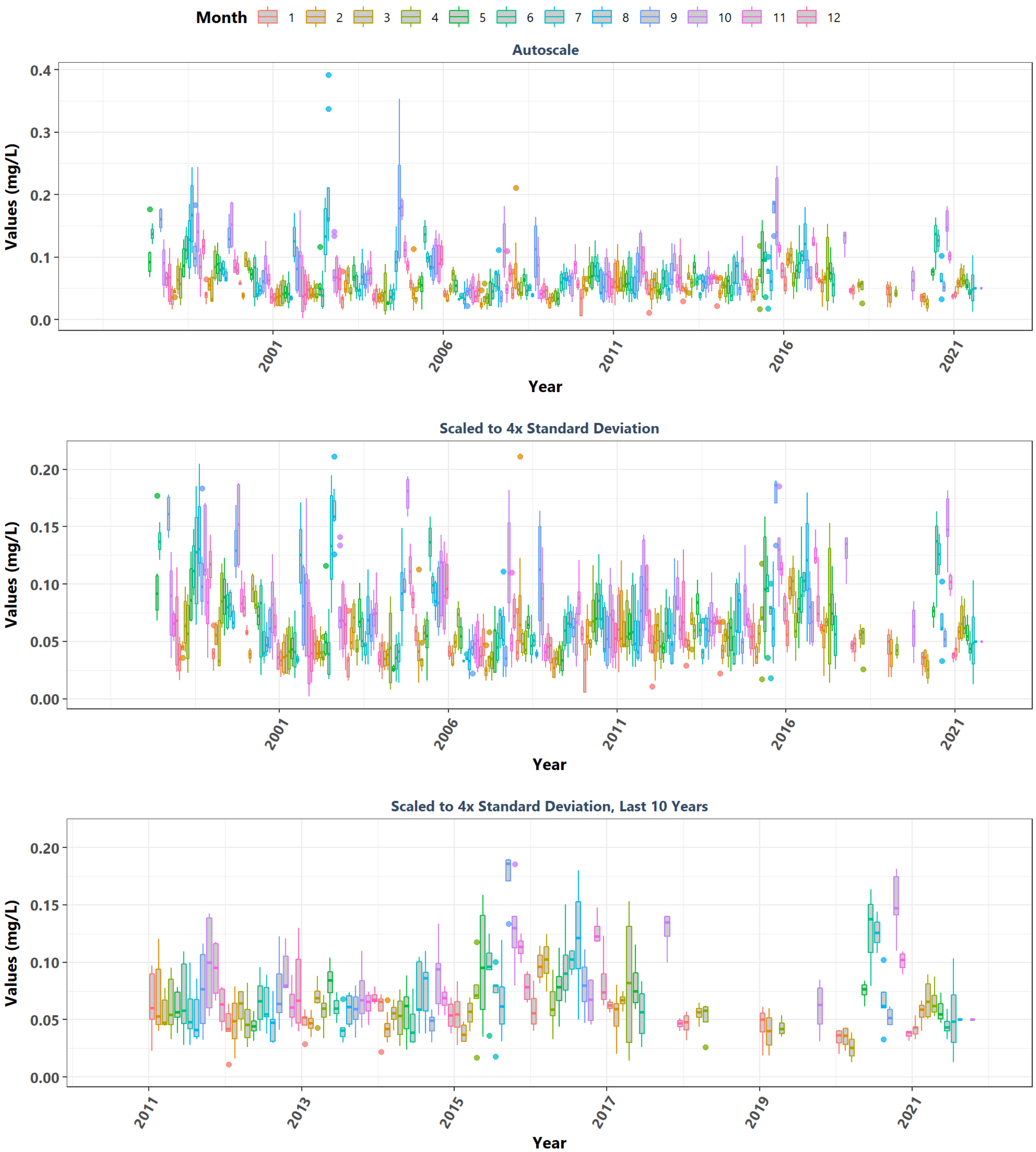
By Year

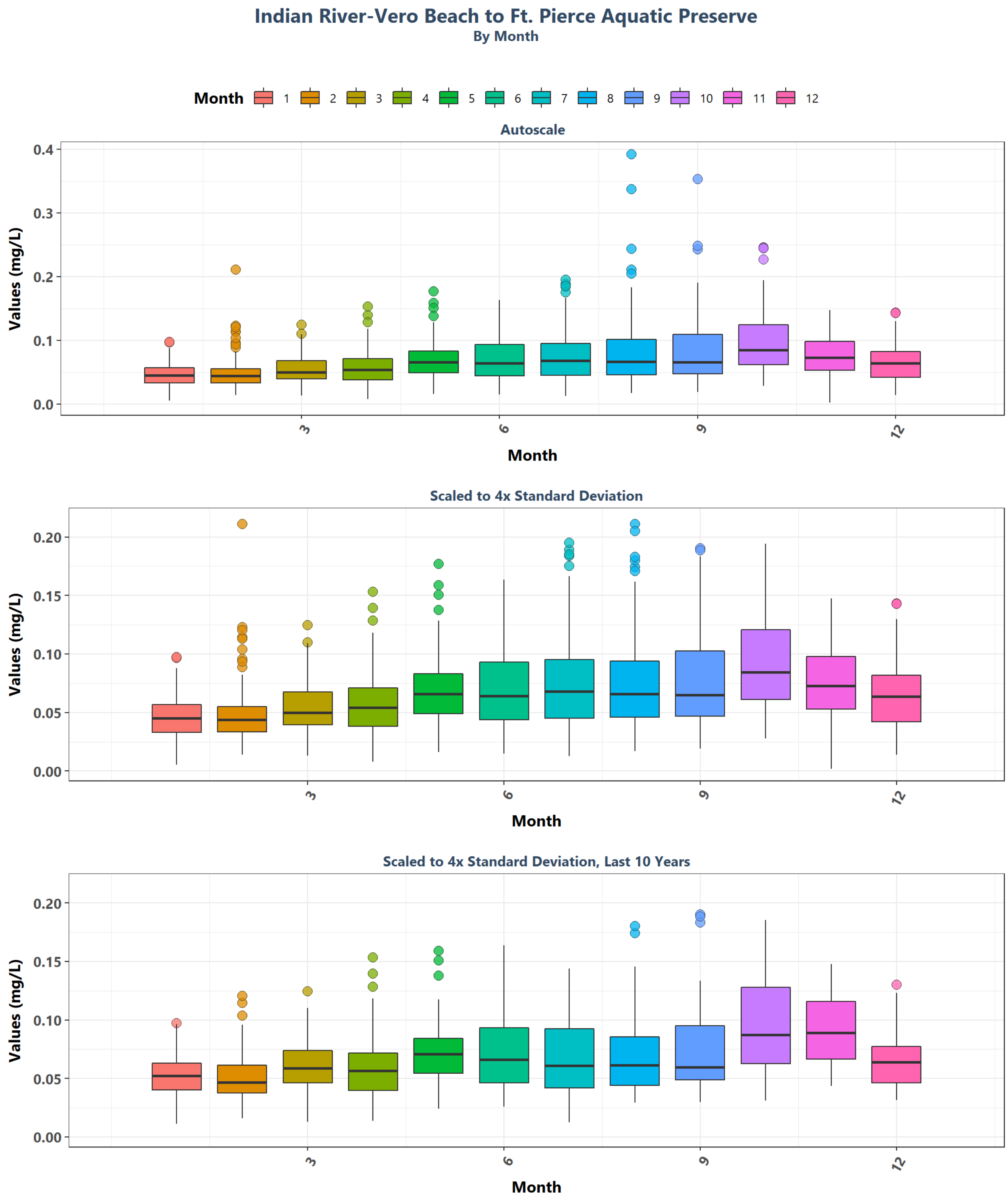




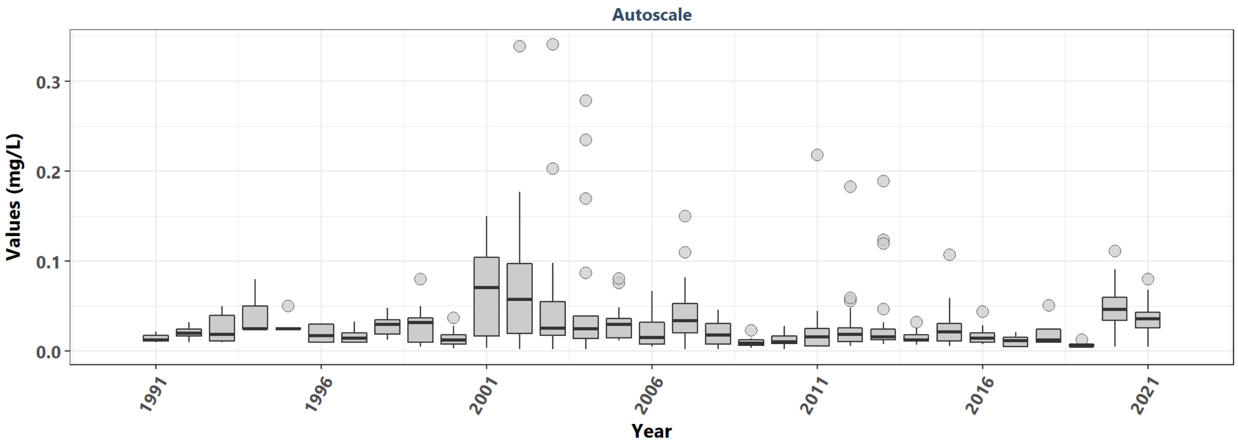


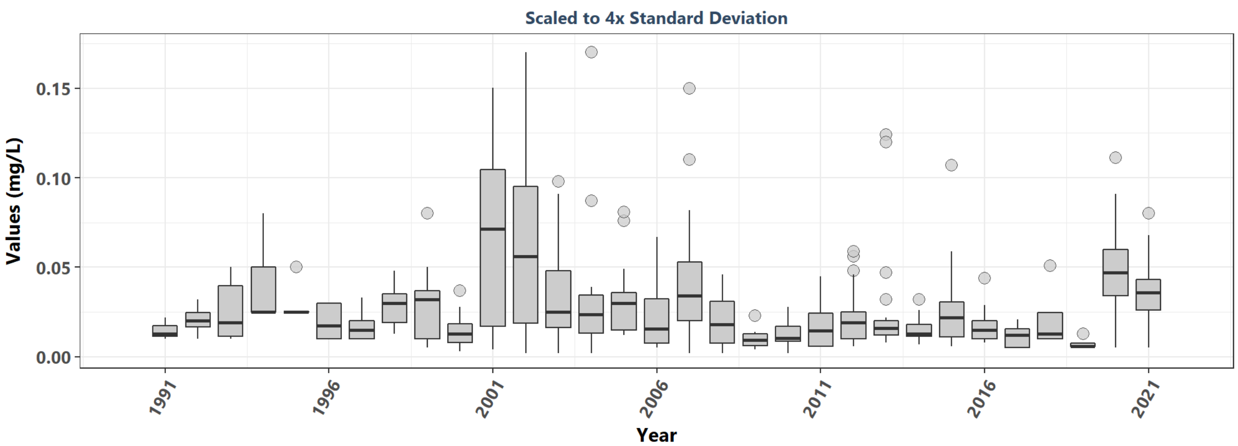
Indian River-Vero Beach to Ft. Pierce Aquatic Preserve
By Year & Month

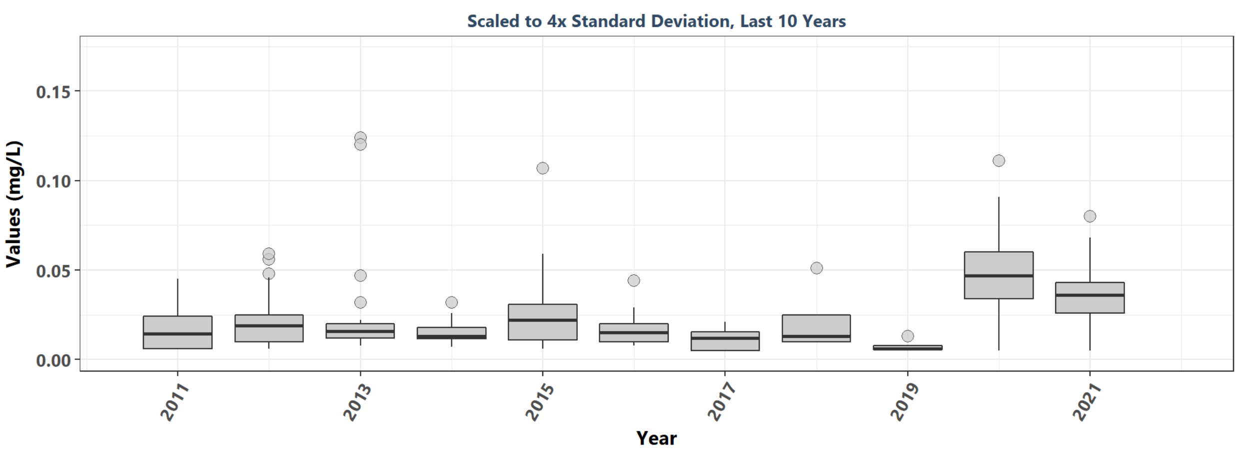


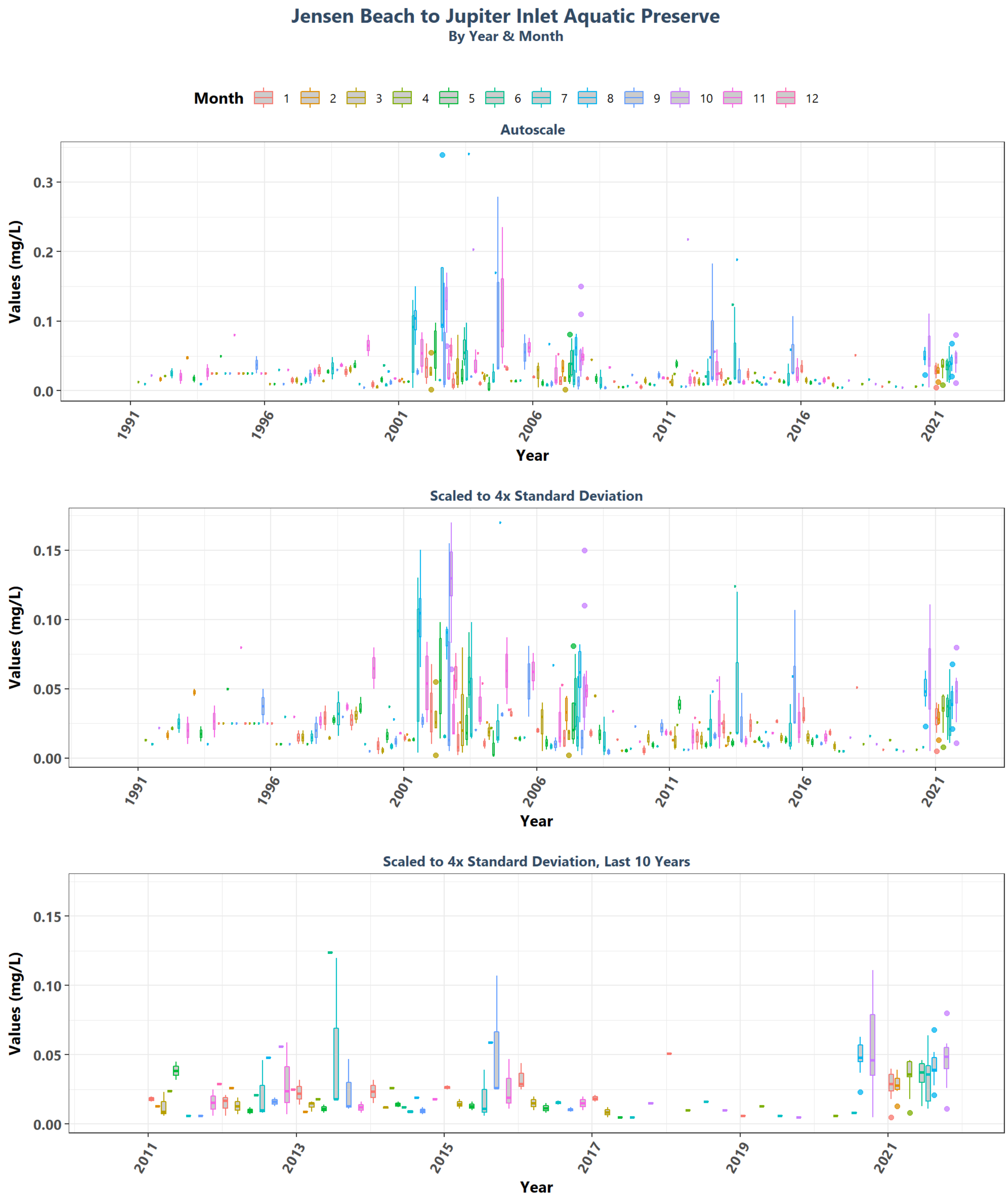

Indian River-Vero Beach to Ft. Pierce Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.1
0.2
0.3
0.4
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
0.20
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
0.20
3
6
9
12
Month

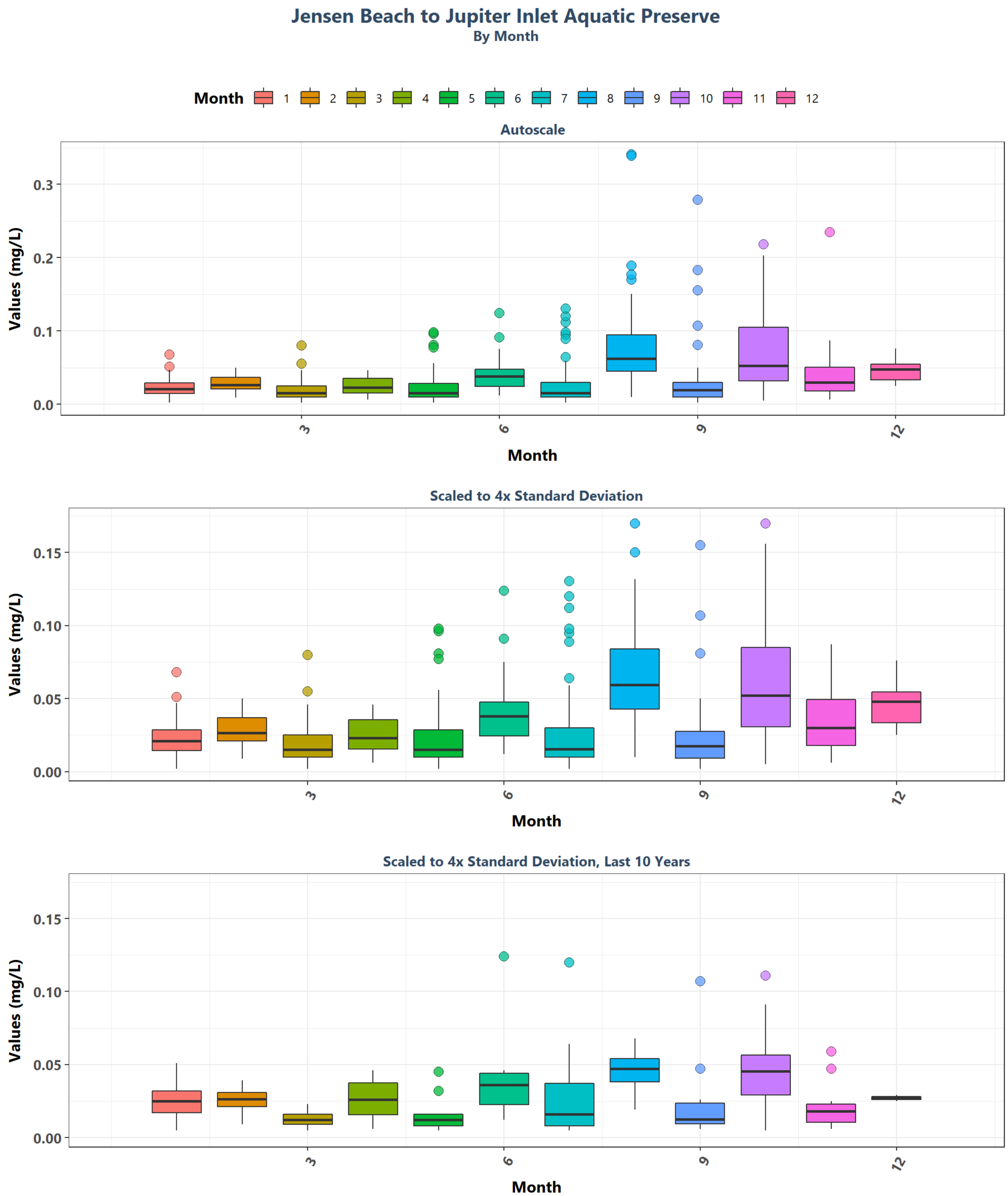
Jensen Beach to Jupiter Inlet Aquatic Preserve
By Year





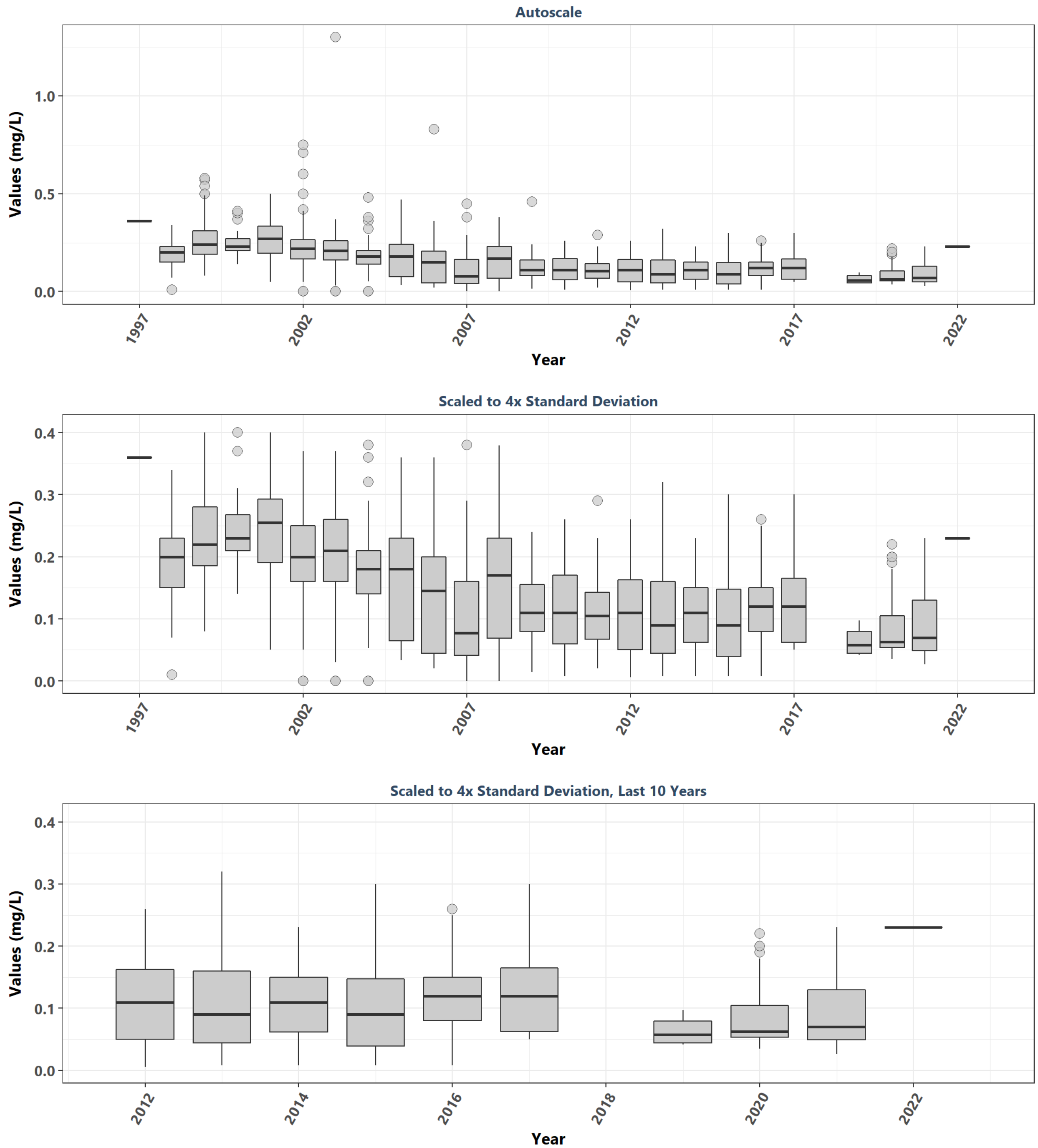


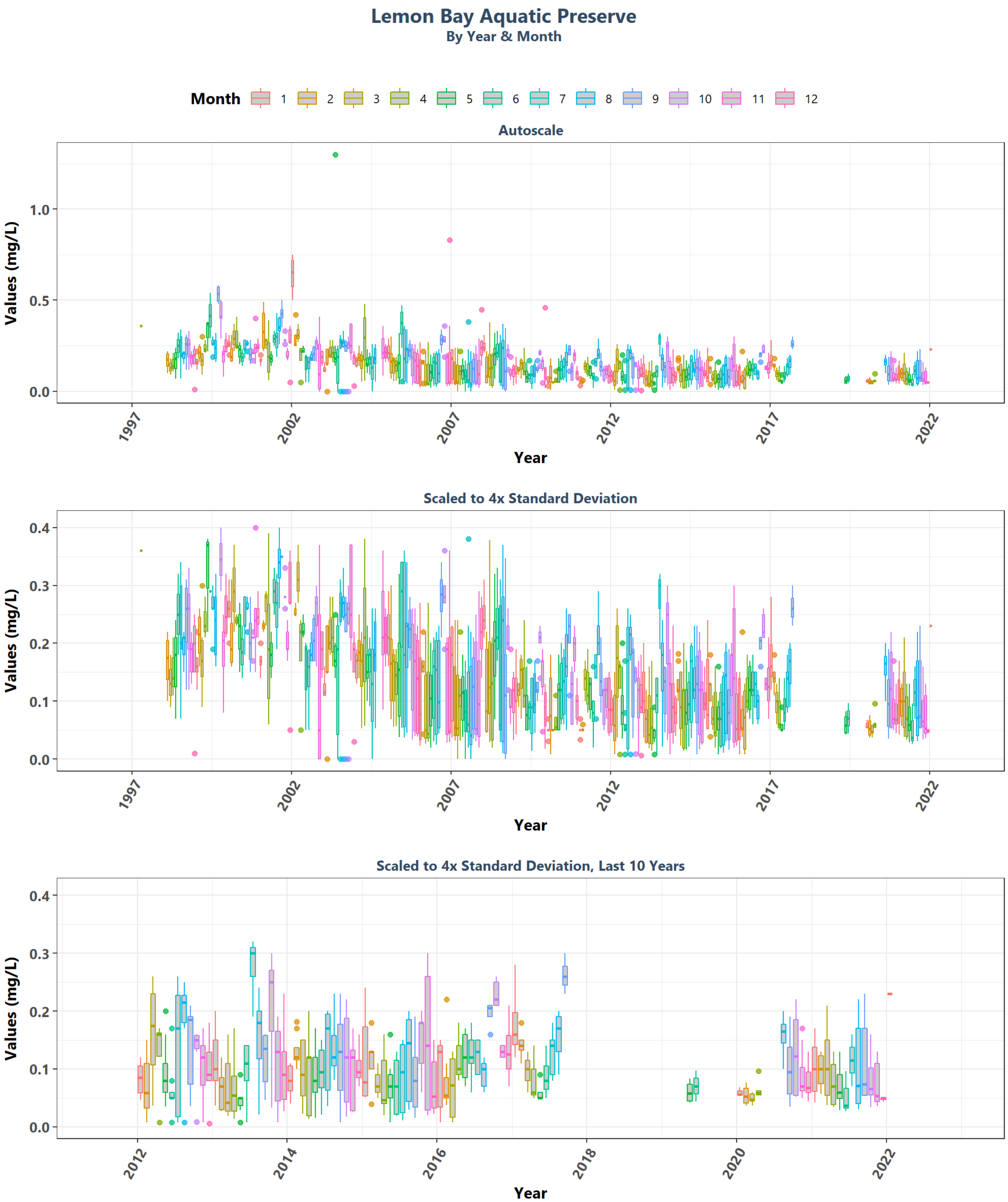

Jensen Beach to Jupiter Inlet Aquatic Preserve
By Year & Month
Month
1
2
3
4
5
6
7
8
9
10
11
12
Autoscale
Values (mg/L)
0.0
0.1
0.2
0.3
1991
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
1991
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
2011
2013
2015
2017
2019
2021
Year


Jensen Beach to Jupiter Inlet Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.1
0.2
0.3
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
3
6
9
12
Month

Lemon Bay Aquatic Preserve

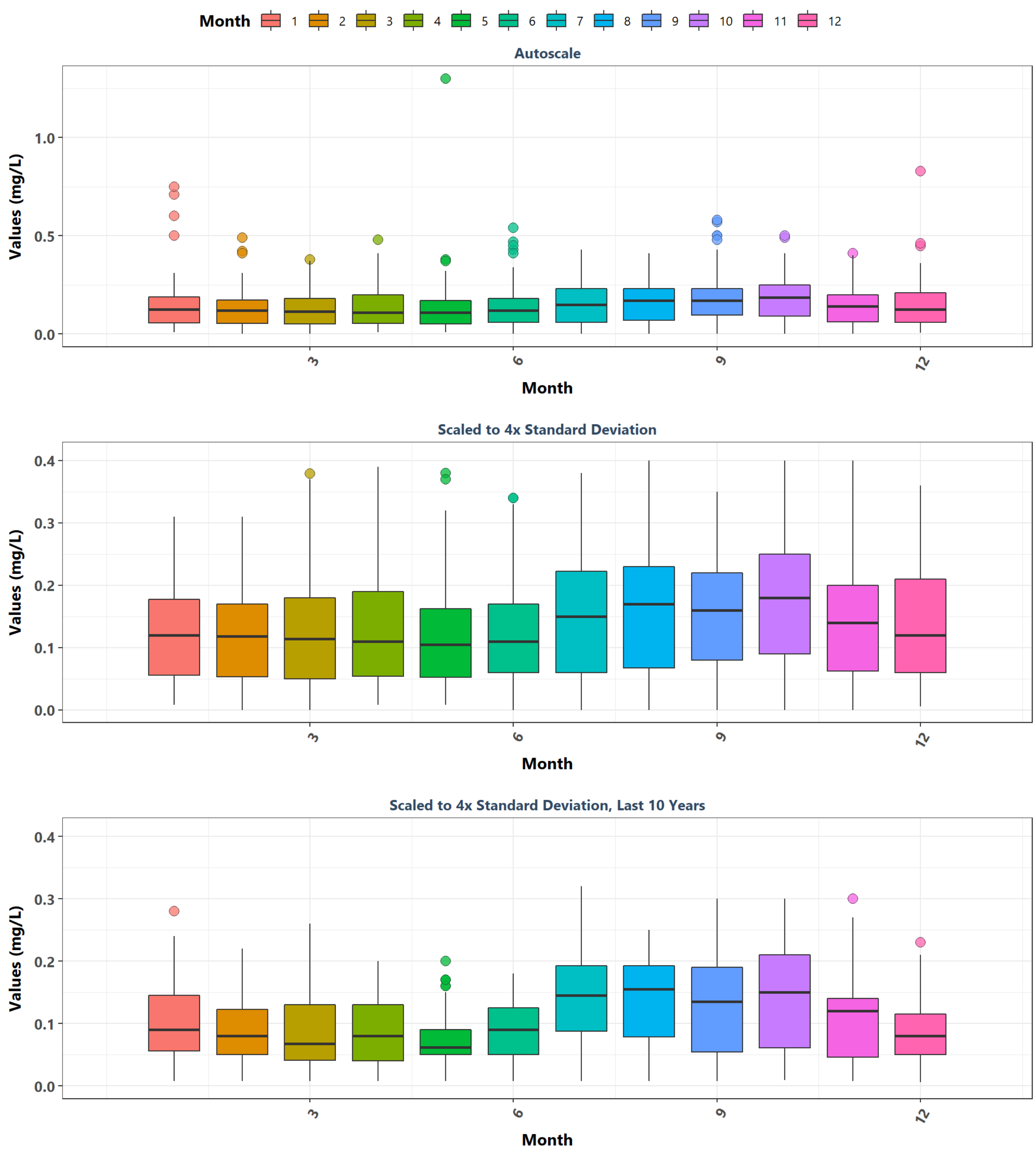
By Year

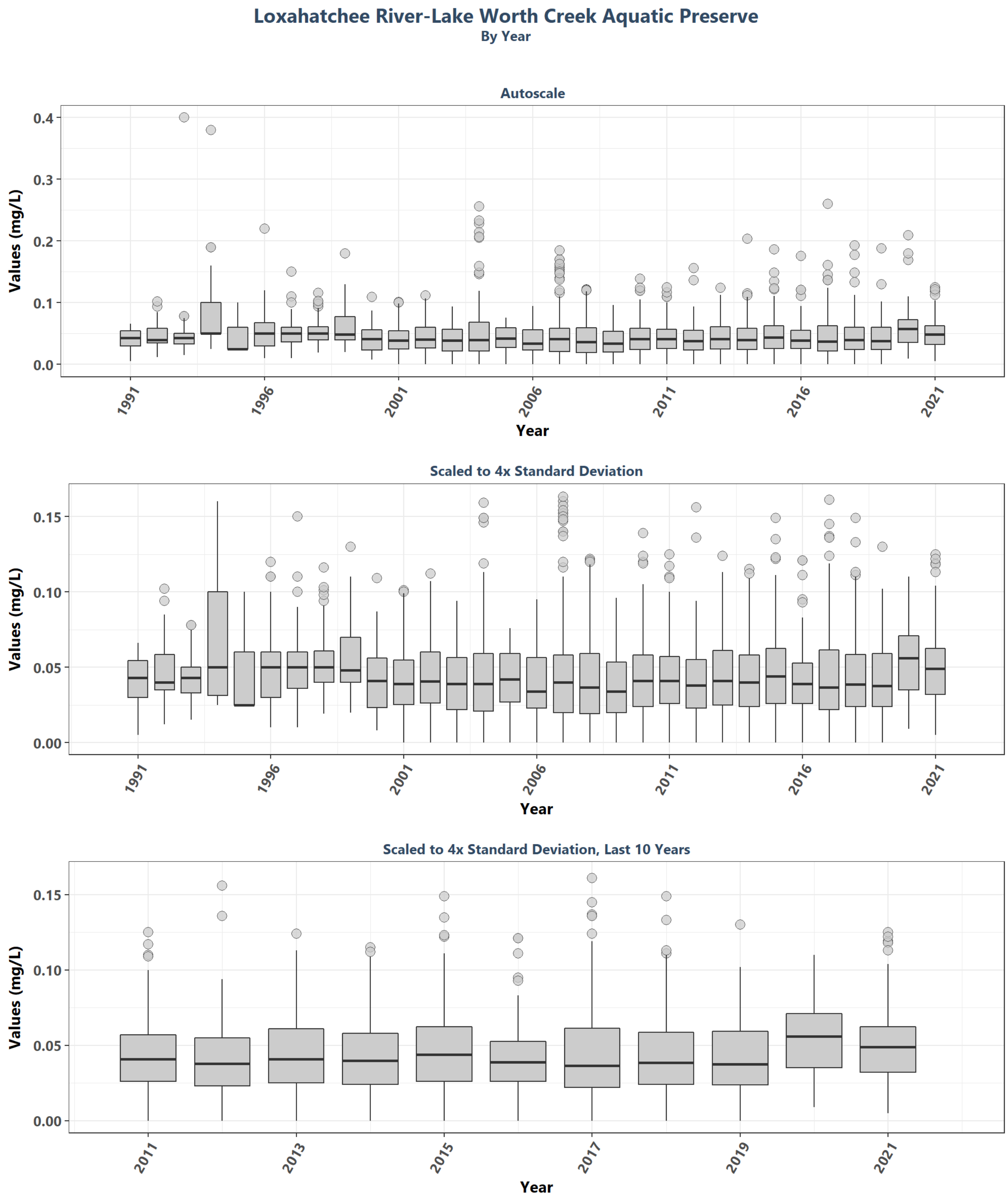


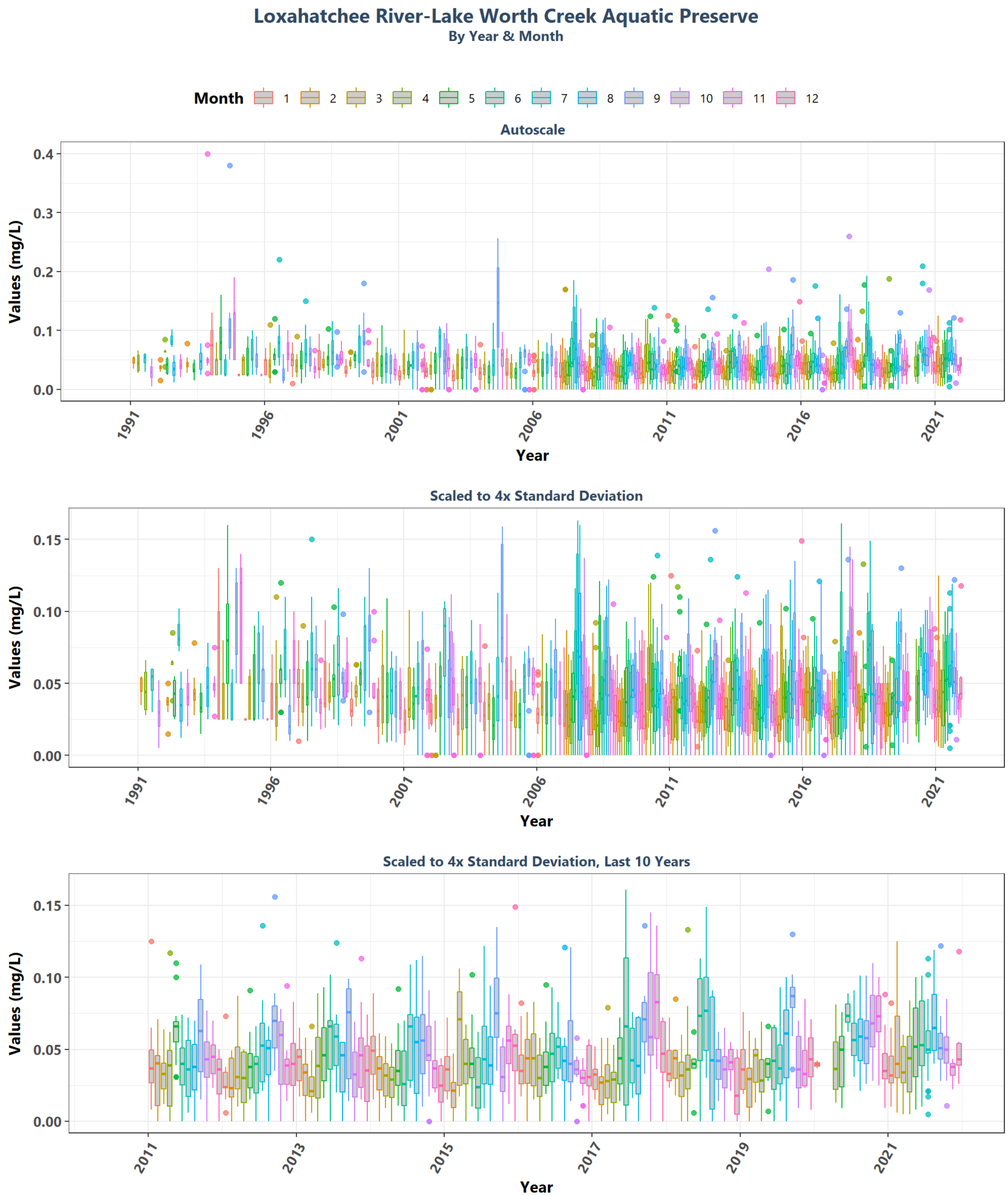

Lemon Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
1.0
0.5
0.0
1997
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.4
0.3
0.2
0.1
0.0
1997
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.4
0.3
0.2
0.1
0.0
2012
2014
2016
2018
2020
2022
Year

Lemon Bay Aquatic Preserve

By Month

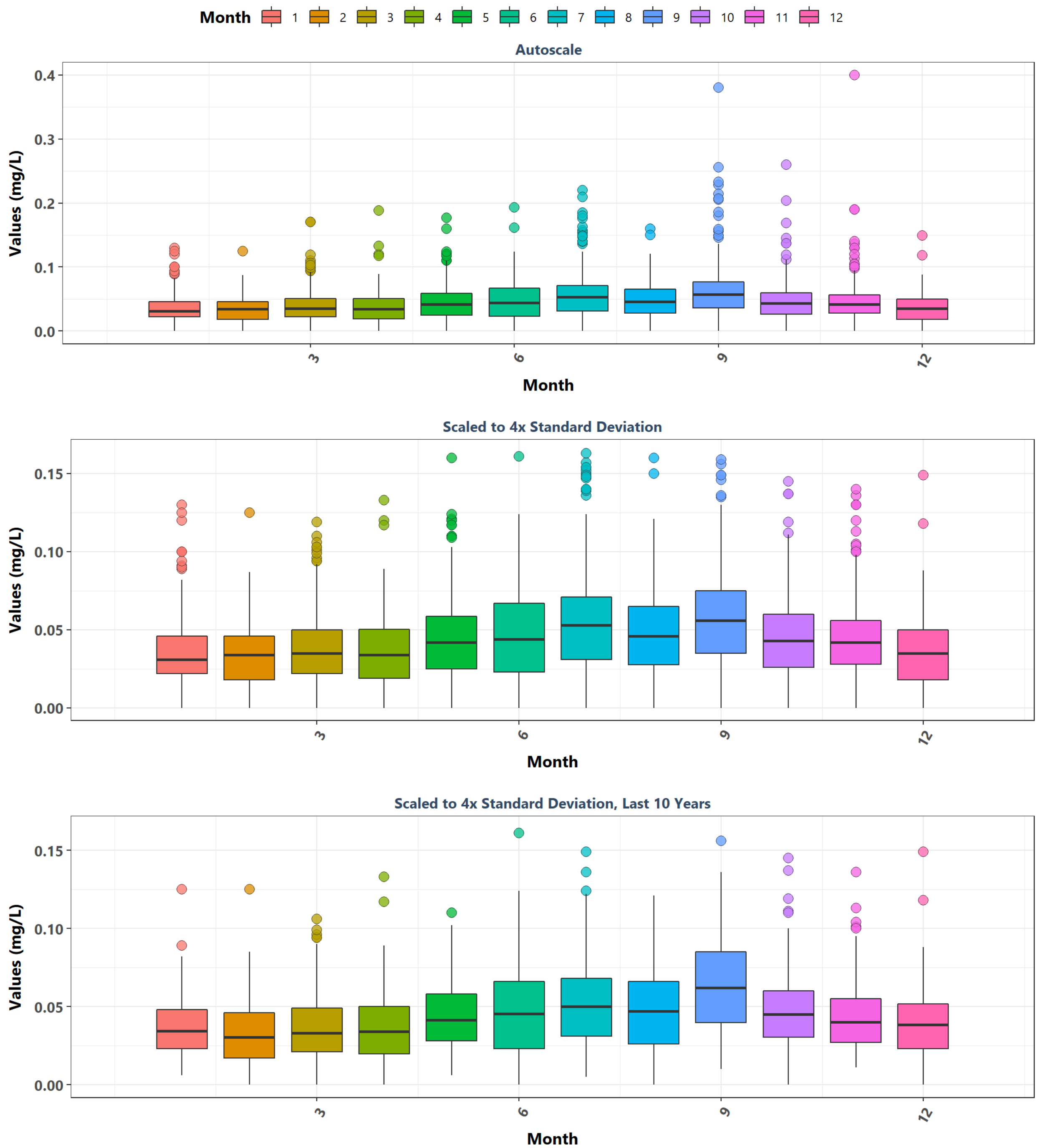



Loxahatchee River-Lake Worth Creek Aquatic Preserve
By Year
Autoscale
Values (mg/L)
0.0
0.1
0.2
0.3
0.4
1991
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
1991
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
2011
2013
2015
2017
2019
2021
Year


Loxahatchee River-Lake Worth Creek Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.1
0.2
0.3
0.4
1991
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
1991
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
2011
2013
2015
2017
2019
2021
Year

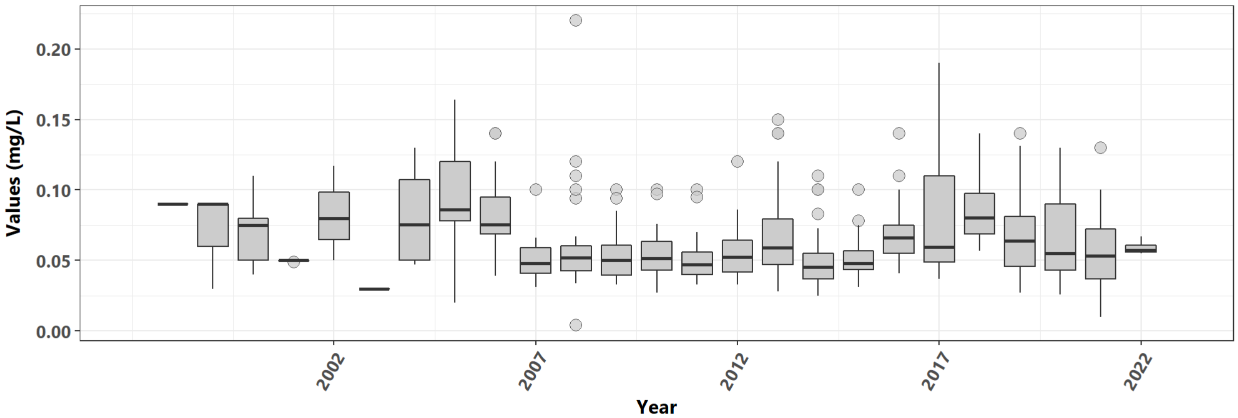
Loxahatchee River-Lake Worth Creek Aquatic Preserve

By Month

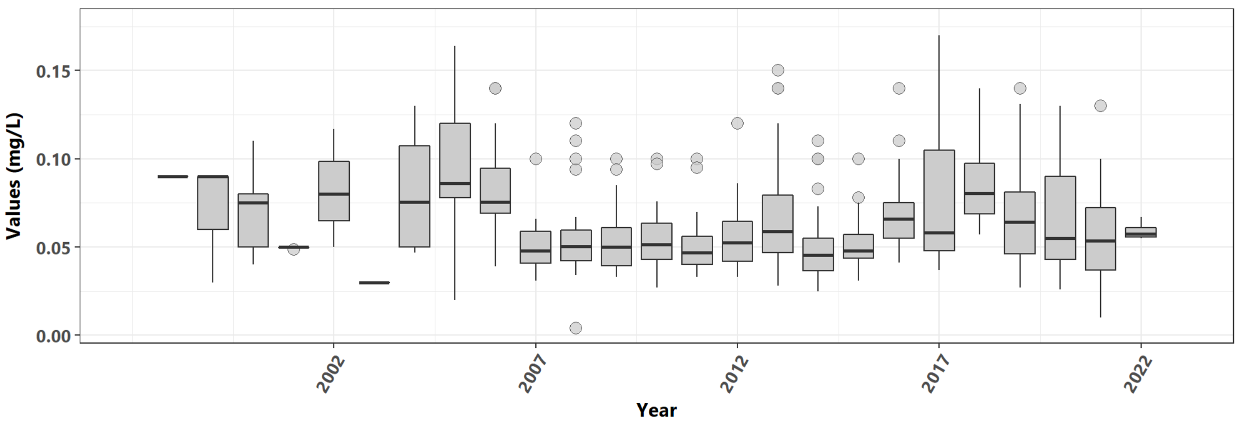


Matlacha Pass Aquatic Preserve
By Year

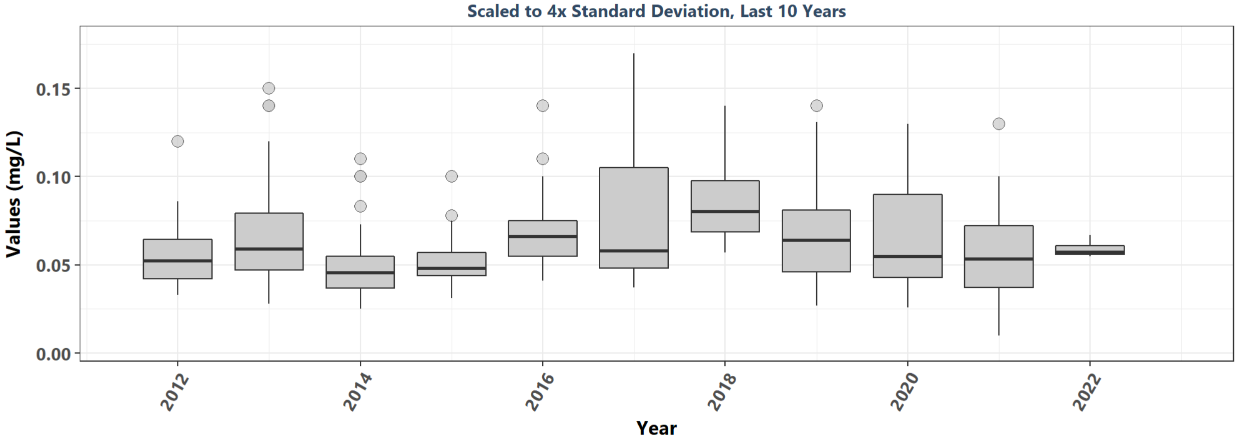
Autoscale

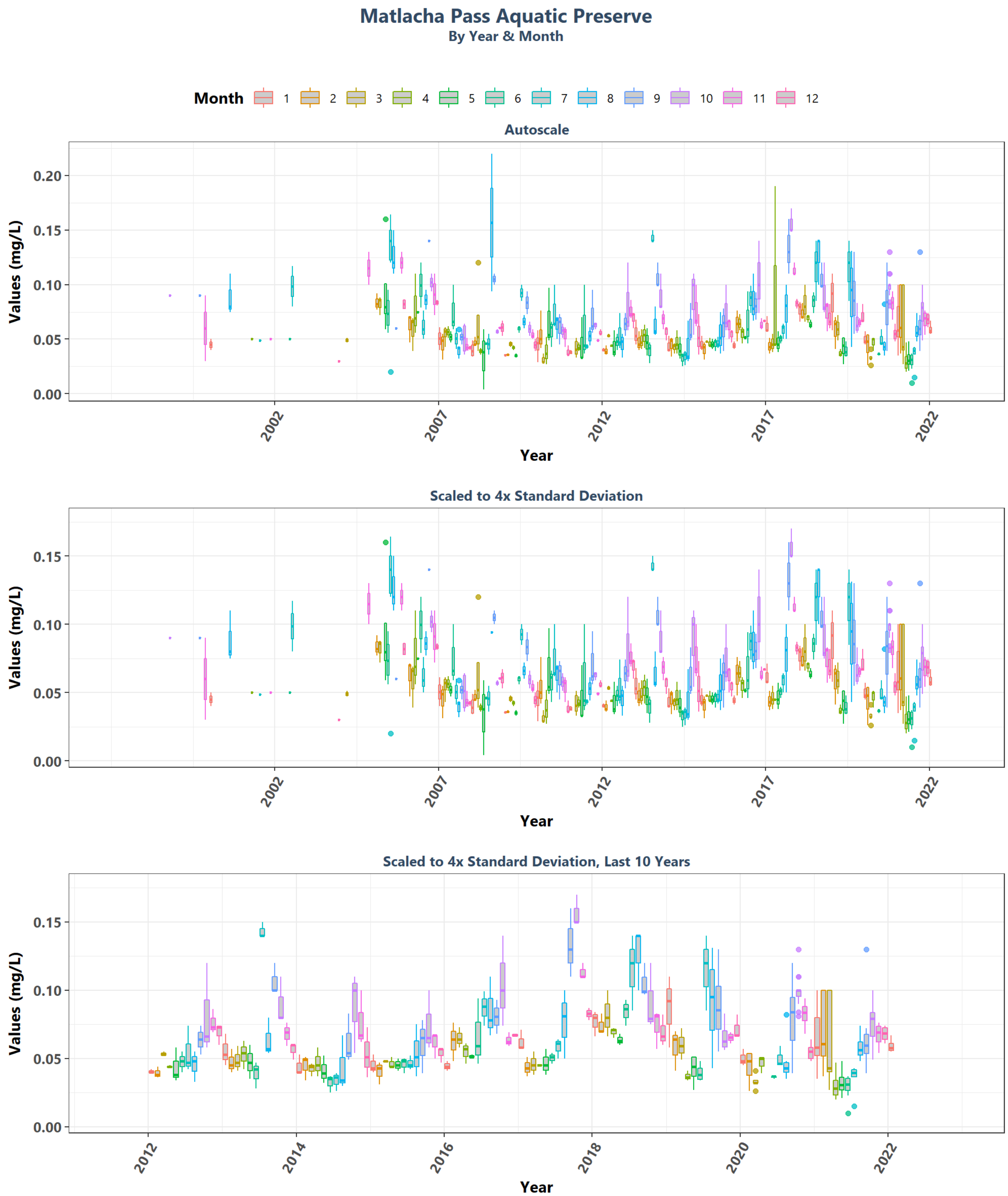


Scaled to 4x Standard Deviation

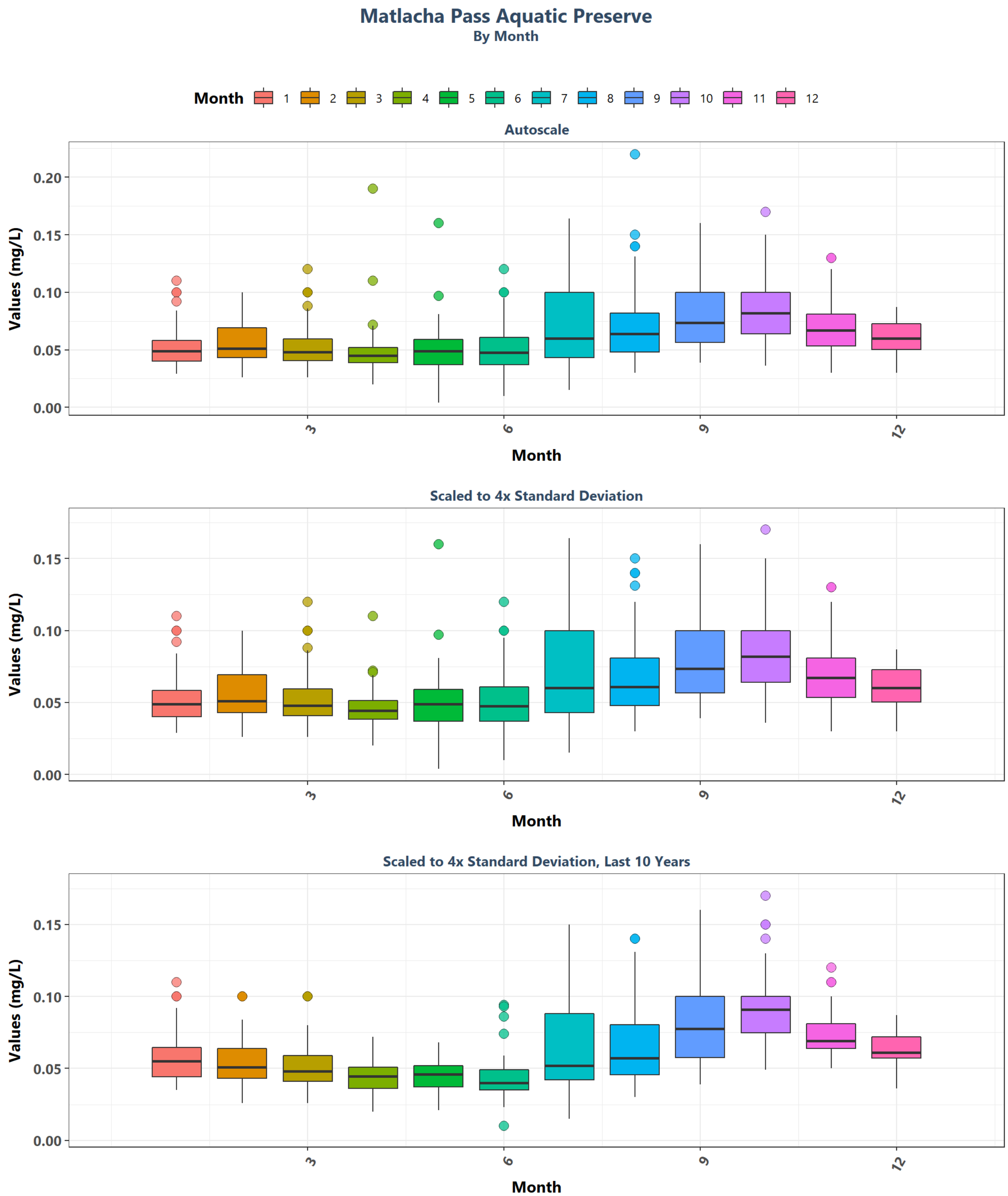


Scaled to 4x Standard Deviation, Last 10 Years

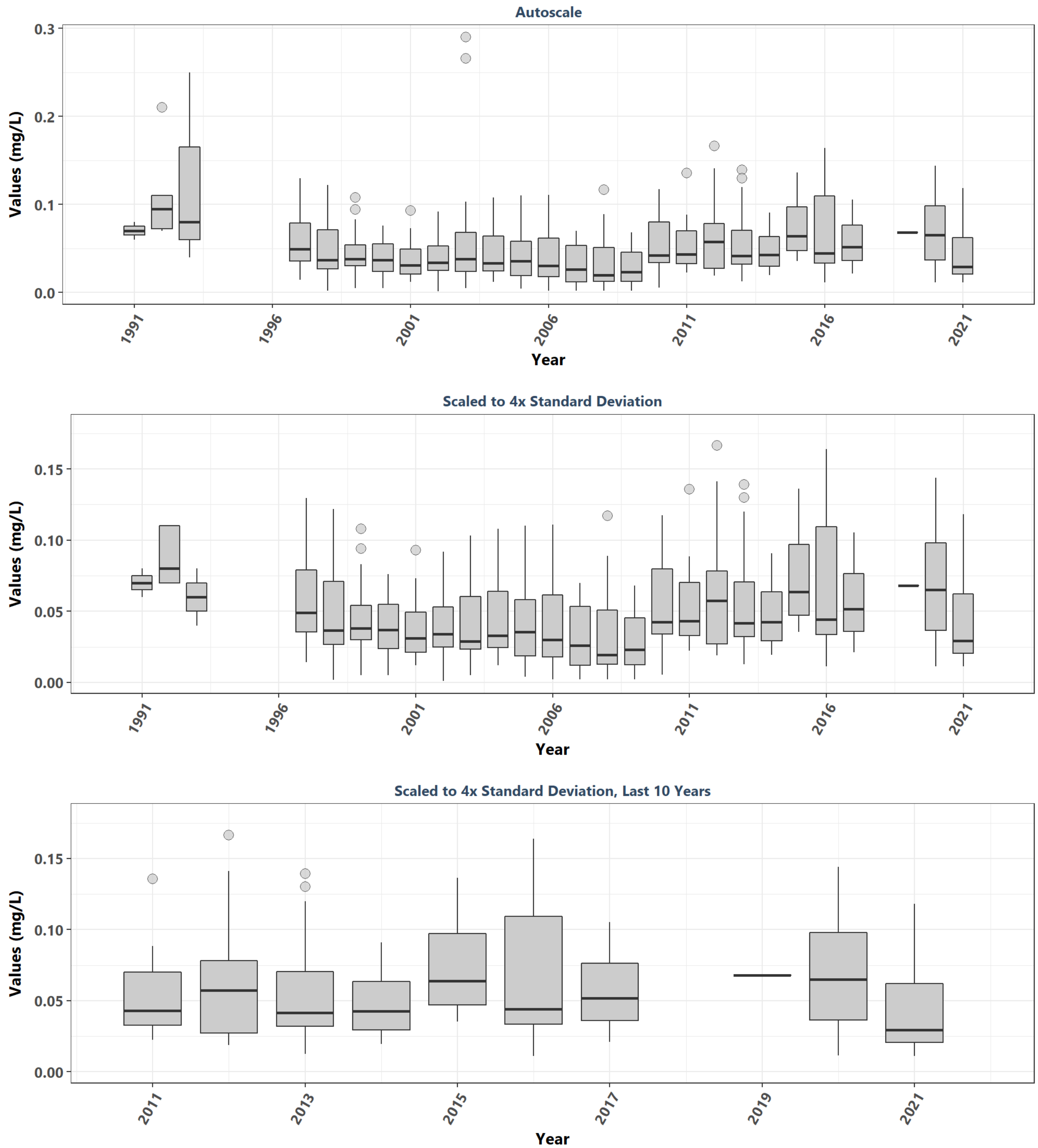




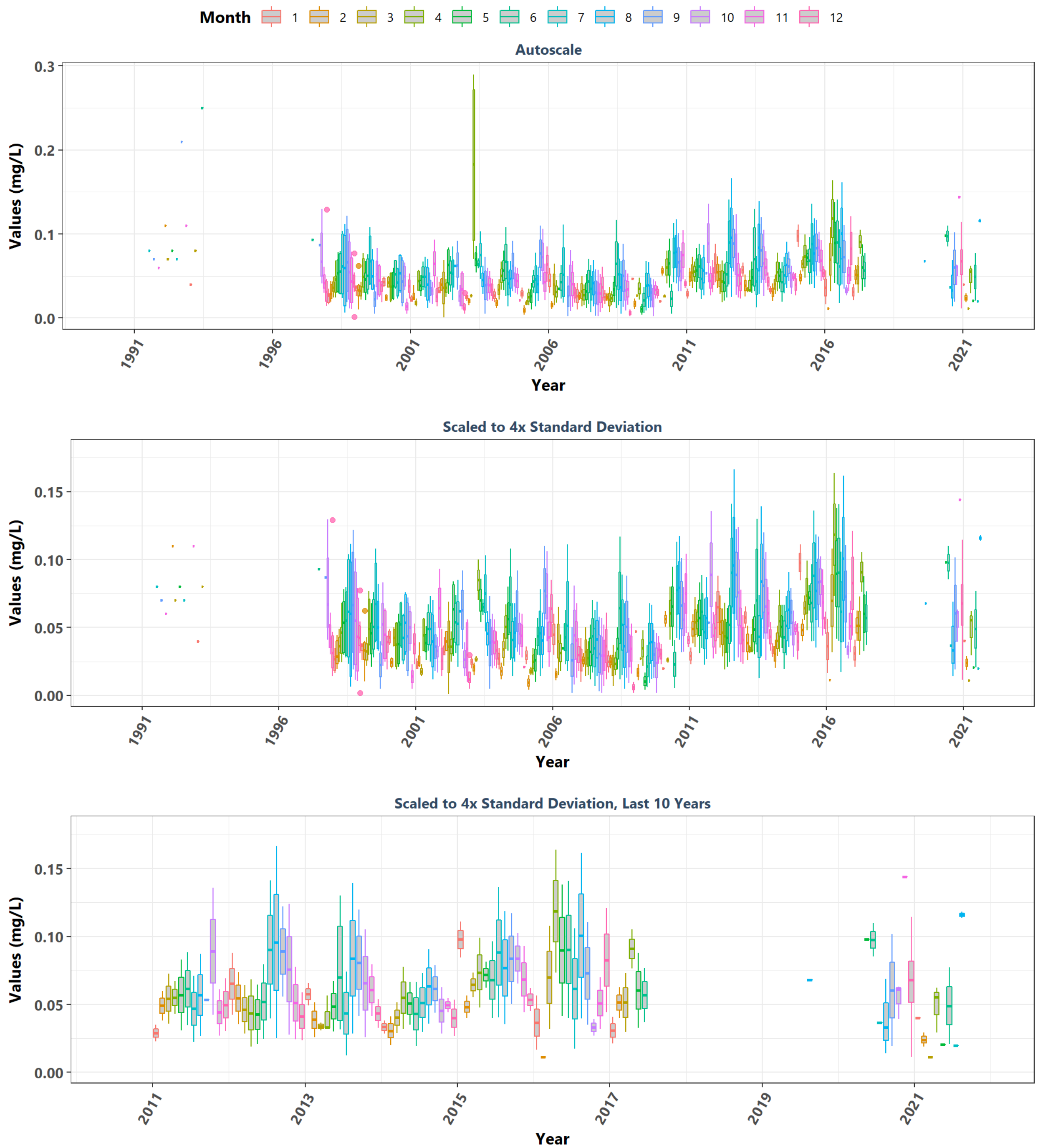
Matlacha Pass Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.00
0.05
0.10
0.15
0.20
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
2012
2014
2016
2018
2020
2022
Year


Matlacha Pass Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.00
0.05
0.10
0.15
0.20
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
3
6
9
12
Month

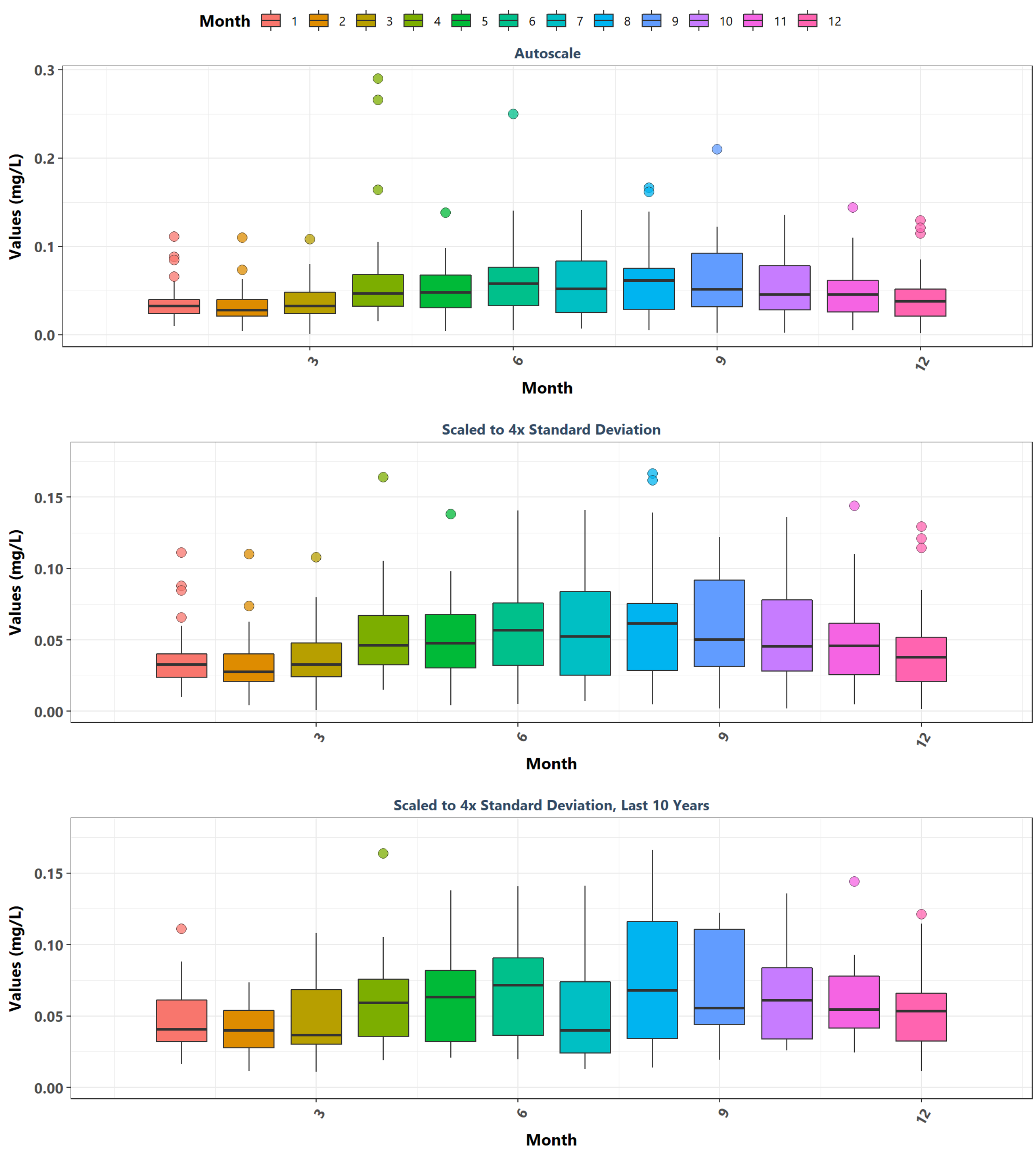
Mosquito Lagoon Aquatic Preserve
By Year



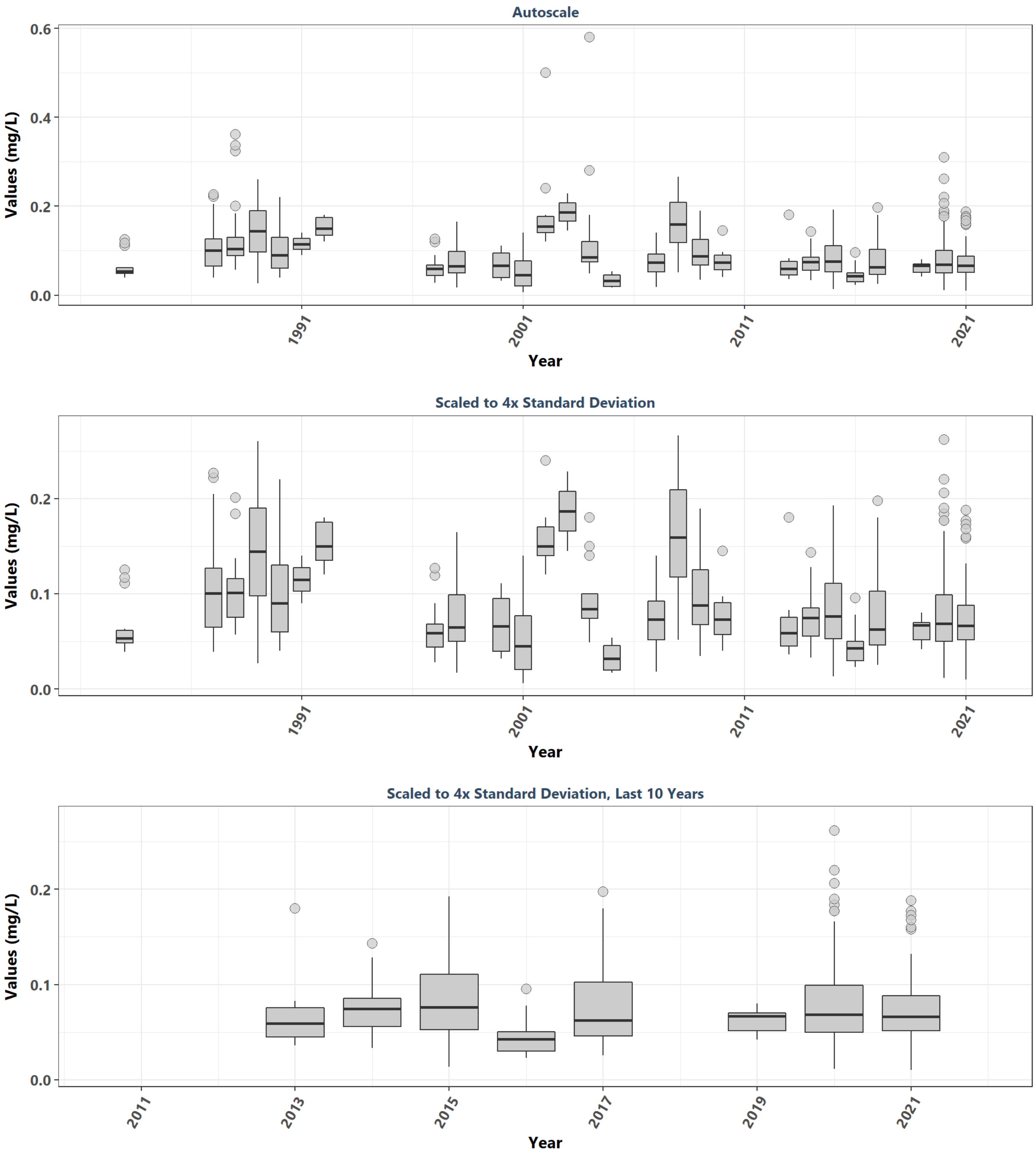
Mosquito Lagoon Aquatic Preserve
By Year & Month



Mosquito Lagoon Aquatic Preserve
By Month

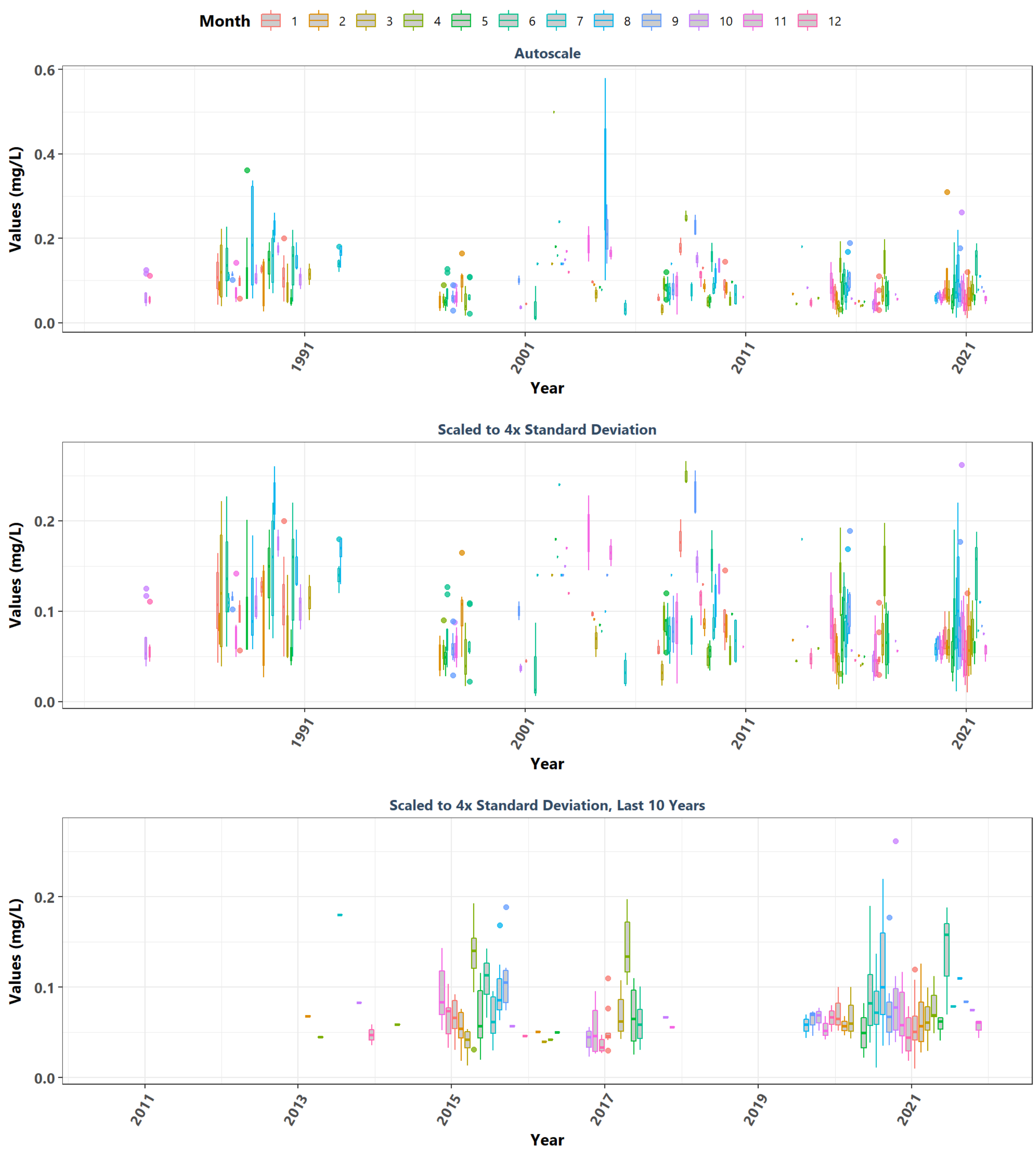


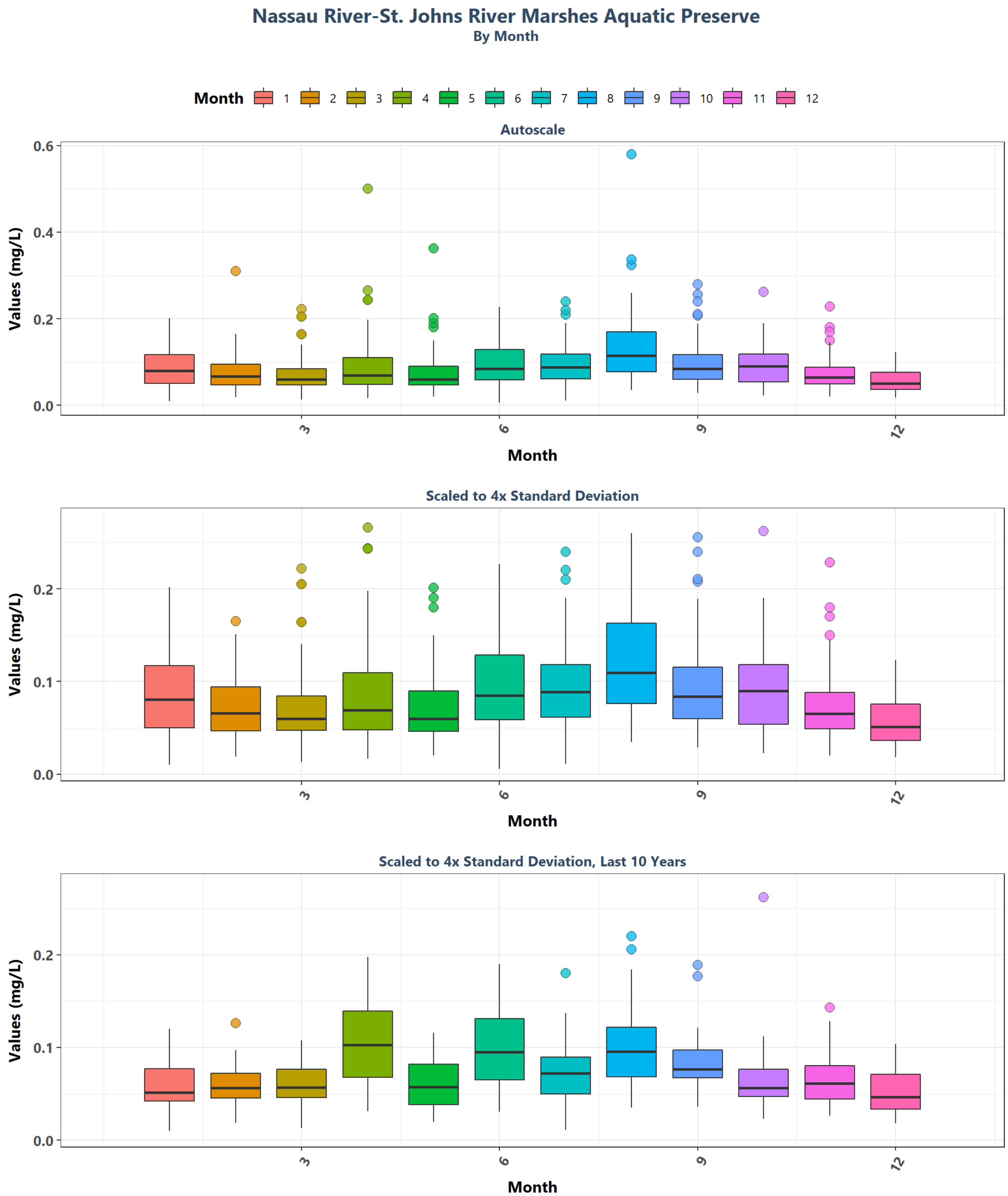
Nassau River-St. Johns River Marshes Aquatic Preserve
By Year



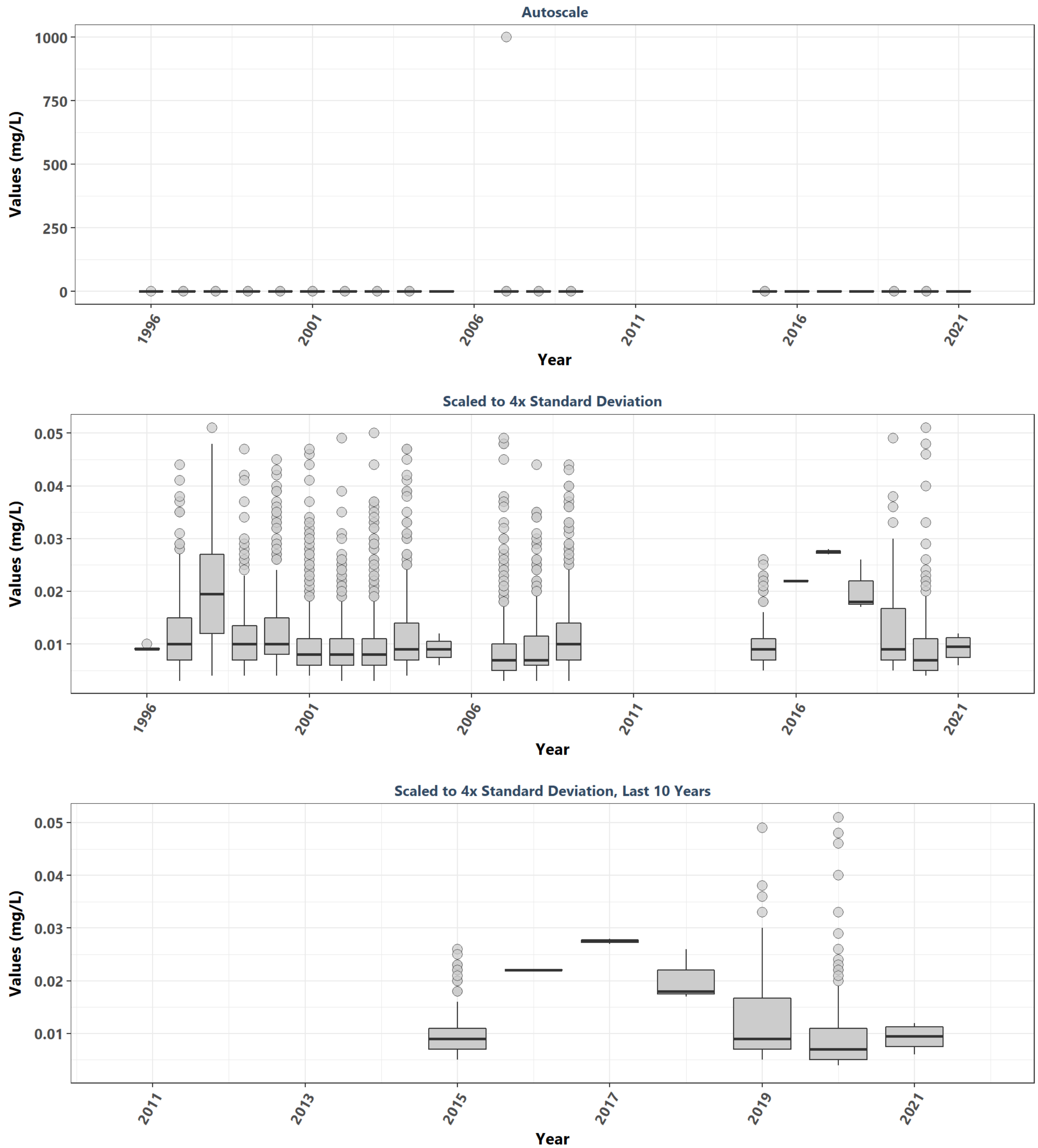
Nassau River-St. Johns River Marshes Aquatic Preserve

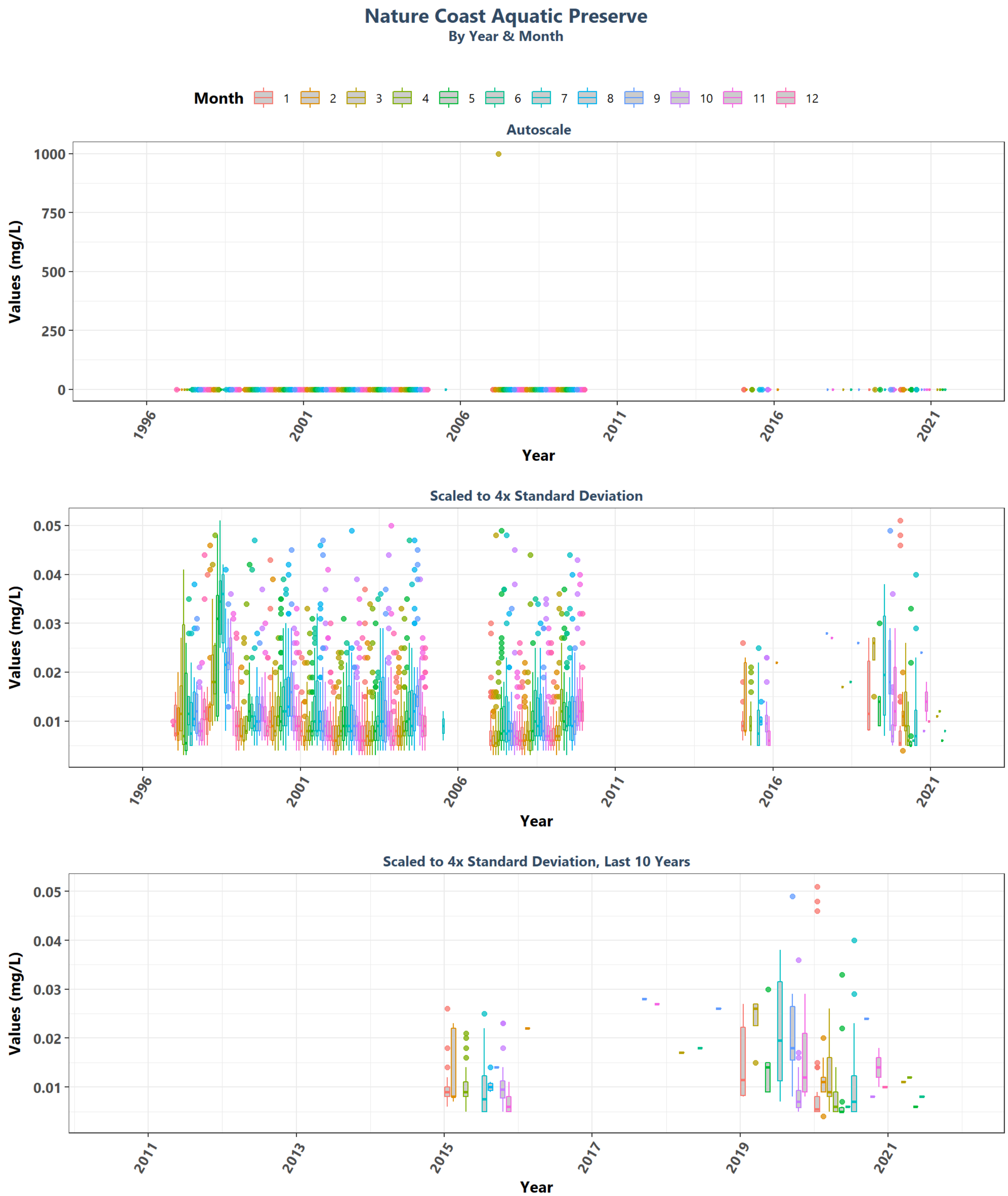
By Year & Month

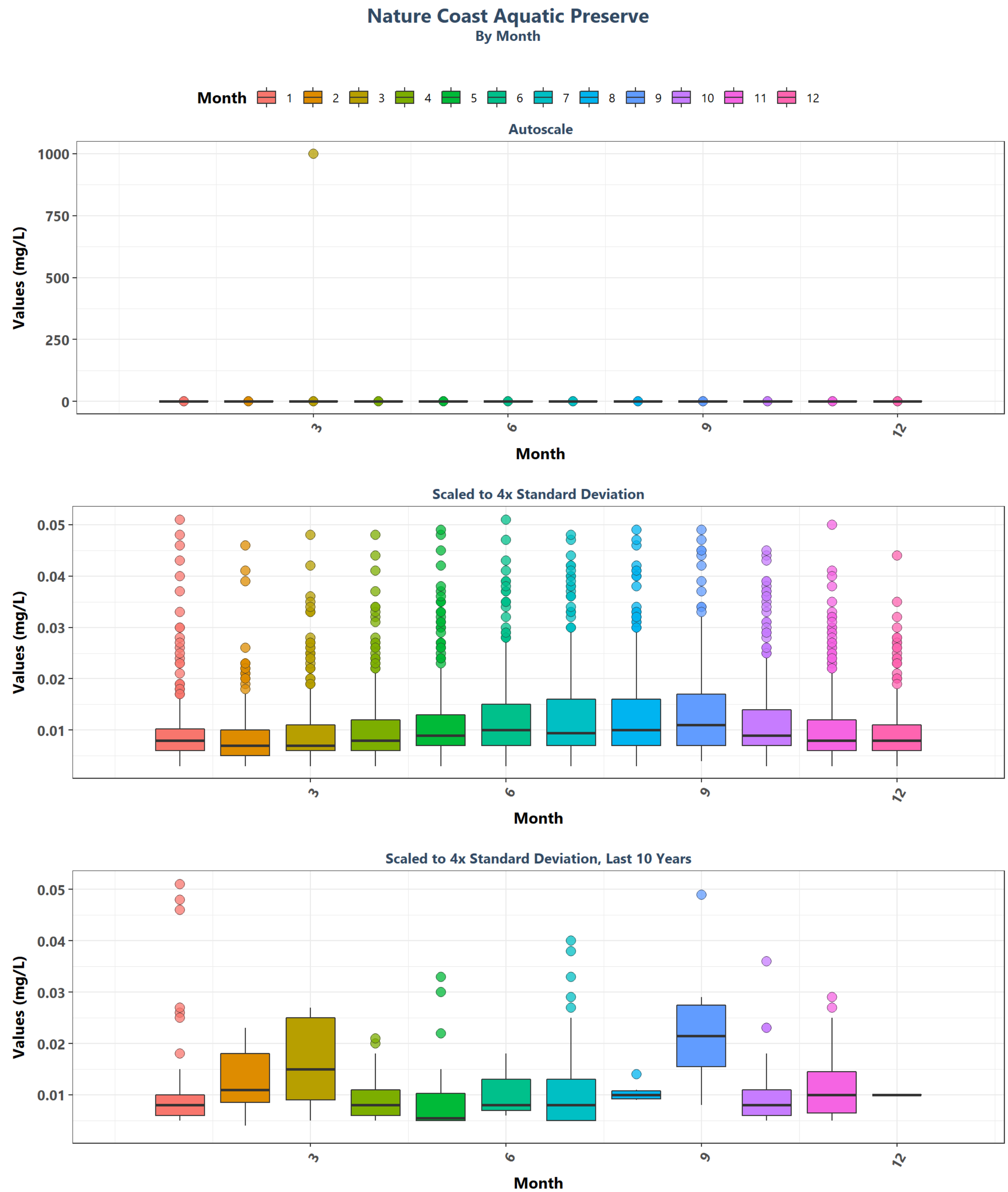



Nassau River-St. Johns River Marshes Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.2
0.4
0.6
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.0
0.1
0.2
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.0
0.1
0.2
3
6
9
12
Month

Nature Coast Aquatic Preserve
By Year

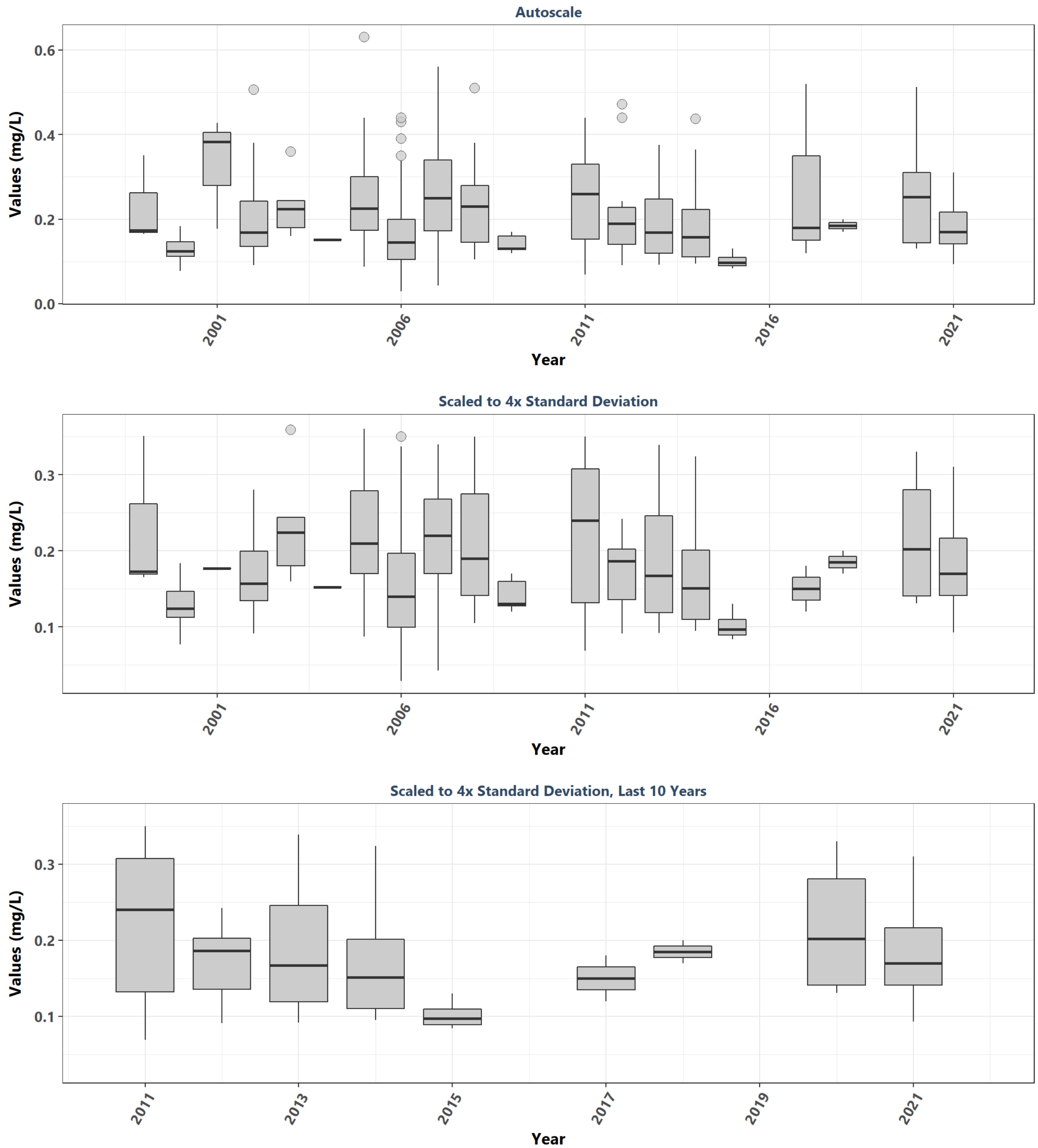



Nature Coast Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0
250
500
750
1000
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.01
0.02
0.03
0.04
0.05
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.01
0.02
0.03
0.04
0.05
2011
2013
2015
2017
2019
2021
Year

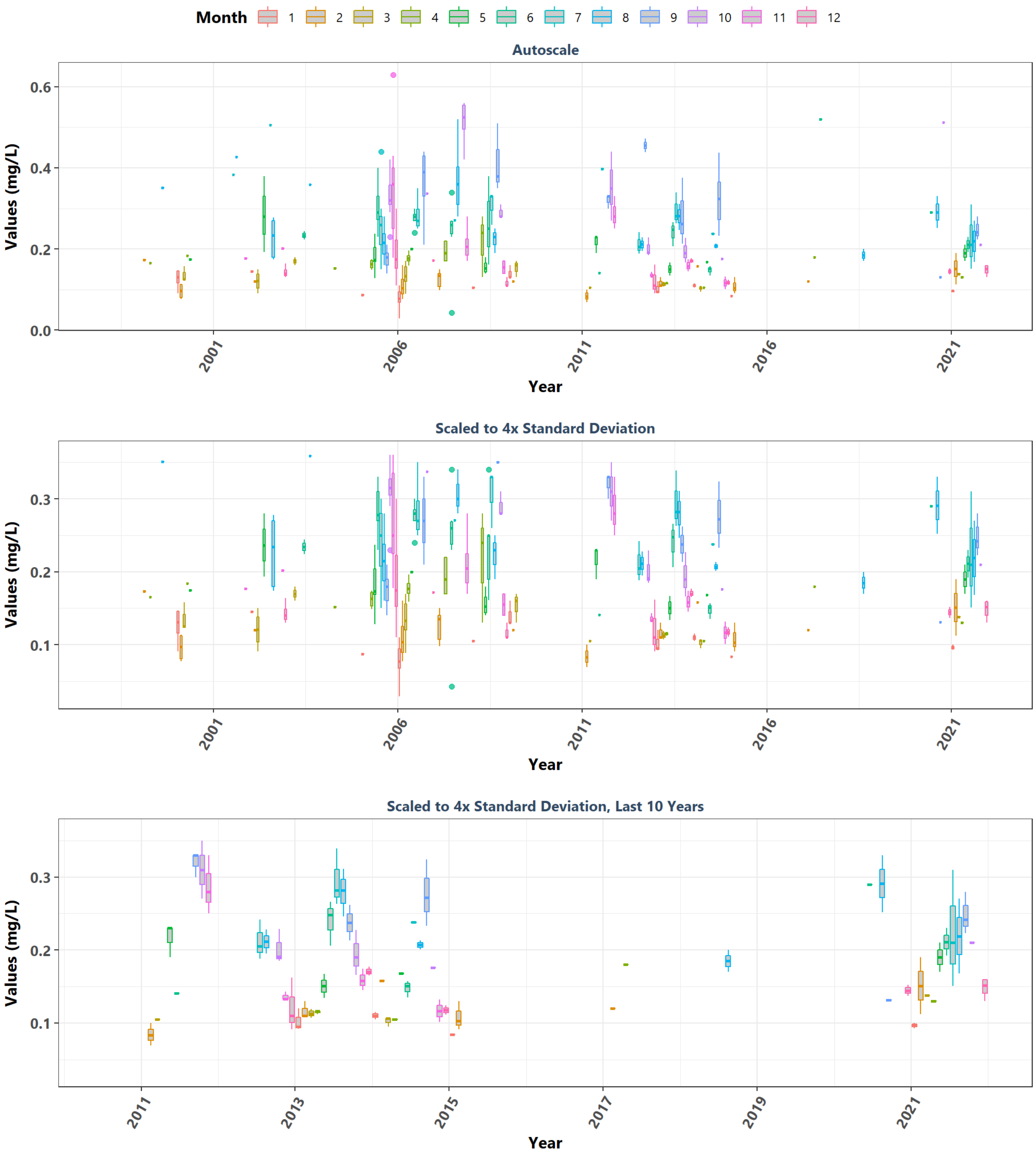

Nature Coast Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
1000
750
500
250
0
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.05
0.04
0.03
0.02
0.01
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.05
0.04
0.03
0.02
0.01
3
6
9
12
Month

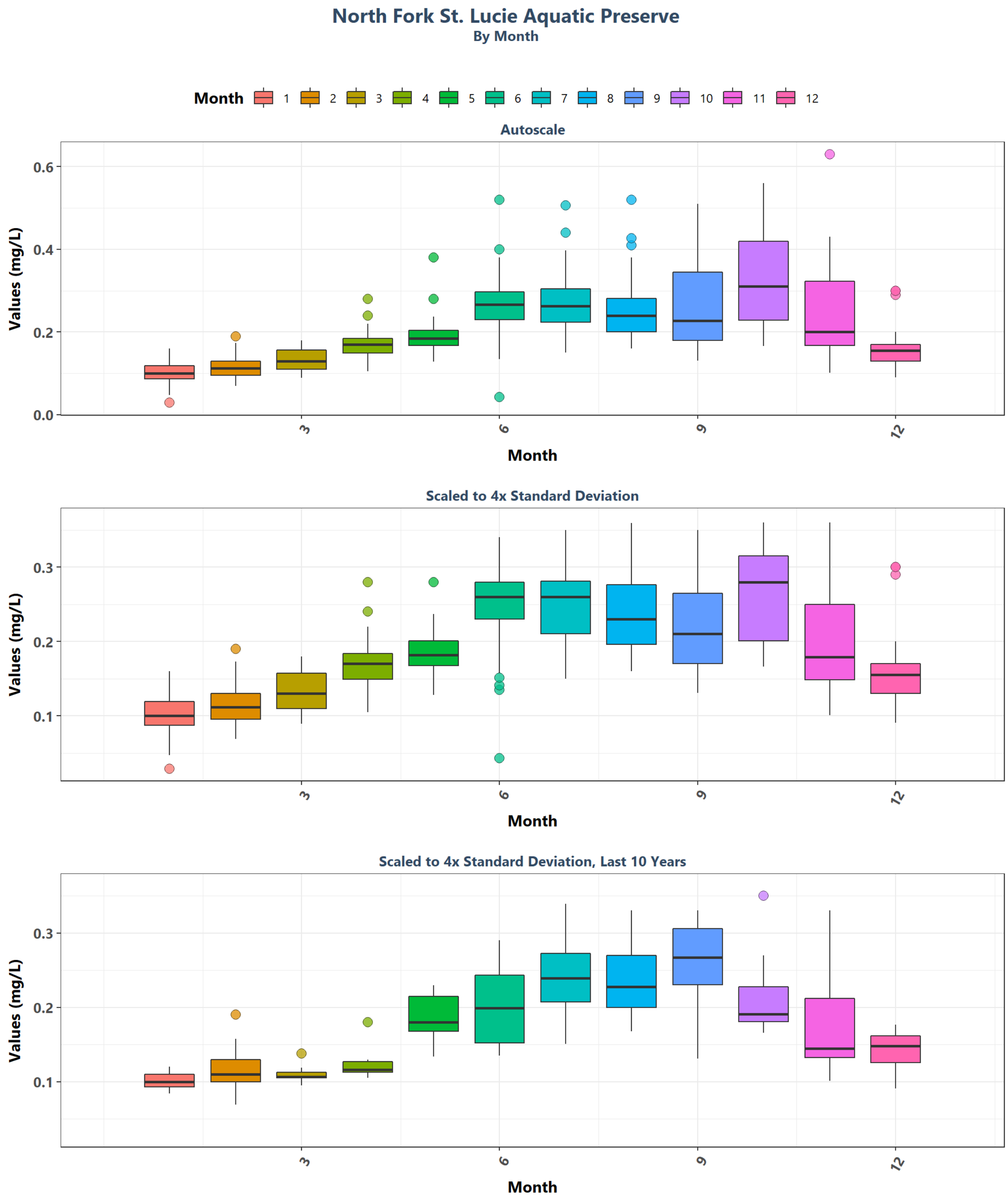
North Fork St. Lucie Aquatic Preserve

By Year

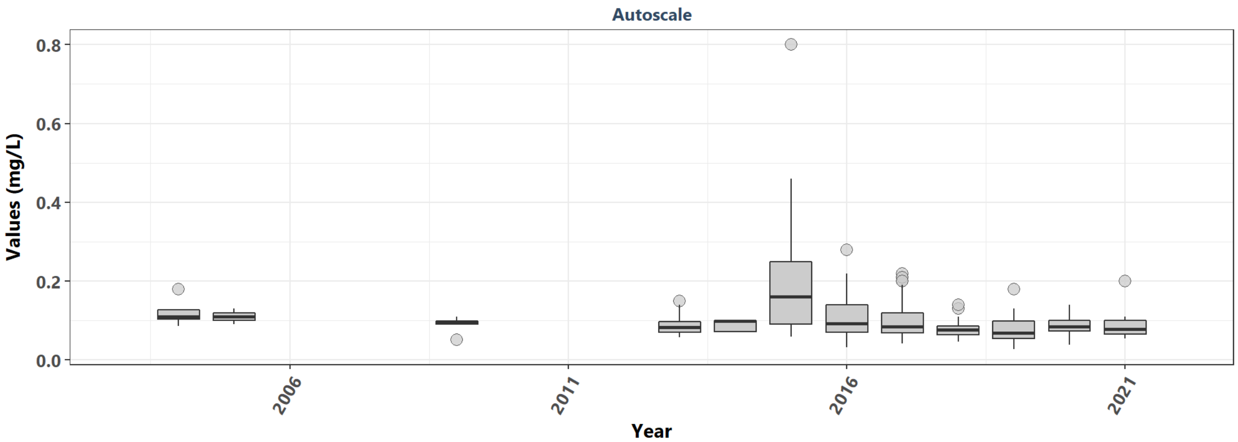


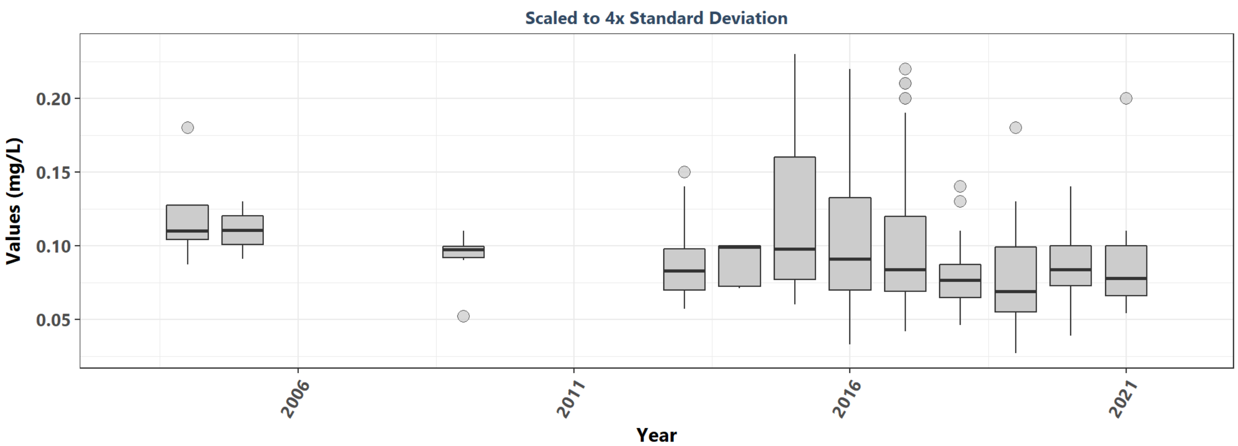
North Fork St. Lucie Aquatic Preserve
By Year & Month

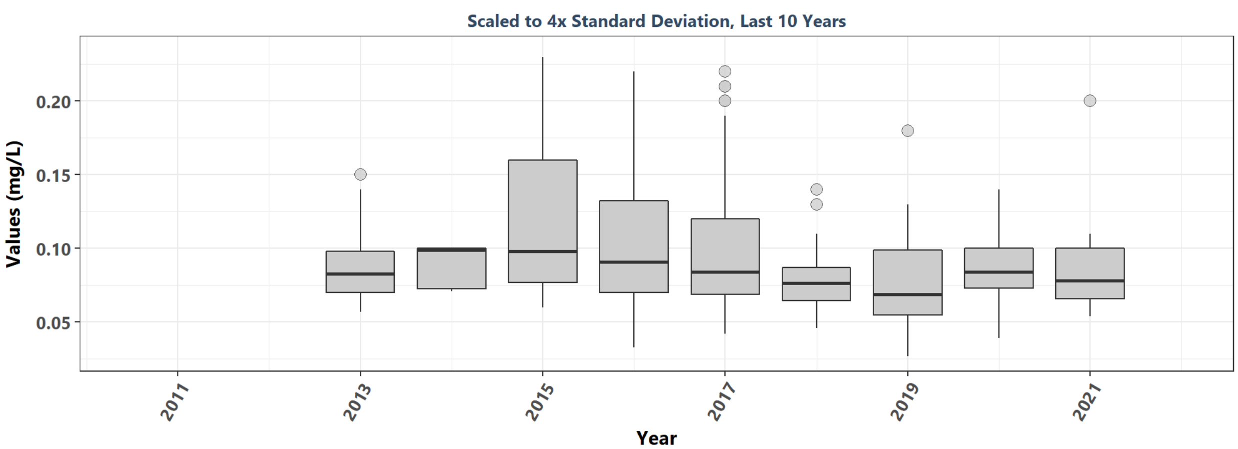


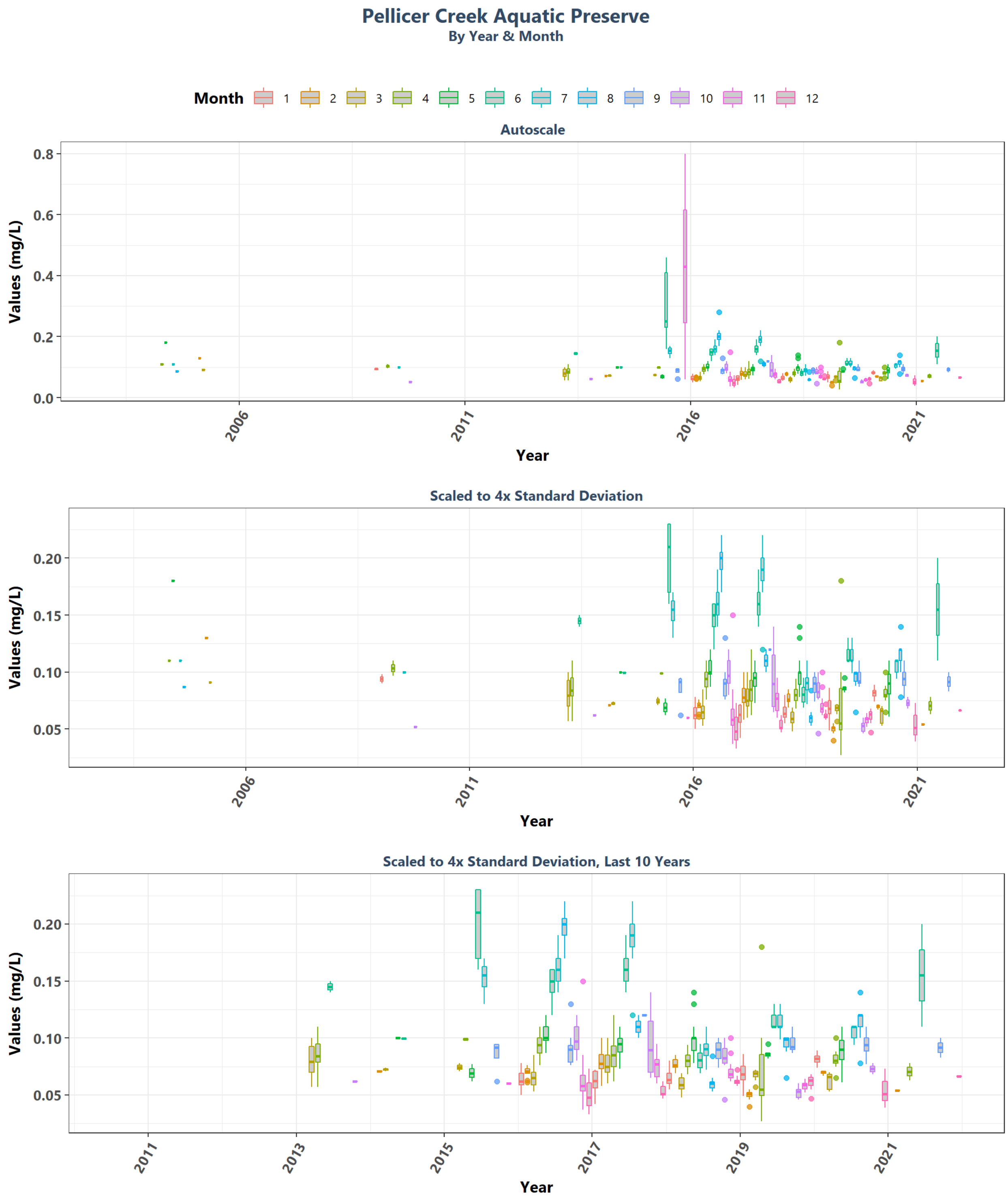

North Fork St. Lucie Aquatic Preserve
By Month
Month
1
2
3
4
5
6
7
8
9
10
11
12
Autoscale
Values (mg/L)
0.0
0.2
0.4
0.6
Month
3
6
9
12
Scaled to 4x Standard Deviation
Values (mg/L)
0.1
0.2
0.3
Month
3
6
9
12
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.1
0.2
0.3
Month
3
6
9
12

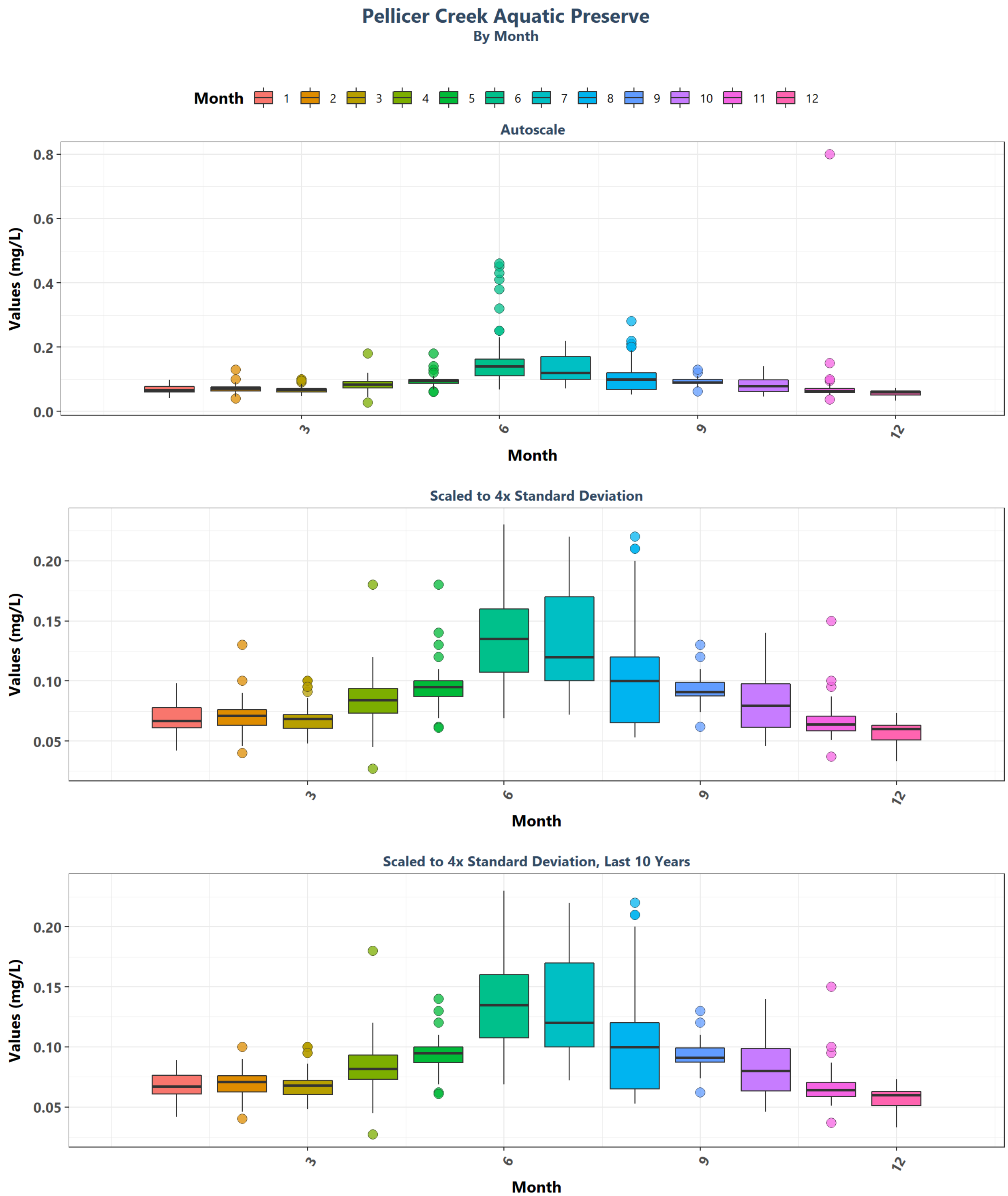
Pellicer Creek Aquatic Preserve
By Year





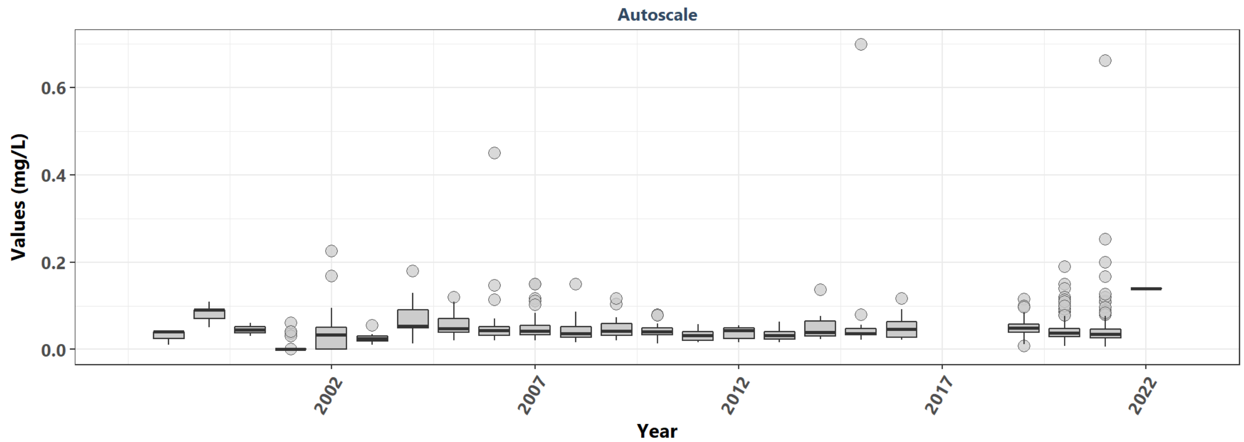


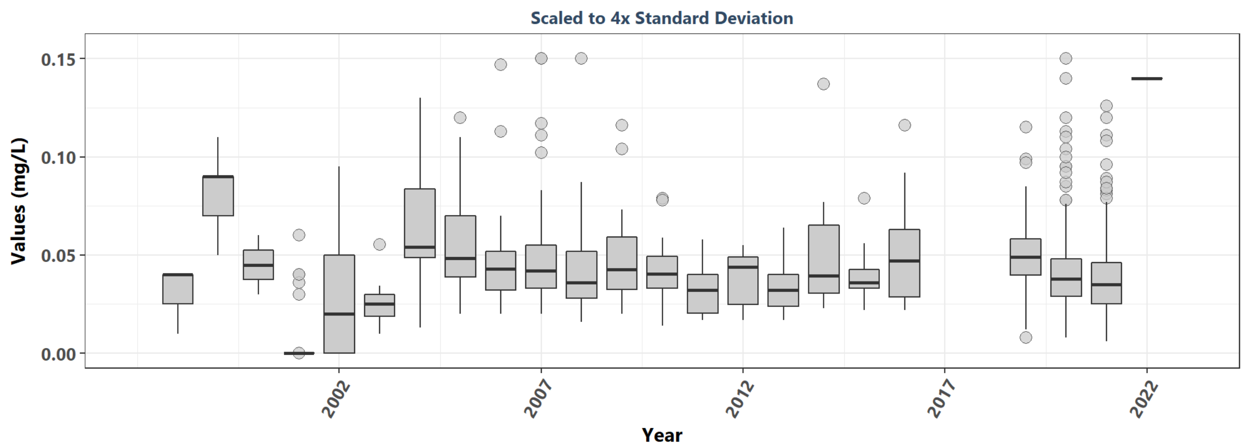

Pellicer Creek Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.2
0.4
0.6
0.8
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.05
0.10
0.15
0.20
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.05
0.10
0.15
0.20
2011
2013
2015
2017
2019
2021
Year

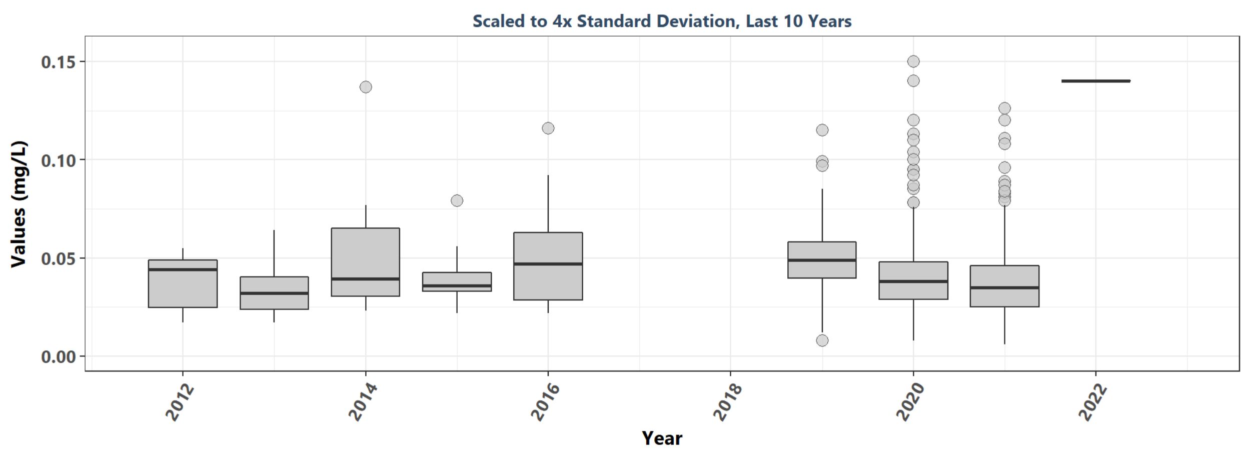


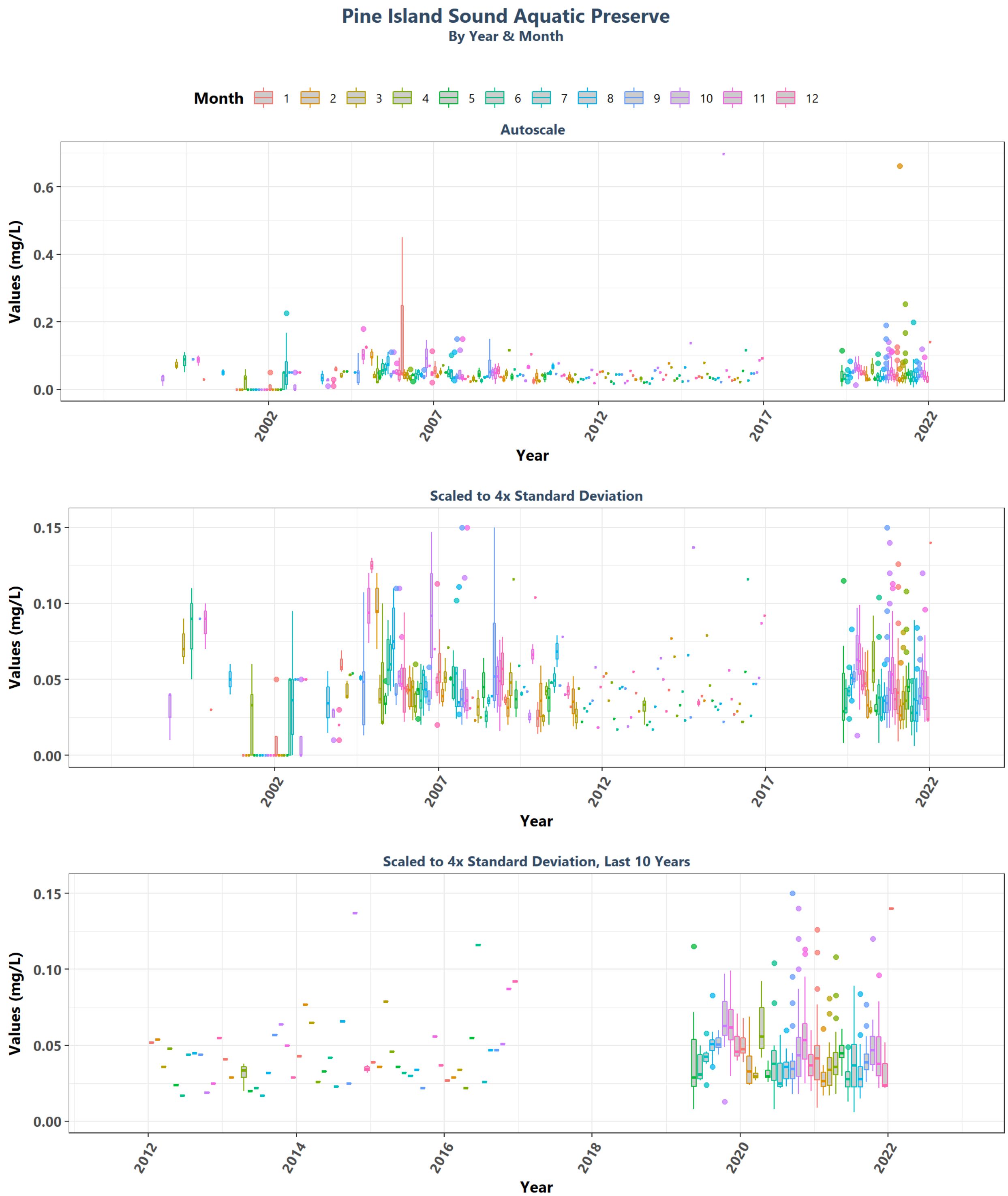
Pellicer Creek Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.2
0.4
0.6
0.8
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.05
0.10
0.15
0.20
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.05
0.10
0.15
0.20
3
6
9
12
Month

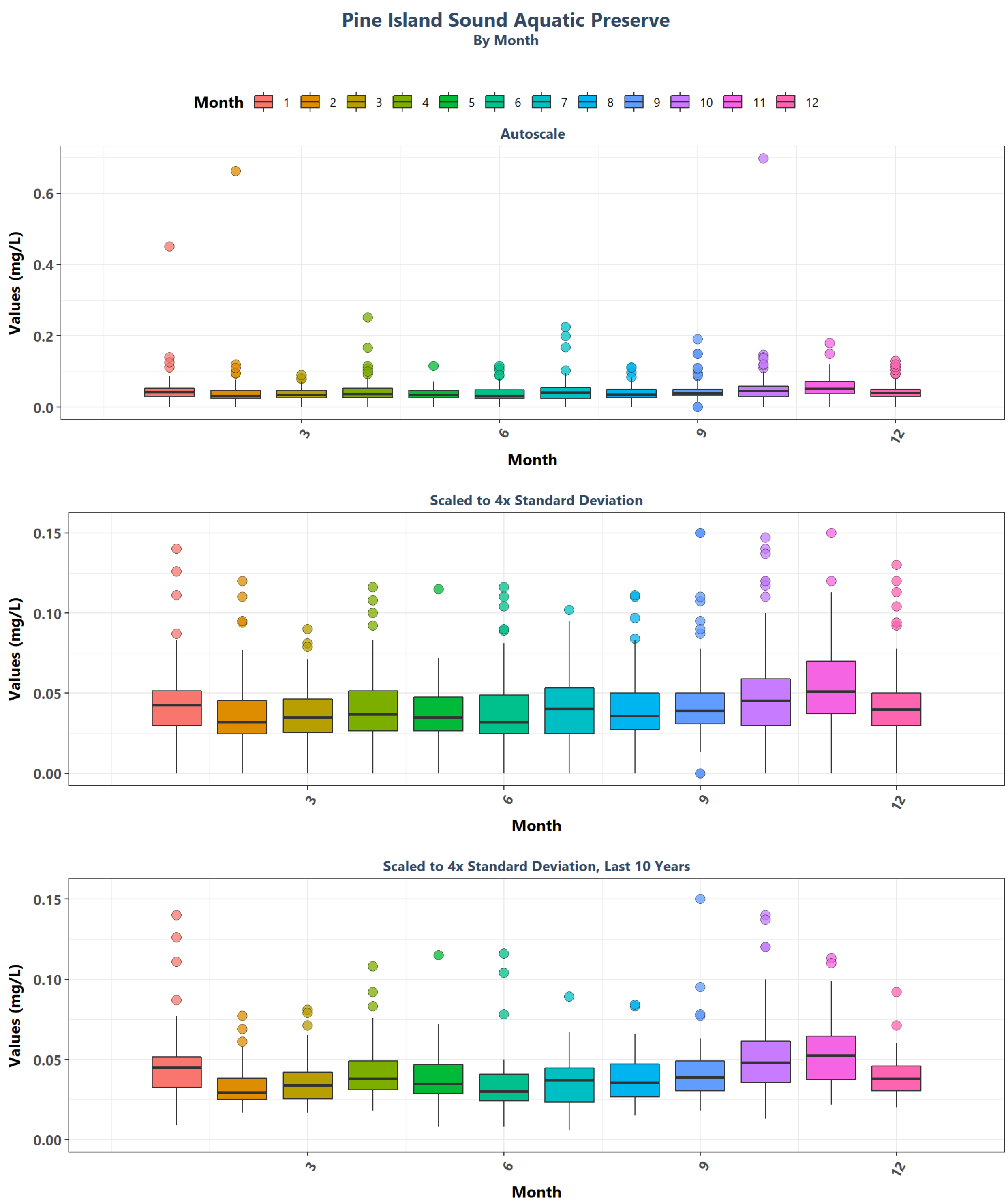
Pine Island Sound Aquatic Preserve
By Year



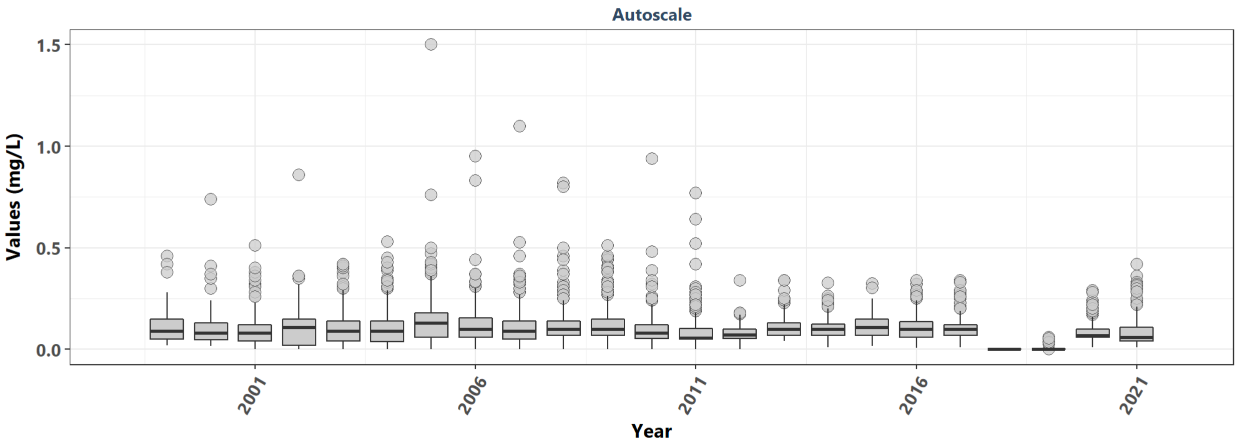


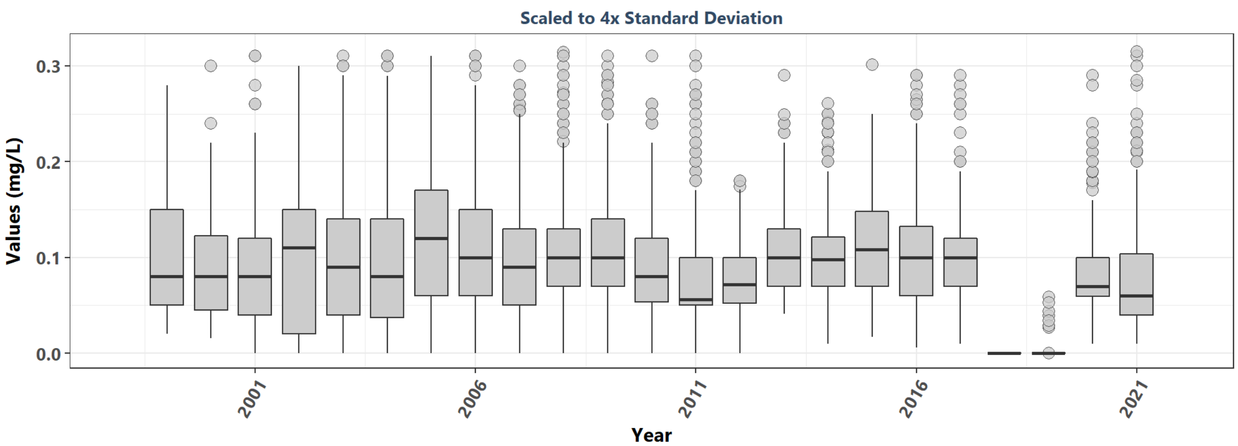


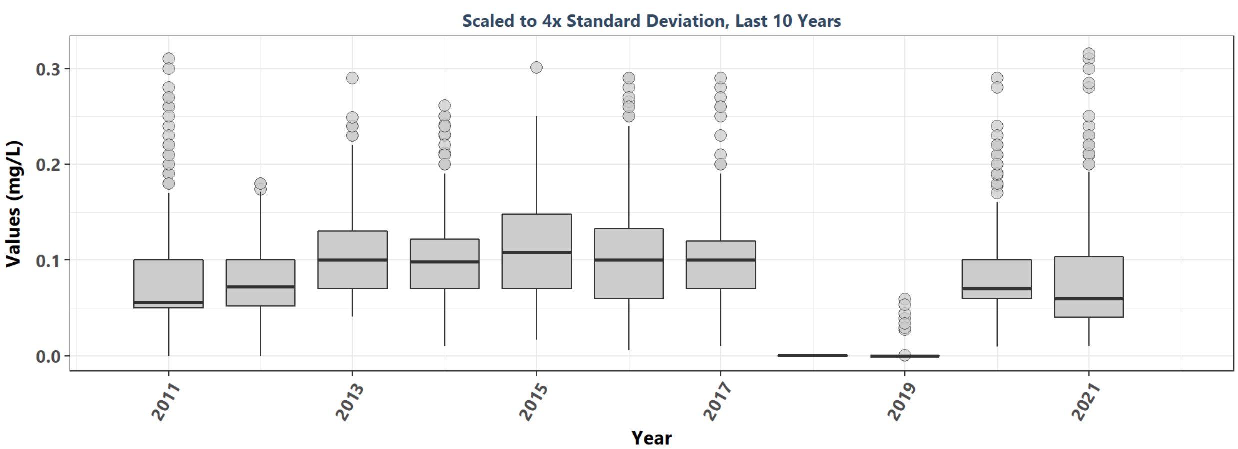

Pine Island Sound Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.2
0.4
0.6
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
2012
2014
2016
2018
2020
2022
Year

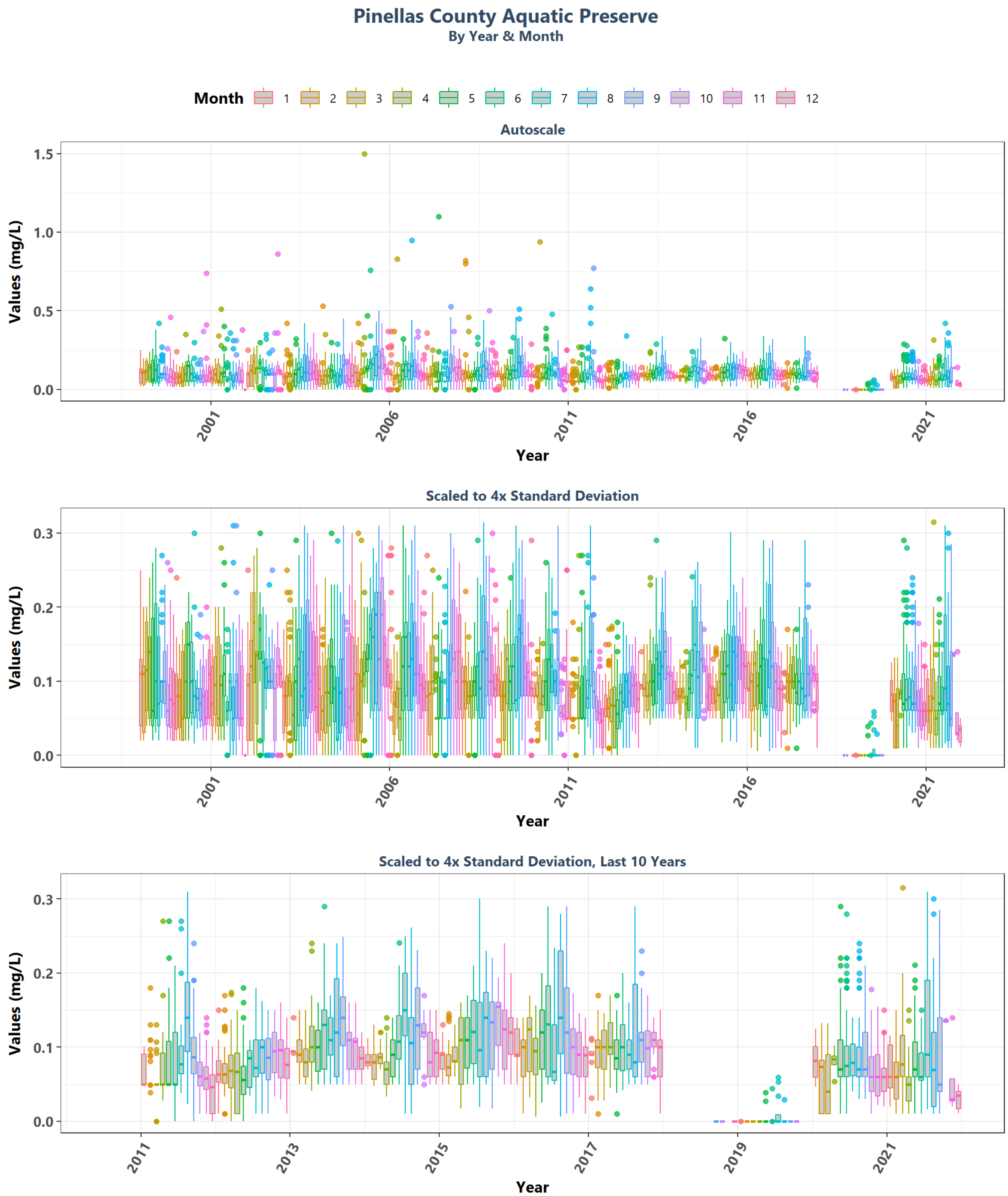

Pine Island Sound Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.2
0.4
0.6
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
0.15
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
0.15
3
6
9
12
Month

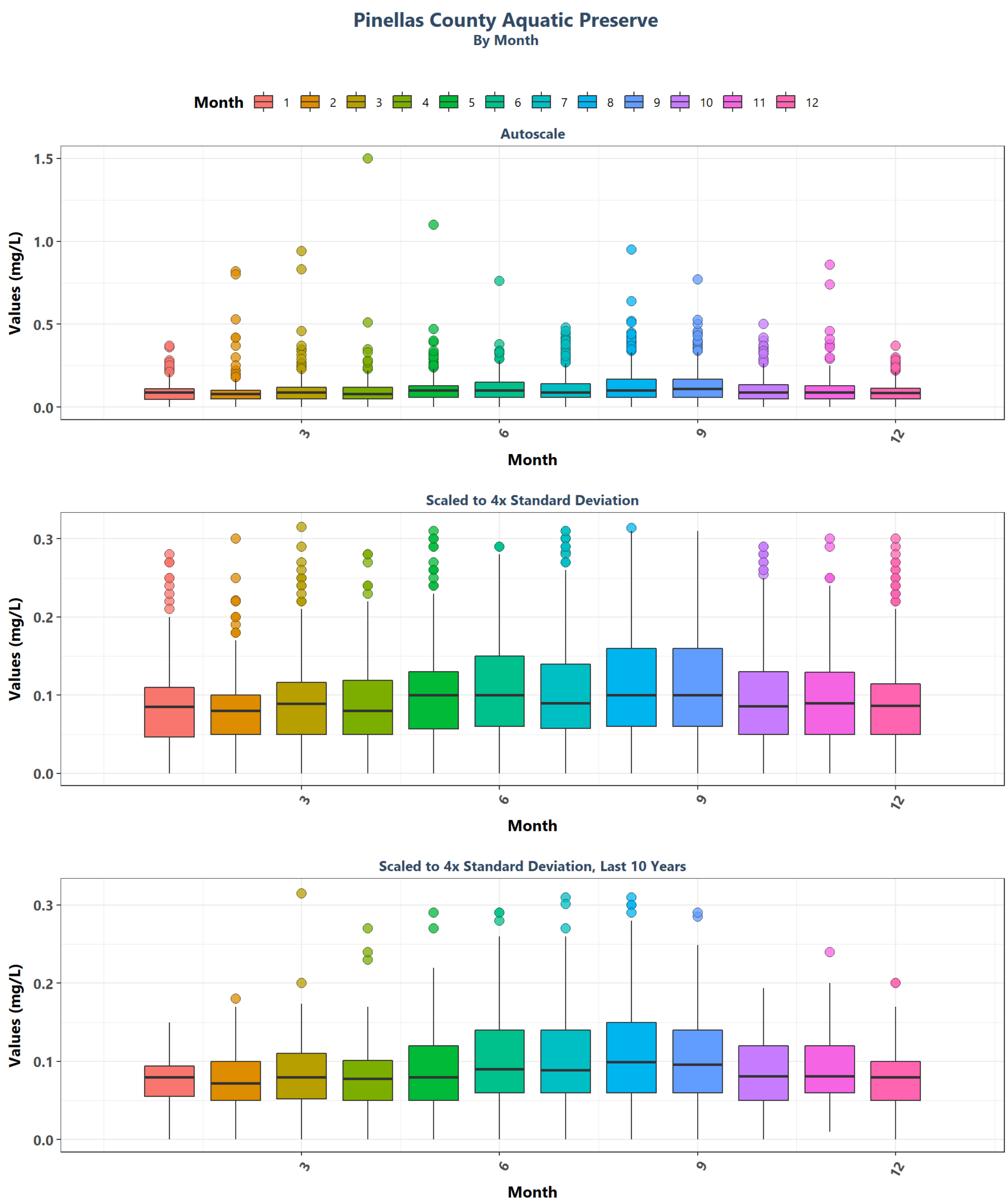
Pinellas County Aquatic Preserve
By Year







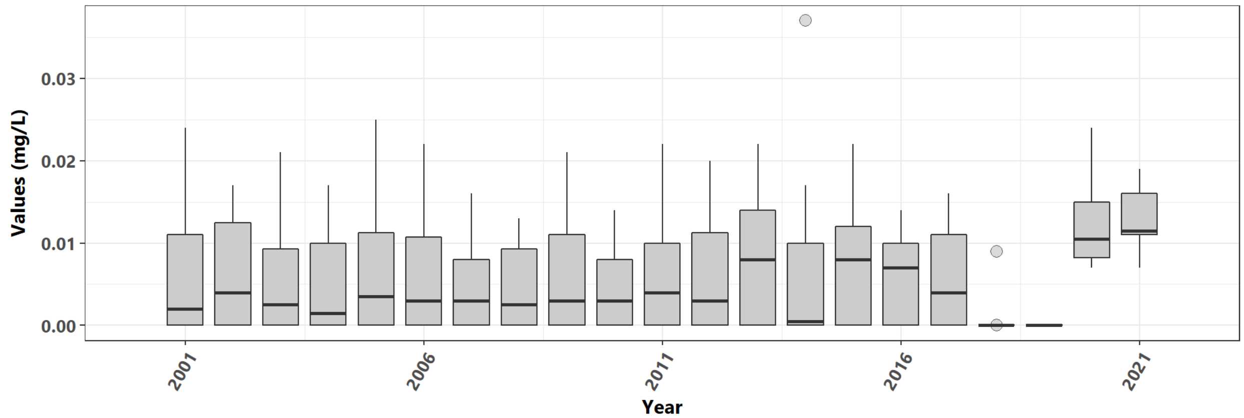

Pinellas County Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
1.5
1.0
0.5
0.0
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
0.3
0.2
0.1
0.0
Scaled to 4x Standard Deviation, Last 10 Years
2011
2013
2015
2017
2019
2021



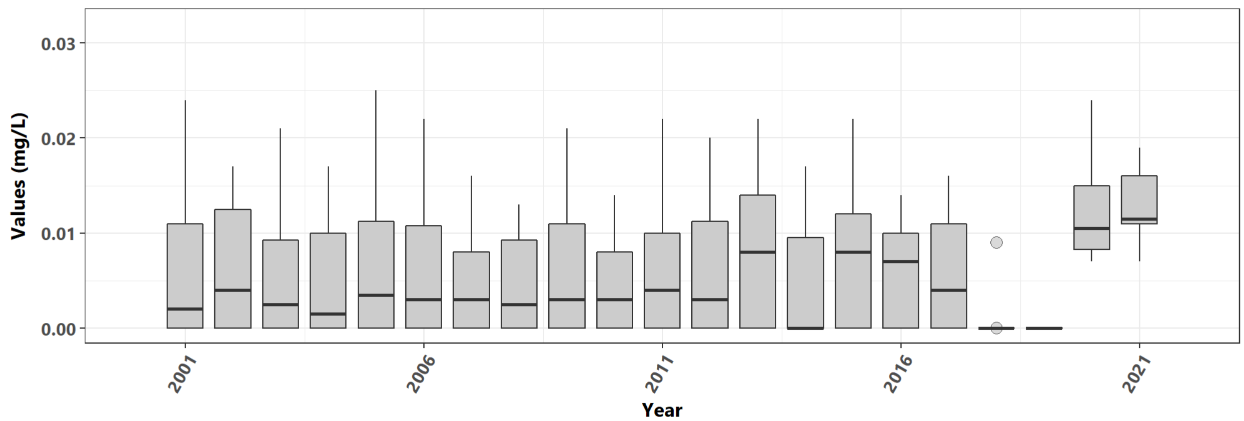
Pinellas County Aquatic Preserve
By Month
Month
1
2
3
4
5
6
7
8
9
10
11
12
Autoscale
Values (mg/L)
1.5
1.0
0.5
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.3
0.2
0.1
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.3
0.2
0.1
0.0
3
6
9
12
Month

Rocky Bayou State Park Aquatic Preserve
By Year

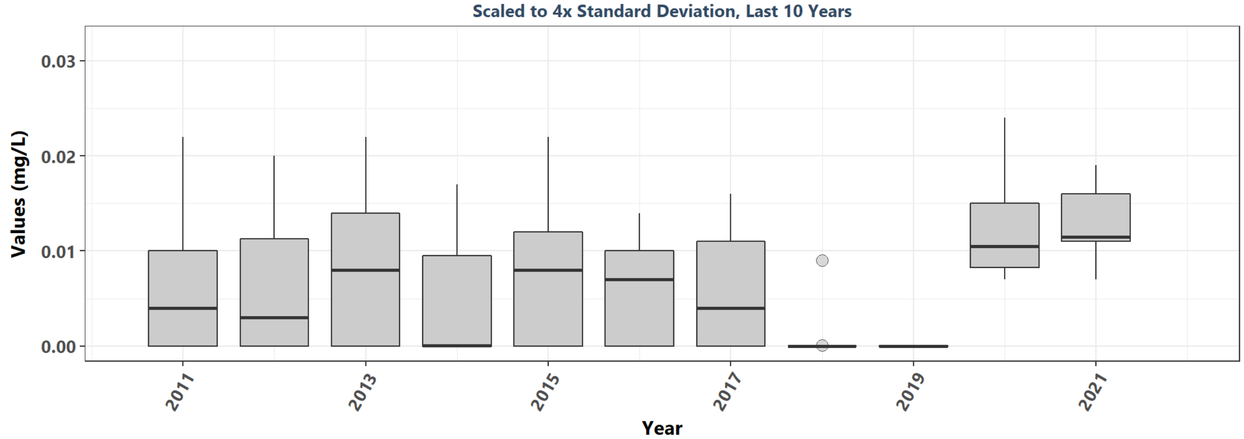
Autoscale



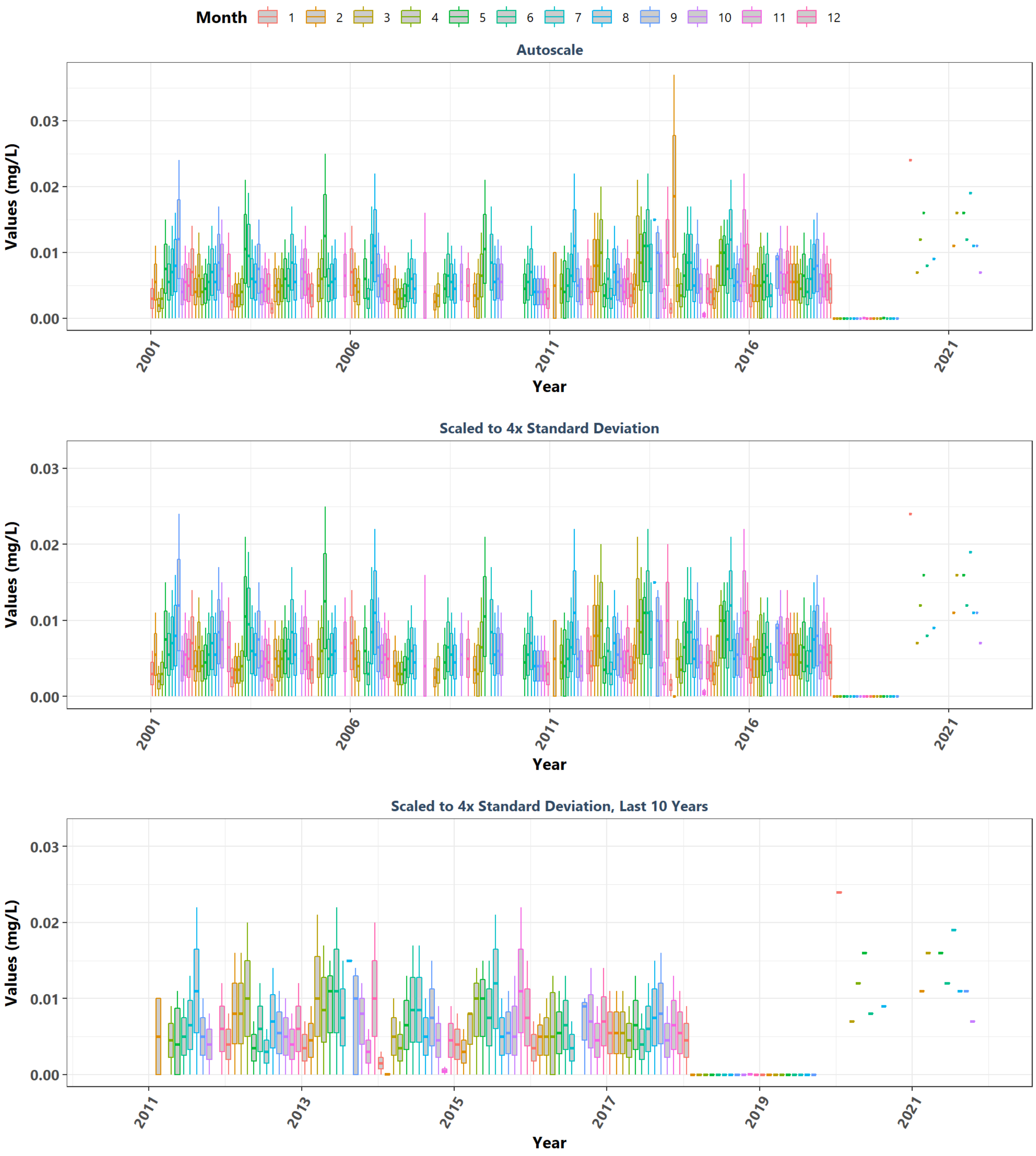
Scaled to 4x Standard Deviation

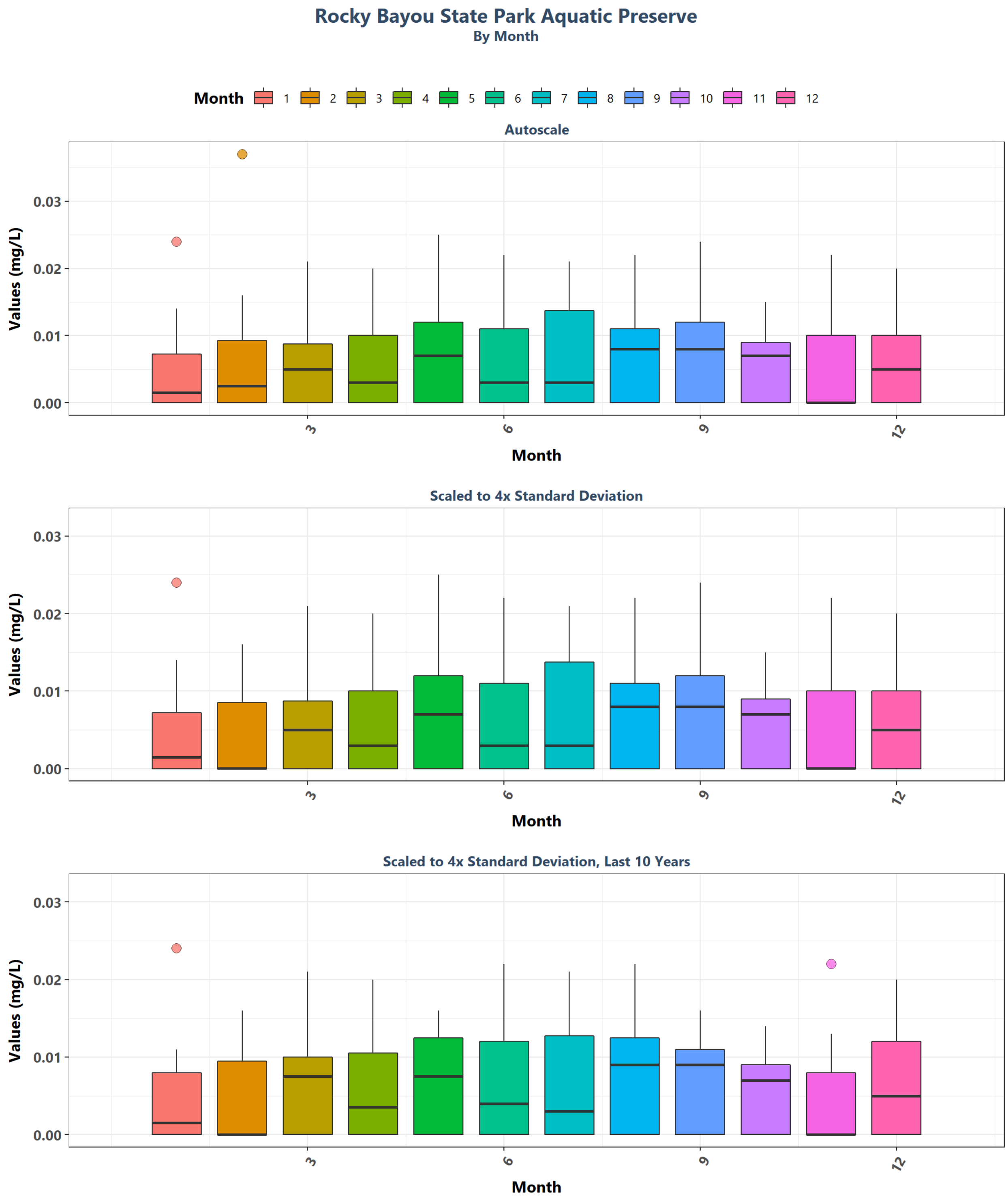


Scaled to 4x Standard Deviation, Last 10 Years



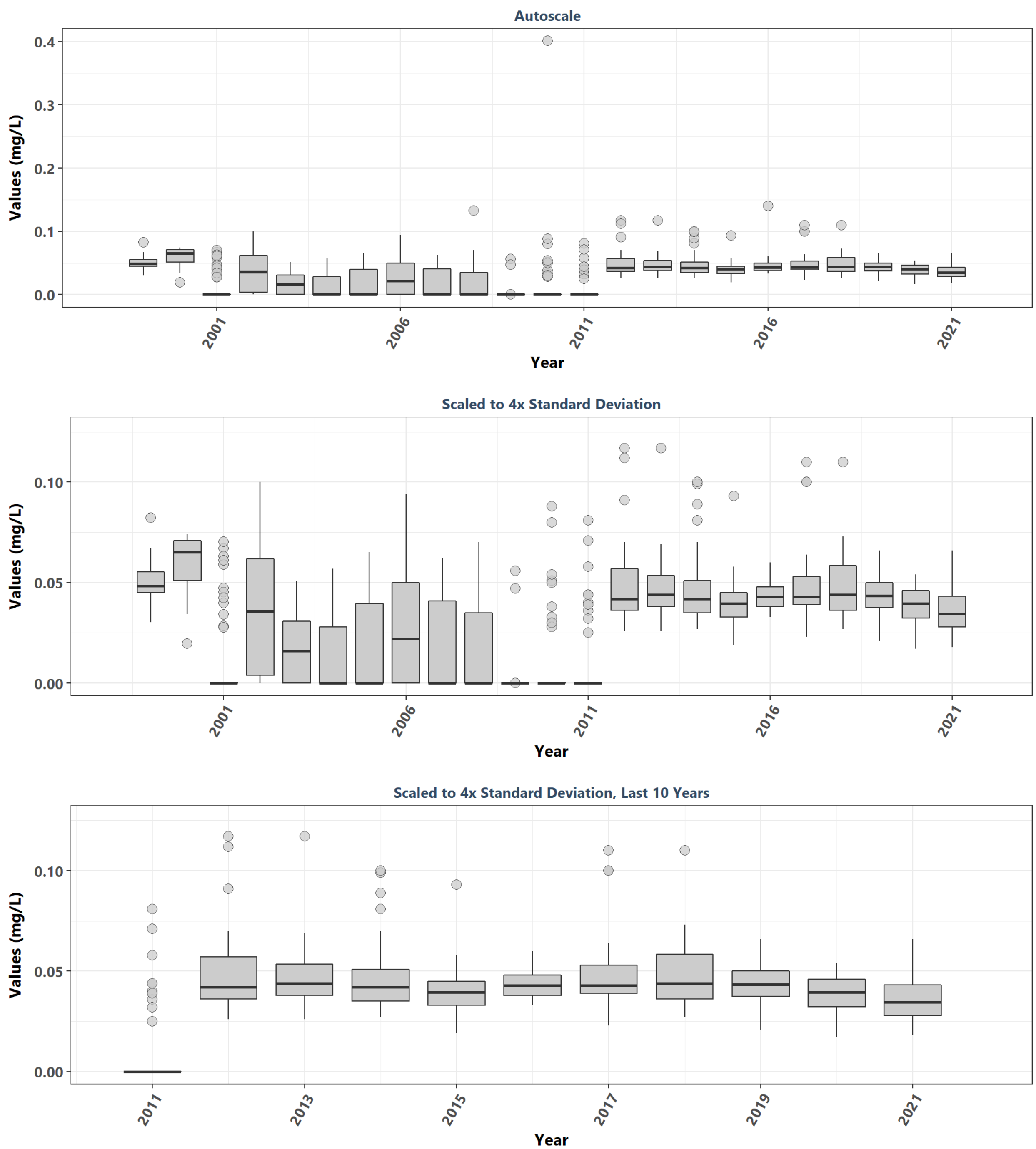
Rocky Bayou State Park Aquatic Preserve
By Year & Month

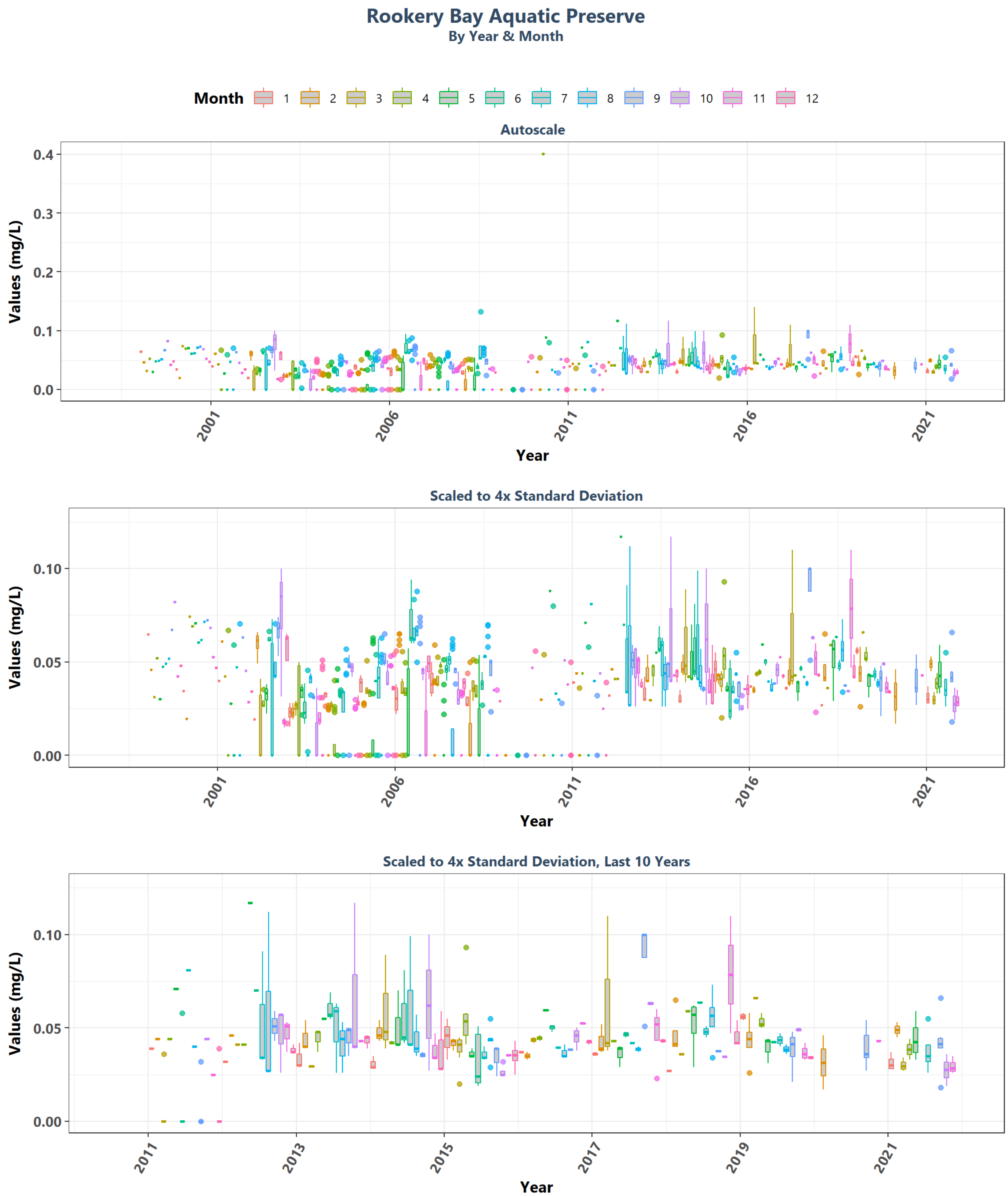


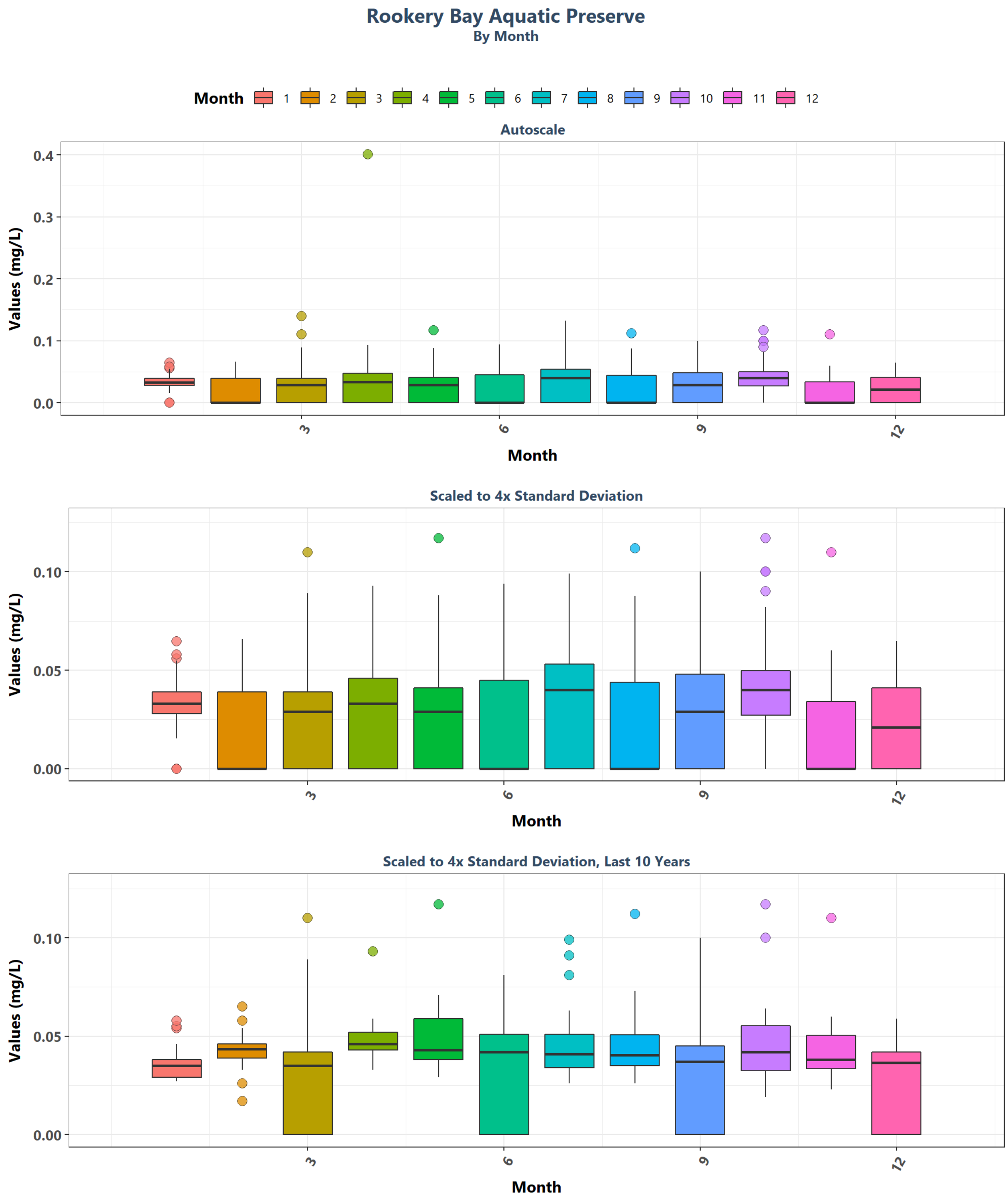


Rocky Bayou State Park Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.00
0.01
0.02
0.03
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.01
0.02
0.03
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.01
0.02
0.03
3
6
9
12
Month

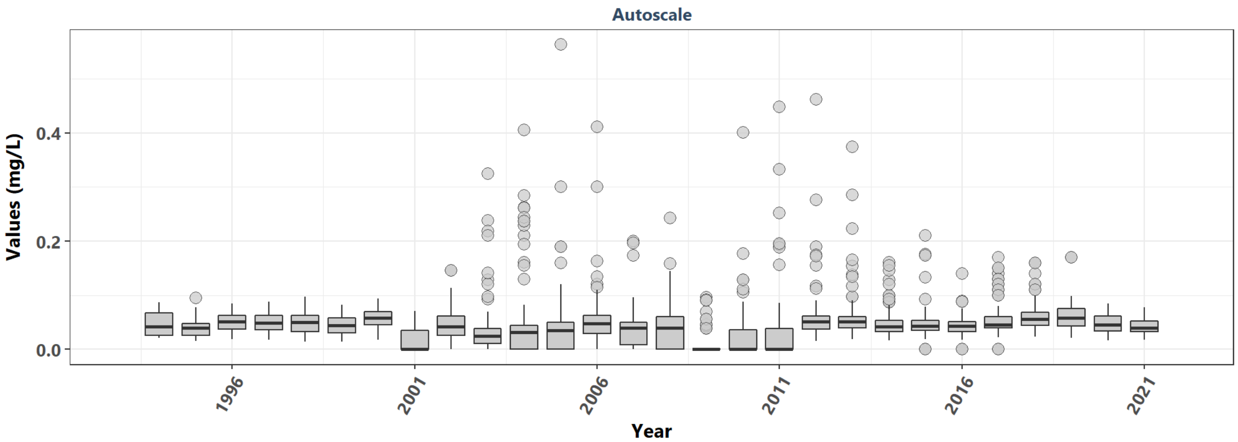
Rookery Bay Aquatic Preserve
By Year

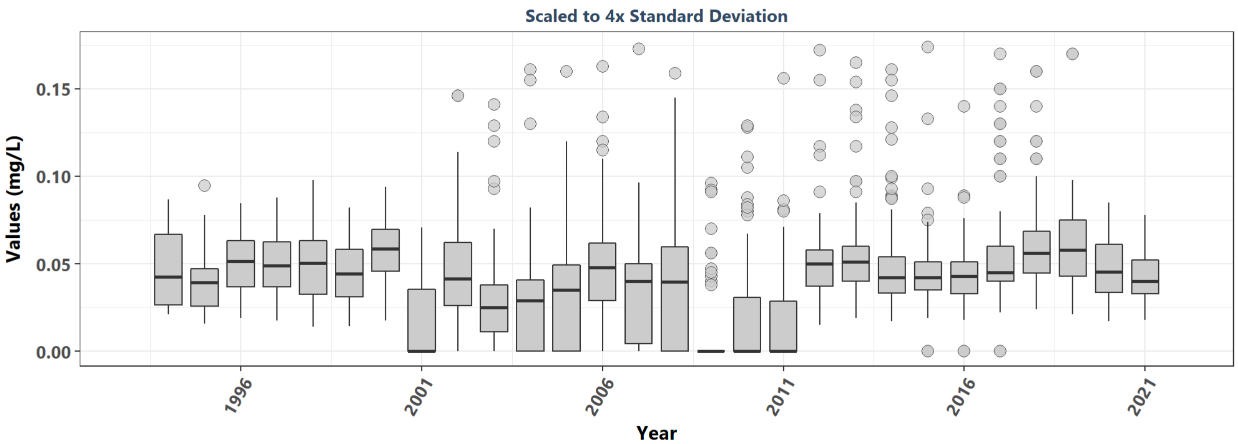


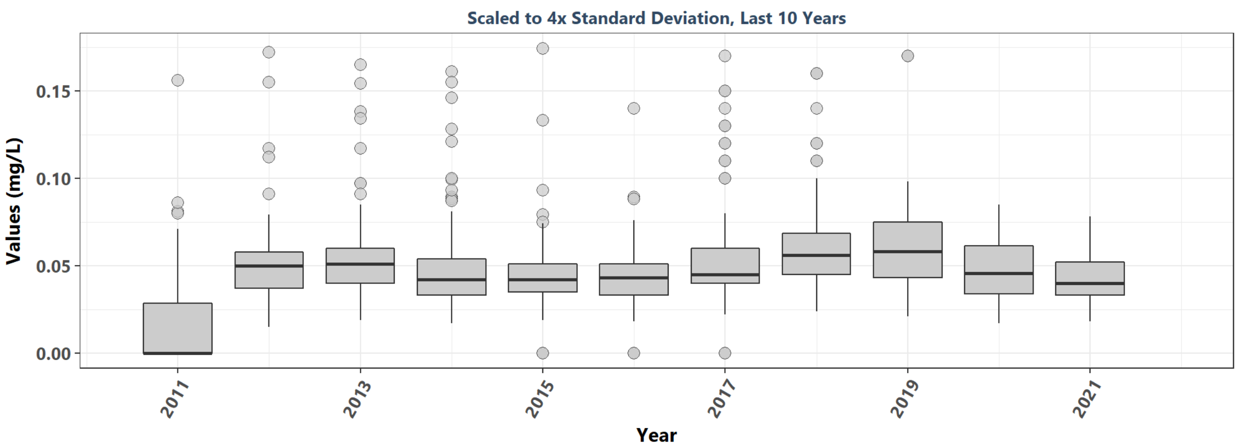

Rookery Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.1
0.2
0.3
0.4
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
2011
2013
2015
2017
2019
2021
Year


Rookery Bay Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.1
0.2
0.3
0.4
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.05
0.10
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.05
0.10
3
6
9
12
Month

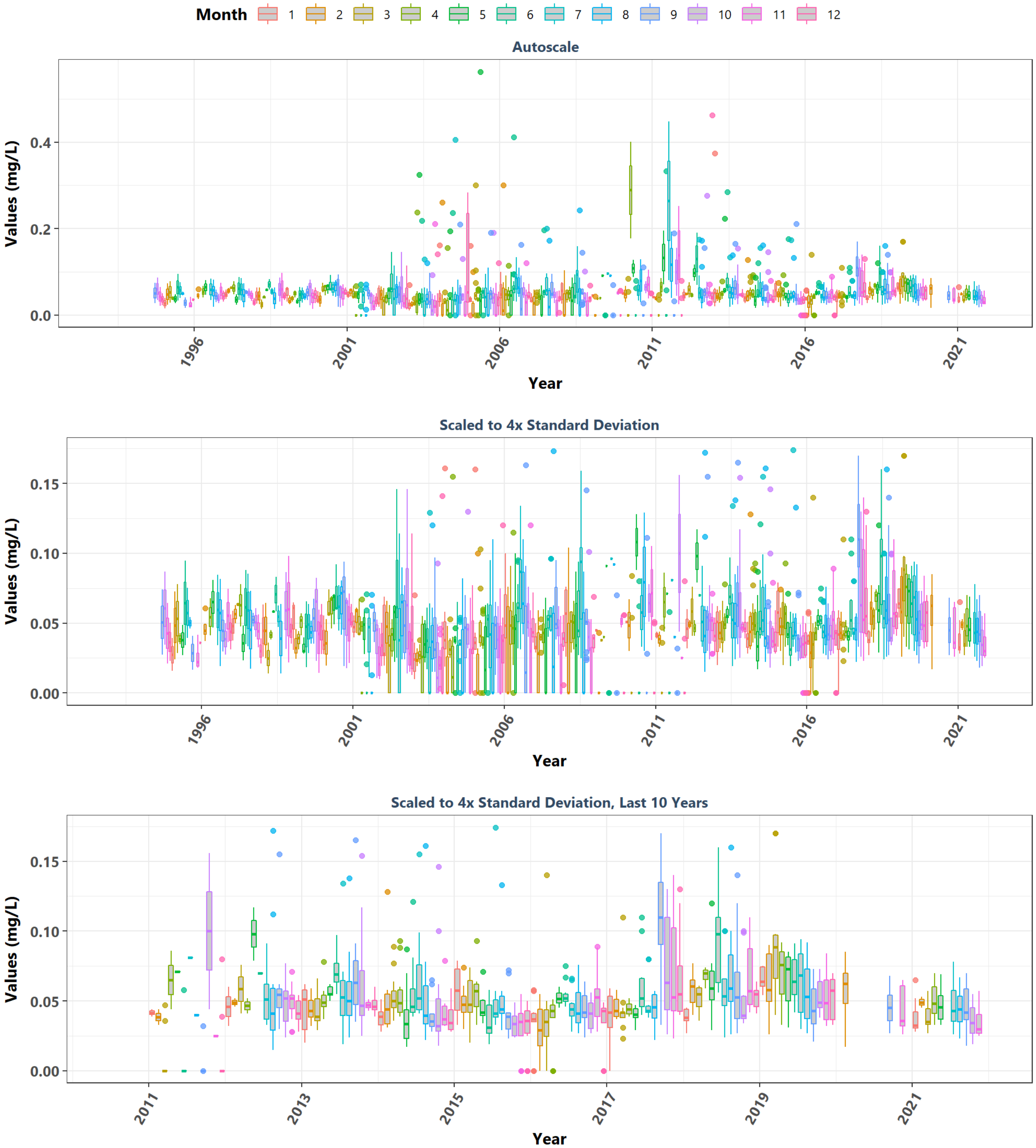
Rookery Bay National Estuarine Research Reserve
By Year



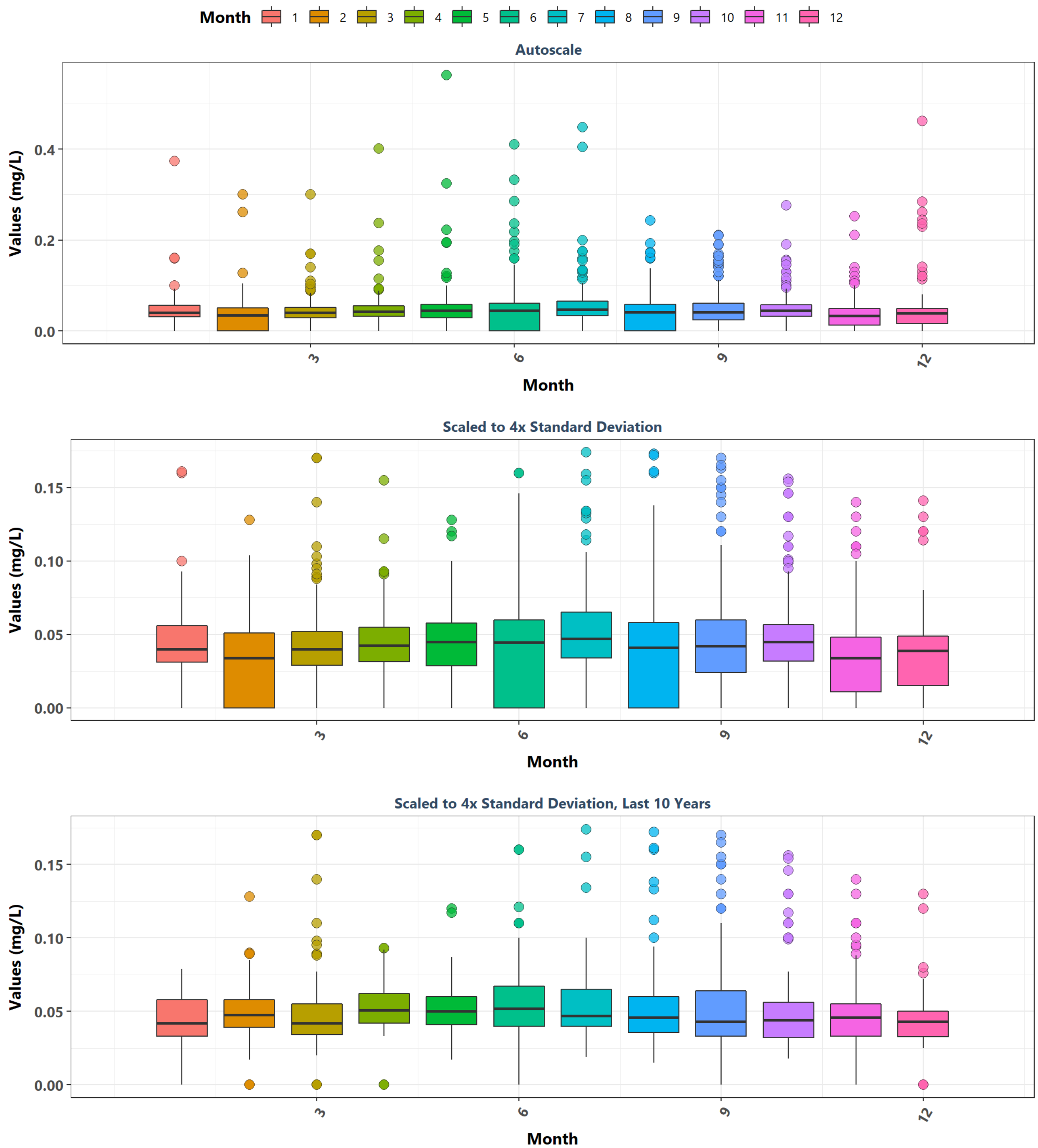




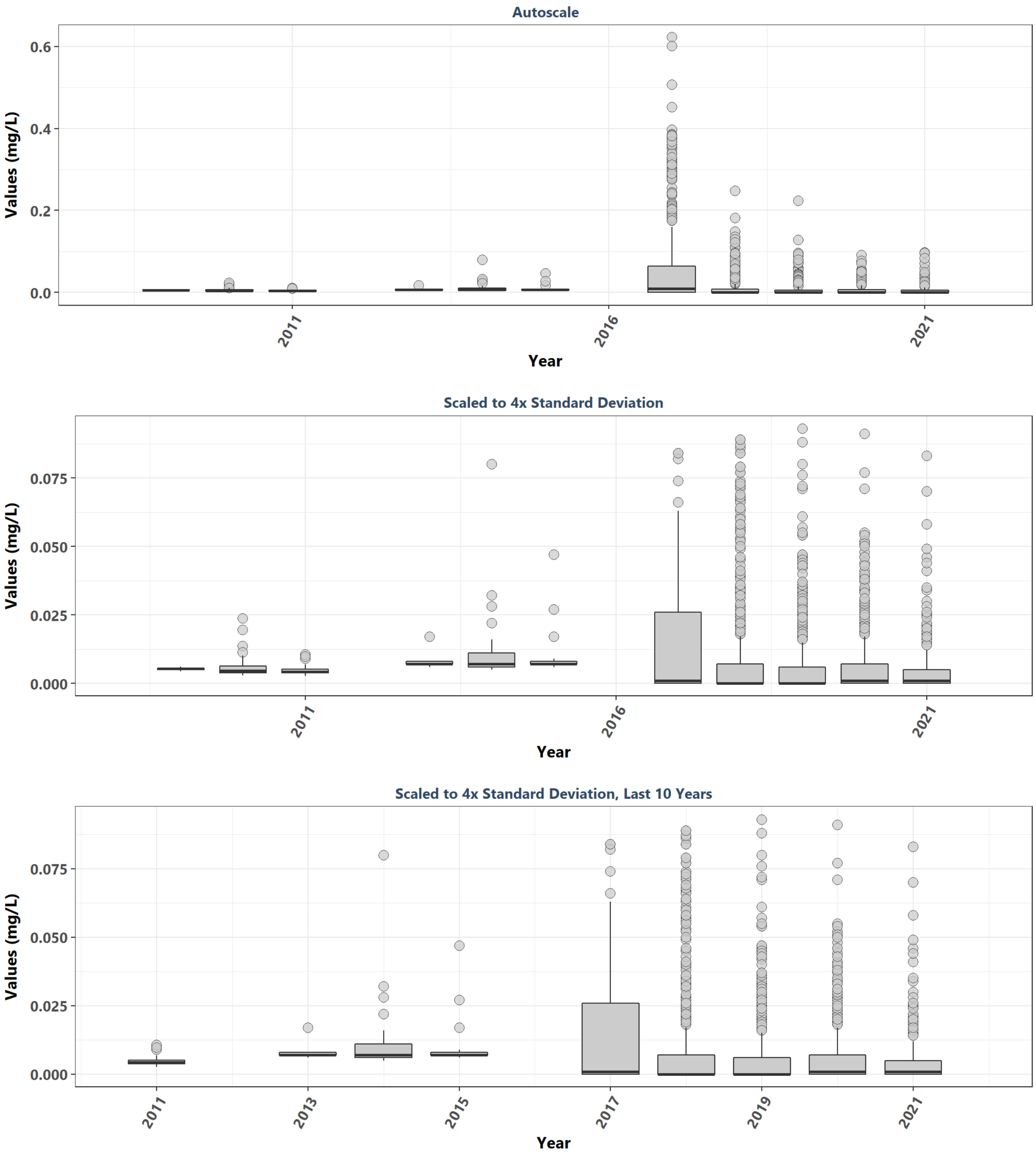
Rookery Bay National Estuarine Research Reserve
By Year & Month

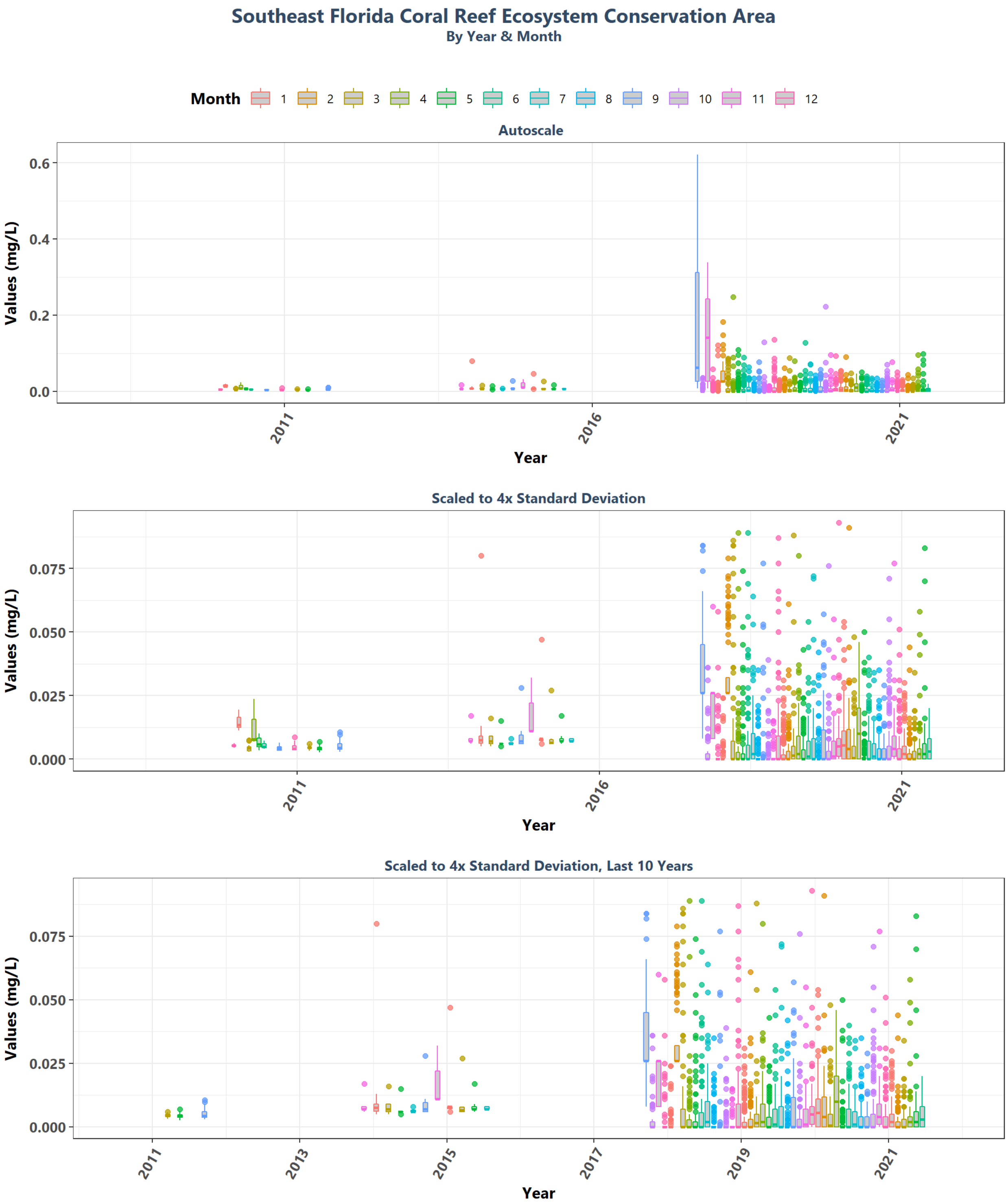


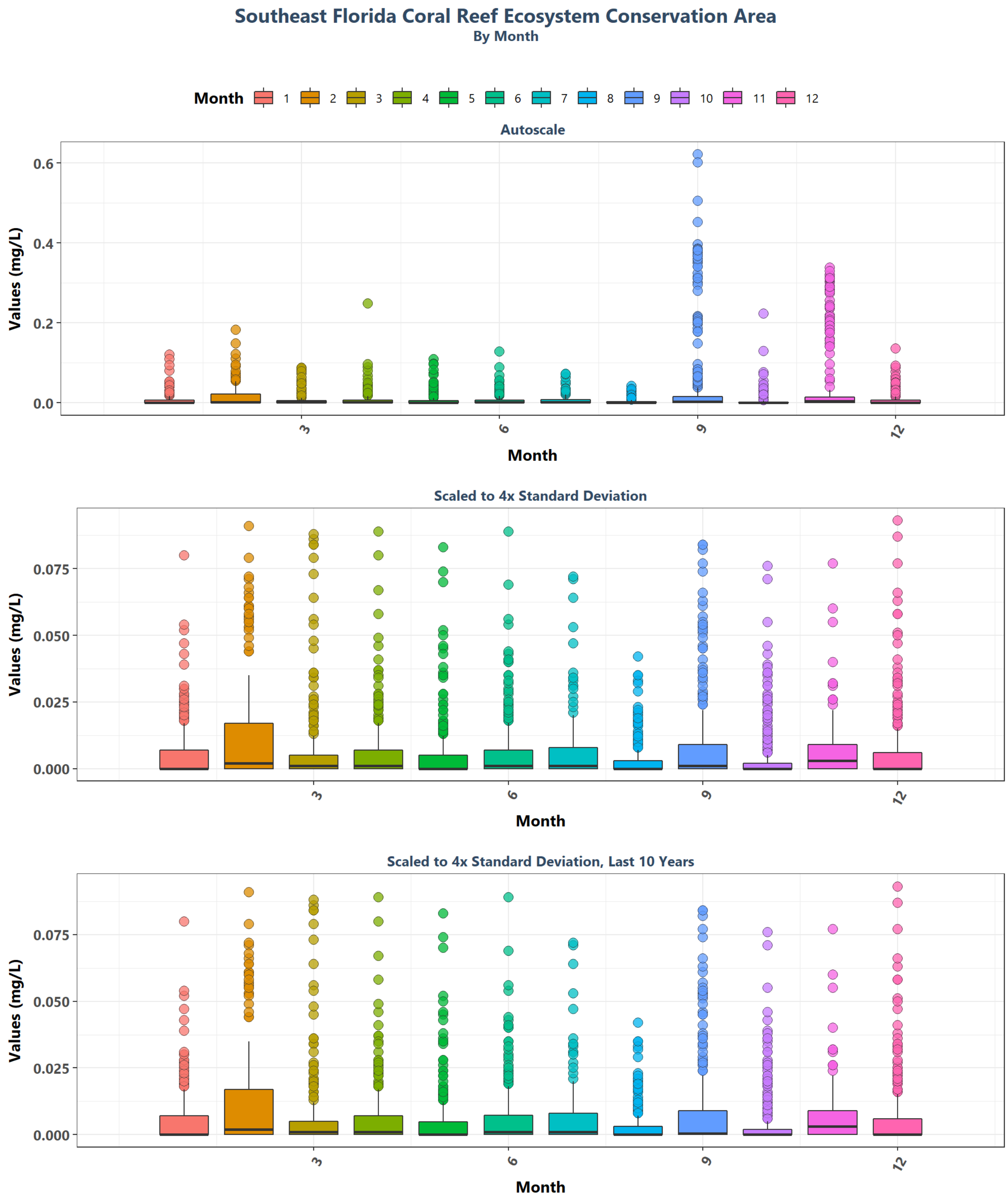
Rookery Bay National Estuarine Research Reserve
By Month



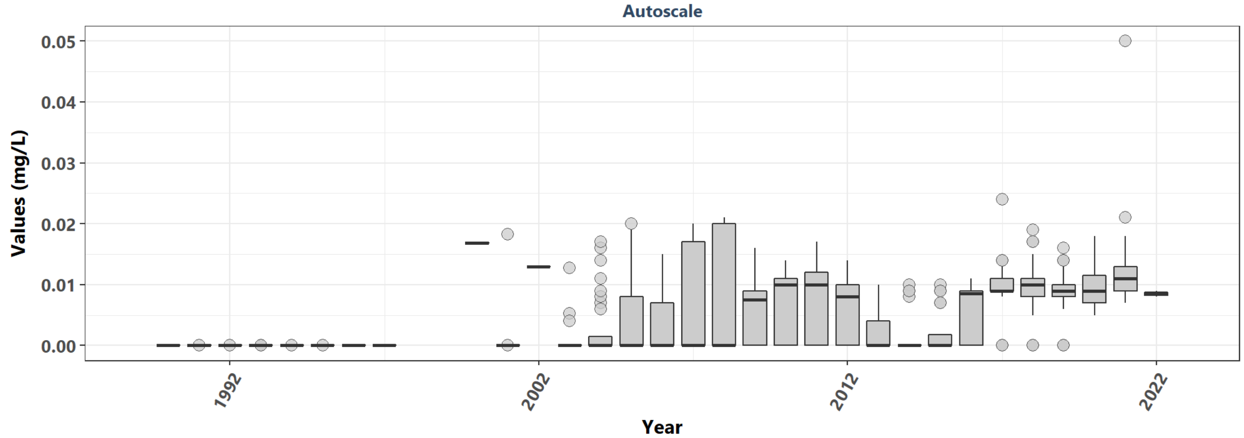
Southeast Florida Coral Reef Ecosystem Conservation Area
By Year

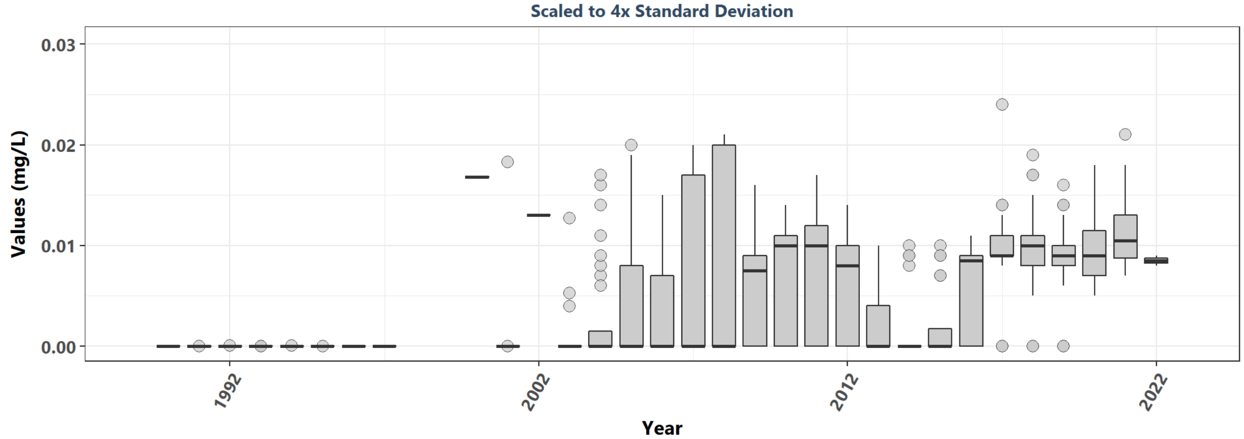


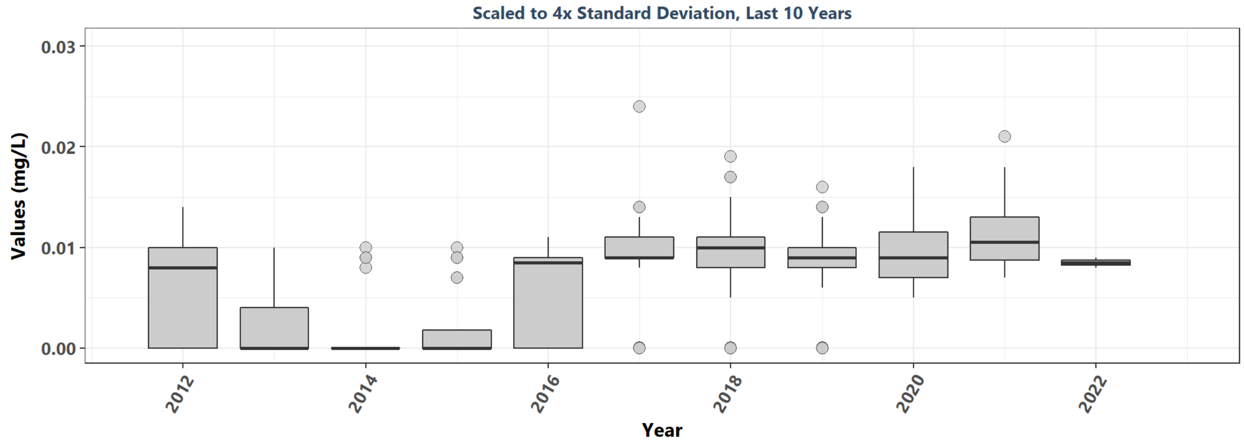

Southeast Florida Coral Reef Ecosystem Conservation Area
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.6
0.4
0.2
0.0
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.075
0.050
0.025
0.000
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.075
0.050
0.025
0.000
2011
2013
2015
2017
2019
2021
Year

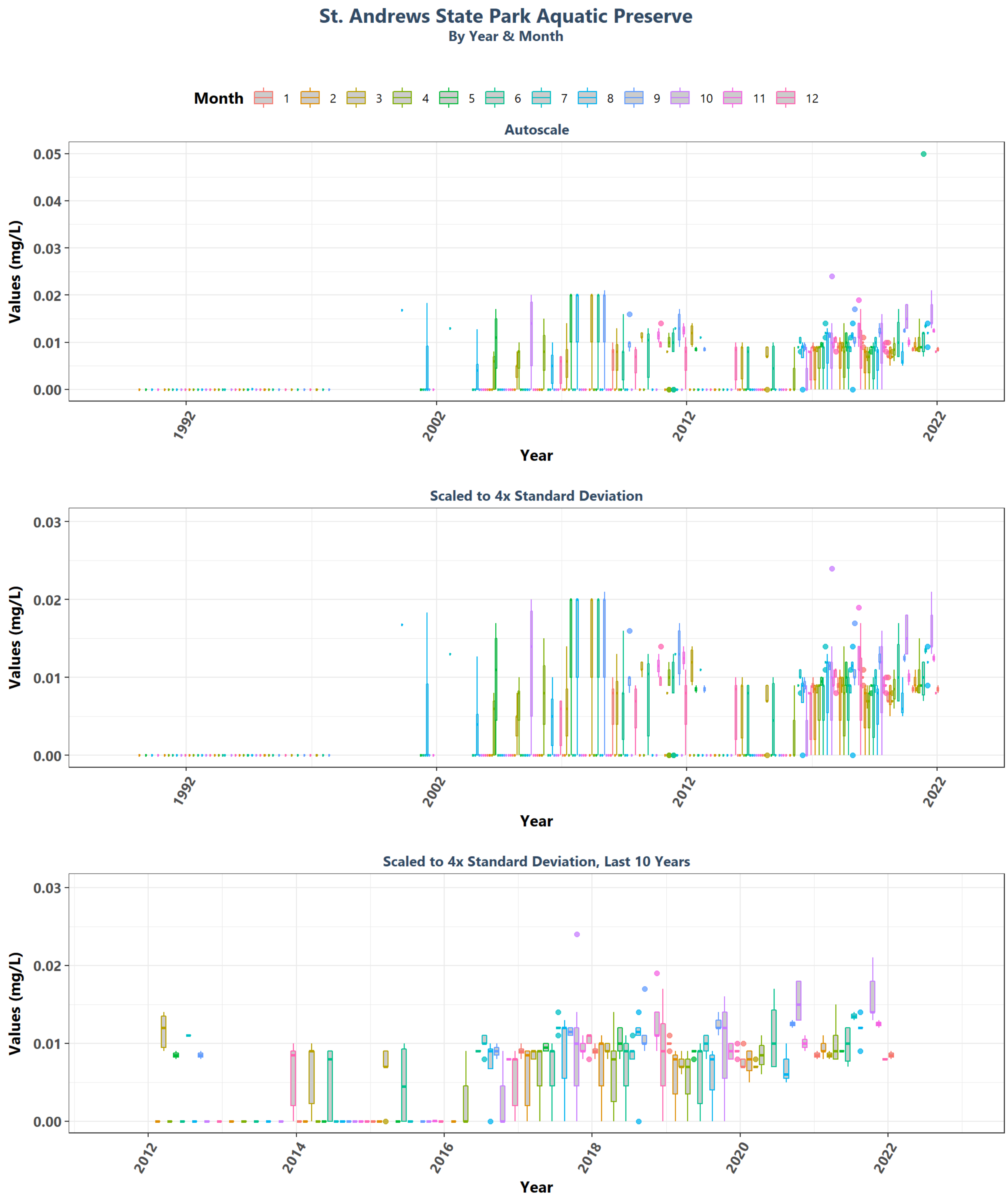

Southeast Florida Coral Reef Ecosystem Conservation Area
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.2
0.4
0.6
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.000
0.025
0.050
0.075
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.000
0.025
0.050
0.075
3
6
9
12
Month

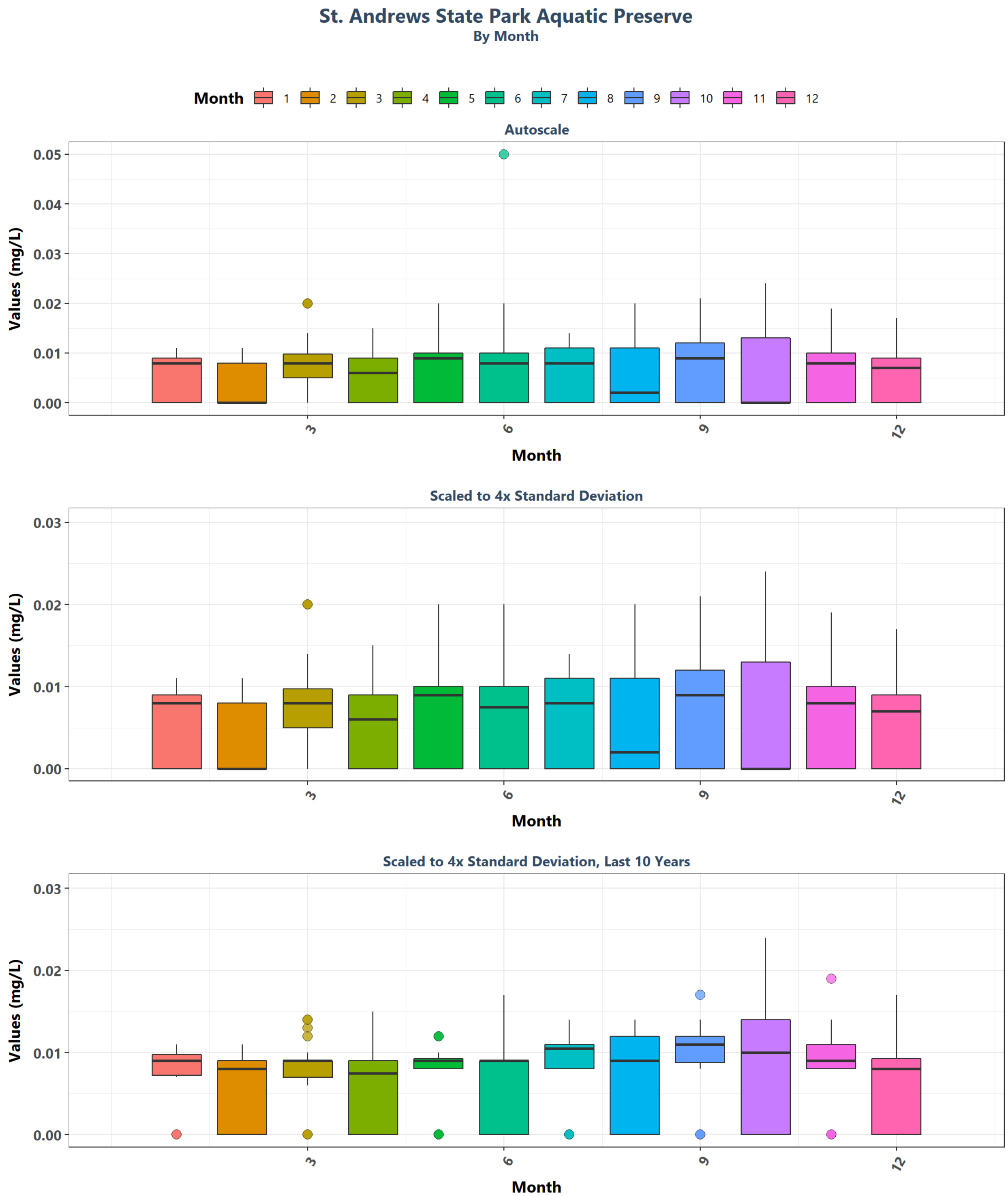
St. Andrews State Park Aquatic Preserve
By Year



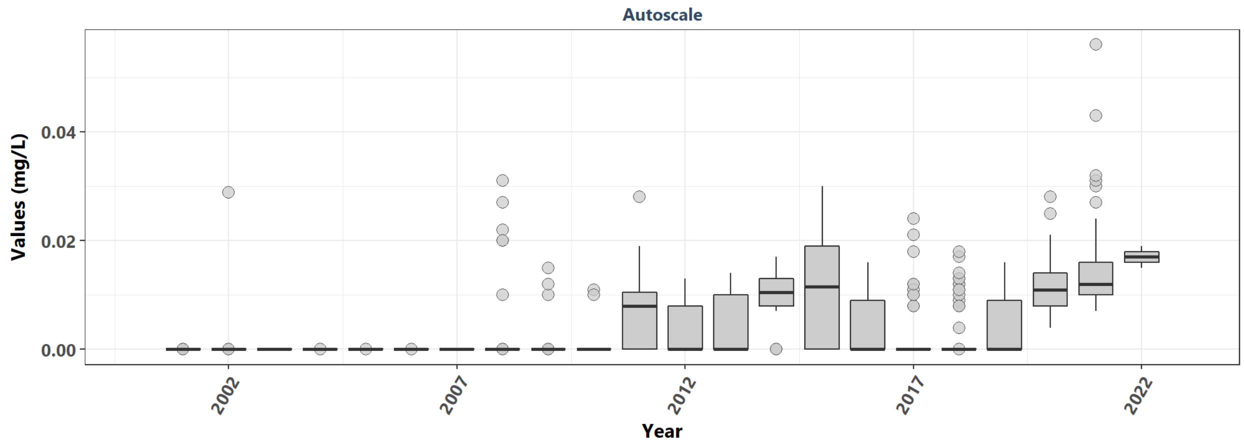


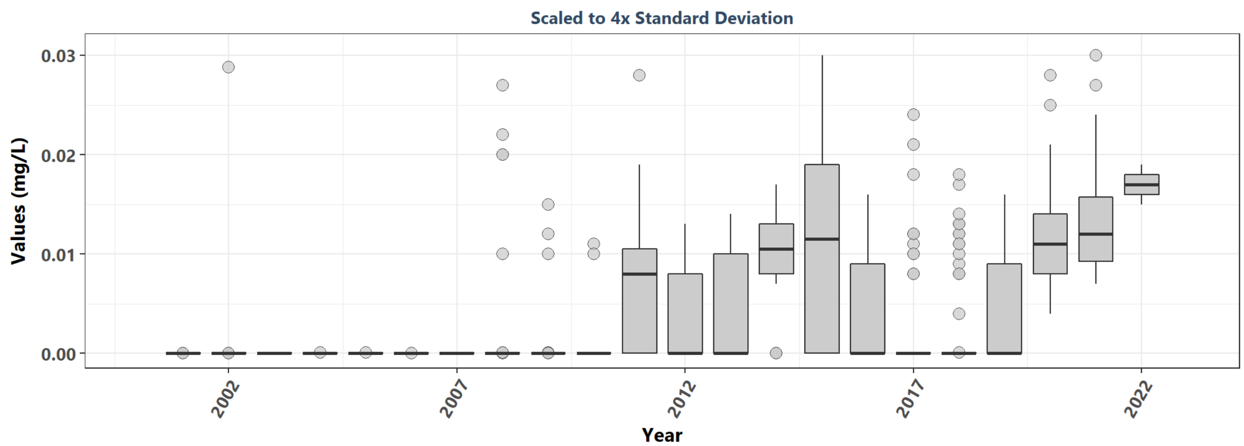


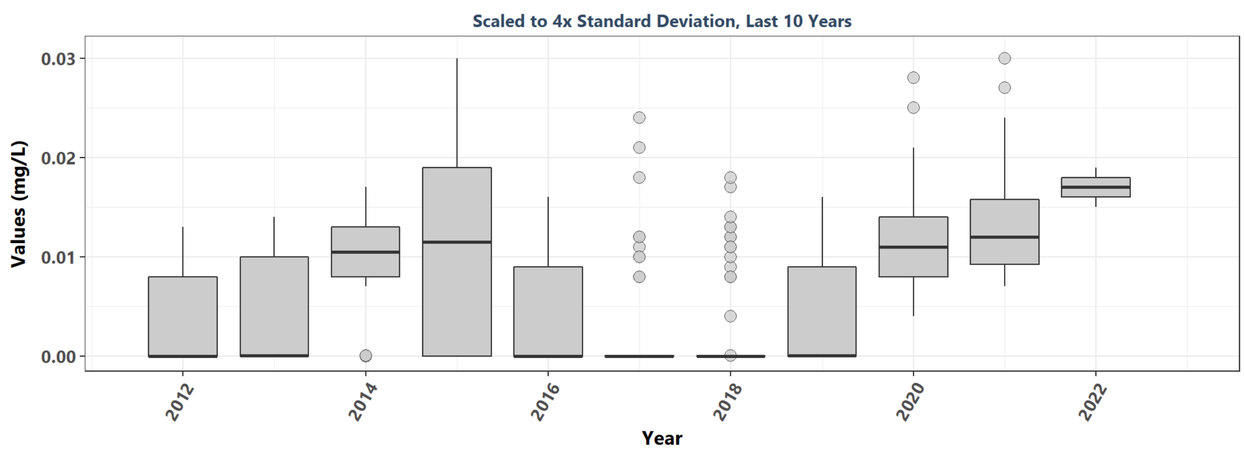

St. Andrews State Park Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.05
0.04
0.03
0.02
0.01
0.00
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.03
0.02
0.01
0.00
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.03
0.02
0.01
0.00
2012
2014
2016
2018
2020
2022
Year

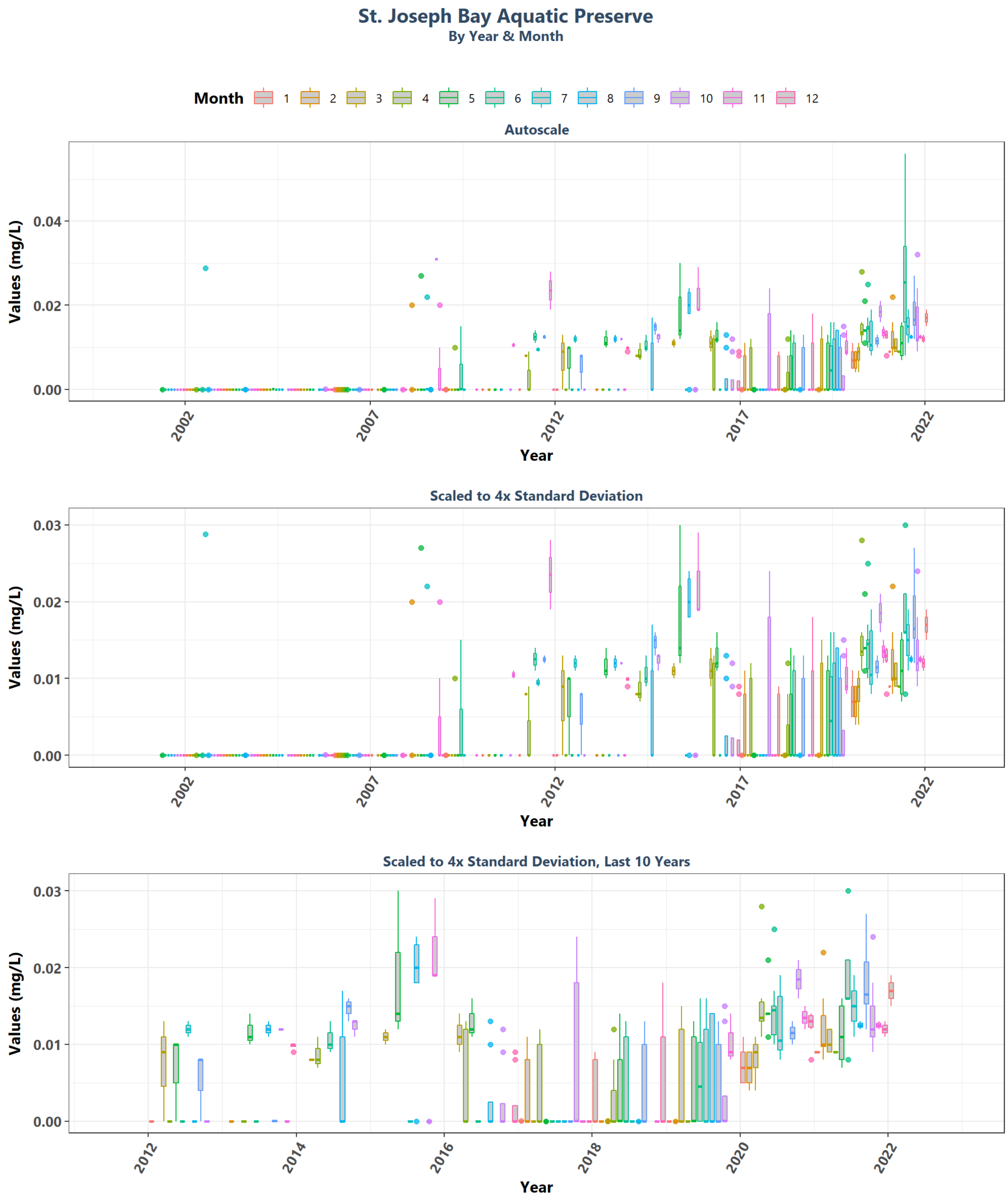

St. Andrews State Park Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.05
0.04
0.03
0.02
0.01
0.00
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.03
0.02
0.01
0.00
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.03
0.02
0.01
0.00
3
6
9
12
Month

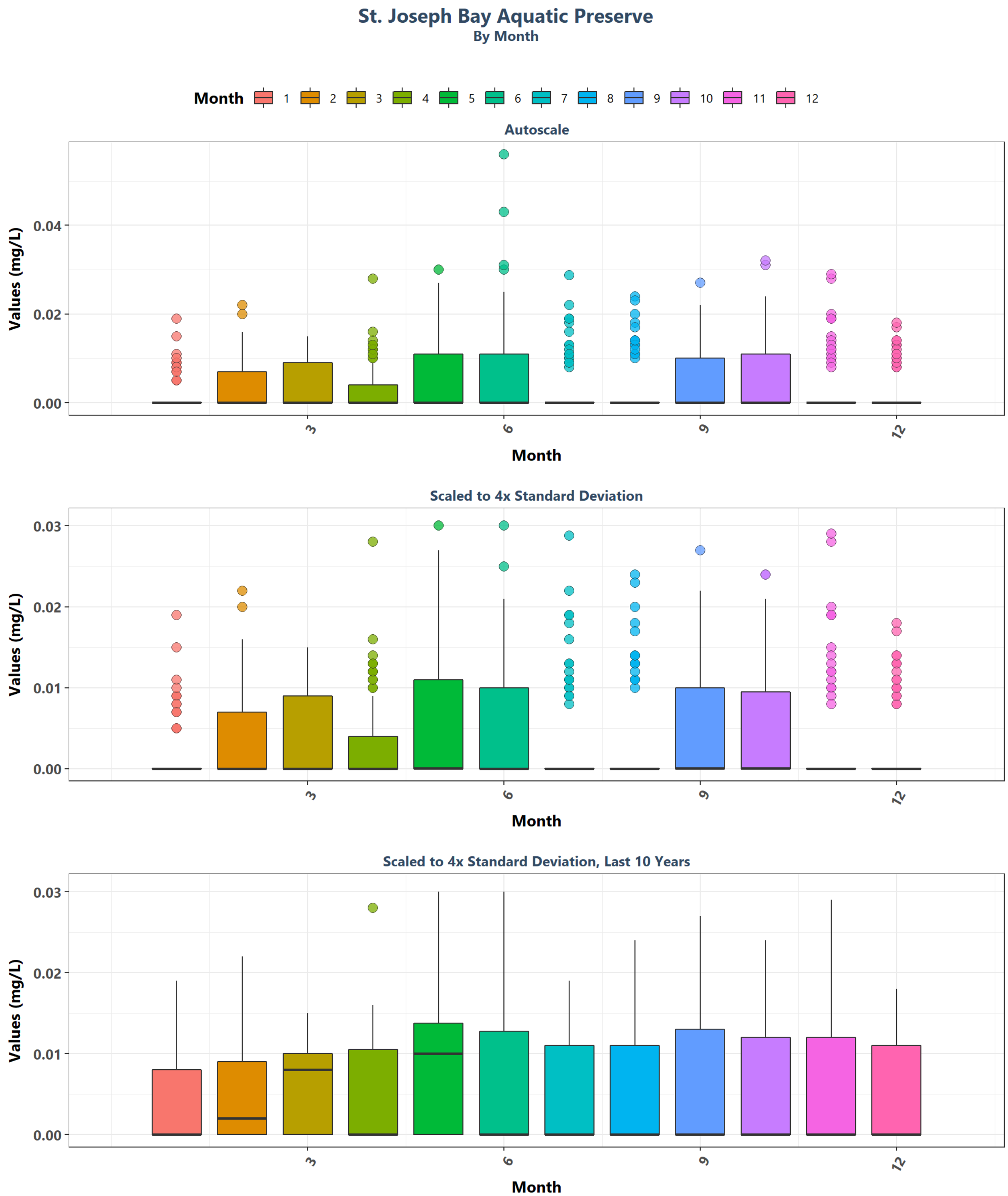
St. Joseph Bay Aquatic Preserve
By Year





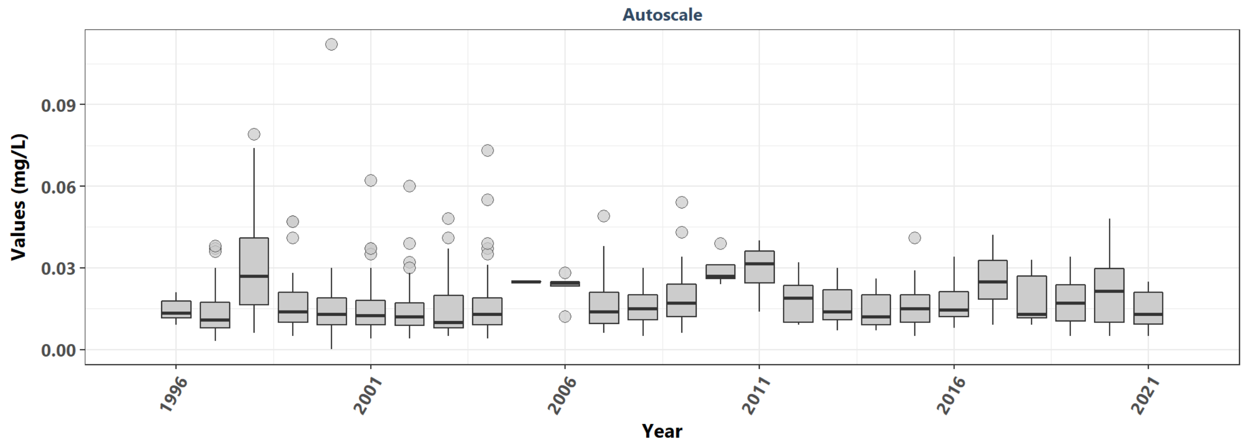


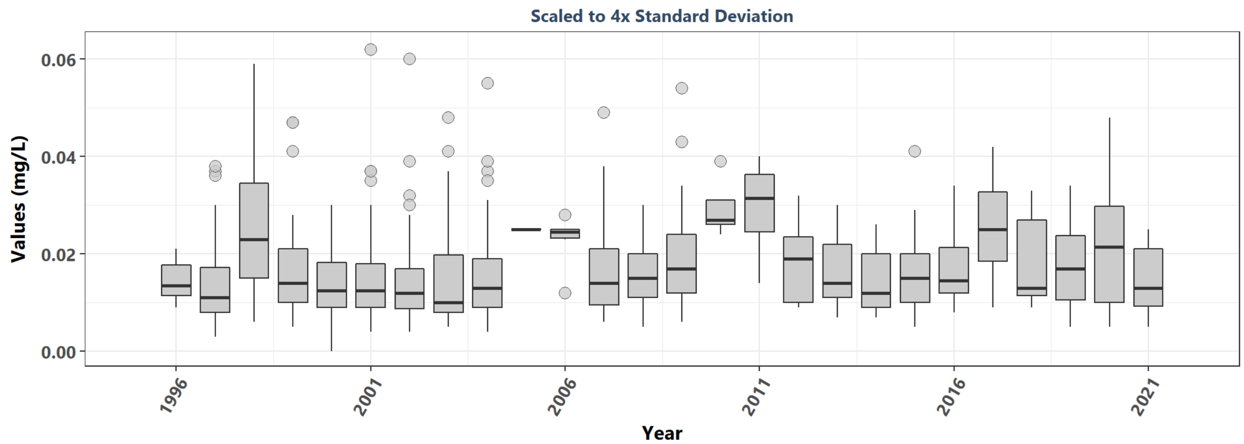

St. Joseph Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.00
0.02
0.04
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.01
0.02
0.03
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.01
0.02
0.03
2012
2014
2016
2018
2020
2022
Year

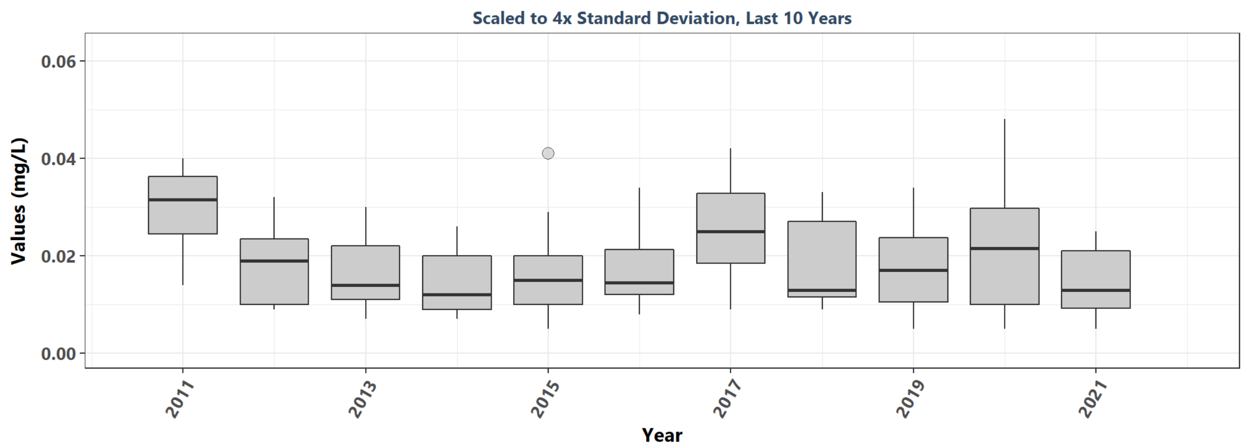

St. Joseph Bay Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.00
0.02
0.04
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.01
0.02
0.03
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.01
0.02
0.03
3
6
9
12
Month

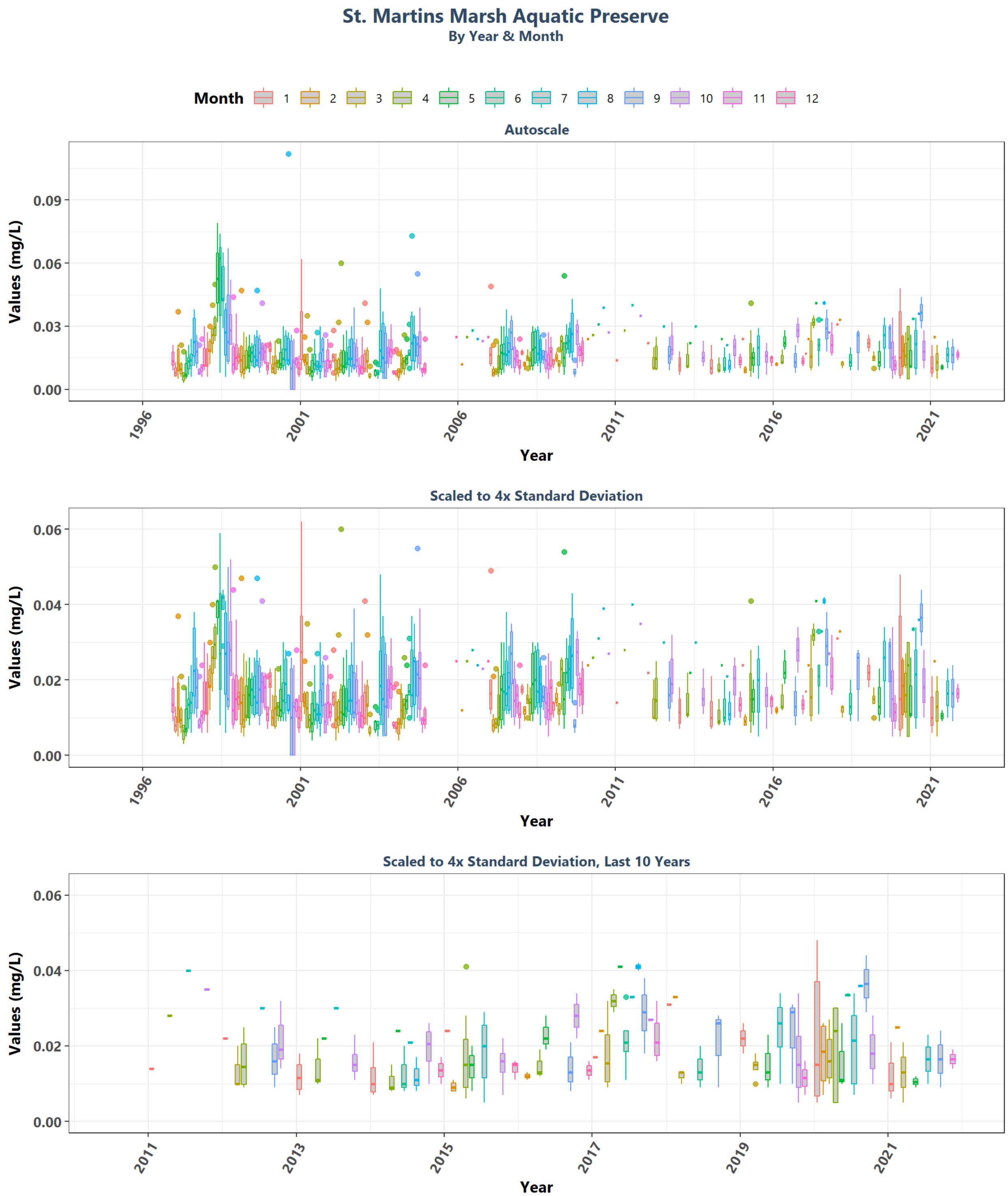
St. Martins Marsh Aquatic Preserve

By Year

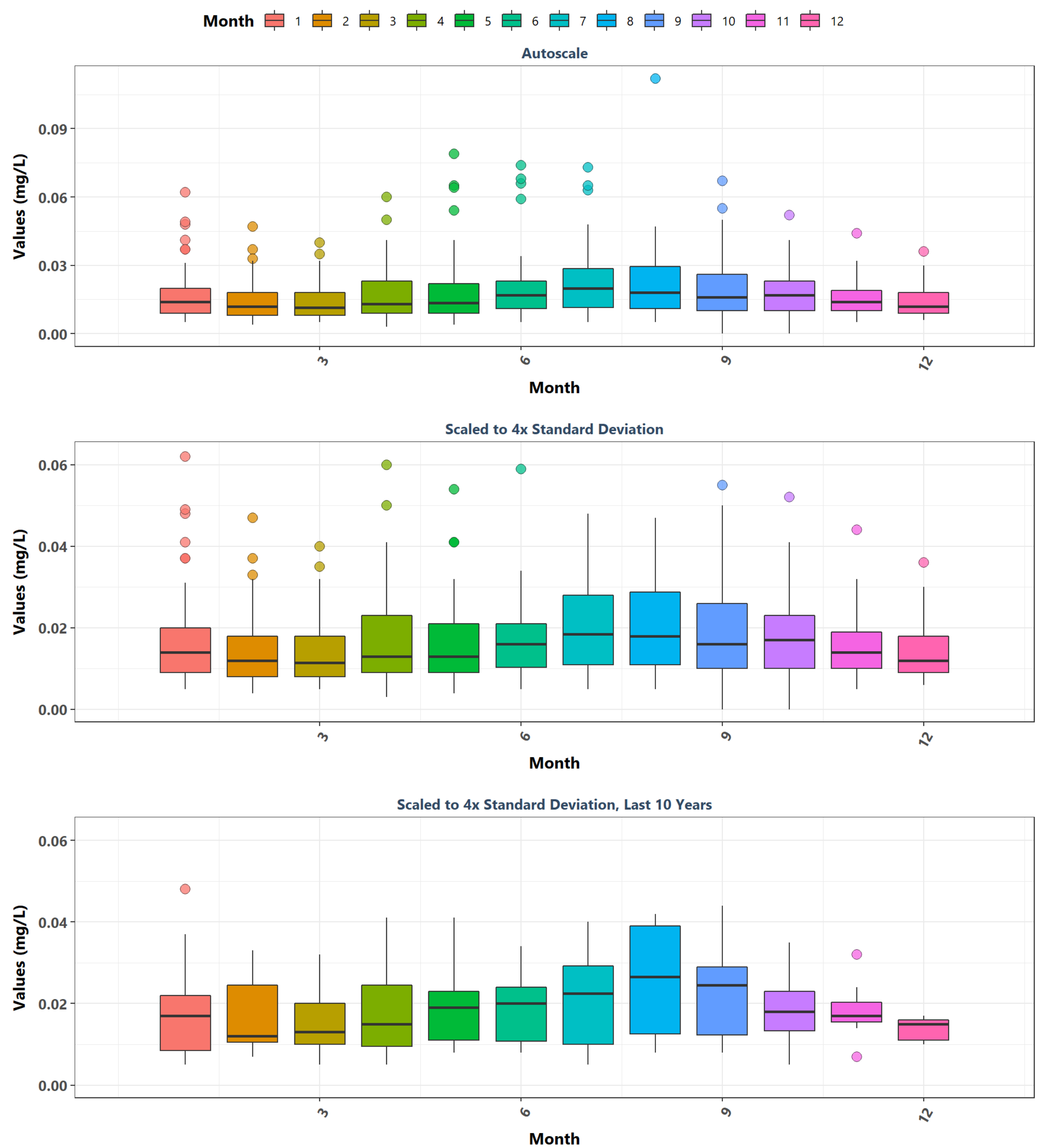




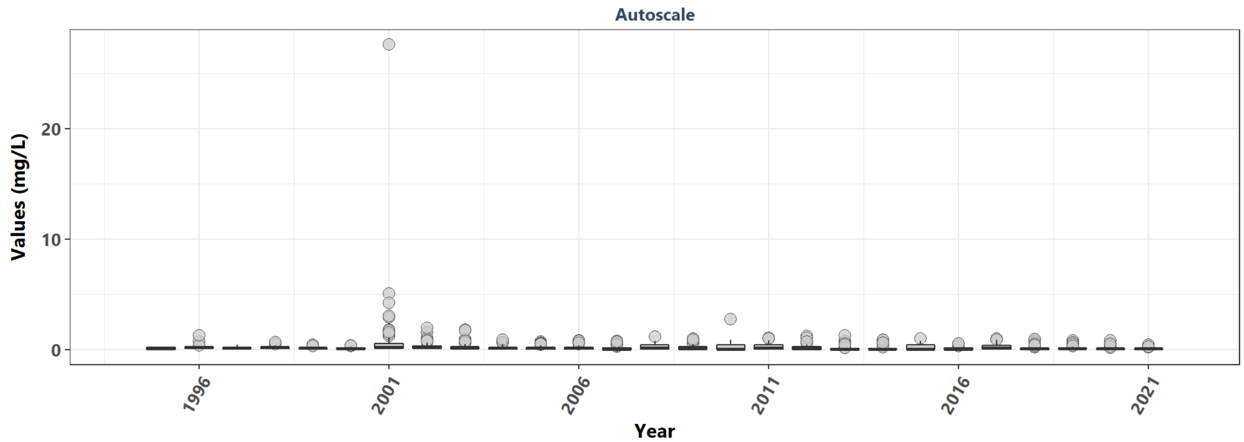


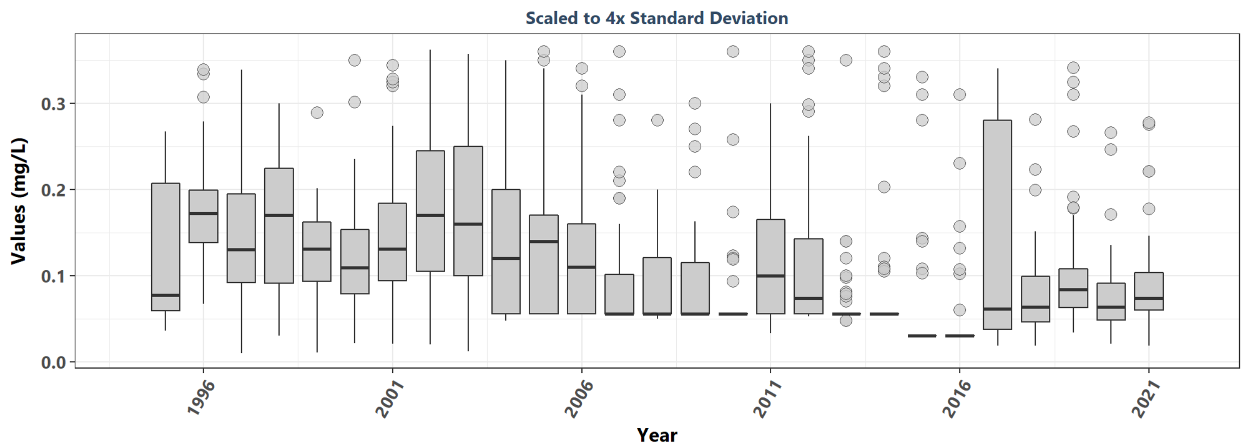

St. Martins Marsh Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.00
0.03
0.06
0.09
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.00
0.02
0.04
0.06
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.00
0.02
0.04
0.06
2011
2013
2015
2017
2019
2021
Year

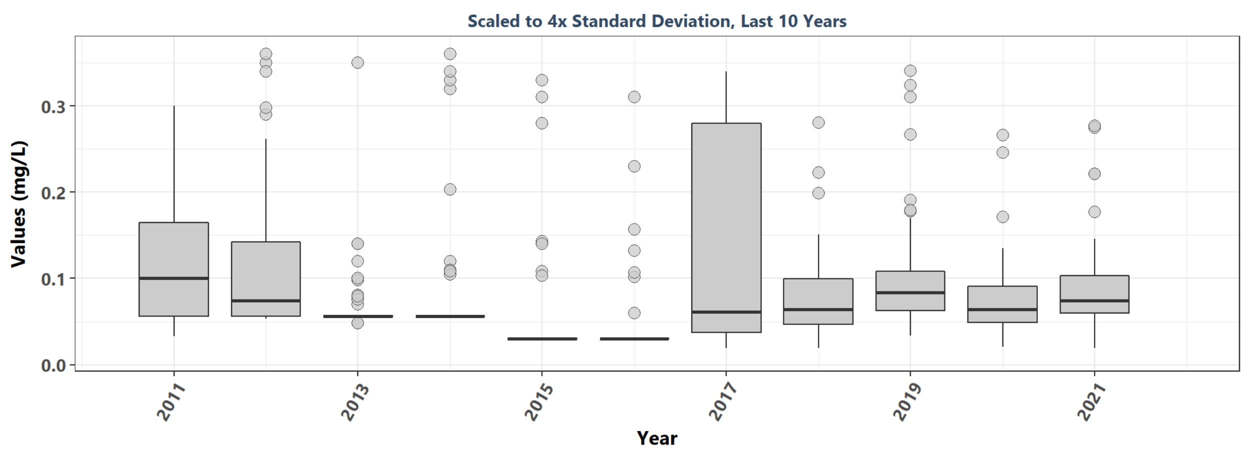
St. Martins Marsh Aquatic Preserve
By Month

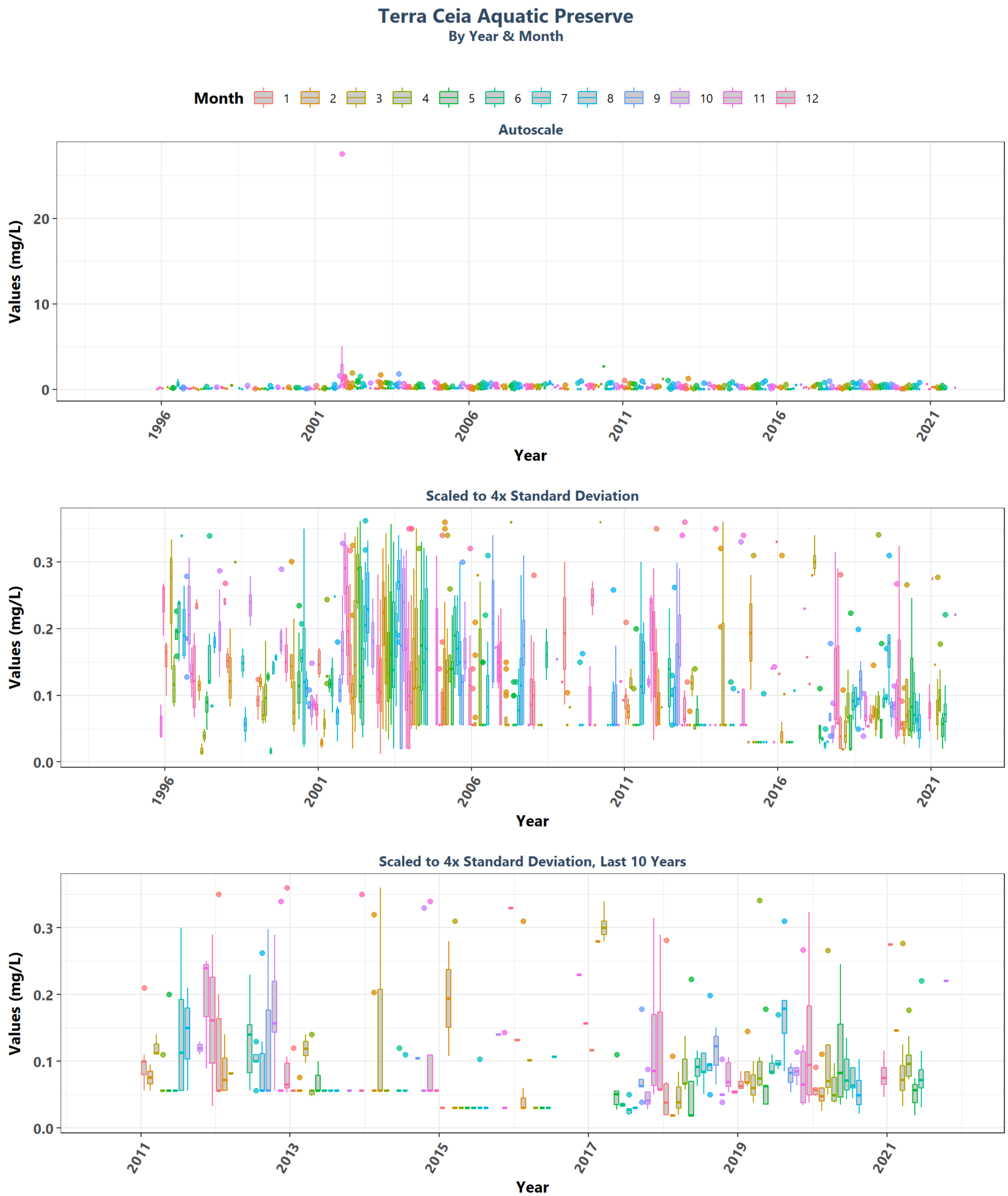


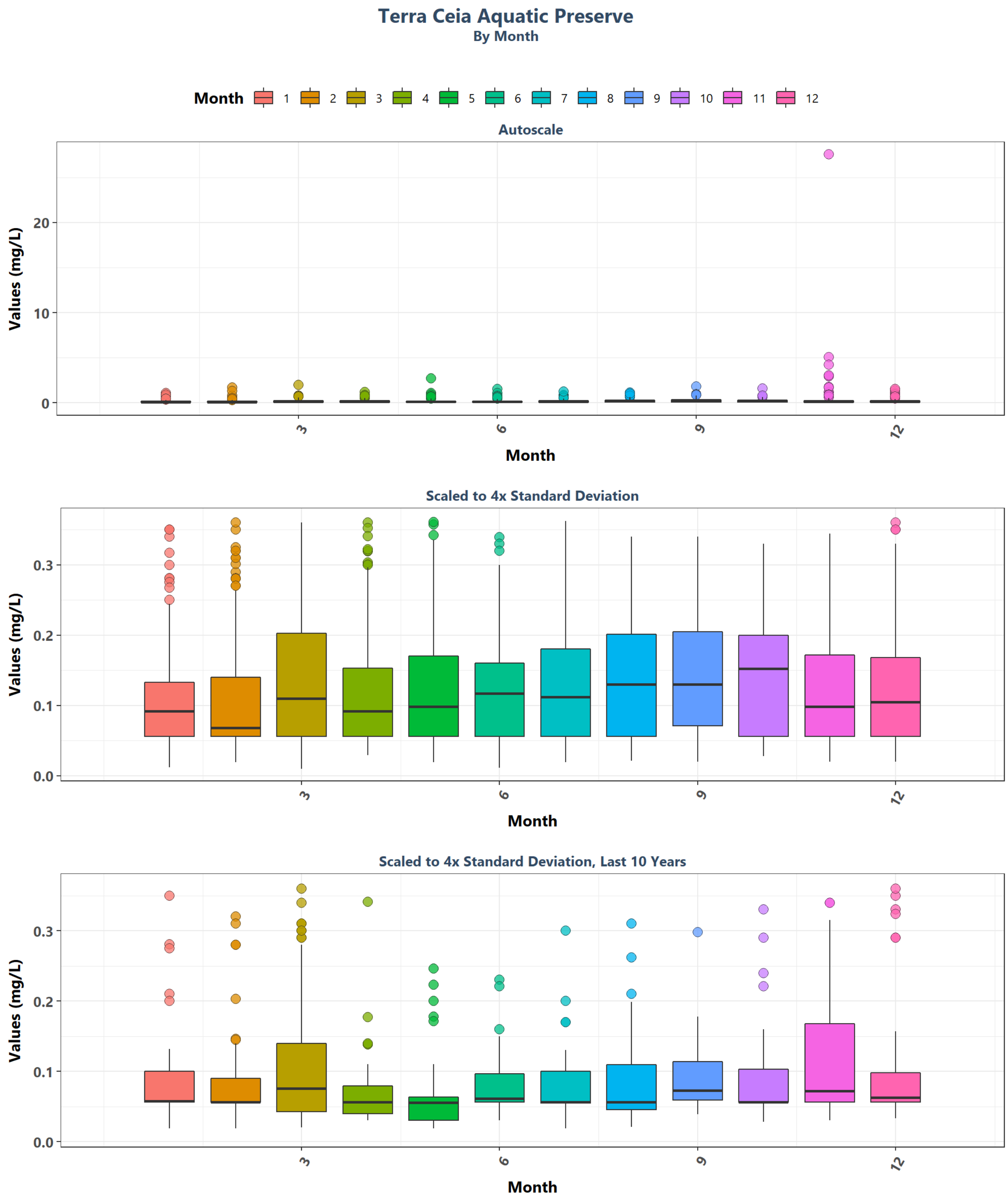
Terra Ceia Aquatic Preserve
By Year







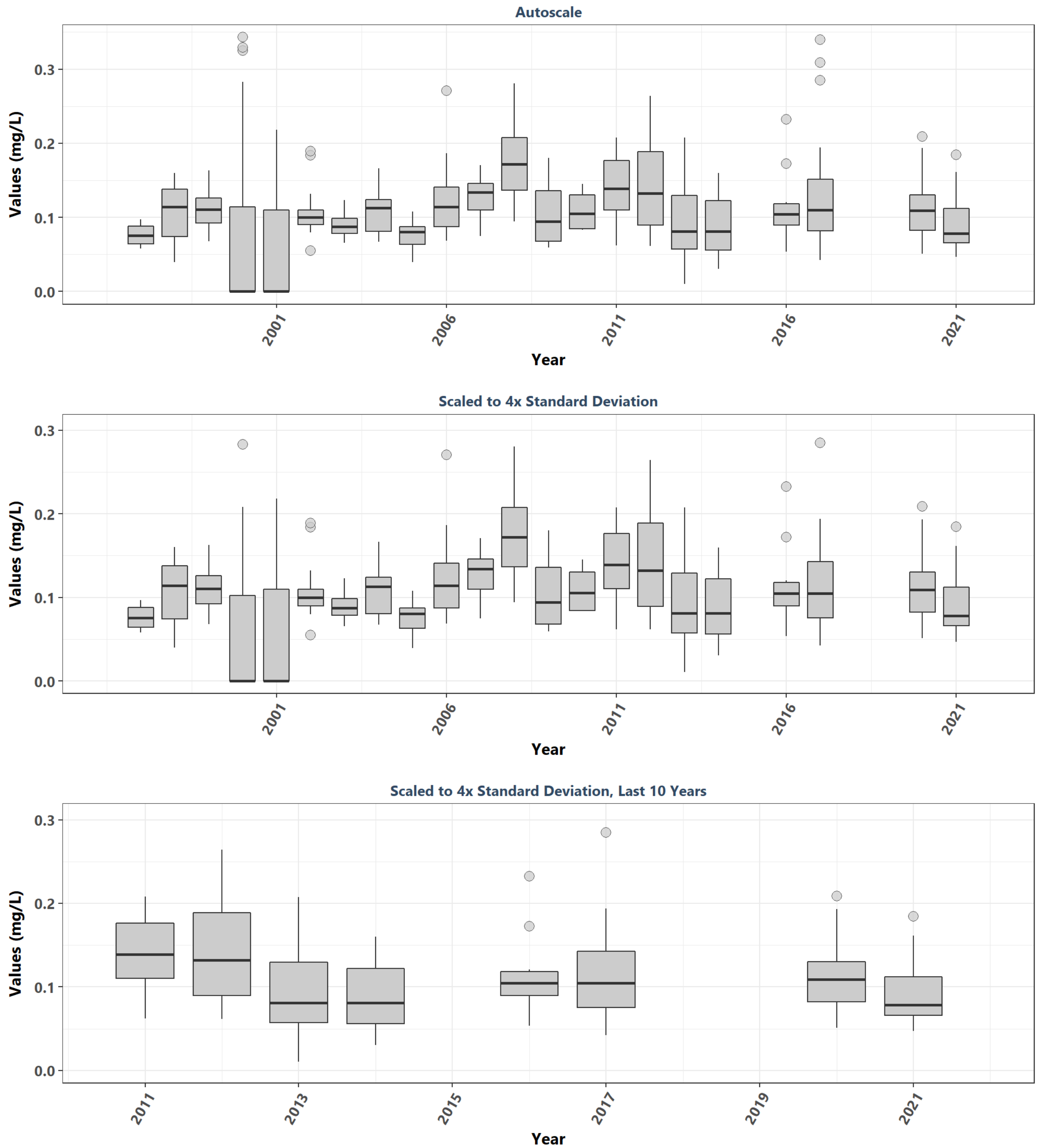

Terra Ceia Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0
10
20
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.0
0.1
0.2
0.3
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.0
0.1
0.2
0.3
2011
2013
2015
2017
2019
2021
Year

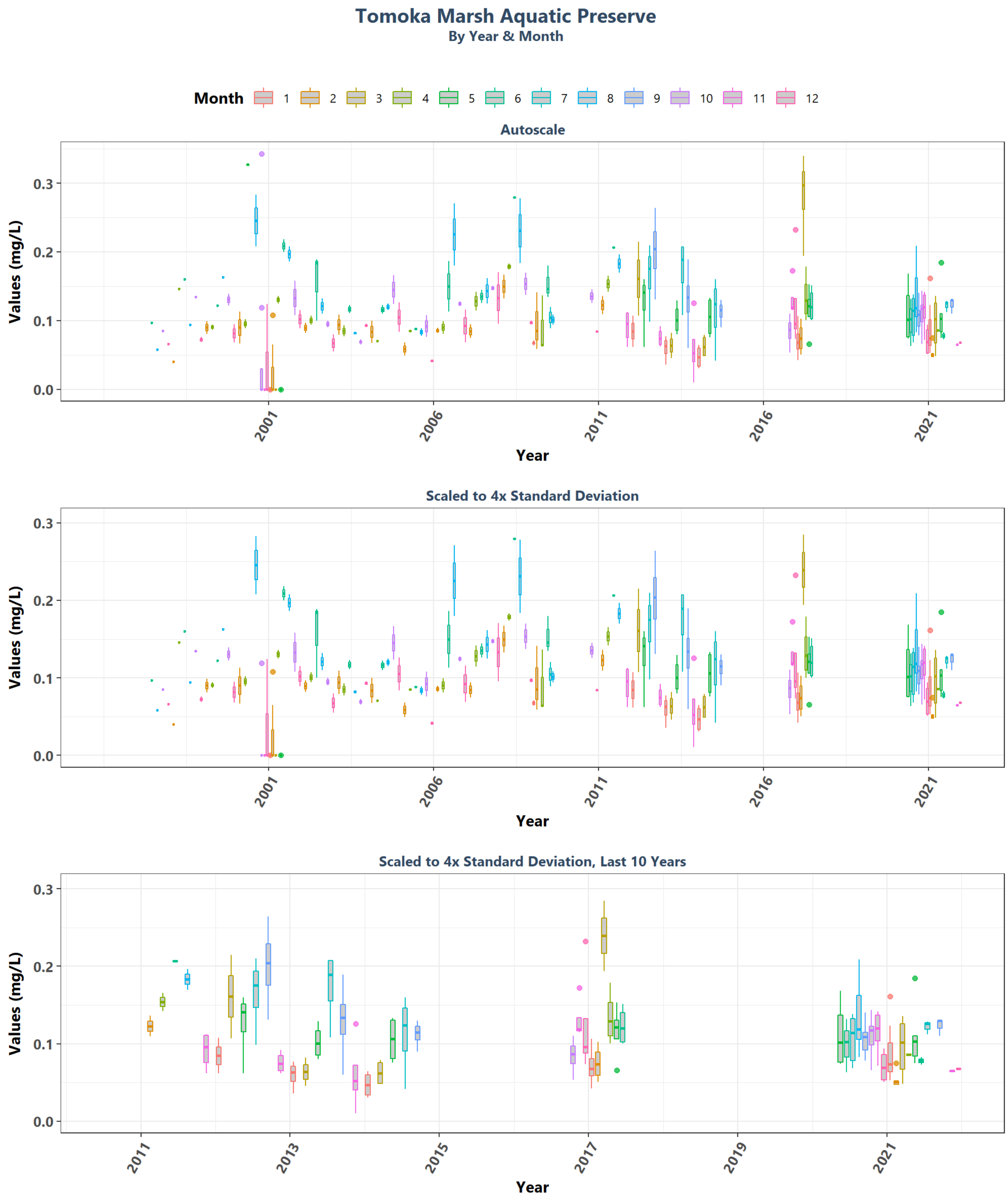


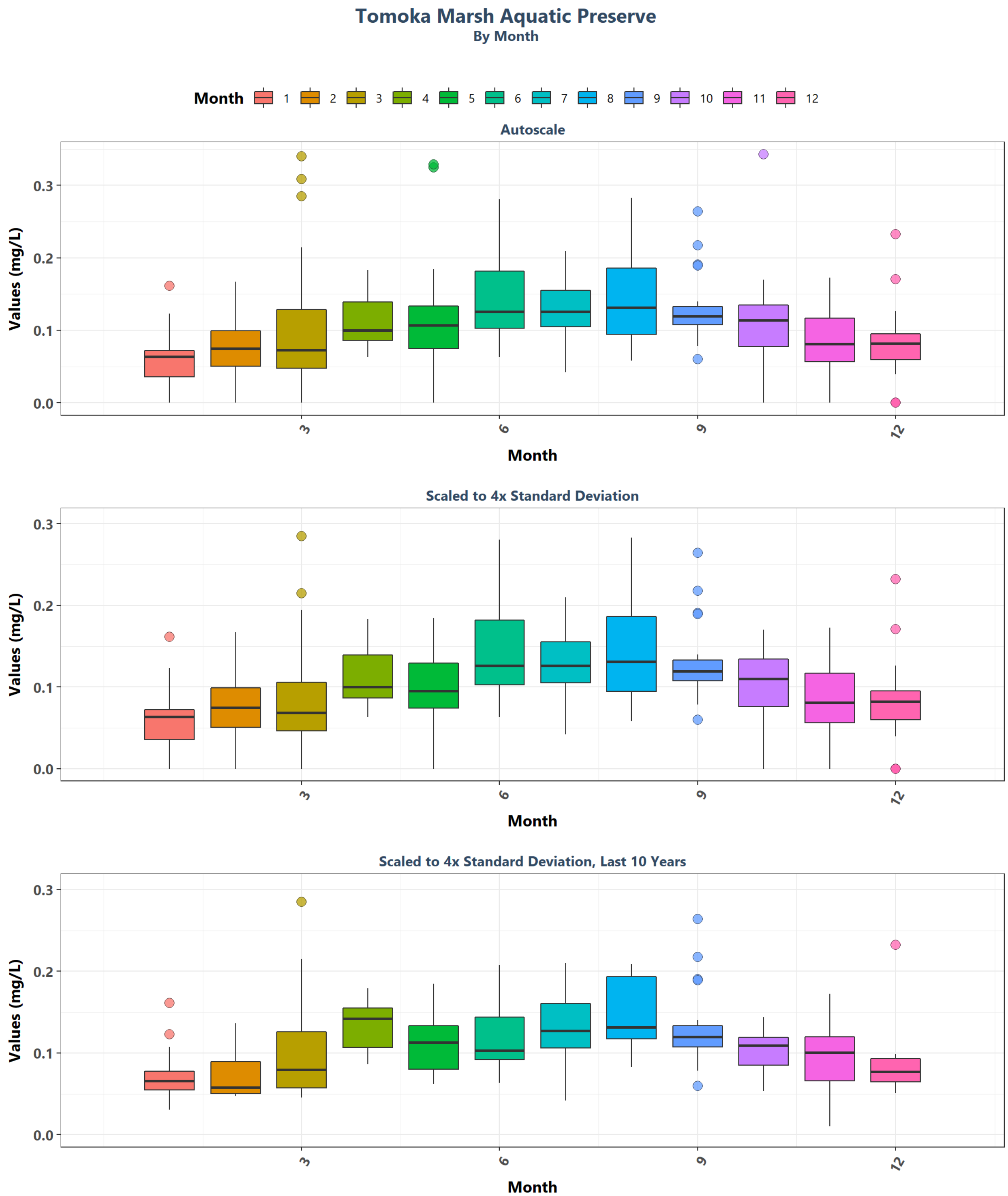
Terra Ceia Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
Month
Scaled to 4x Standard Deviation
Values (mg/L)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
Month

Tomoka Marsh Aquatic Preserve

By Year



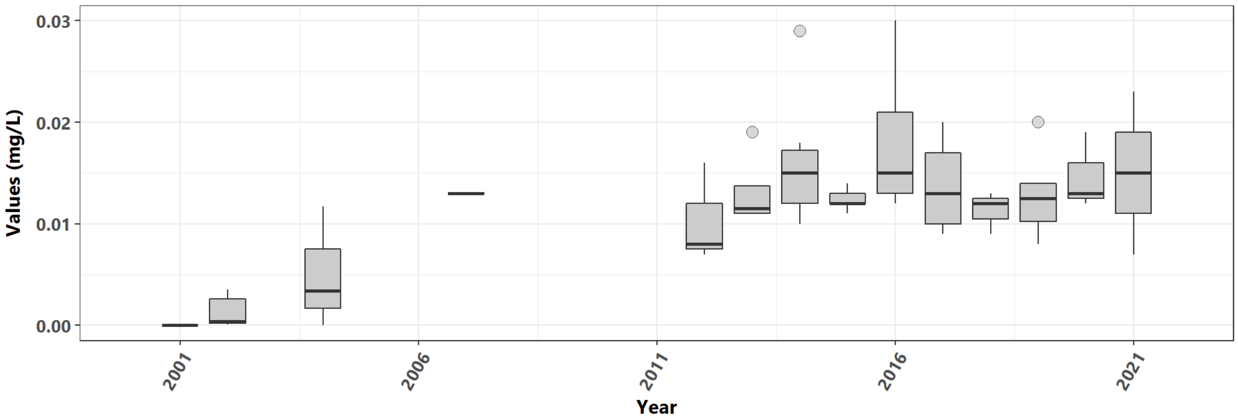

Tomoka Marsh Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.1
0.2
0.3
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.0
0.1
0.2
0.3
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.0
0.1
0.2
0.3
2011
2013
2015
2017
2019
2021
Year



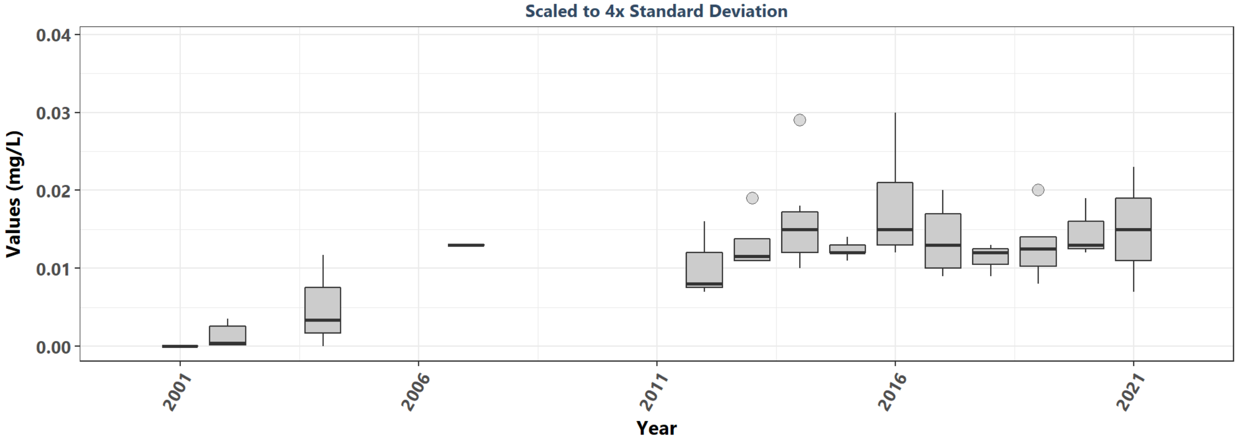
Tomoka Marsh Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.0
0.1
0.2
0.3
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (mg/L)
0.0
0.1
0.2
0.3
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.0
0.1
0.2
0.3
3
6
9
12
Month

Yellow River Marsh Aquatic Preserve
By Year

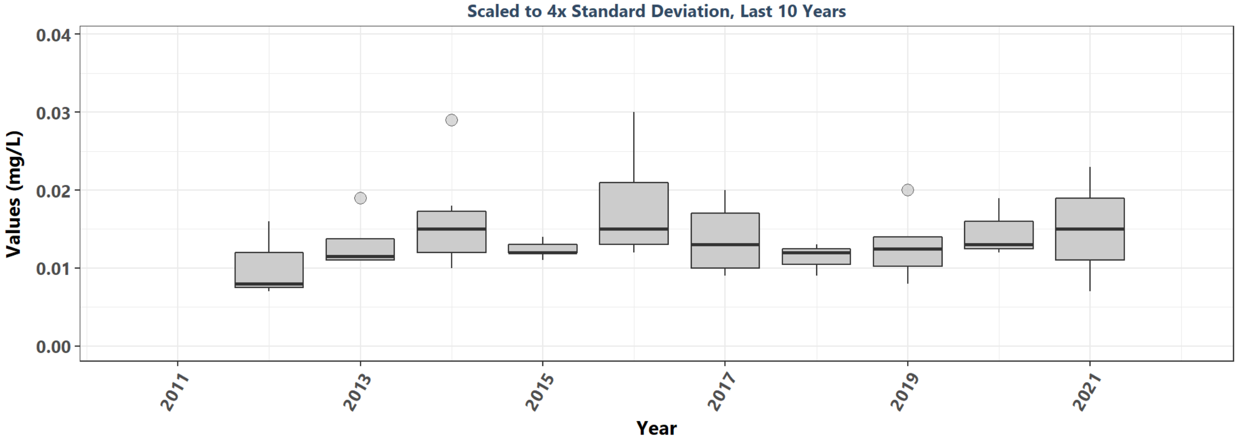
Autoscale

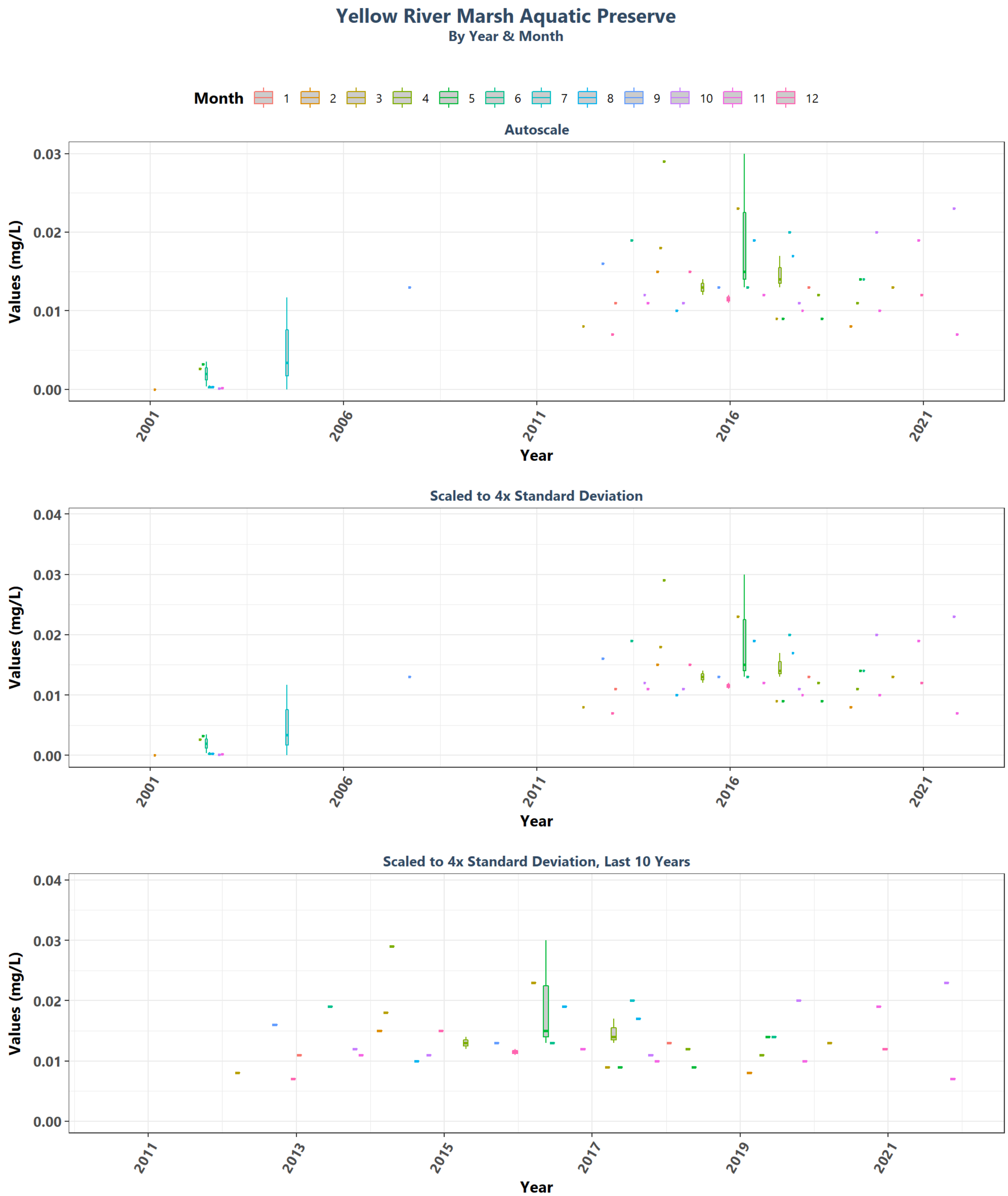


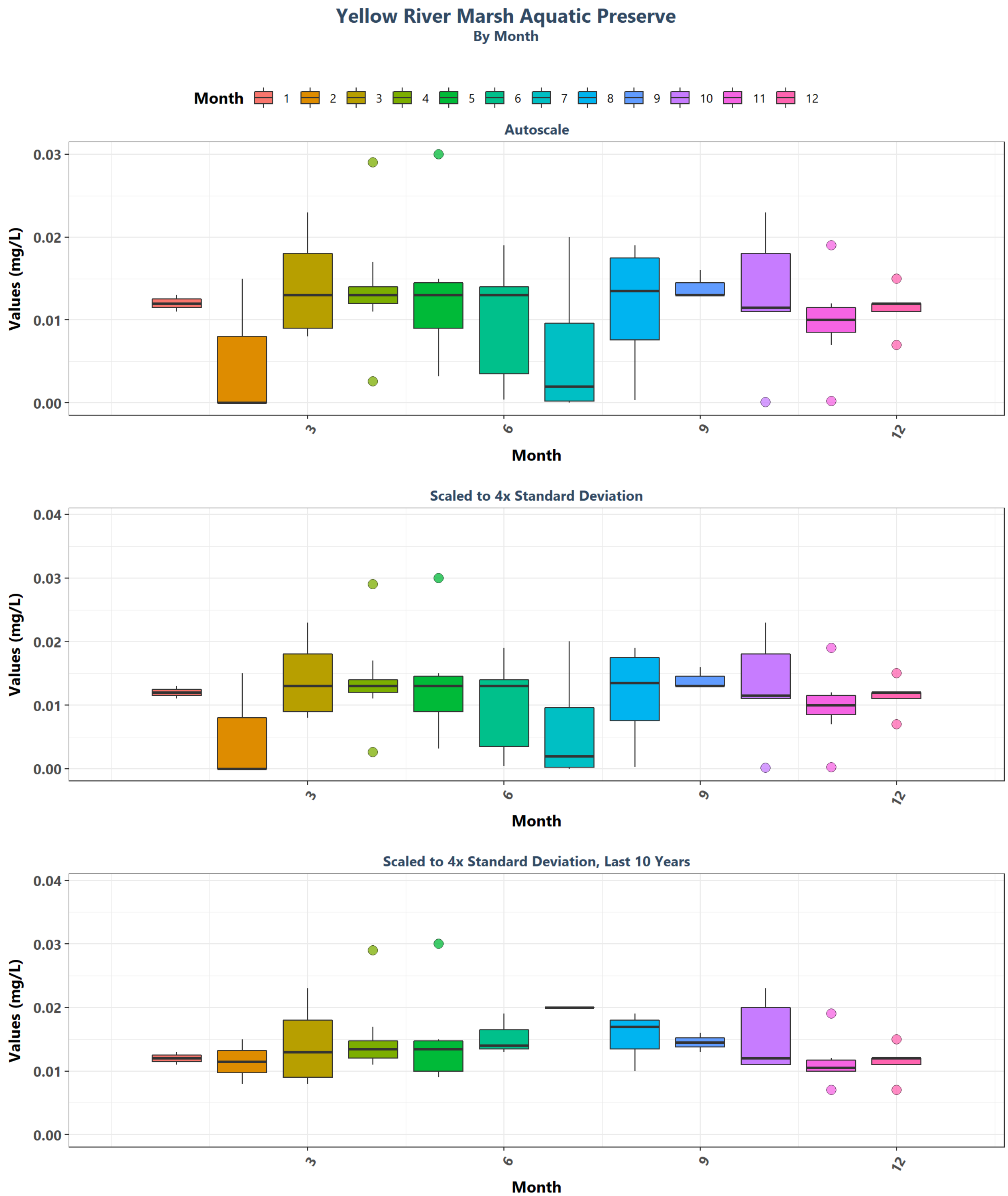
Scaled to 4x Standard Deviation



Scaled to 4x Standard Deviation, Last 10 Years




Yellow River Marsh Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (mg/L)
0.03
0.02
0.01
0.00
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (mg/L)
0.04
0.03
0.02
0.01
0.00
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (mg/L)
0.04
0.03
0.02
0.01
0.00
2011
2013
2015
2017
2019
2021
Year




```
rm(list = setdiff(ls(), c("all_params", "all_depths", "all_activity", "param_name", "depth", "activity"
```