SEACAR Discrete Water Quality Analysis: Field Surface pH

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 **24**

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)
```

```
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: 496521, Number Passed Filter: 496521
## Program 476 H Codes: 0 (0%)
## I Codes: 0 (0%)
## Q Codes: 1 (0.000201%)
## U Codes: 0 (0%)
```

```
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),
            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
```

```
      }
   }
   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)
```

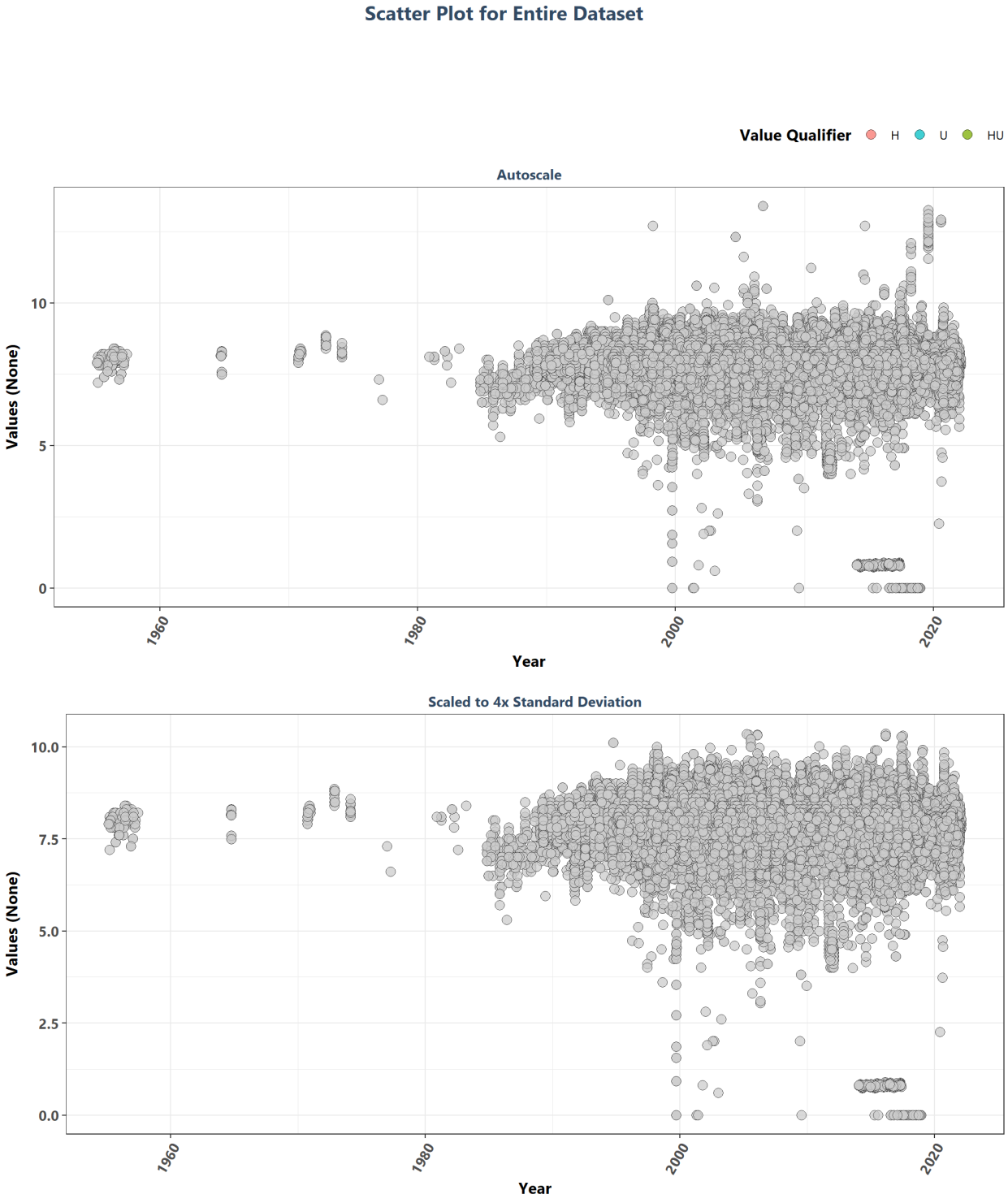
```
KT.Stats2$MonitoringID <- NULL
fwrite(KT.Stats2, paste0(out_dir,"/", param_name, "_", activity, "_", depth,
                         "_KendallTau_Stats.txt"), sep="|")
rm(KT.Stats2)
data <- data[!is.na(data$ResultValue),]
```

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))
```



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",
```

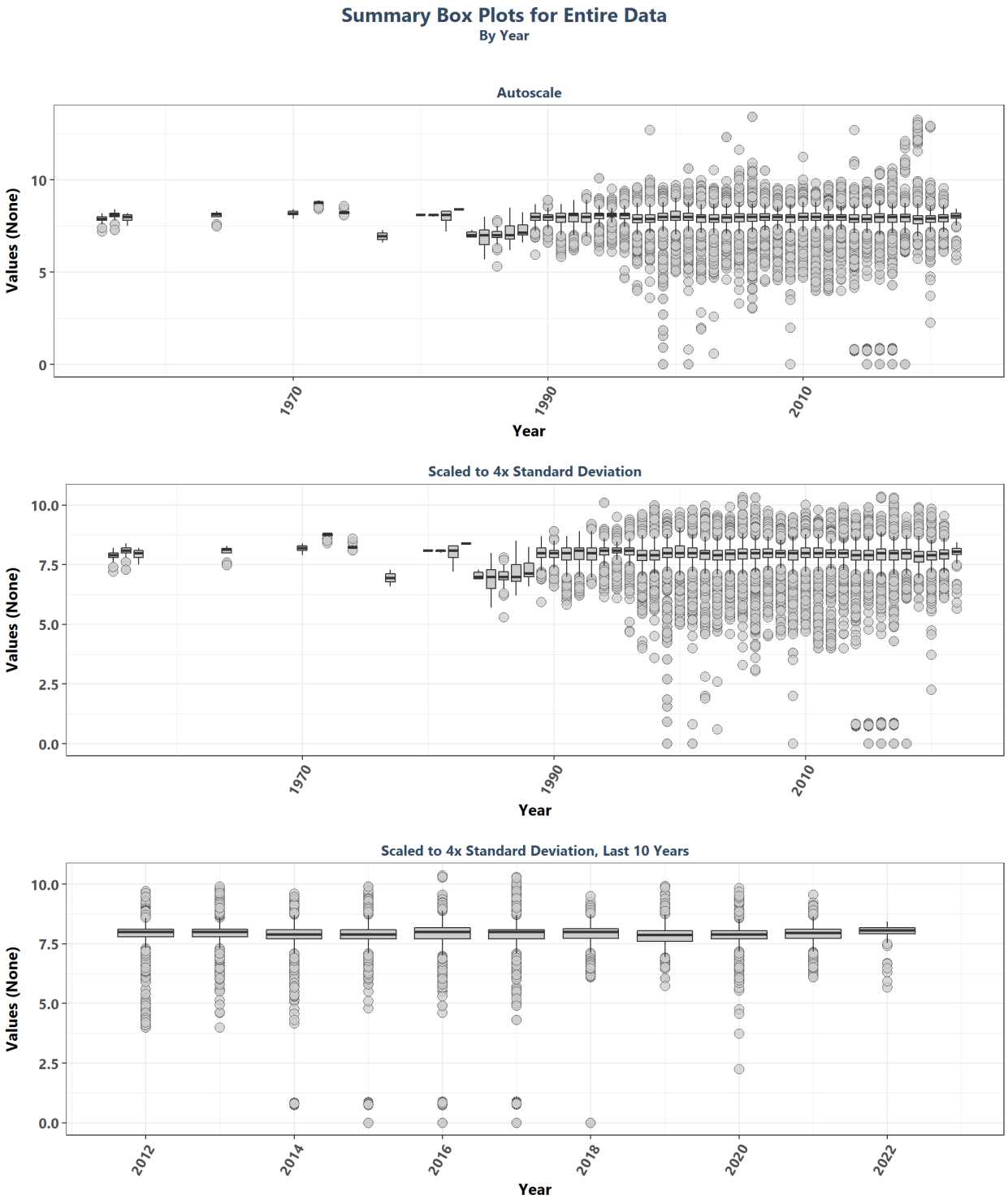
```
                      subtitle="By Year") + plot_theme +
   theme(panel.border=element_blank(), panel.grid.major=element_blank(),
         panel.grid.minor=element_blank(), axis.line=element_blank())

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

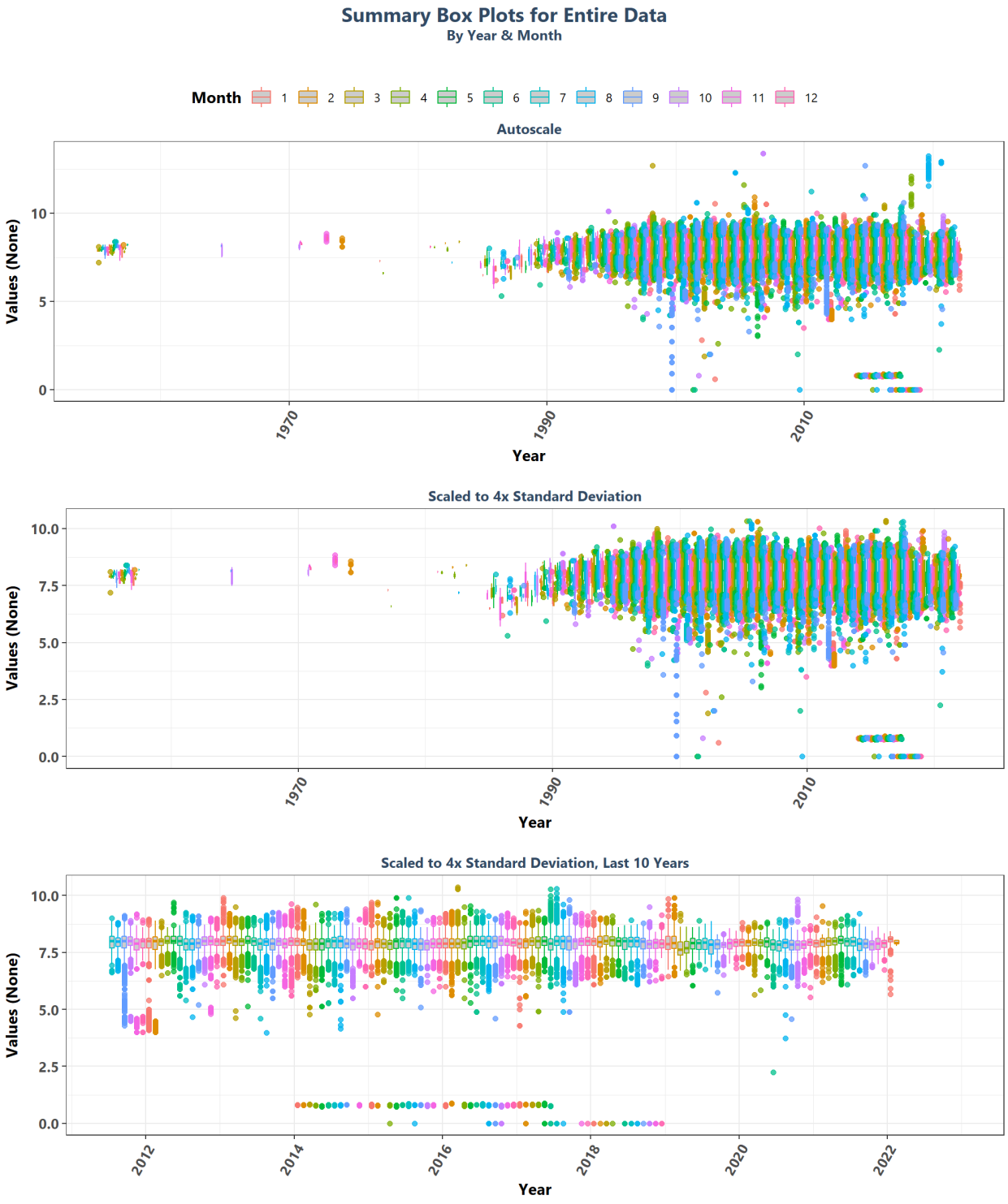
This set of box plots are grouped by year and month with the color being related to the month.

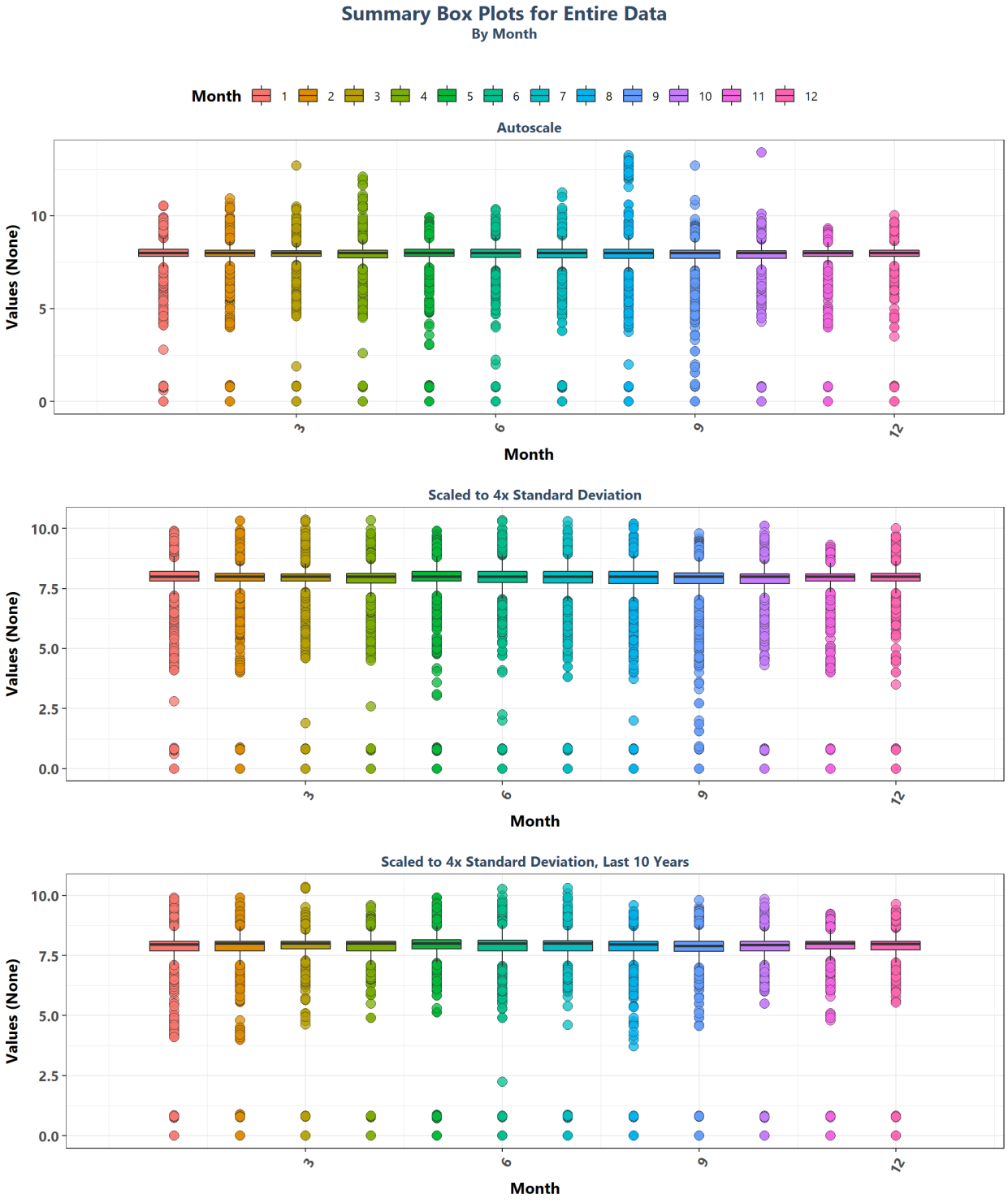
```
p1 <- ggplot(data=data[data$Include==TRUE, ],
             aes(x=YearMonthDec, y=ResultValue,
                 group=YearMonth, color=as.factor(Month))) +
   geom_boxplot(fill="#cccccc", outlier.size=1.5, outlier.alpha=0.75) +
   labs(subtitle="Autoscale", x="Year",
        y=paste0("Values (", unit,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

```
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
Autoscale
Values (None)
10
5
0
1970
1990
2010
Year
Scaled to 4x Standard Deviation
Values (None)
10.0
7.5
5.0
2.5
0.0
1970
1990
2010
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
10.0
7.5
5.0
2.5
0.0
2012
2014
2016
2018
2020
2022
Year

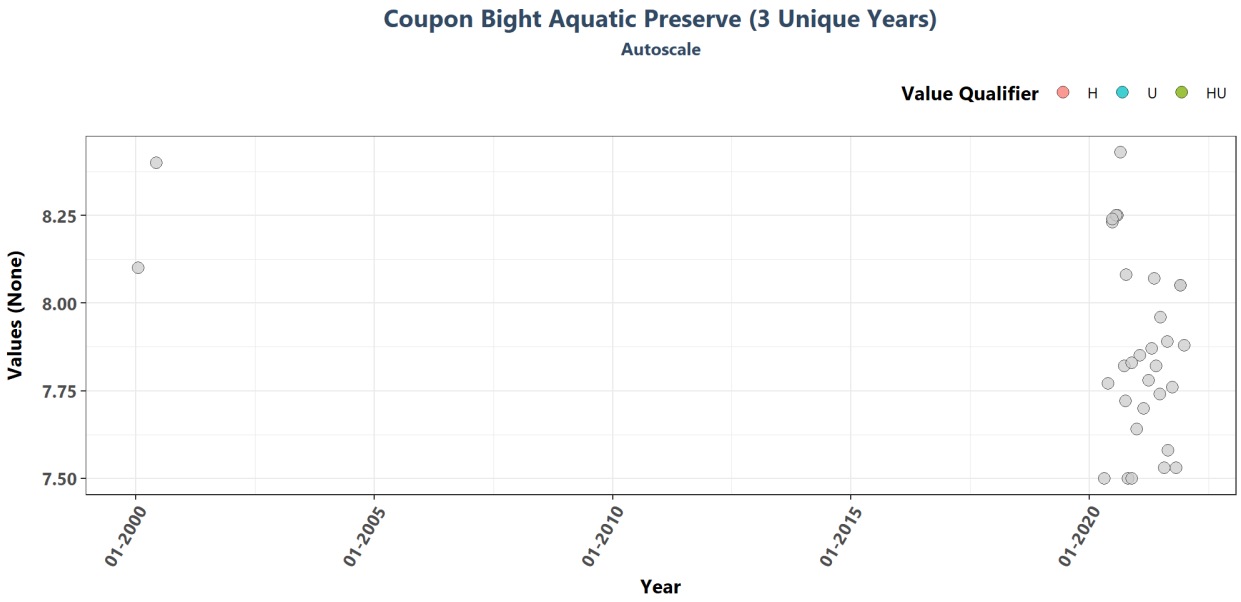

Summary Box Plots for Entire Data
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
10
5
0
1970
1990
2010
Year
Scaled to 4x Standard Deviation
Values (None)
10.0
7.5
5.0
2.5
0.0
1970
1990
2010
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
10.0
7.5
5.0
2.5
0.0
2012
2014
2016
2018
2020
2022
Year

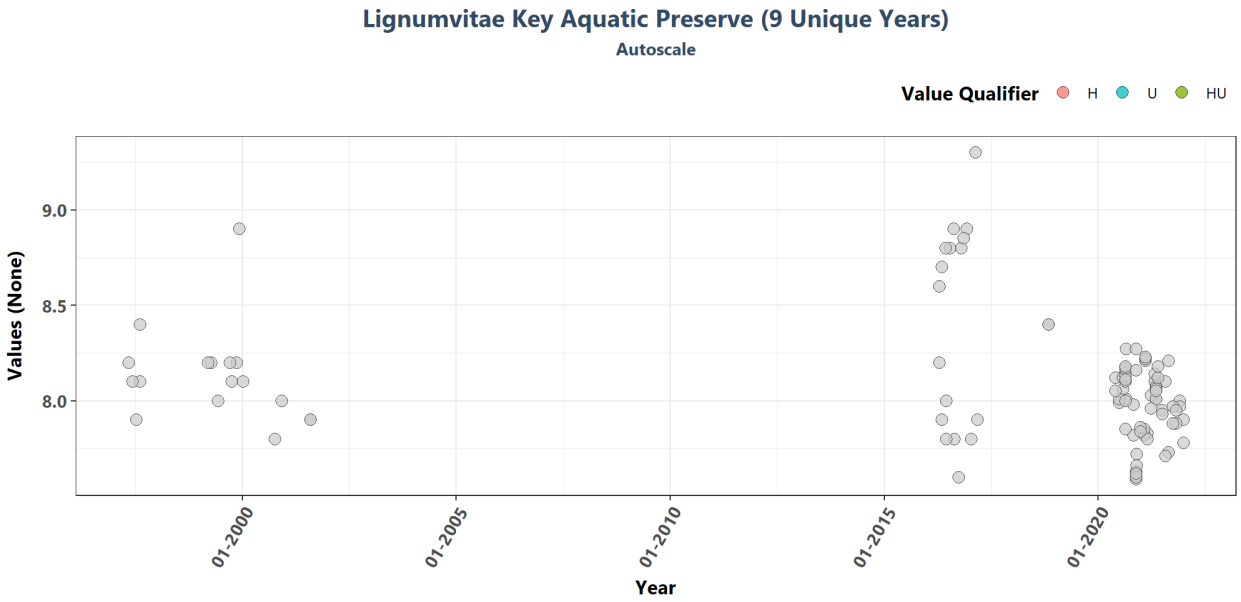


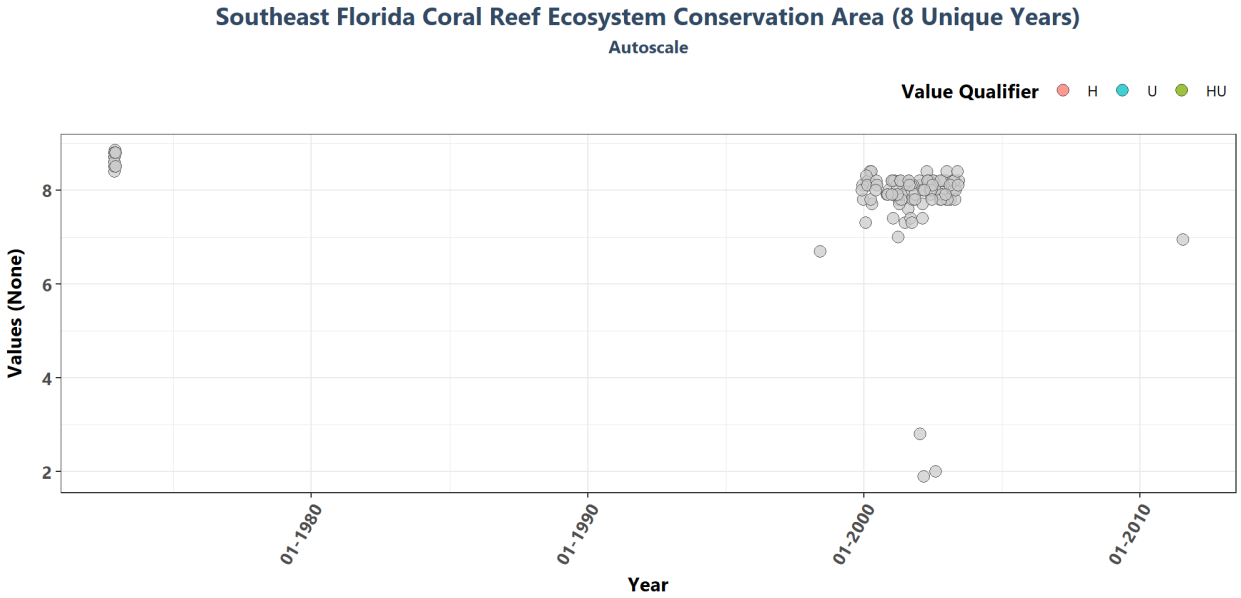
Appendix III: Excluded Managed Areas

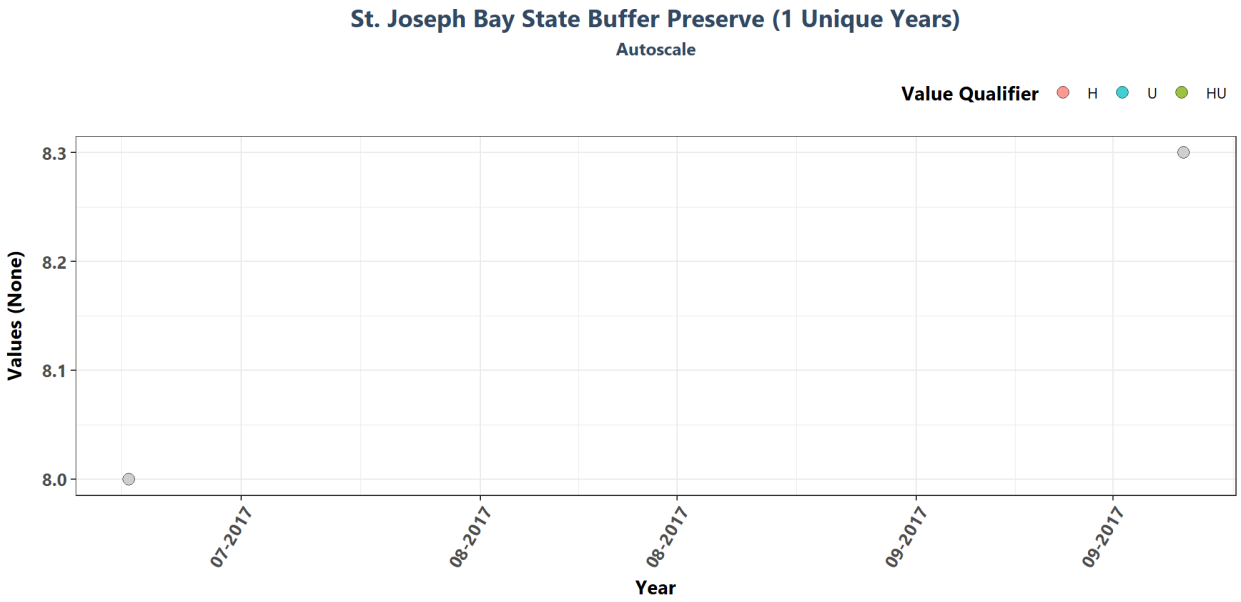
Scatter plots of data values are created for managed areas that have fewer than 10 separate years of data entries. Data points are colored based on specific value qualifiers of interest.

```
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)
   }
}
```




Lignumvitae Key Aquatic Preserve (9 Unique Years)
Autoscale
Value Qualifier H U HU
Values (None)
9.0
8.5
8.0
01-2000
01-2005
01-2010
01-2015
01-2020
Year


Southeast Florida Coral Reef Ecosystem Conservation Area (8 Unique Years)
Autoscale
Value Qualifier H U HU
Values (None)
8
6
4
2
01-1980
01-1990
01-2000
01-2010
Year



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

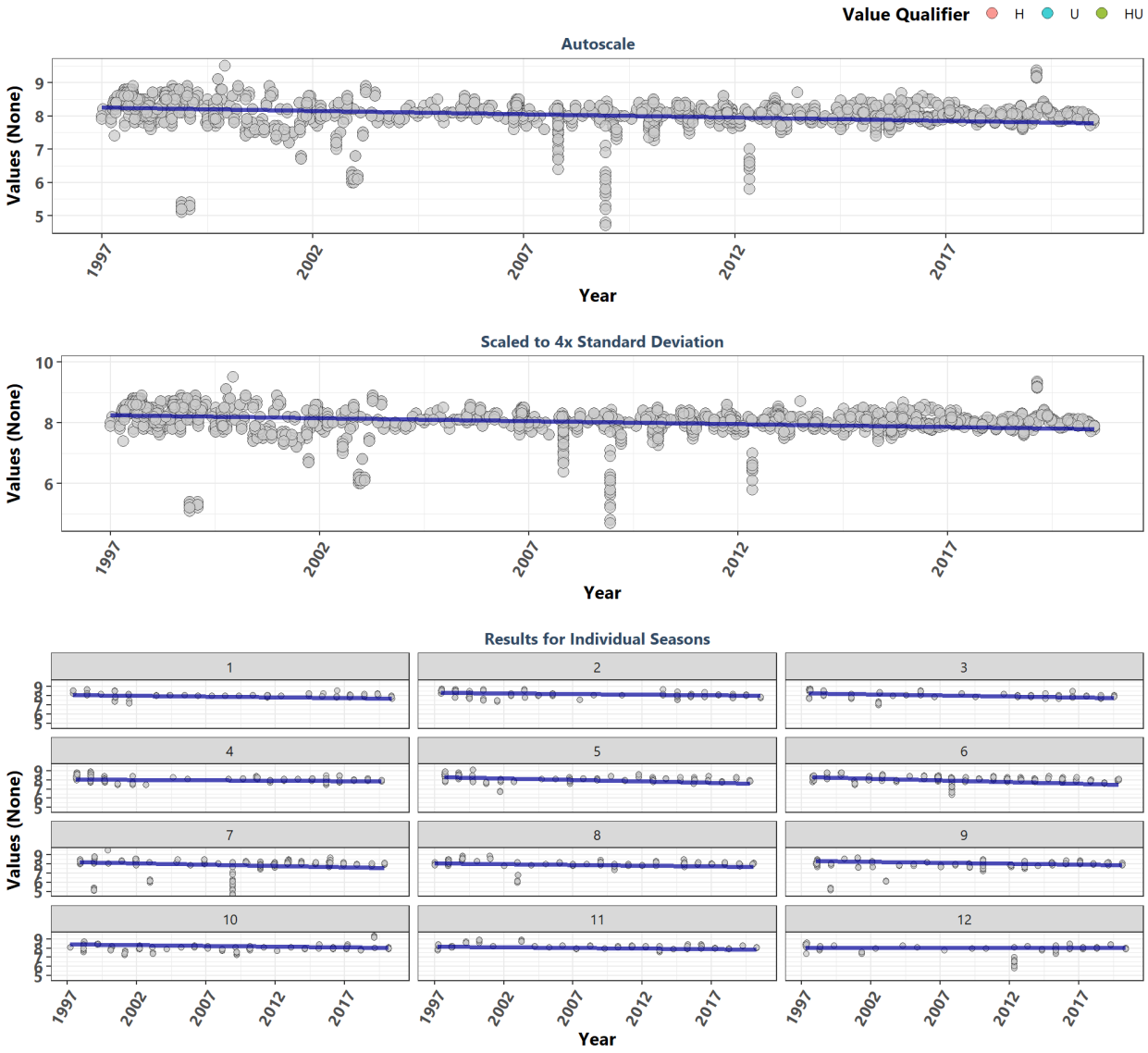
```
if(n==0){
   print("There are no managed areas that qualify.")
} else {
   for (i in 1:n) {
      plot_data <- data[data$SufficientData==TRUE &
                        data$ManagedAreaName==MA_Include[i],]
      plot_data$Season <- factor(plot_data$Month, levels = c("All", seq(1, 12)), ordered = 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$relyear_dd), 5, ceiling) - round_any(min(plot_data$rely

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 = rely
                                                                                         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 = rely
                                                                                          y = relyear_
                                                                                          xend = relye
                                                                                          yend = relye
                 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 = relye
                                                                                          yend = relye
                 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))
```

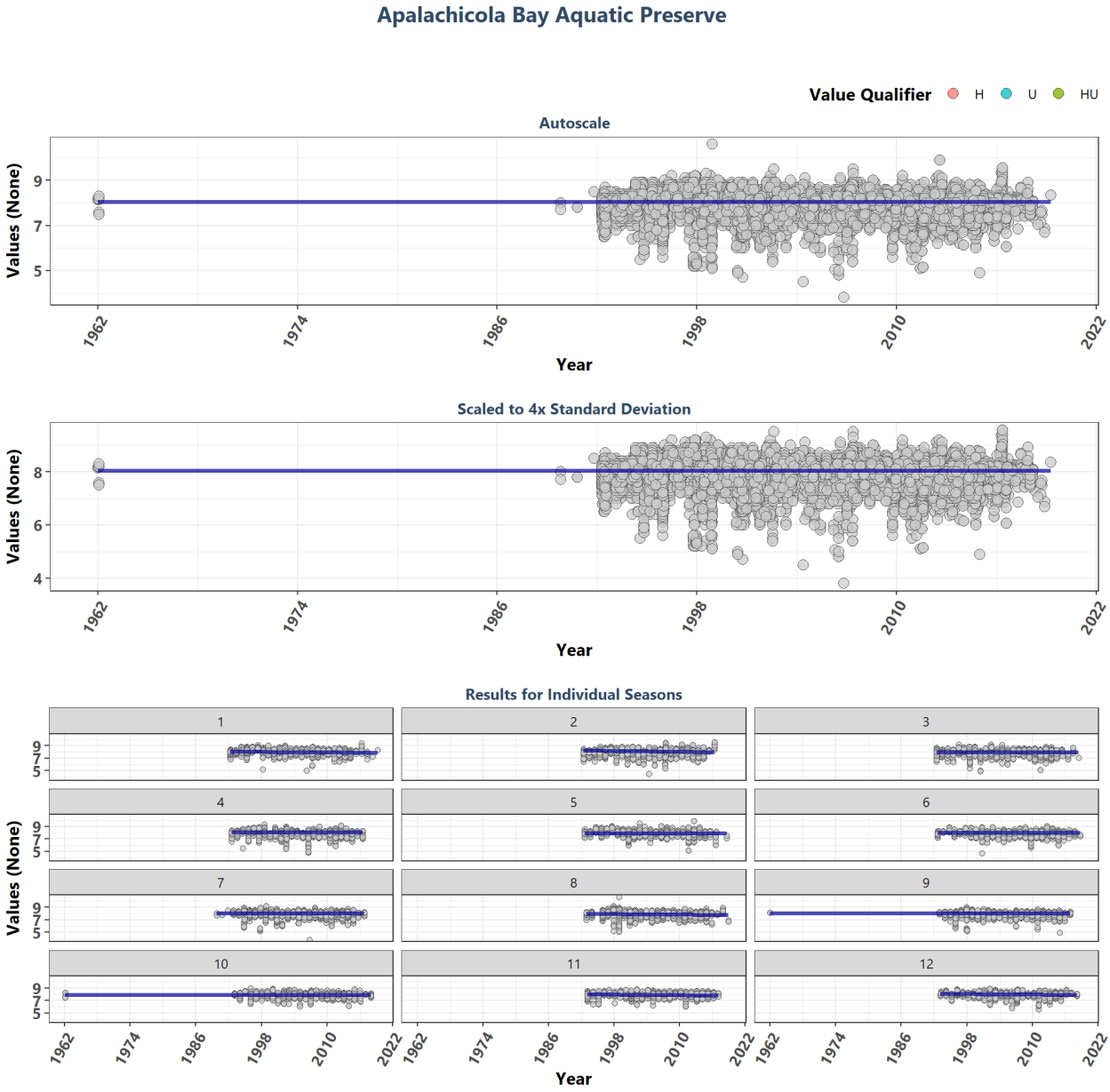
```
      p0 <- ggplot() + labs(title=paste0(MA_Include[i])) +
         plot_theme + theme(panel.border=element_blank(),
                            panel.grid.major=element_blank(),
                            panel.grid.minor=element_blank(),
                            axis.line=element_blank())

      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	4228	8.10	-0.2601	-0.0195	8.2517	-24.5	0.0000	128.4	0	-1
1	206	8.00	-0.1641	-0.0167	8.1000	-2.7	0.0070	NA	NA	-1
2	317	8.20	-0.2557	-0.0125	8.3250	-9.7	0.0000	NA	NA	-1
3	264	8.10	-0.3740	-0.0222	8.2889	-10.5	0.0000	NA	NA	-1
4	407	8.10	-0.1251	-0.0073	8.0364	-11.7	0.0000	NA	NA	-1
5	353	8.00	-0.3608	-0.0312	8.3250	-6.4	0.0000	NA	NA	-1
6	624	8.16	-0.4245	-0.0367	8.3200	-11.1	0.0000	NA	NA	-1
7	506	8.20	-0.3827	-0.0286	8.2145	-8.7	0.0000	NA	NA	-1
8	312	8.10	-0.2275	-0.0176	8.0882	-5.3	0.0000	NA	NA	-1
9	378	8.04	-0.2946	-0.0200	8.3750	-1.2	0.2157	NA	NA	-1
10	355	8.00	-0.2561	-0.0167	8.4000	-4.7	0.0000	NA	NA	-1
11	234	8.20	-0.1991	-0.0133	8.1800	-5.9	0.0000	NA	NA	-1
12	272	8.20	-0.0422	-0.0014	8.0521	-9.5	0.0000	NA	NA	-1

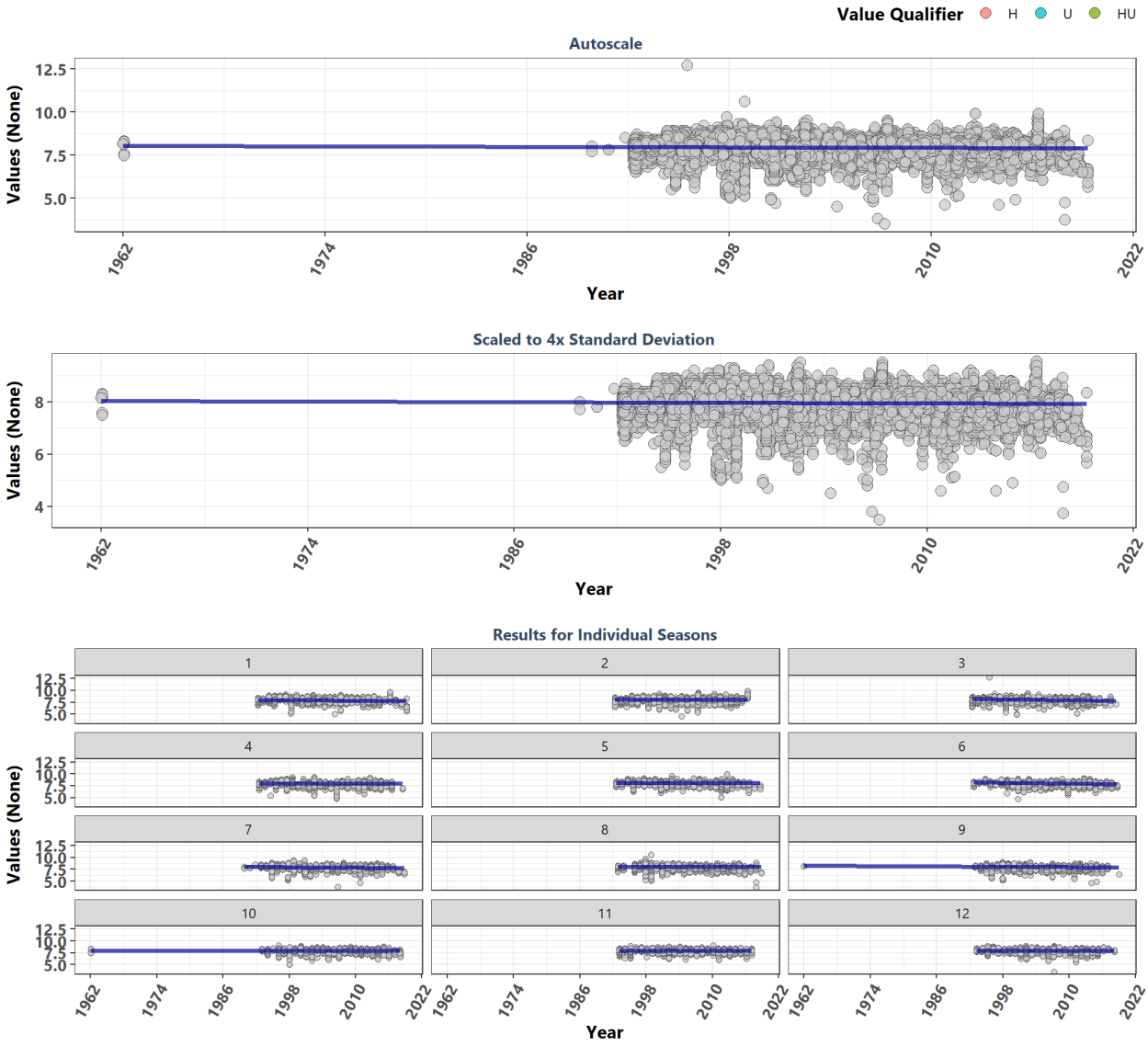
[a] $p < 0.00005$ appear as 0 due to rounding



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	29706	8.0	-0.0613	0.0000	8.0500	-15.7	0.0000	134	0	-1
1	2279	8.1	-0.1224	-0.0077	8.3308	-9.9	0.0000	NA	NA	-1
2	2269	8.0	-0.1378	-0.0105	8.5316	0.0	0.9623	NA	NA	-1
3	2712	7.9	-0.0486	0.0000	8.0000	-1.0	0.3106	NA	NA	-1
4	2175	7.9	-0.0549	0.0000	8.1000	-5.0	0.0000	NA	NA	-1
5	2346	8.0	-0.0129	0.0000	7.9000	-1.9	0.0527	NA	NA	-1
6	2688	8.0	-0.0265	0.0000	8.0000	-7.9	0.0000	NA	NA	-1
7	3188	8.0	-0.0714	0.0000	8.0000	-6.1	0.0000	NA	NA	-1
8	2894	7.9	-0.1014	-0.0077	8.2231	-8.2	0.0000	NA	NA	-1
9	2609	7.9	0.0007	0.0000	8.0000	0.3	0.7710	NA	NA	-1
10	2420	8.0	0.0038	0.0000	7.9000	-3.6	0.0003	NA	NA	-1
11	2187	8.1	-0.0707	-0.0056	8.1389	-3.9	0.0001	NA	NA	-1
12	1939	8.0	-0.1013	-0.0071	8.3143	-8.1	0.0000	NA	NA	-1

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

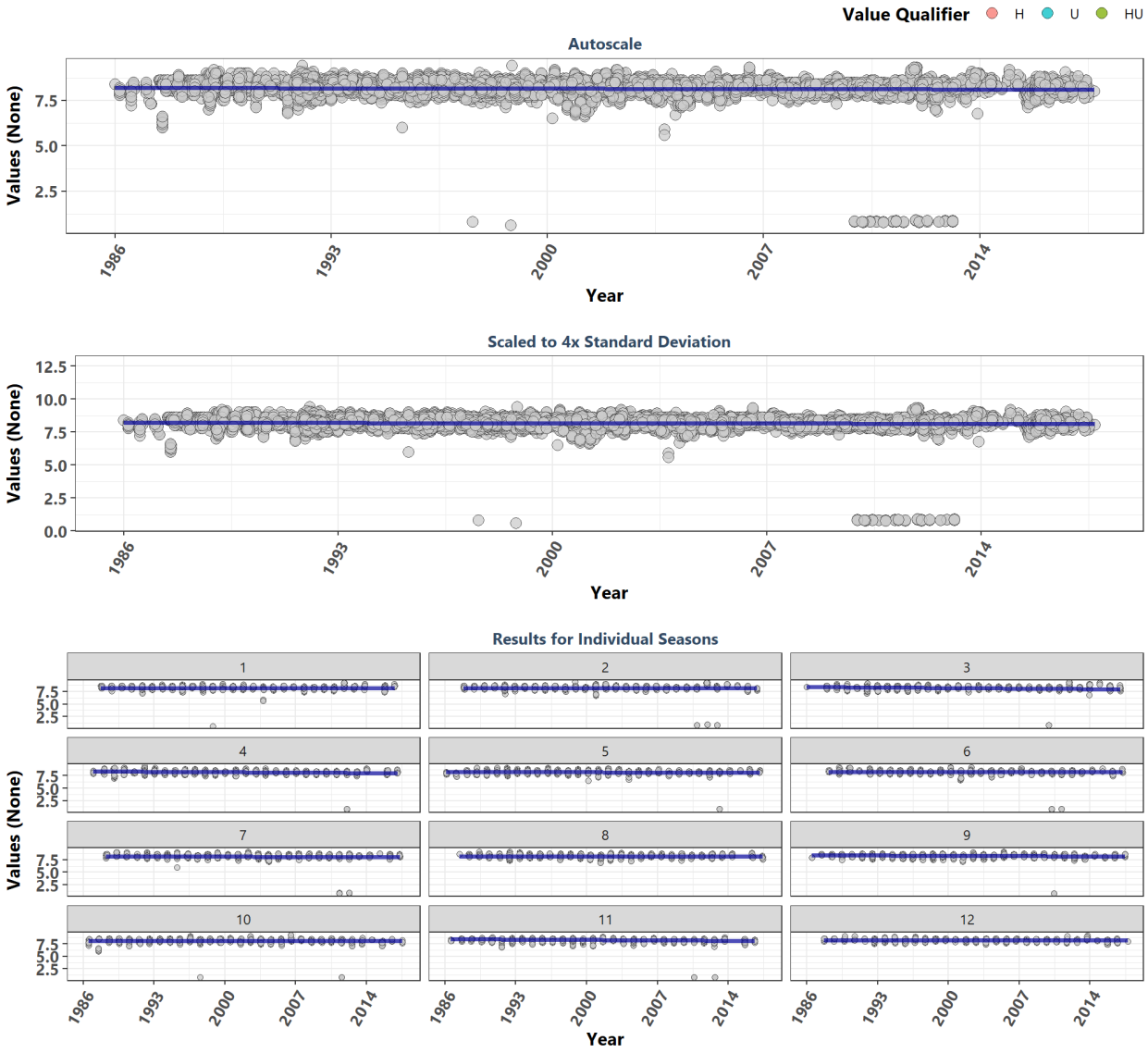
Apalachicola National Estuarine Research Reserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	45516	8.0	-0.0742	-0.0022	8.0500	-23.4	0.0000	163.1	0	-1
1	3617	8.1	-0.0730	-0.0056	8.1389	-14.7	0.0000	NA	NA	-1
2	3651	8.0	-0.0411	0.0000	8.0000	-2.9	0.0033	NA	NA	-1
3	4392	7.9	-0.1360	-0.0100	8.4300	-5.0	0.0000	NA	NA	-1
4	3466	7.9	-0.0323	0.0000	8.0000	-6.5	0.0000	NA	NA	-1
5	3604	8.0	-0.0748	0.0000	8.1000	-3.7	0.0002	NA	NA	-1
6	3856	8.0	-0.1619	-0.0143	8.7000	-8.3	0.0000	NA	NA	-1
7	4502	8.0	-0.1052	-0.0077	8.2308	-7.5	0.0000	NA	NA	-1
8	4151	7.9	-0.0633	0.0000	8.0000	-10.2	0.0000	NA	NA	-1
9	3915	7.9	-0.0882	-0.0059	8.2588	-0.6	0.5332	NA	NA	-1
10	3946	8.0	-0.0500	0.0000	7.9000	-6.0	0.0000	NA	NA	-1
11	3356	8.1	-0.0741	0.0000	8.0000	-6.5	0.0000	NA	NA	-1
12	3060	8.0	-0.0066	0.0000	7.9000	-11.3	0.0000	NA	NA	-1

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

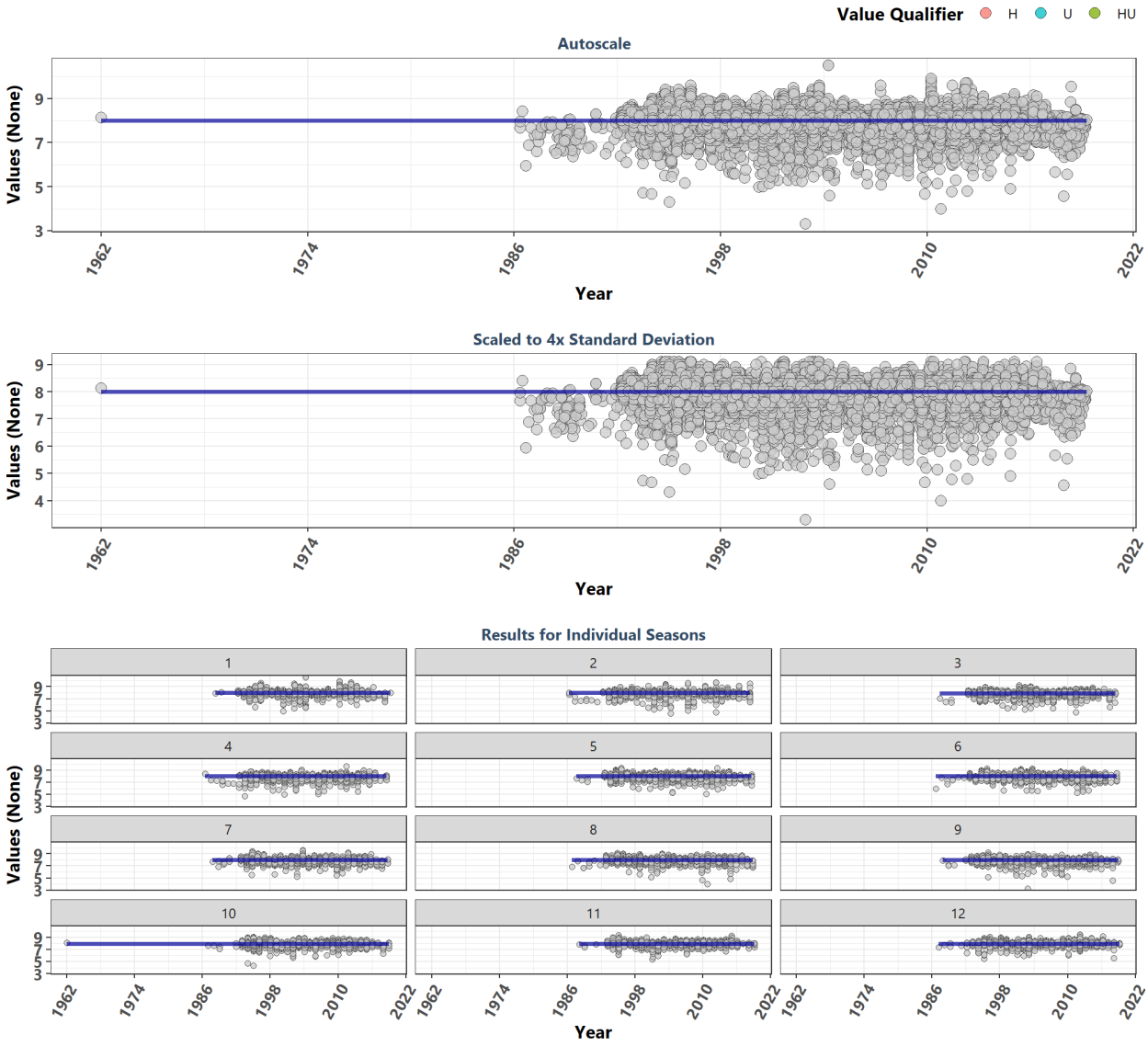
Banana River Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	20243	8.2	-0.0814	-0.0031	8.2127	-17.1	0.0000	187.2	0	-1
1	1321	8.3	-0.0272	0.0000	8.2000	-7.4	0.0000	NA	NA	-1
2	1798	8.3	-0.0436	0.0000	8.2000	-8.7	0.0000	NA	NA	-1
3	1578	8.2	-0.1996	-0.0140	8.3820	2.3	0.0220	NA	NA	-1
4	1869	8.2	-0.1480	-0.0094	8.3228	-7.2	0.0000	NA	NA	-1
5	2174	8.2	-0.1100	-0.0071	8.2857	-3.1	0.0021	NA	NA	-1
6	1679	8.2	-0.0270	0.0000	8.2000	-12.4	0.0000	NA	NA	-1
7	1774	8.2	-0.0906	-0.0048	8.1571	-9.4	0.0000	NA	NA	-1
8	1748	8.2	-0.0754	-0.0023	8.2254	-1.7	0.0855	NA	NA	-1
9	1420	8.2	-0.1354	-0.0075	8.3900	-1.5	0.1227	NA	NA	-1
10	2023	8.1	-0.0310	0.0000	8.1000	-2.1	0.0356	NA	NA	-1
11	1652	8.1	-0.1344	-0.0100	8.4200	-5.6	0.0000	NA	NA	-1
12	1207	8.2	0.0382	0.0000	8.2000	-4.0	0.0001	NA	NA	-1

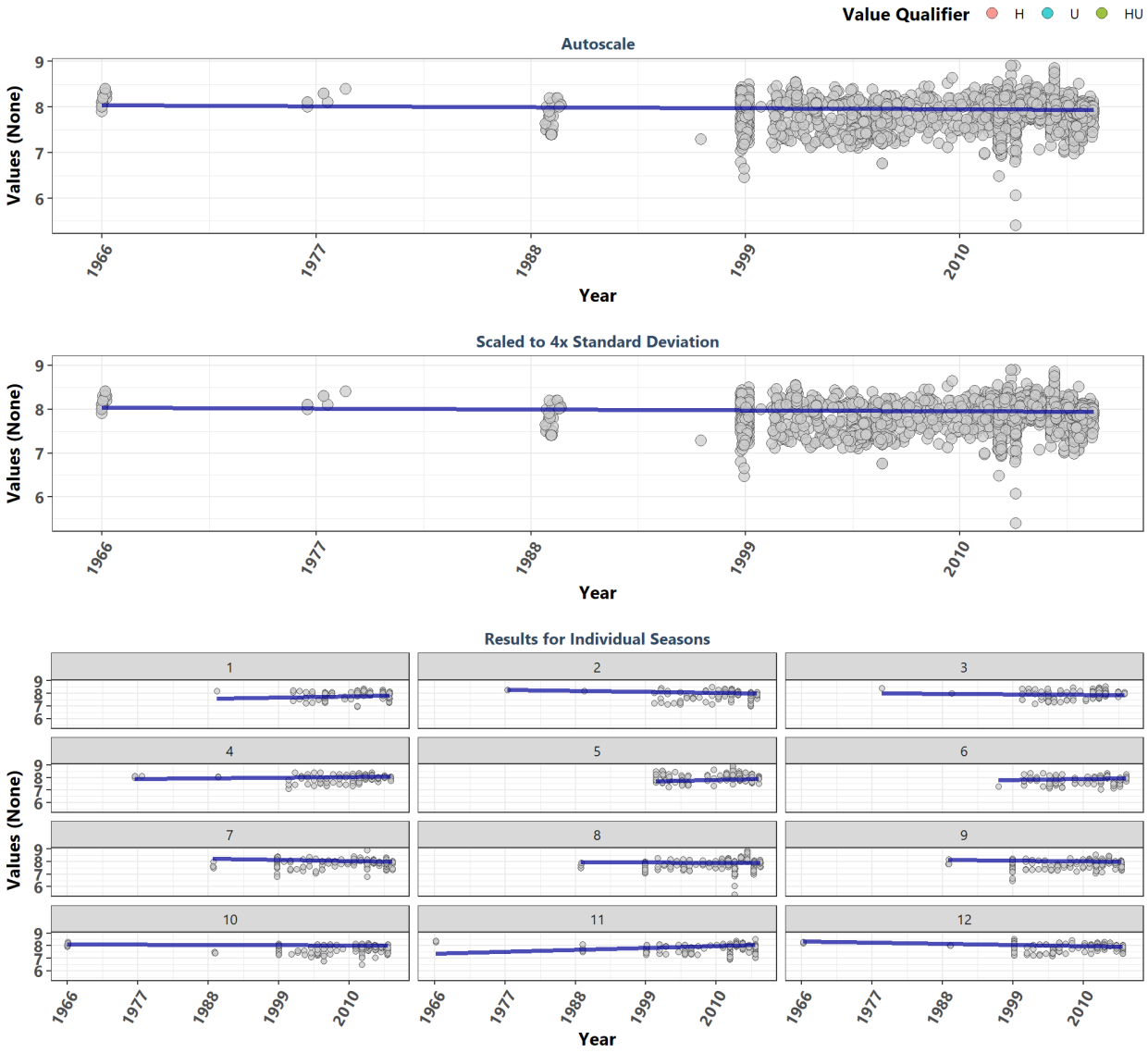
[a] $p < 0.00005$ appear as 0 due to rounding

Big Bend Seagrasses Aquatic Preserve



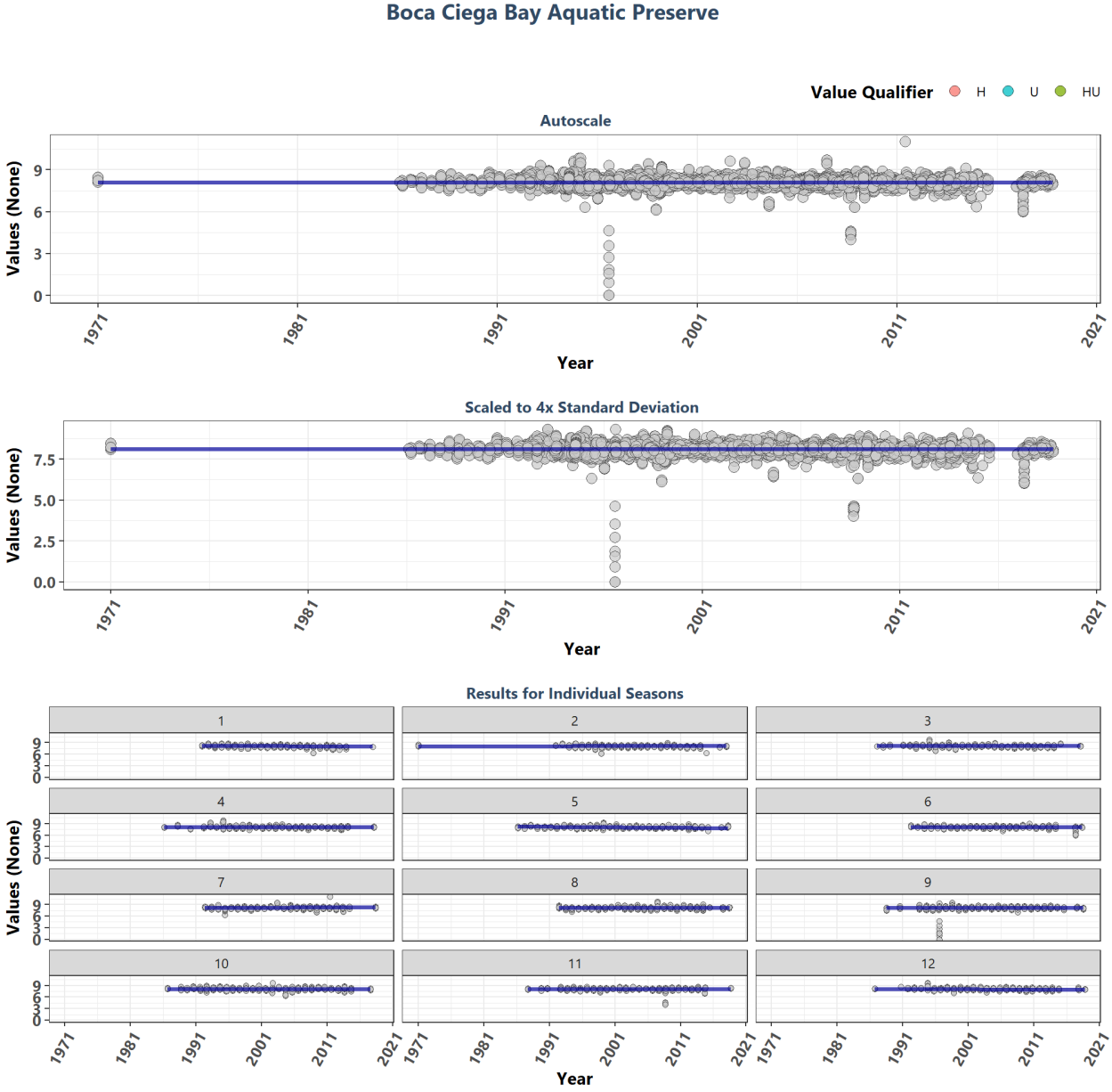
Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	80381	8.00	-0.0465	0	8.00	-20.3	0.0000	153.3	0	-1
1	5483	8.00	-0.0589	0	8.00	-7.6	0.0000	NA	NA	-1
2	5518	8.00	-0.0859	0	8.00	1.3	0.2102	NA	NA	-1
3	6855	8.00	-0.0417	0	7.90	-11.7	0.0000	NA	NA	-1
4	6787	7.90	-0.0931	0	8.00	-5.2	0.0000	NA	NA	-1
5	7139	8.00	-0.0344	0	8.00	-7.5	0.0000	NA	NA	-1
6	6507	8.00	-0.0451	0	8.00	-5.5	0.0000	NA	NA	-1
7	7292	7.90	-0.0349	0	8.00	-2.4	0.0150	NA	NA	-1
8	7832	8.00	0.0111	0	8.00	-11.5	0.0000	NA	NA	-1
9	7443	7.90	-0.0673	0	8.00	-7.3	0.0000	NA	NA	-1
10	7259	7.96	-0.0189	0	7.90	-2.4	0.0143	NA	NA	-1
11	6298	8.00	-0.0190	0	7.96	-4.2	0.0000	NA	NA	-1
12	5968	8.00	-0.0562	0	7.90	-4.0	0.0001	NA	NA	-1

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



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	3434	7.96	-0.0267	-0.0020	8.0396	-2.8	0.0048	85.4	0	-1
1	251	8.01	0.0939	0.0083	7.4117	-2.5	0.0140	NA	NA	1
2	206	8.00	-0.1596	-0.0080	8.3860	-1.1	0.2717	NA	NA	-1
3	181	8.03	-0.1073	-0.0038	8.0662	1.9	0.0615	NA	NA	-1
4	171	8.05	0.0945	0.0050	7.8100	1.9	0.0631	NA	NA	1
5	293	8.03	0.0901	0.0127	7.2491	-0.6	0.5535	NA	NA	1
6	213	7.86	0.0874	0.0056	7.6389	2.0	0.0470	NA	NA	1
7	340	8.01	-0.1016	-0.0080	8.3940	-4.5	0.0000	NA	NA	-1
8	379	7.80	-0.0332	-0.0018	8.0129	2.8	0.0058	NA	NA	-1
9	361	7.93	-0.0498	-0.0058	8.2719	-1.0	0.3405	NA	NA	-1
10	358	7.89	-0.0230	-0.0018	8.1155	-3.1	0.0022	NA	NA	-1
11	346	7.90	0.0919	0.0142	7.3642	2.5	0.0139	NA	NA	1
12	335	7.95	-0.1866	-0.0080	8.3180	-5.2	0.0000	NA	NA	-1

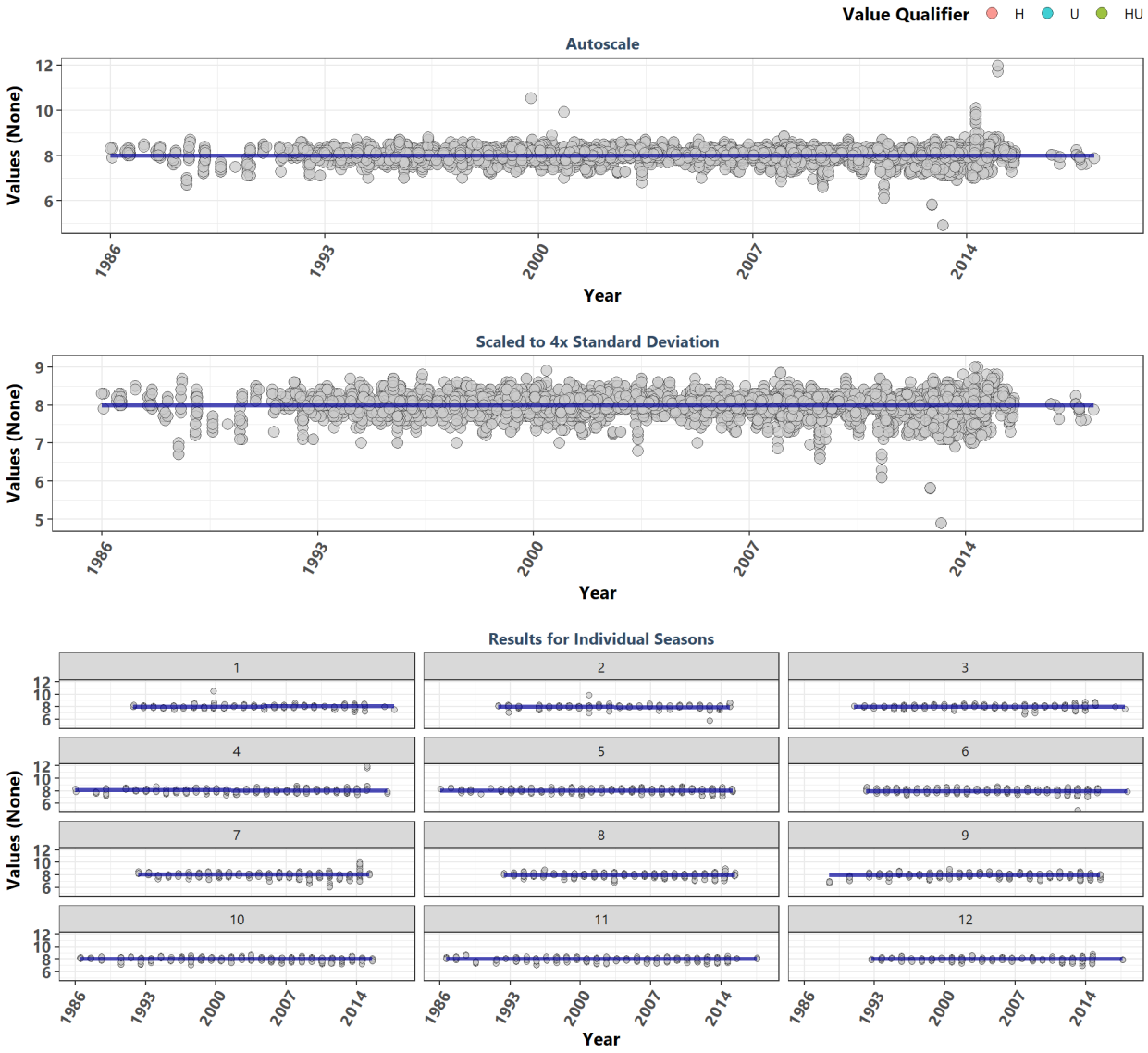
[a] $p < 0.00005$ appear as 0 due to rounding



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	16132	8.10	-0.0109	0.0000	8.1000	1.0	0.3225	338.6	0	0
1	1107	8.05	-0.0711	-0.0020	8.1660	-14.1	0.0000	NA	NA	0
2	1122	8.04	0.0767	0.0025	8.0450	-0.9	0.3737	NA	NA	0
3	1089	8.04	0.0218	0.0000	8.1400	-5.2	0.0000	NA	NA	0
4	1004	8.10	0.0592	0.0011	8.0922	-1.7	0.0924	NA	NA	0
5	1544	8.10	-0.2814	-0.0140	8.4840	1.6	0.1206	NA	NA	0
6	1073	8.10	0.0201	0.0000	8.1000	1.0	0.3223	NA	NA	0
7	1176	8.13	0.0993	0.0040	8.0380	3.1	0.0023	NA	NA	0
8	1437	8.13	-0.0353	0.0000	8.1000	4.4	0.0000	NA	NA	0
9	2241	8.15	-0.0893	-0.0011	8.1337	7.1	0.0000	NA	NA	0
10	1857	8.14	0.0262	0.0000	8.1000	1.4	0.1584	NA	NA	0
11	1263	8.10	-0.0176	0.0000	8.0400	-4.8	0.0000	NA	NA	0
12	1219	8.10	-0.1049	-0.0043	8.1771	-3.7	0.0002	NA	NA	0

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

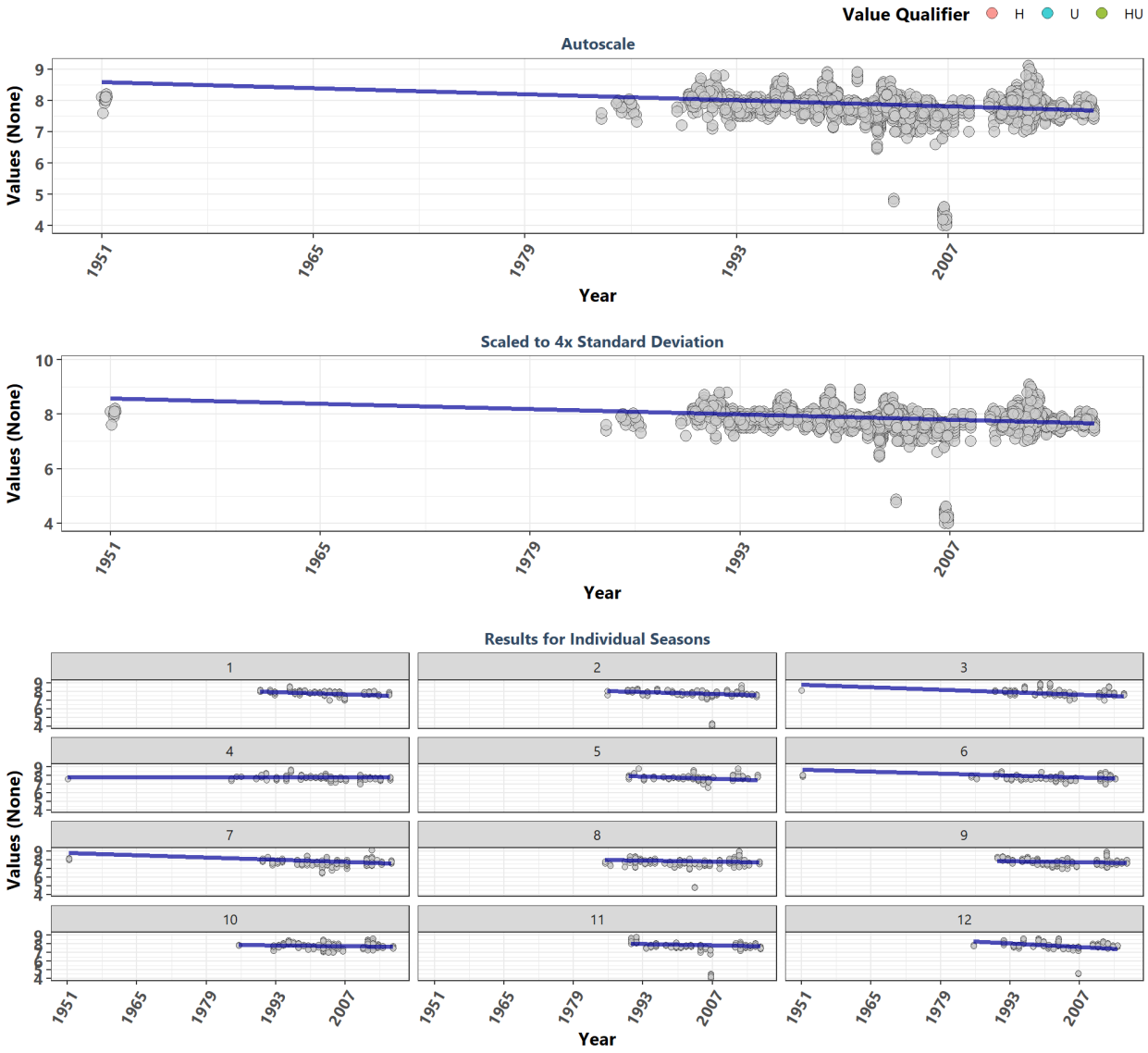
Cape Haze Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	6975	8.00	-0.0078	0.0000	8.0000	-1.4	0.1721	55.3	0	0
1	461	8.00	0.1213	0.0025	8.0500	1.6	0.1072	NA	NA	0
2	487	8.00	-0.0901	-0.0027	8.0507	1.4	0.1546	NA	NA	0
3	510	8.10	-0.0687	0.0000	8.0000	4.1	0.0000	NA	NA	0
4	592	8.07	-0.0785	-0.0018	8.1355	2.1	0.0386	NA	NA	0
5	649	8.10	0.0563	0.0000	8.0650	-0.1	0.9560	NA	NA	0
6	560	8.10	-0.0414	0.0000	8.0000	-2.8	0.0052	NA	NA	0
7	719	8.00	-0.0014	0.0000	8.1000	-0.9	0.3428	NA	NA	0
8	597	8.00	0.0068	0.0000	8.0000	-0.9	0.3489	NA	NA	0
9	619	8.00	-0.0255	0.0000	8.0000	0.3	0.7980	NA	NA	0
10	667	8.00	-0.0235	0.0000	8.0000	-3.5	0.0005	NA	NA	0
11	602	8.00	0.0497	0.0000	8.0000	-1.5	0.1259	NA	NA	0
12	512	8.00	0.0424	0.0000	8.0000	-2.3	0.0189	NA	NA	0

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

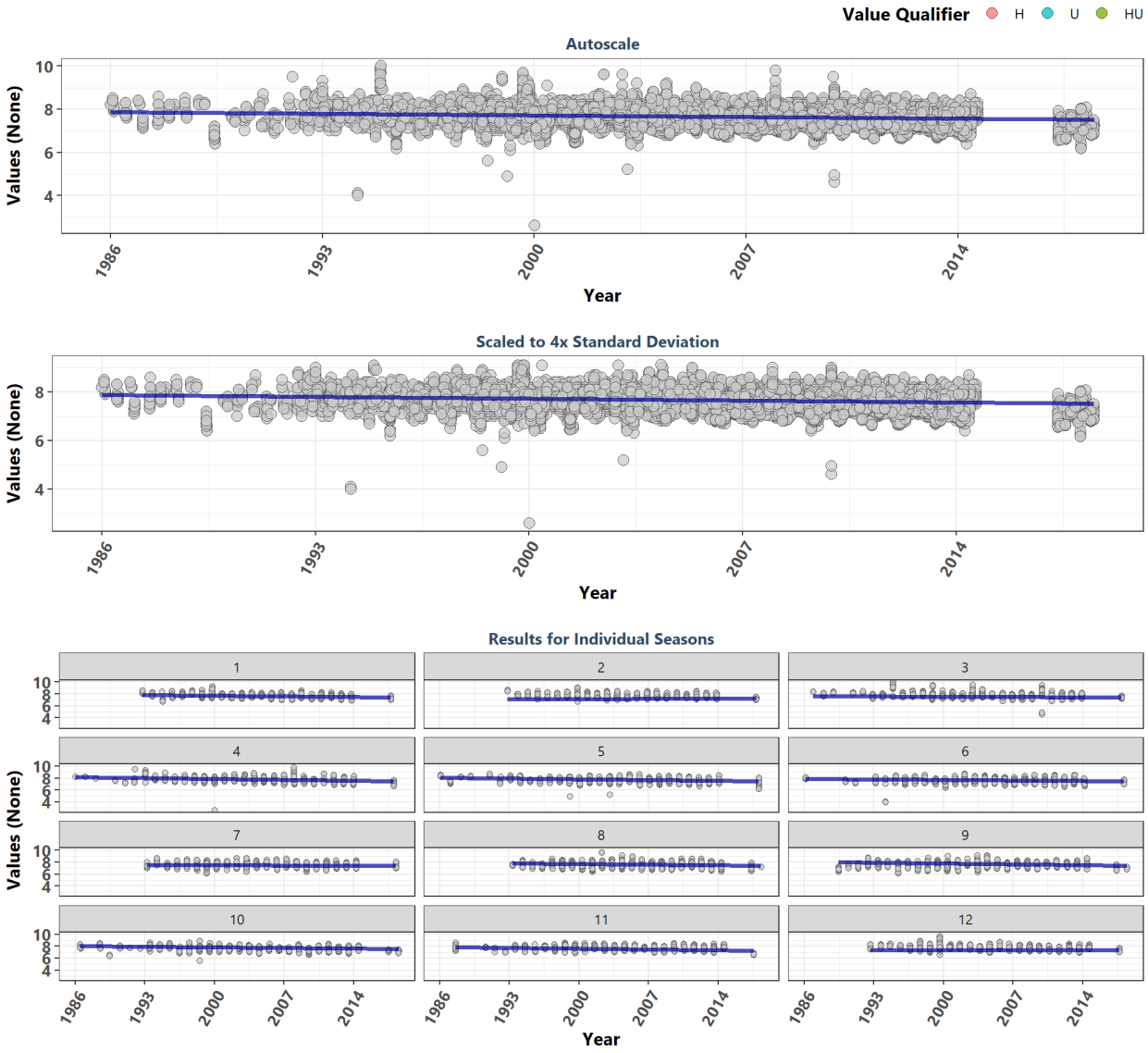
Cape Romano-Ten Thousand Islands Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p__z	chi__sq	p__chi__sq	Trend
All	4262	7.80	-0.2474	-0.0139	8.5885	-23.8	0.0000	120.5	0	-1
1	337	7.90	-0.2606	-0.0160	8.5840	-11.1	0.0000	NA	NA	-1
2	355	7.85	-0.3198	-0.0167	8.6500	-9.4	0.0000	NA	NA	-1
3	260	7.94	-0.2410	-0.0200	8.7800	-4.6	0.0000	NA	NA	-1
4	330	7.80	0.0039	0.0000	7.8000	-8.7	0.0000	NA	NA	-1
5	319	7.80	-0.2973	-0.0165	8.5930	-6.5	0.0000	NA	NA	-1
6	393	7.80	-0.3317	-0.0167	8.7167	-5.5	0.0000	NA	NA	-1
7	326	7.80	-0.4007	-0.0186	8.7914	-3.6	0.0003	NA	NA	-1
8	446	7.80	-0.2399	-0.0091	8.2818	-8.3	0.0000	NA	NA	-1
9	390	7.80	-0.1849	-0.0087	8.1937	-11.7	0.0000	NA	NA	-1
10	413	7.80	-0.1343	-0.0064	8.1279	0.1	0.9064	NA	NA	-1
11	330	7.80	-0.1892	-0.0104	8.4350	-8.1	0.0000	NA	NA	-1
12	363	7.80	-0.3942	-0.0284	9.2495	-6.9	0.0000	NA	NA	-1

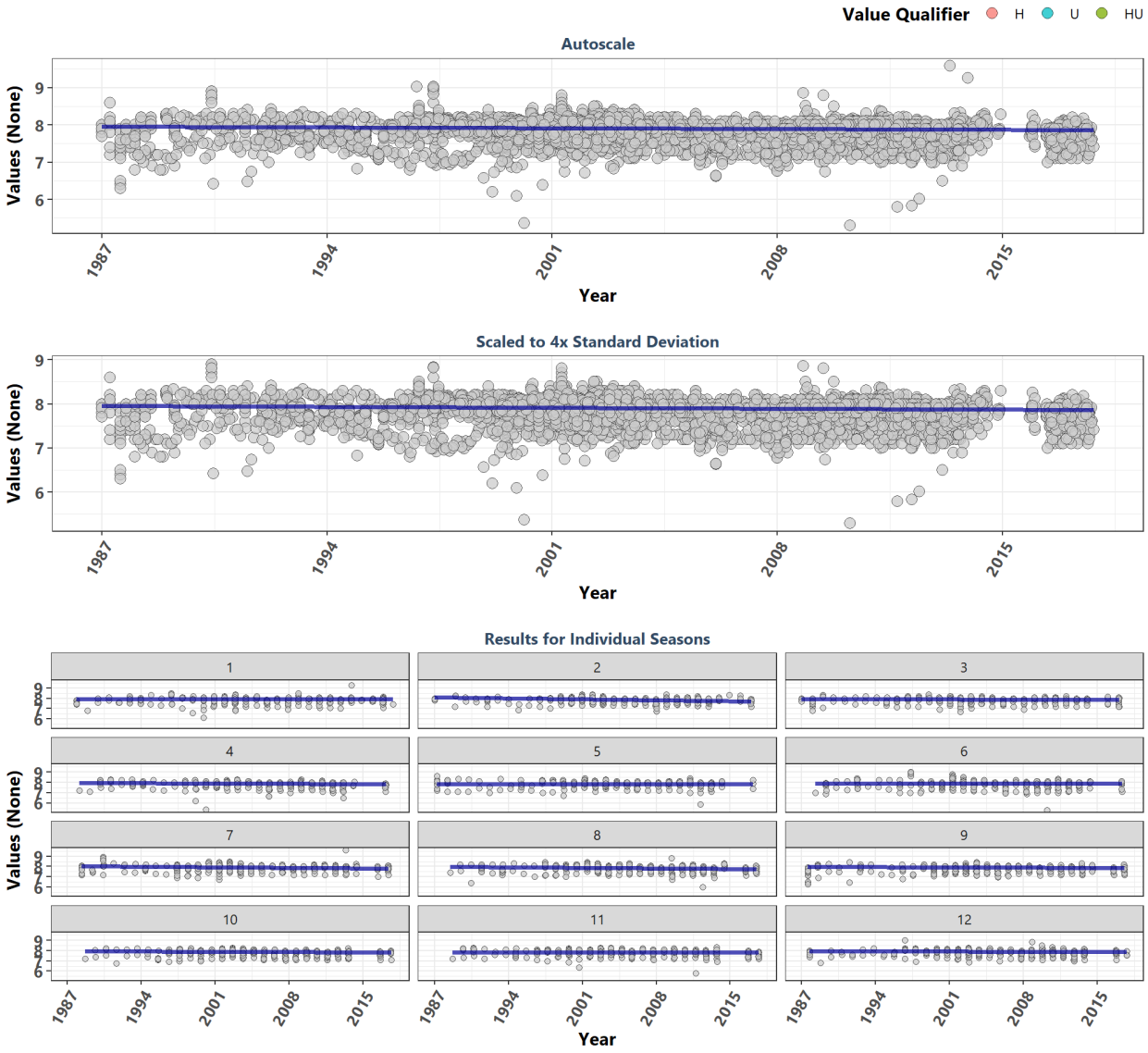
[a] $p < 0.00005$ appear as 0 due to rounding

Cockroach Bay Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	14348	7.60	-0.1285	-0.0106	7.8703	-18.8	0.0000	259.5	0	-1
1	973	7.70	-0.1813	-0.0157	7.8986	-9.6	0.0000	NA	NA	-1
2	975	7.70	0.0288	0.0042	7.1375	-11.7	0.0000	NA	NA	1
3	1045	7.70	-0.0900	-0.0077	7.6538	-9.1	0.0000	NA	NA	-1
4	1013	7.60	-0.2494	-0.0221	8.1411	-9.2	0.0000	NA	NA	-1
5	1199	7.60	-0.2049	-0.0189	8.0778	-7.5	0.0000	NA	NA	-1
6	988	7.50	-0.1432	-0.0125	7.8375	-4.3	0.0000	NA	NA	-1
7	1112	7.50	-0.0537	-0.0044	7.5844	-2.7	0.0071	NA	NA	-1
8	1954	7.20	-0.1912	-0.0167	7.9333	1.9	0.0527	NA	NA	-1
9	1638	7.39	-0.2145	-0.0182	8.0282	0.3	0.7547	NA	NA	-1
10	1216	7.50	-0.1865	-0.0154	8.0077	-13.5	0.0000	NA	NA	-1
11	1174	7.70	-0.2571	-0.0190	7.8420	-11.1	0.0000	NA	NA	-1
12	1061	7.60	0.0051	0.0000	7.3950	-8.9	0.0000	NA	NA	-1

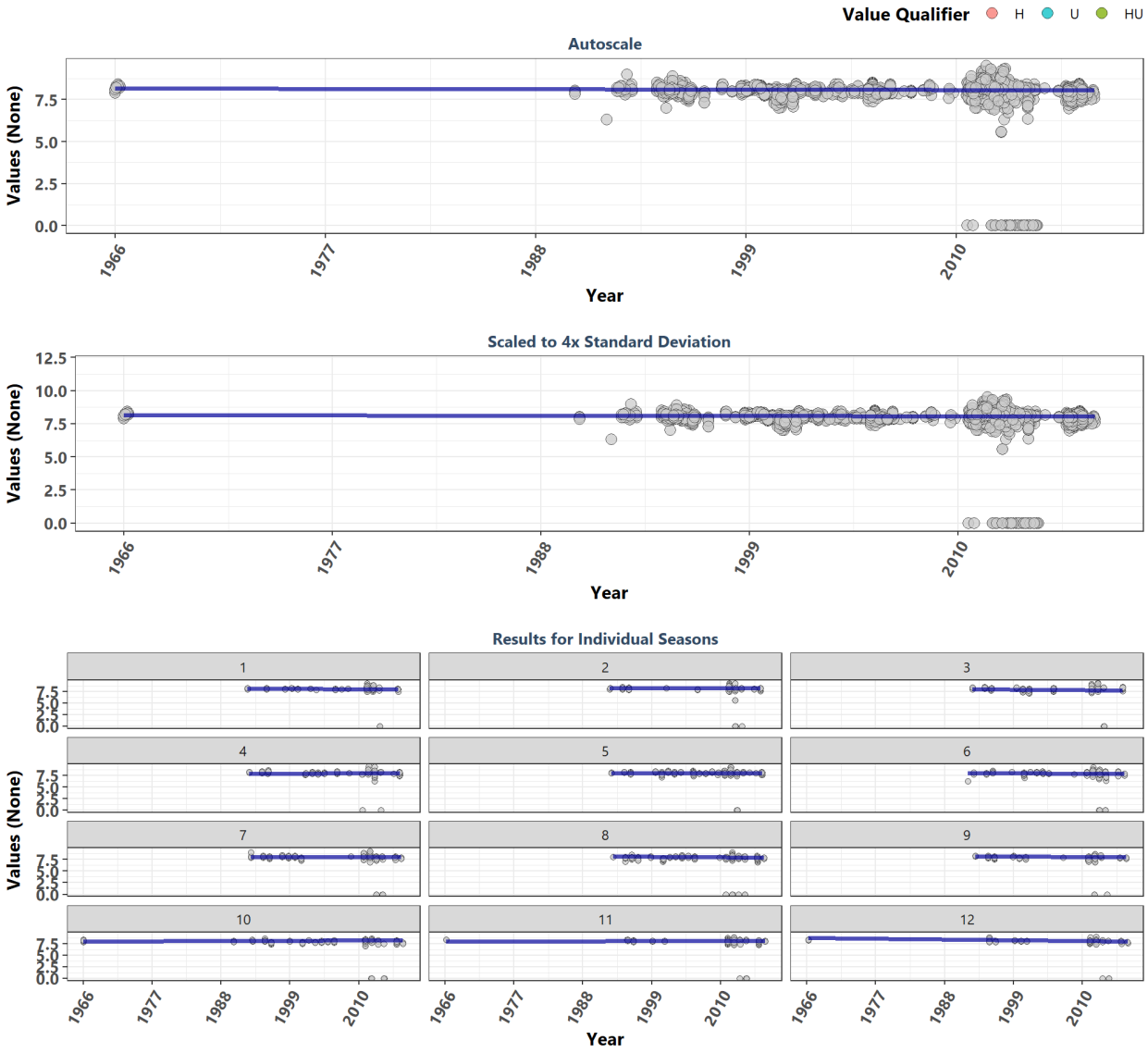
[a] $p < 0.00005$ appear as 0 due to rounding



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	7613	7.90	-0.0880	-0.0030	7.9546	-11.5	0.0000	49.6	0	-1
1	586	7.90	-0.0209	0.0000	7.9000	-3.9	0.0001	NA	NA	-1
2	602	7.90	-0.1888	-0.0125	8.1000	-5.7	0.0000	NA	NA	-1
3	644	7.90	-0.0695	-0.0031	7.9492	-1.4	0.1732	NA	NA	-1
4	628	7.90	-0.1223	-0.0043	7.9686	-4.6	0.0000	NA	NA	-1
5	605	7.90	-0.0657	-0.0006	7.8400	-3.9	0.0001	NA	NA	-1
6	651	7.90	-0.0356	0.0000	7.9000	-2.7	0.0076	NA	NA	-1
7	688	7.90	-0.1540	-0.0072	8.0152	-0.8	0.4080	NA	NA	-1
8	647	7.90	-0.1172	-0.0067	7.9717	-7.2	0.0000	NA	NA	-1
9	632	7.80	-0.1052	-0.0042	7.9667	0.4	0.7147	NA	NA	-1
10	672	7.86	-0.1070	-0.0037	7.9600	-4.6	0.0000	NA	NA	-1
11	629	7.90	0.0097	0.0000	7.8000	-3.3	0.0009	NA	NA	-1
12	629	7.83	-0.0875	-0.0022	7.9356	-2.5	0.0130	NA	NA	-1

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

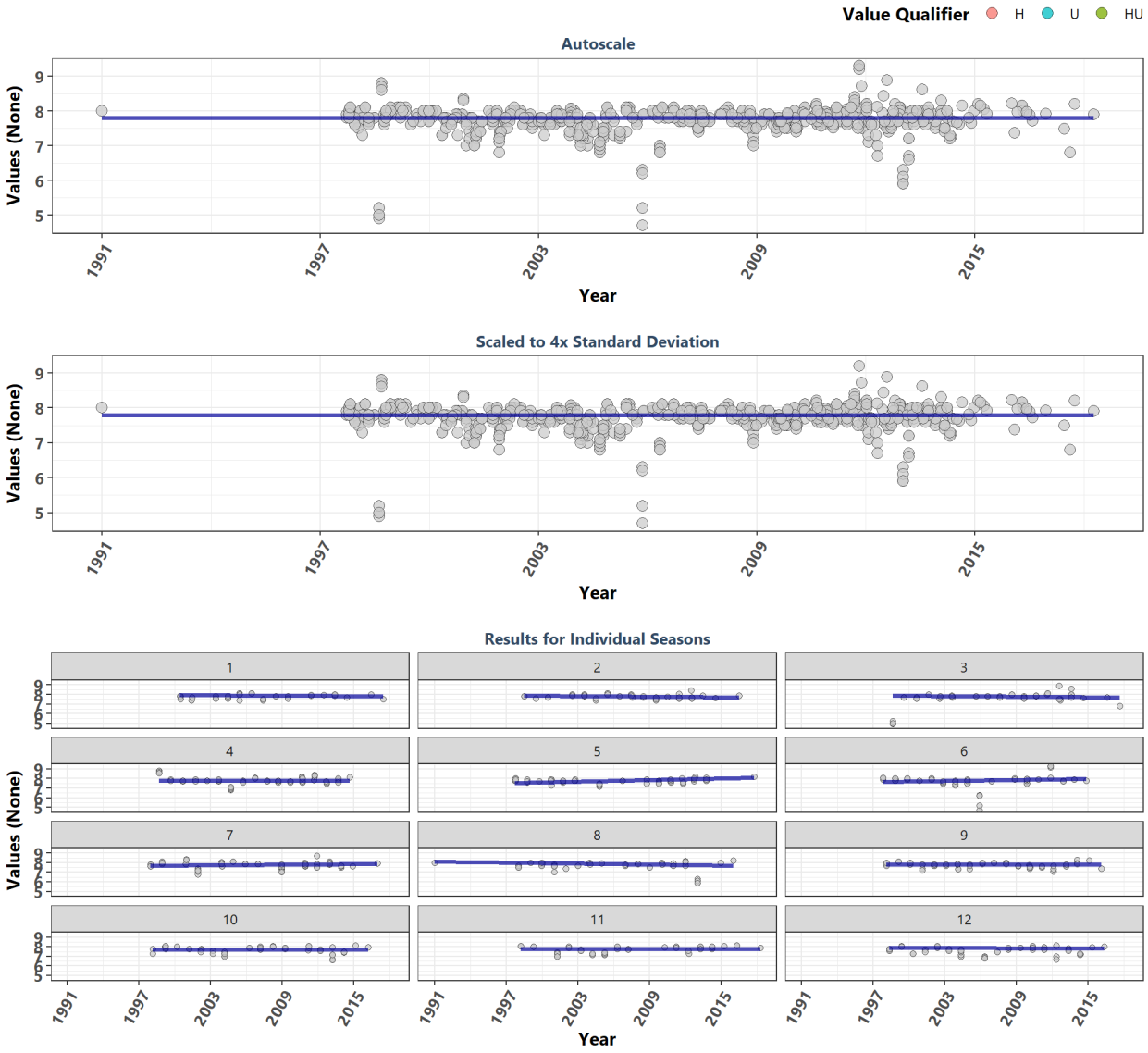
Florida Keys National Marine Sanctuary



Season	N	Median	tau	Slope	Int.	z	p__z	chi__sq	p__chi__sq	Trend
All	3440	8.03	-0.0719	-0.0019	8.1626	-4.8	0.0000	183	0	-1
1	161	8.10	-0.0906	-0.0029	8.0871	-4.5	0.0000	NA	NA	-1
2	281	8.10	-0.0599	-0.0014	8.2257	2.4	0.0150	NA	NA	-1
3	293	8.15	-0.2182	-0.0088	8.2000	3.0	0.0028	NA	NA	-1
4	295	8.16	0.1253	0.0040	7.7980	-1.6	0.1143	NA	NA	1
5	513	8.10	-0.0312	0.0000	8.0000	-5.2	0.0000	NA	NA	-1
6	225	8.00	-0.0966	-0.0048	8.1619	-0.7	0.4808	NA	NA	-1
7	227	8.00	-0.1164	-0.0033	8.1633	-2.2	0.0282	NA	NA	-1
8	474	7.99	-0.2352	-0.0073	8.3473	4.3	0.0000	NA	NA	-1
9	206	7.90	-0.1520	-0.0059	8.3353	-4.7	0.0000	NA	NA	-1
10	337	7.94	0.1160	0.0045	7.9864	-2.7	0.0075	NA	NA	1
11	250	8.00	0.0930	0.0025	7.9825	-10.5	0.0000	NA	NA	1
12	178	8.05	-0.4332	-0.0150	8.6825	-2.3	0.0193	NA	NA	-1

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

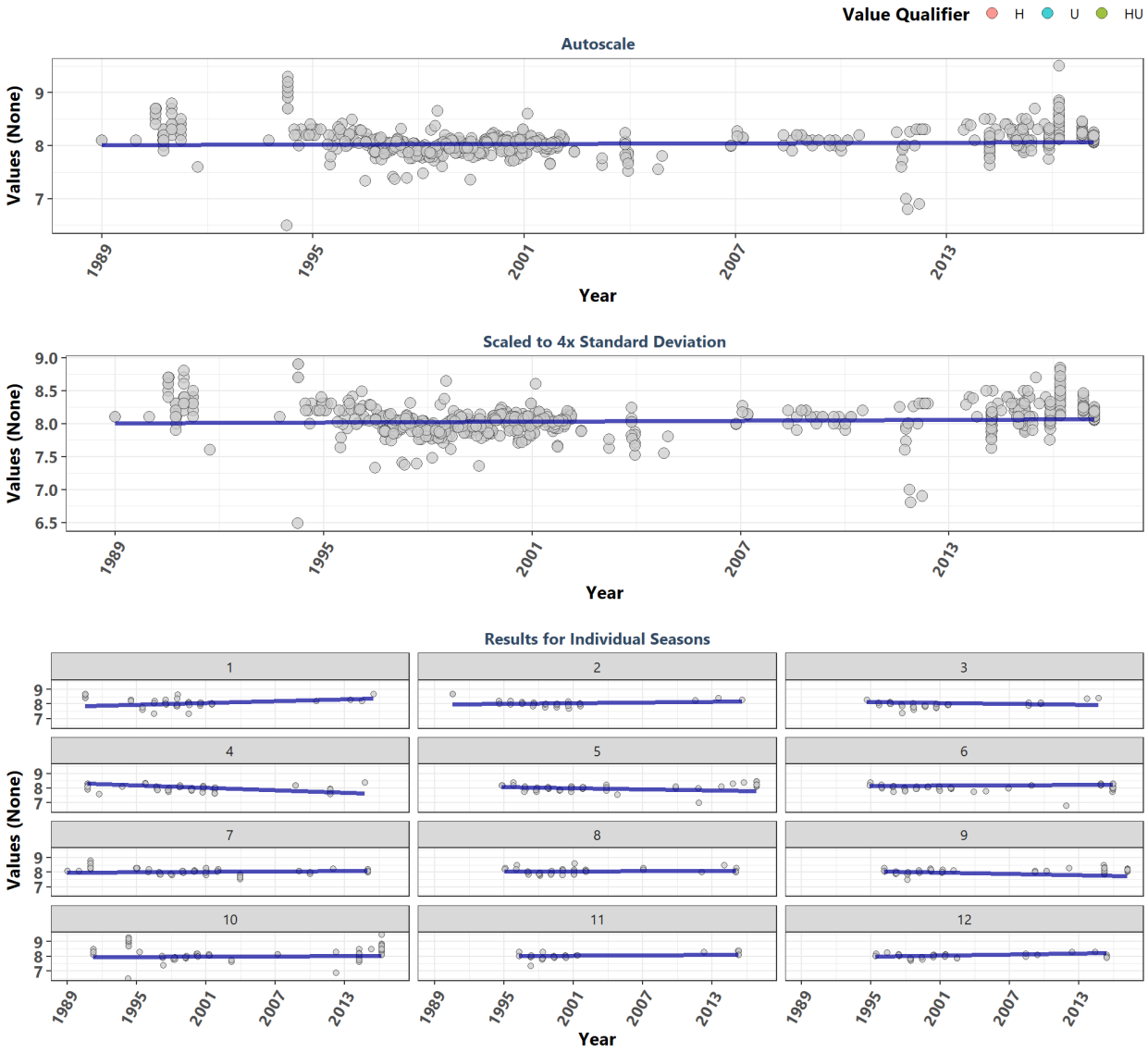
Fort Clinch State Park Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	803	7.8	0.0093	0.0000	7.8000	0.3	0.8008	38	0.0001	0
1	50	7.8	-0.1282	-0.0071	7.9929	2.2	0.0285	NA	NA	0
2	55	7.8	-0.1545	-0.0111	7.9667	-2.5	0.0111	NA	NA	0
3	68	7.8	-0.0796	-0.0068	7.9087	2.4	0.0185	NA	NA	0
4	73	7.8	-0.0209	0.0000	7.8000	-0.3	0.7950	NA	NA	0
5	79	7.8	0.2788	0.0286	7.3429	1.1	0.2684	NA	NA	0
6	73	7.9	0.2106	0.0188	7.5187	-1.6	0.1039	NA	NA	0
7	75	7.9	0.1914	0.0111	7.6000	-1.0	0.3310	NA	NA	0
8	58	7.8	-0.2323	-0.0167	8.0833	0.8	0.3968	NA	NA	0
9	70	7.8	0.0750	0.0000	7.8000	-1.9	0.0557	NA	NA	0
10	67	7.8	0.0097	0.0000	7.7000	-1.0	0.3387	NA	NA	0
11	64	7.9	0.0831	0.0000	7.8000	3.3	0.0009	NA	NA	0
12	71	7.7	-0.0760	-0.0033	7.9467	0.1	0.9081	NA	NA	0

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

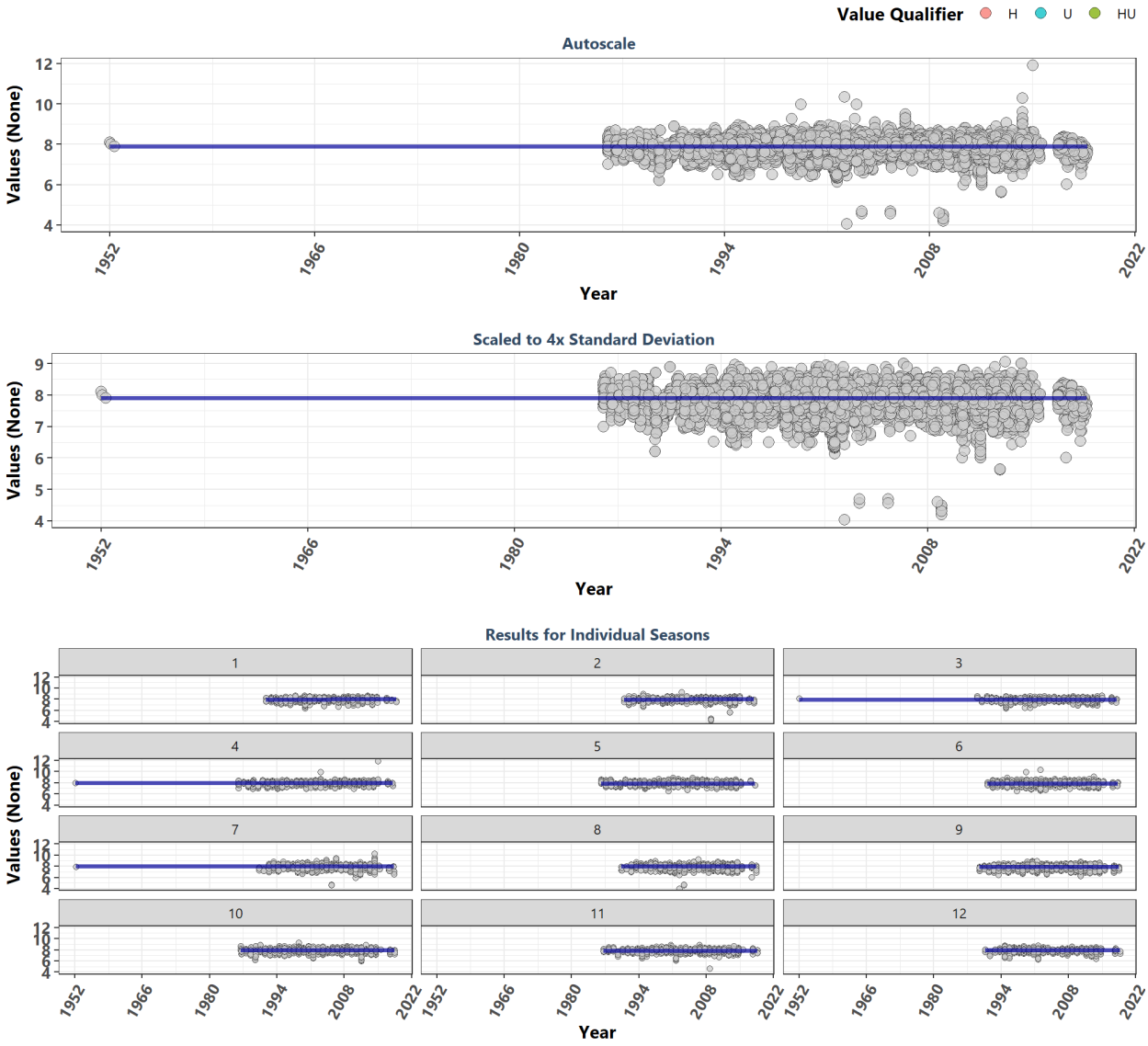
Fort Pickens State Park Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	668	8.08	0.0450	0.0020	8.0083	1.8	0.0752	64.2	0	0
1	42	8.09	0.4163	0.0213	7.8267	-1.9	0.0570	NA	NA	0
2	34	8.01	0.1317	0.0065	7.9888	-1.1	0.2617	NA	NA	0
3	37	7.97	-0.2428	-0.0100	8.2100	0.4	0.6821	NA	NA	0
4	59	8.04	-0.2033	-0.0288	8.3775	-3.0	0.0024	NA	NA	0
5	69	8.10	-0.2671	-0.0133	8.1600	4.6	0.0000	NA	NA	0
6	68	8.07	0.0406	0.0029	8.1388	1.1	0.2609	NA	NA	0
7	68	8.10	0.2150	0.0050	7.9800	-3.0	0.0032	NA	NA	0
8	41	8.06	0.0913	0.0025	8.0225	1.2	0.2254	NA	NA	0
9	77	8.11	-0.1355	-0.0150	8.1600	2.9	0.0041	NA	NA	0
10	109	8.18	0.0480	0.0025	7.9425	0.6	0.5268	NA	NA	0
11	29	8.04	0.0958	0.0042	7.9942	3.2	0.0014	NA	NA	0
12	35	8.04	0.3721	0.0126	7.9115	0.8	0.4226	NA	NA	0

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

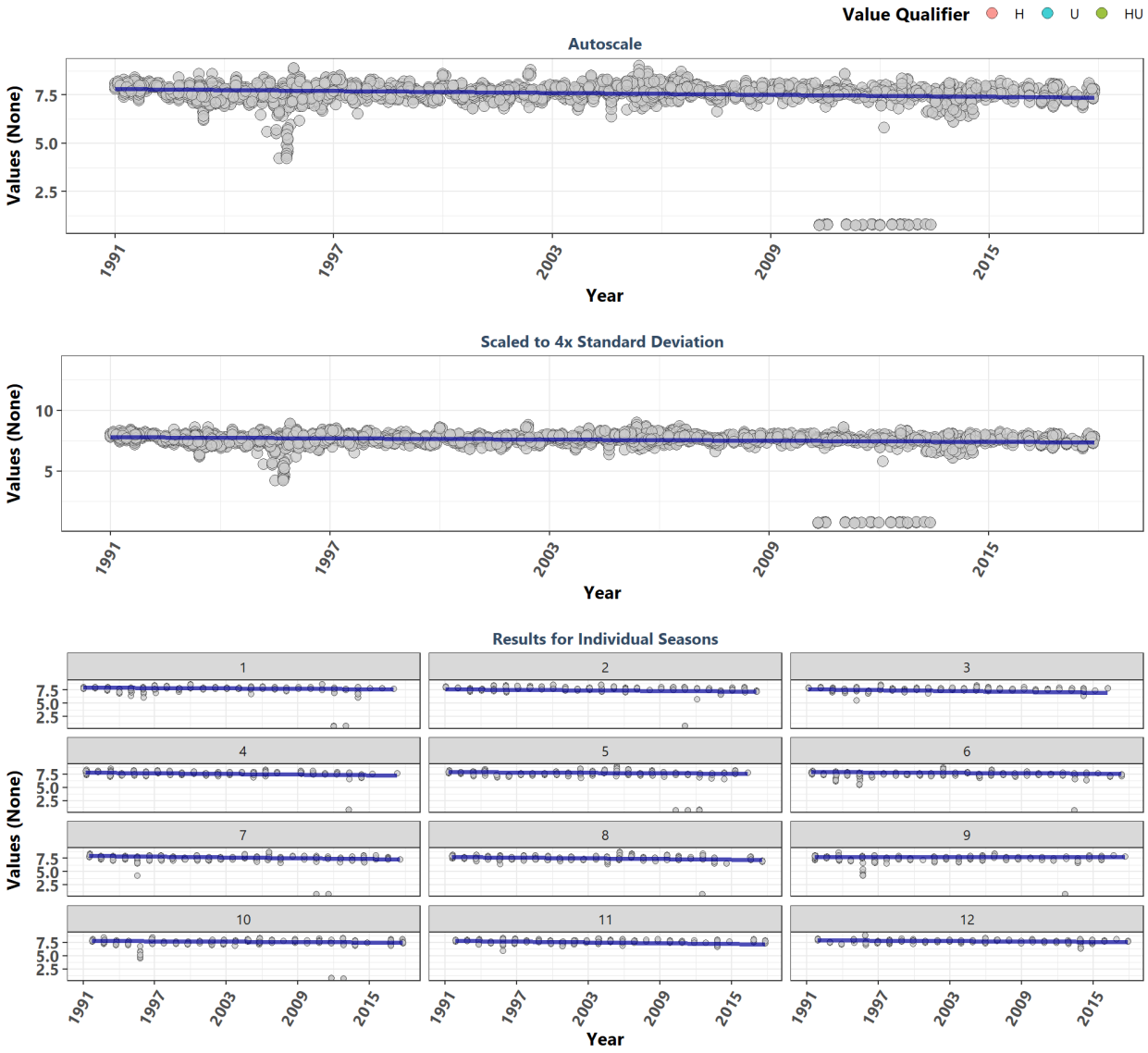
Gasparilla Sound-Charlotte Harbor Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	34852	7.90	0.0263	0.0000	7.9000	7.5	0.0000	133.3	0	-1
1	2490	7.90	0.1008	0.0039	7.7717	2.8	0.0055	NA	NA	1
2	2594	7.97	0.1029	0.0040	7.7280	-1.2	0.2168	NA	NA	1
3	2570	7.94	-0.0201	0.0000	7.9000	7.9	0.0000	NA	NA	-1
4	3173	7.97	0.0131	0.0000	7.9600	8.6	0.0000	NA	NA	-1
5	3081	8.00	0.0376	0.0000	7.9000	1.3	0.2105	NA	NA	-1
6	2934	7.96	0.0369	0.0000	7.9000	4.5	0.0000	NA	NA	-1
7	2938	7.96	-0.0161	0.0000	7.9700	1.1	0.2835	NA	NA	-1
8	2755	7.90	0.0150	0.0000	8.0000	1.5	0.1286	NA	NA	-1
9	2965	7.80	-0.0116	0.0000	7.9000	-1.6	0.0996	NA	NA	-1
10	3577	7.90	0.0553	0.0014	7.8907	3.4	0.0007	NA	NA	1
11	3049	7.90	-0.0201	0.0000	7.8000	-1.7	0.0951	NA	NA	-1
12	2726	7.90	0.0193	0.0000	7.9000	-0.9	0.3632	NA	NA	-1

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

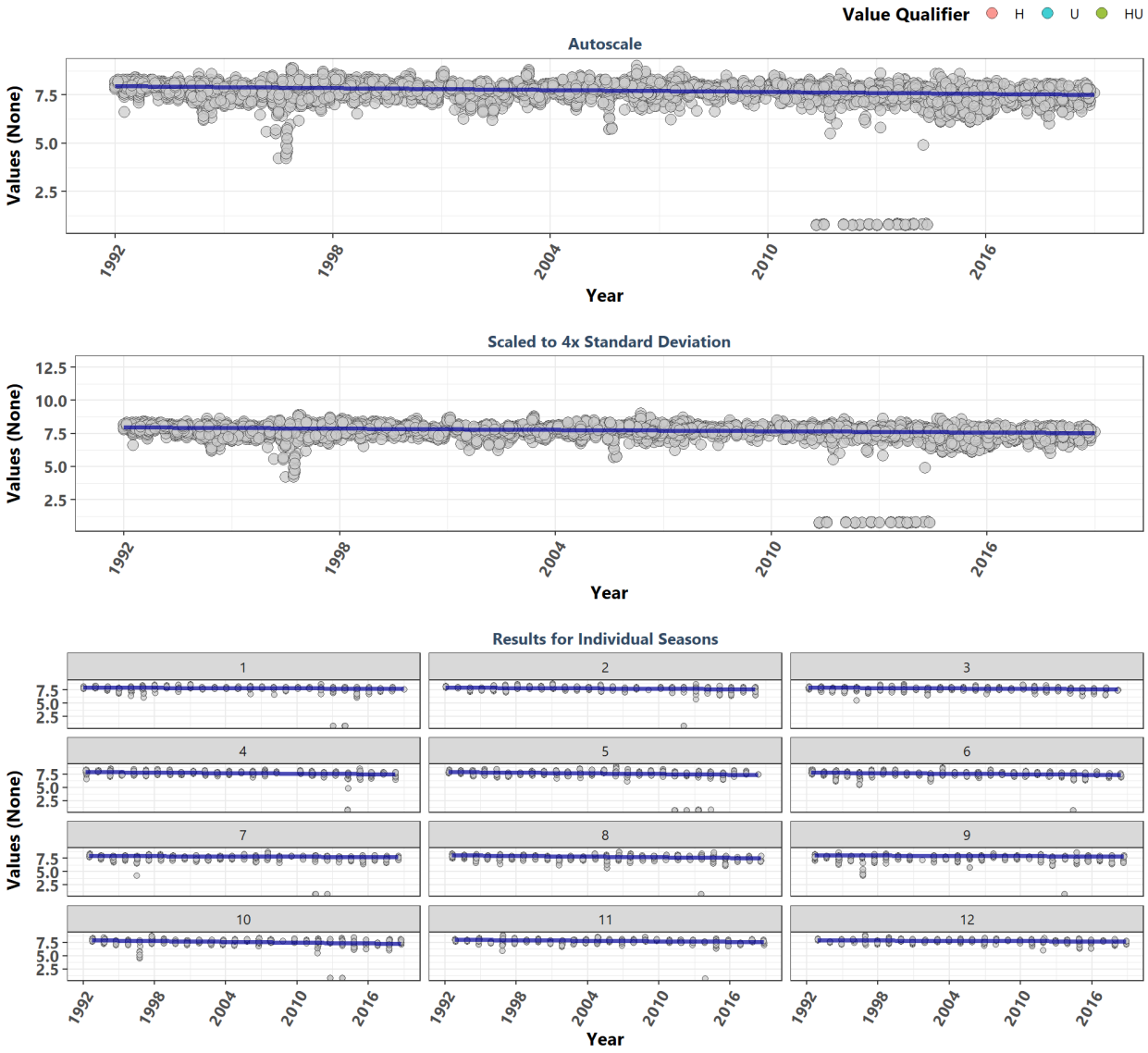
Guana River Marsh Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p__z	chi__sq	p__chi__sq	Trend
All	5105	7.70	-0.2113	-0.0167	7.8073	-22.8	0.0000	96.9	0	-1
1	483	7.87	-0.1631	-0.0143	7.9843	-5.4	0.0000	NA	NA	-1
2	390	7.80	-0.2614	-0.0200	7.6800	-4.2	0.0000	NA	NA	-1
3	332	7.72	-0.3621	-0.0250	7.6500	-4.7	0.0000	NA	NA	-1
4	526	7.80	-0.2393	-0.0210	7.8095	-5.6	0.0000	NA	NA	-1
5	519	7.70	-0.1629	-0.0091	7.8636	-8.5	0.0000	NA	NA	-1
6	368	7.55	-0.1402	-0.0136	7.8955	-10.4	0.0000	NA	NA	-1
7	543	7.60	-0.2487	-0.0250	7.9750	-9.4	0.0000	NA	NA	-1
8	494	7.60	-0.2418	-0.0192	7.7154	-8.1	0.0000	NA	NA	-1
9	341	7.60	0.0305	0.0007	7.7336	-7.3	0.0000	NA	NA	1
10	502	7.60	-0.1717	-0.0133	7.8050	-8.0	0.0000	NA	NA	-1
11	358	7.80	-0.2696	-0.0250	7.8000	-5.8	0.0000	NA	NA	-1
12	249	7.74	-0.2031	-0.0111	7.8889	0.7	0.4718	NA	NA	-1

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

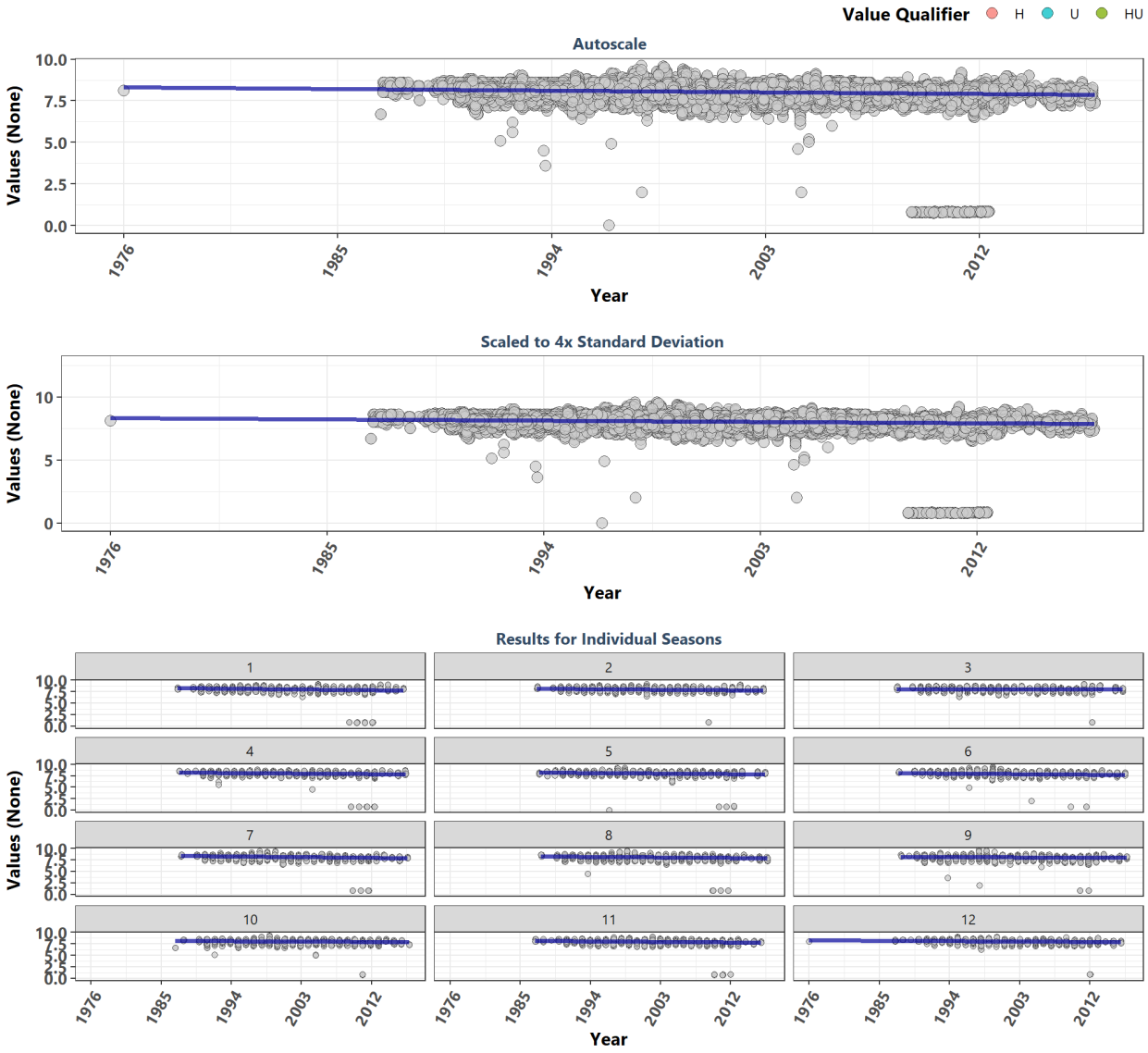
Guana Tolomato Matanzas National Estuarine Research Reserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	14410	7.80	-0.2192	-0.0160	7.9560	-39.8	0	198	0	-1
1	1313	7.90	-0.1761	-0.0111	8.0000	-16.9	0	NA	NA	-1
2	1046	7.99	-0.2432	-0.0143	8.0143	-5.5	0	NA	NA	-1
3	1224	7.90	-0.2437	-0.0157	8.0186	-5.9	0	NA	NA	-1
4	1201	7.90	-0.2382	-0.0176	7.9412	-12.7	0	NA	NA	-1
5	1363	7.83	-0.2385	-0.0200	7.9400	-13.5	0	NA	NA	-1
6	1100	7.70	-0.2518	-0.0167	7.8300	-16.5	0	NA	NA	-1
7	1534	7.80	-0.1121	-0.0071	7.9429	-14.0	0	NA	NA	-1
8	1487	7.80	-0.3100	-0.0231	8.1077	-13.8	0	NA	NA	-1
9	1082	7.73	-0.1136	-0.0091	8.0627	-12.5	0	NA	NA	-1
10	1349	7.80	-0.3299	-0.0262	7.8838	-10.7	0	NA	NA	-1
11	980	7.90	-0.1937	-0.0154	7.9692	-8.3	0	NA	NA	-1
12	731	7.82	-0.1115	-0.0075	7.8800	-4.5	0	NA	NA	-1

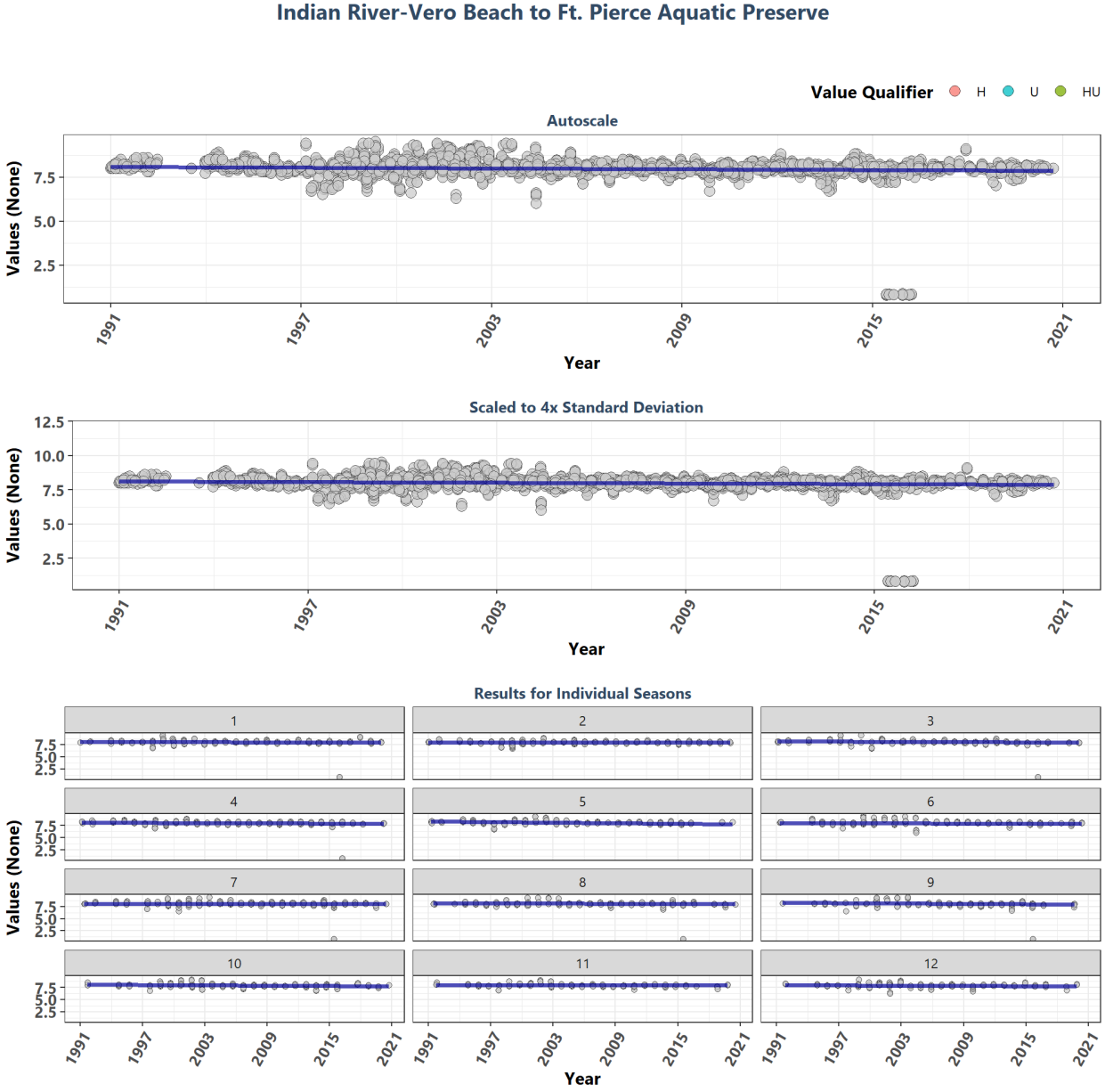
[a] $p < 0.00005$ appear as 0 due to rounding

Indian River-Malabar to Vero Beach Aquatic Preserve



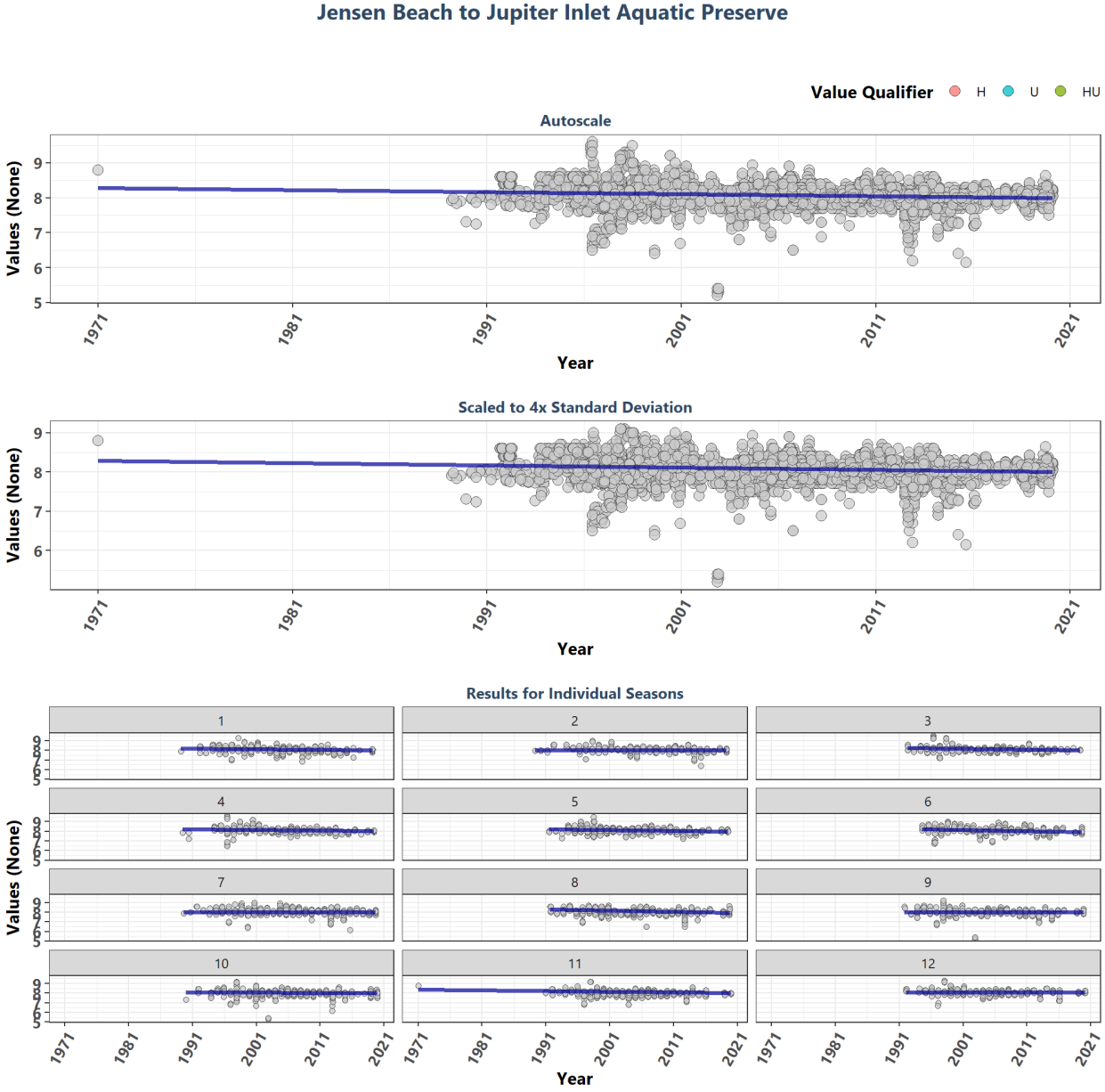
Season	N	Median	tau	Slope	Int.	z	p__z	chi__sq	p__chi__sq	Trend
All	38291	8.00	-0.1600	-0.0114	8.3300	-47.2	0	197.2	0	-1
1	3236	8.05	-0.1865	-0.0154	8.4154	-15.5	0	NA	NA	-1
2	2926	8.04	-0.1557	-0.0125	8.3250	-6.8	0	NA	NA	-1
3	2911	8.00	-0.0655	0.0000	8.0000	-5.3	0	NA	NA	-1
4	3529	8.00	-0.1479	-0.0095	8.2571	-13.3	0	NA	NA	-1
5	3119	8.03	-0.1768	-0.0125	8.3350	-15.9	0	NA	NA	-1
6	3151	8.01	-0.1425	-0.0111	8.2000	-14.9	0	NA	NA	-1
7	3514	8.00	-0.2336	-0.0167	8.5333	-16.6	0	NA	NA	-1
8	3314	8.10	-0.1811	-0.0133	8.4100	-20.3	0	NA	NA	-1
9	3360	8.00	-0.0837	-0.0050	8.1700	-13.6	0	NA	NA	-1
10	3566	8.00	-0.1894	-0.0125	8.3675	-11.9	0	NA	NA	-1
11	2953	7.90	-0.2169	-0.0143	8.3714	-11.7	0	NA	NA	-1
12	2712	8.00	-0.1326	-0.0100	8.2600	-17.0	0	NA	NA	-1

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



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	5165	8.00	-0.1491	-0.0077	8.0987	-15.9	0.0000	75	0	-1
1	509	8.04	-0.1359	-0.0071	8.1043	-6.6	0.0000	NA	NA	-1
2	440	7.99	-0.0583	-0.0009	8.0018	-1.8	0.0661	NA	NA	-1
3	382	8.10	-0.1099	-0.0050	8.1550	-10.0	0.0000	NA	NA	-1
4	562	8.10	-0.1940	-0.0100	8.1700	-7.4	0.0000	NA	NA	-1
5	418	8.10	-0.3398	-0.0218	8.3618	-3.4	0.0007	NA	NA	-1
6	332	8.07	-0.1191	-0.0045	8.0405	-2.1	0.0363	NA	NA	-1
7	604	8.10	-0.0765	-0.0023	8.0931	-4.5	0.0000	NA	NA	-1
8	413	8.00	-0.1216	-0.0070	8.2050	-3.6	0.0003	NA	NA	-1
9	373	7.99	-0.2061	-0.0111	8.2444	-3.9	0.0001	NA	NA	-1
10	471	7.90	-0.2529	-0.0125	8.0925	-4.7	0.0000	NA	NA	-1
11	302	7.96	-0.0309	0.0000	8.0000	-6.6	0.0000	NA	NA	-1
12	359	8.00	-0.1445	-0.0071	8.0000	-0.9	0.3764	NA	NA	-1

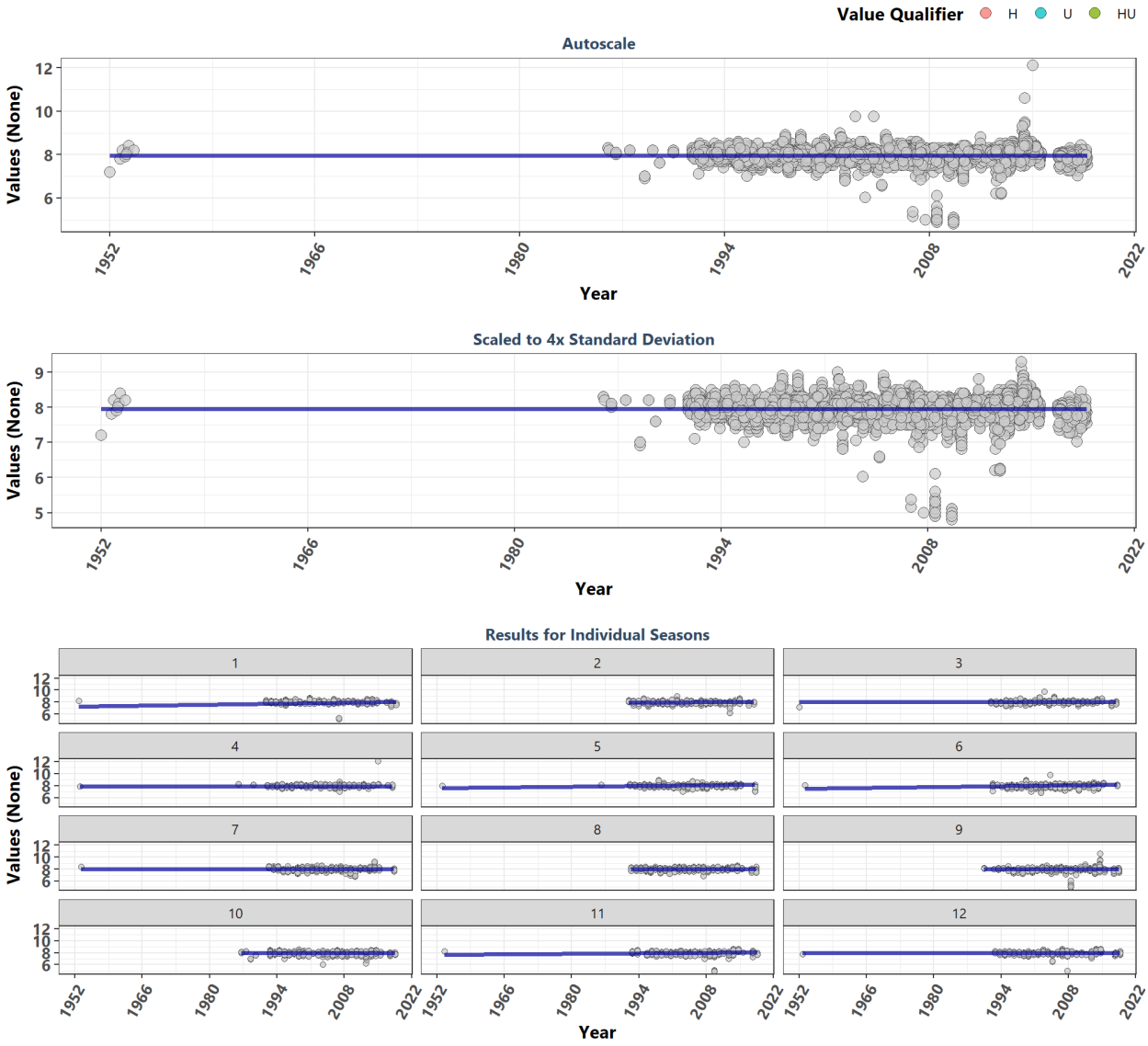
[a] $p < 0.00005$ appear as 0 due to rounding



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	6465	8.10	-0.1251	-0.0059	8.2900	-15.1	0.0000	40.6	0	-1
1	525	8.10	-0.1273	-0.0067	8.3050	-5.7	0.0000	NA	NA	-1
2	544	8.10	-0.0637	0.0000	8.0000	-5.9	0.0000	NA	NA	-1
3	502	8.10	-0.1928	-0.0091	8.4273	-5.1	0.0000	NA	NA	-1
4	524	8.10	-0.1500	-0.0050	8.2750	-6.7	0.0000	NA	NA	-1
5	535	8.08	-0.1640	-0.0083	8.4000	-6.7	0.0000	NA	NA	-1
6	532	8.10	-0.1933	-0.0110	8.4650	-6.0	0.0000	NA	NA	-1
7	646	8.07	-0.0401	0.0000	8.0000	-4.9	0.0000	NA	NA	-1
8	606	8.04	-0.1743	-0.0111	8.5000	-2.7	0.0079	NA	NA	-1
9	536	8.00	-0.0689	0.0000	8.0000	-2.4	0.0164	NA	NA	-1
10	538	8.00	-0.0718	-0.0016	8.1003	-1.4	0.1608	NA	NA	-1
11	505	8.00	-0.1670	-0.0077	8.3846	-2.2	0.0309	NA	NA	-1
12	472	8.10	-0.0914	0.0000	8.1000	-3.0	0.0027	NA	NA	-1

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

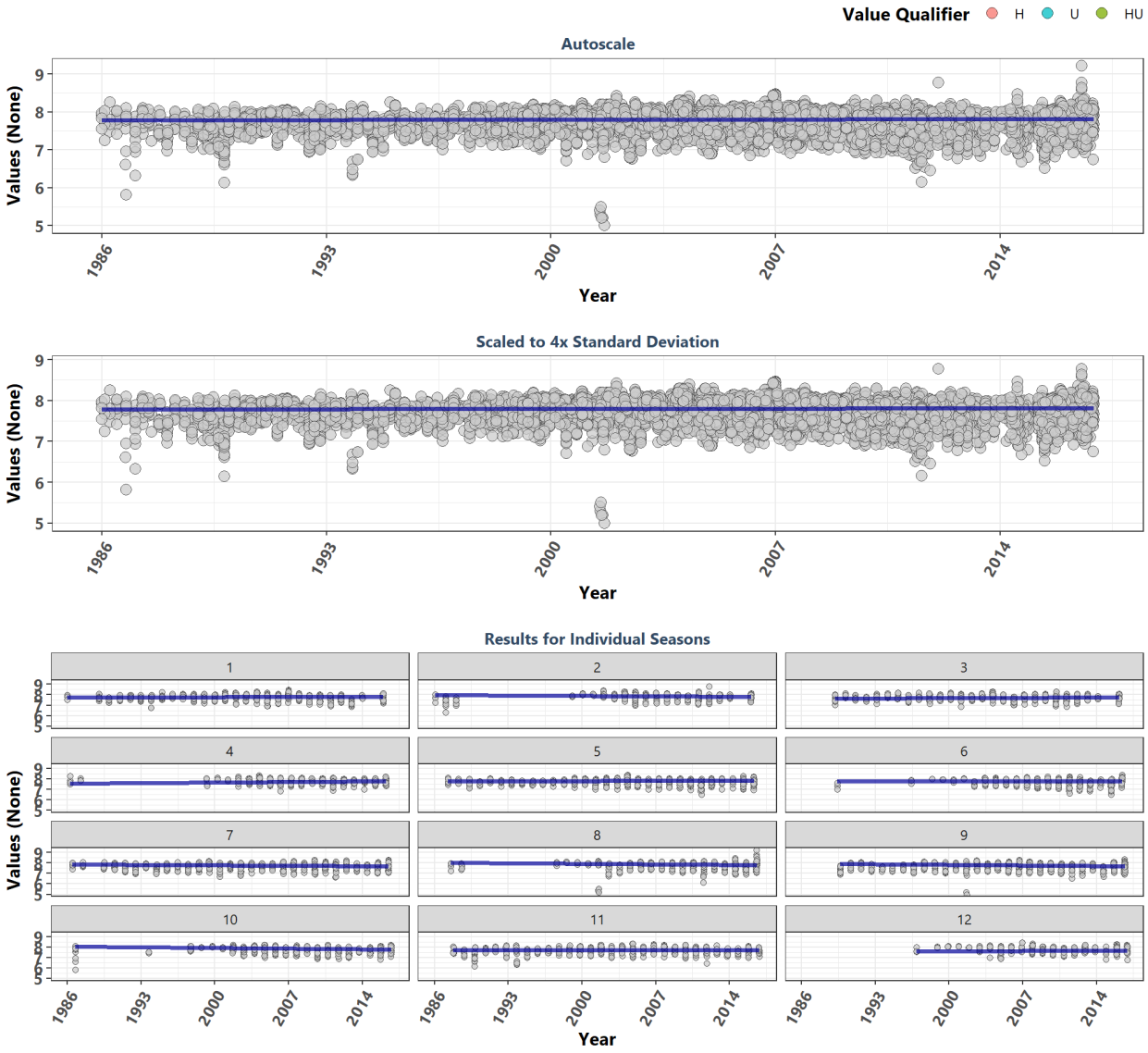
Lemon Bay Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	7485	8.00	0.0773	0.0000	7.9500	10.2	0.0000	89.4	0	-1
1	444	8.00	0.1822	0.0111	7.2889	0.3	0.7709	NA	NA	1
2	567	7.90	0.1035	0.0048	7.7324	-1.4	0.1486	NA	NA	1
3	513	8.00	-0.0258	0.0000	8.0000	1.7	0.0953	NA	NA	-1
4	567	8.00	-0.0403	0.0000	7.9000	2.9	0.0037	NA	NA	-1
5	600	8.00	0.1710	0.0091	7.6091	-1.0	0.3397	NA	NA	1
6	692	7.98	0.1393	0.0100	7.4600	4.1	0.0000	NA	NA	1
7	788	8.00	0.0489	0.0000	8.0000	2.3	0.0229	NA	NA	-1
8	701	8.10	0.0092	0.0000	8.0000	6.8	0.0000	NA	NA	-1
9	598	8.00	0.0537	0.0000	8.0000	5.1	0.0000	NA	NA	-1
10	775	8.00	0.0804	0.0000	8.0000	1.2	0.2396	NA	NA	-1
11	647	7.90	0.1436	0.0067	7.6400	7.0	0.0000	NA	NA	1
12	593	8.00	0.0280	0.0000	8.0000	5.3	0.0000	NA	NA	-1

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

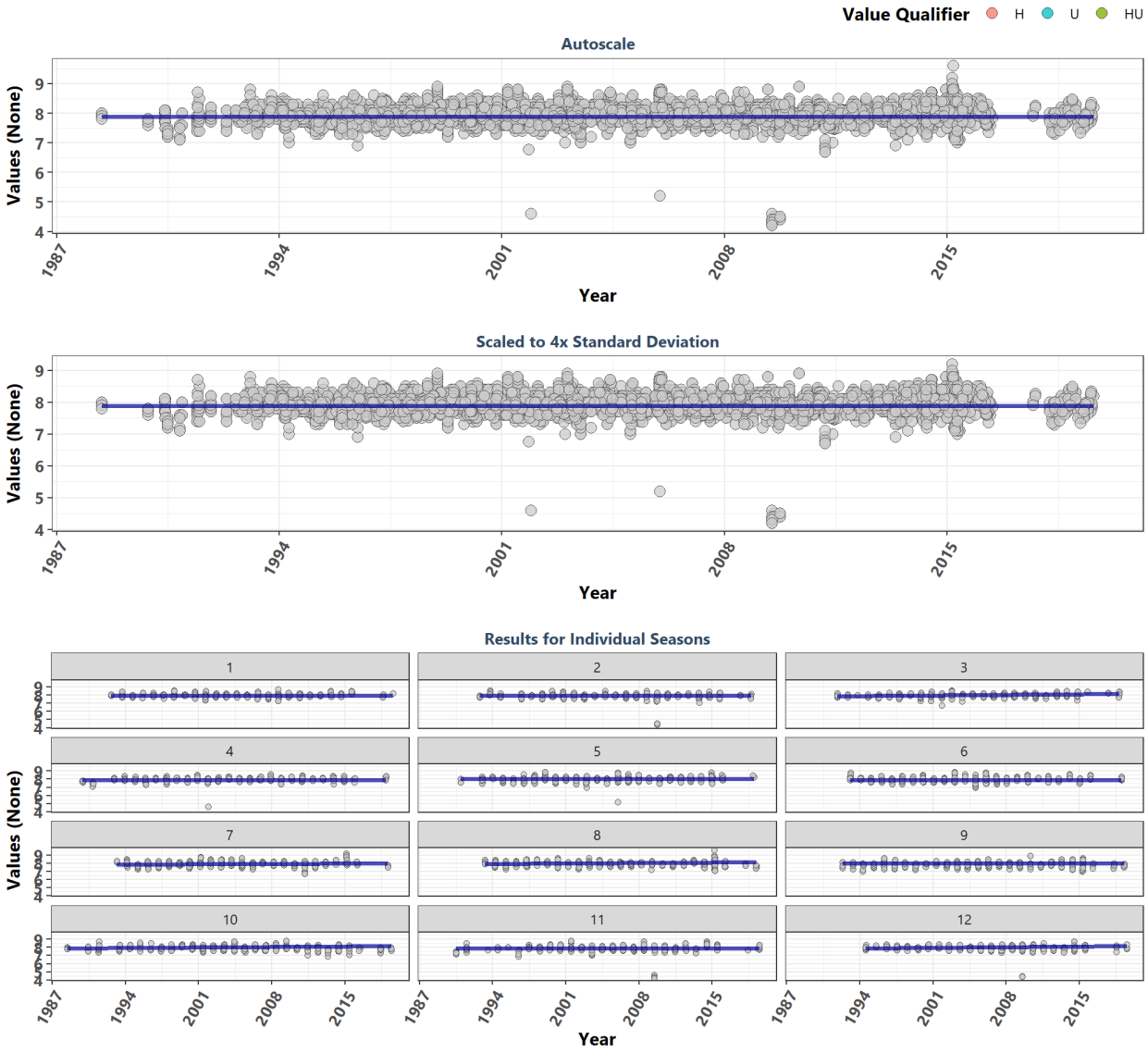
Loxahatchee River-Lake Worth Creek Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	7384	7.75	0.0025	0.0010	7.7865	2.2	0.0267	100.4	0	1
1	763	7.80	0.0164	0.0007	7.7671	0.6	0.5304	NA	NA	1
2	426	7.84	-0.0851	-0.0052	7.9757	-3.8	0.0002	NA	NA	-1
3	731	7.78	0.0611	0.0039	7.6096	0.7	0.5056	NA	NA	1
4	504	7.85	0.1348	0.0076	7.5452	-2.9	0.0042	NA	NA	1
5	791	7.72	0.0152	0.0007	7.7880	0.8	0.4514	NA	NA	1
6	472	7.74	0.0000	0.0000	7.7850	-3.2	0.0016	NA	NA	-1
7	796	7.69	-0.0814	-0.0064	7.8664	5.7	0.0000	NA	NA	-1
8	494	7.78	-0.1104	-0.0083	8.0417	0.0	0.9998	NA	NA	-1
9	759	7.61	-0.0969	-0.0088	7.9500	3.1	0.0020	NA	NA	-1
10	490	7.72	-0.1221	-0.0100	8.0600	-2.7	0.0069	NA	NA	-1
11	729	7.68	0.0179	0.0008	7.7042	2.5	0.0135	NA	NA	1
12	429	7.85	0.0750	0.0050	7.5150	-3.4	0.0006	NA	NA	1

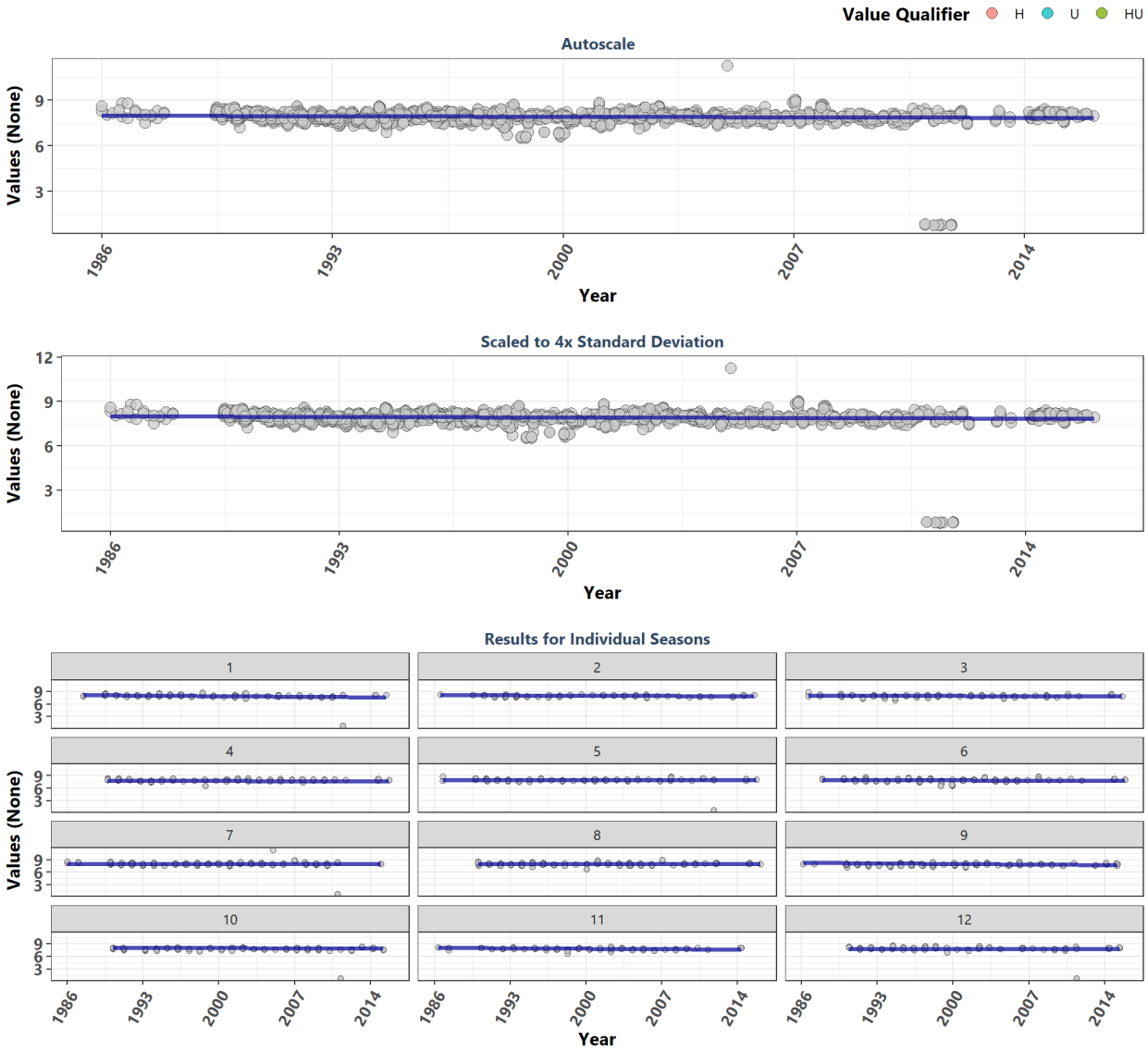
[a] $p < 0.00005$ appear as 0 due to rounding

Matlacha Pass Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	6427	7.93	0.0685	0.0000	7.8886	8.3	0.0000	129.2	0	-1
1	450	8.00	-0.0849	0.0000	7.9000	1.4	0.1607	NA	NA	-1
2	515	7.90	-0.0768	0.0000	7.9000	-1.4	0.1762	NA	NA	-1
3	513	8.00	0.1389	0.0091	7.8545	5.8	0.0000	NA	NA	1
4	543	8.00	0.0666	0.0000	7.9000	7.8	0.0000	NA	NA	-1
5	580	8.00	0.0438	0.0000	8.0000	5.3	0.0000	NA	NA	-1
6	547	8.00	0.0876	0.0014	7.8771	4.9	0.0000	NA	NA	1
7	585	7.90	0.1077	0.0050	7.8100	3.2	0.0014	NA	NA	1
8	551	7.99	0.1680	0.0071	7.8714	1.1	0.2682	NA	NA	1
9	509	7.90	0.0313	0.0000	7.9900	2.3	0.0239	NA	NA	-1
10	641	7.90	0.1454	0.0083	7.8583	-2.9	0.0034	NA	NA	1
11	544	7.90	-0.0394	0.0000	7.9000	3.8	0.0002	NA	NA	-1
12	449	7.90	0.2228	0.0100	7.8300	-2.7	0.0066	NA	NA	1

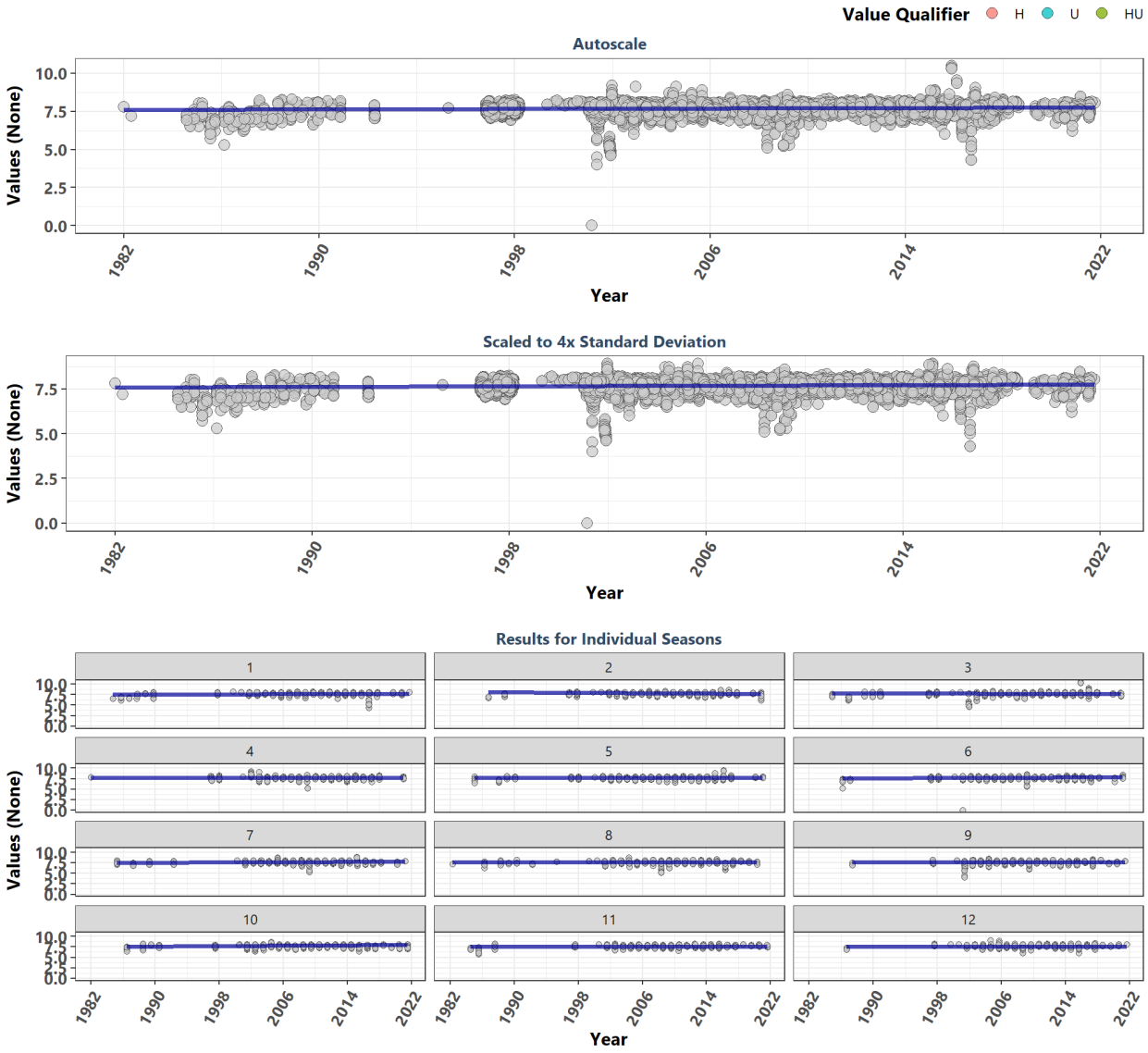
[a] $p < 0.00005$ appear as 0 due to rounding



Season	N	Median	tau	Slope	Int.	z	p__z	chi__sq	p__chi__sq	Trend
All	3497	7.90	-0.1199	-0.0059	7.9997	-10.0	0.0000	115.1	0	-1
1	329	8.00	-0.2806	-0.0200	8.2200	-8.5	0.0000	NA	NA	-1
2	209	8.00	-0.2183	-0.0100	8.1300	-1.3	0.1977	NA	NA	-1
3	256	8.00	-0.1032	-0.0038	7.9519	-3.6	0.0004	NA	NA	-1
4	288	7.90	-0.0998	-0.0055	7.7600	-2.6	0.0084	NA	NA	-1
5	245	7.90	-0.0634	0.0000	7.9000	-2.6	0.0088	NA	NA	-1
6	341	7.90	-0.1112	-0.0062	7.9738	-1.8	0.0788	NA	NA	-1
7	345	8.00	-0.0592	0.0000	8.0000	-6.1	0.0000	NA	NA	-1
8	354	8.00	0.0774	0.0000	8.0000	2.2	0.0283	NA	NA	-1
9	345	7.80	-0.3128	-0.0200	8.2400	1.6	0.1101	NA	NA	-1
10	305	7.70	-0.1473	-0.0071	8.0714	-2.6	0.0087	NA	NA	-1
11	213	7.80	-0.2547	-0.0153	7.9993	-5.6	0.0000	NA	NA	-1
12	267	7.98	0.0571	0.0000	7.8000	-6.9	0.0000	NA	NA	-1

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

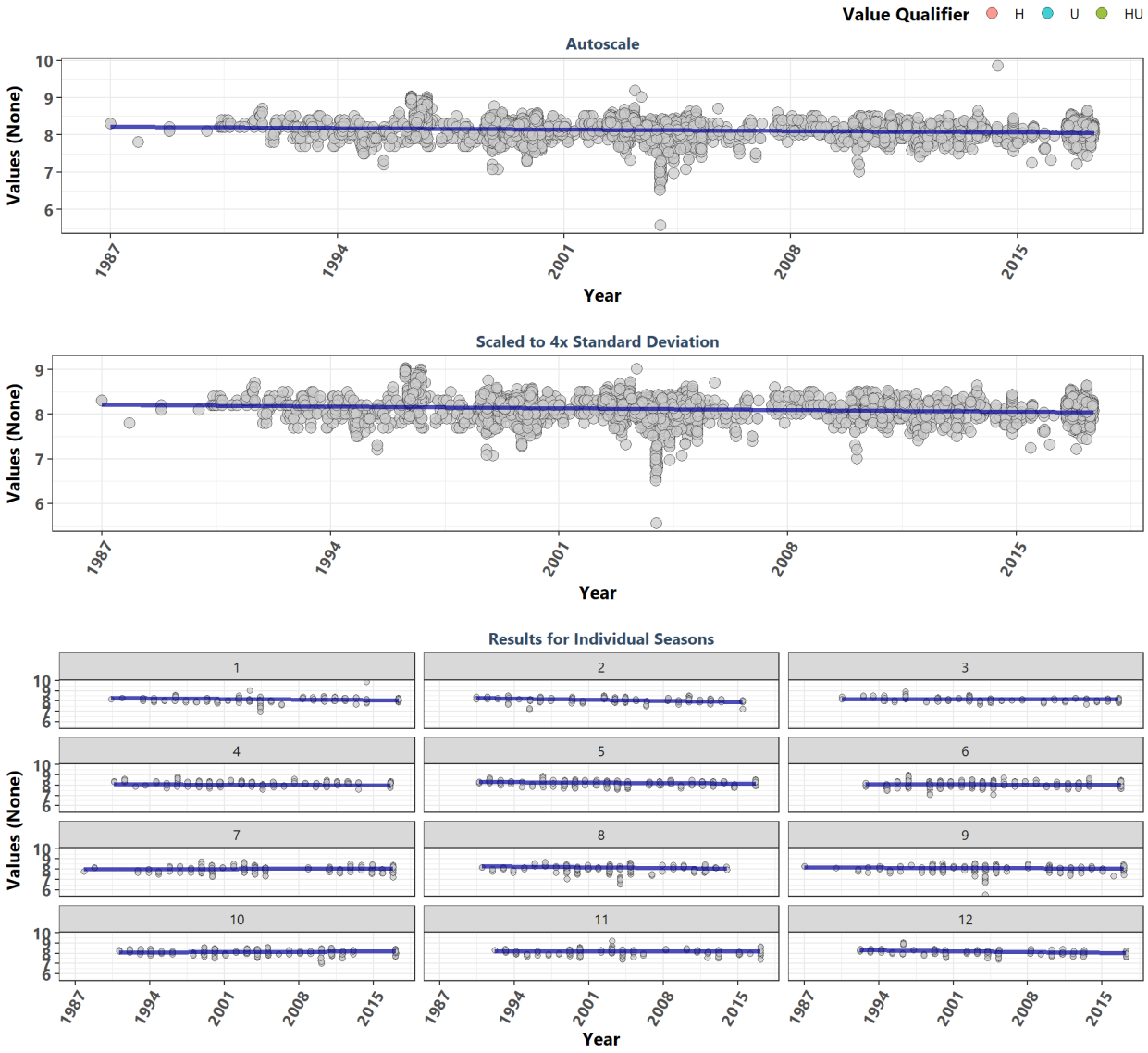
Nassau River-St. Johns River Marshes Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	17457	7.66	0.0472	0.0038	7.6000	14.0	0.0000	541.2	0	1
1	1137	7.72	0.0683	0.0036	7.4955	-1.2	0.2289	NA	NA	1
2	1647	7.71	-0.2172	-0.0125	8.0875	1.9	0.0634	NA	NA	-1
3	1971	7.64	-0.1035	-0.0058	7.8625	9.8	0.0000	NA	NA	-1
4	2120	7.61	0.0297	0.0007	7.6993	8.5	0.0000	NA	NA	1
5	1223	7.60	-0.0234	0.0000	7.7200	2.1	0.0324	NA	NA	-1
6	2357	7.66	0.1432	0.0094	7.4894	15.0	0.0000	NA	NA	1
7	873	7.60	0.1178	0.0075	7.4900	-0.8	0.4285	NA	NA	1
8	775	7.50	0.0403	0.0000	7.6000	1.5	0.1392	NA	NA	-1
9	794	7.60	0.0304	0.0000	7.6000	1.3	0.1972	NA	NA	-1
10	1859	7.55	0.1965	0.0153	7.4147	4.9	0.0000	NA	NA	1
11	1602	7.78	-0.0178	0.0000	7.6000	-6.6	0.0000	NA	NA	-1
12	1099	7.80	0.0353	0.0000	7.5000	-11.0	0.0000	NA	NA	-1

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

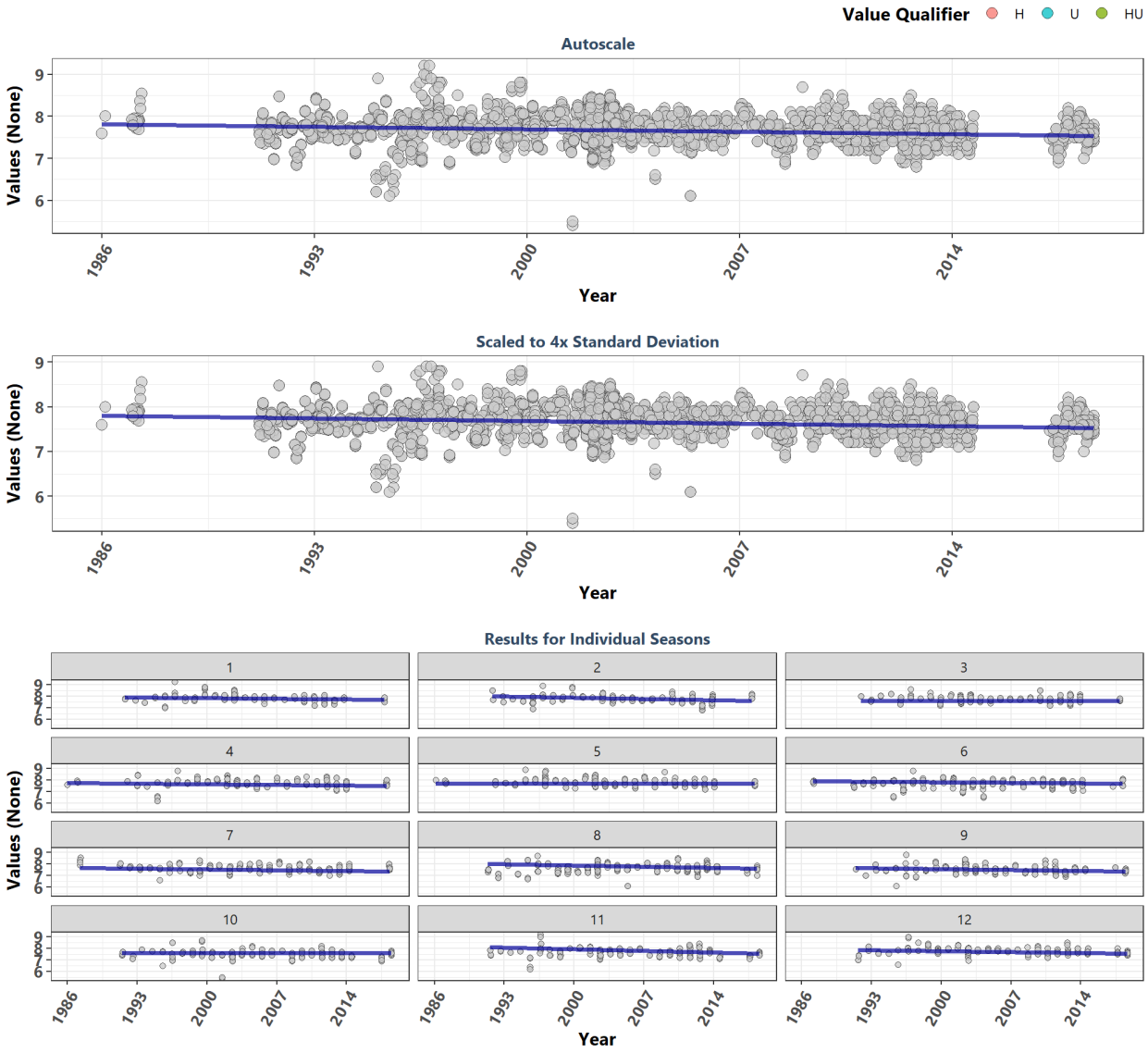
Nature Coast Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	4081	8.15	-0.1109	-0.0056	8.2156	-10.5	0.0000	137.3	0	-1
1	334	8.12	-0.2236	-0.0092	8.3569	-2.3	0.0198	NA	NA	-1
2	291	8.19	-0.3571	-0.0175	8.3800	-5.8	0.0000	NA	NA	-1
3	310	8.17	-0.0294	-0.0012	8.2212	-7.3	0.0000	NA	NA	-1
4	400	8.20	-0.0393	-0.0036	8.1182	-6.7	0.0000	NA	NA	-1
5	373	8.23	-0.1166	-0.0065	8.3340	-3.4	0.0007	NA	NA	-1
6	364	8.13	-0.0374	-0.0018	8.1209	1.3	0.2069	NA	NA	-1
7	347	8.05	0.0208	0.0008	8.0358	0.6	0.5606	NA	NA	1
8	269	8.06	-0.2762	-0.0100	8.3400	-1.0	0.3335	NA	NA	-1
9	348	8.09	-0.0849	-0.0030	8.1717	-1.0	0.2957	NA	NA	-1
10	380	8.20	0.0441	0.0031	8.0808	-0.7	0.5032	NA	NA	1
11	326	8.20	-0.0229	-0.0006	8.2100	-0.8	0.4260	NA	NA	-1
12	339	8.10	-0.2271	-0.0118	8.3782	-9.9	0.0000	NA	NA	-1

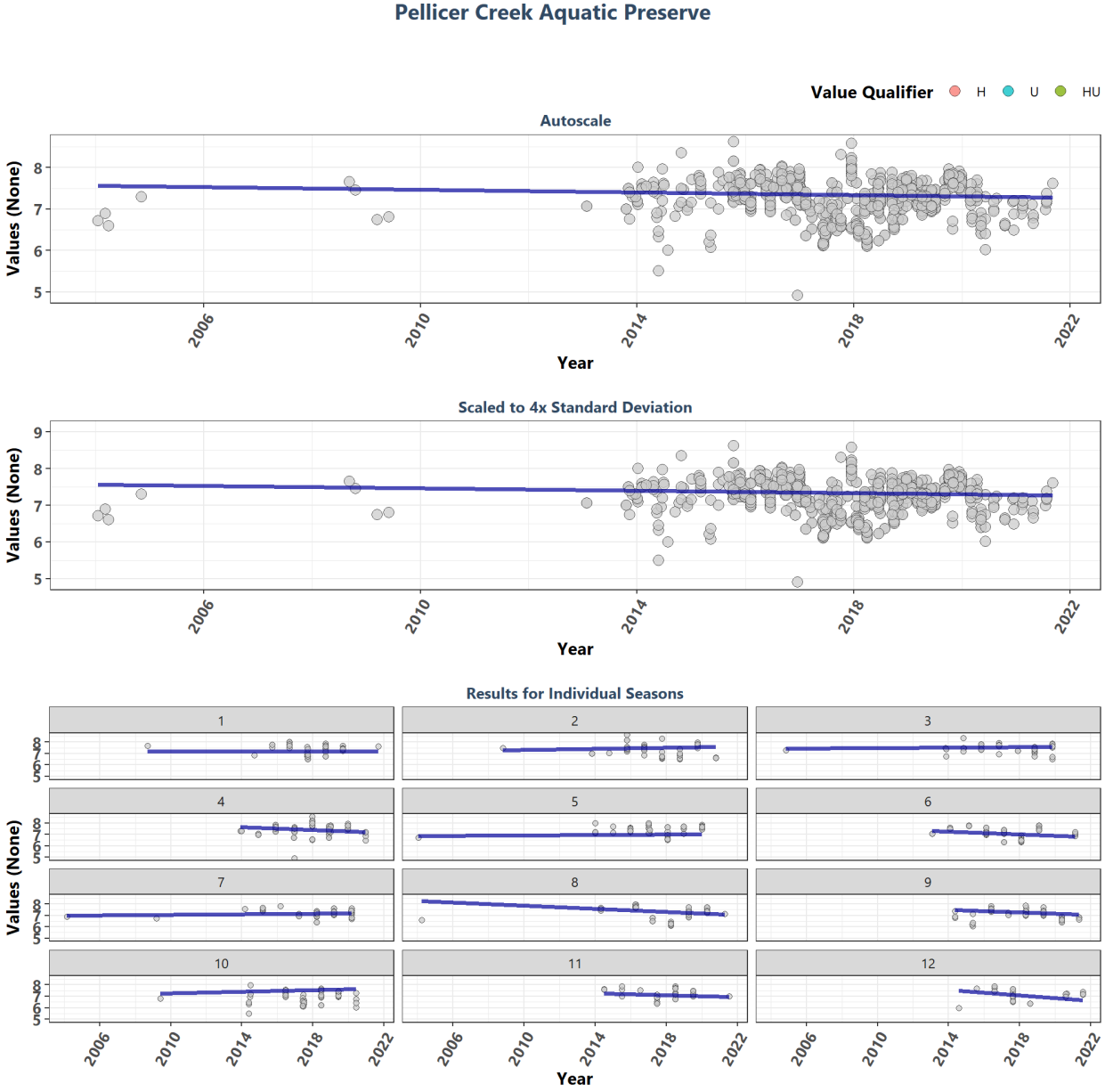
[a] $p < 0.00005$ appear as 0 due to rounding

North Fork St. Lucie Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	3154	7.70	-0.1341	-0.0083	7.8072	-11.1	0.0000	51.9	0	-1
1	240	7.80	-0.1349	-0.0080	7.9310	-6.5	0.0000	NA	NA	-1
2	245	7.80	-0.2311	-0.0143	8.0429	-4.9	0.0000	NA	NA	-1
3	269	7.80	-0.0096	0.0000	7.5750	-2.3	0.0223	NA	NA	-1
4	236	7.80	-0.1135	-0.0065	7.7294	-3.1	0.0019	NA	NA	-1
5	286	7.80	0.0013	0.0000	7.7000	-5.9	0.0000	NA	NA	-1
6	262	7.70	-0.0926	-0.0050	7.8850	0.0	0.9746	NA	NA	-1
7	269	7.60	-0.1712	-0.0105	7.6916	-2.8	0.0053	NA	NA	-1
8	286	7.58	-0.2104	-0.0143	8.0857	-0.2	0.8088	NA	NA	-1
9	298	7.50	-0.1392	-0.0100	7.7150	-3.6	0.0003	NA	NA	-1
10	252	7.46	-0.0333	0.0000	7.6000	-4.1	0.0000	NA	NA	-1
11	252	7.60	-0.2804	-0.0194	8.1875	-0.8	0.4280	NA	NA	-1
12	259	7.71	-0.2136	-0.0111	7.9211	-5.2	0.0000	NA	NA	-1

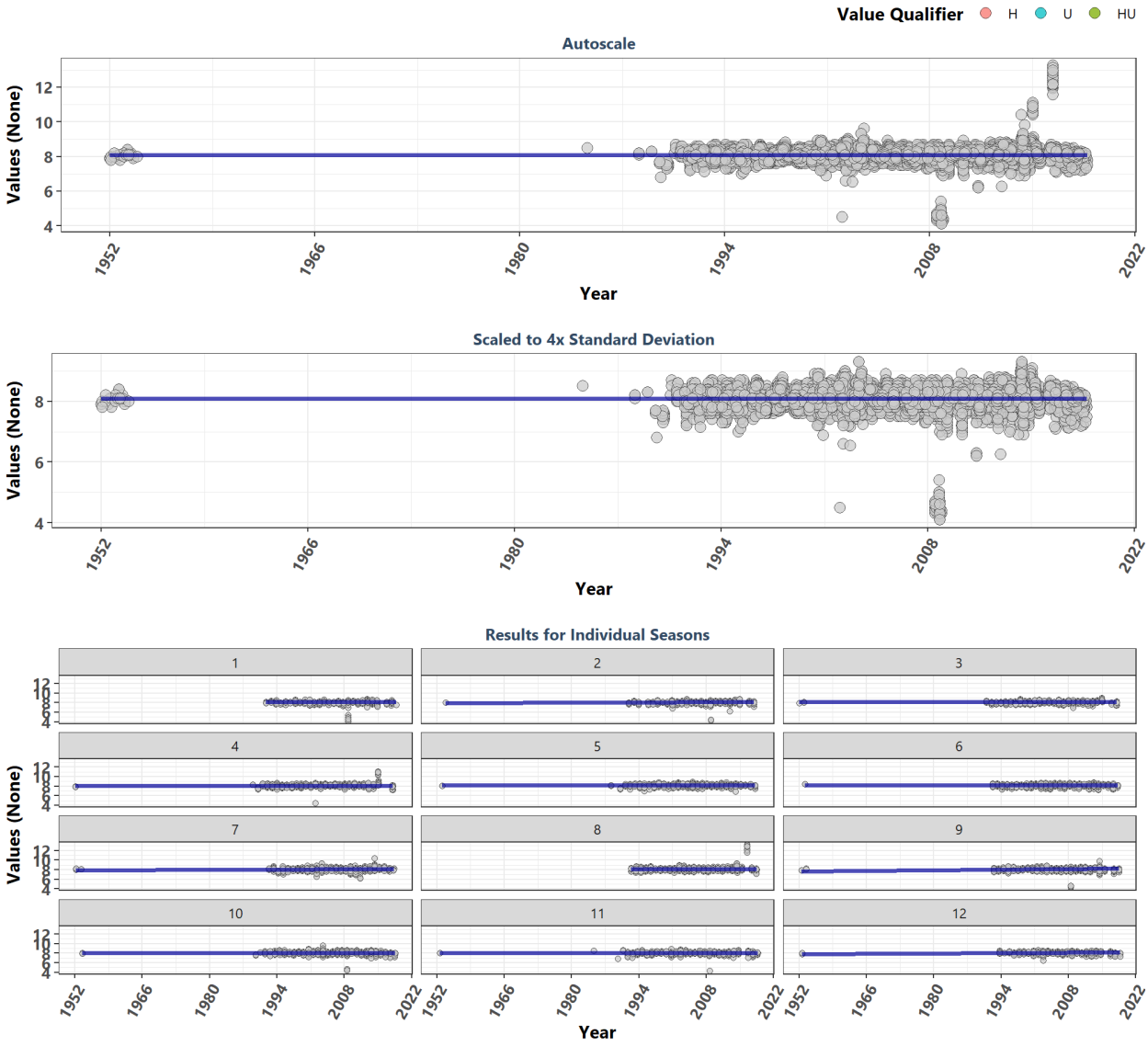
[a] $p < 0.00005$ appear as 0 due to rounding



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	748	7.29	-0.0480	-0.0167	7.6023	-1.6	0.1017	20.3	0.0411	0
1	61	7.40	-0.0063	0.0000	7.2000	-1.8	0.0666	NA	NA	0
2	75	7.31	0.0675	0.0250	7.1000	-1.5	0.1453	NA	NA	0
3	65	7.54	0.0389	0.0079	7.4221	0.5	0.6442	NA	NA	0
4	85	7.50	-0.1574	-0.0650	8.4400	0.9	0.3537	NA	NA	0
5	73	7.51	0.0167	0.0071	6.8587	1.4	0.1662	NA	NA	0
6	70	7.14	-0.1333	-0.0632	8.0137	-1.3	0.2109	NA	NA	0
7	57	7.20	0.0098	0.0100	6.9800	-0.1	0.9492	NA	NA	0
8	62	7.28	-0.1132	-0.0675	8.3900	-1.1	0.2632	NA	NA	0
9	61	7.09	-0.0957	-0.0567	8.1817	-3.5	0.0005	NA	NA	0
10	68	6.96	0.1088	0.0380	6.9400	0.2	0.8411	NA	NA	0
11	41	7.14	-0.1006	-0.0425	7.7825	0.1	0.9349	NA	NA	0
12	30	7.06	-0.2984	-0.1200	9.0100	-1.1	0.2735	NA	NA	0

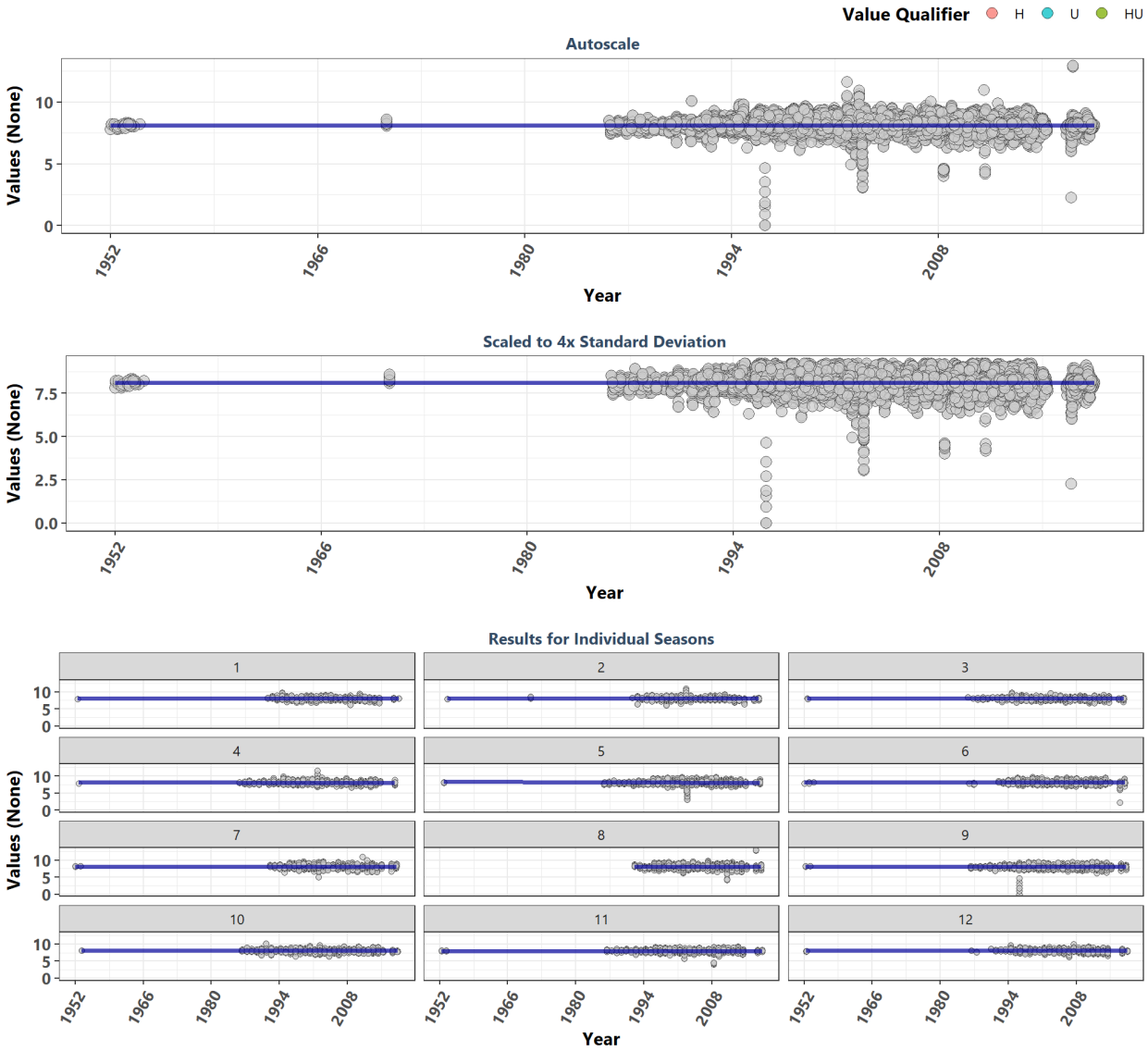
[a] $p < 0.00005$ appear as 0 due to rounding

Pine Island Sound Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	19518	8.10	0.1031	0.0000	8.1000	22.0	0.0000	73.1	0	-1
1	1419	8.00	0.0788	0.0000	8.1000	3.3	0.0009	NA	NA	-1
2	1391	8.10	0.1311	0.0046	7.8508	7.4	0.0000	NA	NA	1
3	1539	8.10	0.1175	0.0000	8.1000	7.7	0.0000	NA	NA	-1
4	1574	8.10	0.0527	0.0000	8.1000	7.1	0.0000	NA	NA	-1
5	1590	8.20	0.1080	0.0000	8.2000	6.4	0.0000	NA	NA	-1
6	1656	8.20	0.1064	0.0000	8.2000	6.7	0.0000	NA	NA	-1
7	1911	8.12	0.1296	0.0038	7.8962	10.3	0.0000	NA	NA	1
8	1814	8.12	0.0339	0.0000	8.1000	10.1	0.0000	NA	NA	-1
9	1727	8.10	0.1561	0.0077	7.6969	5.0	0.0000	NA	NA	1
10	1757	8.10	0.0581	0.0000	8.0000	2.2	0.0312	NA	NA	-1
11	1681	8.10	0.0952	0.0000	8.1000	5.9	0.0000	NA	NA	-1
12	1459	8.10	0.1567	0.0067	7.7533	3.1	0.0022	NA	NA	1

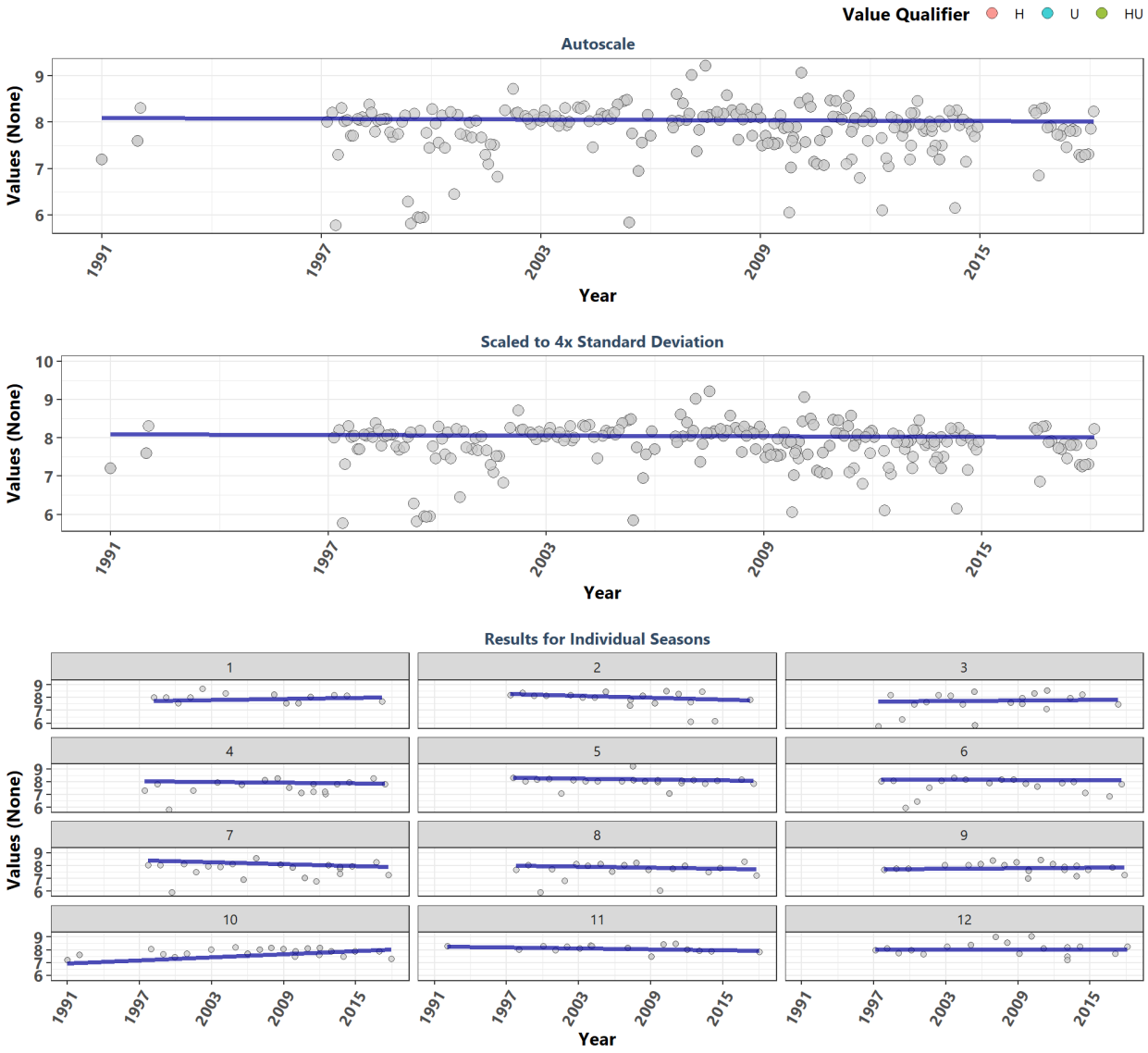
[a] $p < 0.00005$ appear as 0 due to rounding



Season	N	Median	tau	Slope	Int.	z	p__z	chi__sq	p__chi__sq	Trend
All	51197	8.10	-0.0154	0.0000	8.1050	-3.7	0.0002	156.7	0	-1
1	3446	8.06	-0.0611	-0.0007	8.1371	-8.0	0.0000	NA	NA	-1
2	3509	8.03	-0.0112	0.0000	8.1100	-2.5	0.0116	NA	NA	-1
3	3525	8.04	-0.0185	0.0000	8.1000	-3.7	0.0002	NA	NA	-1
4	3719	8.10	-0.0406	-0.0014	8.1471	-1.7	0.0891	NA	NA	-1
5	5297	8.10	-0.0902	-0.0045	8.2918	1.1	0.2746	NA	NA	-1
6	4216	8.10	0.0100	0.0000	8.1000	3.4	0.0008	NA	NA	-1
7	3743	8.11	0.0221	0.0000	8.1200	-1.0	0.3017	NA	NA	-1
8	4493	8.10	0.0345	0.0008	8.0550	3.7	0.0002	NA	NA	1
9	6359	8.12	0.0372	0.0012	8.0338	2.6	0.0081	NA	NA	1
10	5080	8.10	-0.0454	-0.0011	8.1556	-4.9	0.0000	NA	NA	-1
11	3717	8.10	-0.0283	0.0000	8.0300	-5.6	0.0000	NA	NA	-1
12	4093	8.07	-0.0417	0.0000	8.0400	-3.9	0.0001	NA	NA	-1

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

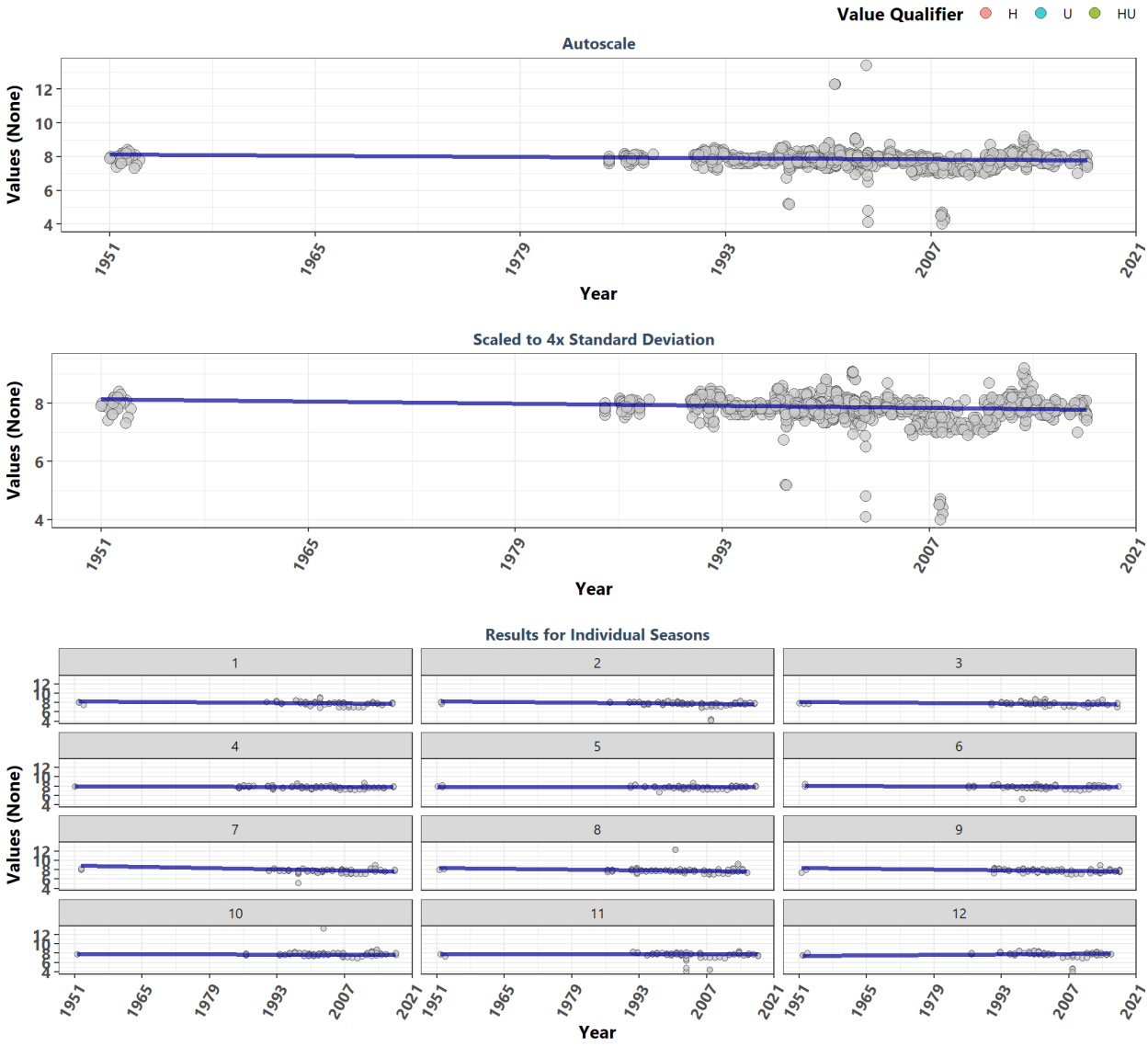
Rocky Bayou State Park Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	303	8.00	-0.0299	-0.0027	8.0953	-0.6	0.5482	14.3	0.2155	0
1	17	8.03	0.2273	0.0137	7.6574	0.0	1.0000	NA	NA	0
2	26	8.14	-0.1839	-0.0255	8.4436	-0.9	0.3518	NA	NA	0
3	25	7.67	0.0906	0.0075	7.6275	1.6	0.1151	NA	NA	0
4	24	7.77	-0.0923	-0.0100	8.0950	0.6	0.5494	NA	NA	0
5	28	8.05	-0.1524	-0.0113	8.3825	-1.6	0.1207	NA	NA	0
6	23	7.88	-0.0109	-0.0008	8.1635	-1.2	0.2125	NA	NA	0
7	26	7.92	-0.1323	-0.0250	8.5775	-0.6	0.5212	NA	NA	0
8	26	7.80	-0.1897	-0.0127	8.0955	0.2	0.8769	NA	NA	0
9	30	7.96	0.0246	0.0050	7.7175	-1.4	0.1563	NA	NA	0
10	33	7.89	0.2267	0.0400	6.9500	1.9	0.0638	NA	NA	0
11	21	8.18	-0.2090	-0.0114	8.2599	-0.9	0.3468	NA	NA	0
12	24	8.15	0.0000	-0.0001	8.0311	0.0	0.9602	NA	NA	0

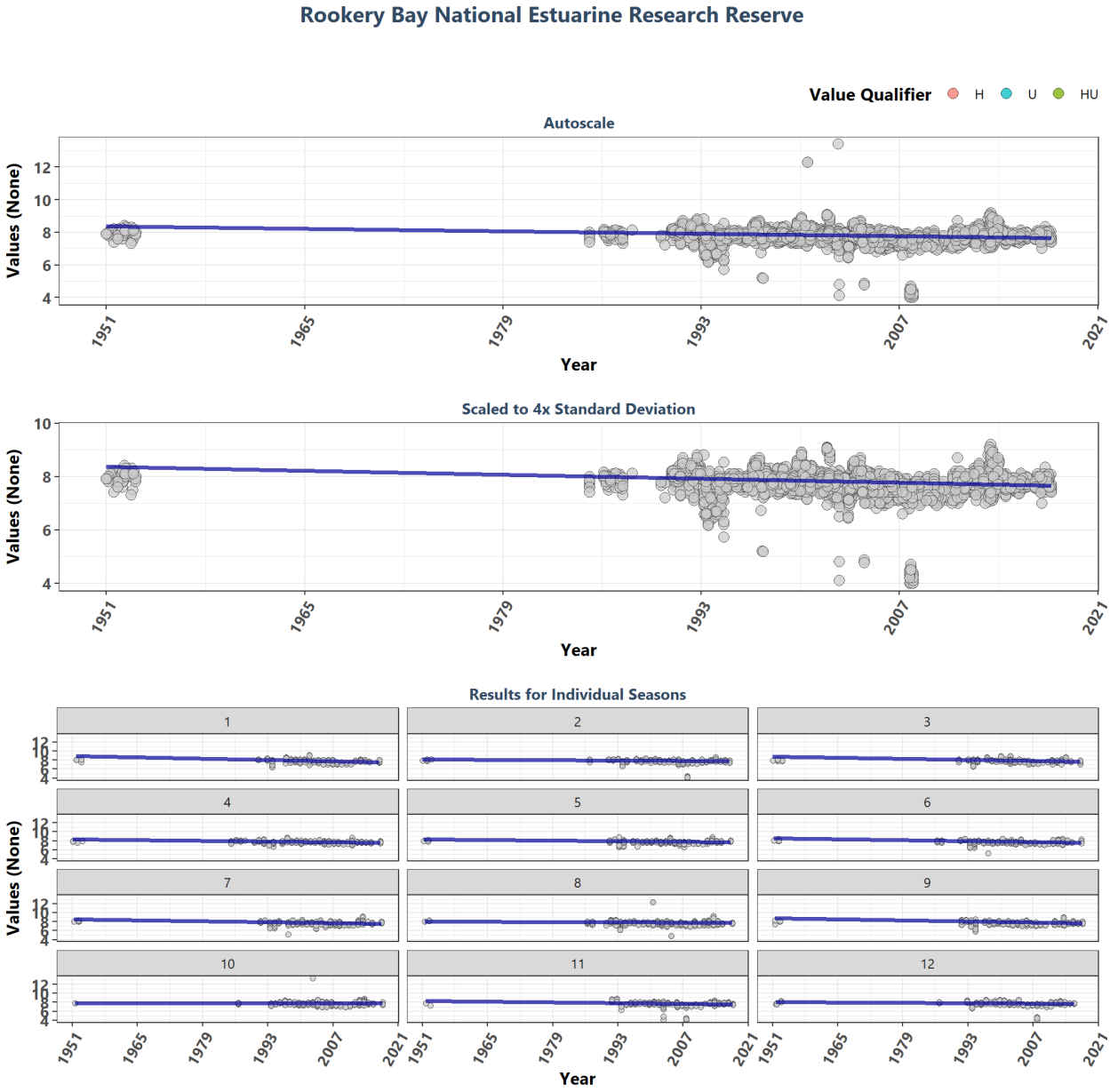
[a] $p < 0.00005$ appear as 0 due to rounding

Rookery Bay Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	1500	7.84	-0.1123	-0.0054	8.1372	-6.5	0.0000	47	0	-1
1	104	7.90	-0.1481	-0.0083	8.3250	-2.2	0.0253	NA	NA	-1
2	116	7.85	-0.1670	-0.0100	8.3300	-3.4	0.0006	NA	NA	-1
3	123	7.90	-0.2168	-0.0080	8.1920	-0.7	0.4704	NA	NA	-1
4	147	7.80	-0.0438	-0.0013	7.9613	-3.9	0.0001	NA	NA	-1
5	98	7.82	-0.0046	0.0000	7.8200	-0.1	0.9483	NA	NA	-1
6	142	7.78	-0.0950	-0.0040	8.0825	-0.4	0.6786	NA	NA	-1
7	112	7.76	-0.2903	-0.0177	8.8546	-0.6	0.5213	NA	NA	-1
8	140	7.88	-0.2176	-0.0100	8.3400	-1.7	0.0950	NA	NA	-1
9	131	7.84	-0.2145	-0.0095	8.3345	-3.7	0.0002	NA	NA	-1
10	129	7.84	-0.0409	-0.0016	7.8375	2.4	0.0153	NA	NA	-1
11	127	7.83	-0.0235	-0.0004	7.8009	-2.8	0.0052	NA	NA	-1
12	131	7.97	0.1439	0.0075	7.4600	-4.9	0.0000	NA	NA	1

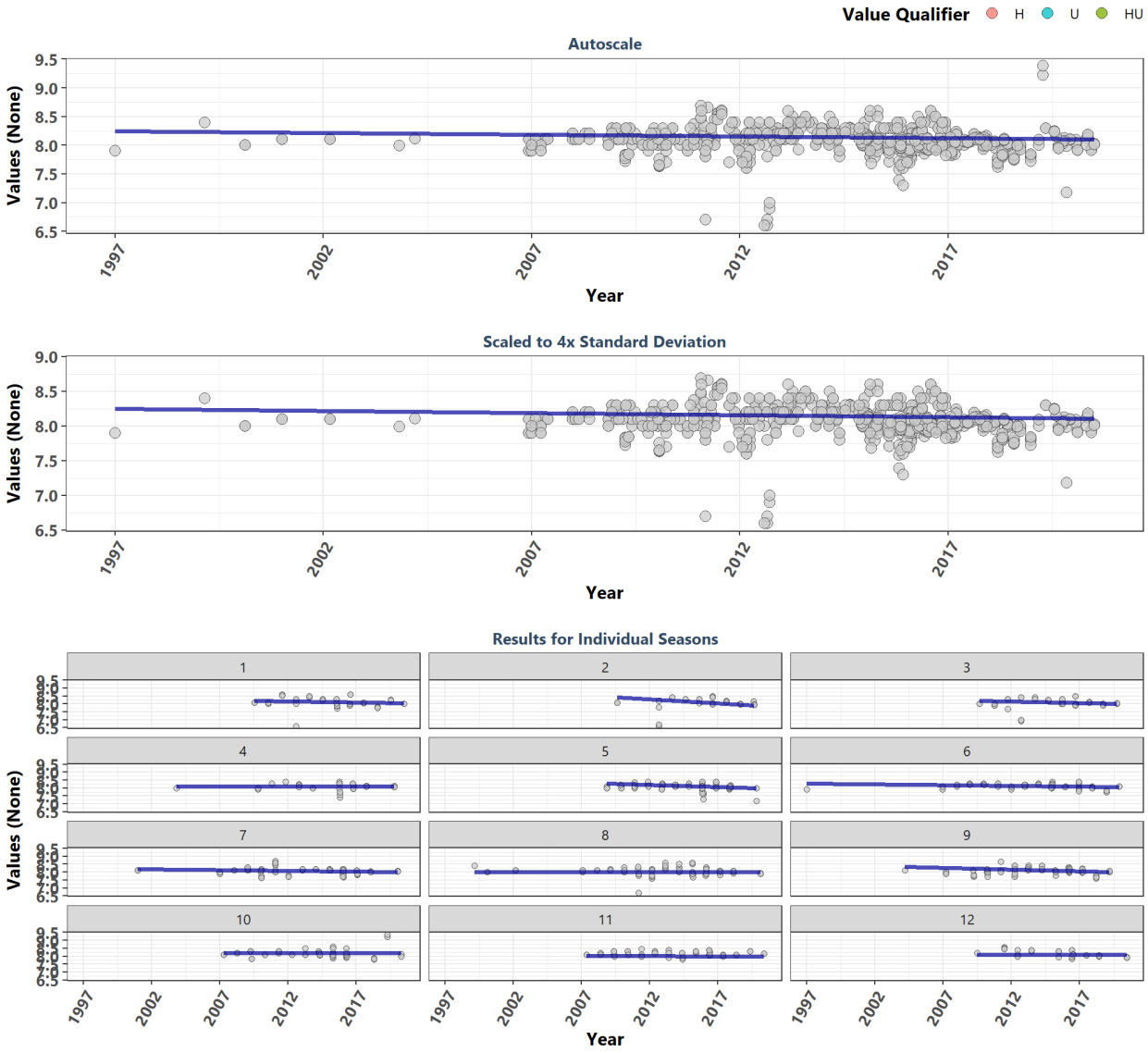
[a] $p < 0.00005$ appear as 0 due to rounding



Season	N	Median	tau	Slope	Int.	z	p__z	chi__sq	p__chi__sq	Trend
All	5980	7.80	-0.1955	-0.0104	8.3596	-22.4	0.0000	114.5	0	-1
1	459	7.90	-0.3098	-0.0214	8.8929	-10.2	0.0000	NA	NA	-1
2	488	7.83	-0.1363	-0.0056	8.0981	-9.5	0.0000	NA	NA	-1
3	405	7.90	-0.2304	-0.0178	8.7913	-5.3	0.0000	NA	NA	-1
4	506	7.80	-0.2442	-0.0111	8.3526	-8.2	0.0000	NA	NA	-1
5	437	7.80	-0.1754	-0.0087	8.3261	-4.3	0.0000	NA	NA	-1
6	557	7.80	-0.2860	-0.0143	8.5729	-4.8	0.0000	NA	NA	-1
7	460	7.80	-0.2413	-0.0141	8.5045	-3.1	0.0019	NA	NA	-1
8	606	7.80	-0.0963	-0.0048	8.0379	-7.2	0.0000	NA	NA	-1
9	540	7.80	-0.3158	-0.0164	8.7214	-10.8	0.0000	NA	NA	-1
10	545	7.80	0.0077	0.0000	7.8000	0.3	0.7877	NA	NA	-1
11	471	7.80	-0.1944	-0.0113	8.3667	-7.9	0.0000	NA	NA	-1
12	506	7.90	-0.1350	-0.0062	8.0971	-7.8	0.0000	NA	NA	-1

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

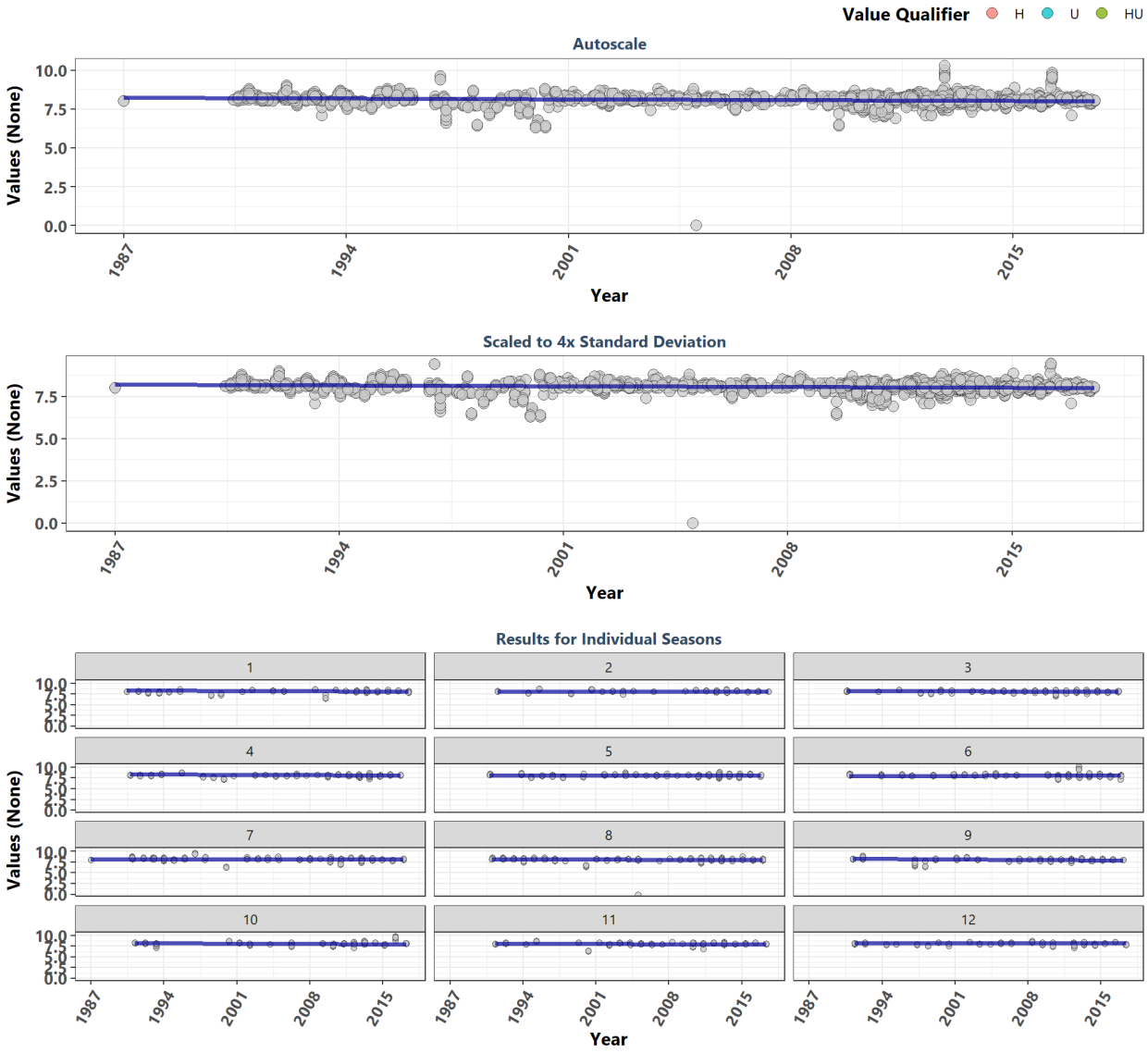
St. Andrews State Park Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	928	8.10	-0.1385	-0.0059	8.2449	-5.7	0.0000	27.7	0.0036	-1
1	47	8.08	-0.1758	-0.0114	8.3286	-2.3	0.0193	NA	NA	-1
2	45	8.10	-0.4708	-0.0525	9.0825	-1.7	0.0848	NA	NA	-1
3	50	8.00	-0.2458	-0.0150	8.3700	-0.1	0.8848	NA	NA	-1
4	39	8.10	0.0668	0.0000	8.1000	0.1	0.8809	NA	NA	-1
5	103	8.05	-0.2350	-0.0271	8.5957	-2.3	0.0200	NA	NA	-1
6	96	8.10	-0.1577	-0.0095	8.2710	-3.6	0.0003	NA	NA	-1
7	109	8.10	-0.1529	-0.0080	8.2020	-2.5	0.0133	NA	NA	-1
8	112	8.00	-0.0147	0.0000	8.0000	-1.3	0.2015	NA	NA	-1
9	104	8.10	-0.2406	-0.0200	8.4600	1.0	0.3078	NA	NA	-1
10	87	8.10	-0.1229	-0.0011	8.2189	-3.4	0.0007	NA	NA	-1
11	98	8.20	-0.0808	-0.0011	8.0178	-1.8	0.0664	NA	NA	-1
12	38	8.09	0.0175	0.0000	8.1000	-4.2	0.0000	NA	NA	-1

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

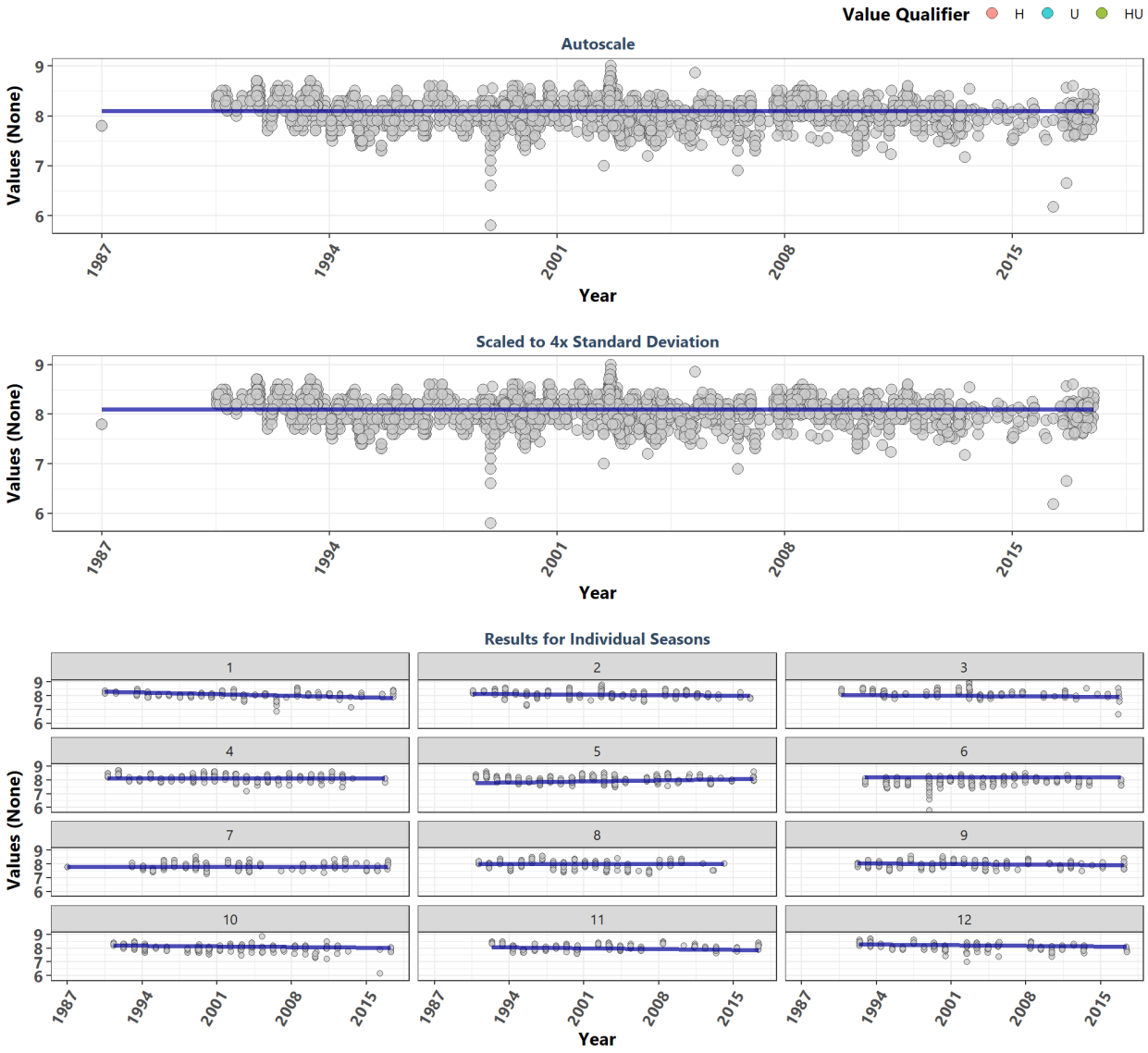
St. Joseph Bay Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	3464	8.10	-0.1450	-0.0067	8.2186	-14.7	0.0000	150.5	0	-1
1	230	8.04	-0.2394	-0.0091	8.2864	1.6	0.1073	NA	NA	-1
2	195	8.10	-0.0107	0.0000	8.1000	-2.5	0.0142	NA	NA	-1
3	275	8.10	-0.1137	-0.0056	8.2278	-3.7	0.0002	NA	NA	-1
4	284	8.10	-0.3223	-0.0111	8.4222	-1.5	0.1361	NA	NA	-1
5	316	8.20	-0.0589	0.0000	8.1000	-2.1	0.0322	NA	NA	-1
6	375	8.20	0.0707	0.0008	8.0227	-9.4	0.0000	NA	NA	1
7	394	8.20	-0.0263	0.0000	8.1000	-7.2	0.0000	NA	NA	-1
8	430	8.20	-0.1479	-0.0062	8.2188	-10.3	0.0000	NA	NA	-1
9	268	8.02	-0.3291	-0.0125	8.3125	-1.9	0.0530	NA	NA	-1
10	238	8.10	-0.1172	-0.0074	8.2185	-2.6	0.0084	NA	NA	-1
11	191	8.10	-0.0788	-0.0040	8.1080	-0.5	0.5867	NA	NA	-1
12	268	8.10	-0.0803	-0.0043	8.2913	-0.3	0.7912	NA	NA	-1

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

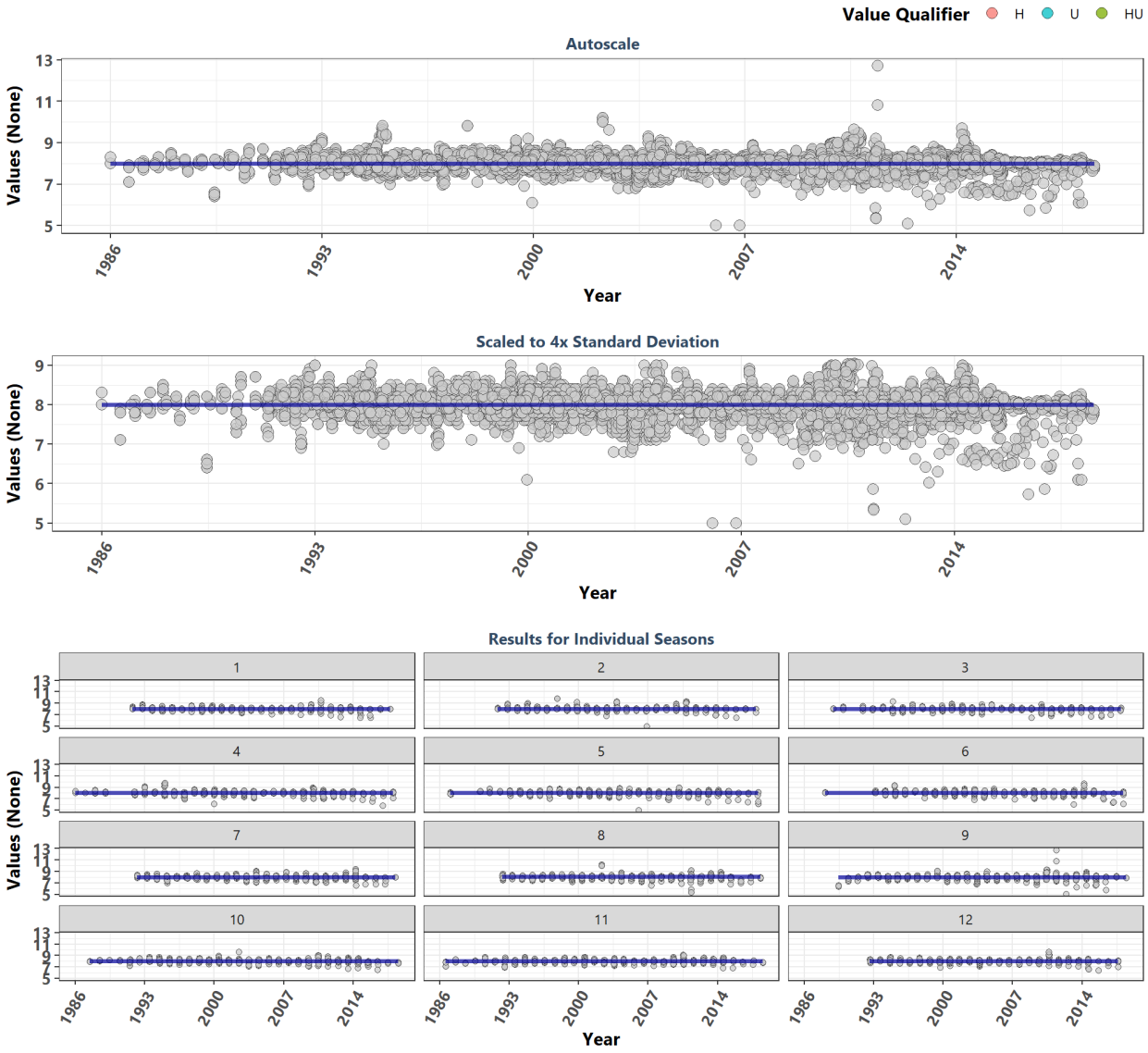
St. Martins Marsh Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	4102	8.1	-0.0784	0.0000	8.1019	-7.6	0.0000	152.9	0	-1
1	344	8.1	-0.3058	-0.0167	8.3500	-1.9	0.0549	NA	NA	-1
2	472	8.1	-0.1128	-0.0046	8.1600	-5.4	0.0000	NA	NA	-1
3	286	8.2	-0.1262	-0.0059	8.0941	-4.4	0.0000	NA	NA	-1
4	415	8.2	-0.0683	0.0000	8.1000	0.6	0.5407	NA	NA	-1
5	443	8.0	0.2057	0.0125	7.7125	-4.0	0.0001	NA	NA	1
6	375	7.9	0.0199	0.0000	8.2000	6.0	0.0000	NA	NA	-1
7	248	7.8	0.0731	0.0008	7.7875	1.7	0.0841	NA	NA	1
8	234	8.0	-0.0366	0.0000	8.0000	-2.4	0.0147	NA	NA	-1
9	254	8.0	-0.1219	-0.0048	8.0667	-0.9	0.3811	NA	NA	-1
10	386	8.0	-0.1632	-0.0063	8.2062	-3.6	0.0003	NA	NA	-1
11	303	8.1	-0.1065	-0.0077	8.1038	-3.0	0.0031	NA	NA	-1
12	342	8.1	-0.1706	-0.0071	8.3143	-8.5	0.0000	NA	NA	-1

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

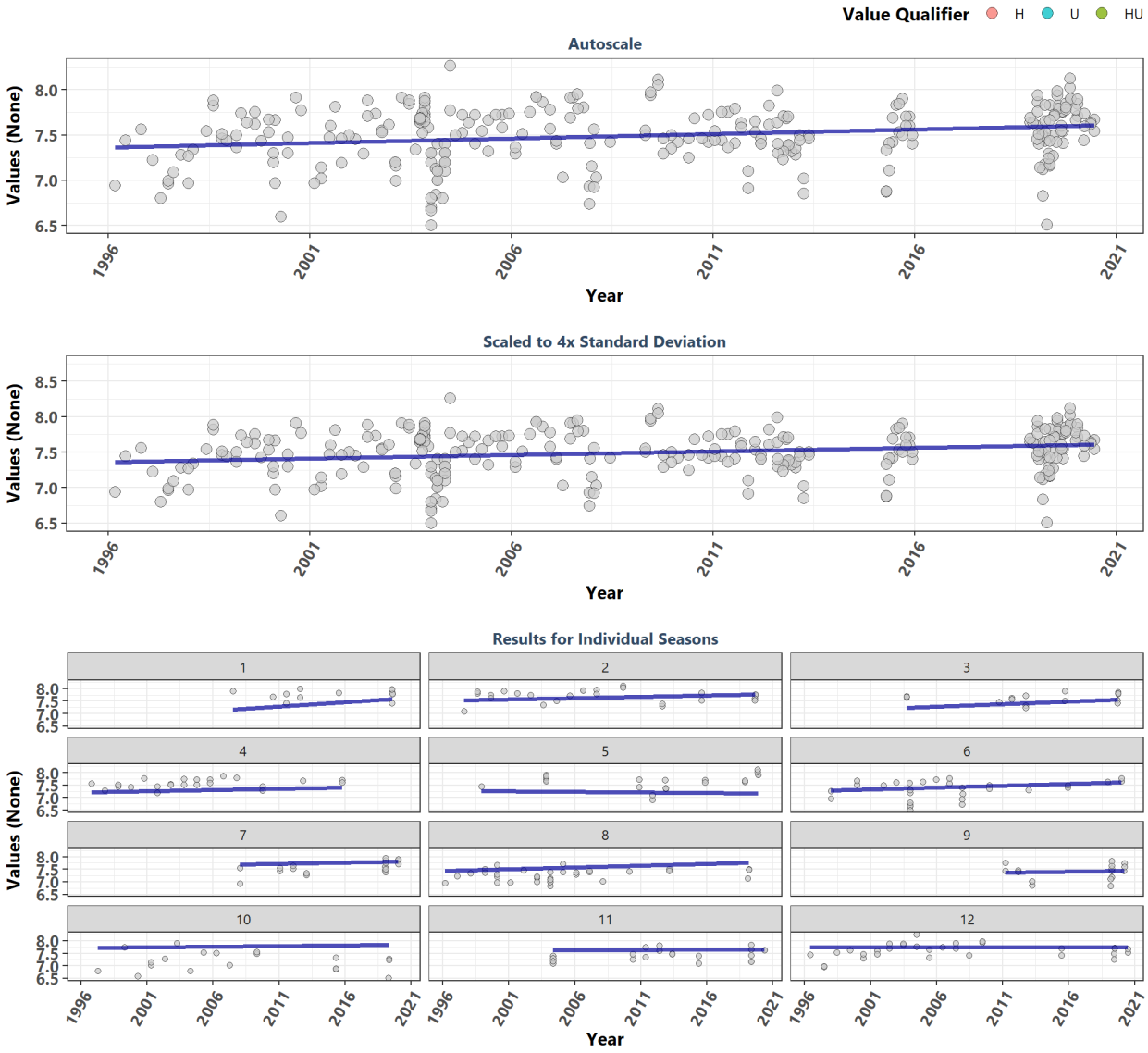
Terra Ceia Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p__z	chi__sq	p__chi__sq	Trend
All	16146	8.0	-0.0084	0	8.0	-1.2	0.2426	45.6	0	0
1	1199	8.0	-0.0381	0	8.0	-4.8	0.0000	NA	NA	0
2	1155	8.0	0.0334	0	8.0	-1.4	0.1644	NA	NA	0
3	1247	8.0	0.0293	0	8.0	-2.9	0.0041	NA	NA	0
4	1245	8.0	-0.0060	0	8.0	0.2	0.8423	NA	NA	0
5	1435	8.1	0.0238	0	8.0	0.7	0.4572	NA	NA	0
6	1261	8.0	0.0037	0	8.0	-0.3	0.7479	NA	NA	0
7	1365	8.0	0.0084	0	8.0	1.6	0.1027	NA	NA	0
8	1689	8.0	0.0130	0	8.1	2.1	0.0385	NA	NA	0
9	1448	8.0	-0.0921	0	8.0	0.5	0.6311	NA	NA	0
10	1575	8.0	-0.0538	0	8.0	-1.4	0.1744	NA	NA	0
11	1301	8.0	-0.0270	0	8.0	1.3	0.1963	NA	NA	0
12	1226	8.0	-0.0227	0	8.0	-2.0	0.0438	NA	NA	0

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

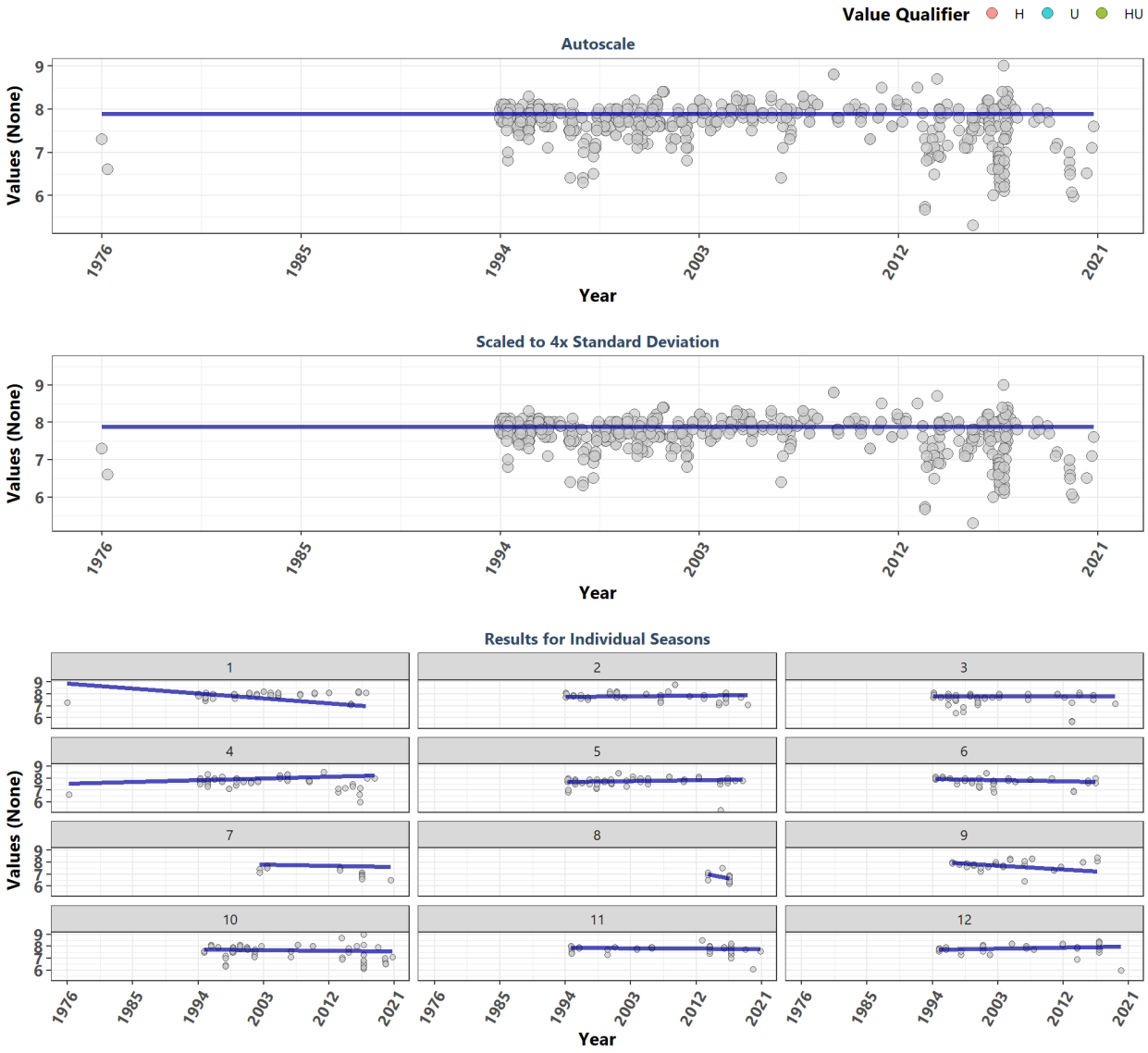
Tomoka Marsh Aquatic Preserve



Season	N	Median	tau	Slope	Int.	z	p_z	chi_sq	p_chi_sq	Trend
All	279	7.51	0.1628	0.0100	7.3622	3.9	0.0001	8.6	0.6565	1
1	12	7.80	0.3725	0.0333	6.7833	0.4	0.6660	NA	NA	1
2	25	7.75	0.1467	0.0106	7.5194	0.0	0.9813	NA	NA	1
3	19	7.64	0.3033	0.0191	7.0945	0.0	1.0000	NA	NA	1
4	23	7.55	0.0882	0.0100	7.2100	1.8	0.0671	NA	NA	1
5	25	7.70	-0.0409	-0.0047	7.2874	1.0	0.3044	NA	NA	-1
6	30	7.40	0.1862	0.0144	7.2678	1.4	0.1479	NA	NA	1
7	18	7.55	0.1806	0.0100	7.5700	2.2	0.0275	NA	NA	1
8	35	7.29	0.2767	0.0141	7.4371	1.8	0.0647	NA	NA	1
9	17	7.44	0.2185	0.0094	7.2147	0.5	0.6341	NA	NA	1
10	19	7.24	0.1061	0.0048	7.7164	-0.2	0.8326	NA	NA	1
11	25	7.40	0.0058	0.0022	7.6022	2.2	0.0309	NA	NA	1
12	31	7.66	0.0067	0.0005	7.7445	1.4	0.1568	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	523	7.80	-0.0551	0.0000	7.8864	-1.7	0.0933	47	0	0
1	44	7.95	-0.5500	-0.0462	8.8769	3.7	0.0002	NA	NA	0
2	44	7.80	0.1004	0.0044	7.6933	-2.1	0.0322	NA	NA	0
3	56	7.70	-0.0496	0.0000	7.8000	-1.0	0.3259	NA	NA	0
4	58	7.80	0.3763	0.0154	7.5500	-0.5	0.5831	NA	NA	0
5	63	7.80	0.1573	0.0083	7.5500	1.2	0.2410	NA	NA	0
6	49	7.80	-0.2220	-0.0129	8.1875	-2.0	0.0423	NA	NA	0
7	16	7.10	-0.1973	-0.0105	8.0737	-3.1	0.0017	NA	NA	0
8	13	6.70	-0.2692	-0.1133	11.2333	-1.5	0.1278	NA	NA	0
9	31	7.80	-0.2647	-0.0367	8.7000	0.9	0.3623	NA	NA	0
10	63	7.60	-0.0896	-0.0045	7.8045	-3.1	0.0020	NA	NA	0
11	45	7.80	-0.1212	-0.0045	7.9682	-1.2	0.2304	NA	NA	0
12	41	7.80	0.1161	0.0100	7.5300	1.5	0.1426	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
```

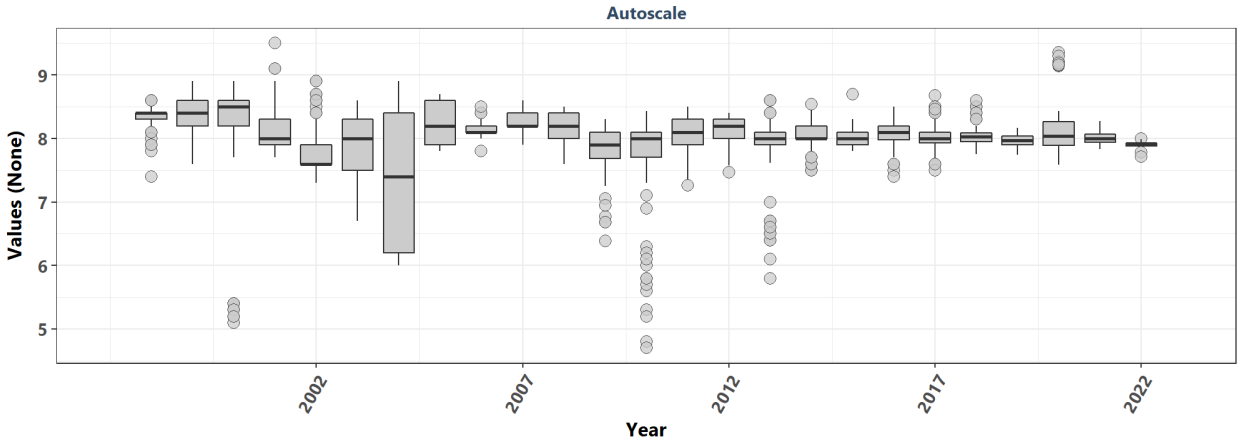
```
p2 <- ggplot(data=plot_data,
             aes(x=Year, y=ResultValue, group=Year)) +
   geom_boxplot(color="#333333", fill="#cccccc", outlier.shape=21,
                outlier.size=3, outlier.color="#333333",
                outlier.fill="#cccccc", outlier.alpha=0.75) +
   labs(subtitle="Scaled to 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") +
```

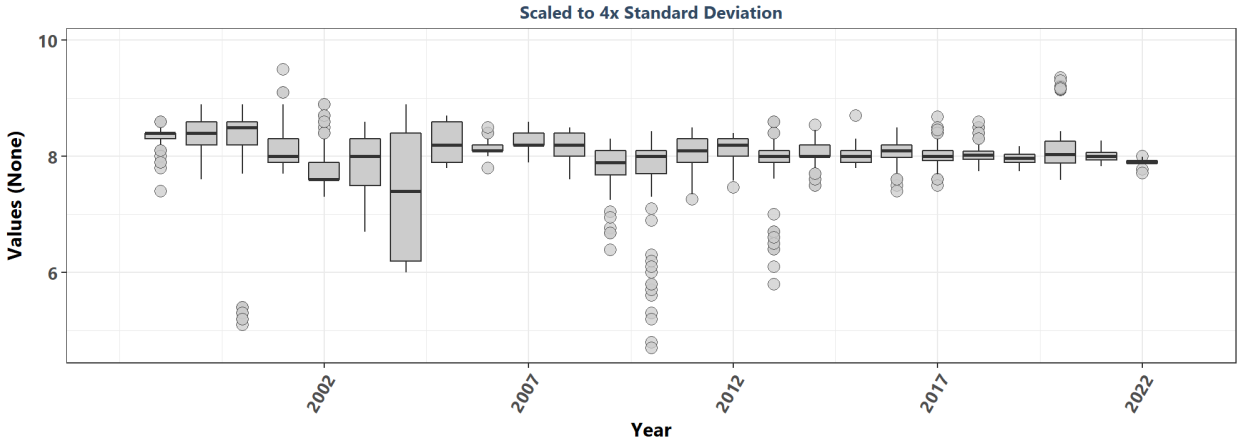
```
   ylim(min_RV, y_scale) +
   scale_x_continuous(limits=c(year_lower - 1, year_upper + 1),
                      breaks=rev(seq(year_upper,
                                     year_lower, -x_scale))) +
   plot_theme +
   theme(legend.position="top", legend.box="horizontal") +
   guides(color=guide_legend(nrow=1))

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") +
```

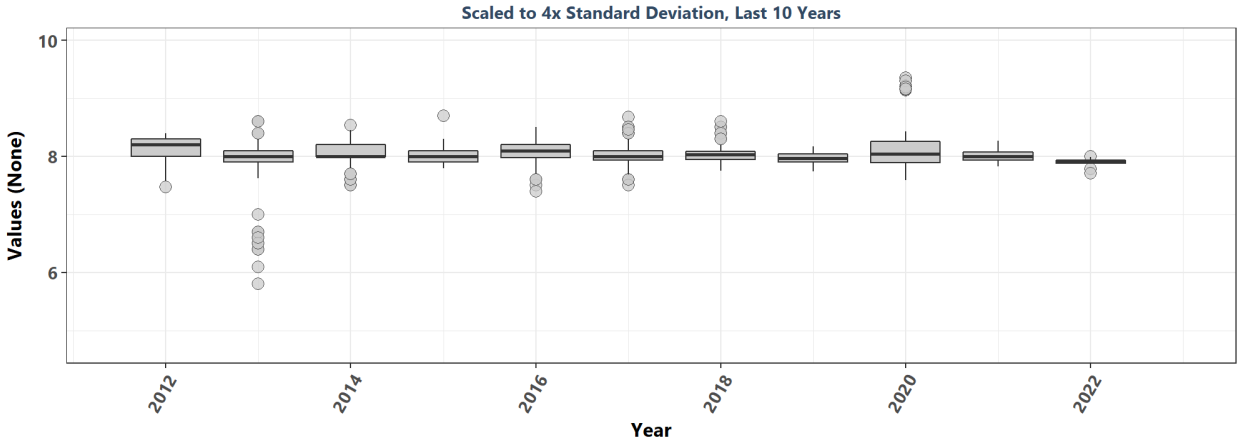
```
      guides(fill=guide_legend(nrow=1))

   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)
  }
}
```

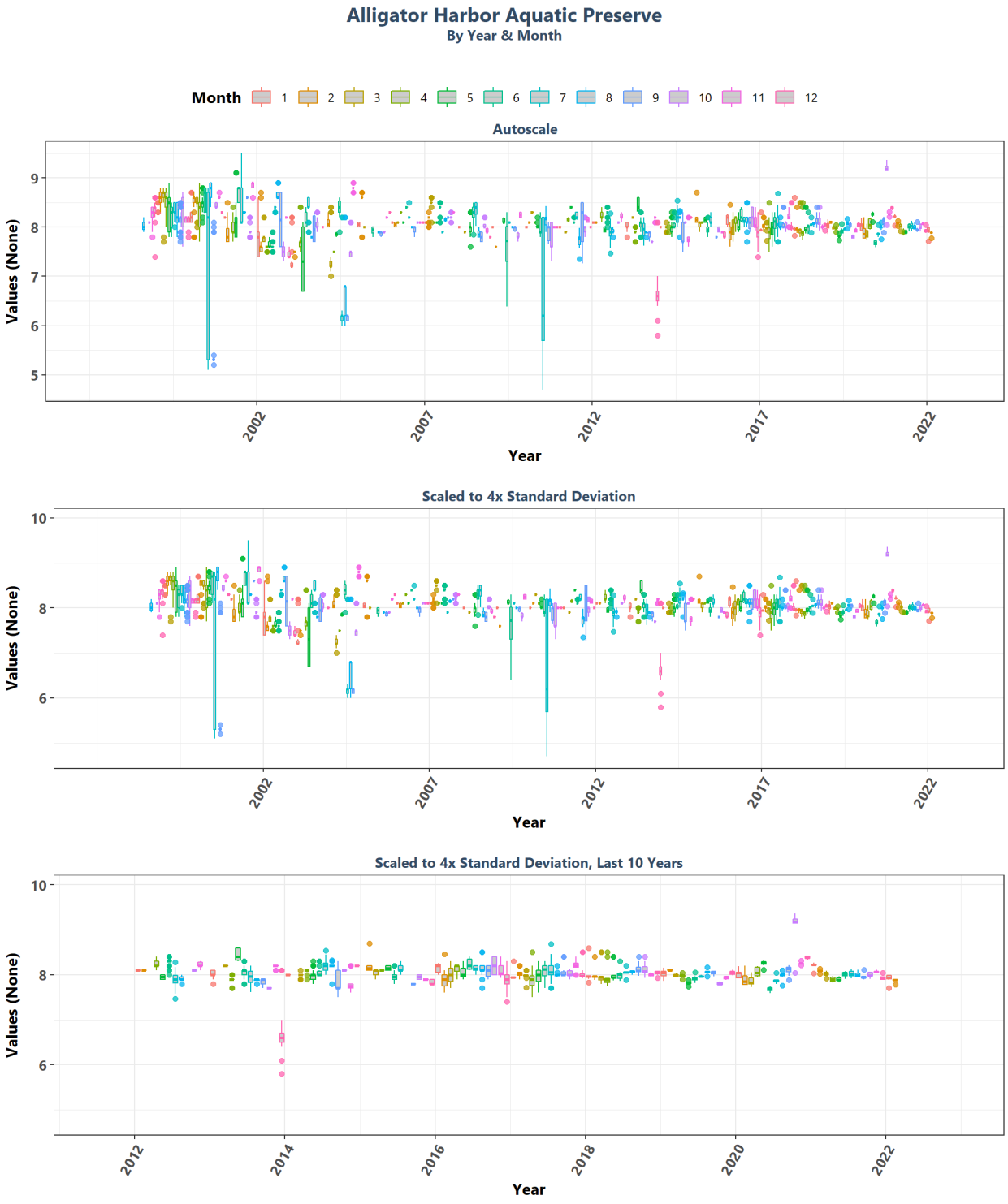
Alligator Harbor Aquatic Preserve

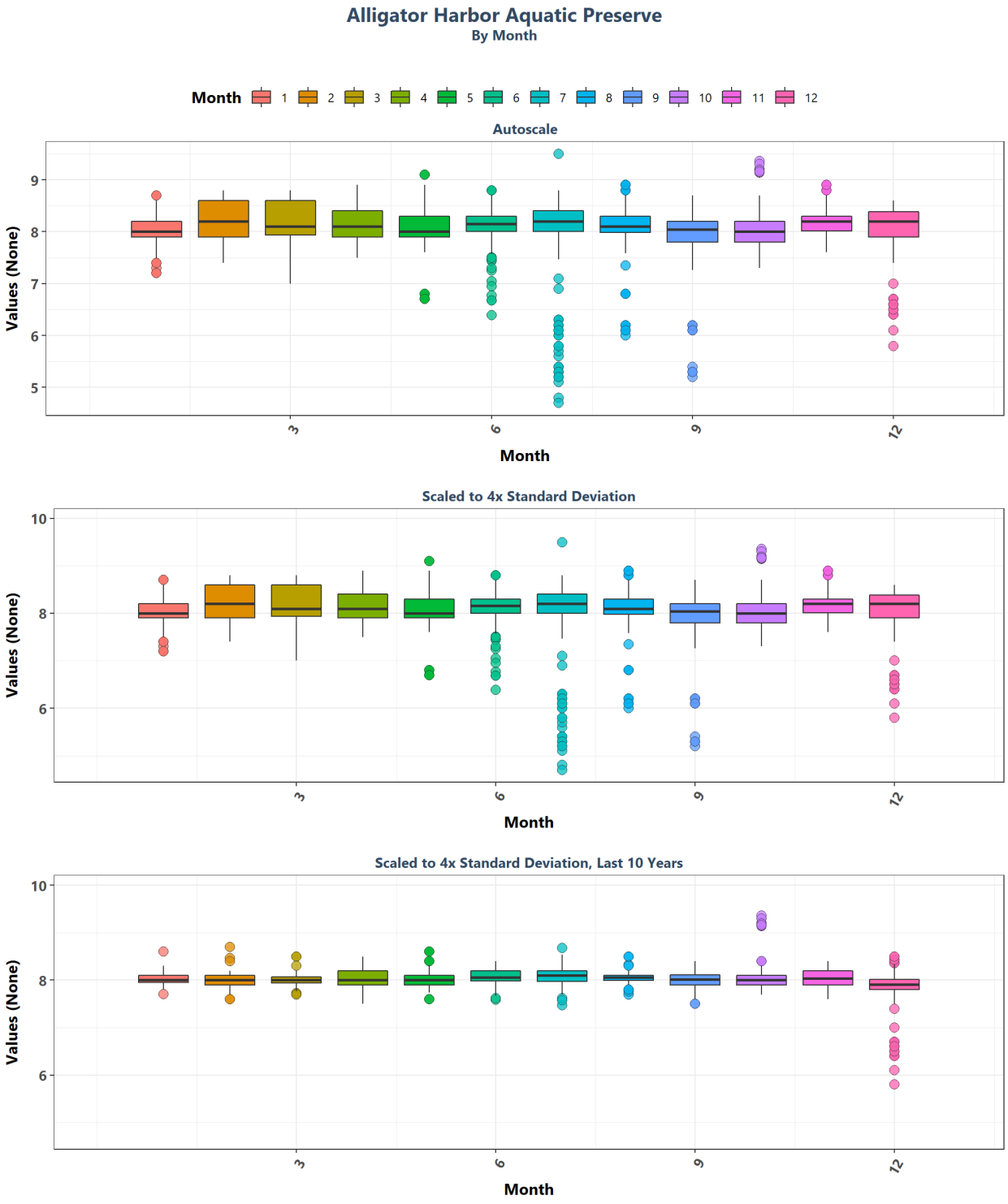
By Year





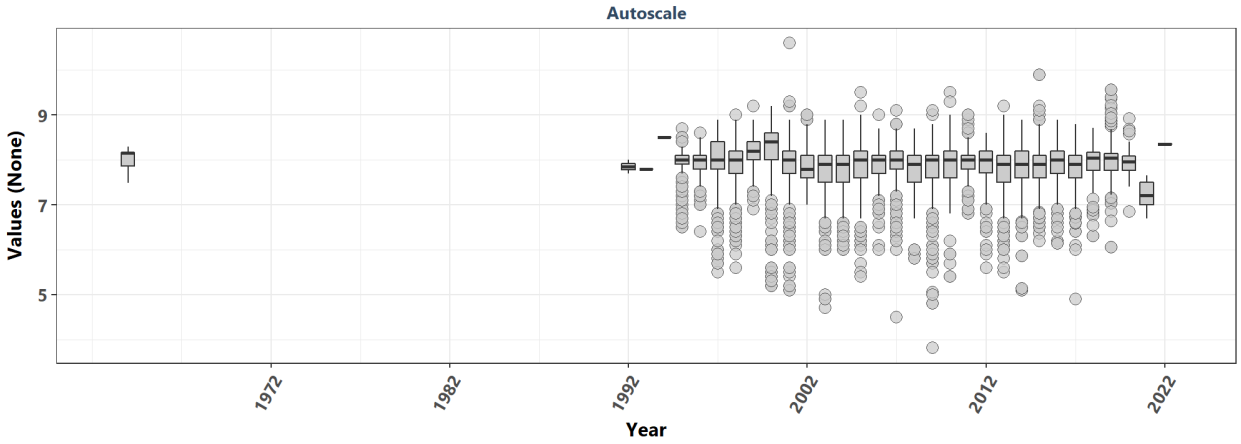


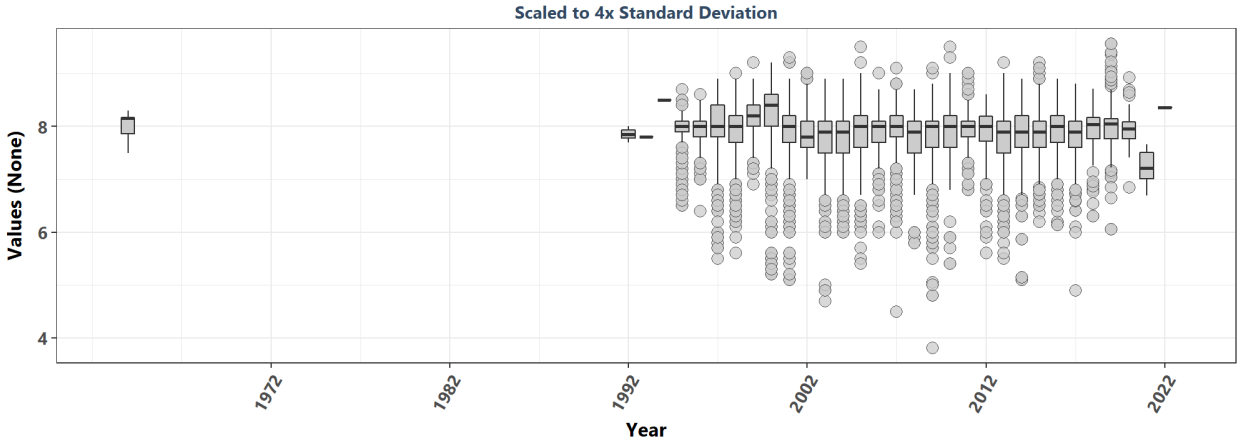

Alligator Harbor Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

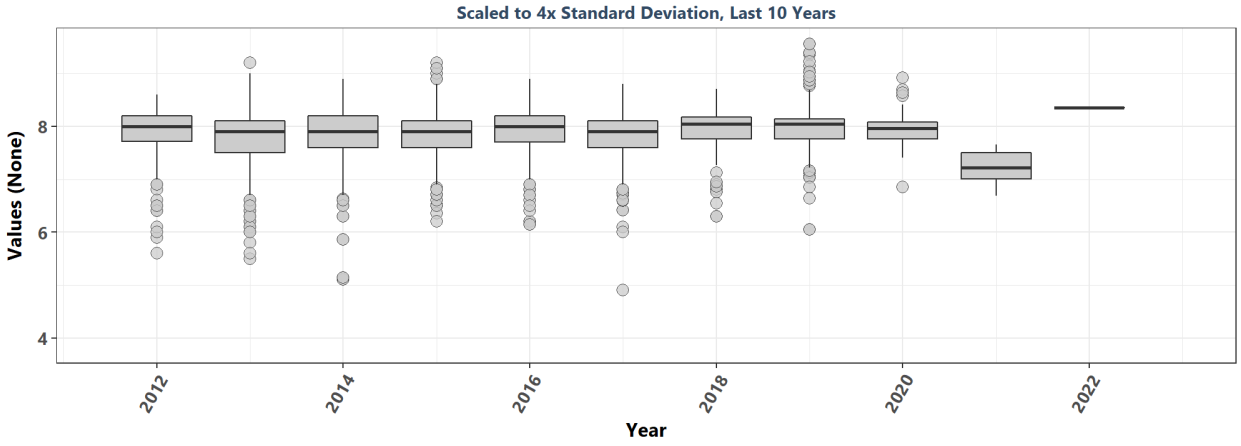

Alligator Harbor Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

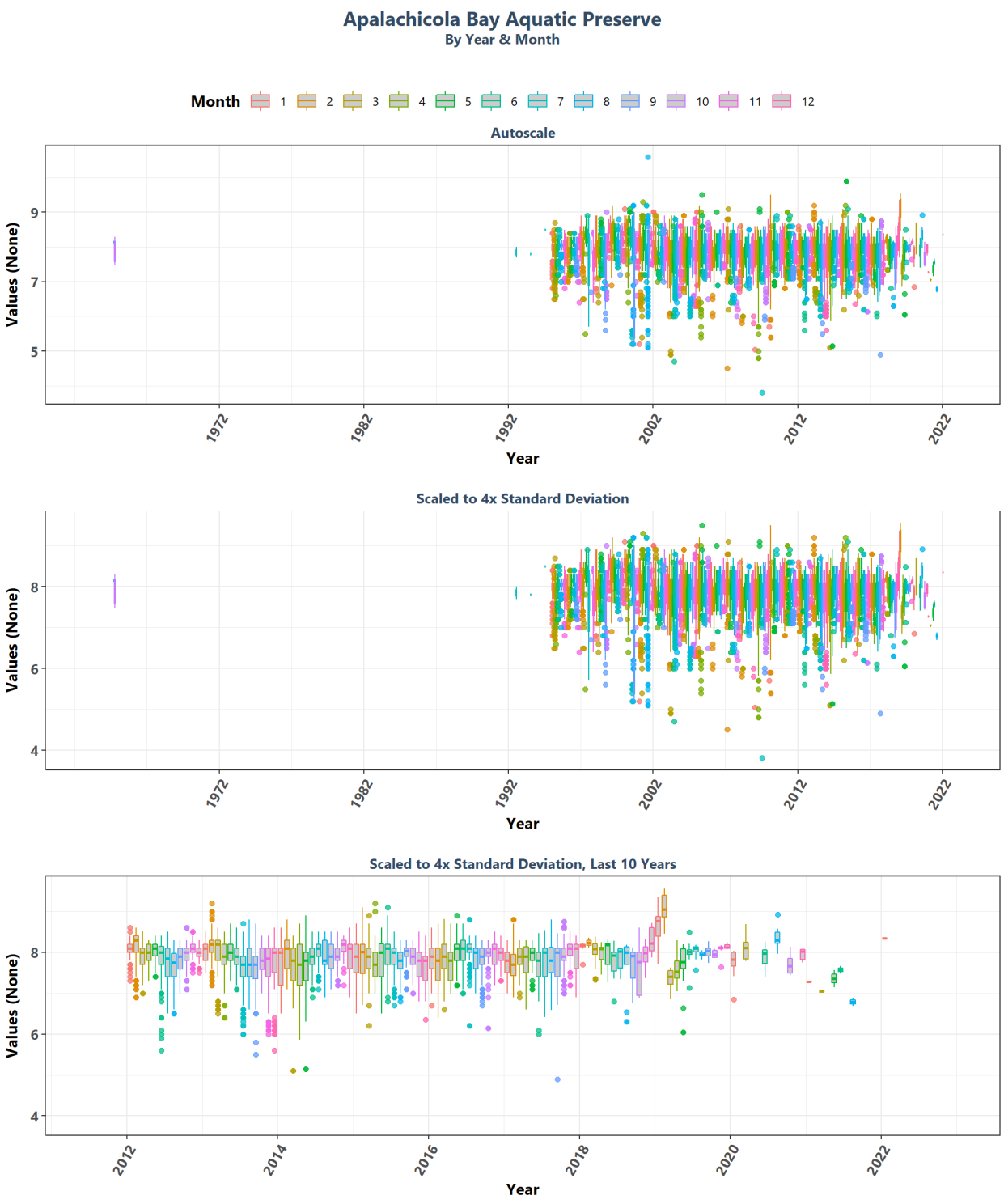
Apalachicola Bay Aquatic Preserve

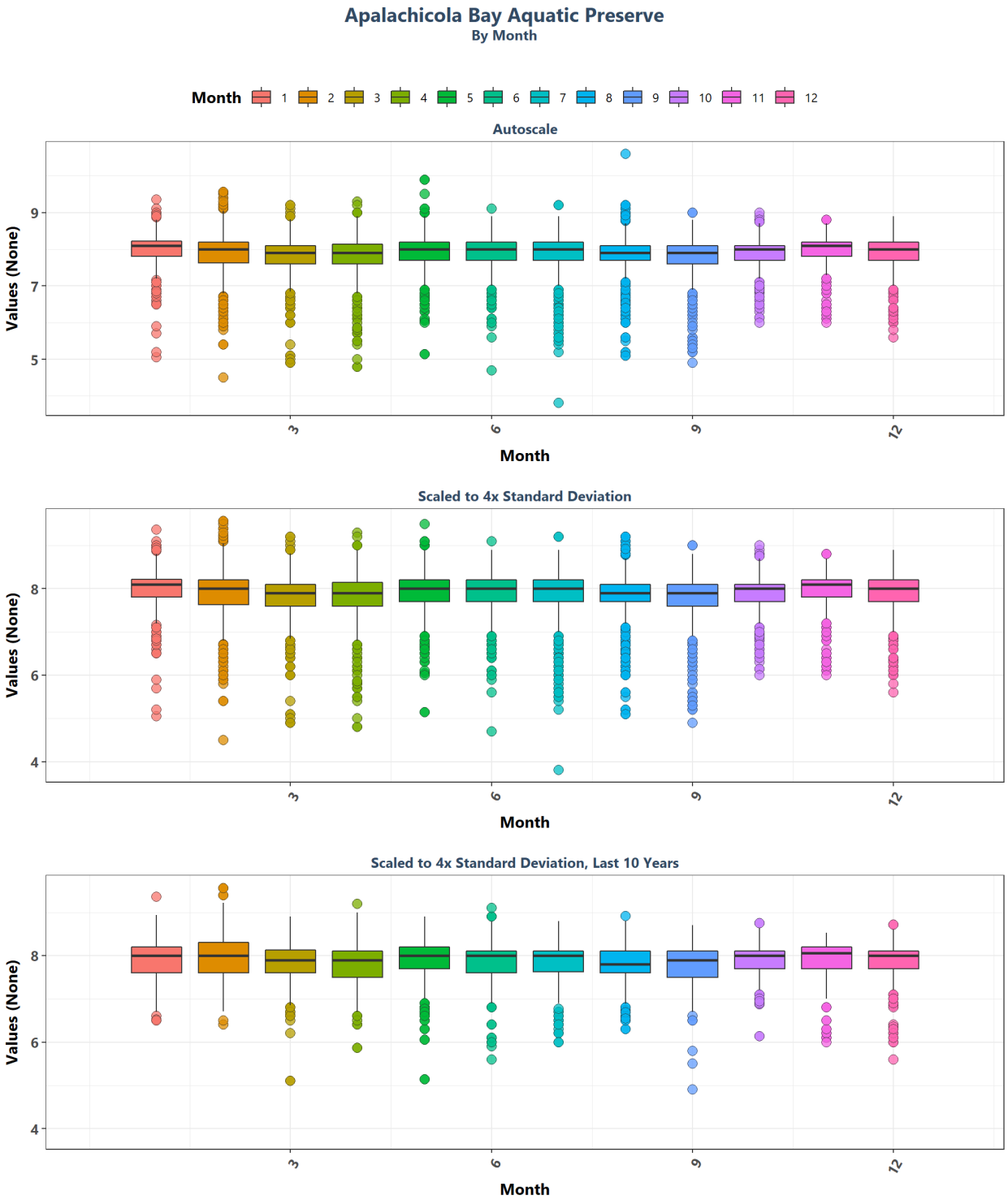
By Year





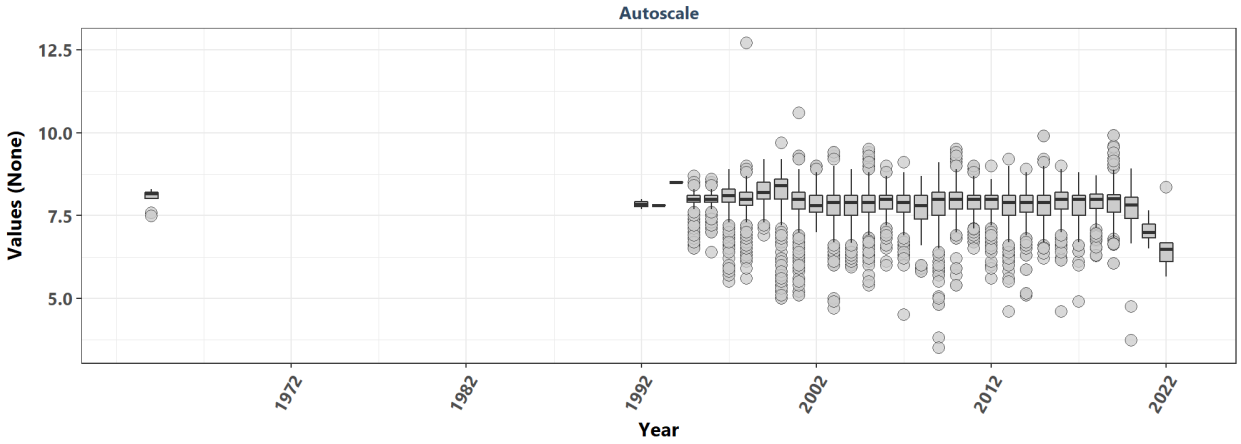


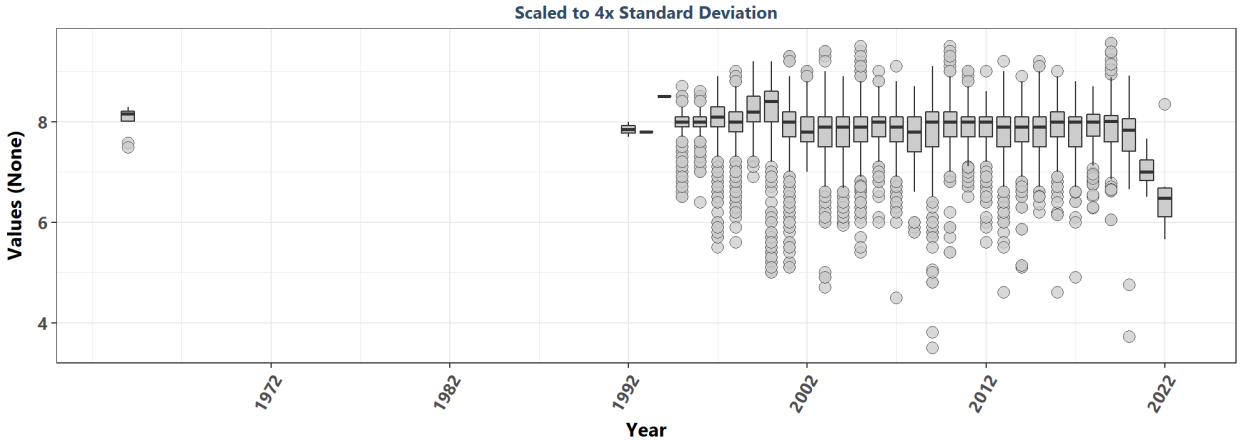

Apalachicola Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

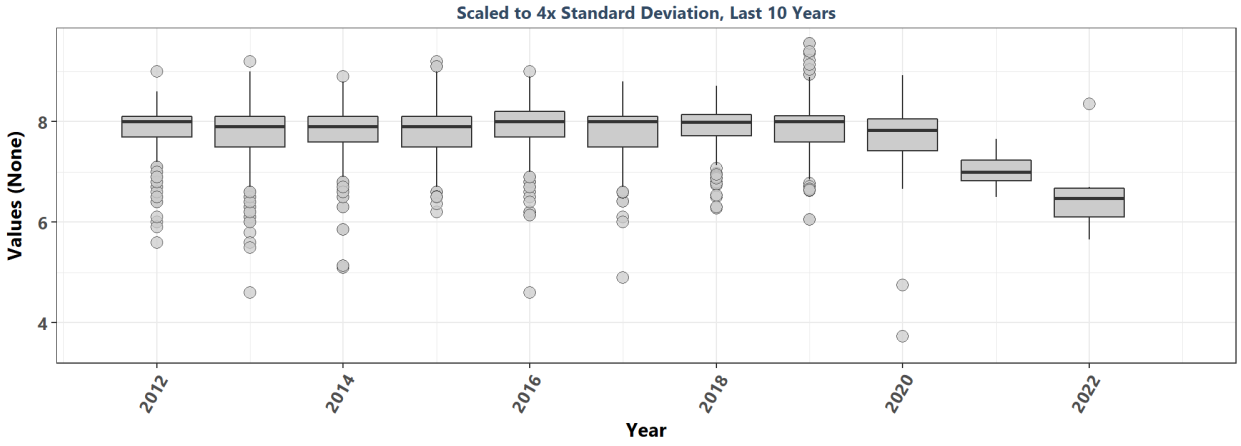

Apalachicola Bay Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
9
7
5
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
8
6
4
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
8
6
4
3
6
9
12
Month

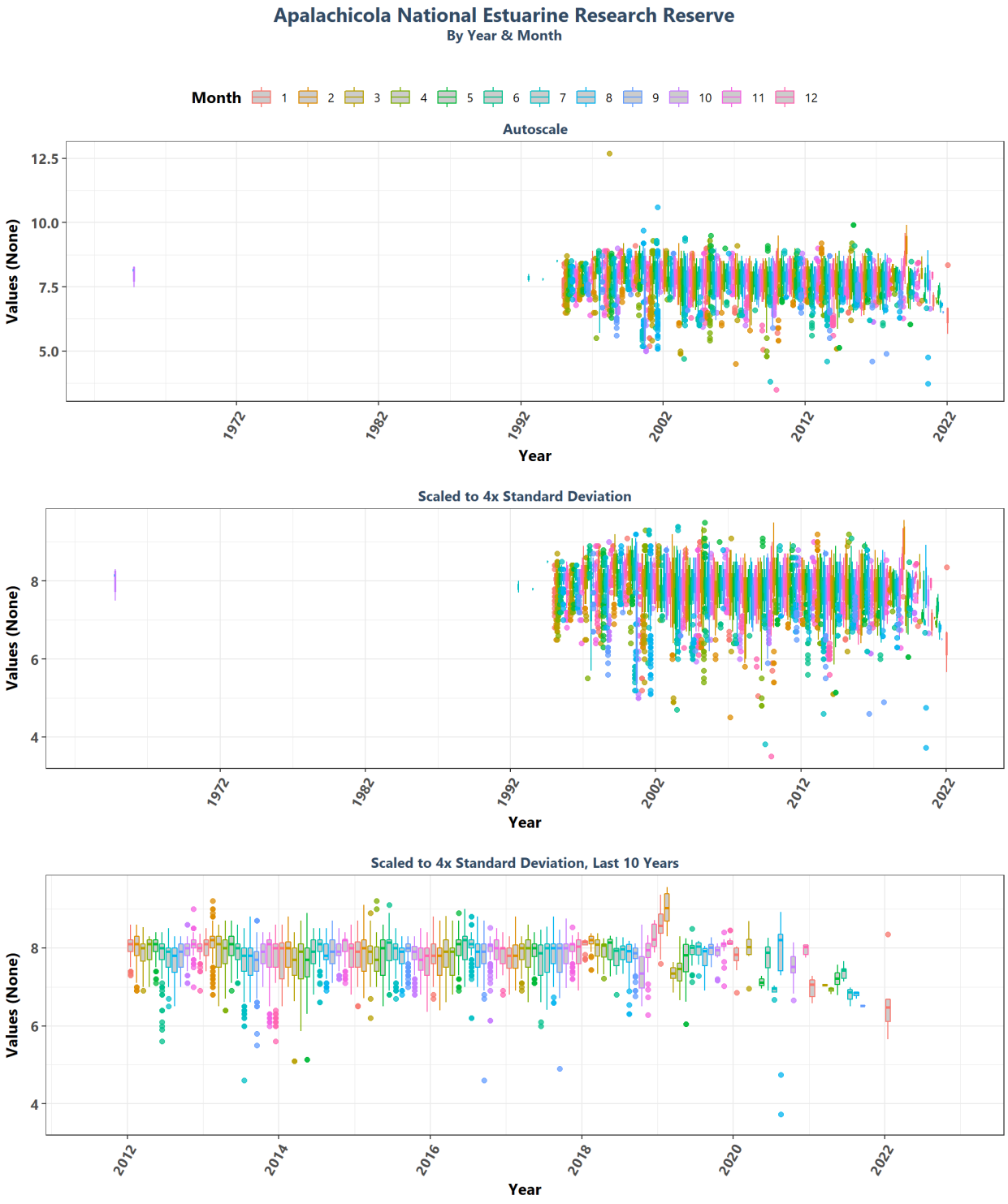
Apalachicola National Estuarine Research Reserve

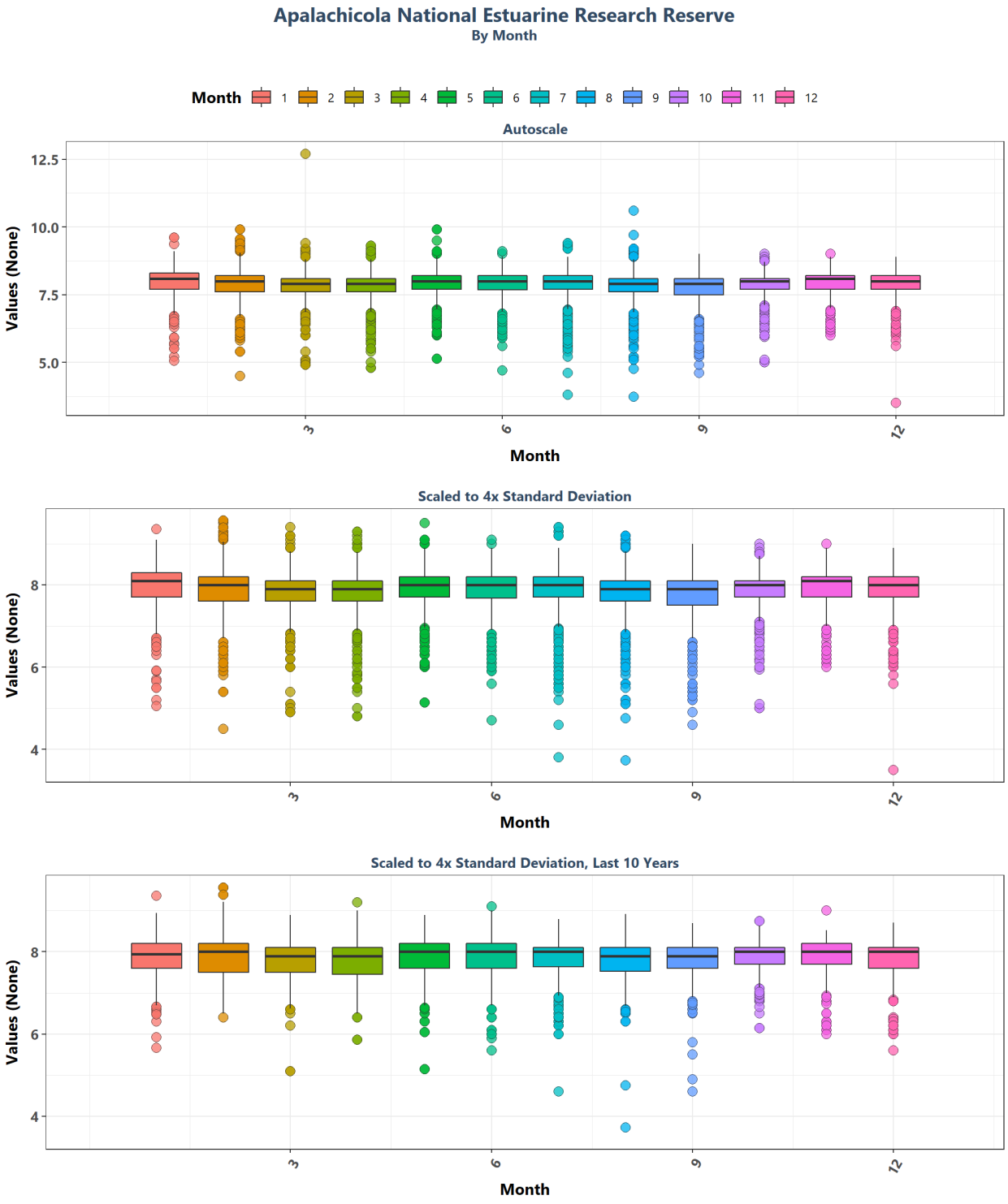
By Year





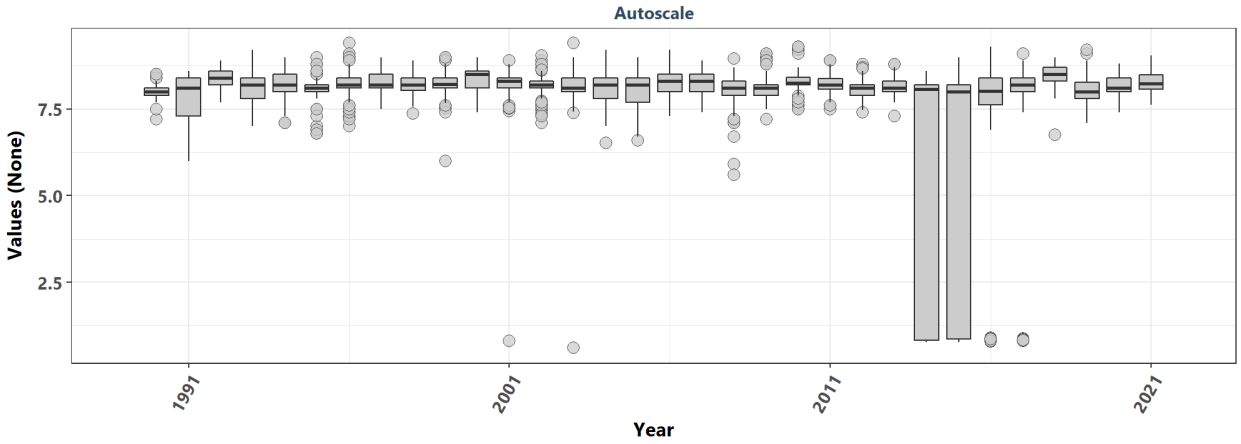


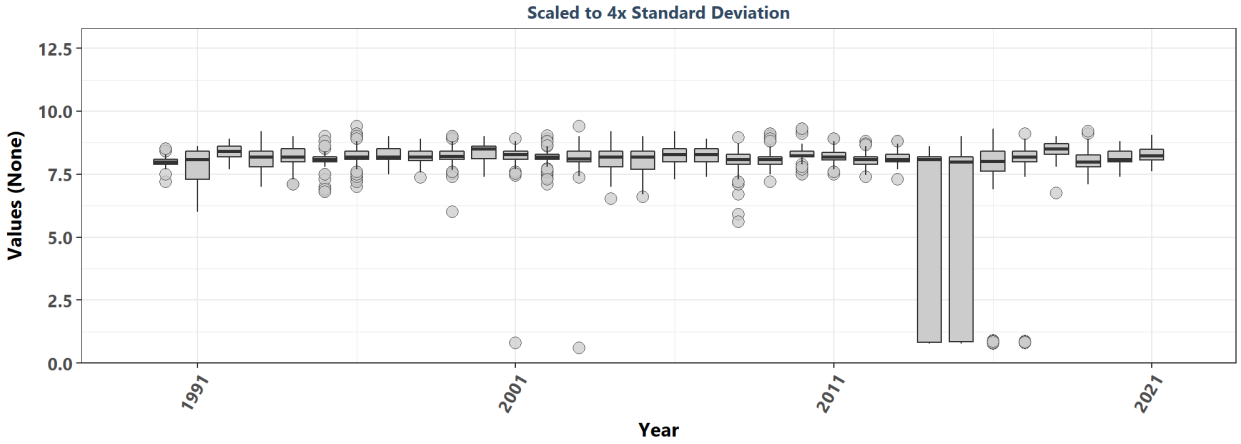

Apalachicola National Estuarine Research Reserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
12.5
10.0
7.5
5.0
1972
1982
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation
Values (None)
8
6
4
1972
1982
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
8
6
4
2012
2014
2016
2018
2020
2022
Year

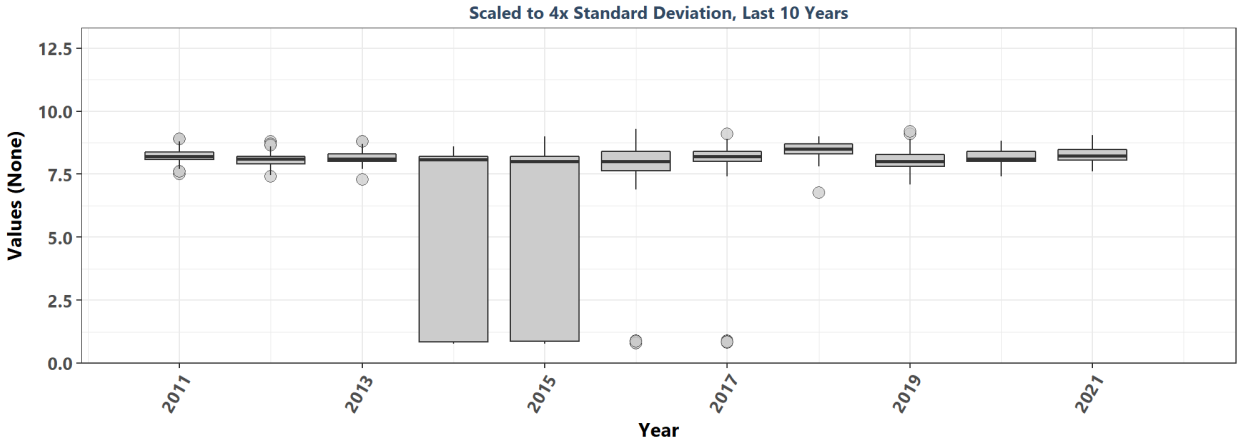

Apalachicola National Estuarine Research Reserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
12.5
10.0
7.5
5.0
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
8
6
4
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
8
6
4
3
6
9
12
Month

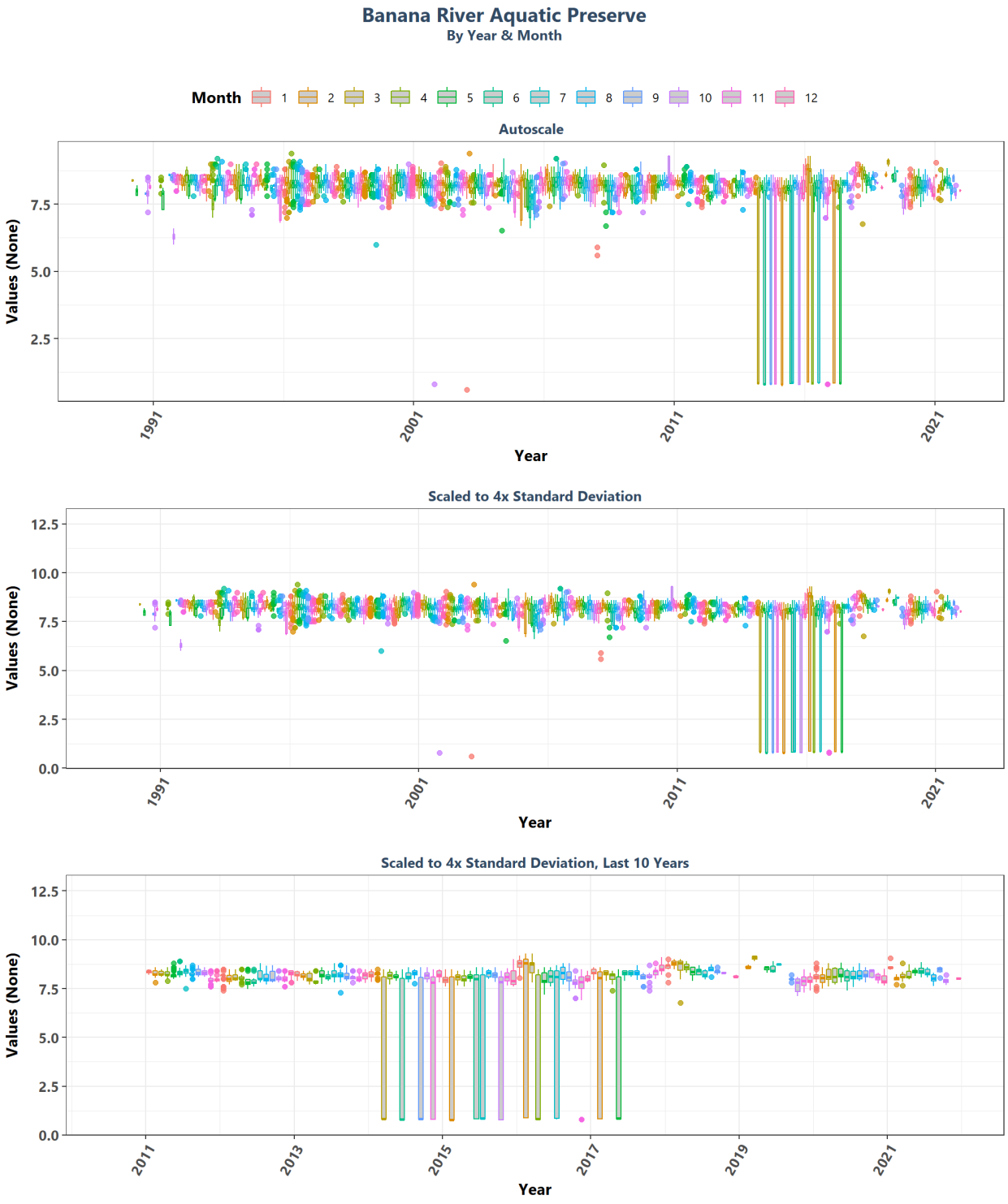
Banana River Aquatic Preserve

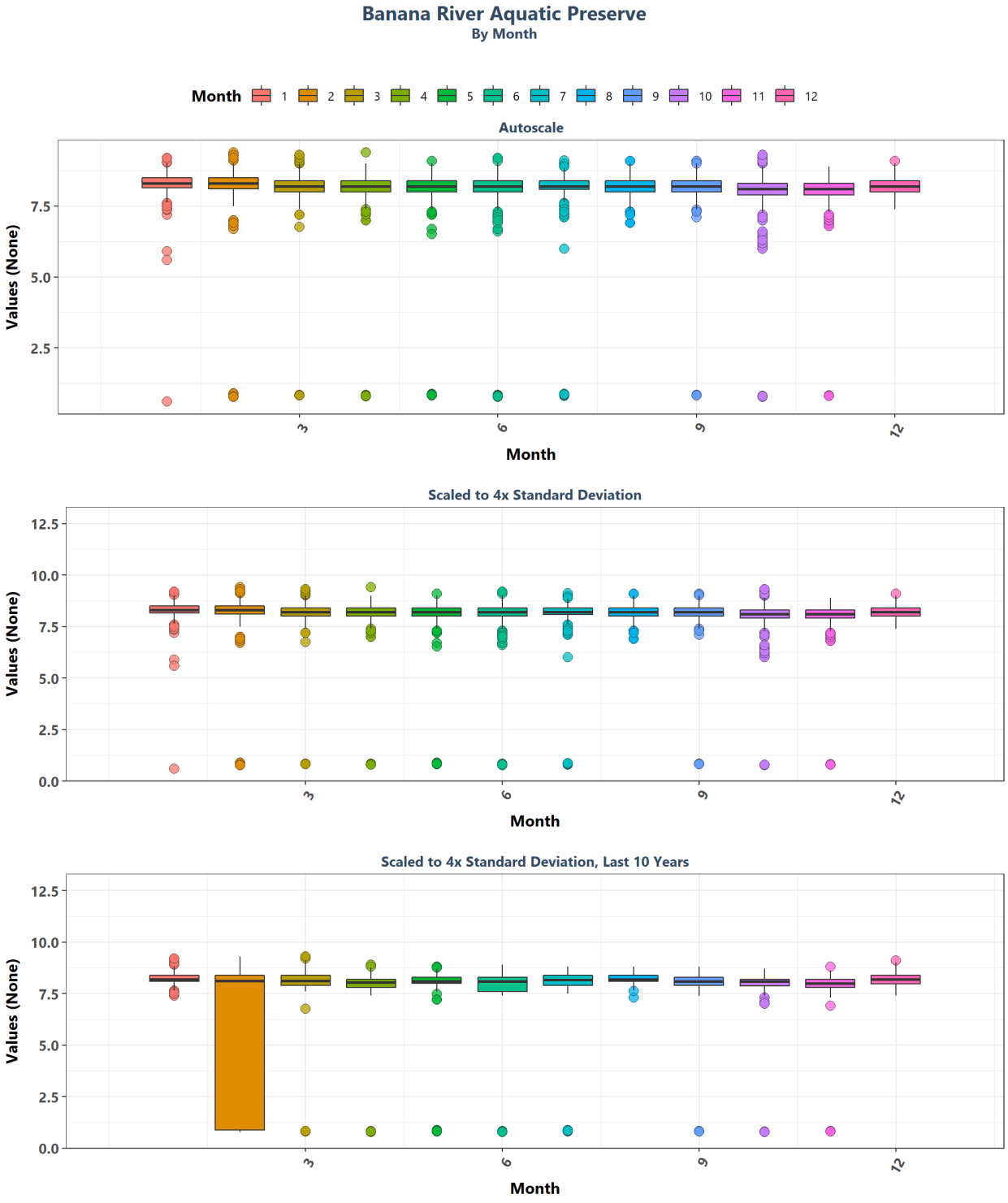
By Year





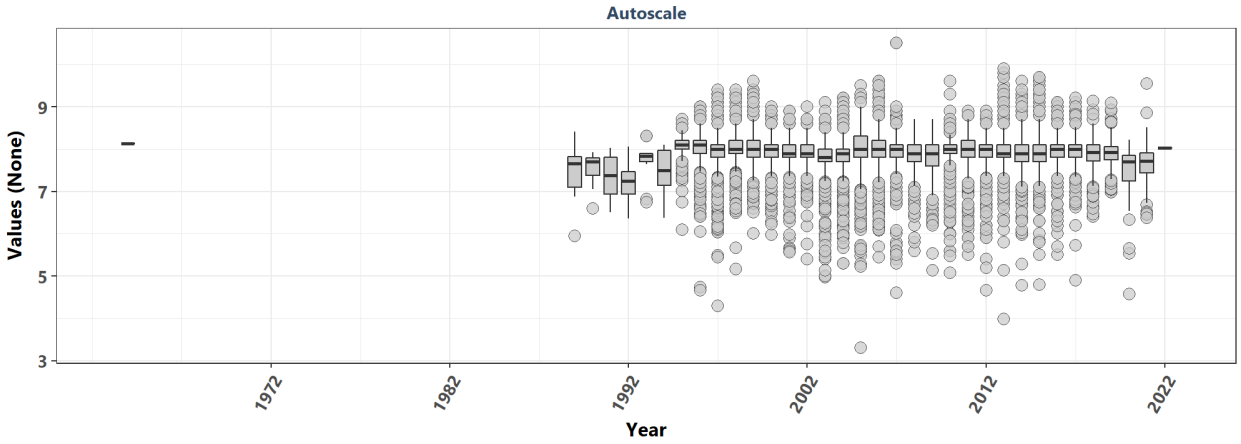


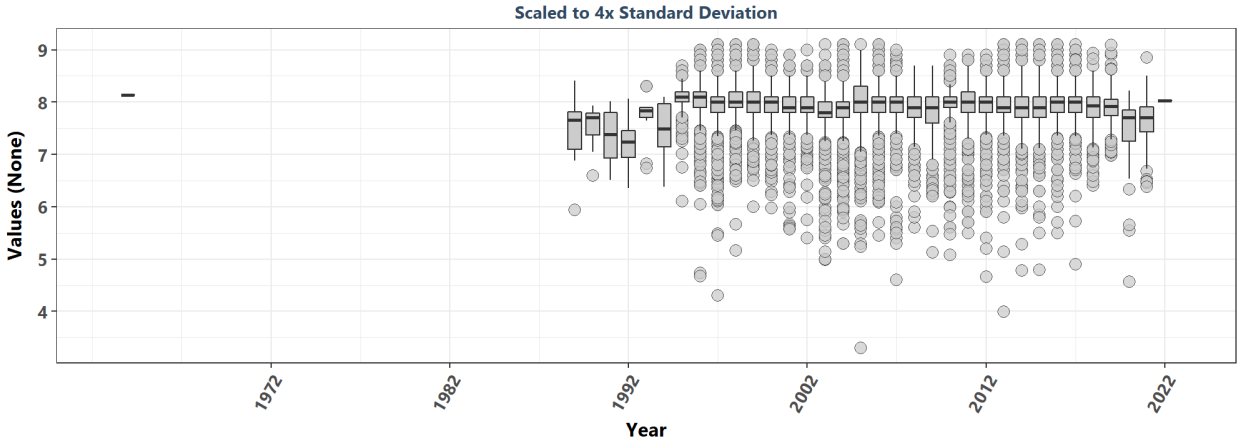

Banana River Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

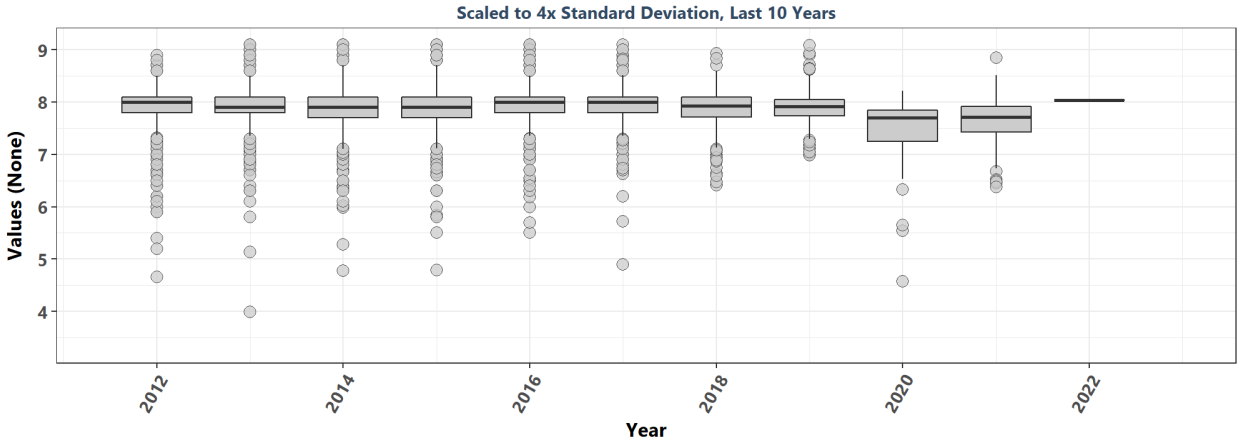

Banana River Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
7.5
5.0
2.5
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
12.5
10.0
7.5
5.0
2.5
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
12.5
10.0
7.5
5.0
2.5
0.0
3
6
9
12
Month

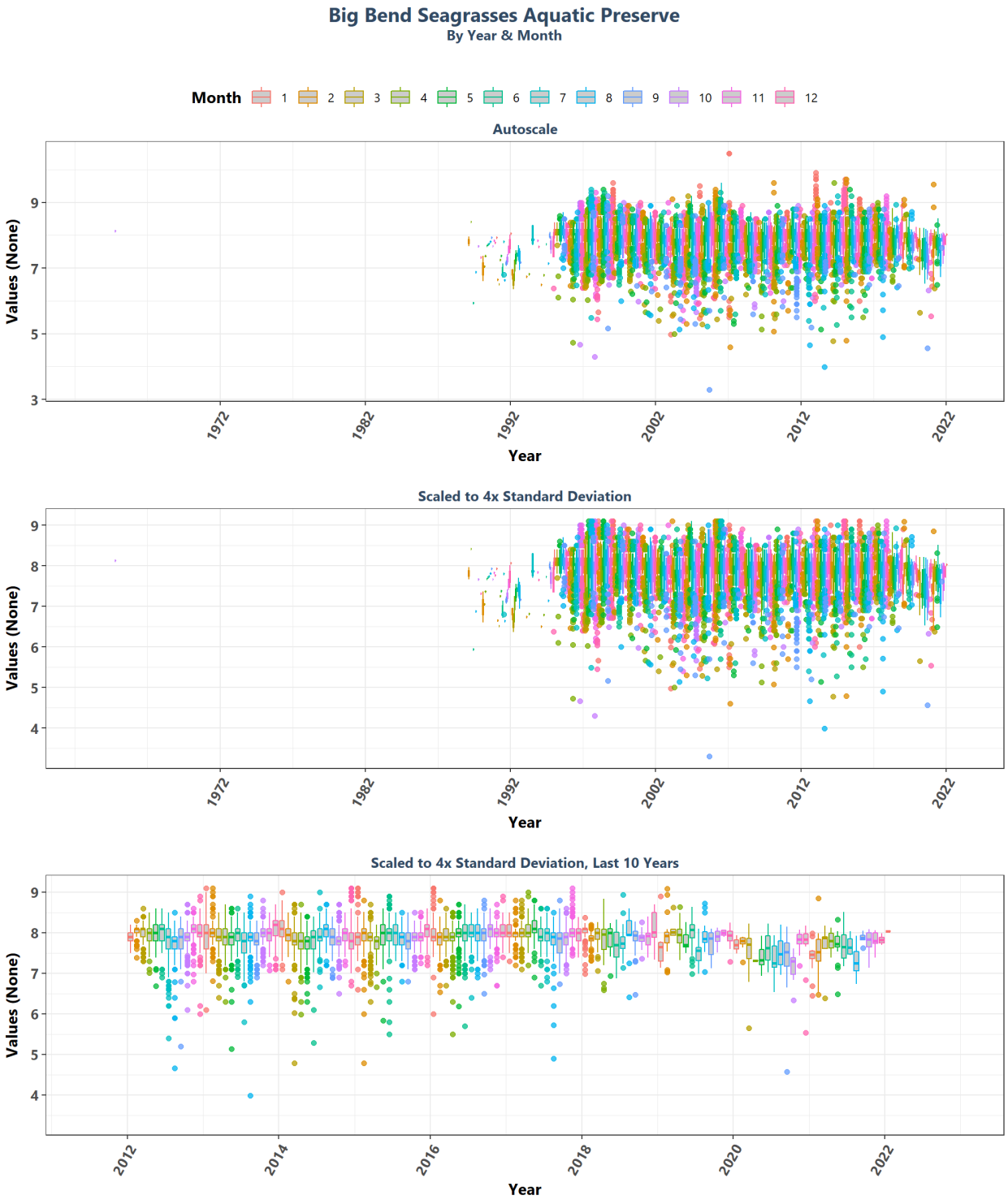
Big Bend Seagrasses Aquatic Preserve

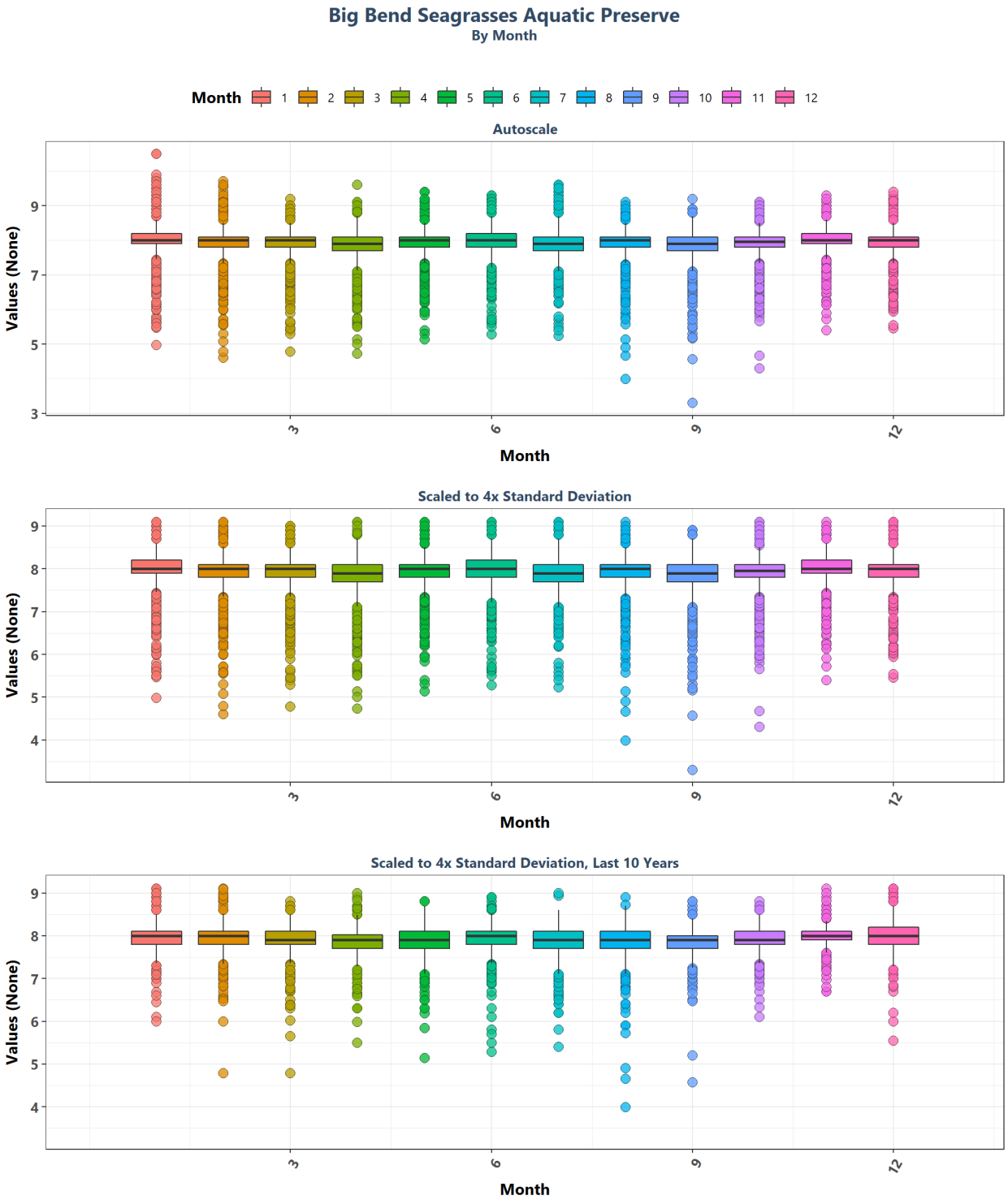
By Year





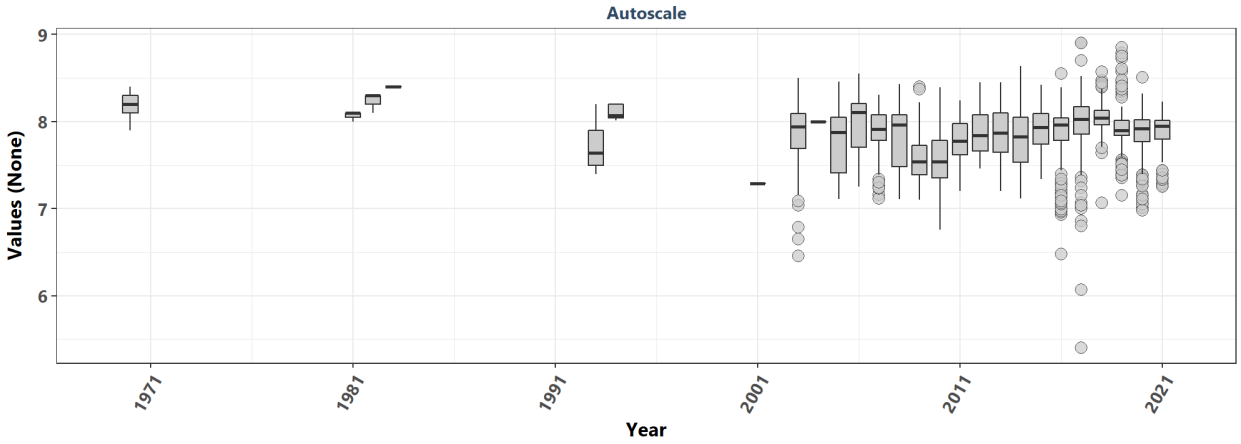


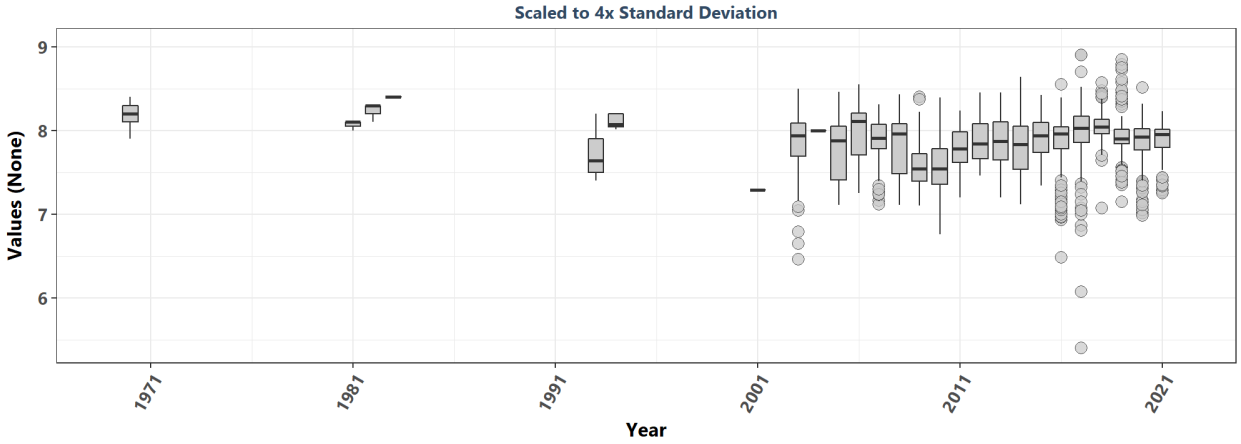

Big Bend Seagrasses Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

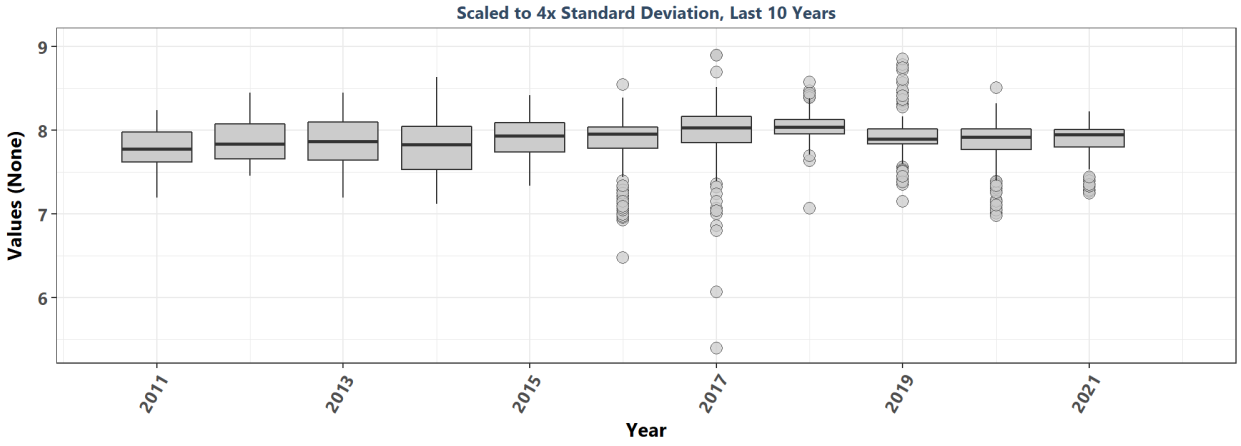

Big Bend Seagrasses Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

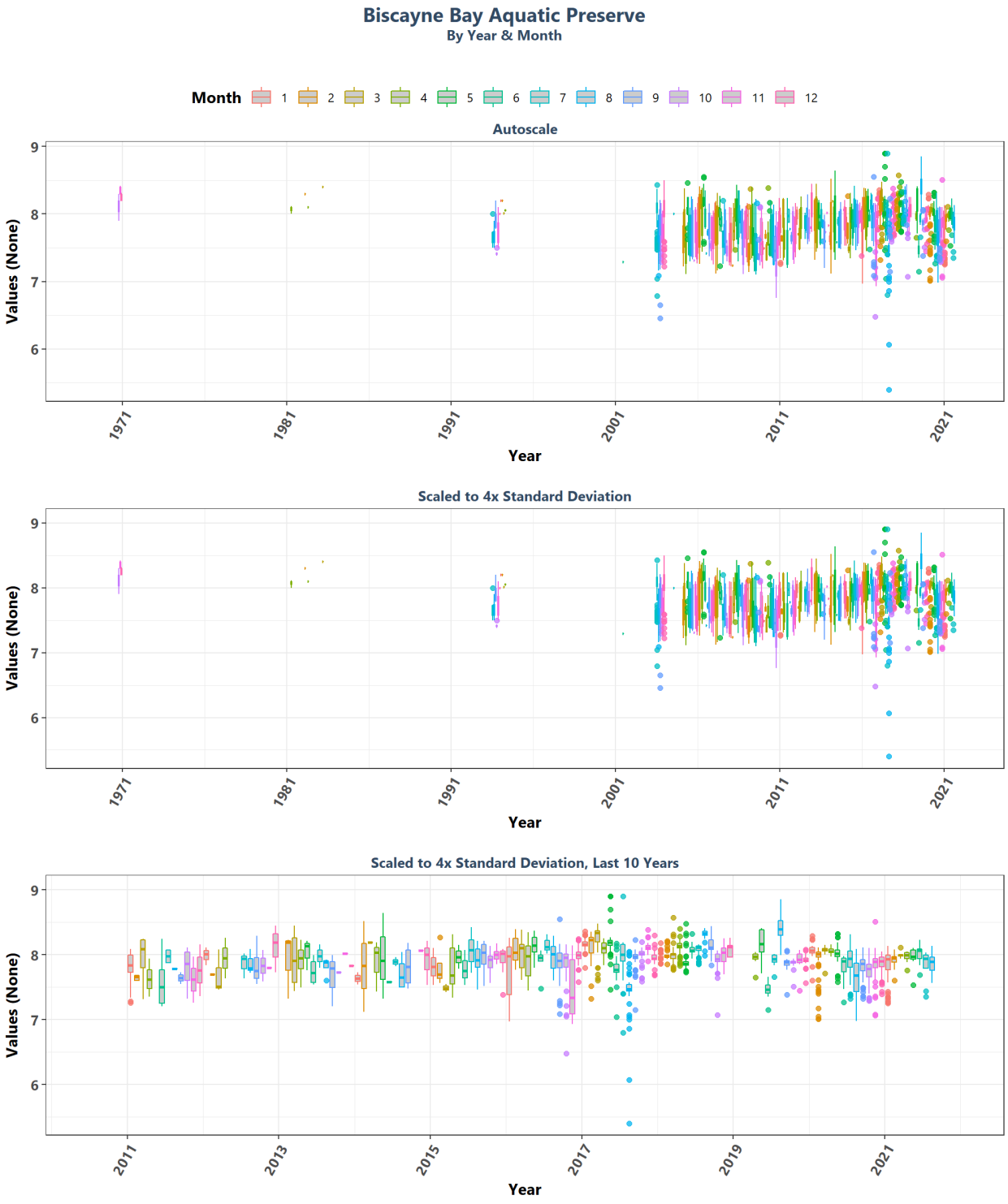
Biscayne Bay Aquatic Preserve

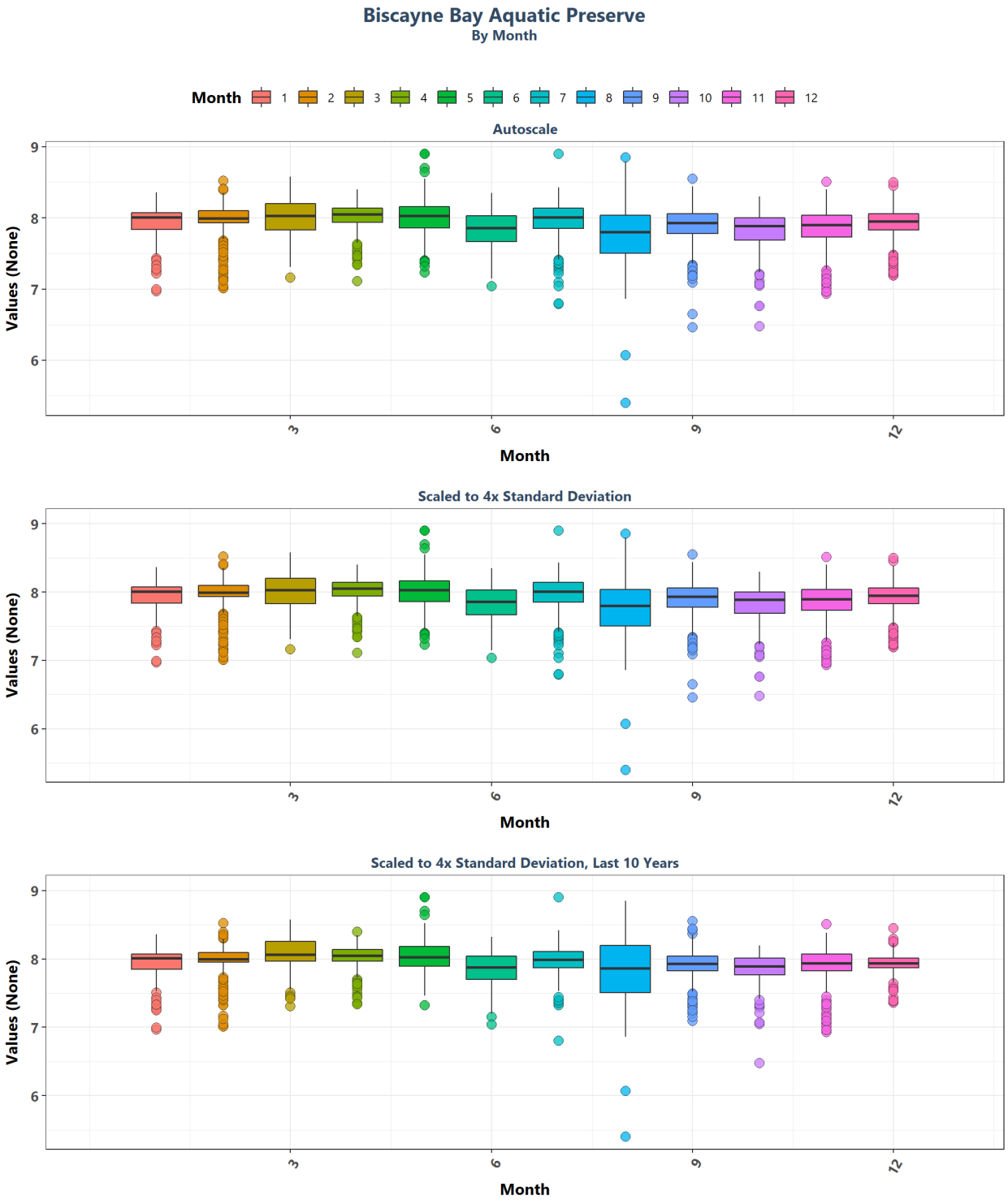
By Year

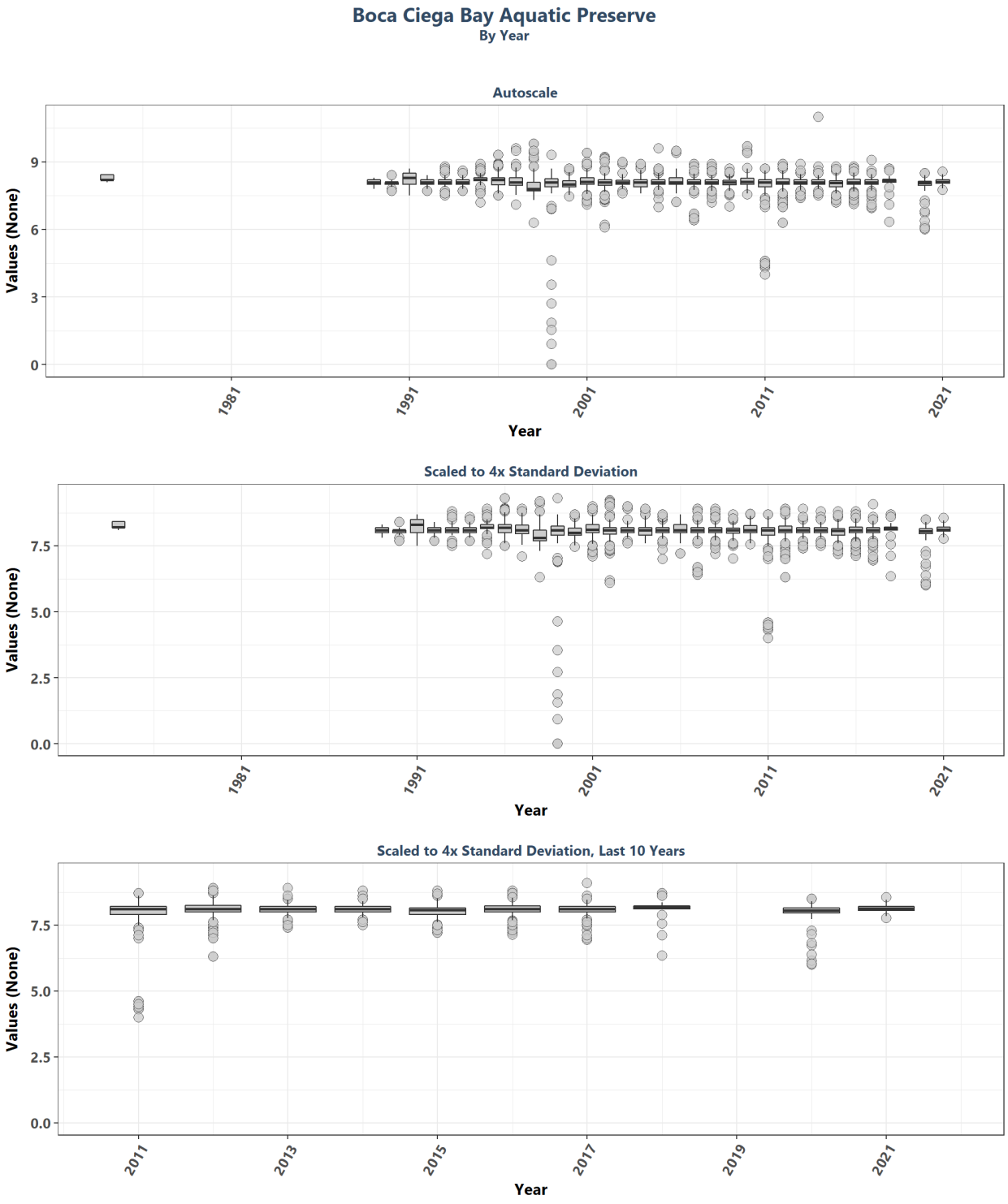


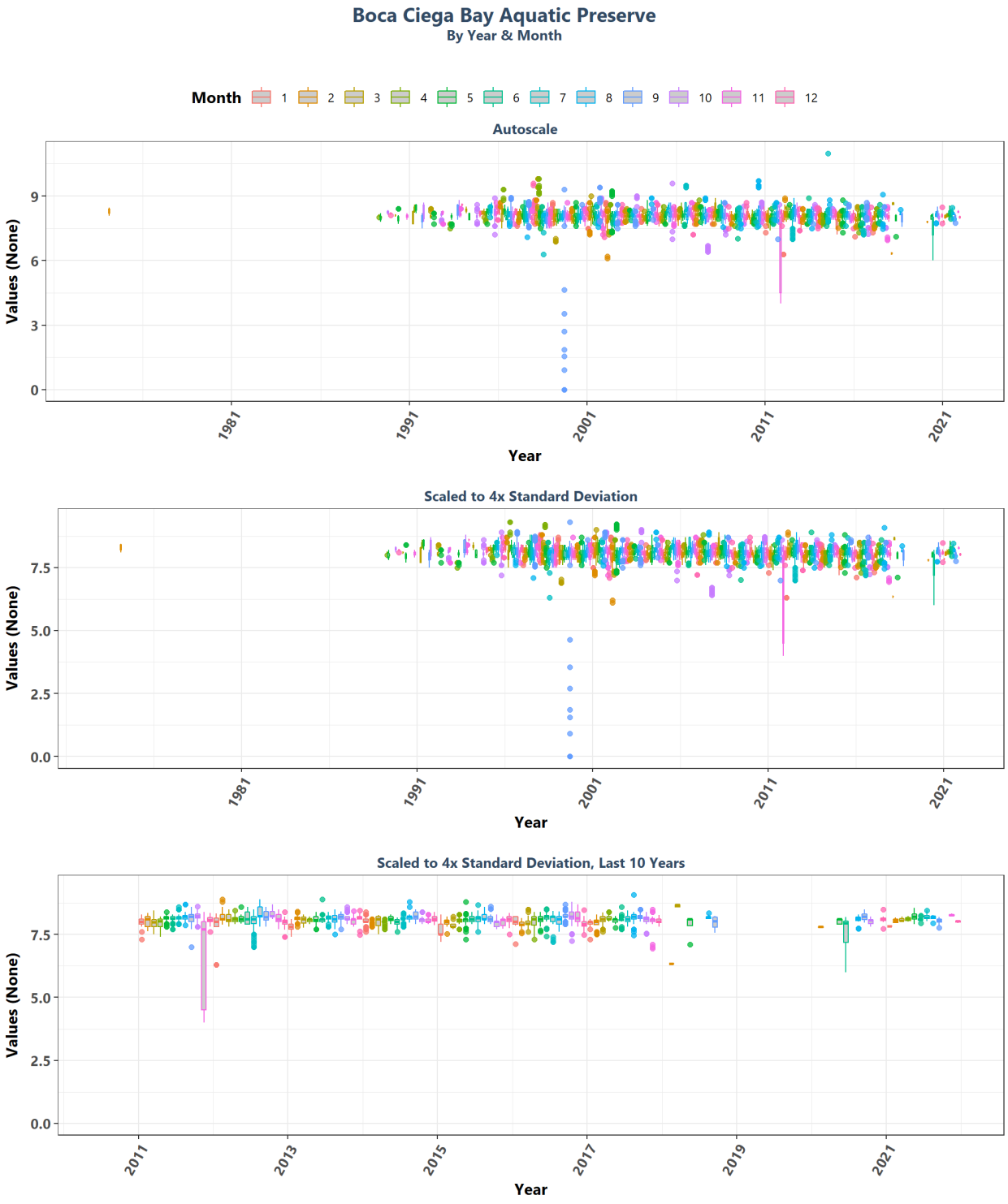


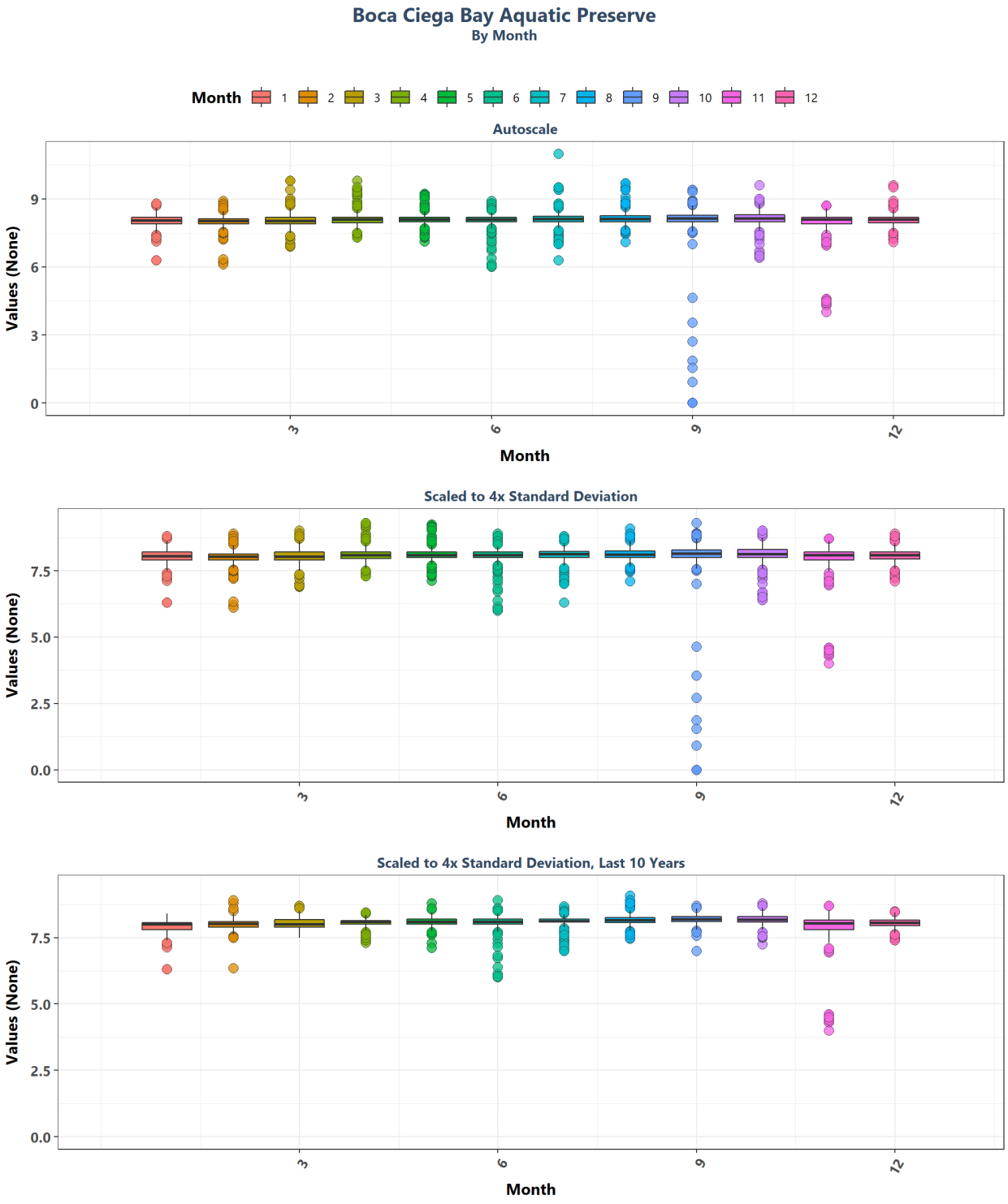



Biscayne Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year


Biscayne Bay Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

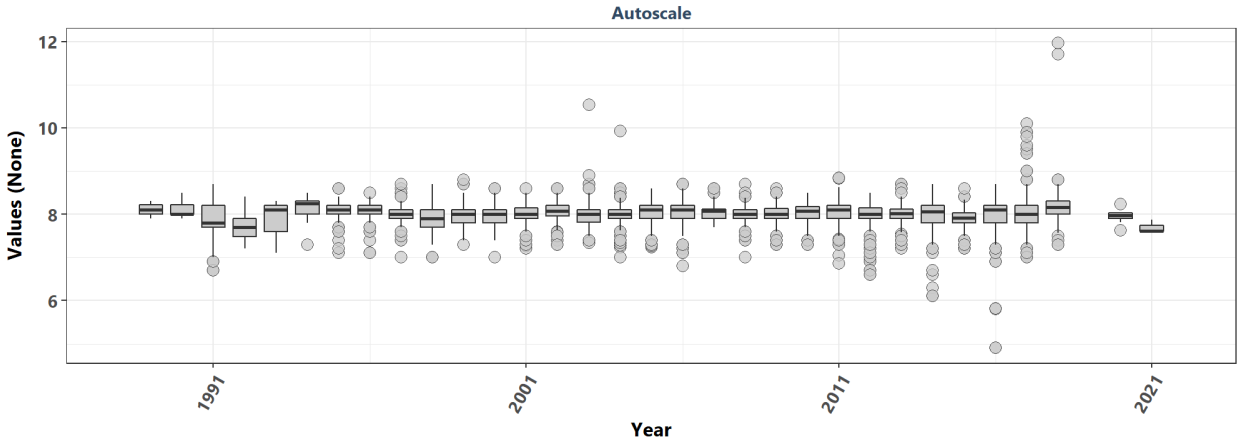

Boca Ciega Bay Aquatic Preserve
By Year
Autoscale
Values (None)
0
3
6
9
1981
1991
2001
2011
2021
Year
Scaled to 4x Standard Deviation
Values (None)
0.0
2.5
5.0
7.5
1981
1991
2001
2011
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
0.0
2.5
5.0
7.5
2011
2013
2015
2017
2019
2021
Year

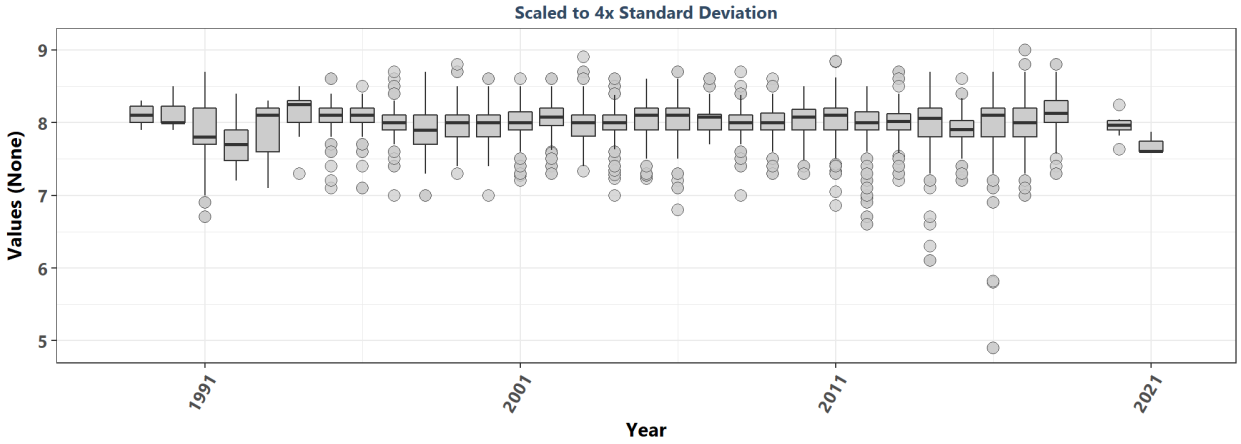

Boca Ciega Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

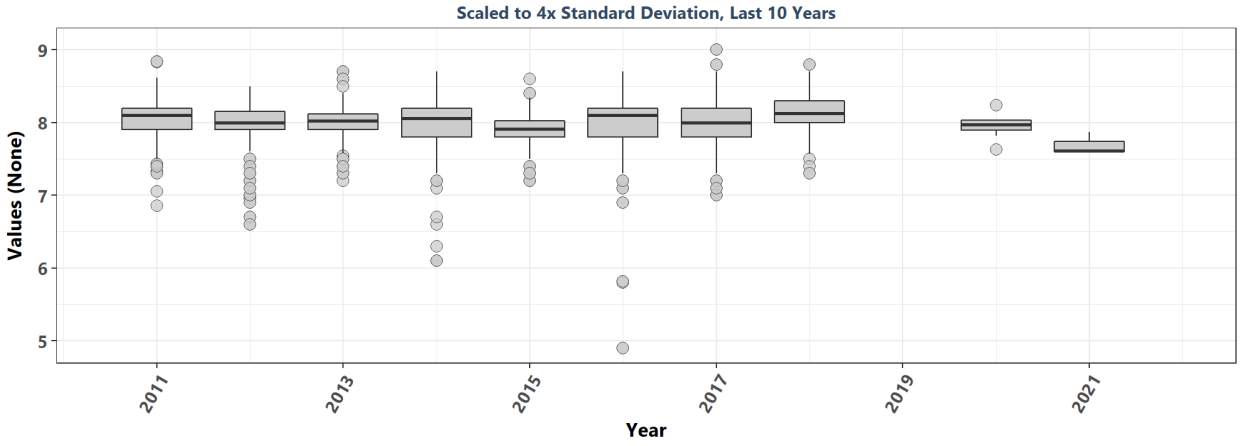

Boca Ciega Bay Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
0
3
6
9
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
0.0
2.5
5.0
7.5
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
0.0
2.5
5.0
7.5
3
6
9
12
Month

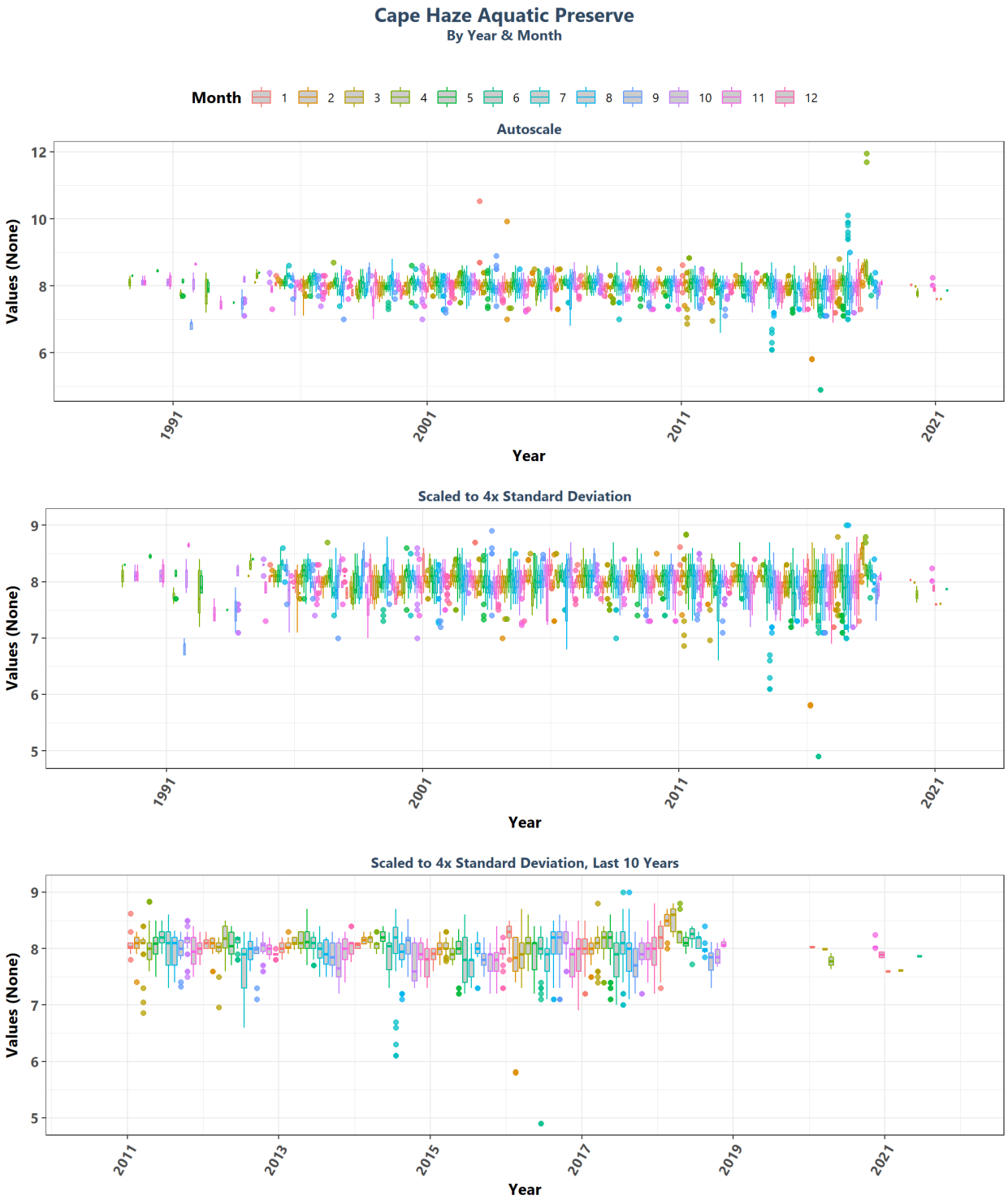
Cape Haze Aquatic Preserve

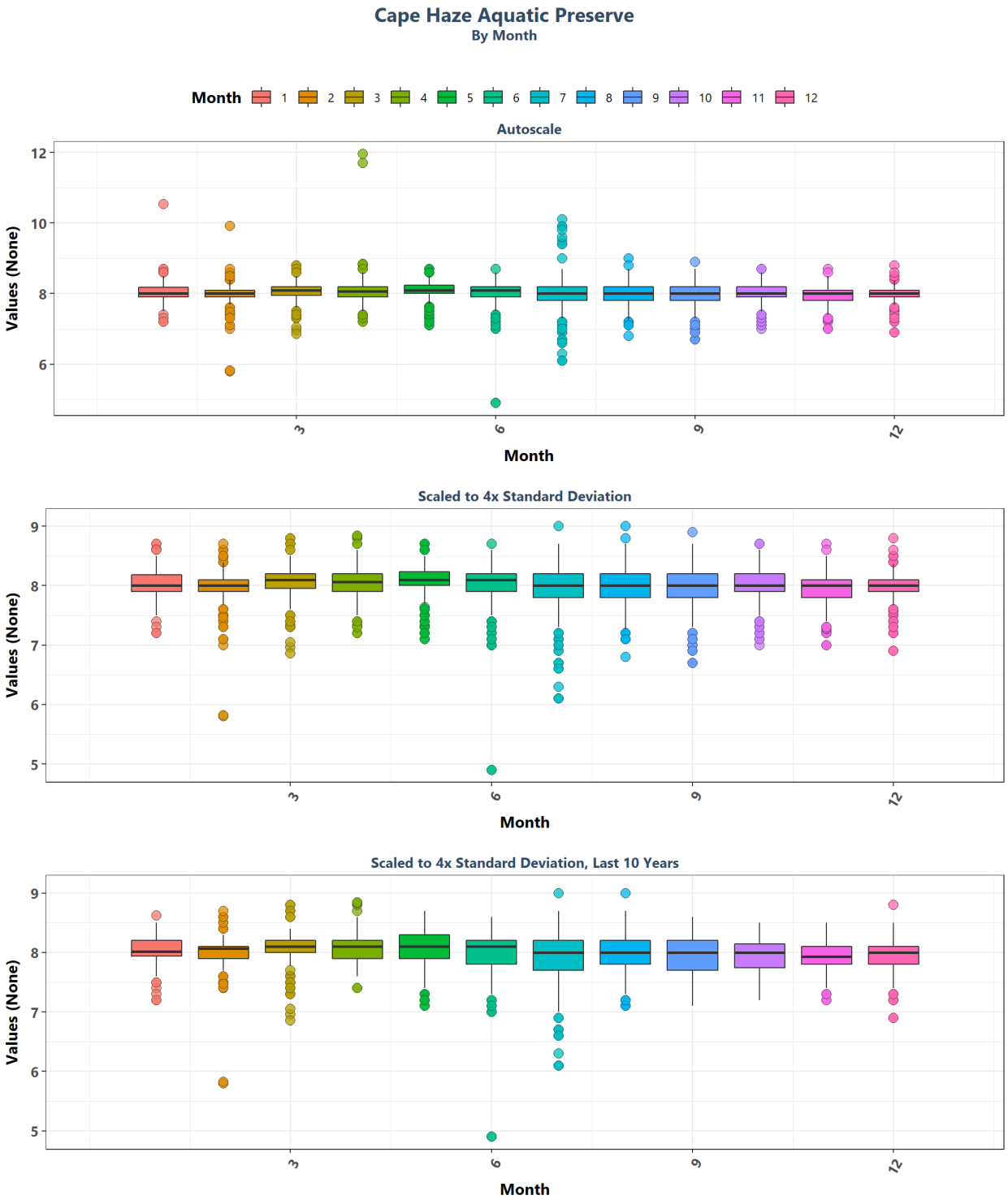
By Year

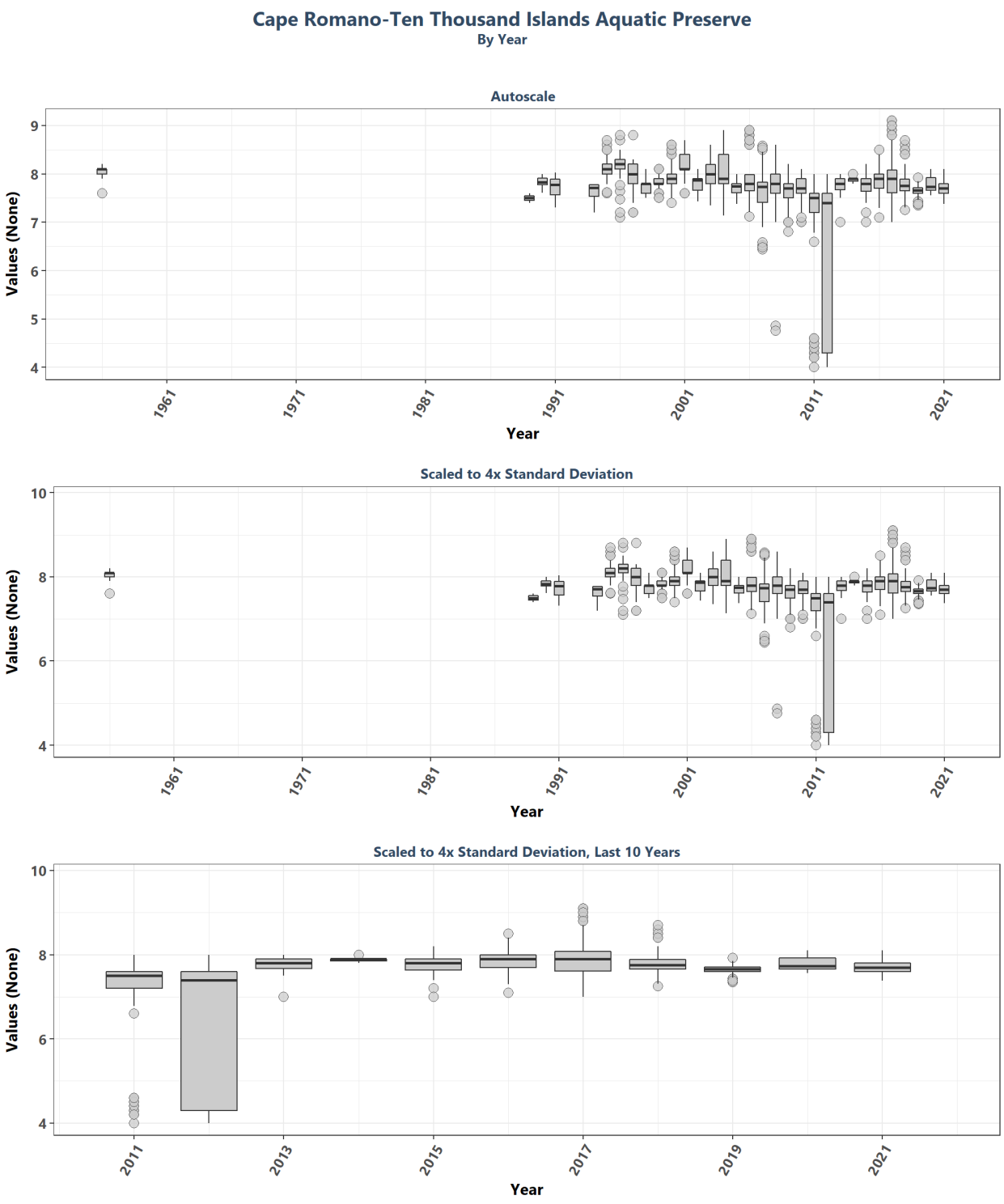


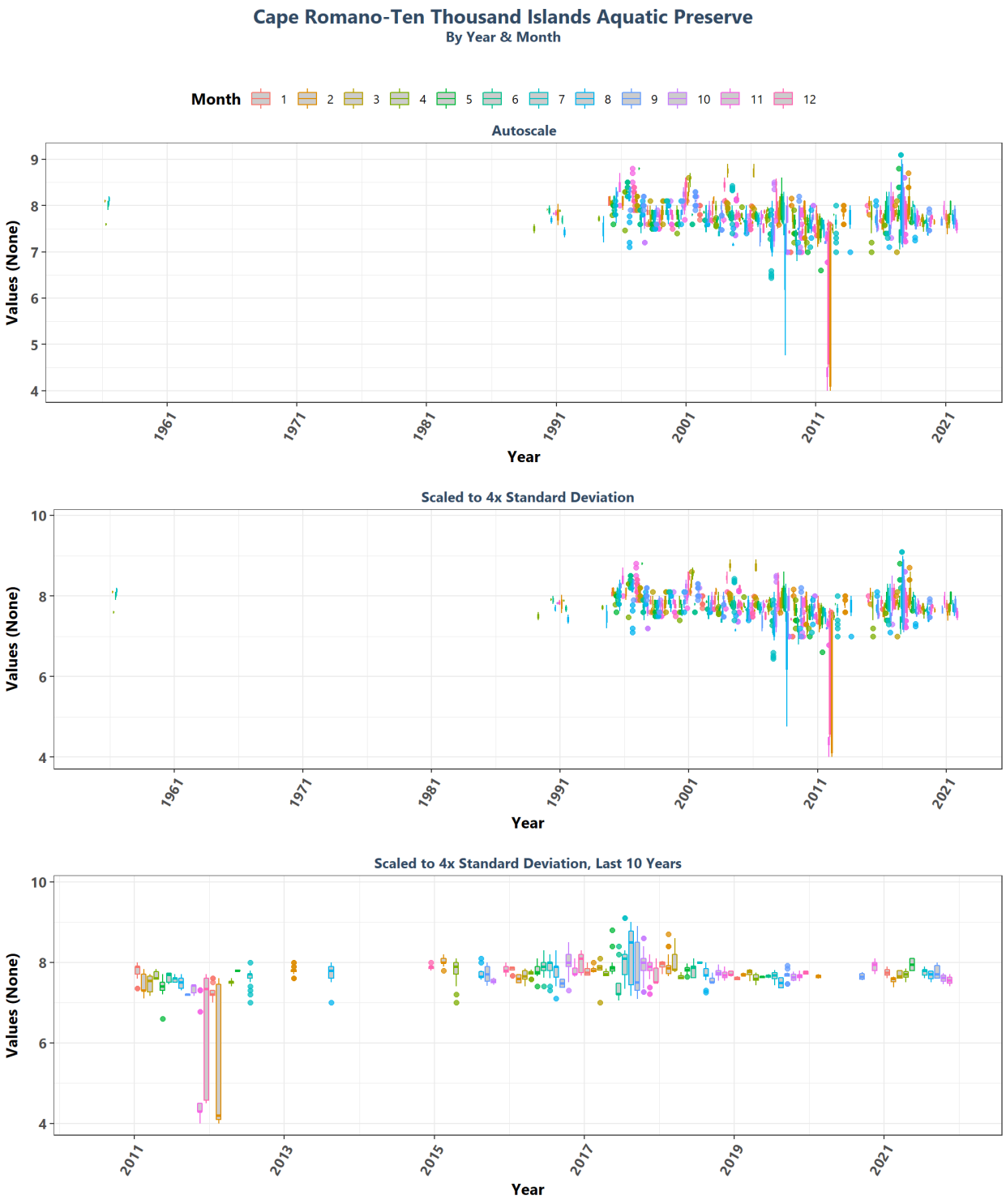


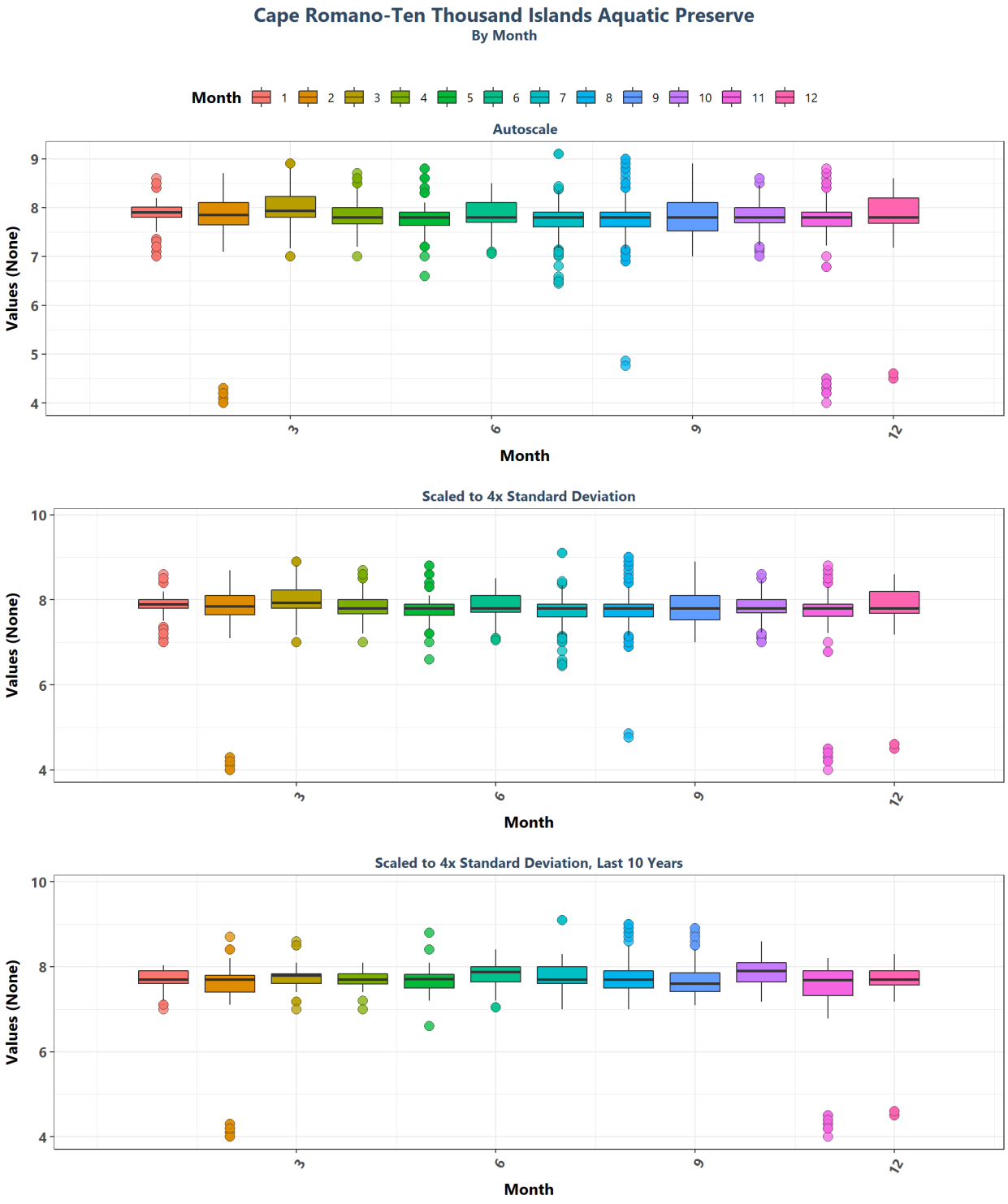


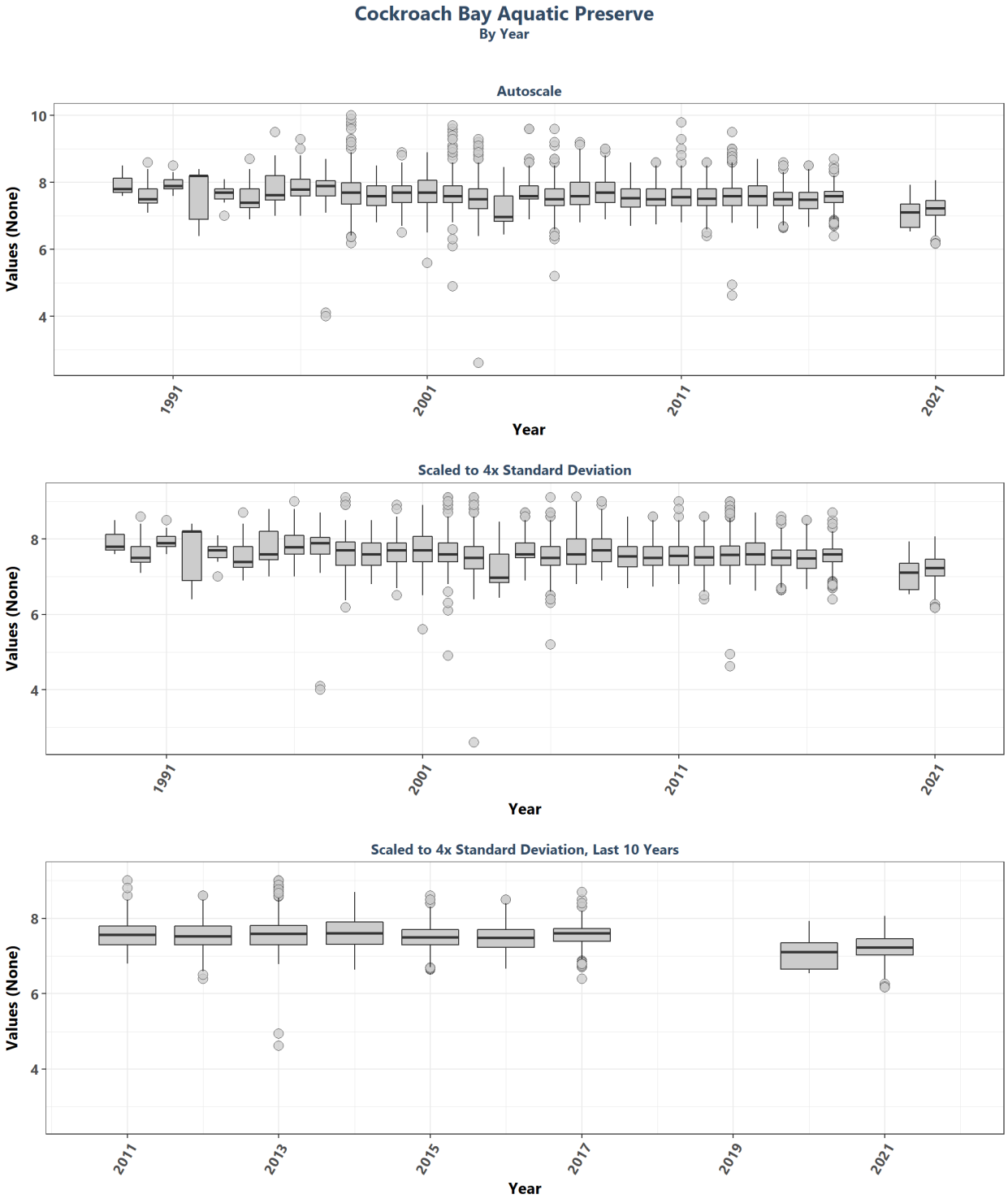

Cape Haze Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

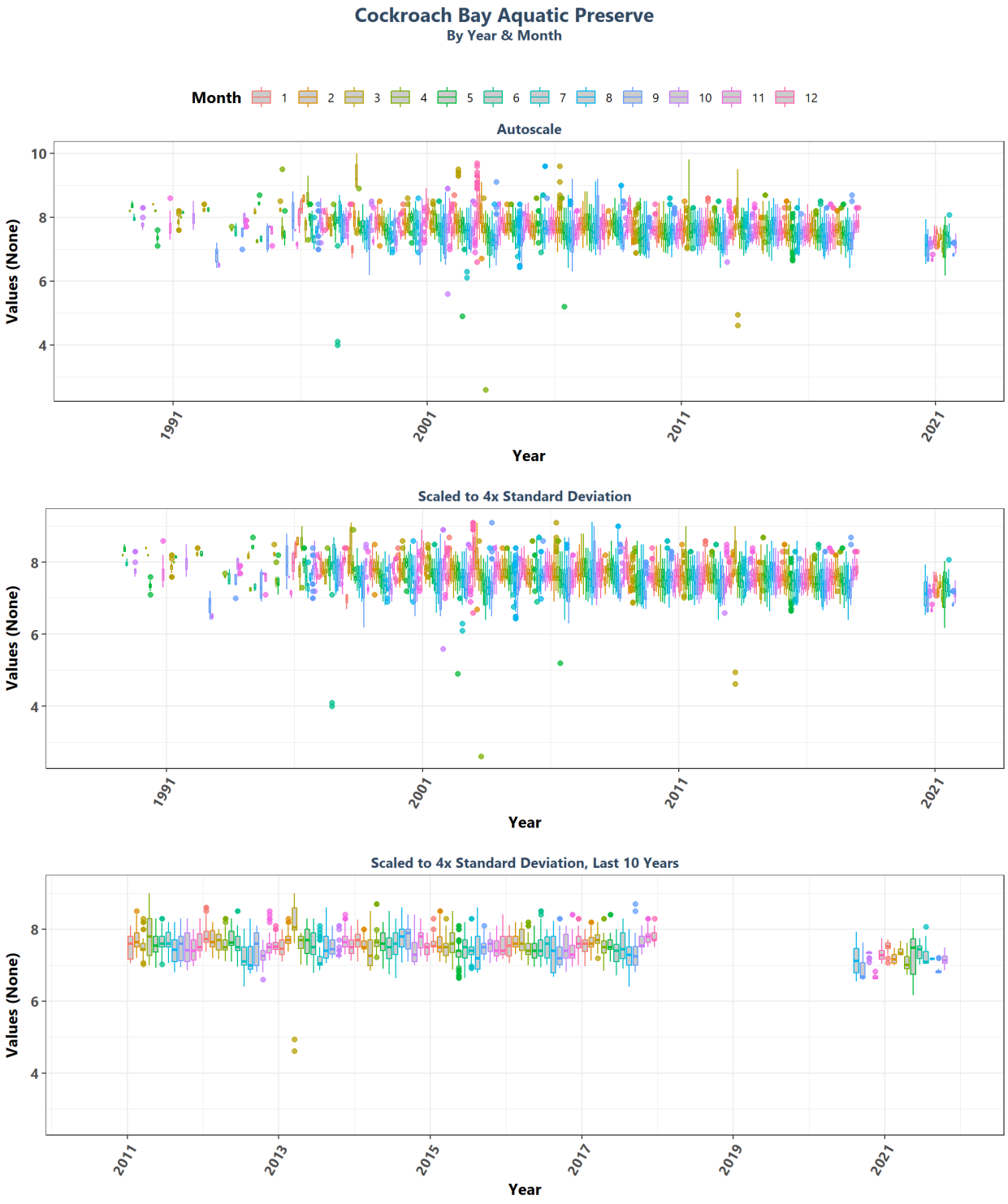

Cape Haze Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

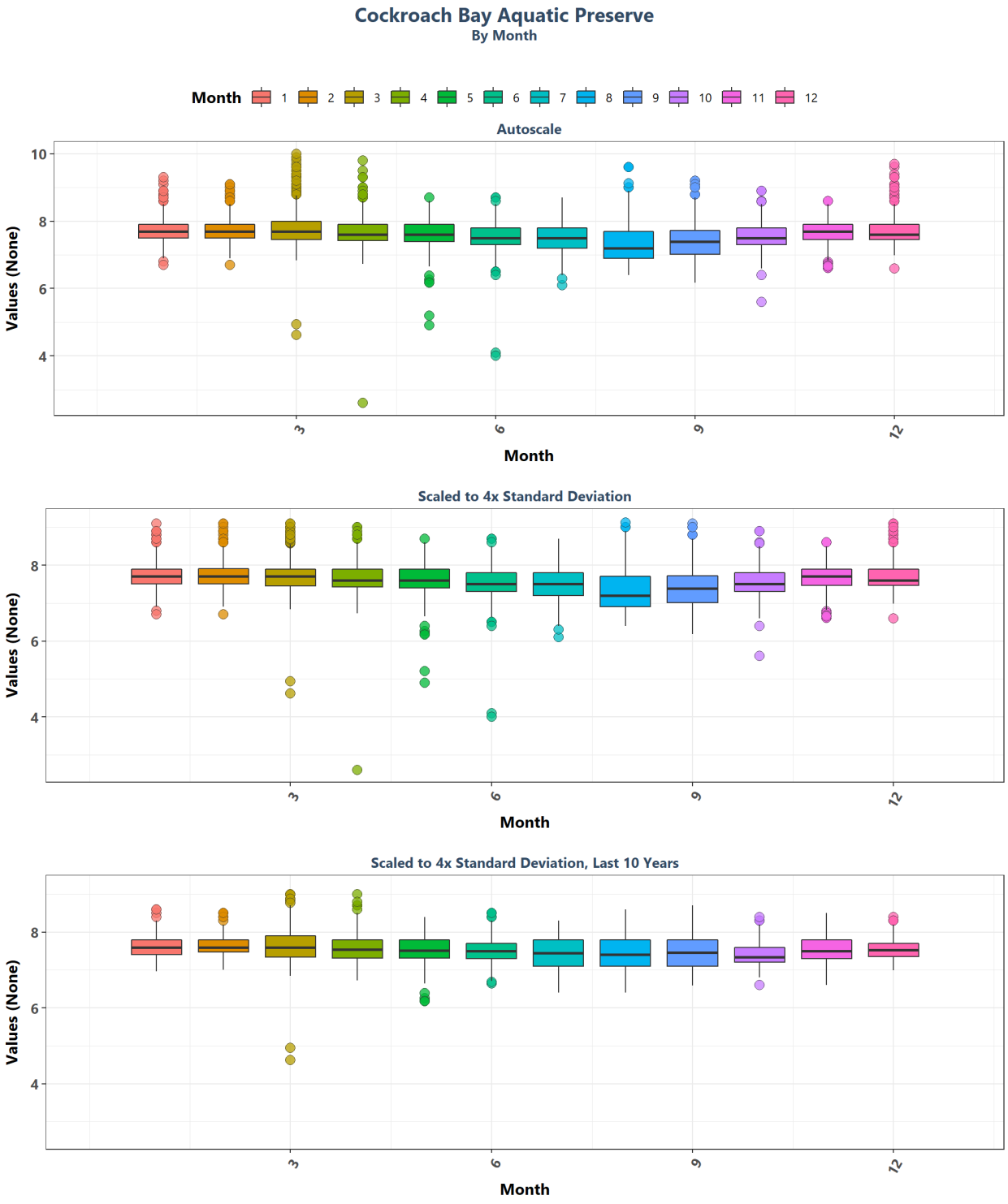

Cape Romano-Ten Thousand Islands Aquatic Preserve
By Year
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year


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 (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

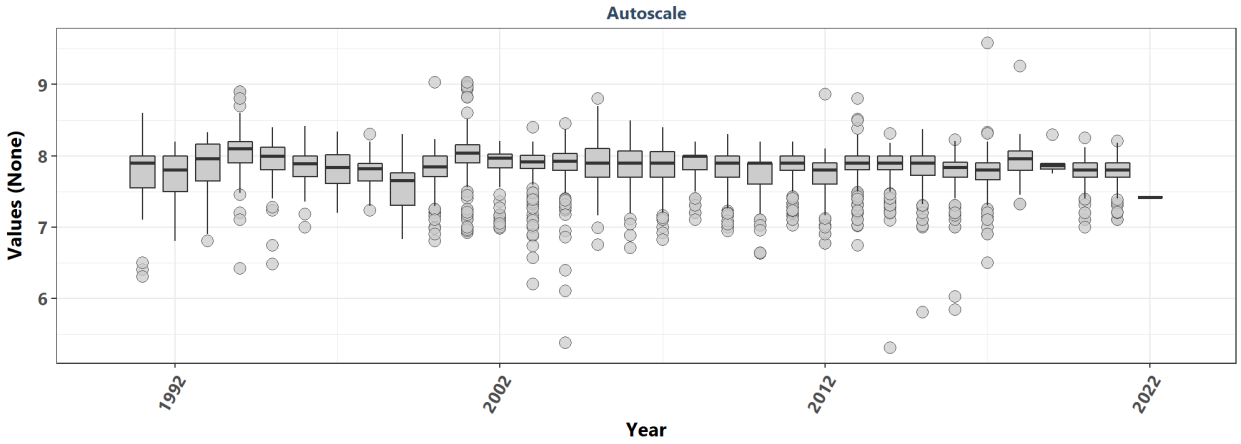

Cape Romano-Ten Thousand Islands Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

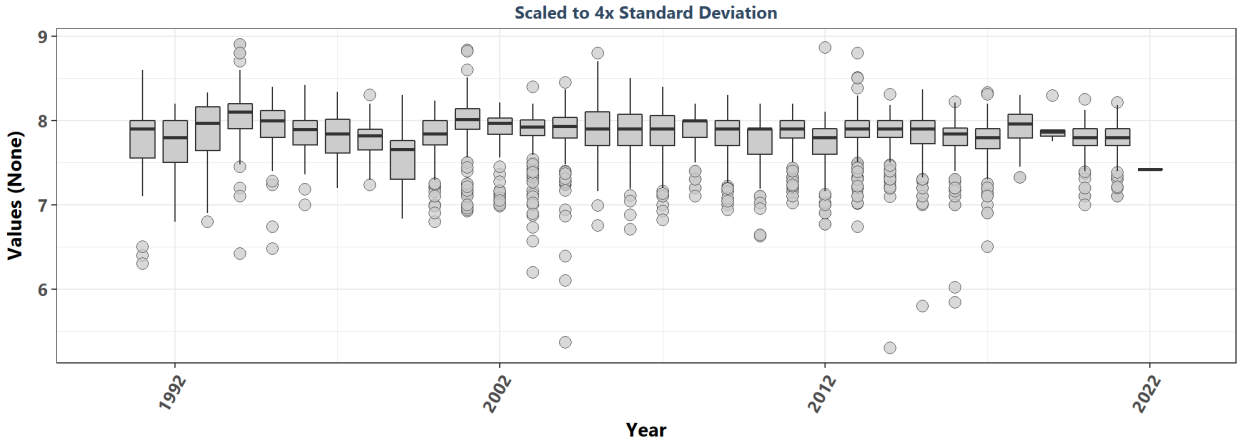

Cockroach Bay Aquatic Preserve
By Year
Autoscale
Values (None)
10
8
6
4
1991
2001
2011
2021
Year
Scaled to 4x Standard Deviation
Values (None)
8
6
4
1991
2001
2011
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
8
6
4
2011
2013
2015
2017
2019
2021
Year

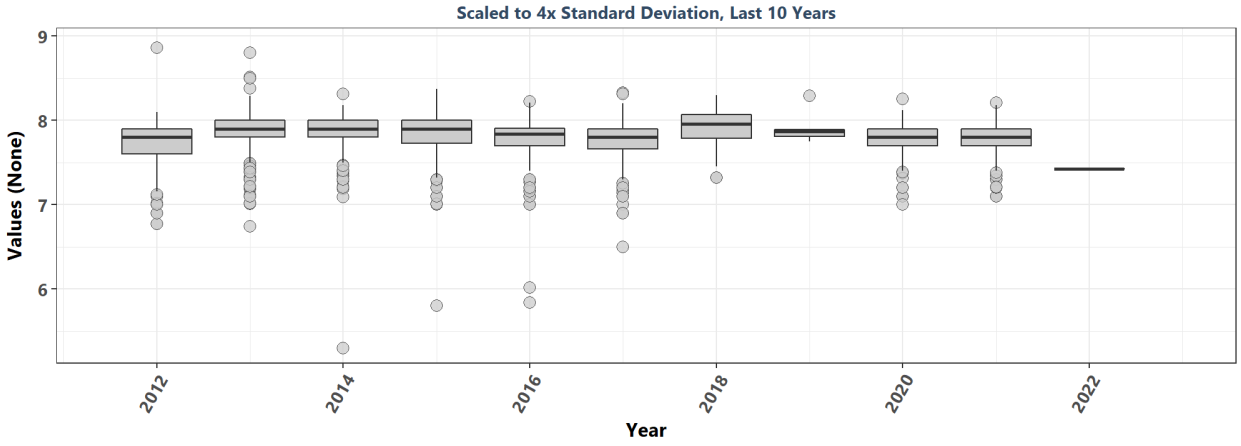

Cockroach Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

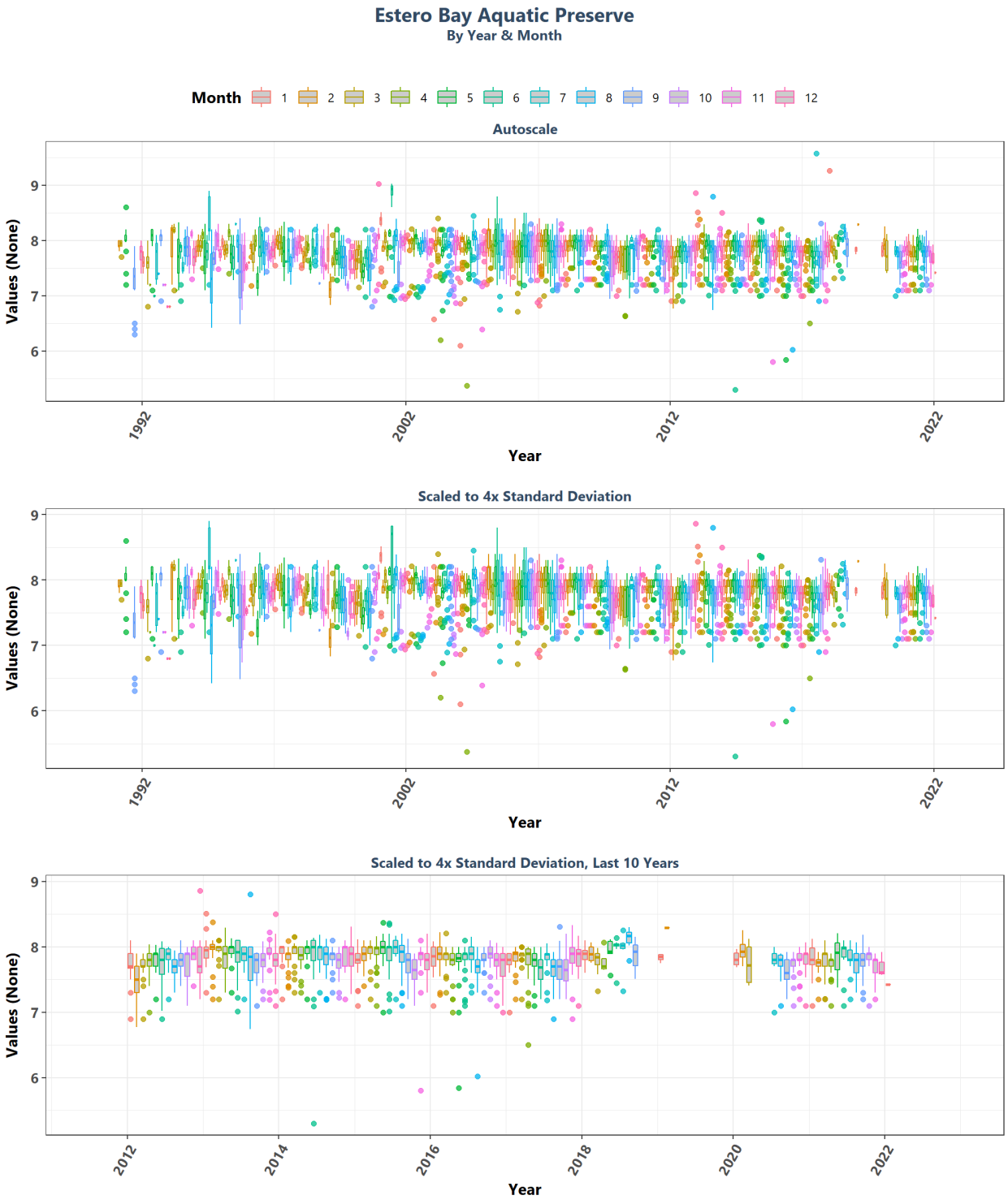

Cockroach Bay Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

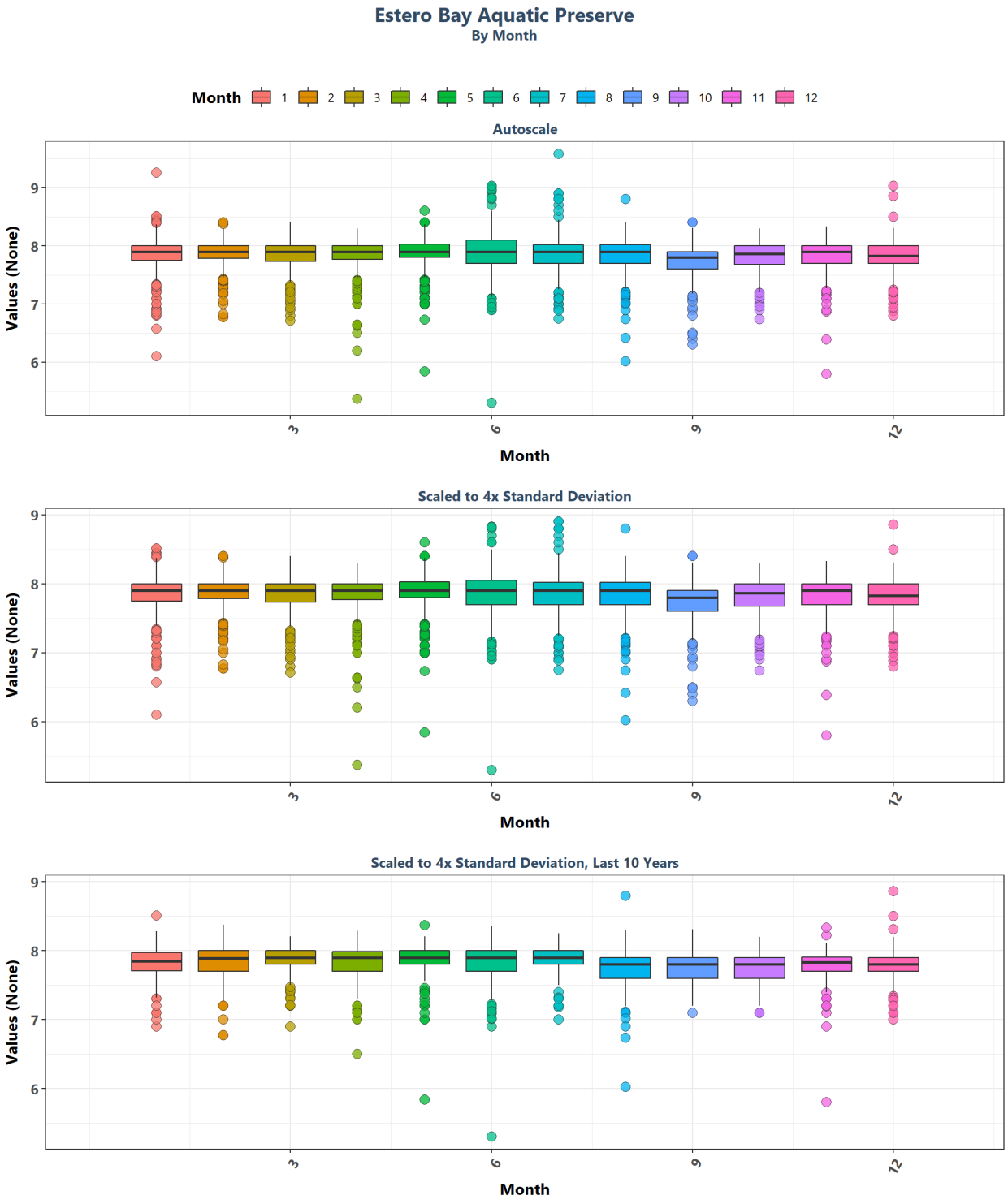
Estero Bay Aquatic Preserve
By Year





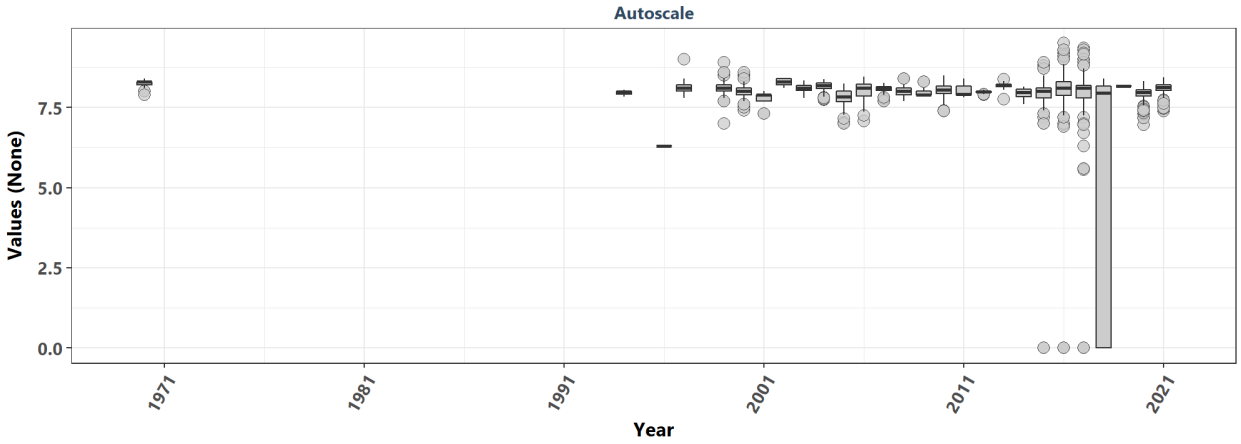


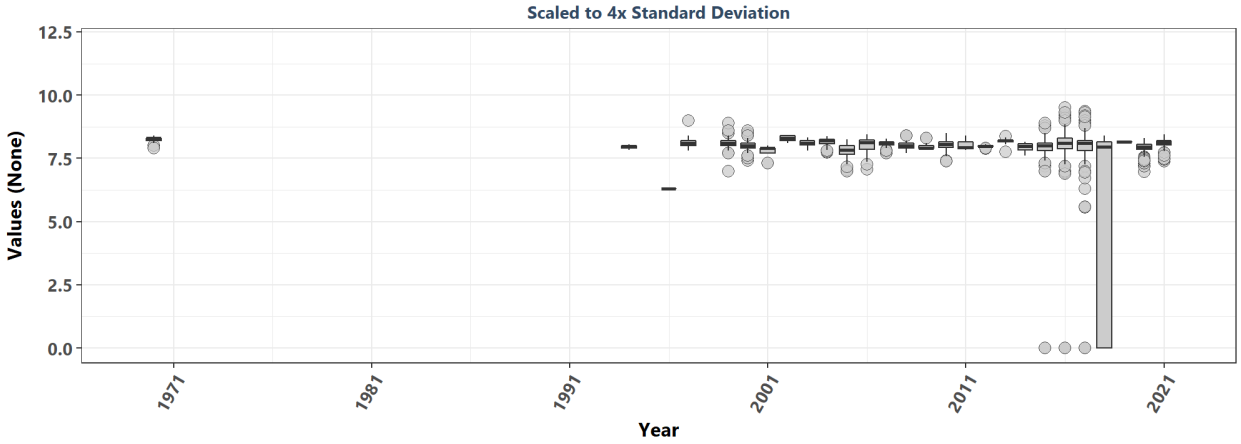

Estero Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

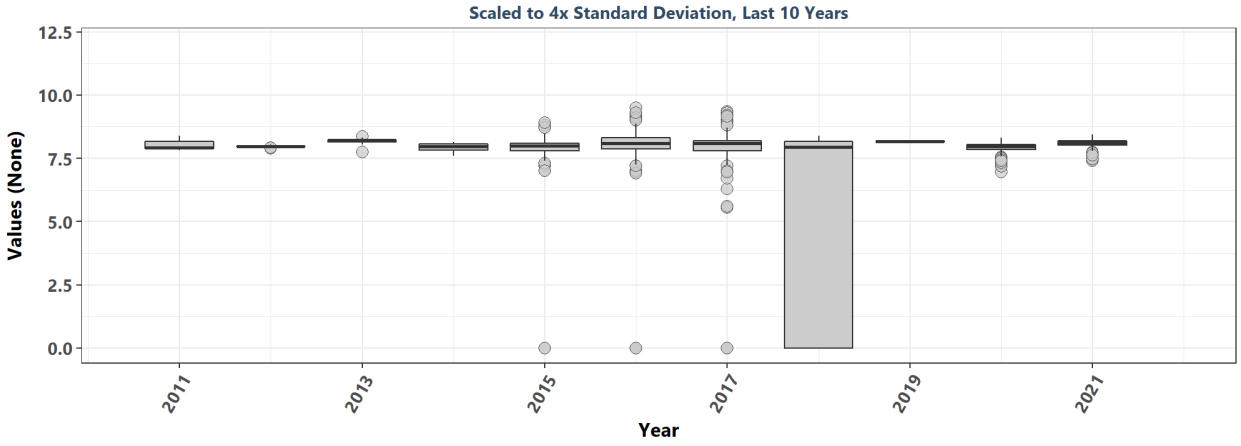

Estero Bay Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

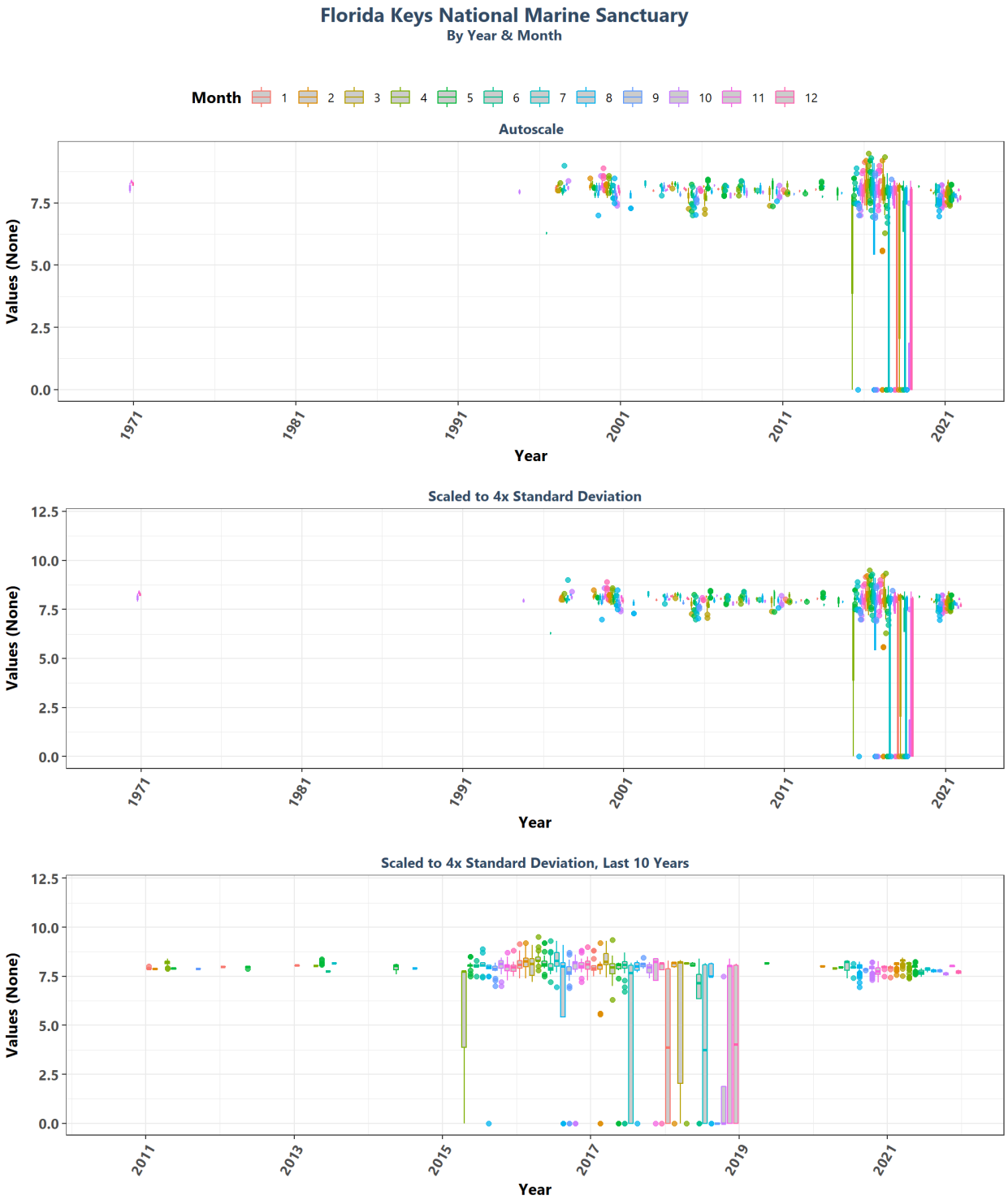
Florida Keys National Marine Sanctuary

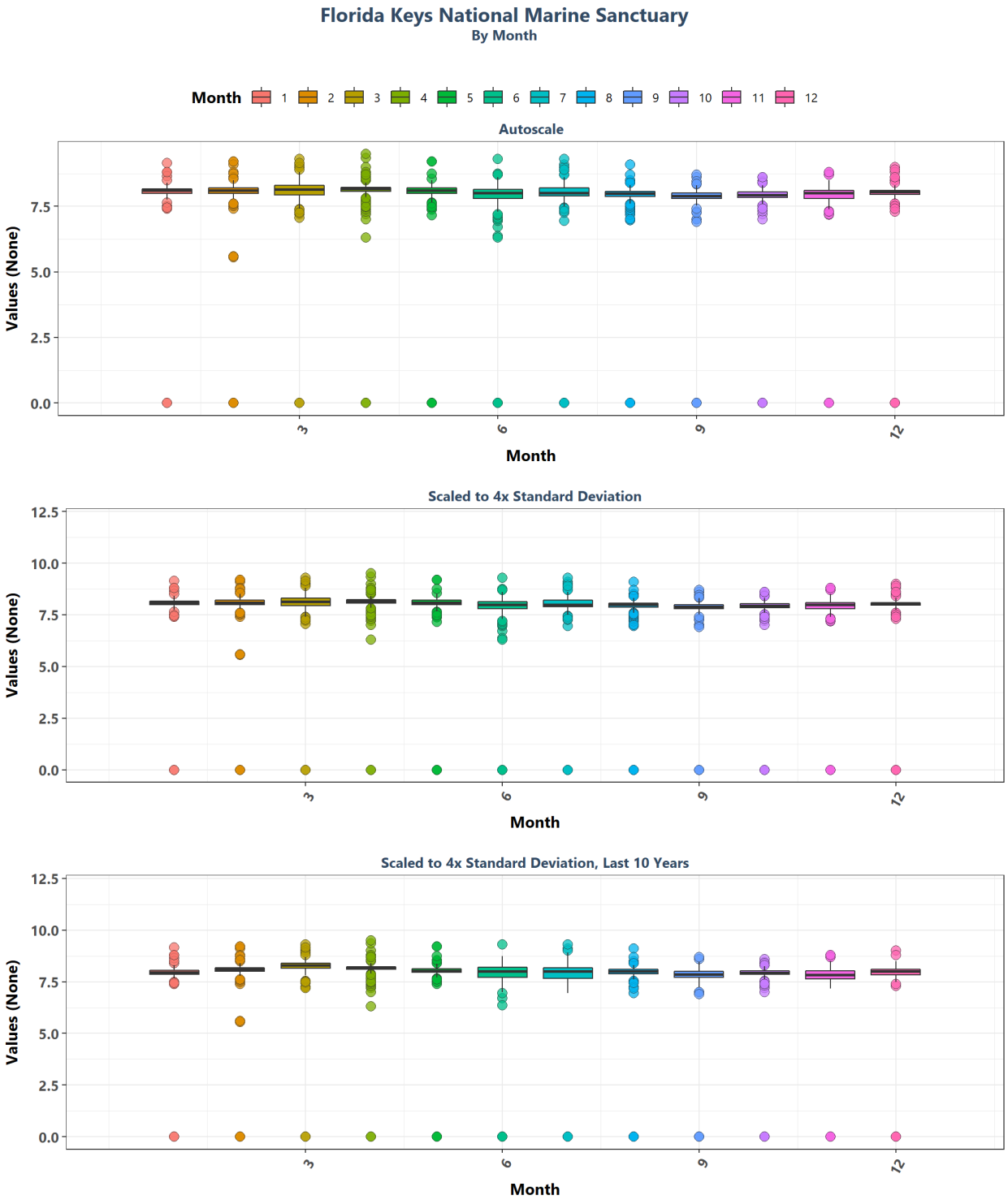
By Year





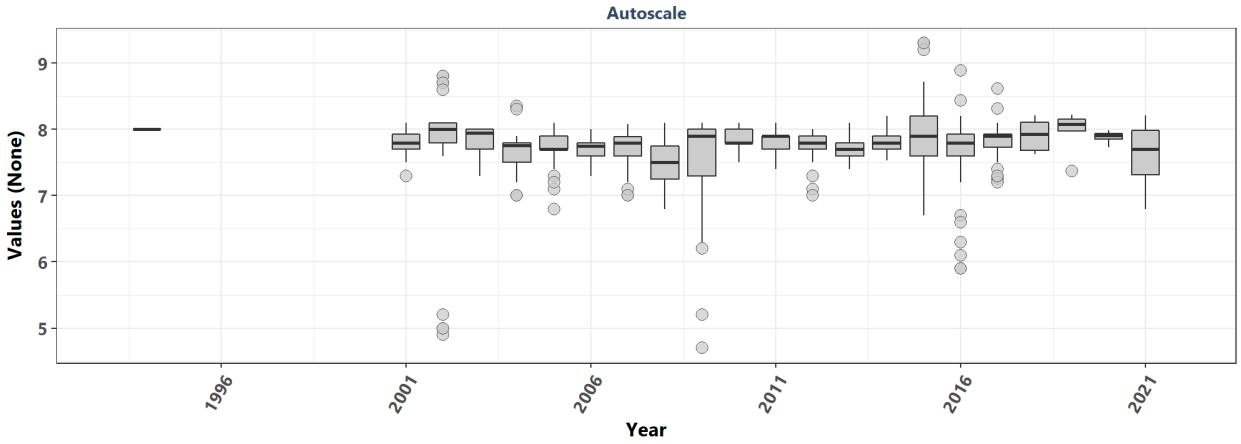


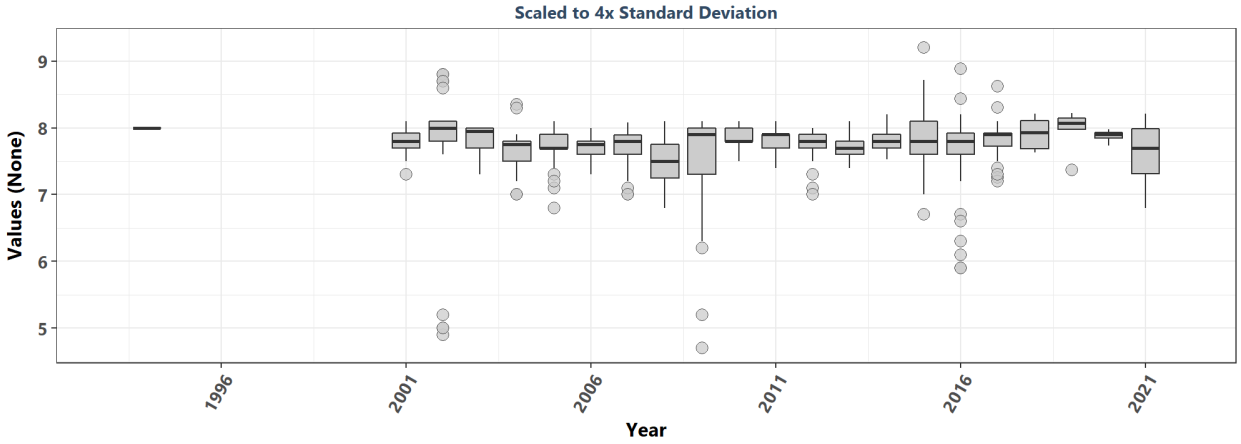

Florida Keys National Marine Sanctuary
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

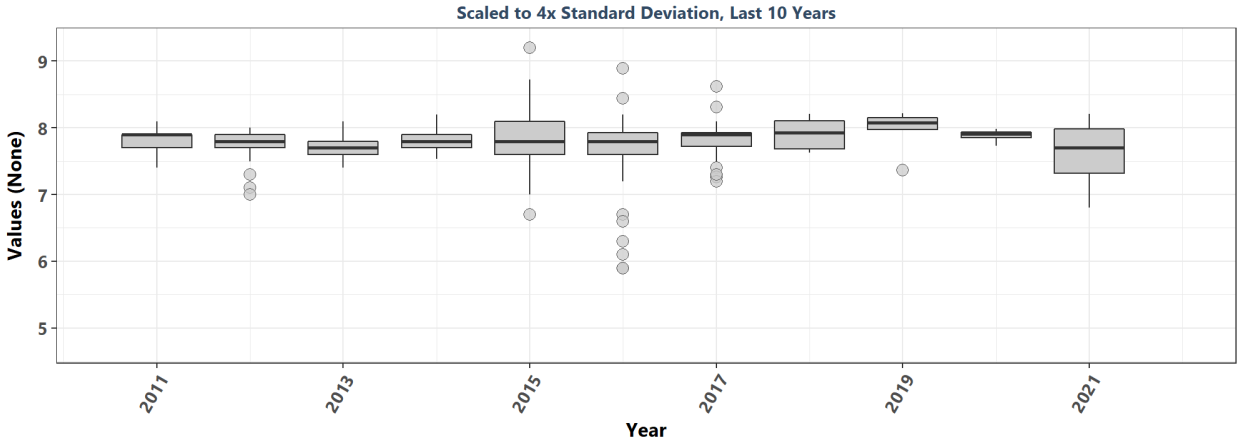

Florida Keys National Marine Sanctuary
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

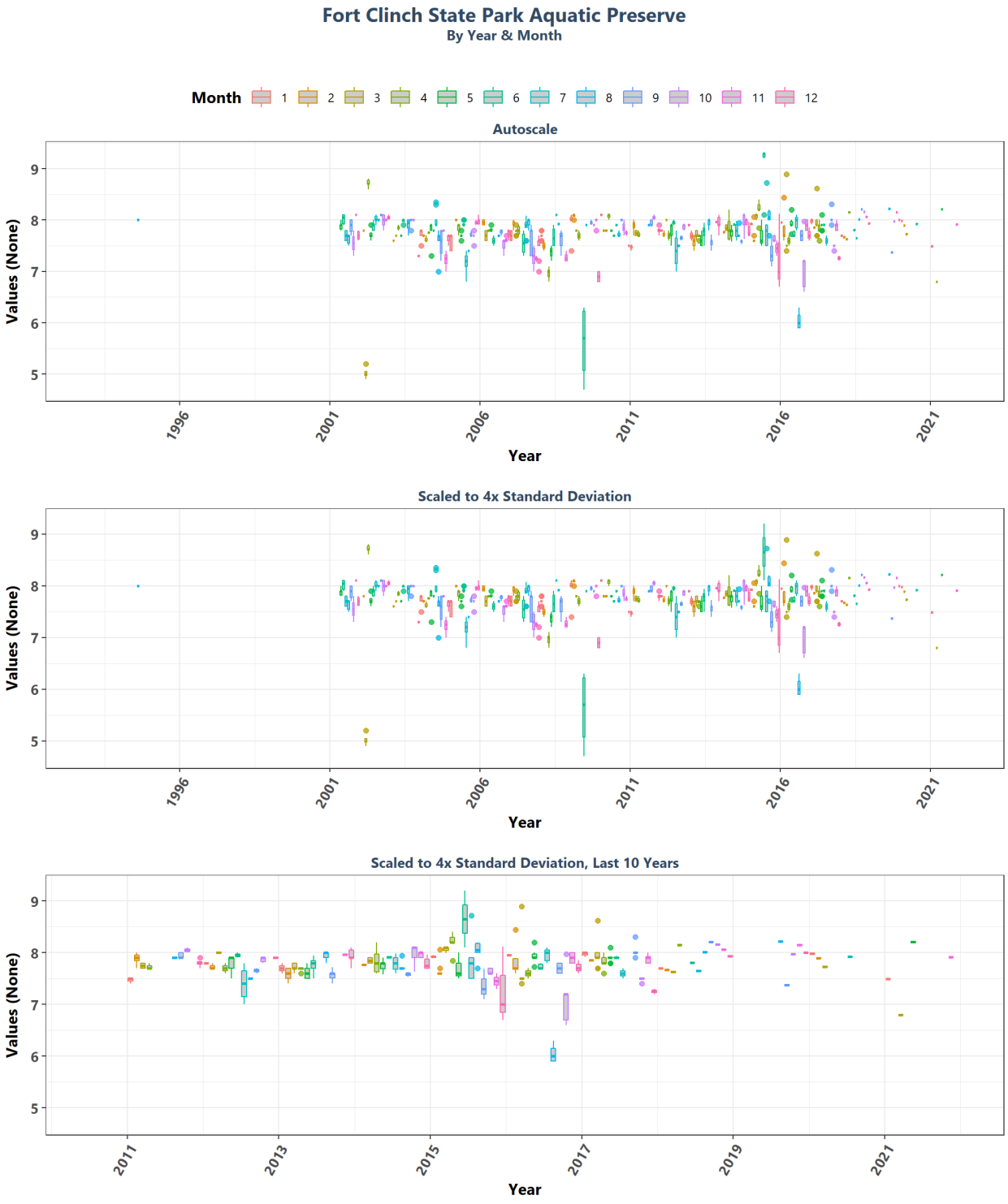
Fort Clinch State Park Aquatic Preserve

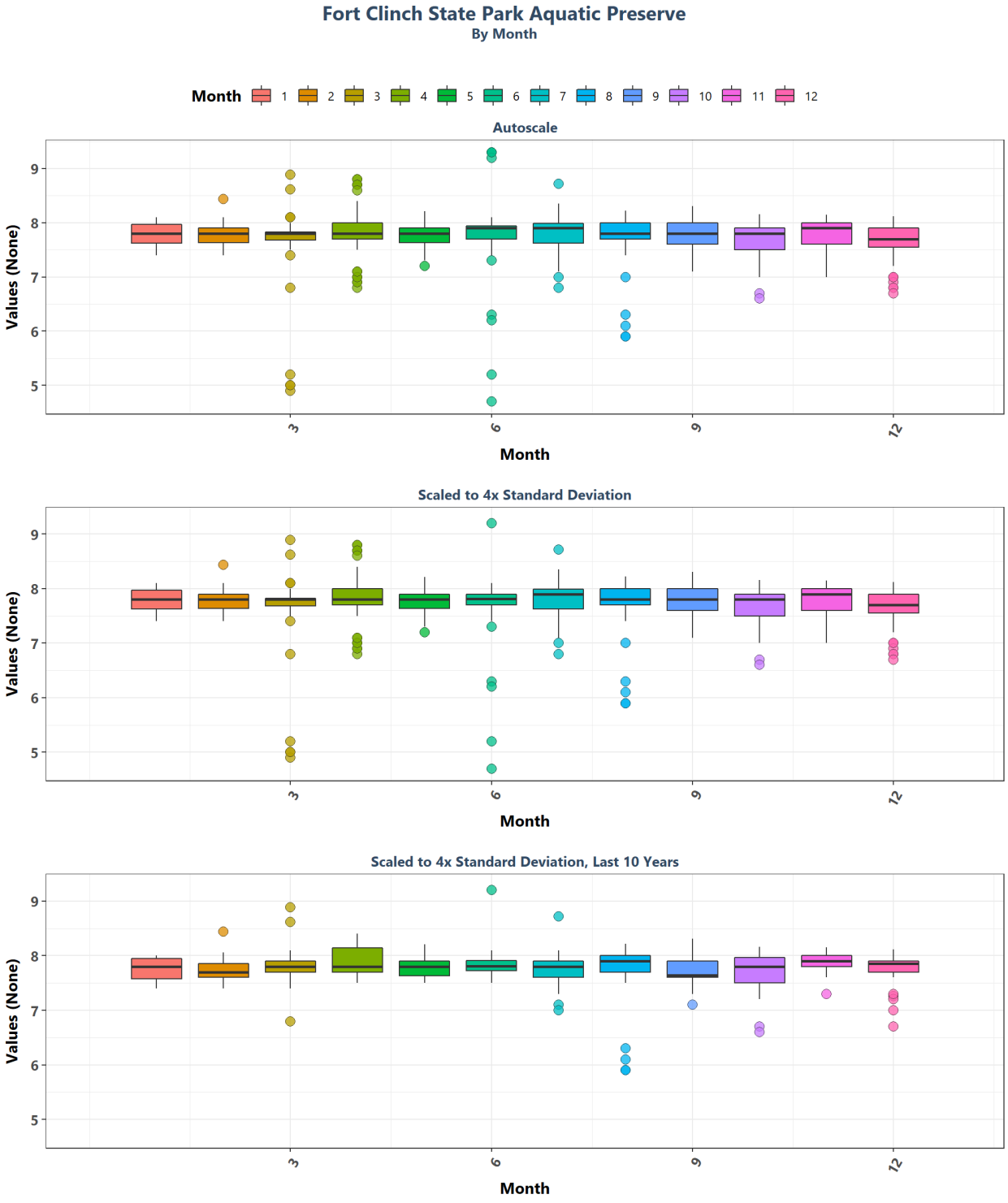
By Year





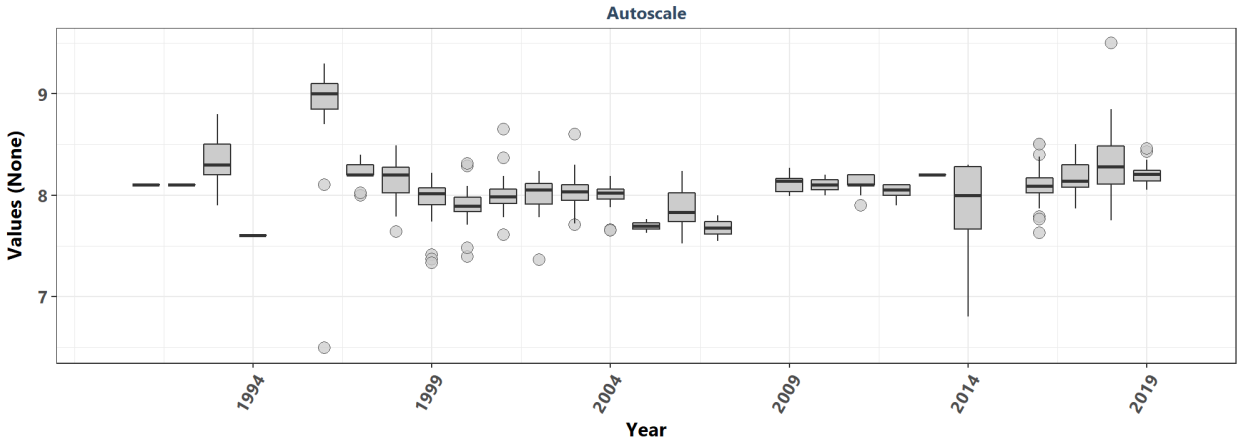


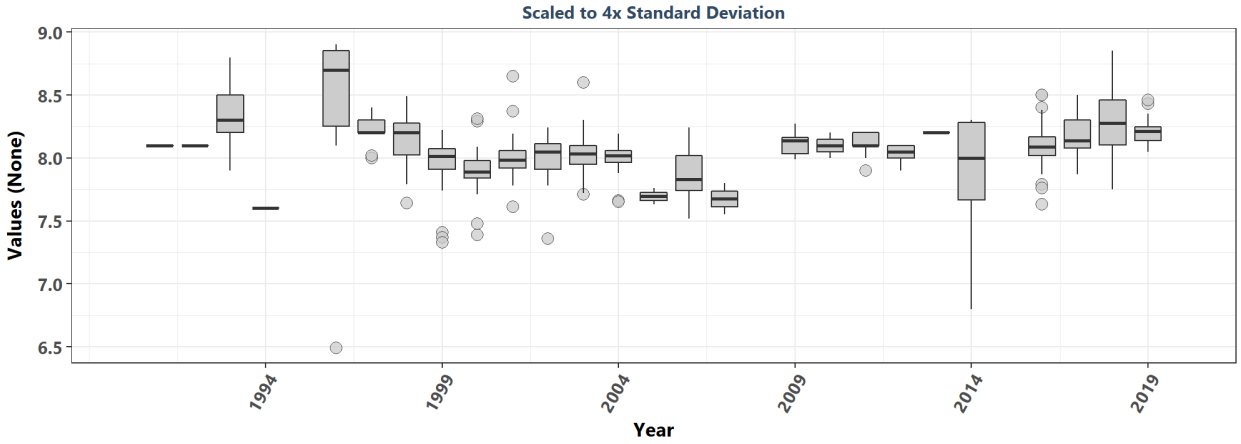

Fort Clinch State Park Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

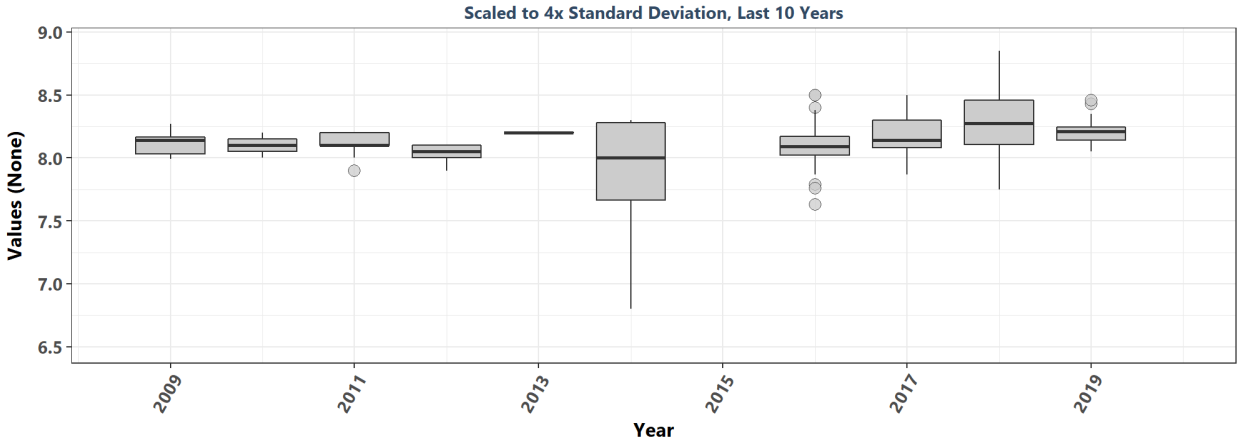

Fort Clinch State Park Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

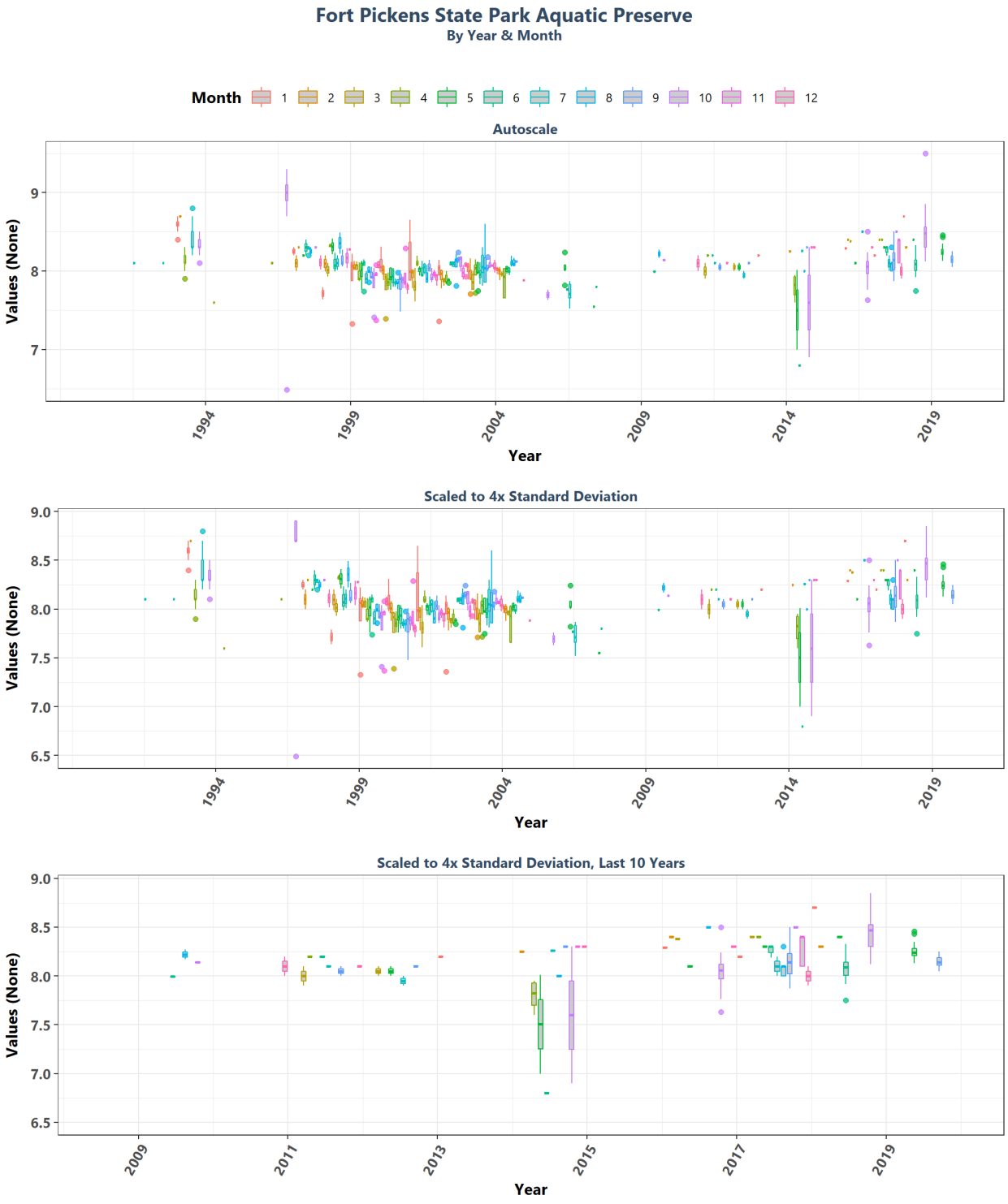
Fort Pickens State Park Aquatic Preserve

By Year



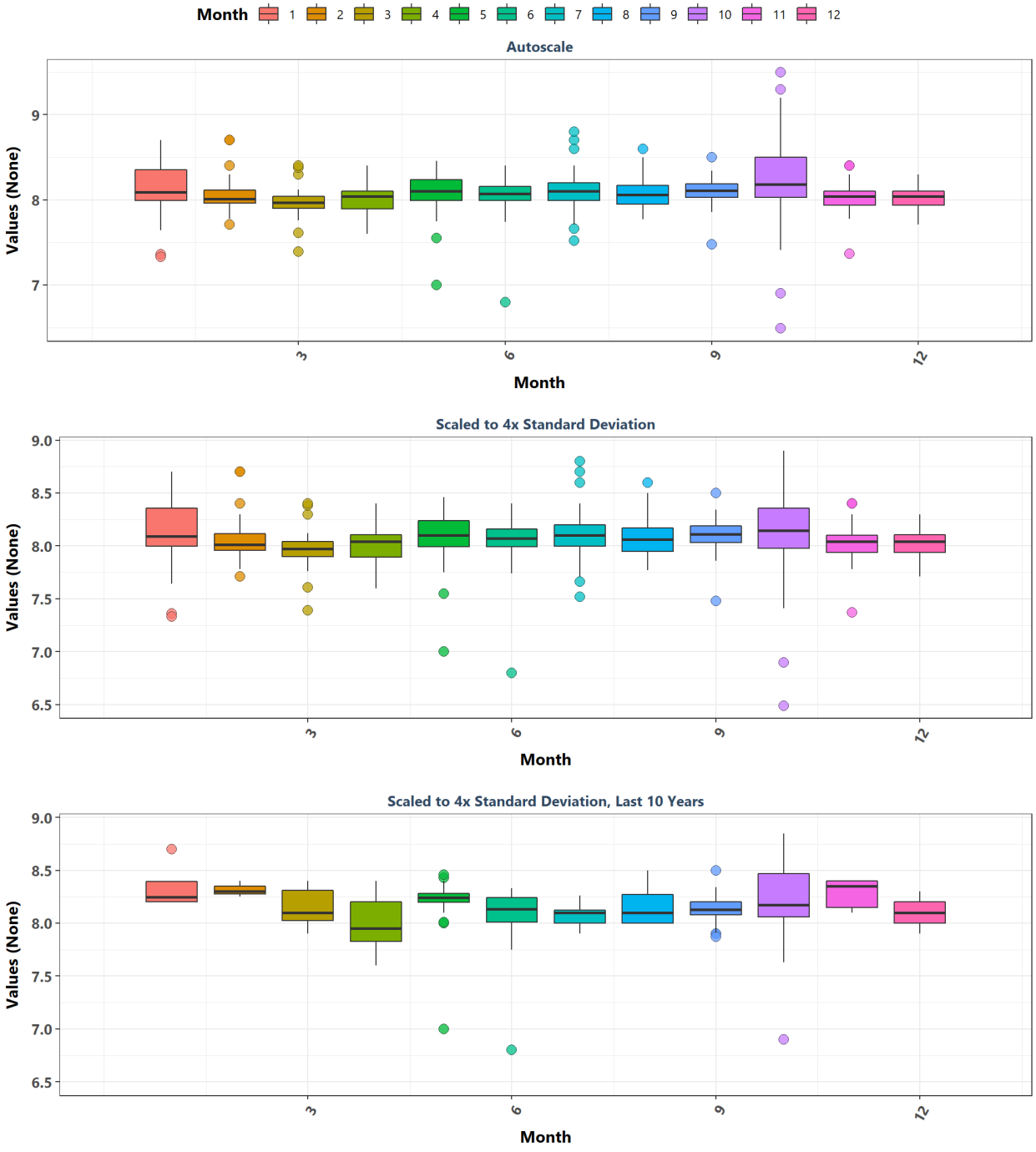


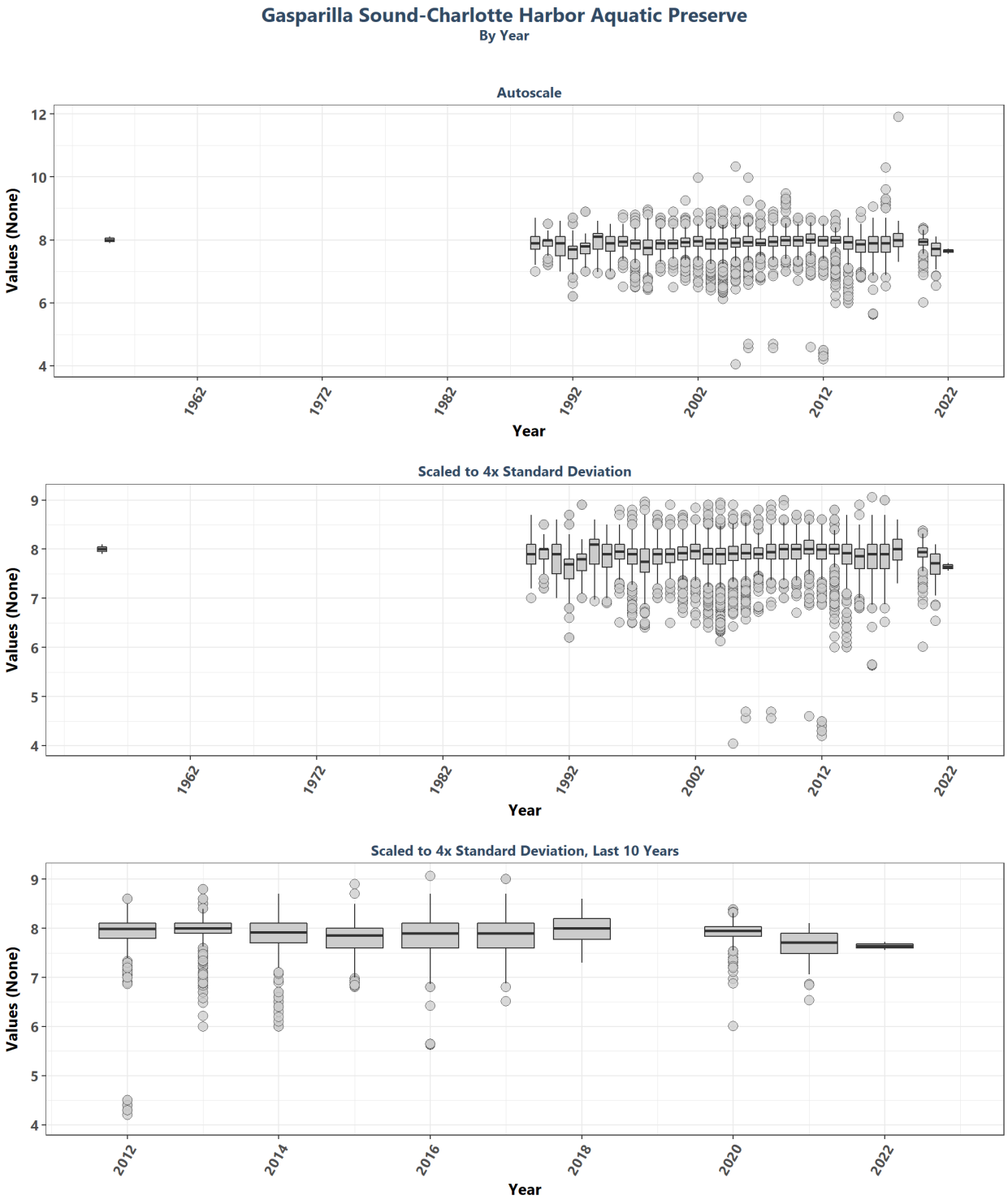


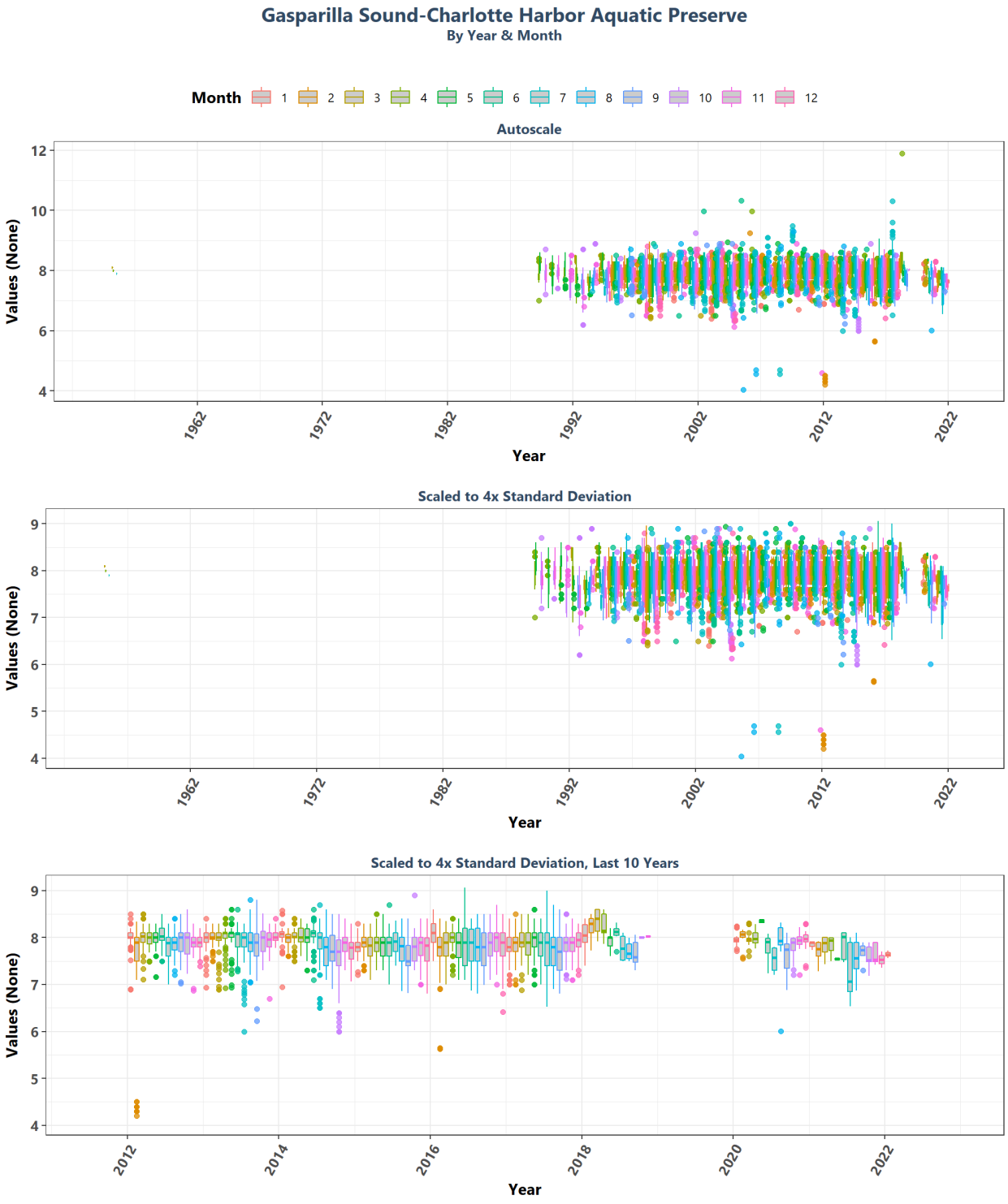

Fort Pickens State Park Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

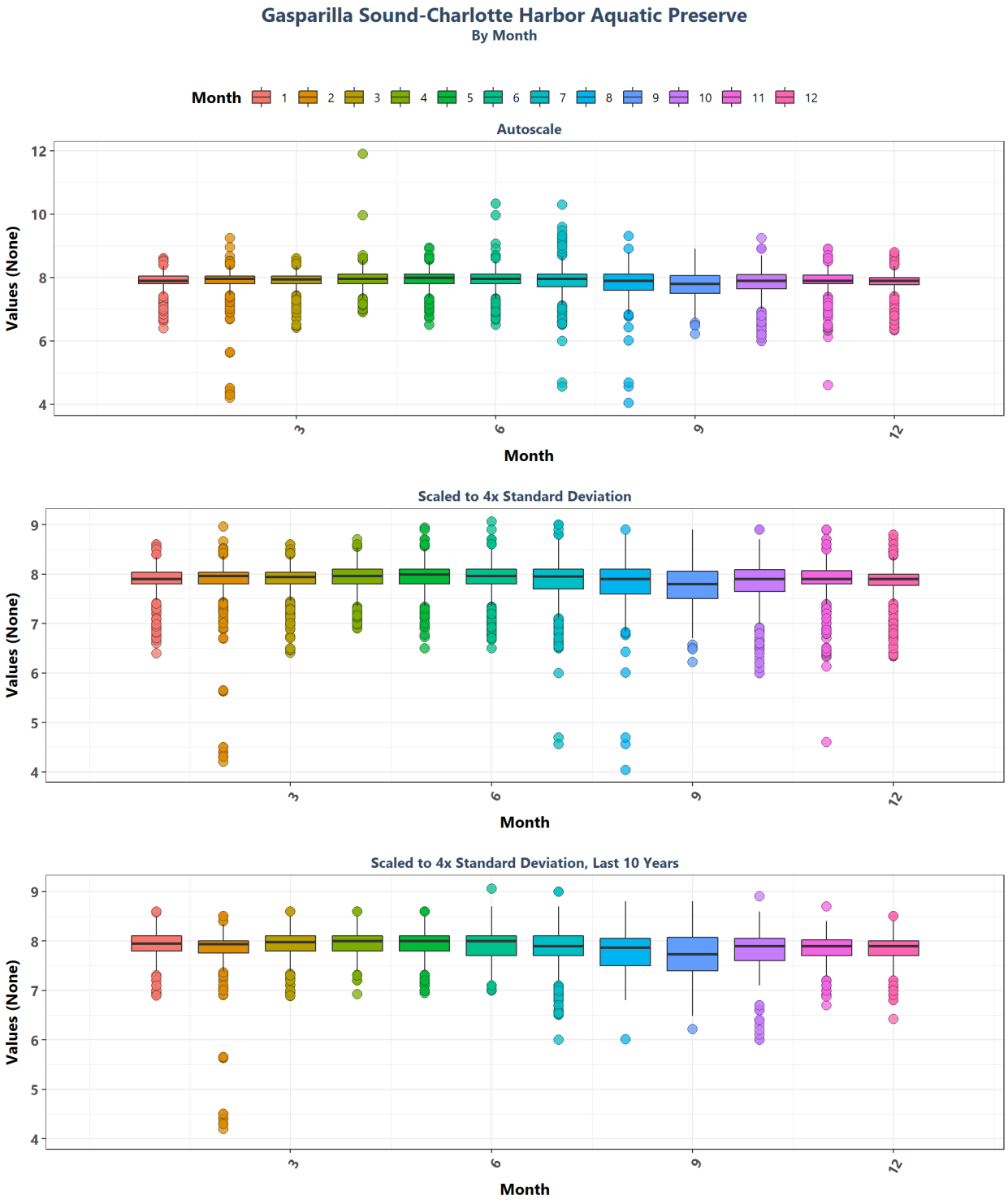
Fort Pickens State Park Aquatic Preserve

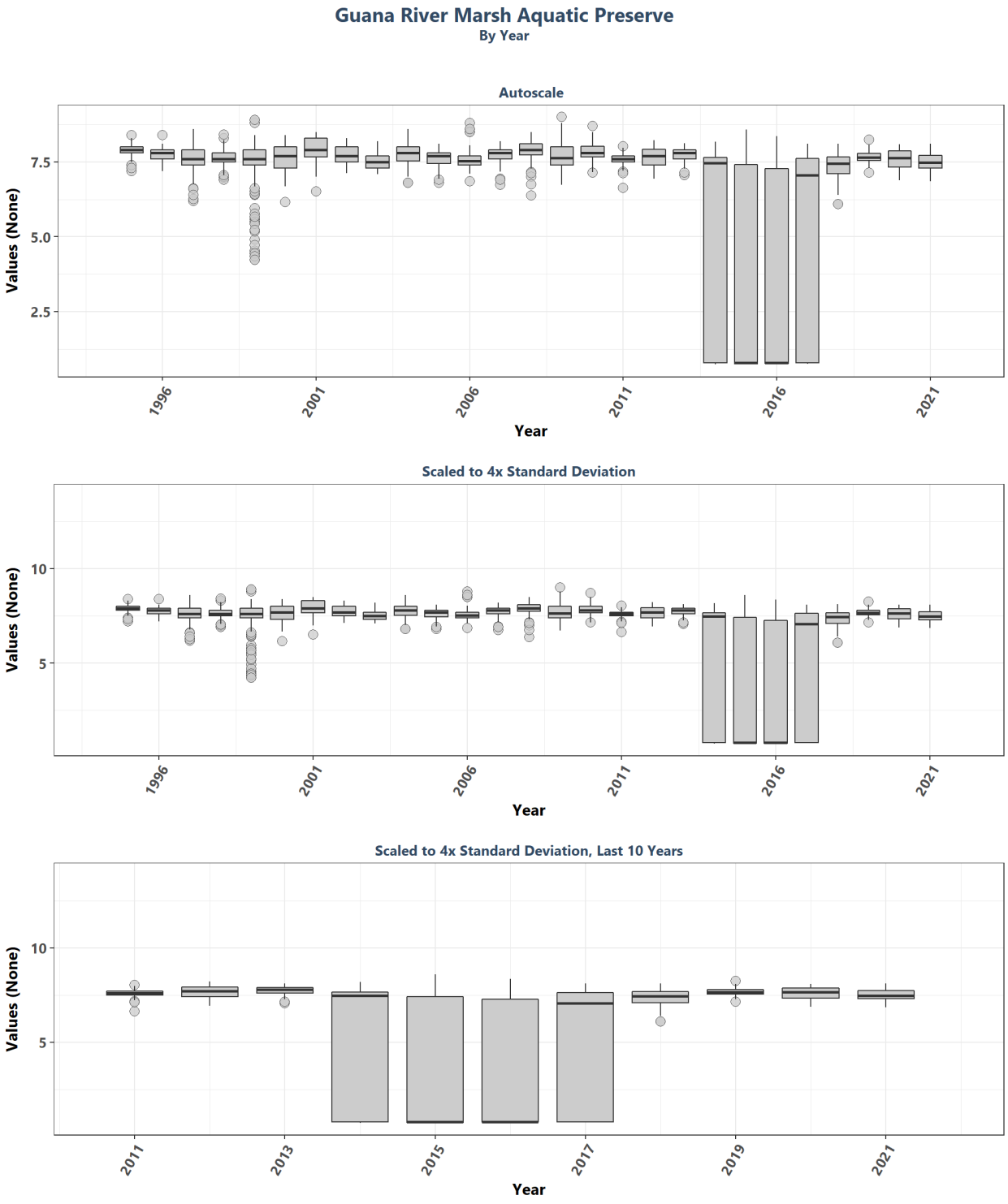
By Month

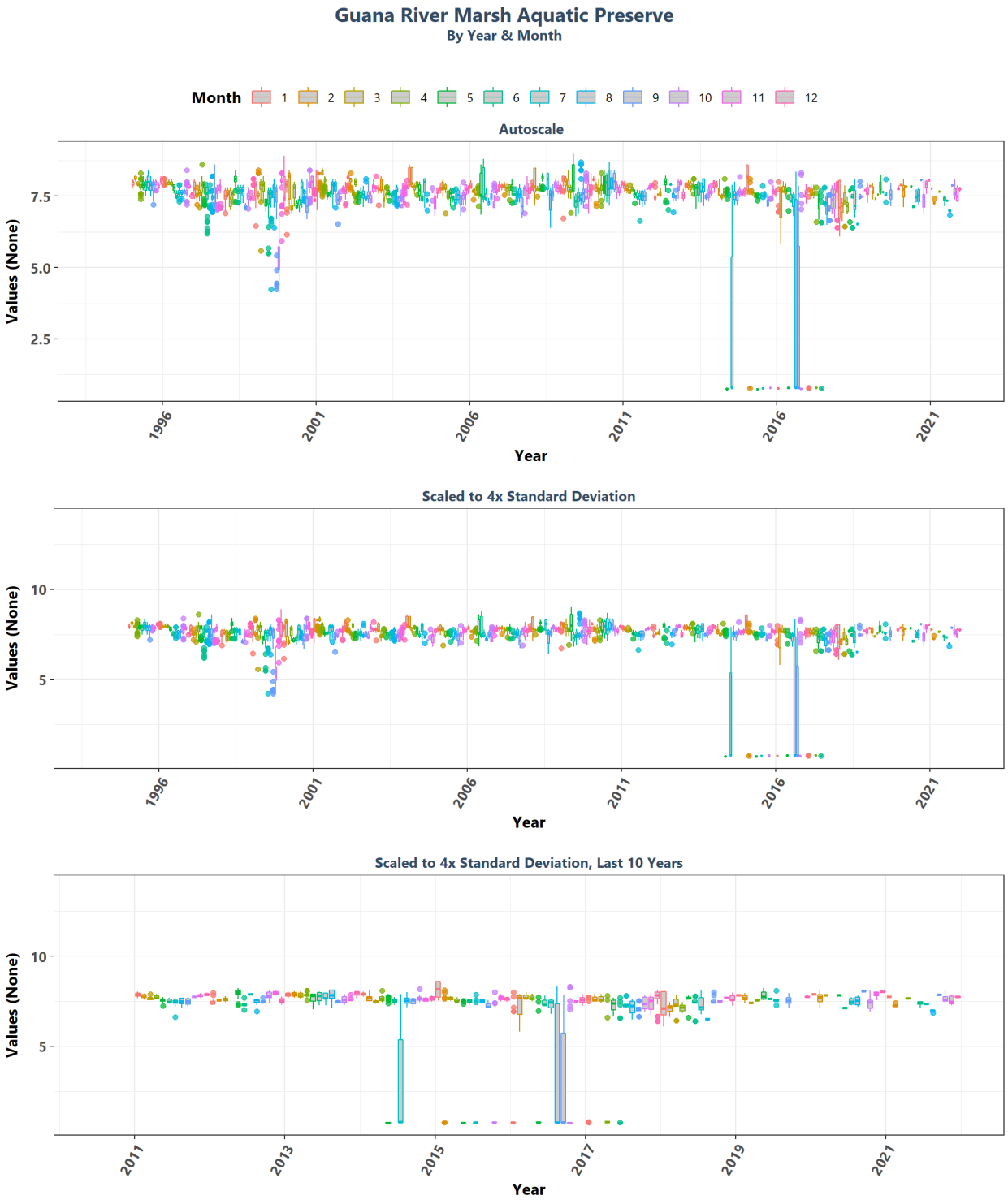


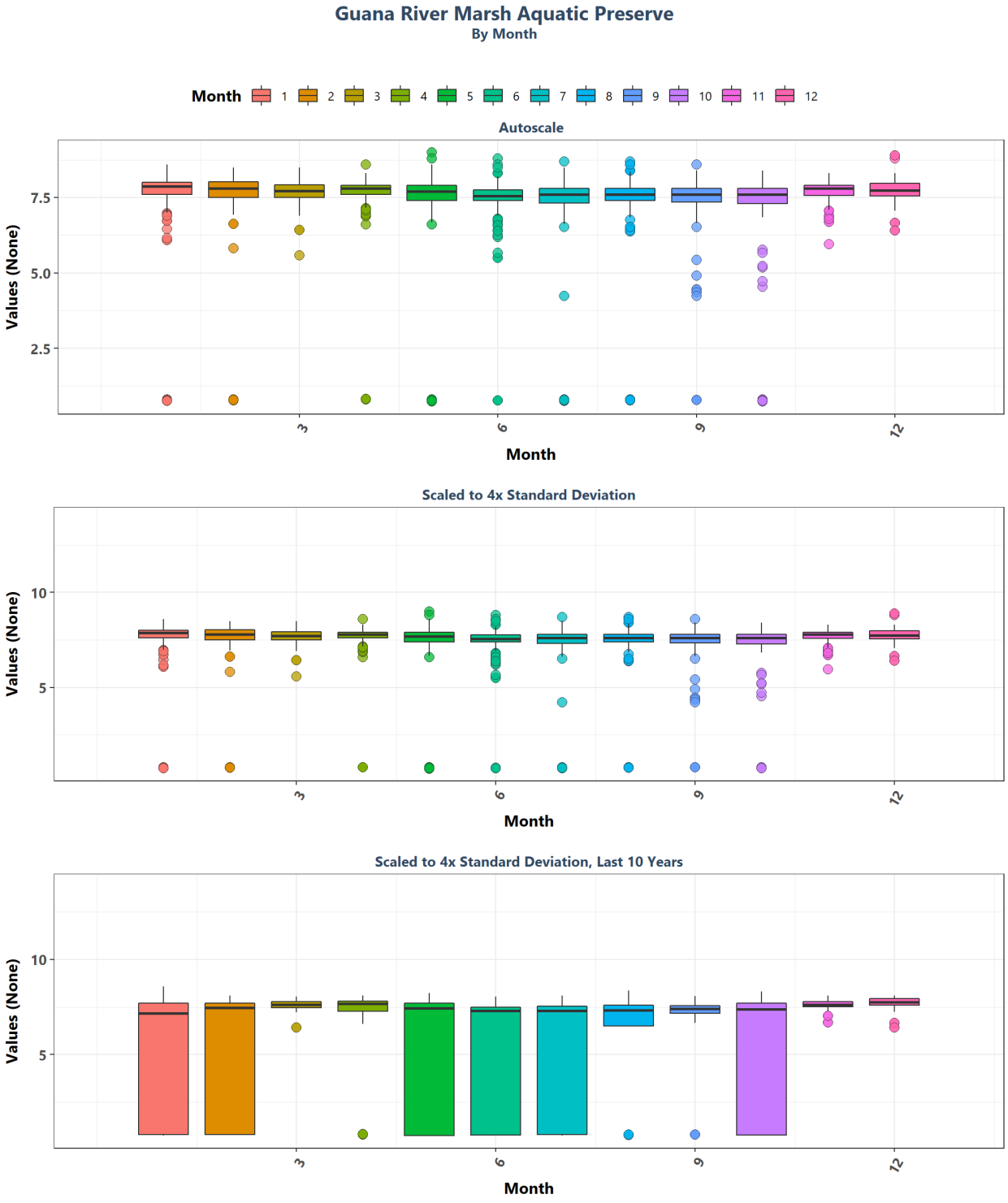

Gasparilla Sound-Charlotte Harbor Aquatic Preserve
By Year
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year


Gasparilla Sound-Charlotte Harbor Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year


Gasparilla Sound-Charlotte Harbor Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

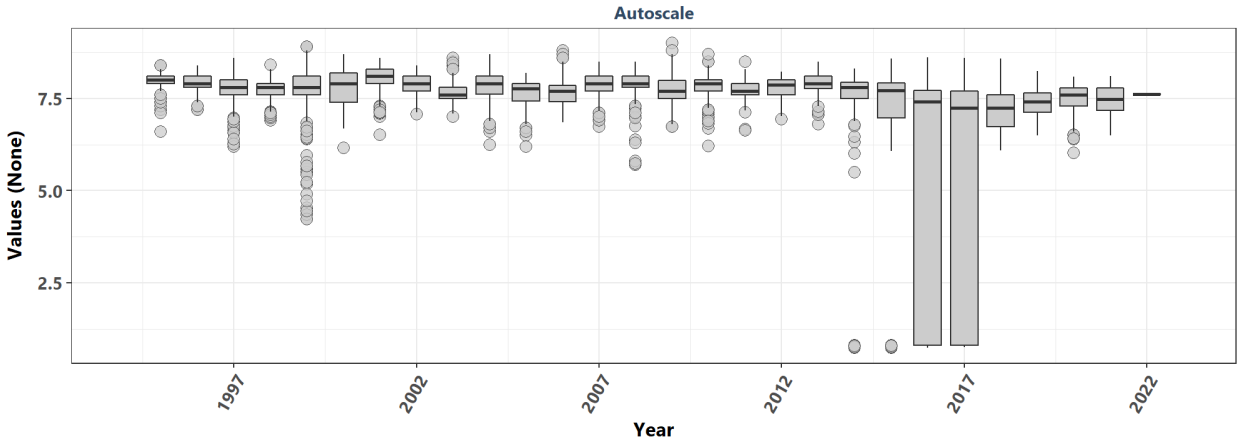

Guana River Marsh Aquatic Preserve
By Year
Autoscale
Values (None)
7.5
5.0
2.5
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (None)
10
5
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
10
5
2011
2013
2015
2017
2019
2021
Year

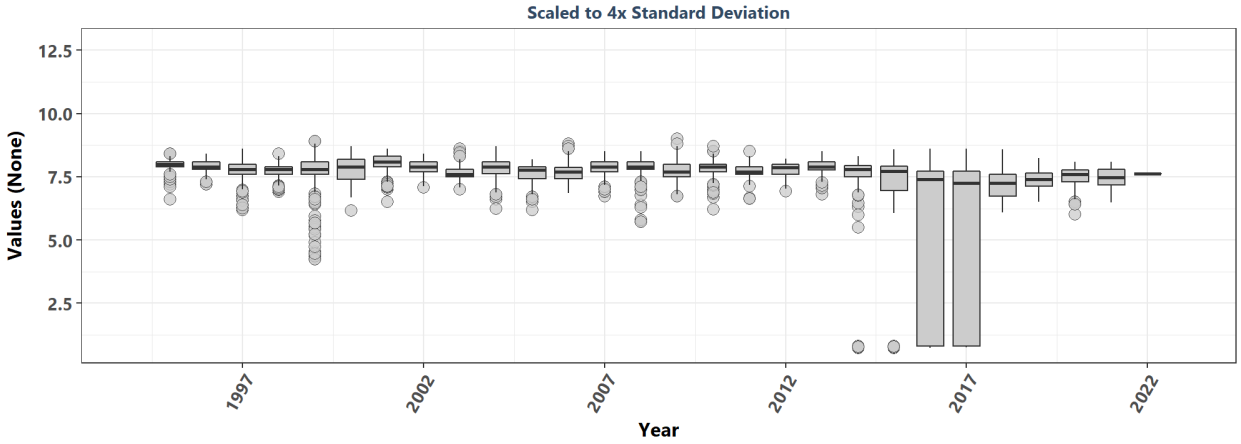

Guana River Marsh Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
7.5
5.0
2.5
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (None)
10
5
1996
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
10
5
2011
2013
2015
2017
2019
2021
Year

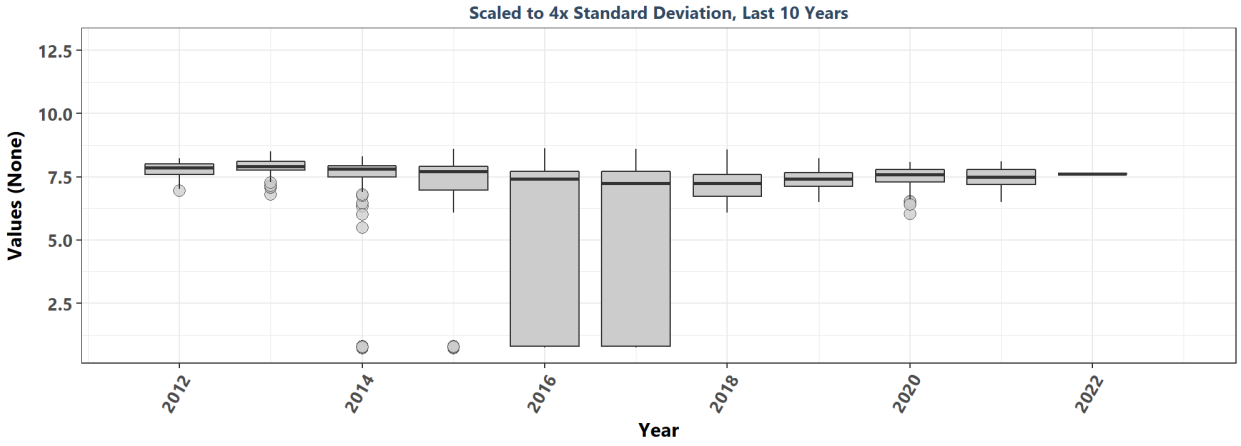

Guana River Marsh Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
7.5
5.0
2.5
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
10
5
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
10
5
3
6
9
12
Month

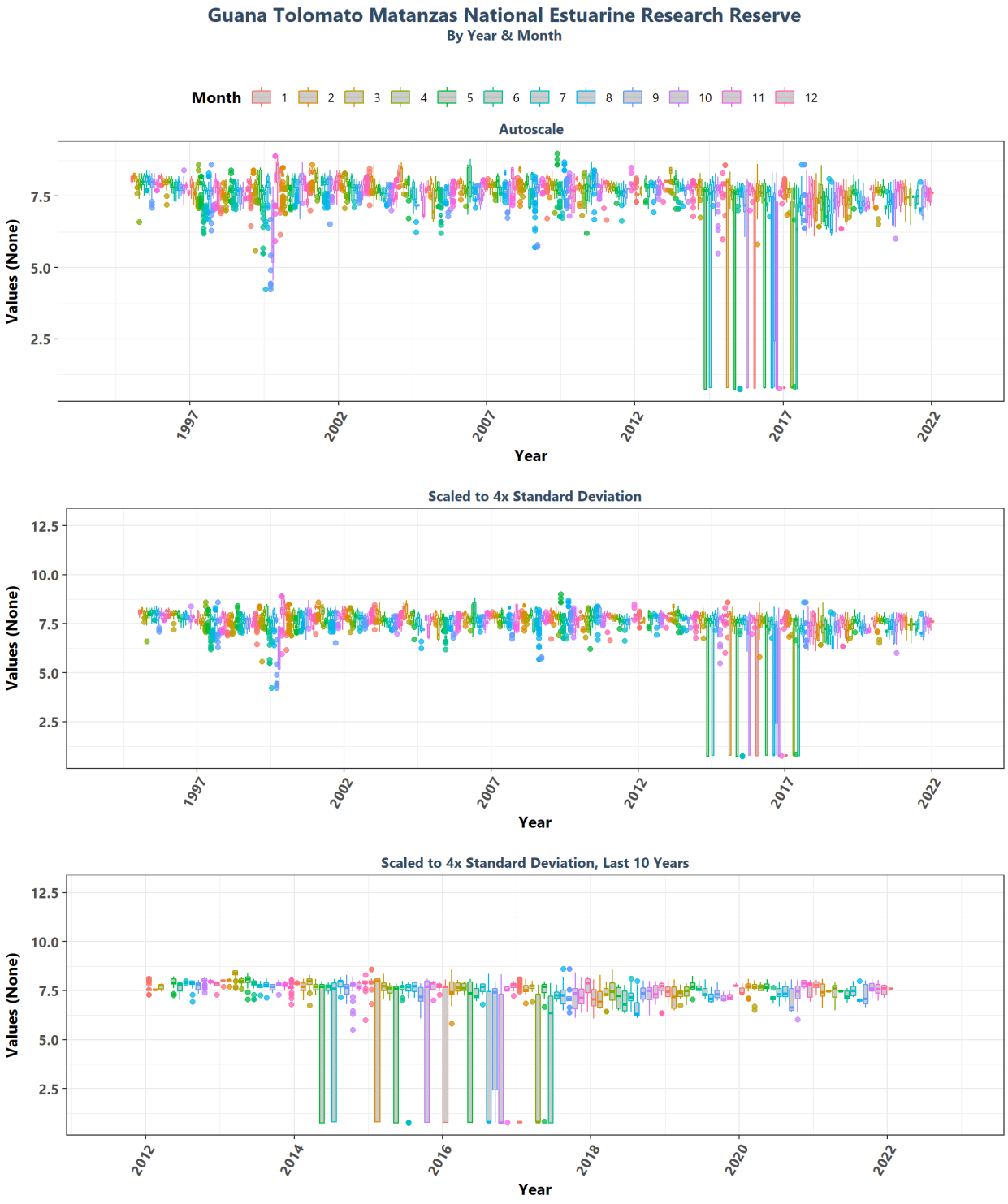
Guana Tolomato Matanzas National Estuarine Research Reserve

By Year



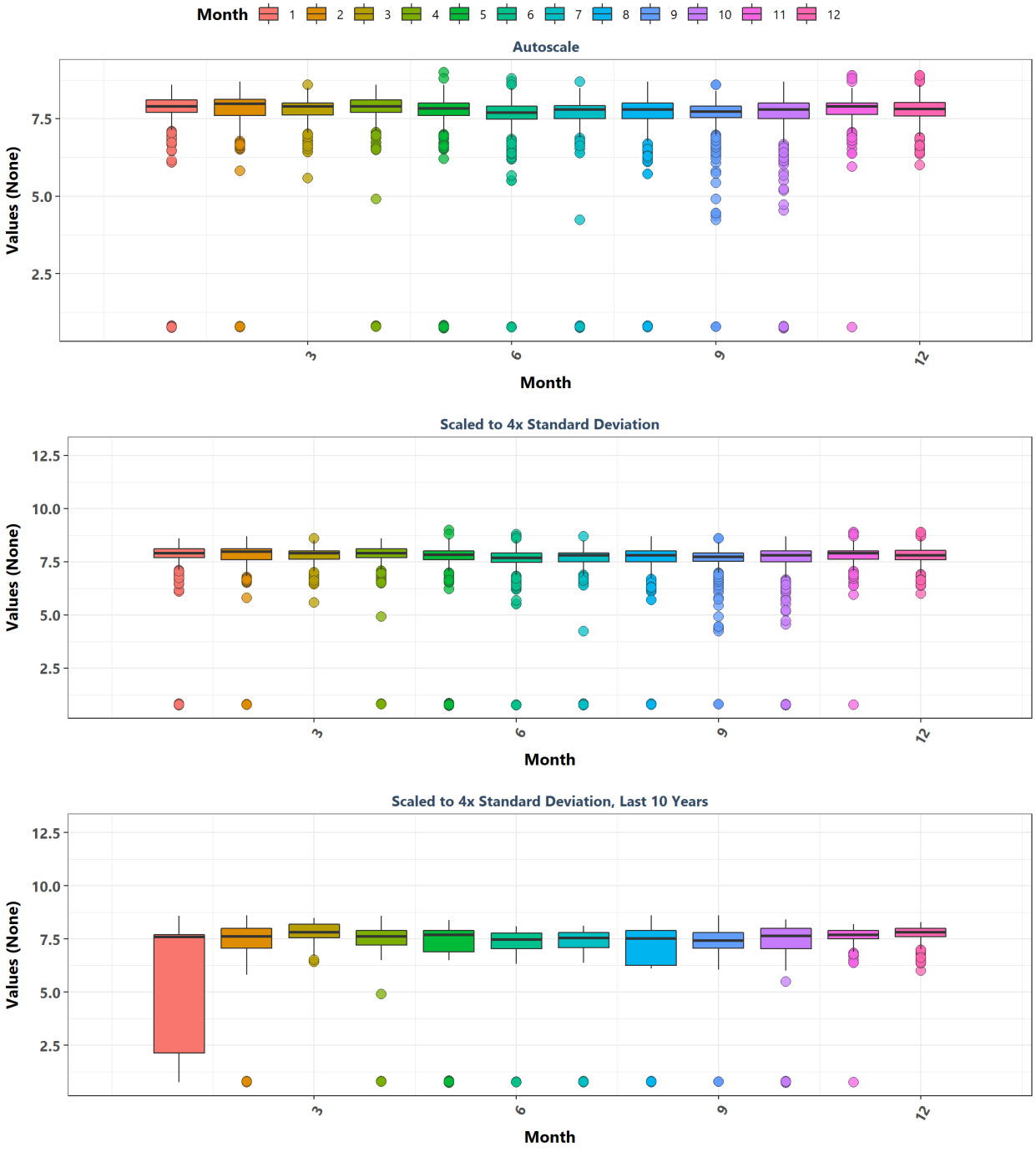


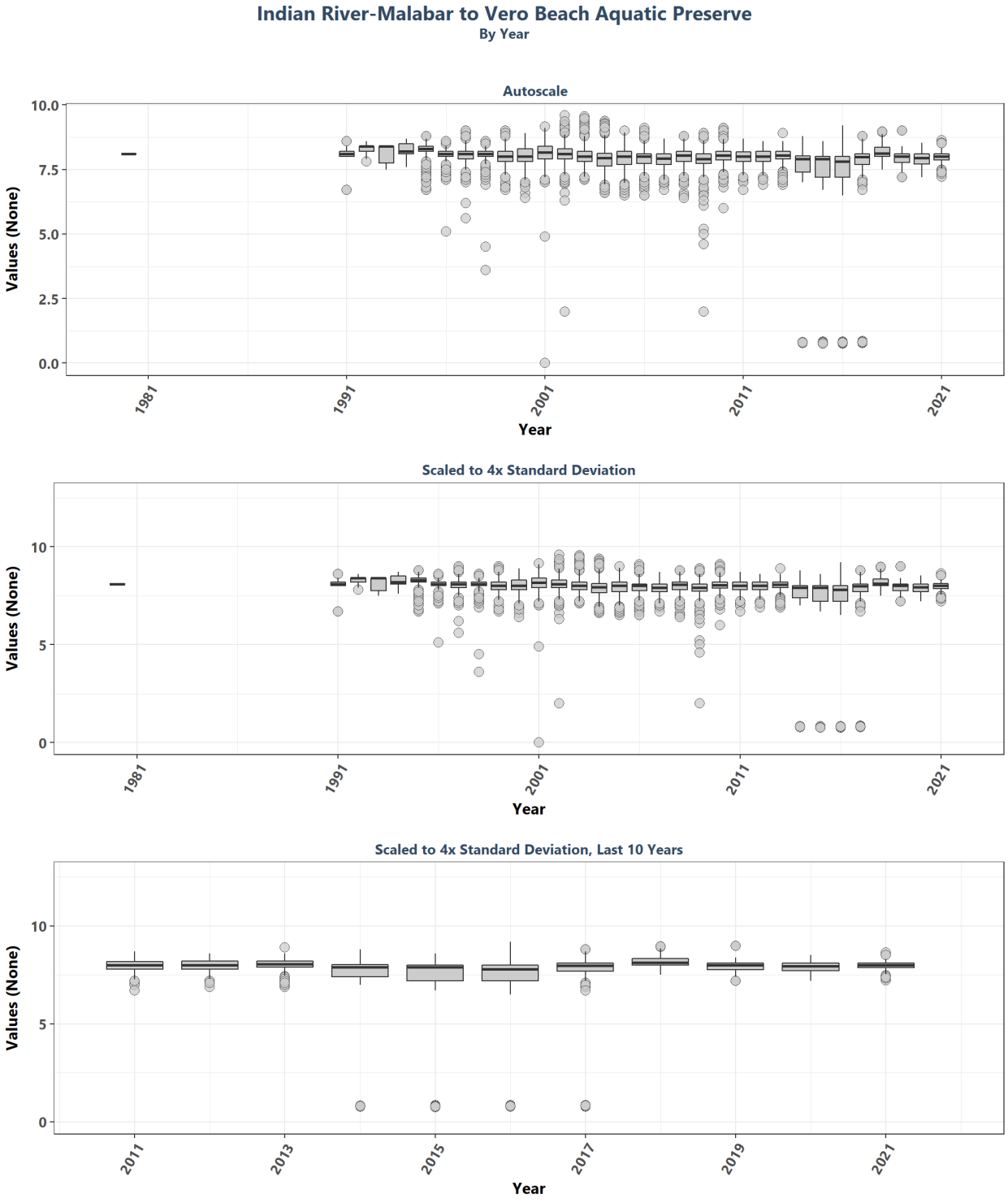


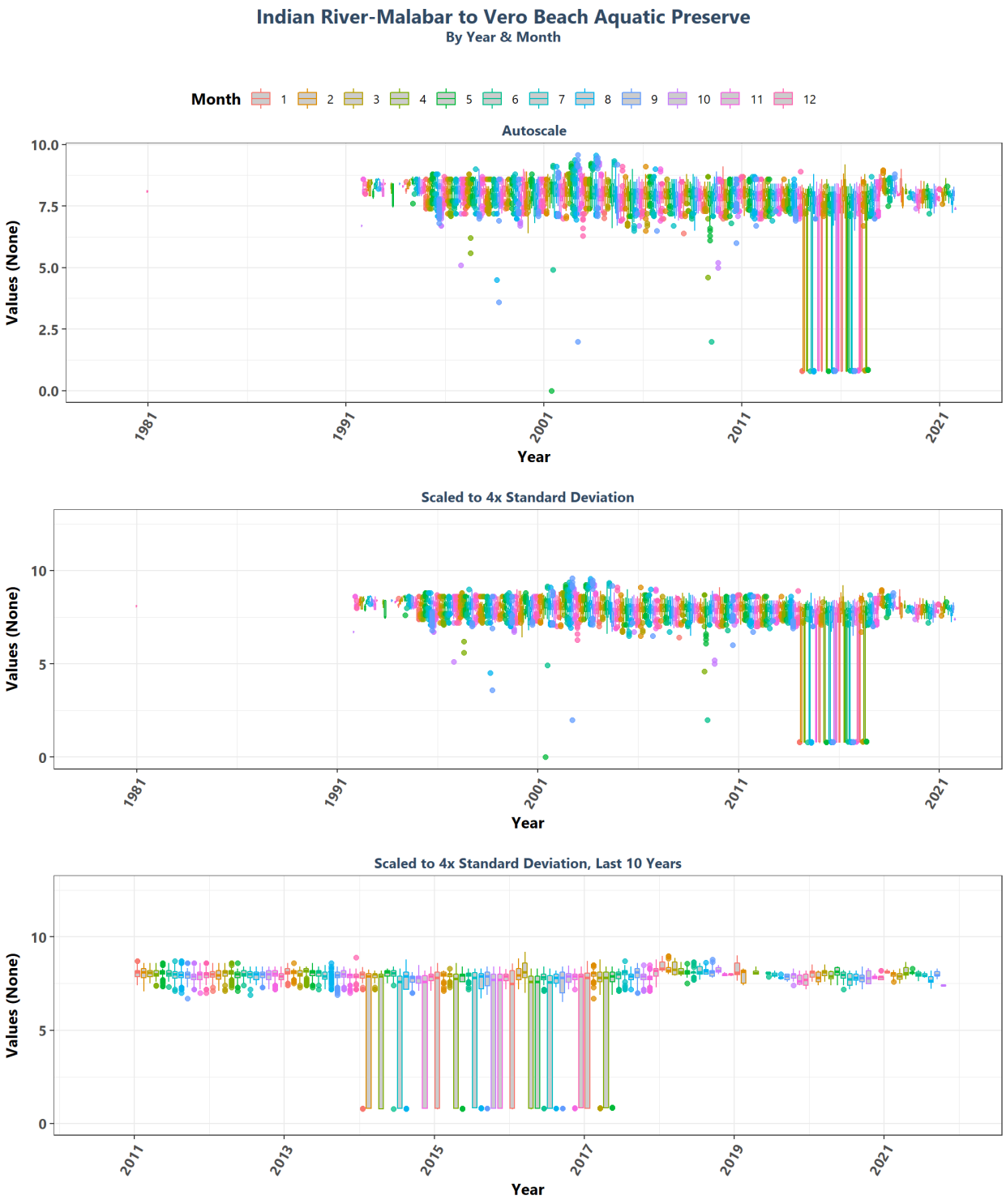

Guana Tolomato Matanzas National Estuarine Research Reserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
7.5
5.0
2.5
1997
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation
Values (None)
12.5
10.0
7.5
5.0
2.5
1997
2002
2007
2012
2017
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
12.5
10.0
7.5
5.0
2.5
2012
2014
2016
2018
2020
2022
Year

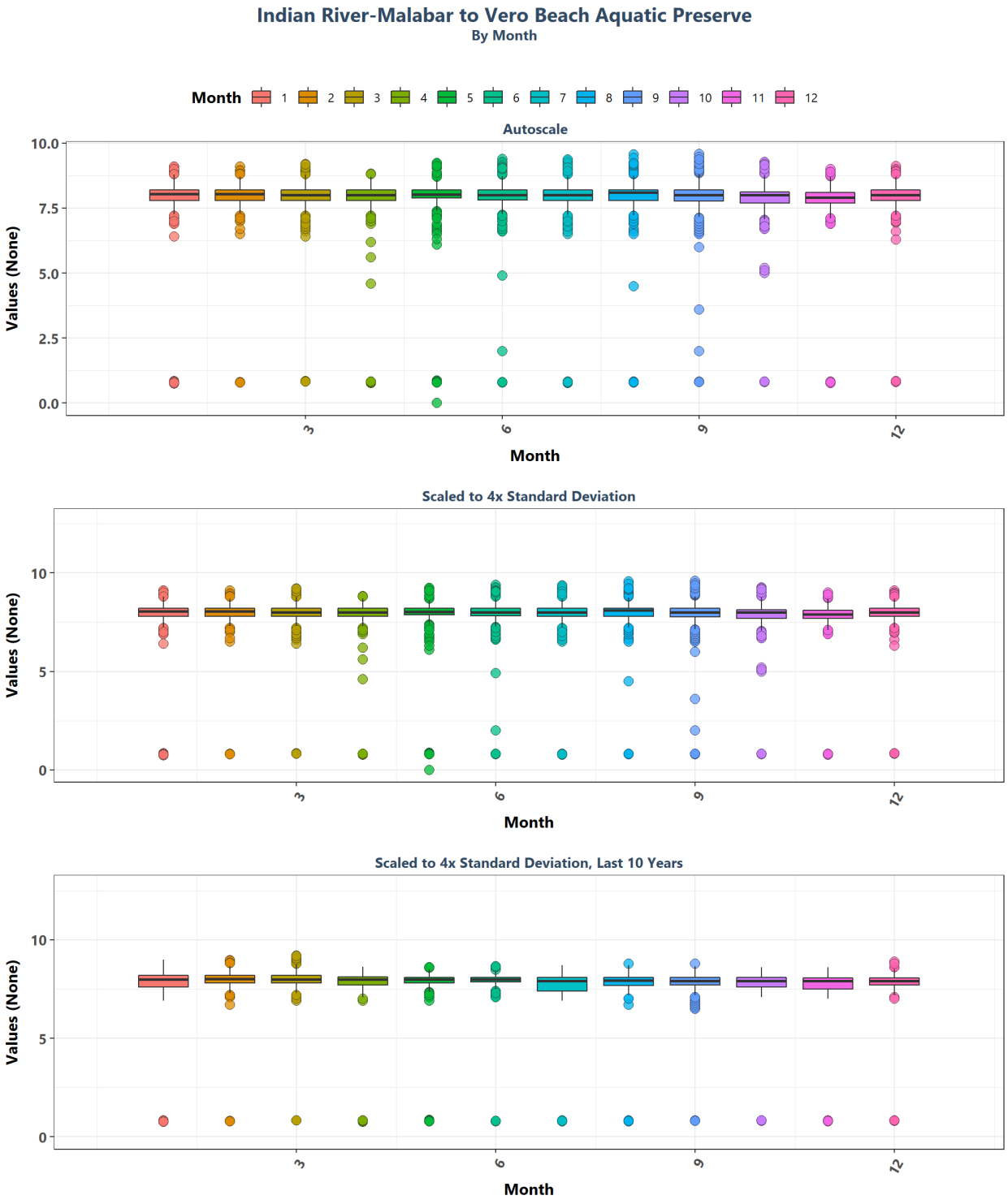
Guana Tolomato Matanzas National Estuarine Research Reserve

By Month



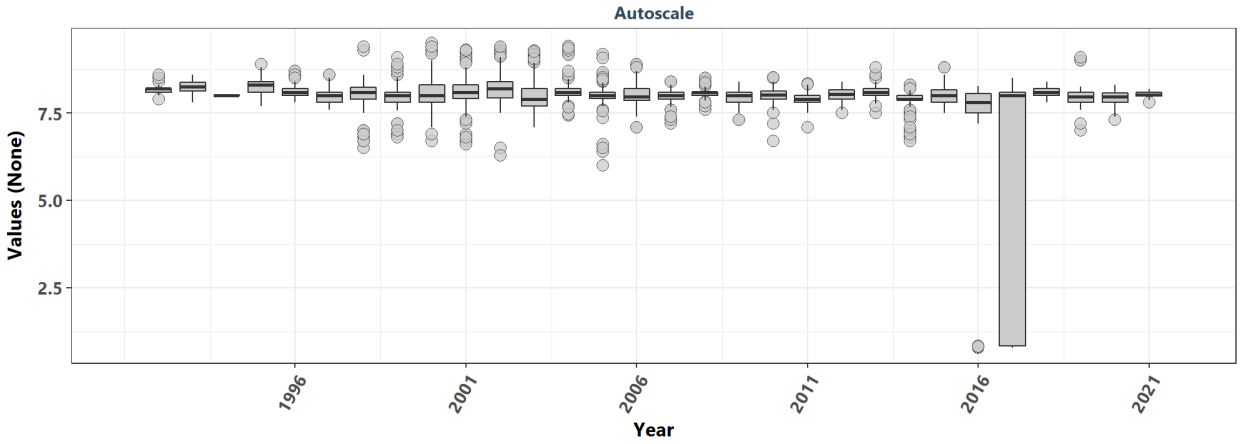

Indian River-Malabar to Vero Beach Aquatic Preserve
By Year
Autoscale
Values (None)
10.0
7.5
5.0
2.5
0.0
1981
1991
2001
2011
2021
Year
Scaled to 4x Standard Deviation
Values (None)
10
5
0
1981
1991
2001
2011
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
10
5
0
2011
2013
2015
2017
2019
2021
Year

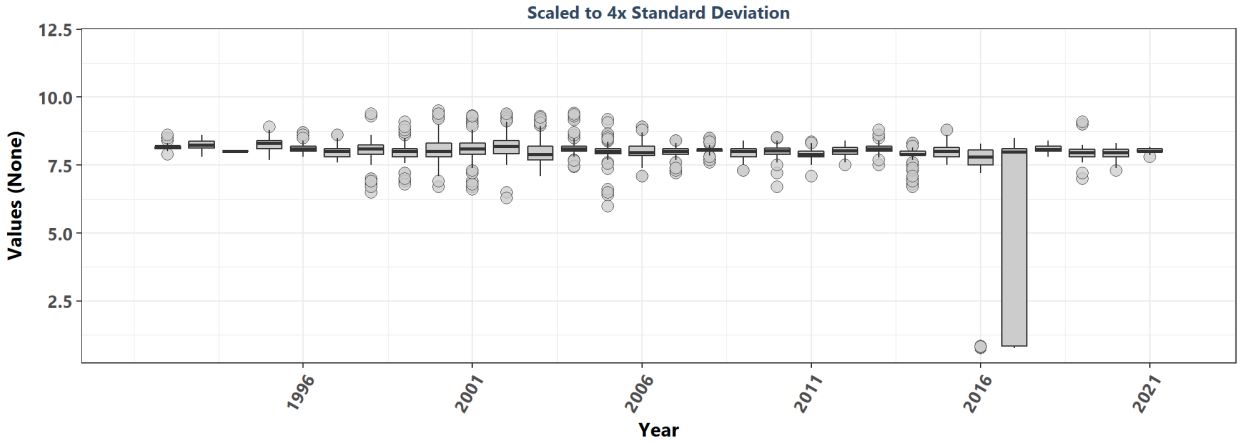

Indian River-Malabar to Vero Beach Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
10.0
7.5
5.0
2.5
0.0
1981
1991
2001
2011
2021
Year
Scaled to 4x Standard Deviation
Values (None)
10
5
0
1981
1991
2001
2011
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
10
5
0
2011
2013
2015
2017
2019
2021
Year

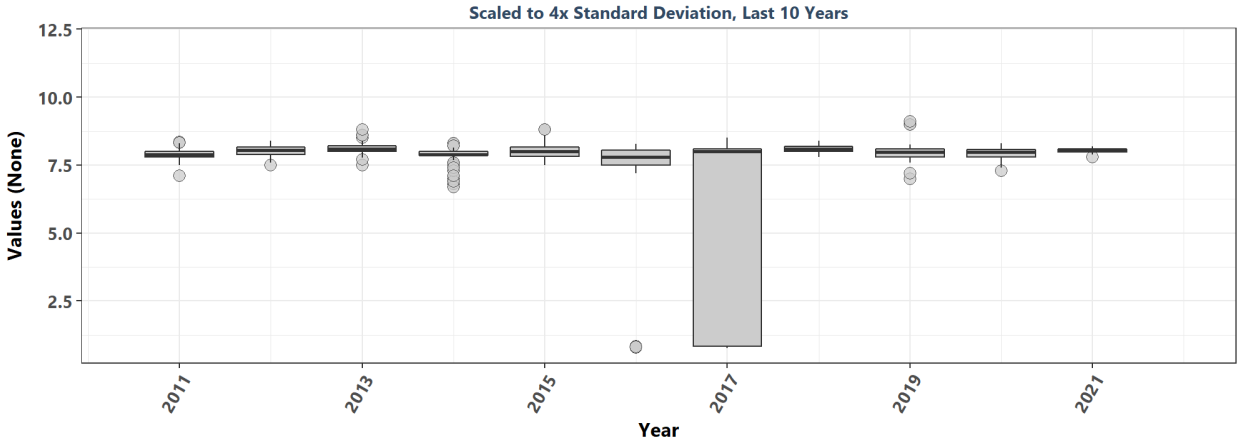

Indian River-Malabar to Vero Beach Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
10.0
7.5
5.0
2.5
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
10
5
0
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
10
5
0
3
6
9
12
Month

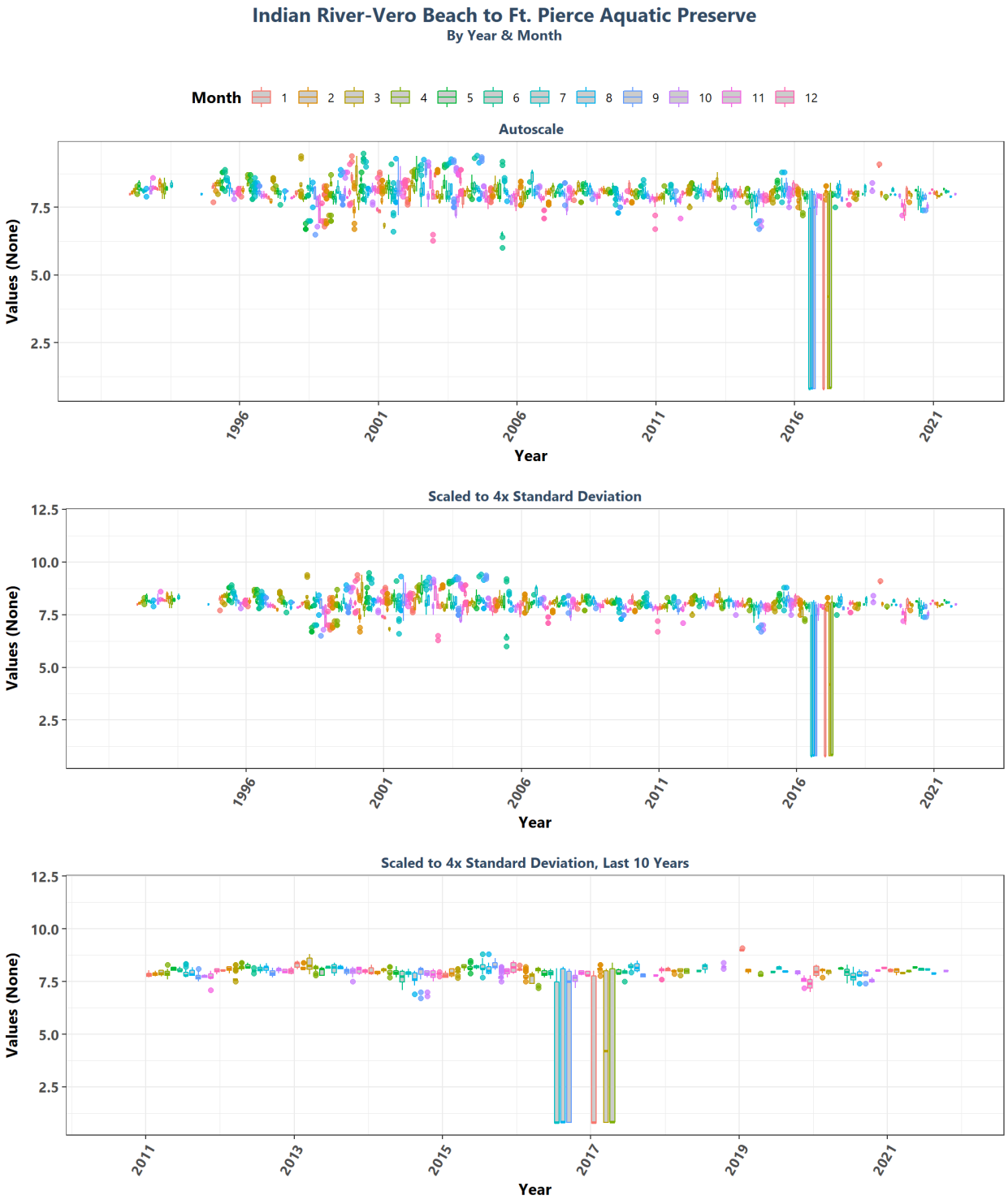
Indian River-Vero Beach to Ft. Pierce Aquatic Preserve

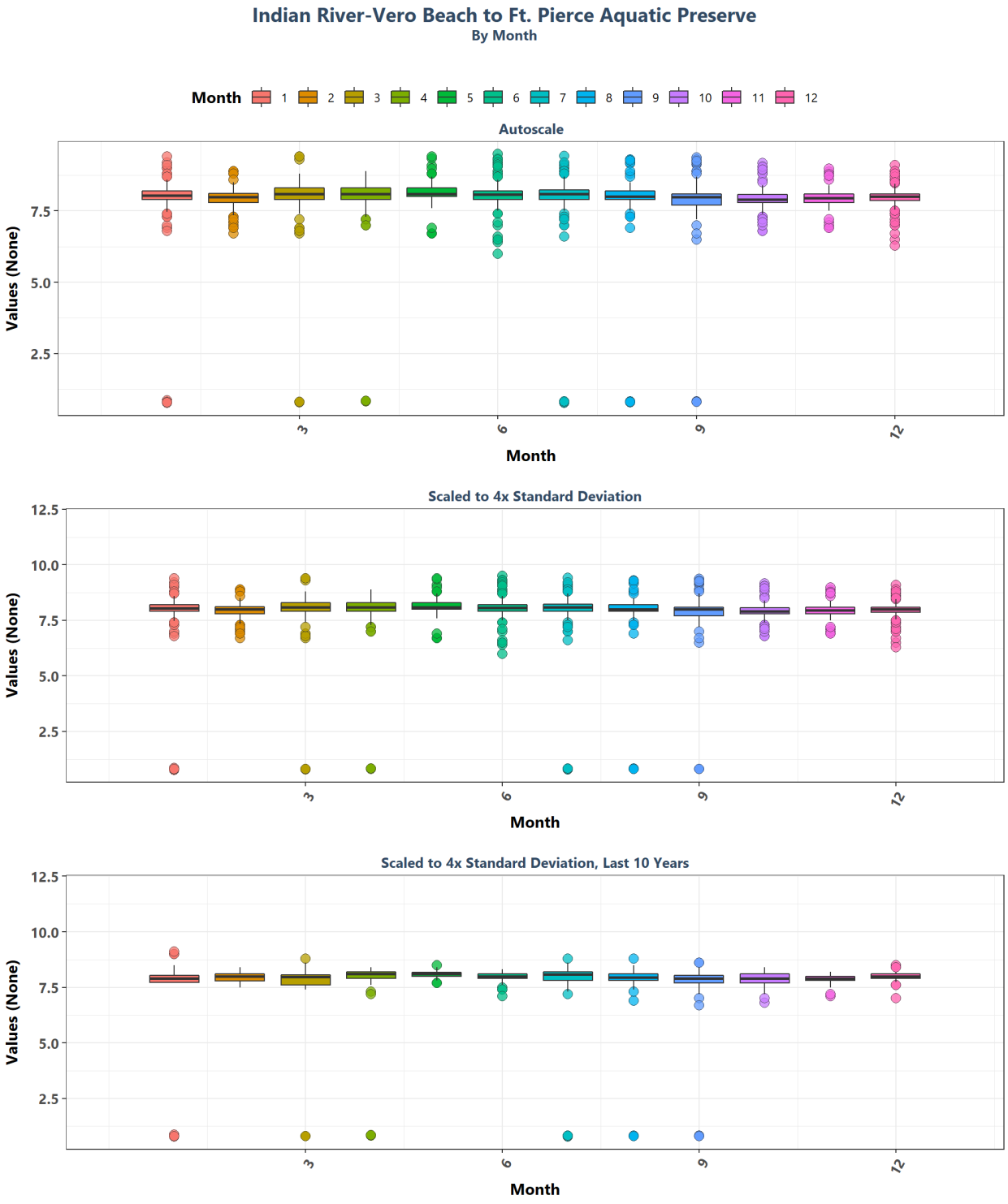
By Year

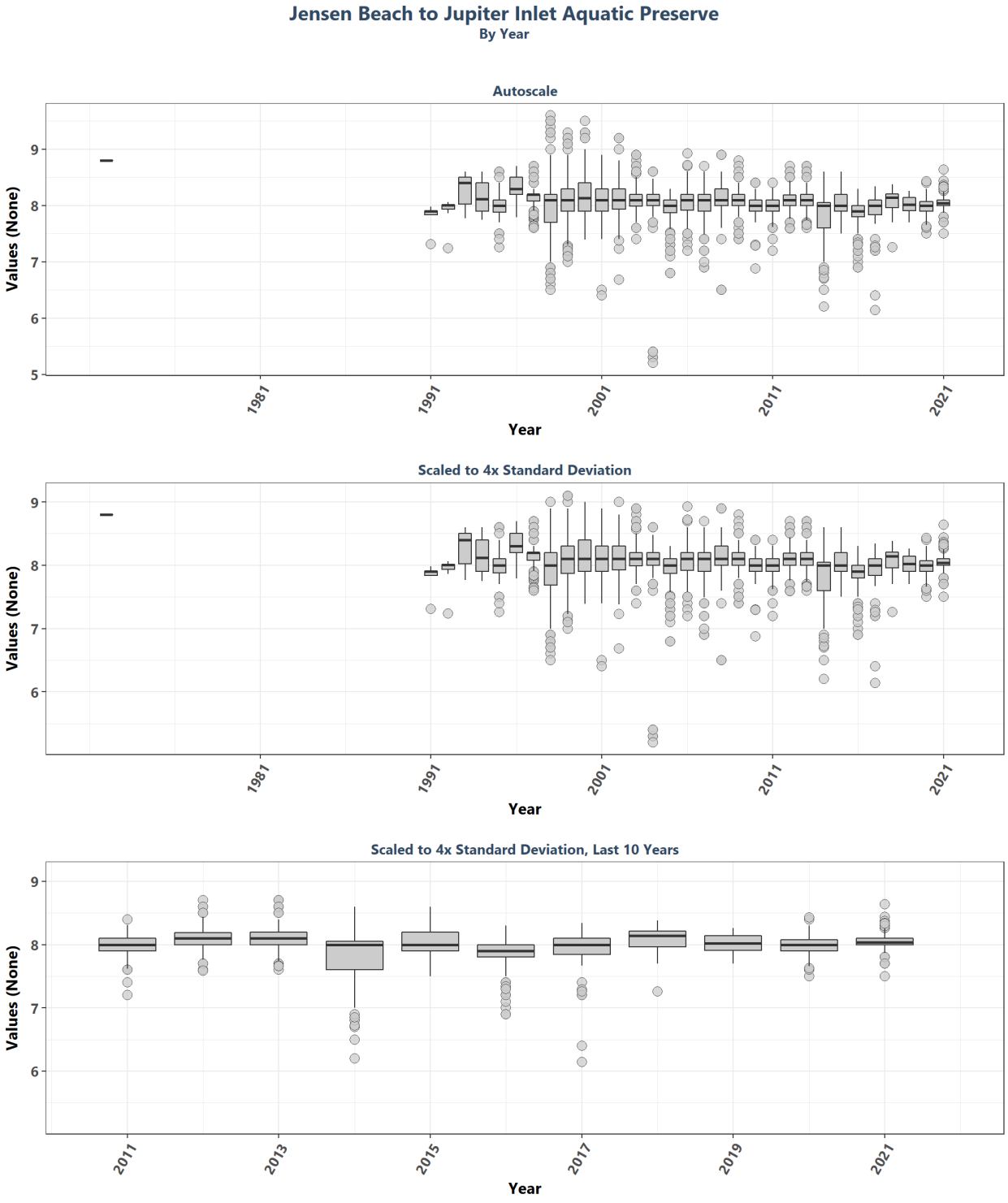


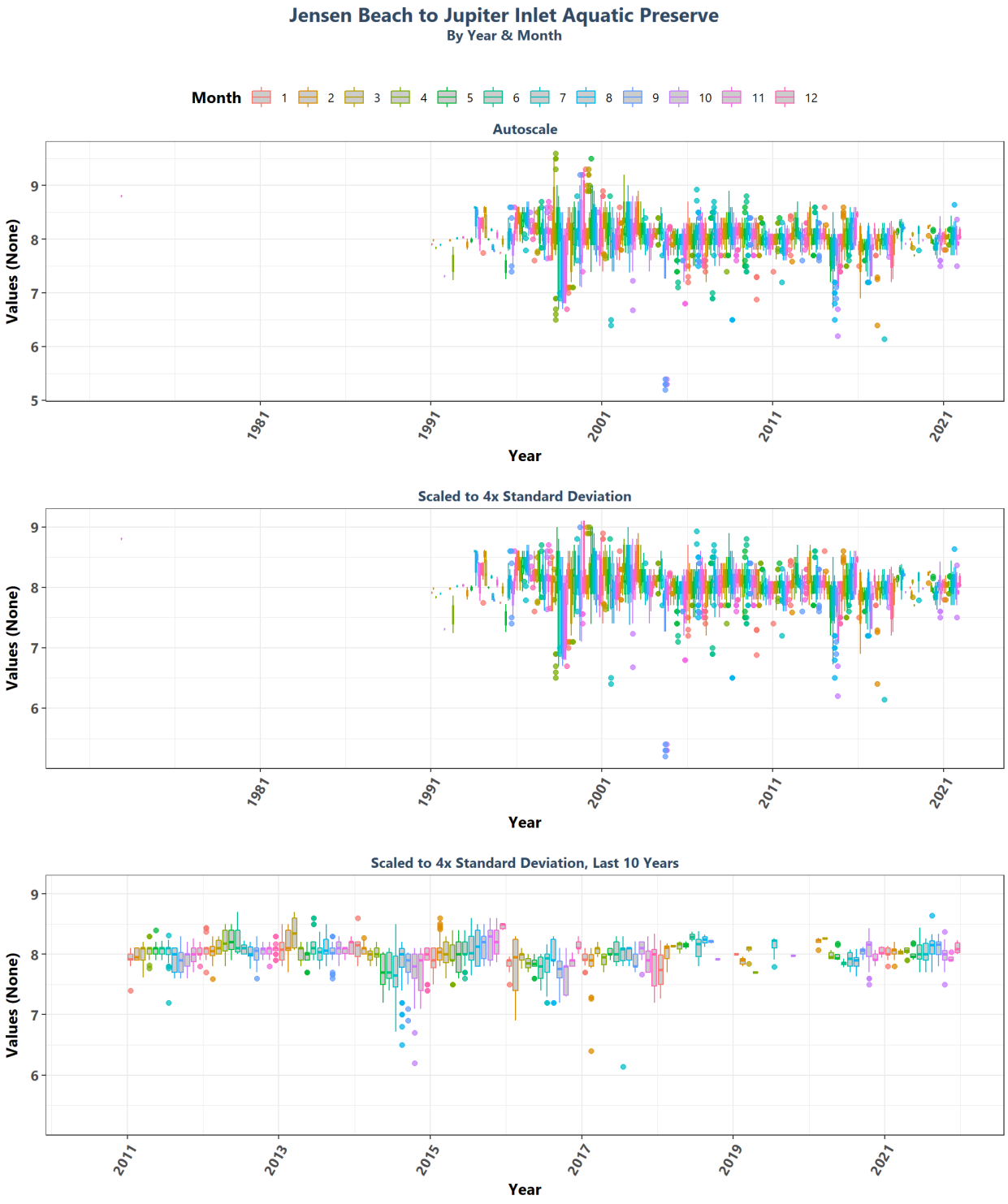


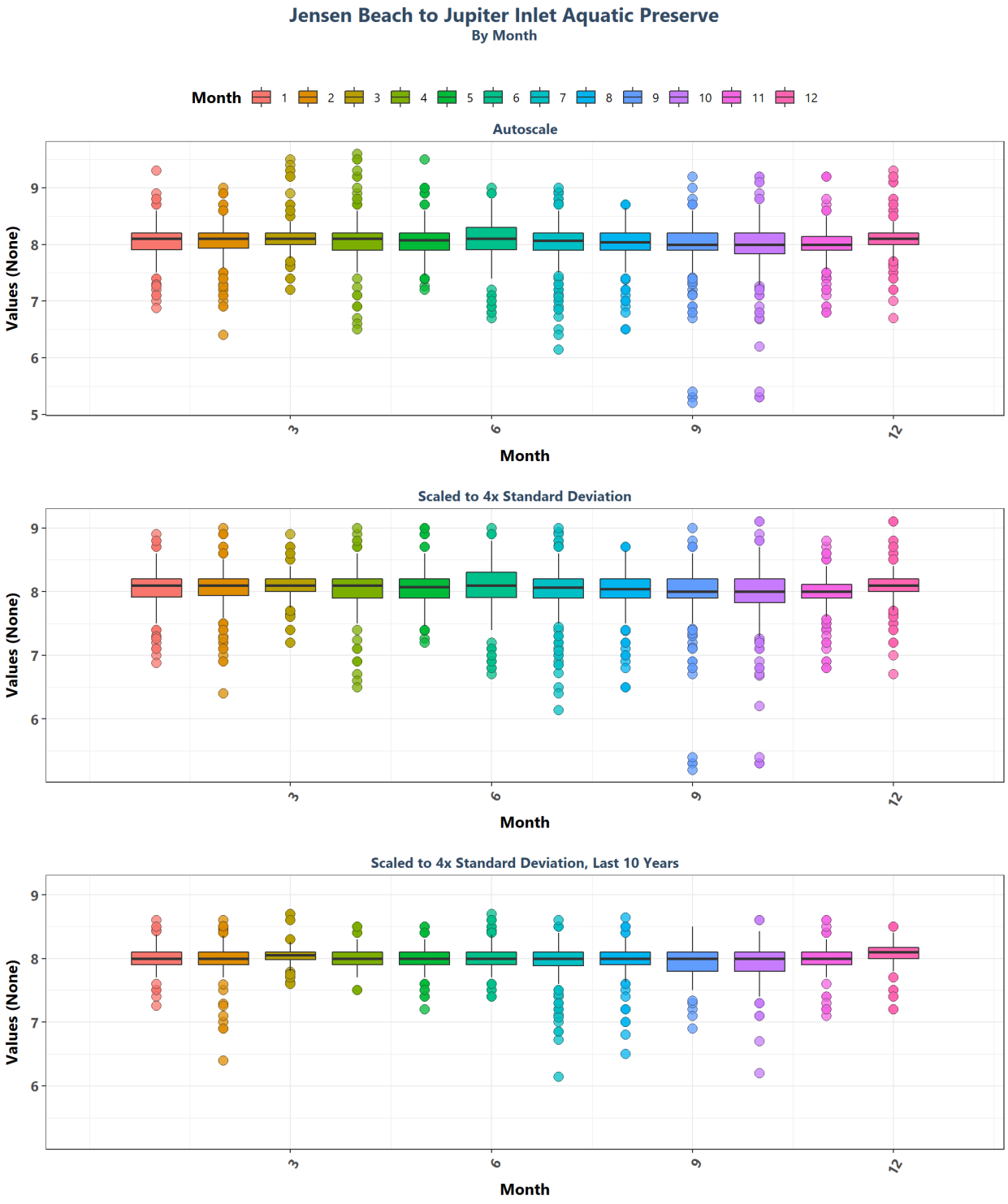


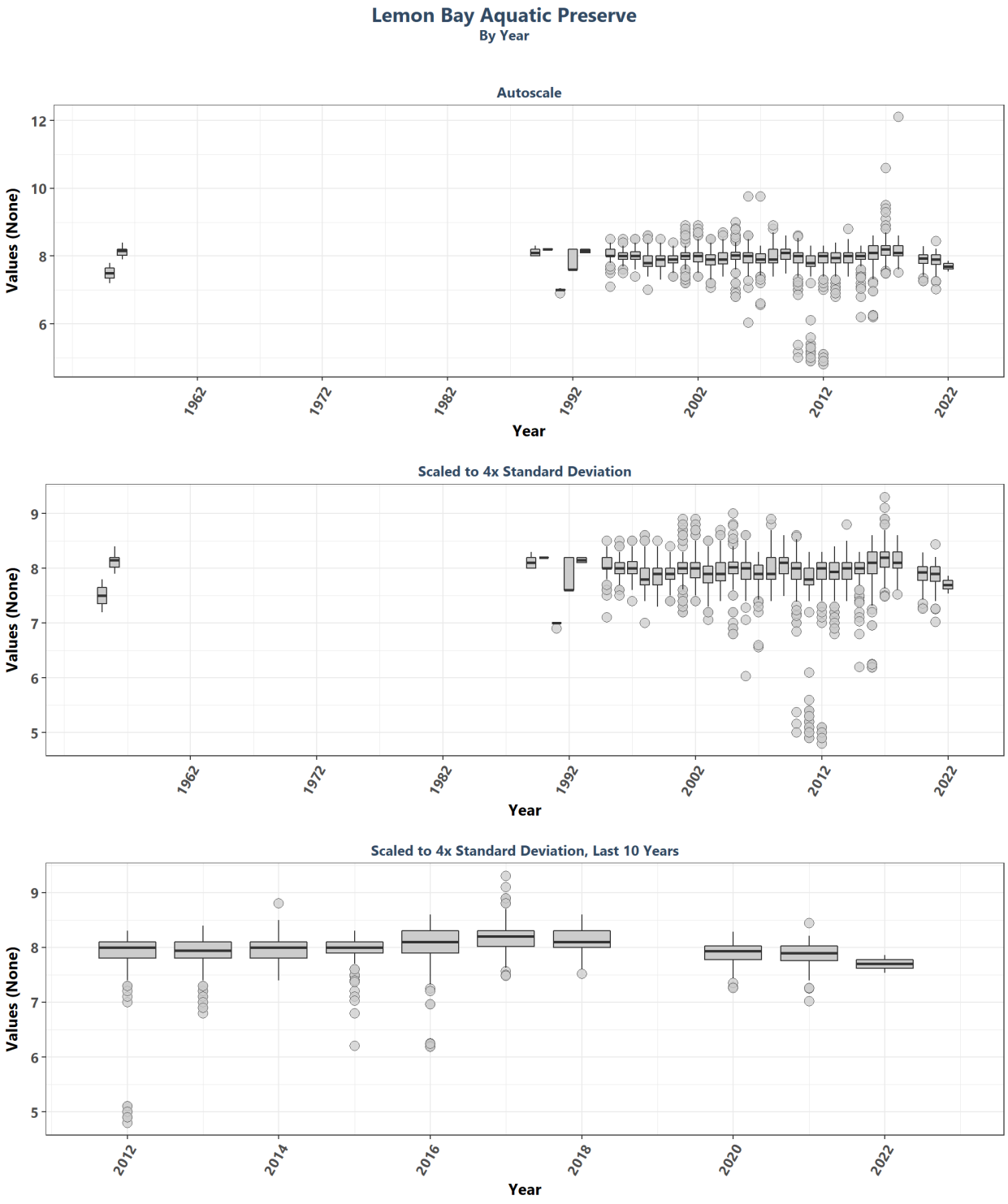

Indian River-Vero Beach to Ft. Pierce Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

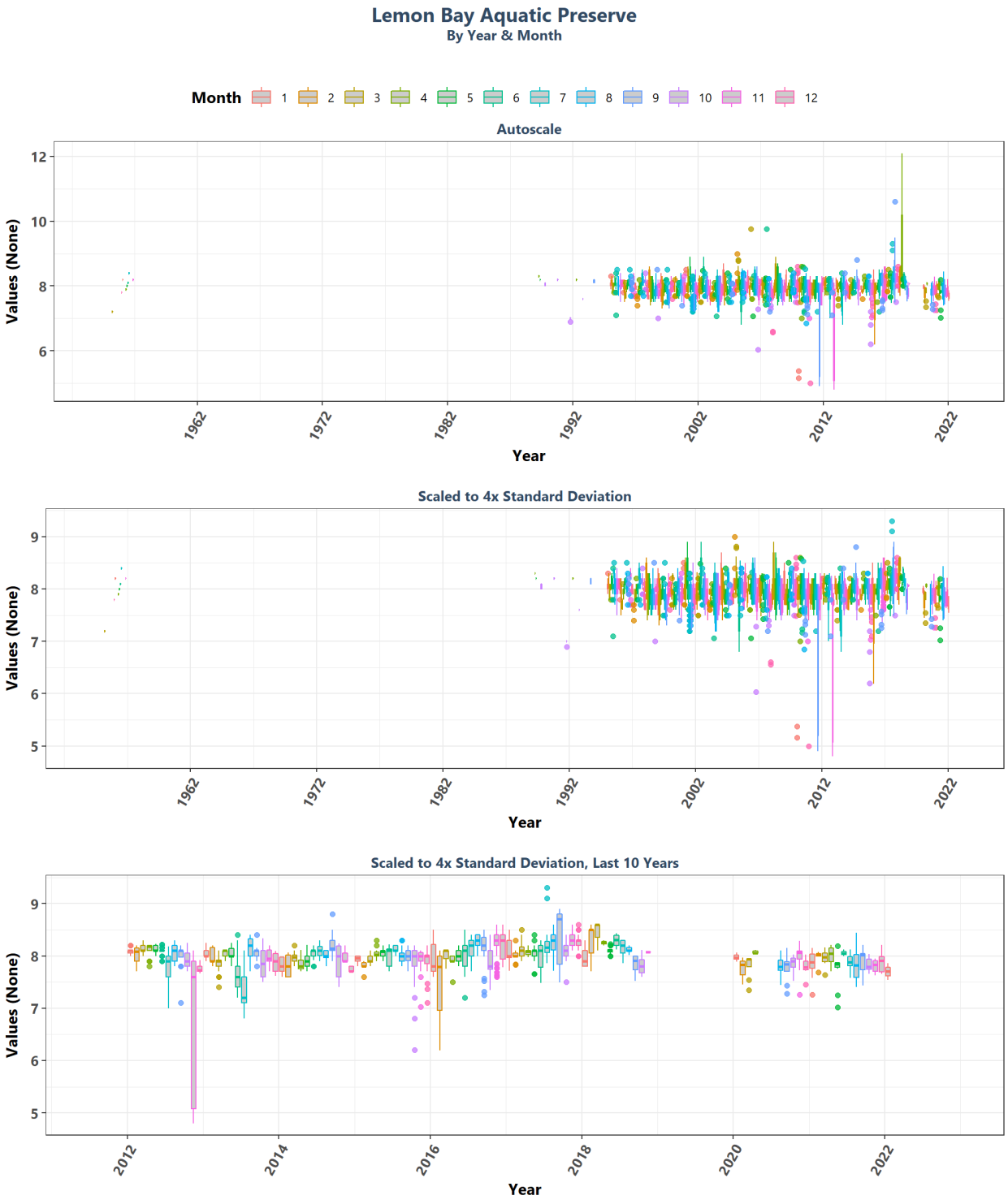

Indian River-Vero Beach to Ft. Pierce Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
7.5
5.0
2.5
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
12.5
10.0
7.5
5.0
2.5
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
12.5
10.0
7.5
5.0
2.5
3
6
9
12
Month

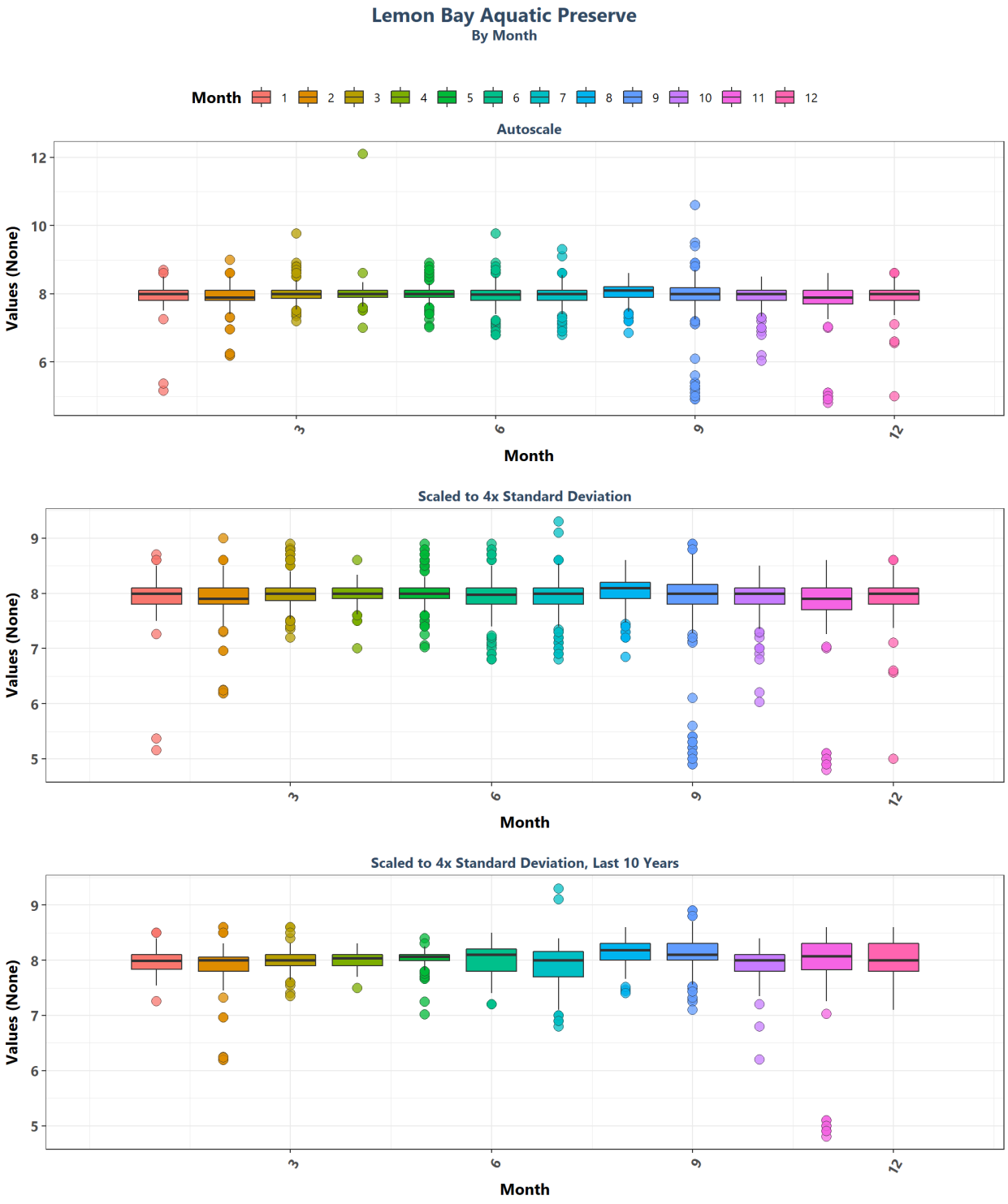

Jensen Beach to Jupiter Inlet Aquatic Preserve
By Year
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year


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 (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
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 (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

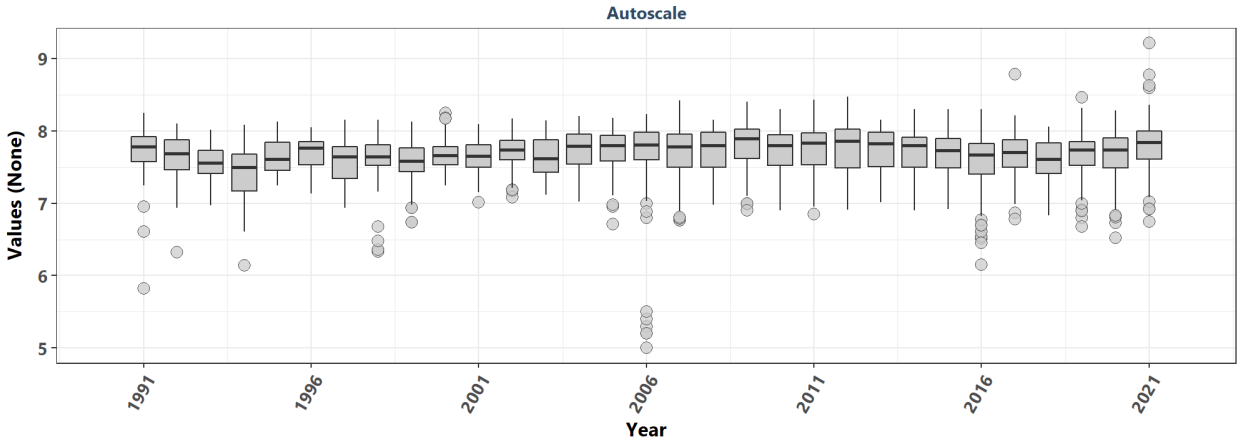

Lemon Bay Aquatic Preserve
By Year
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

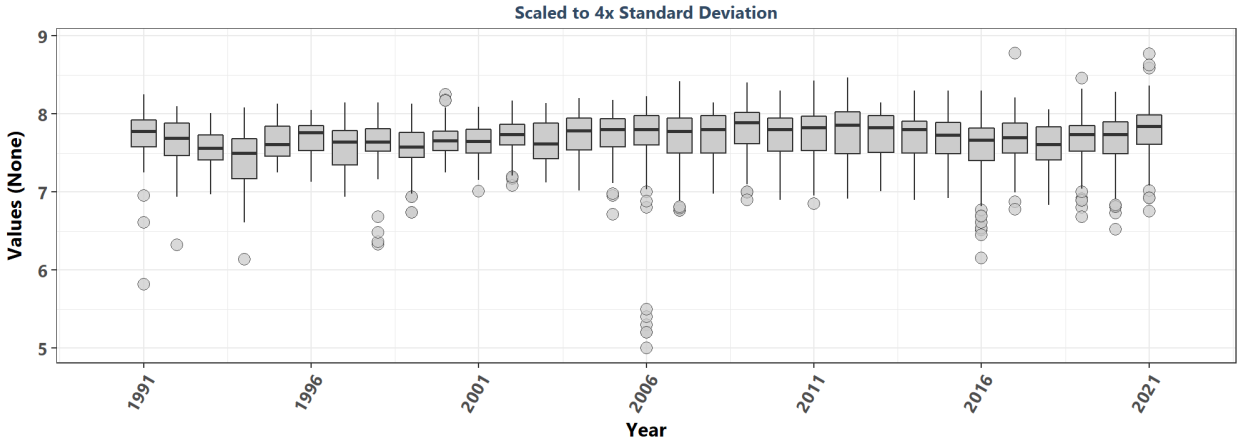

Lemon Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

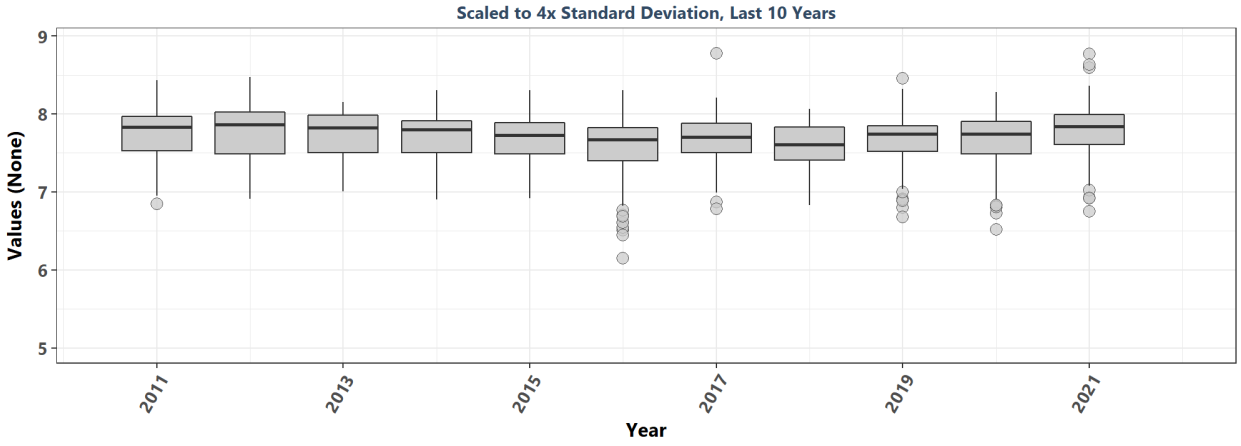

Lemon Bay Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

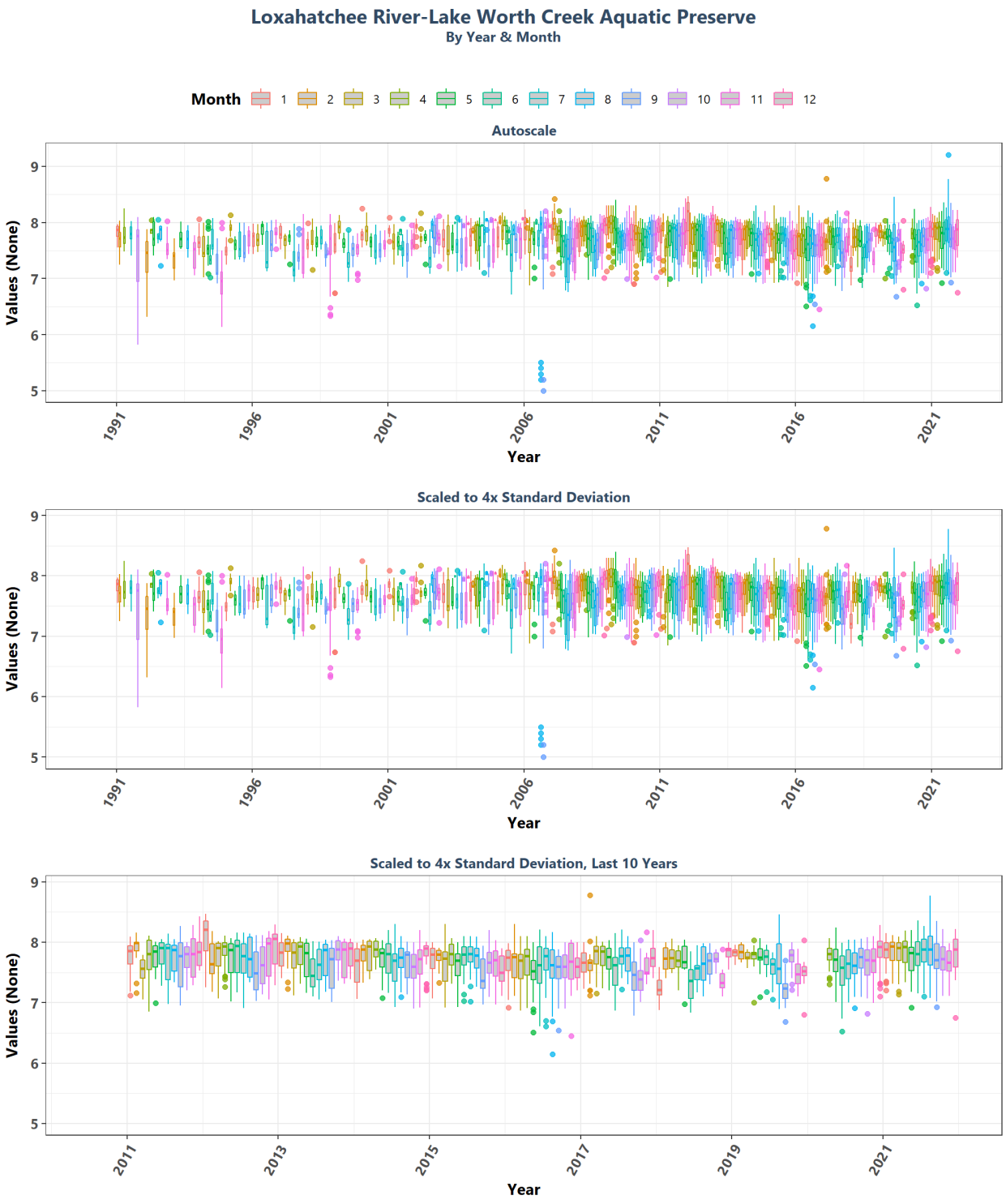
Loxahatchee River-Lake Worth Creek Aquatic Preserve

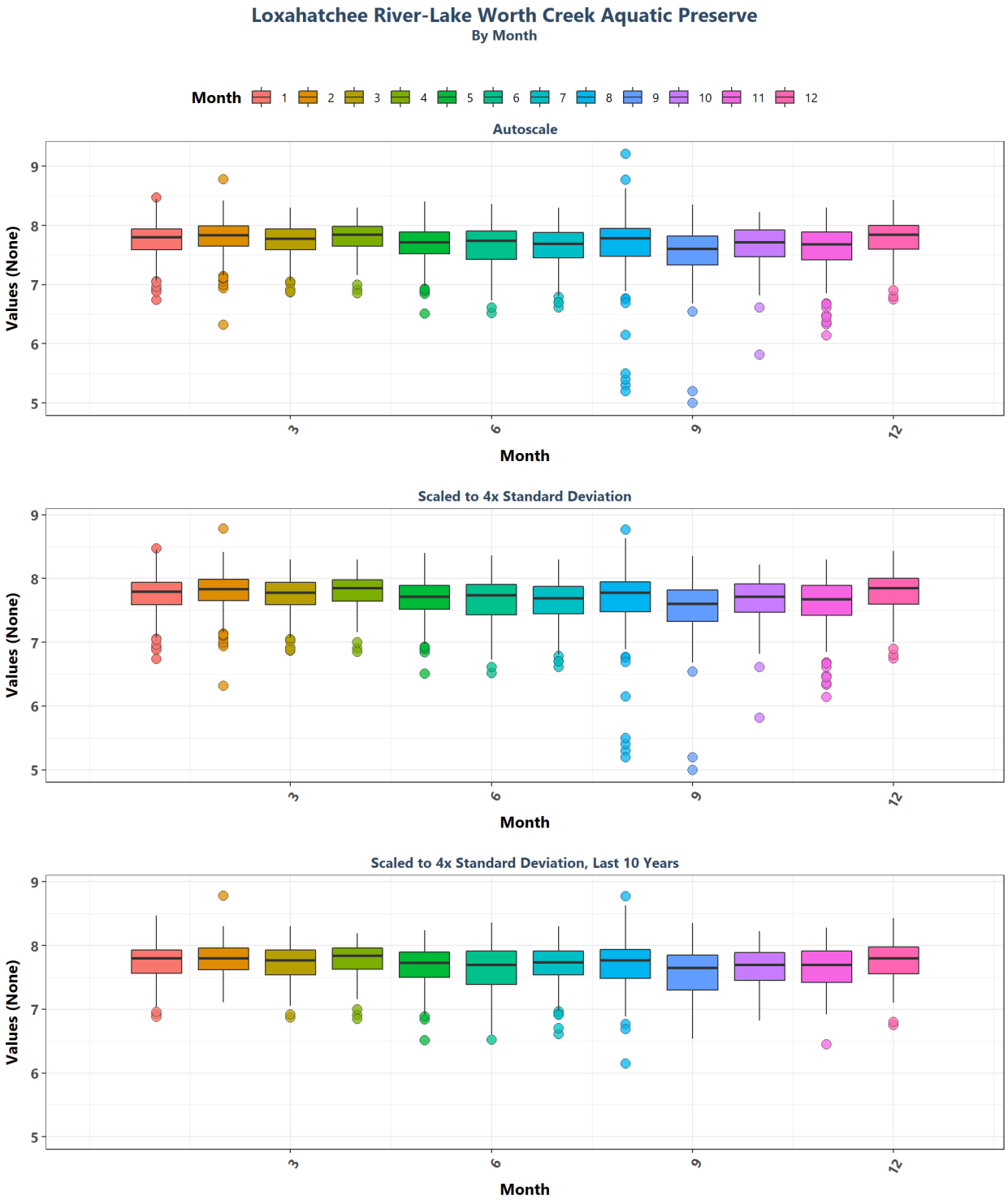
By Year





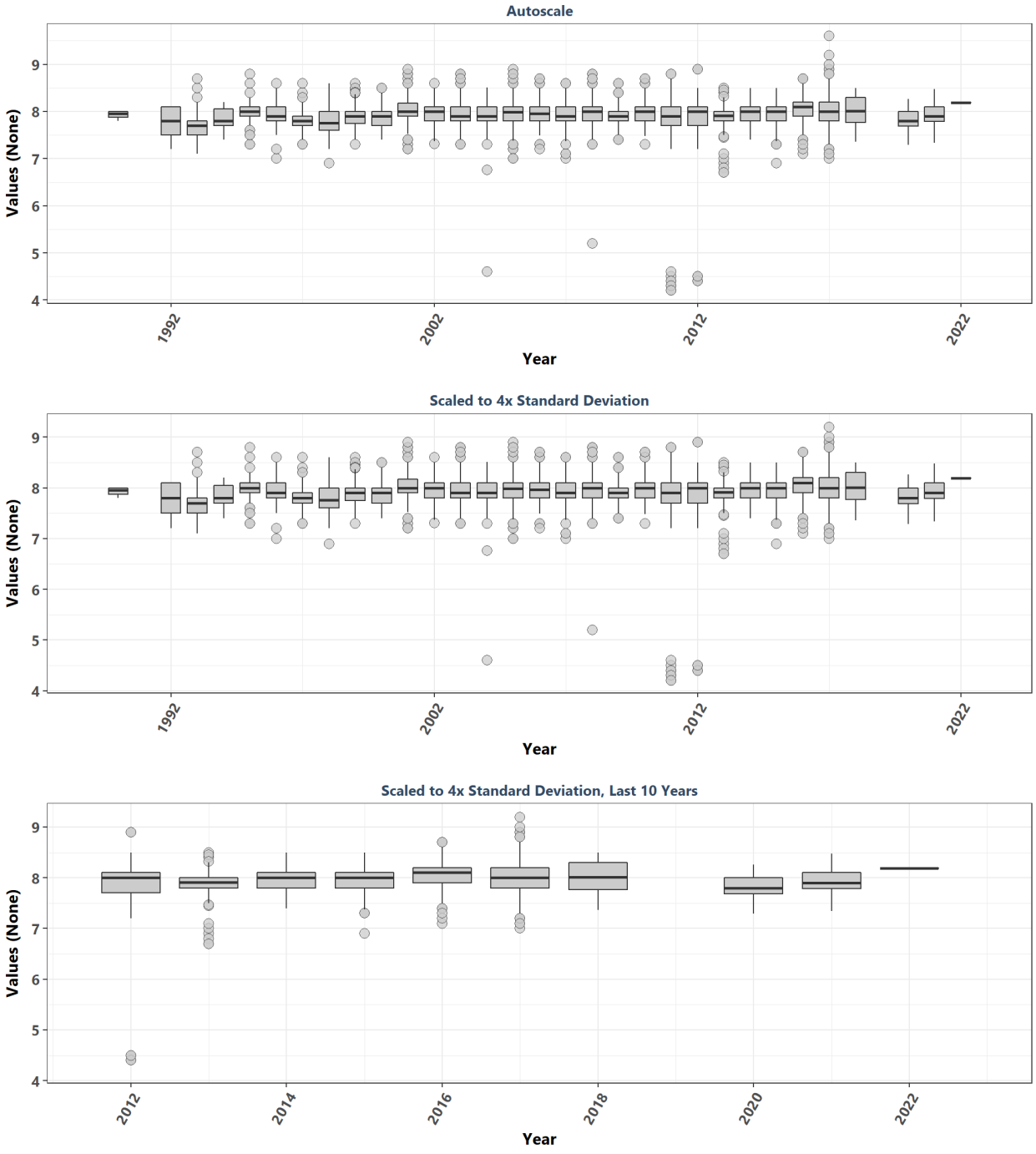


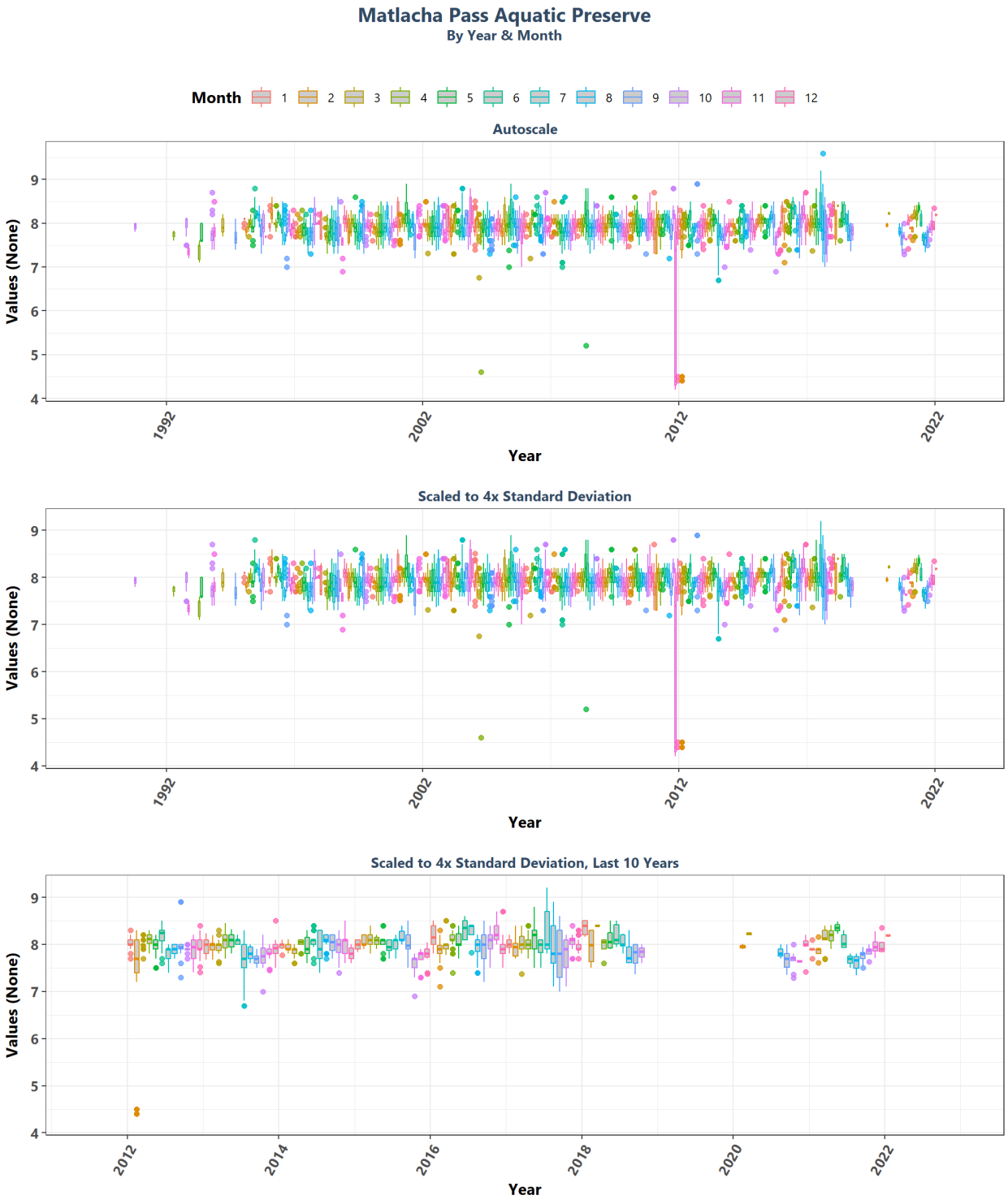

Loxahatchee River-Lake Worth Creek Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

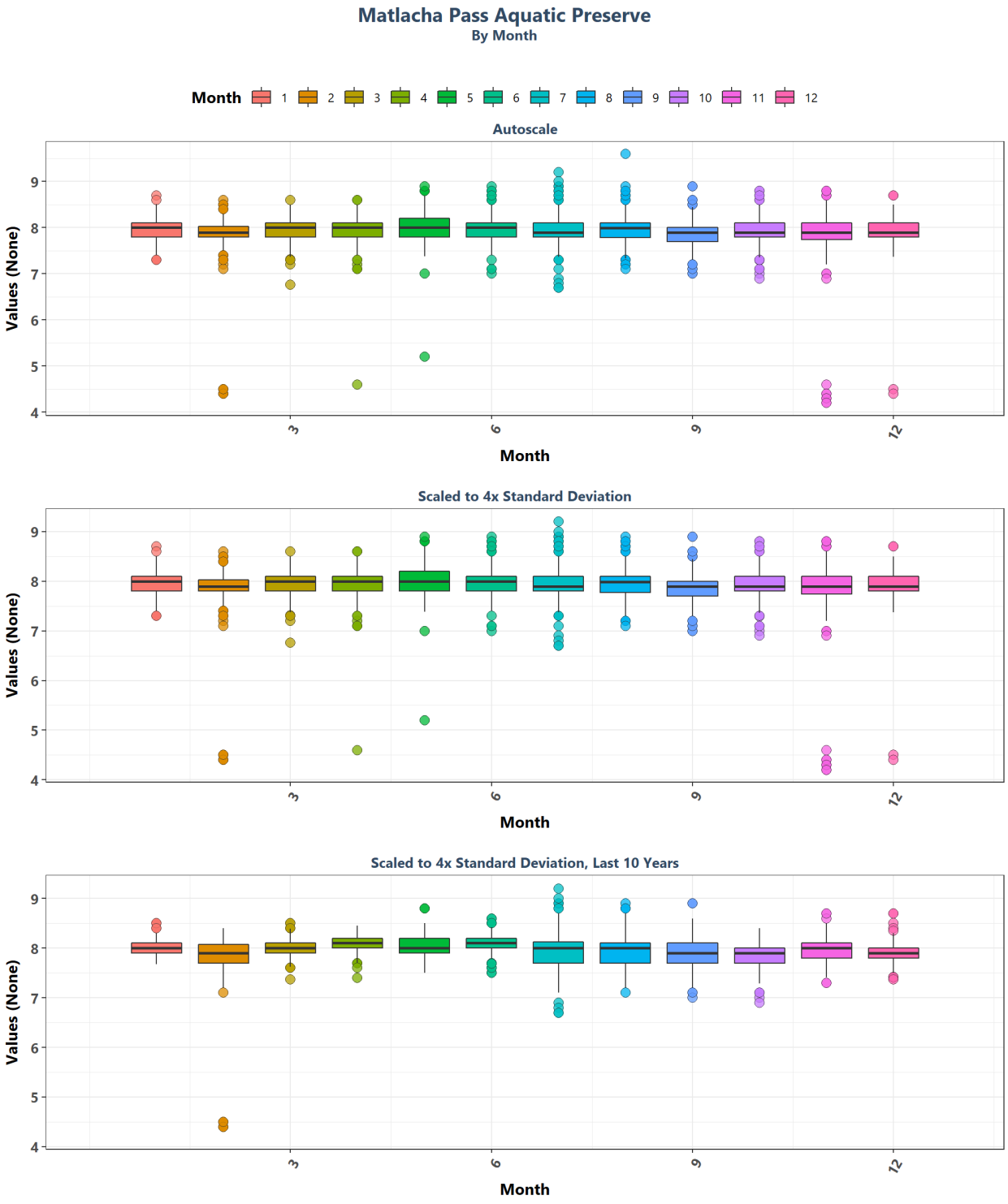

Loxahatchee River-Lake Worth Creek Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

Matlacha Pass Aquatic Preserve

By Year

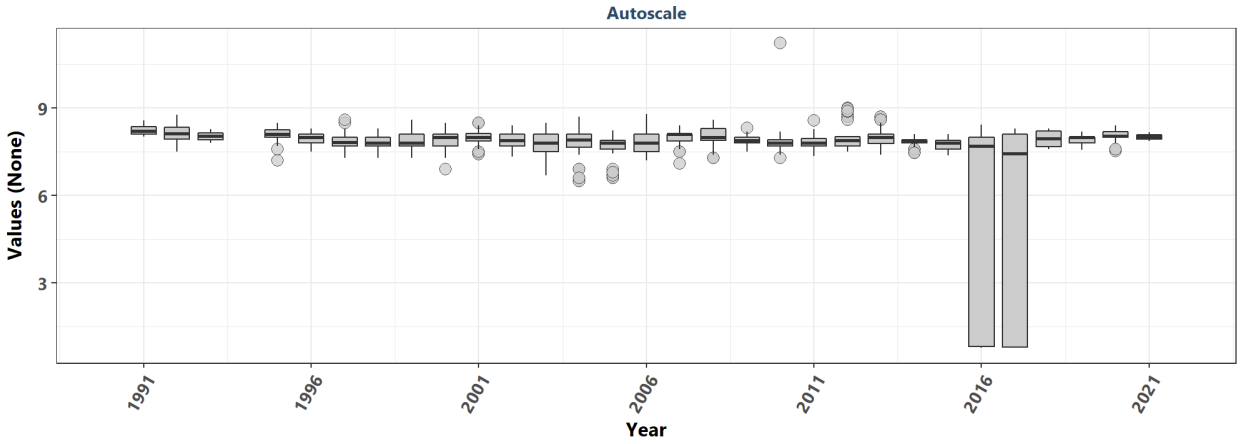


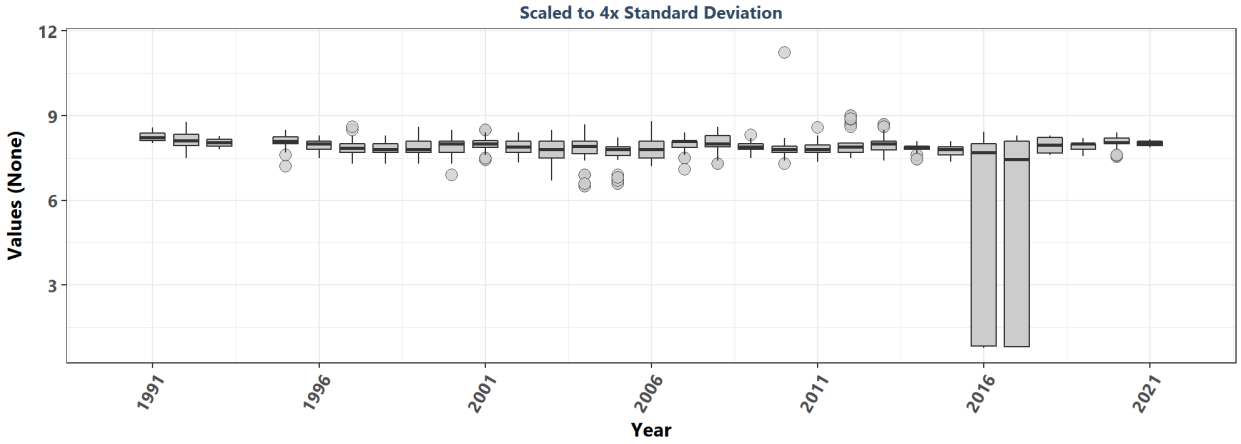

Matlacha Pass Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

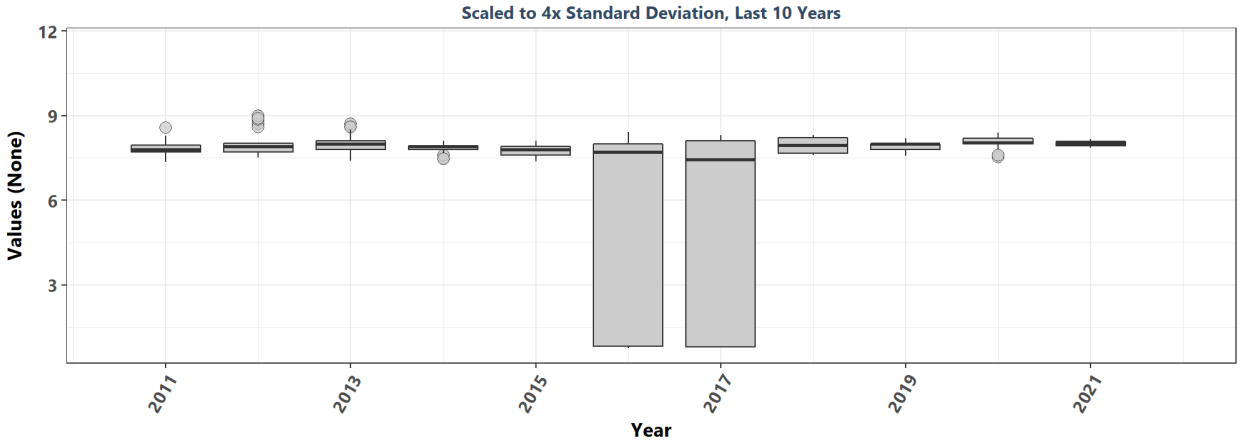

Matlacha Pass Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

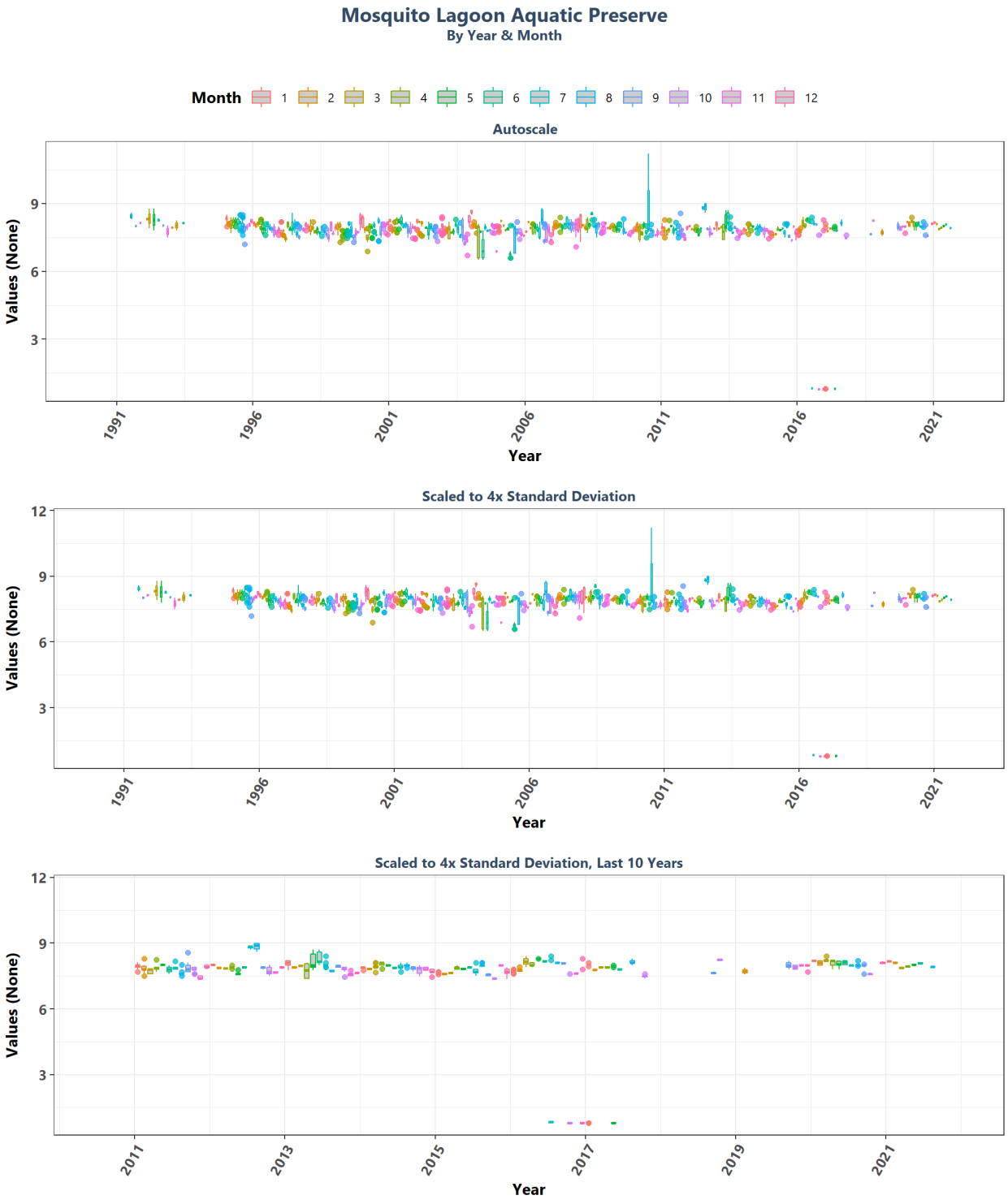
Mosquito Lagoon Aquatic Preserve

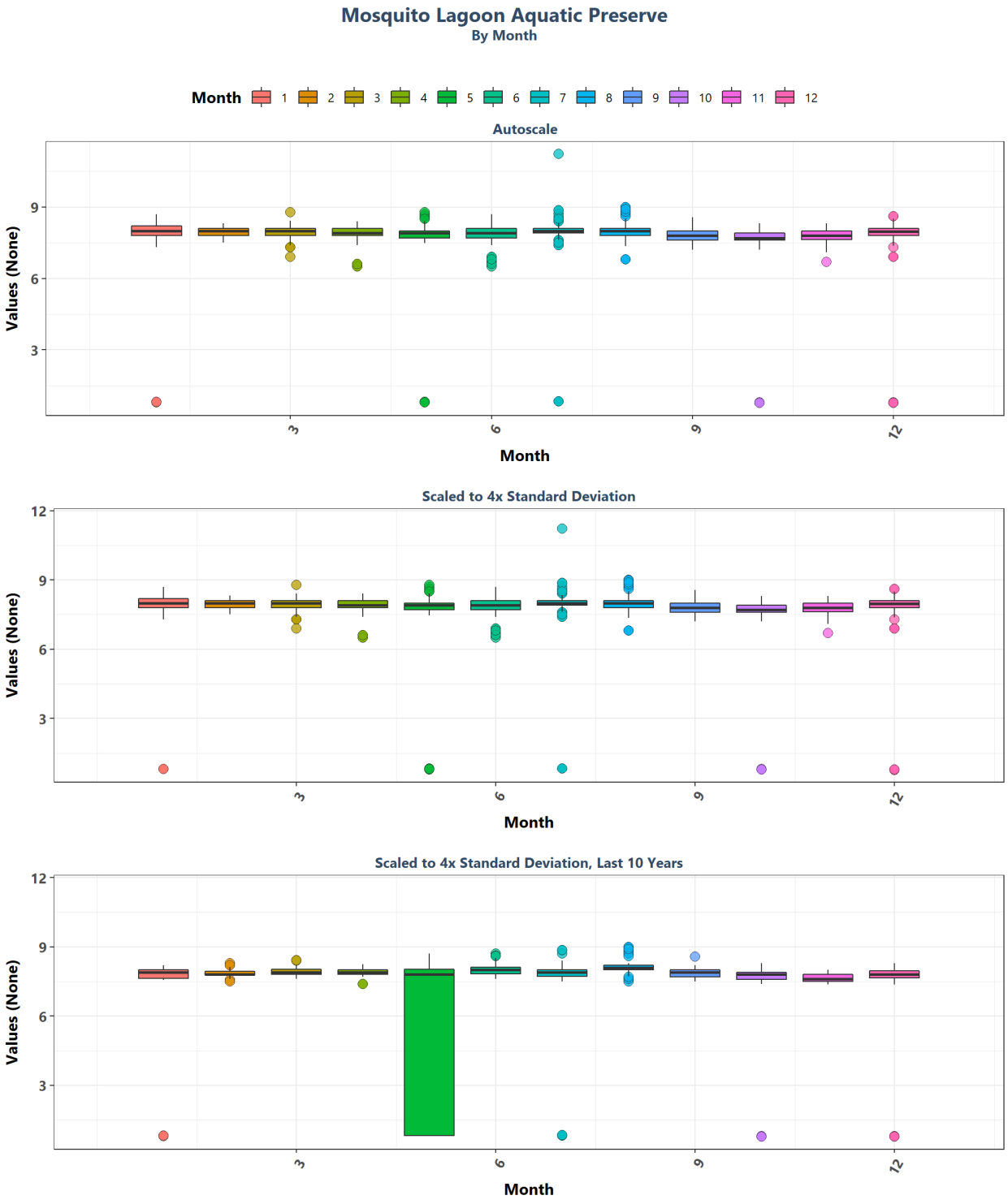
By Year

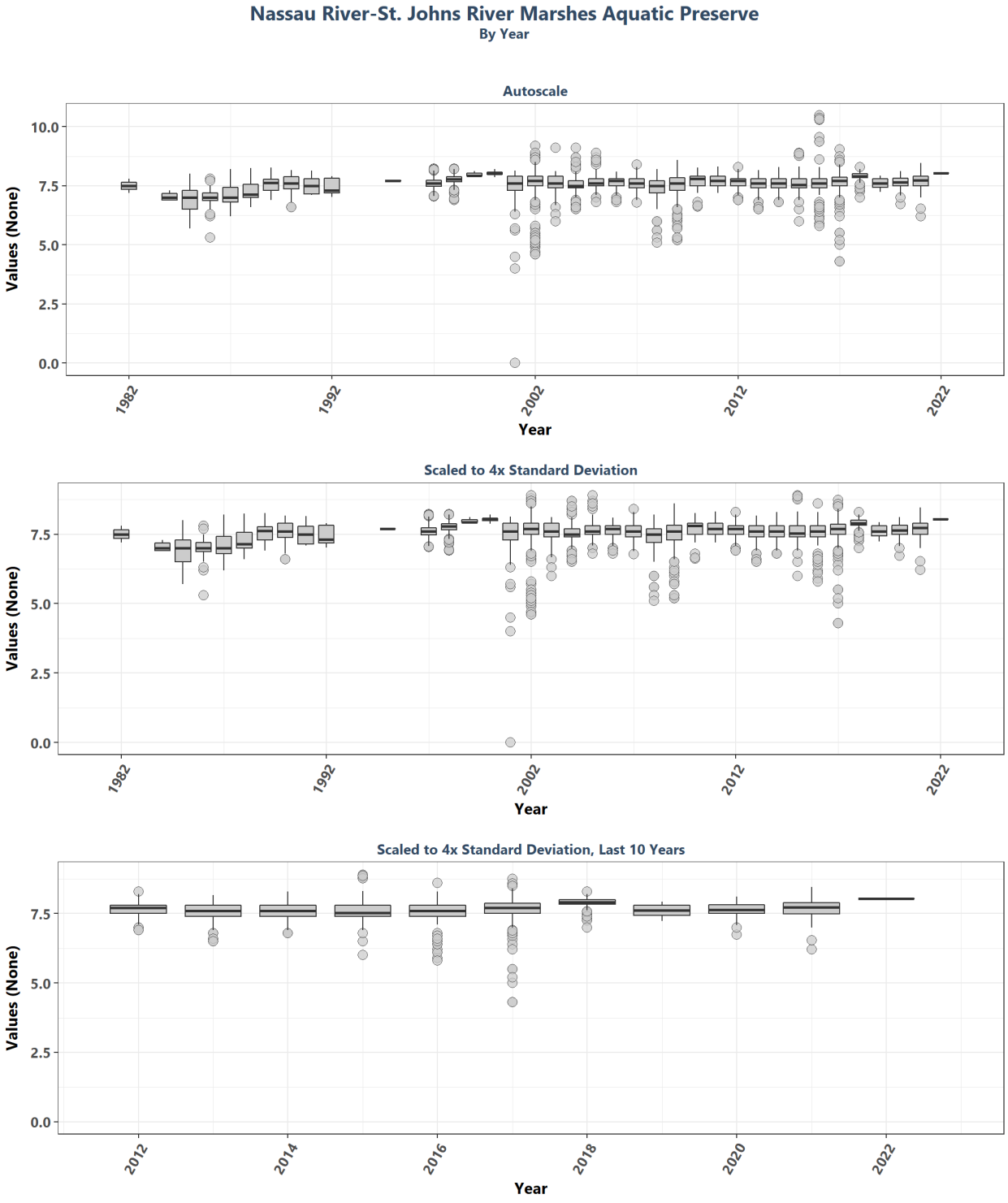


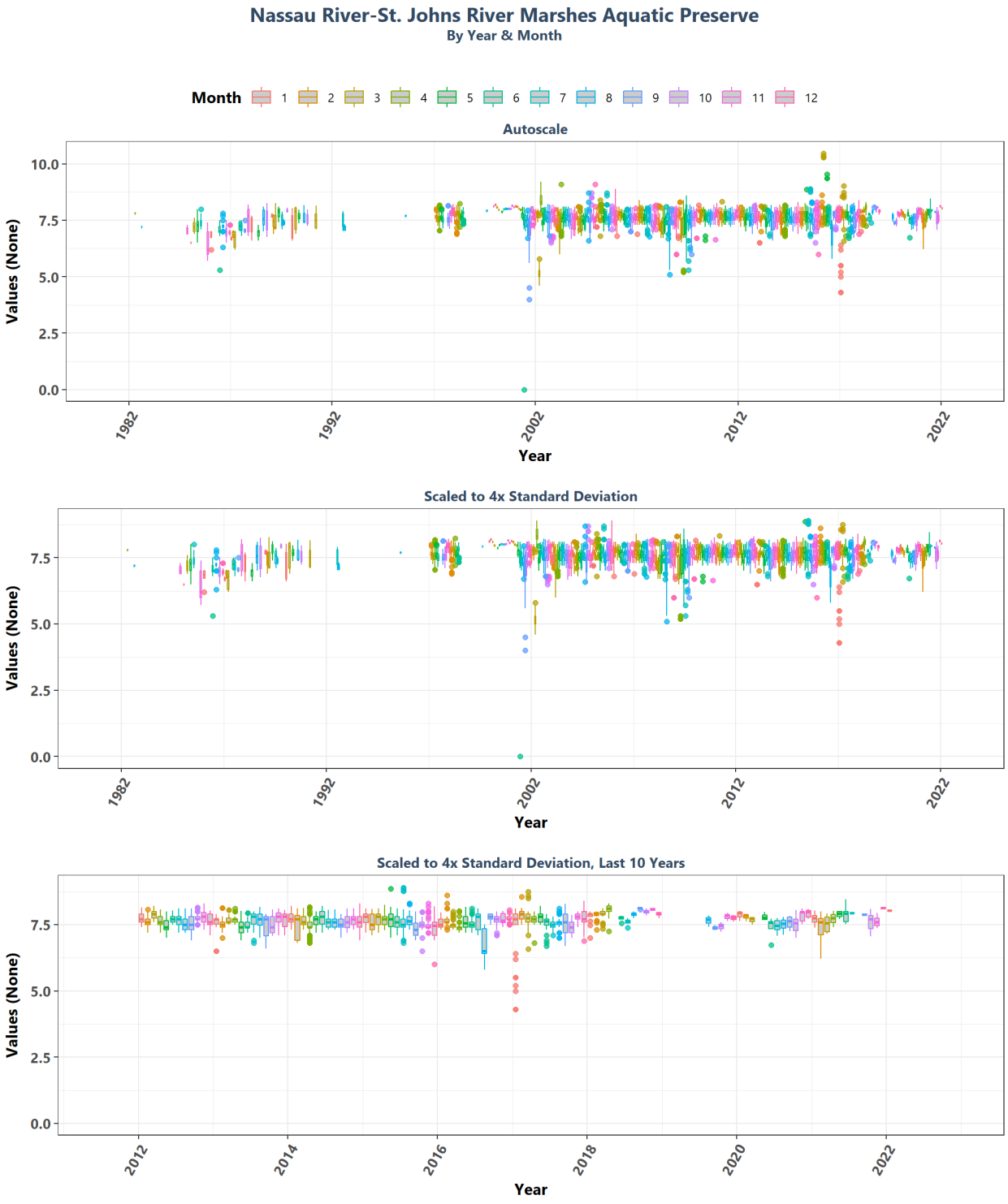


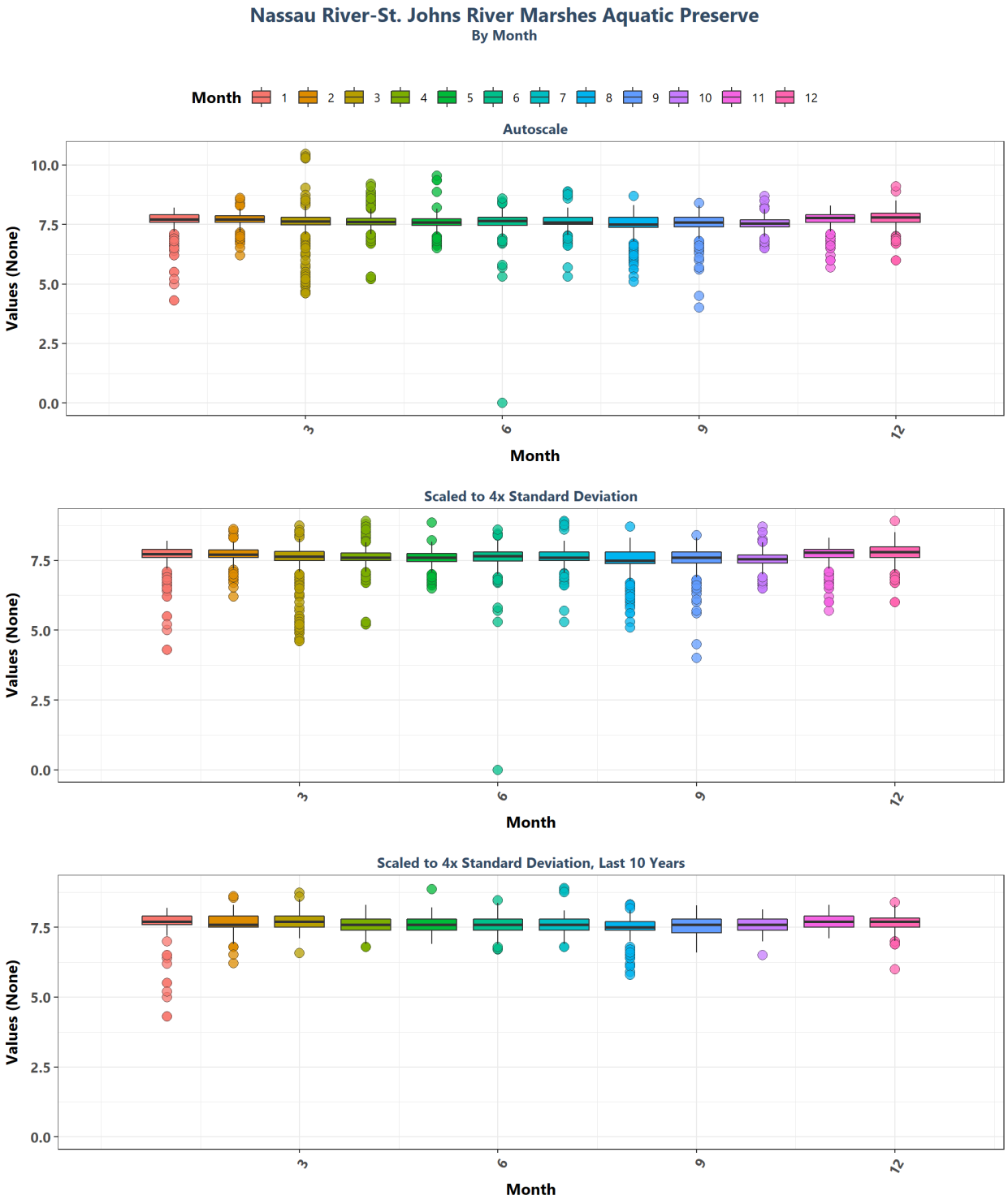


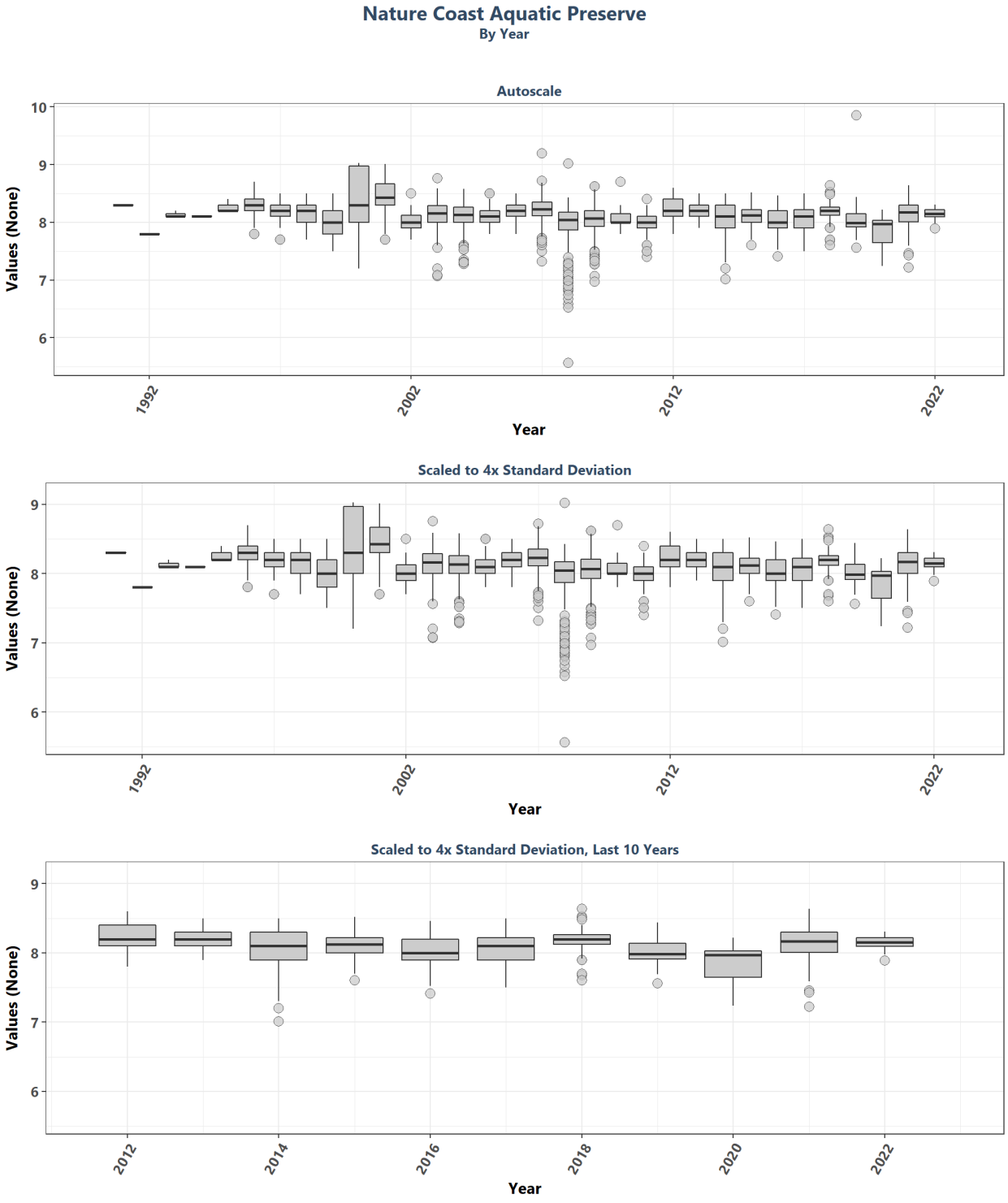

Mosquito Lagoon Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

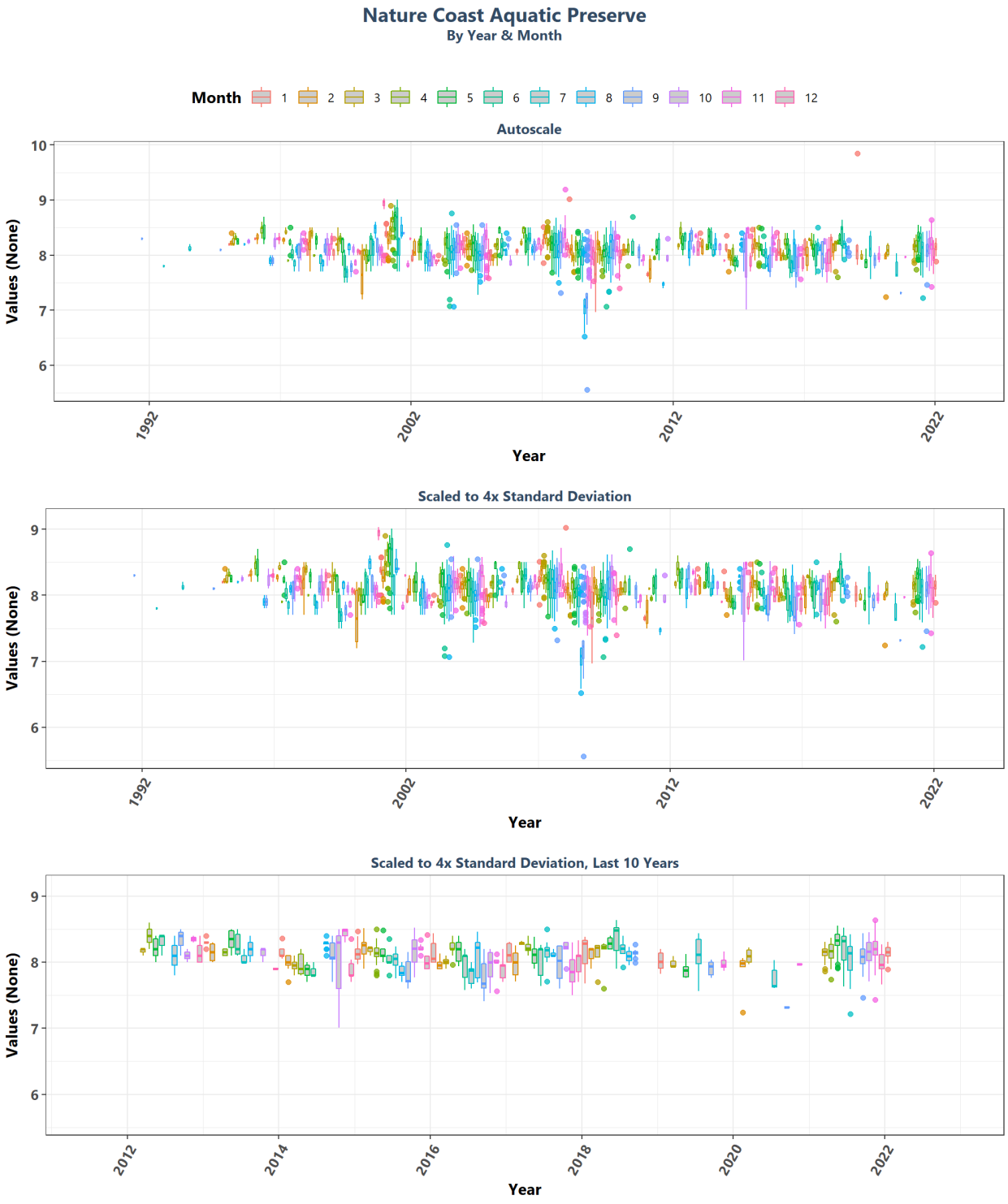

Mosquito Lagoon Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
9
6
3
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
12
9
6
3
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
12
9
6
3
3
6
9
12
Month

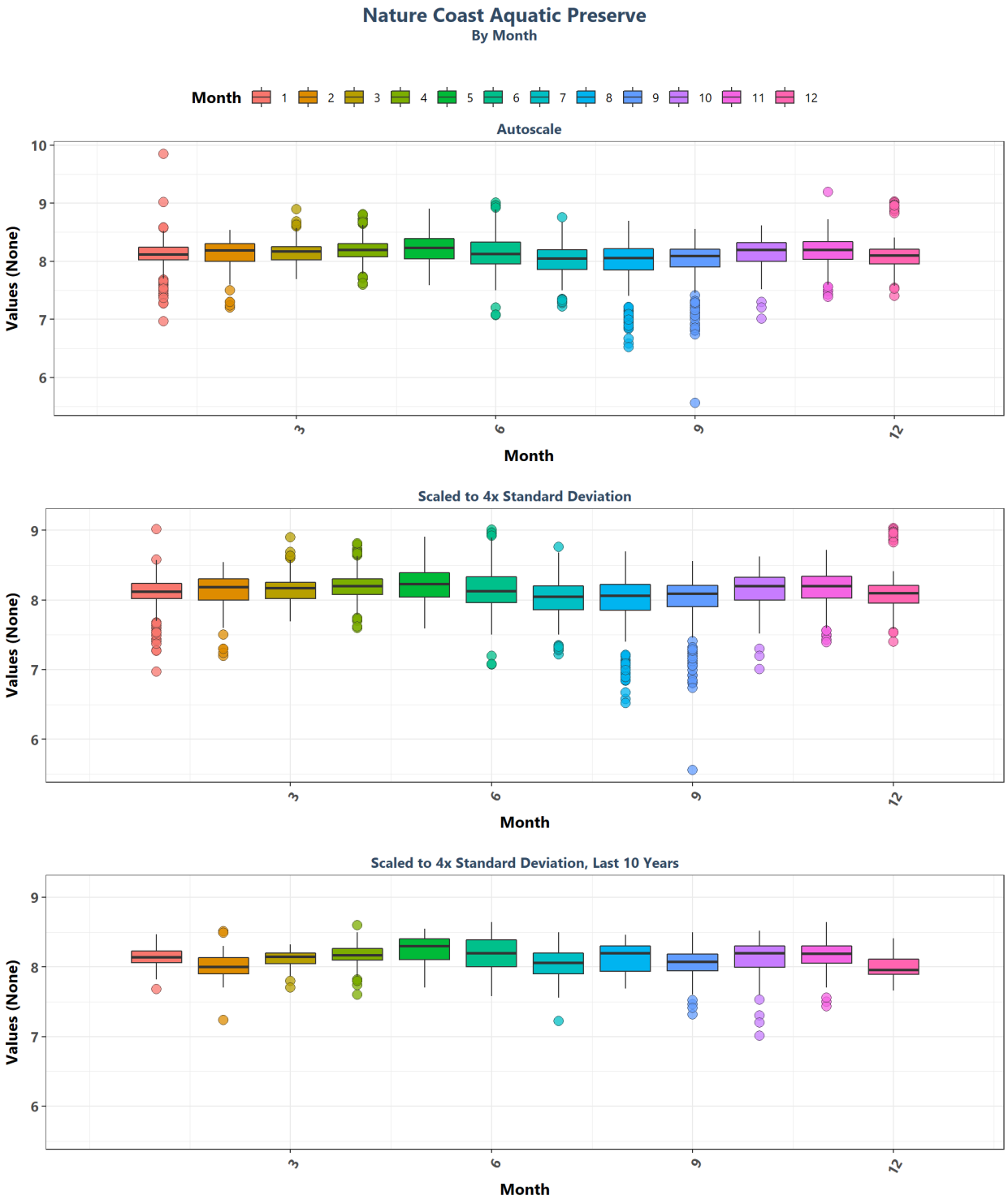

Nassau River-St. Johns River Marshes Aquatic Preserve
By Year
Autoscale
Values (None)
10.0
7.5
5.0
2.5
0.0
1982
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation
Values (None)
7.5
5.0
2.5
0.0
1982
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
7.5
5.0
2.5
0.0
2012
2014
2016
2018
2020
2022
Year

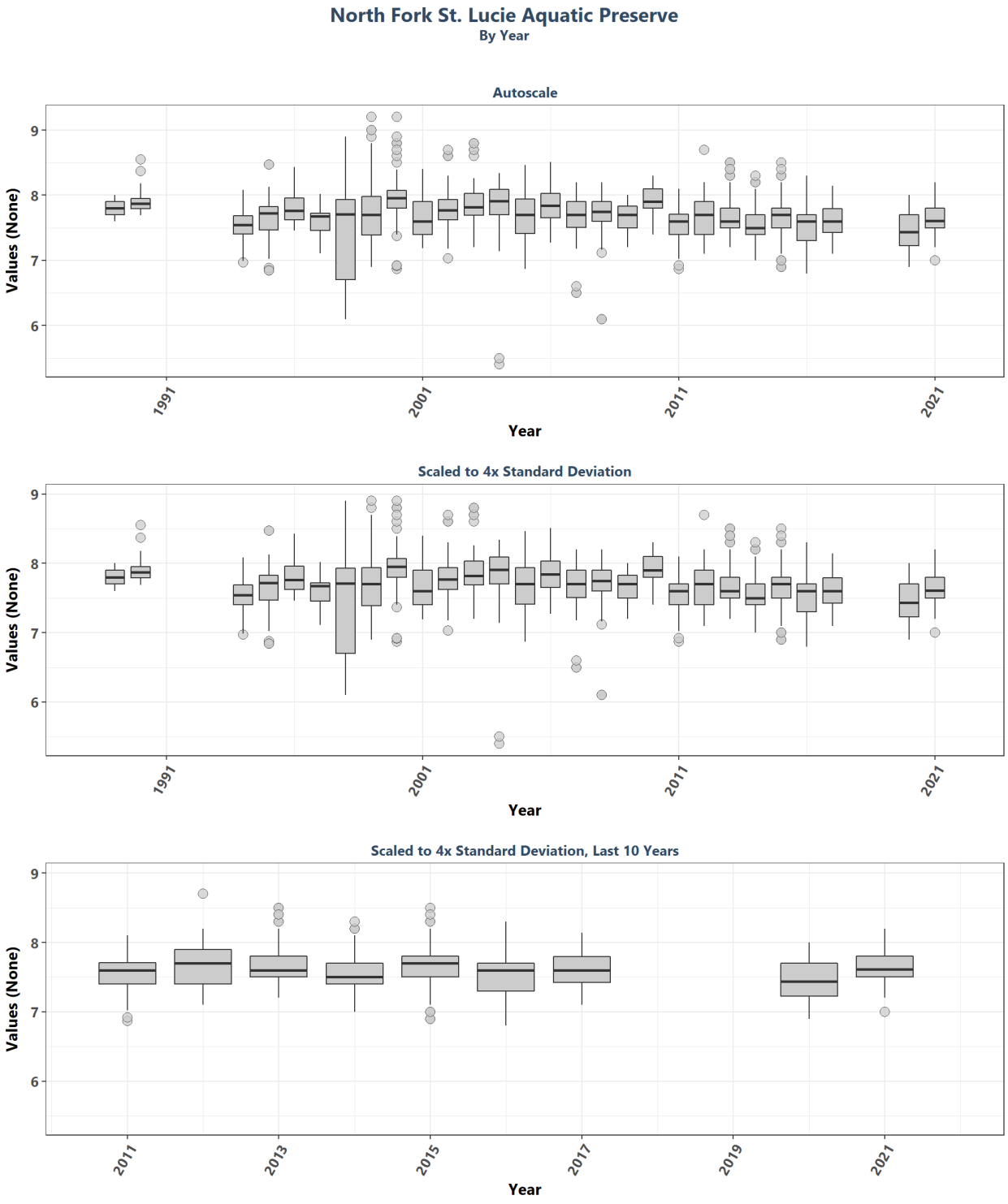

Nassau River-St. Johns River Marshes Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
10.0
7.5
5.0
2.5
0.0
1982
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation
Values (None)
7.5
5.0
2.5
0.0
1982
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
7.5
5.0
2.5
0.0
2012
2014
2016
2018
2020
2022
Year

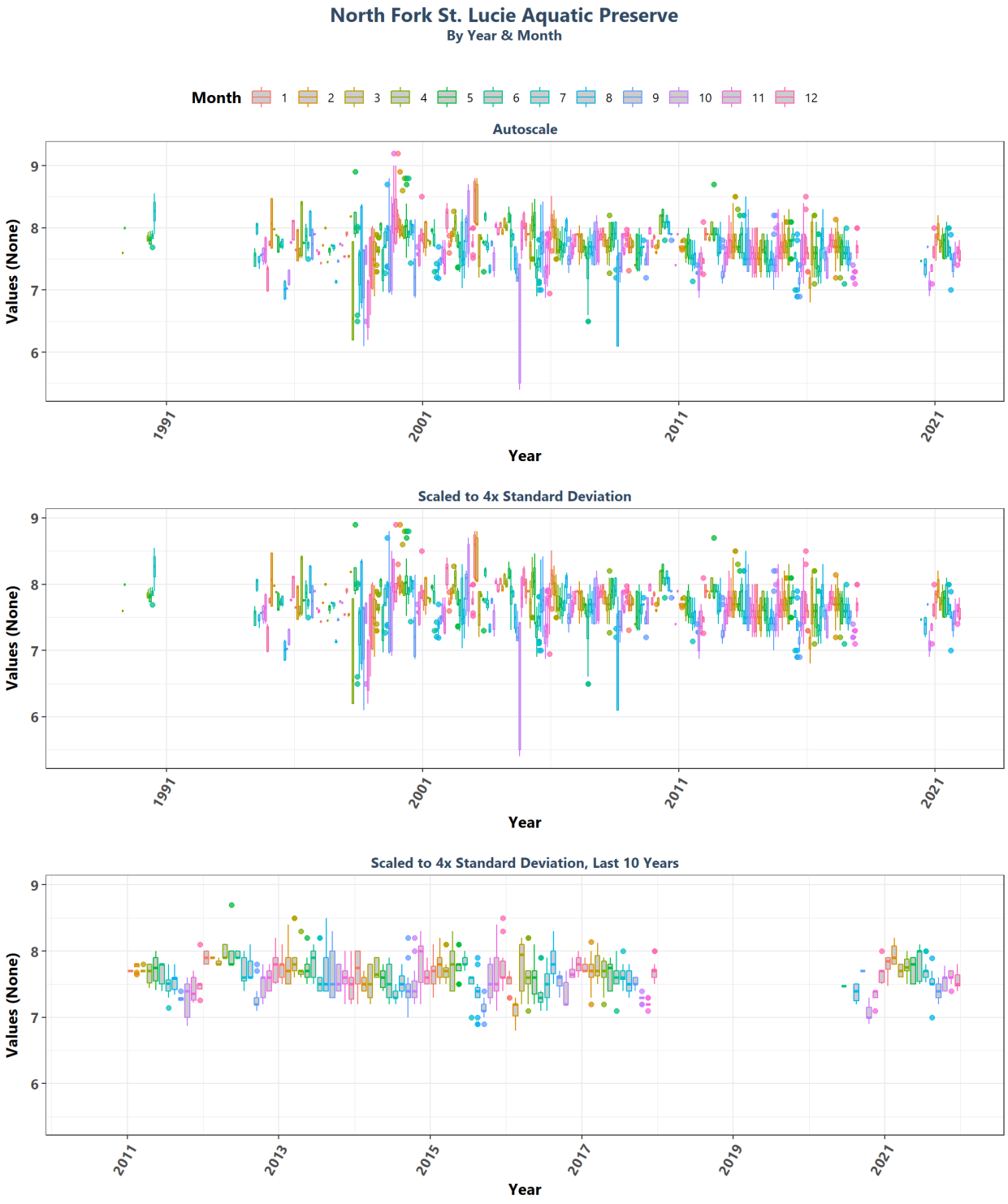

Nassau River-St. Johns River Marshes Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
10.0
7.5
5.0
2.5
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
7.5
5.0
2.5
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
7.5
5.0
2.5
0.0
3
6
9
12
Month

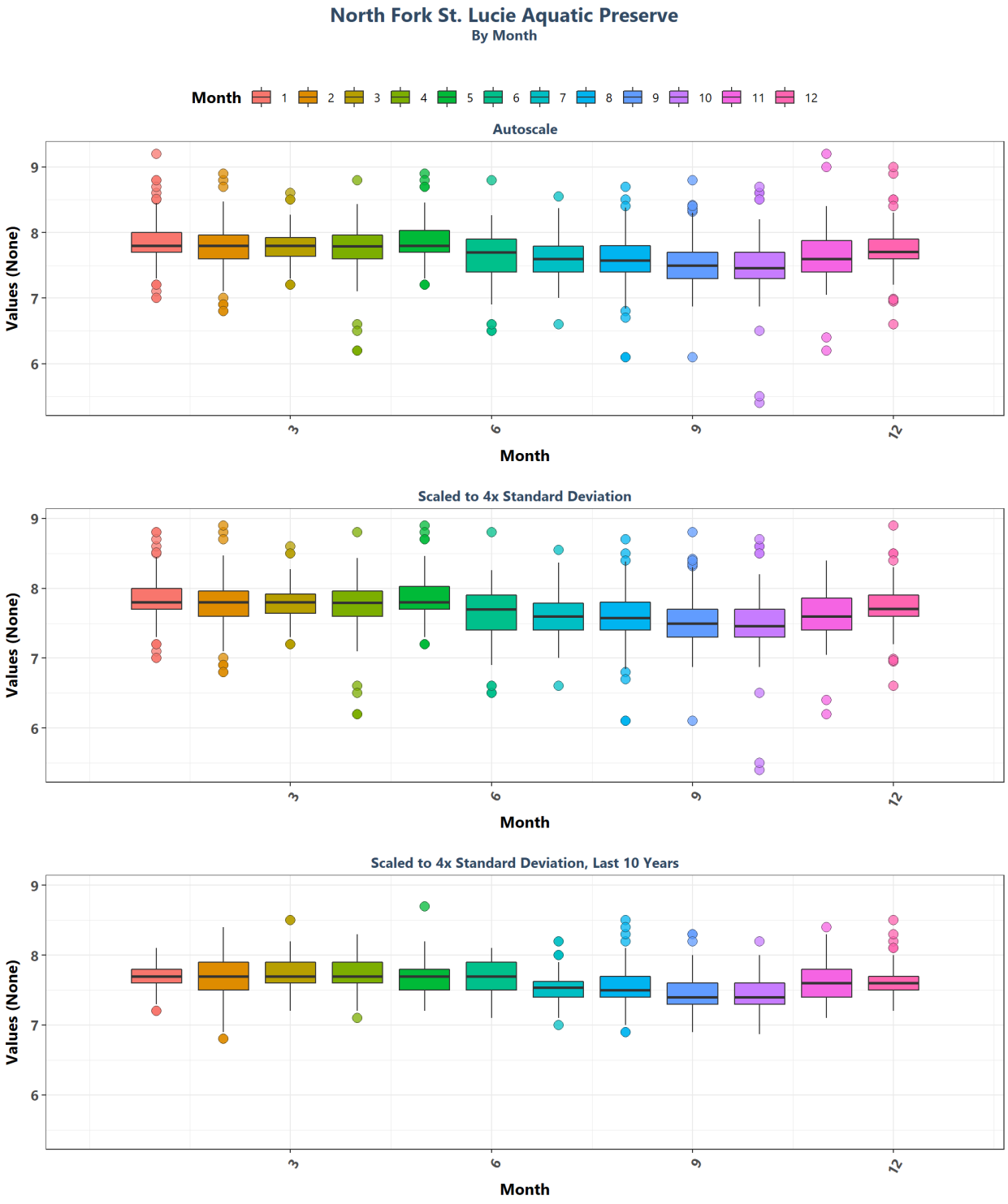

Nature Coast Aquatic Preserve
By Year
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

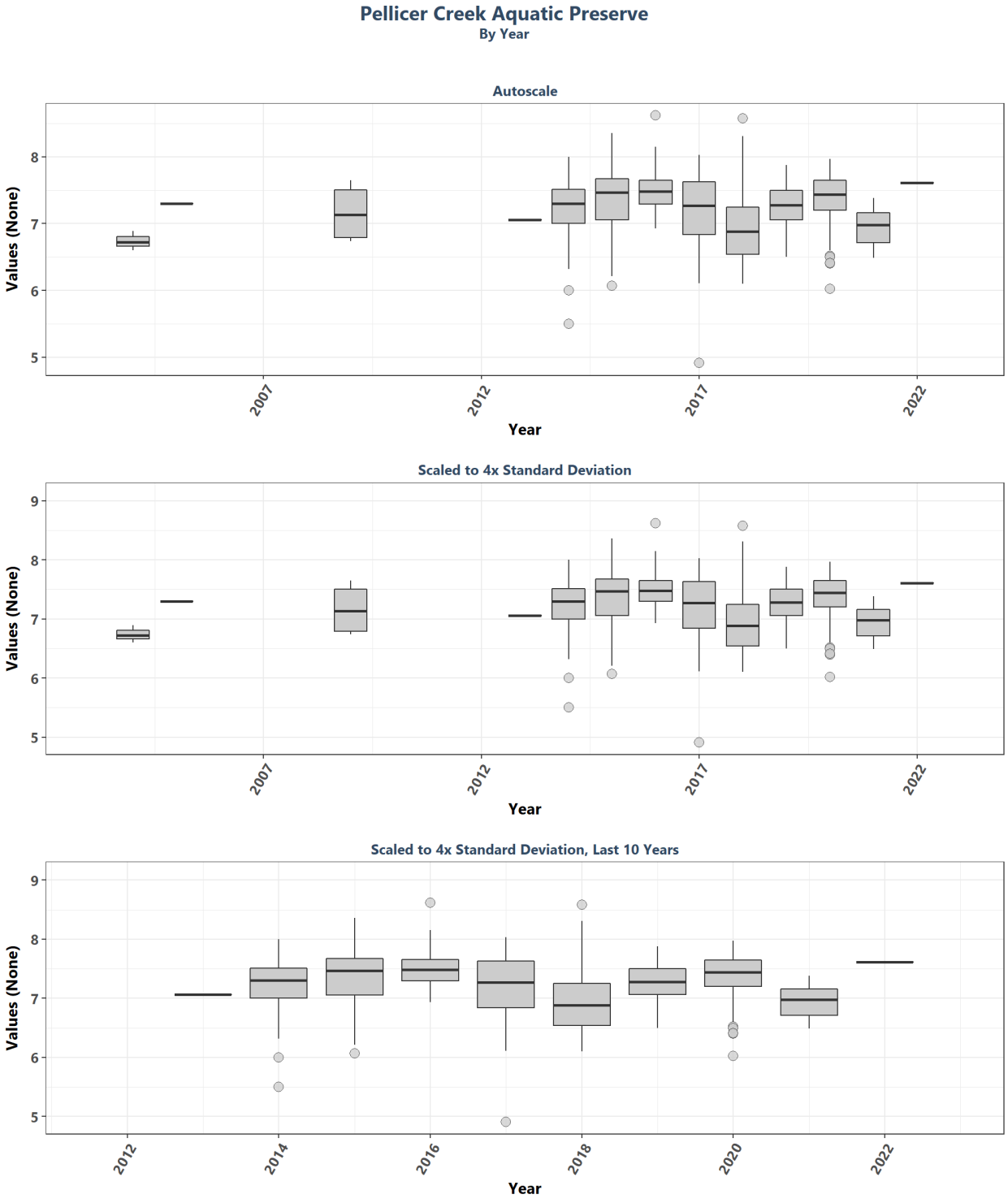

Nature Coast Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

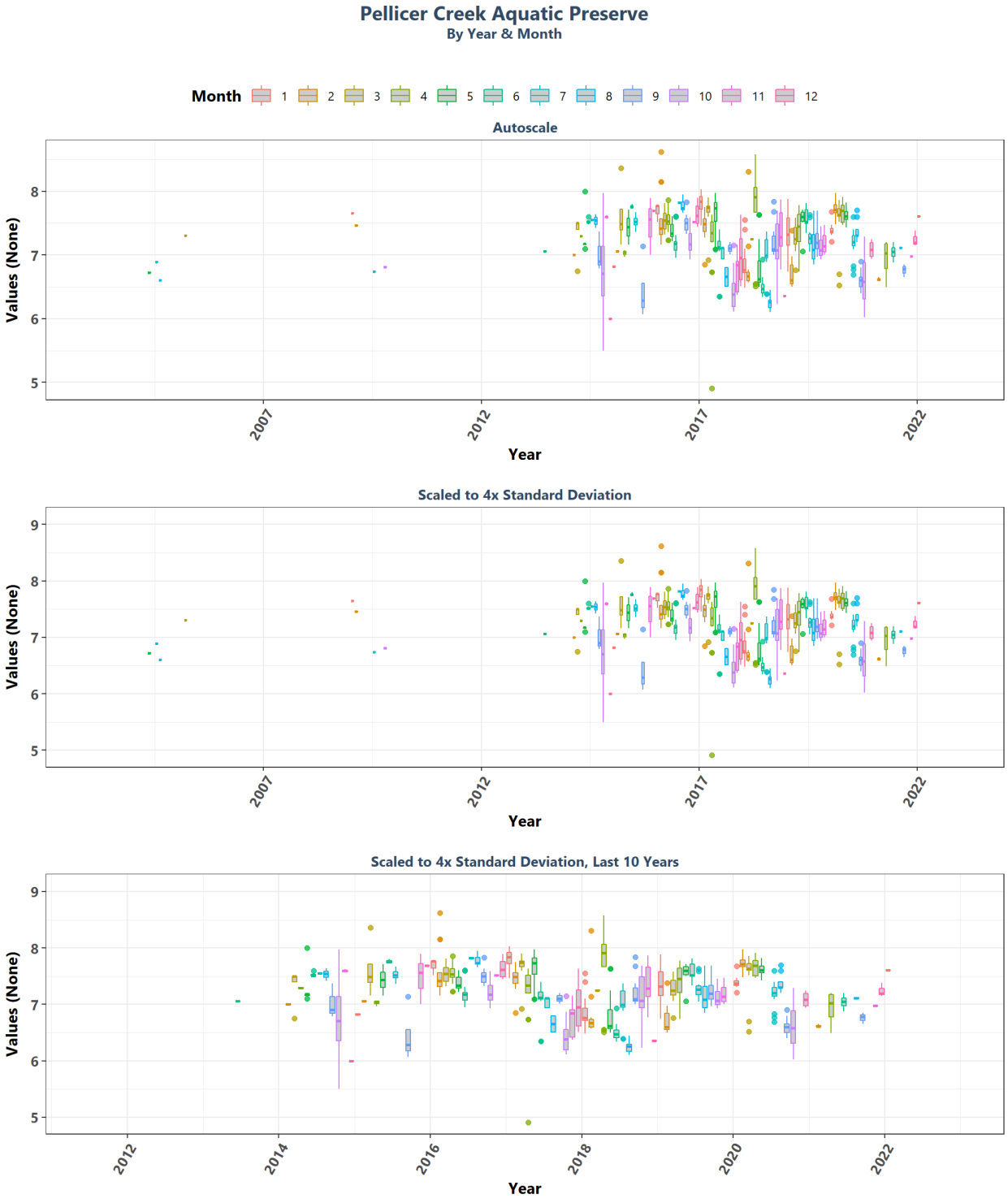

Nature Coast Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

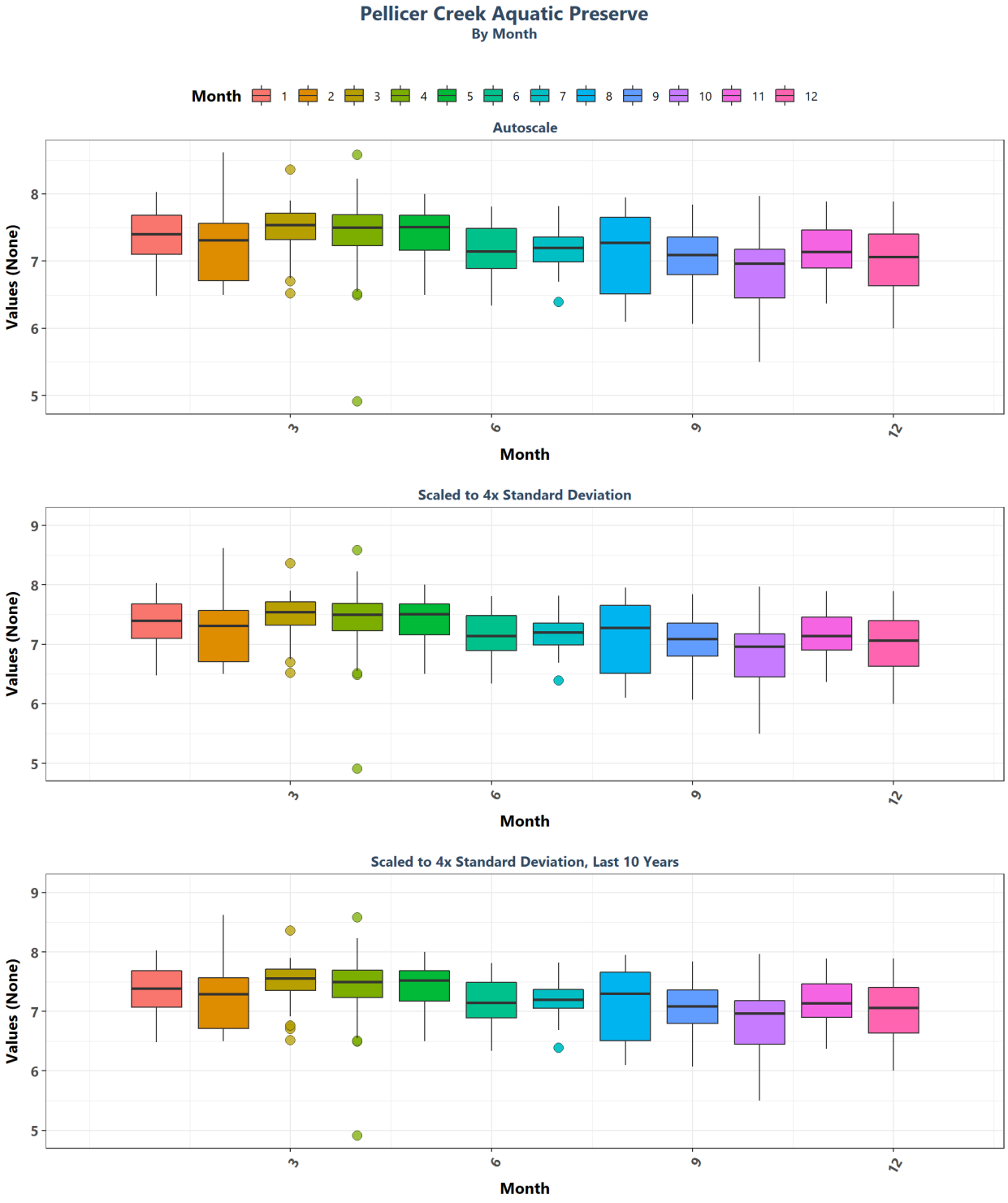

North Fork St. Lucie Aquatic Preserve
By Year
Autoscale
Values (None)
9
8
7
6
1991
2001
2011
2021
Year
Scaled to 4x Standard Deviation
Values (None)
9
8
7
6
1991
2001
2011
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
9
8
7
6
2011
2013
2015
2017
2019
2021
Year

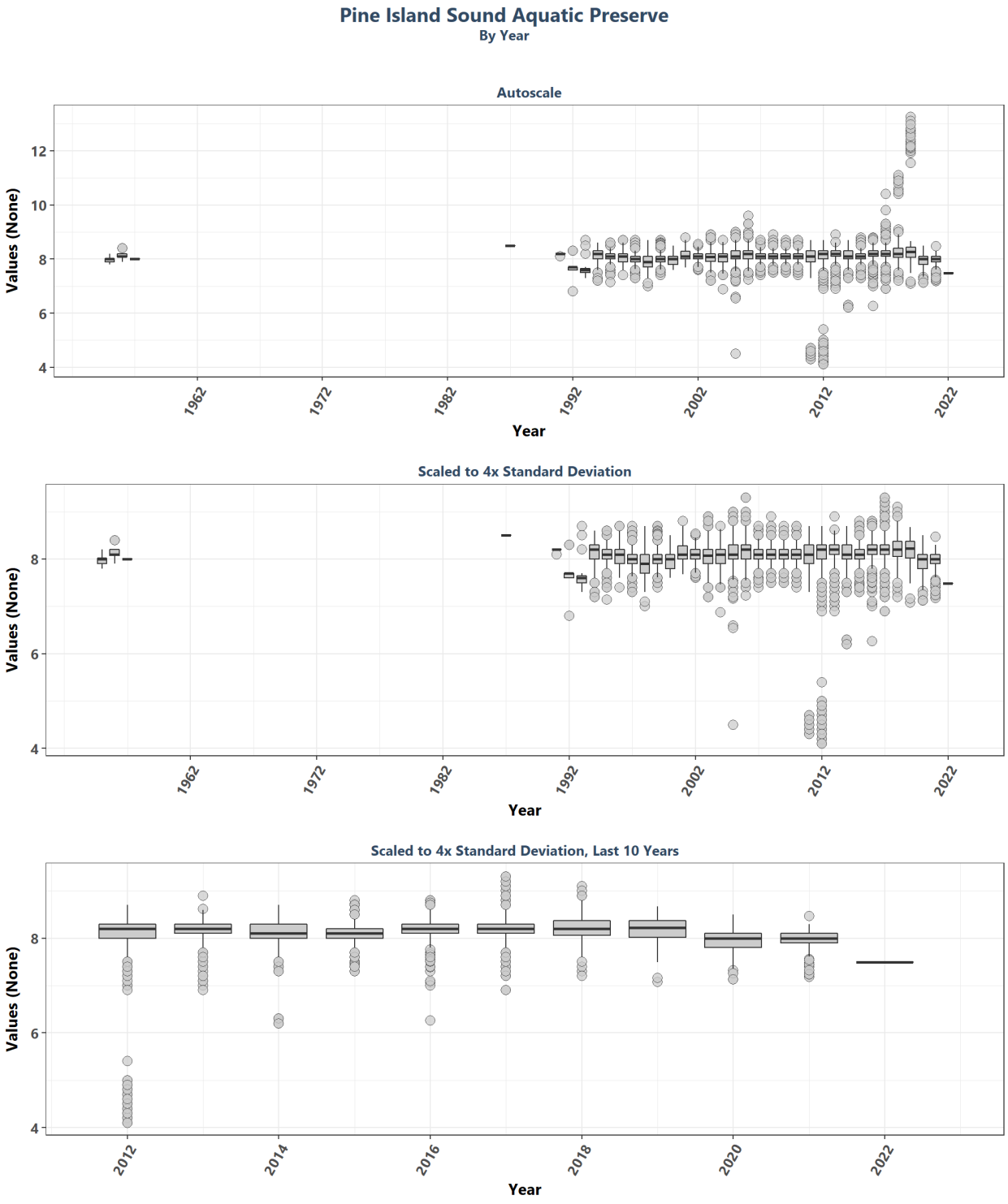

North Fork St. Lucie Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

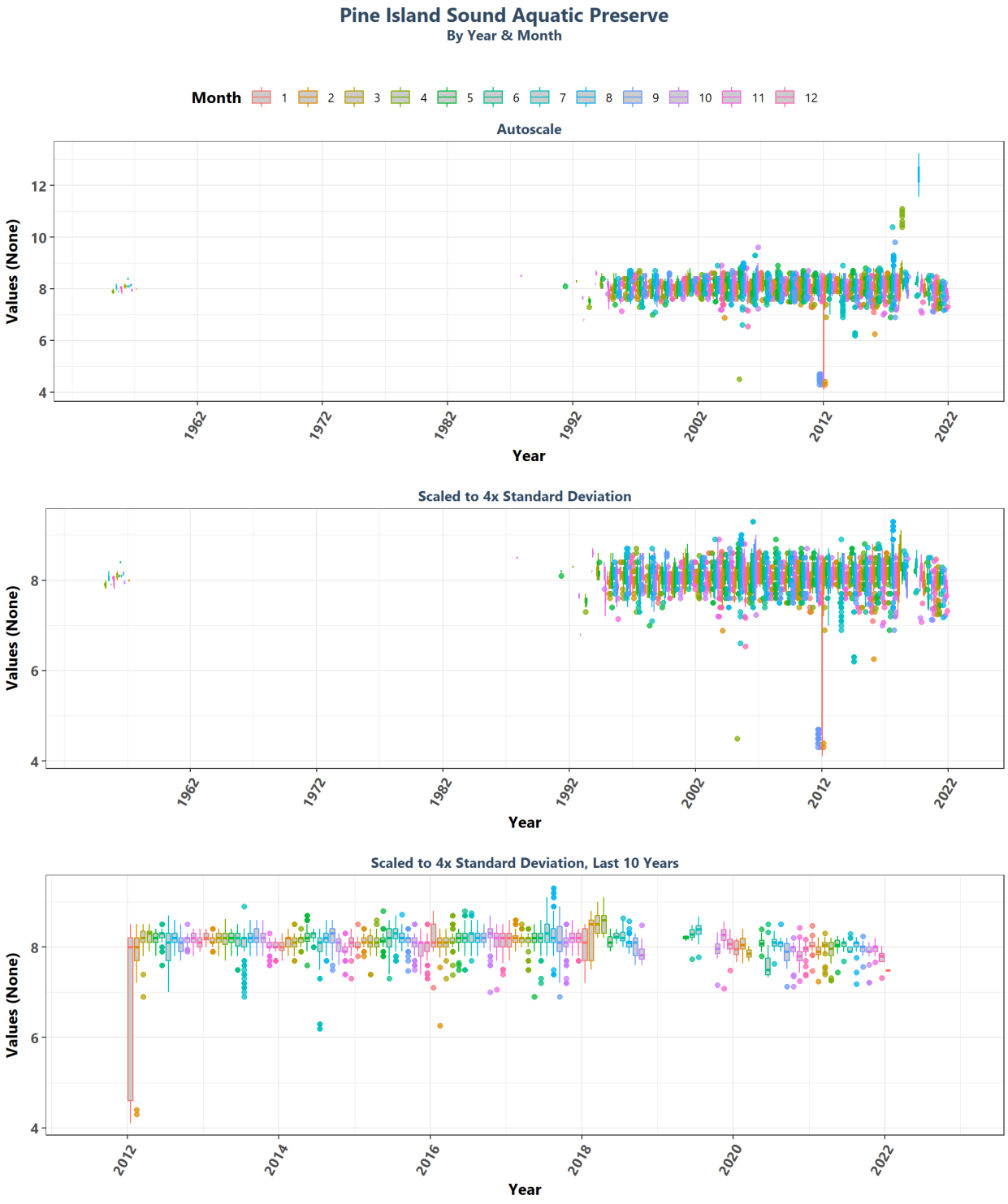

North Fork St. Lucie Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

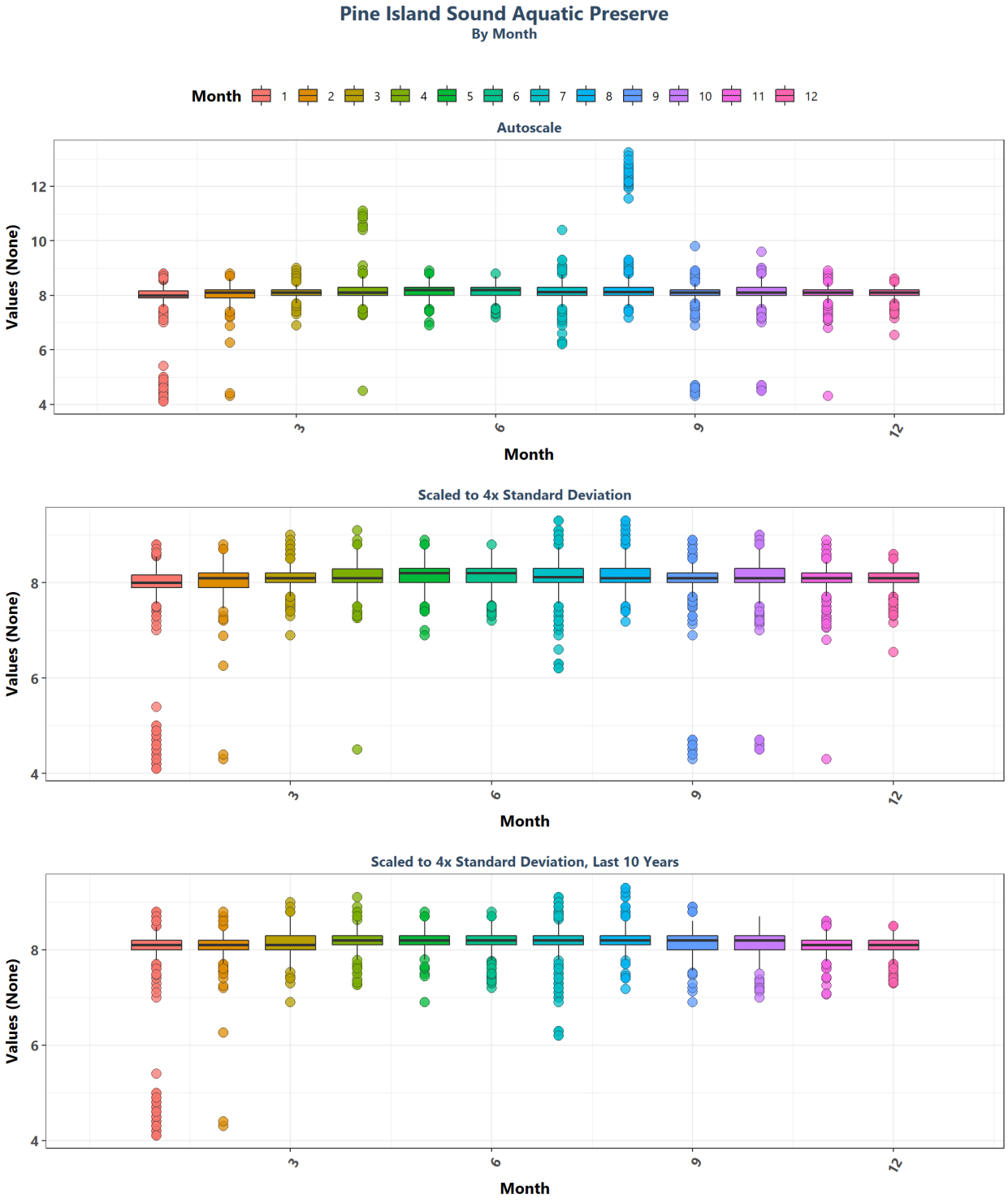

Pellicer Creek Aquatic Preserve
By Year
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

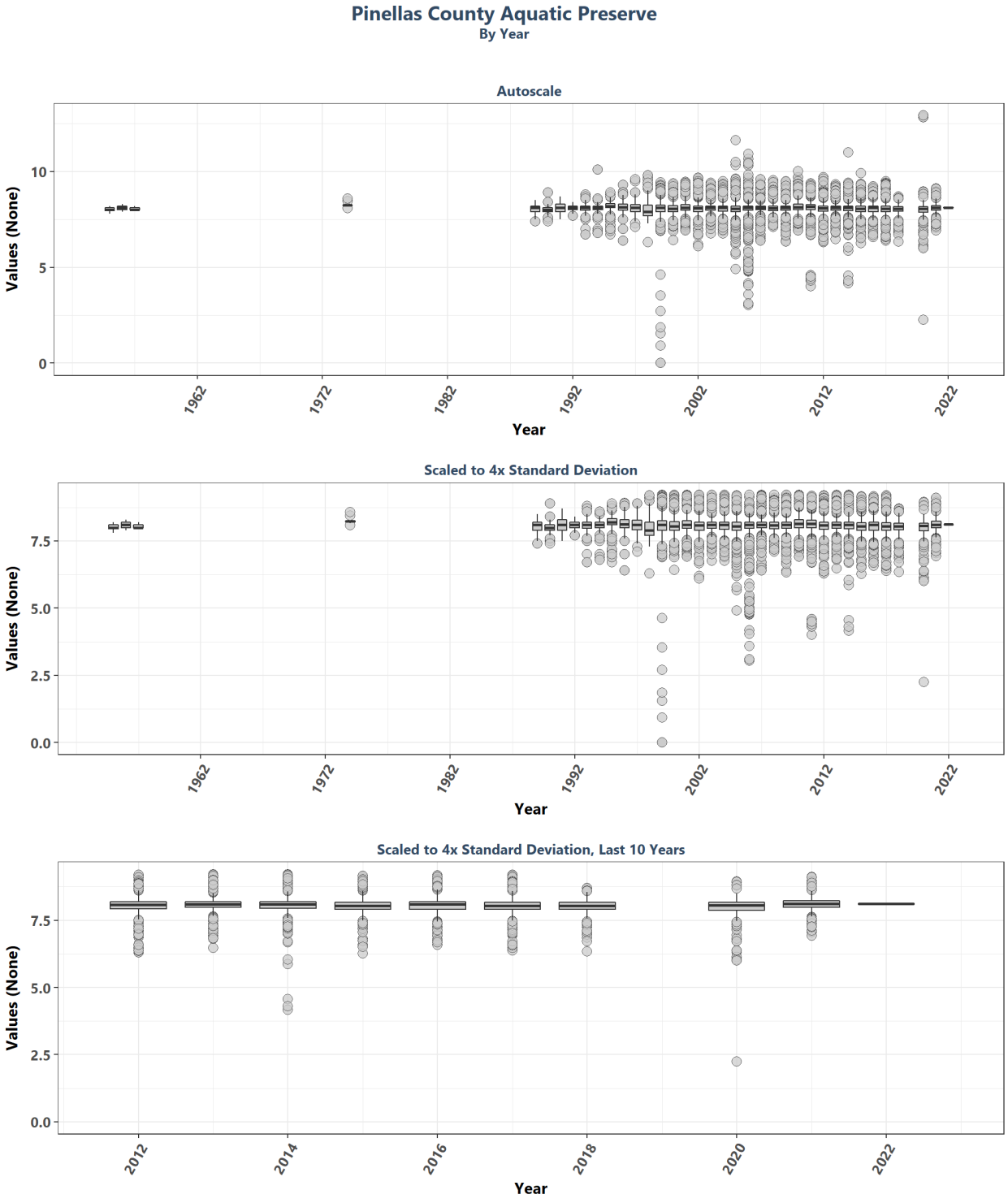

Pellicer Creek Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

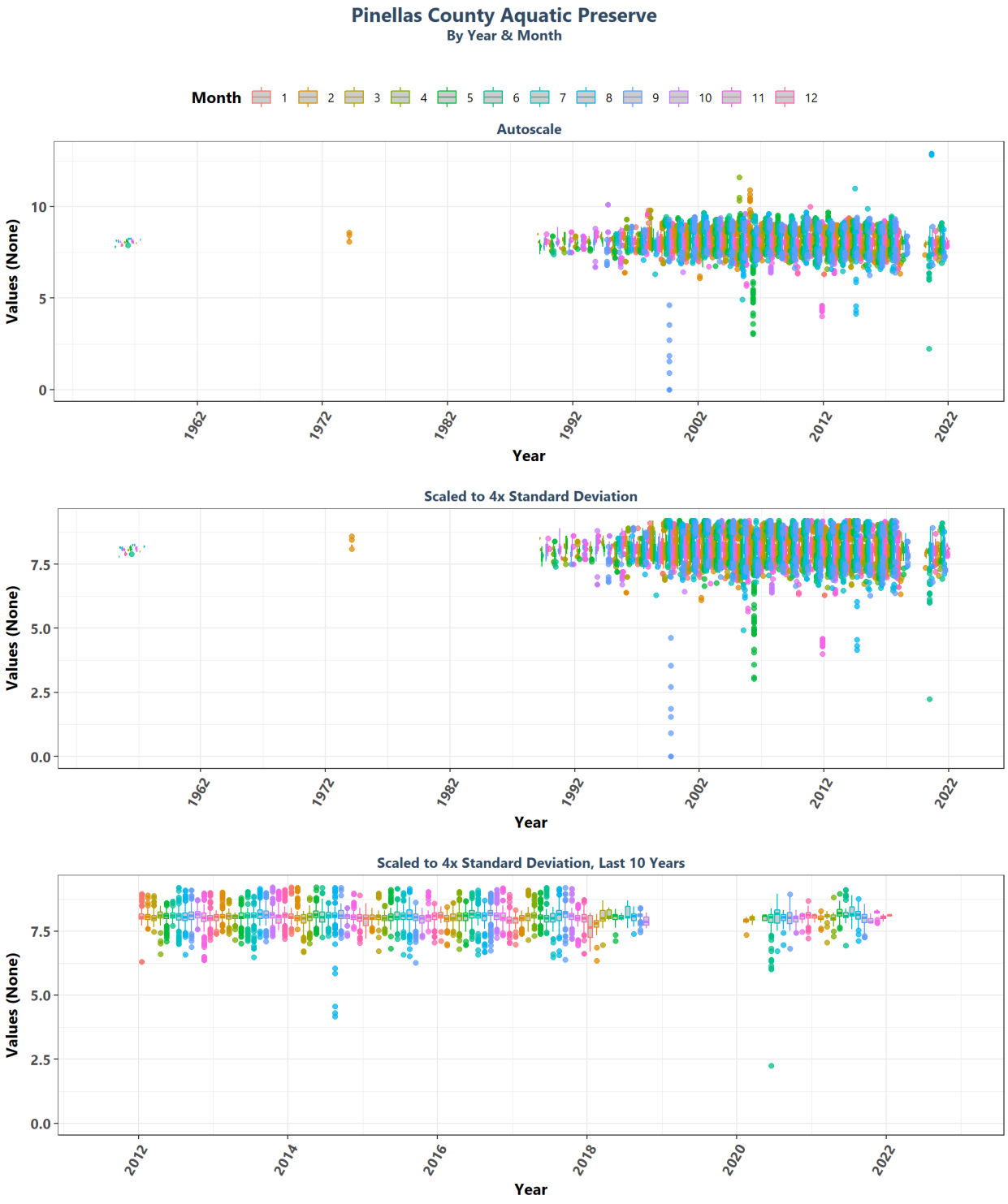

Pellicer Creek Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

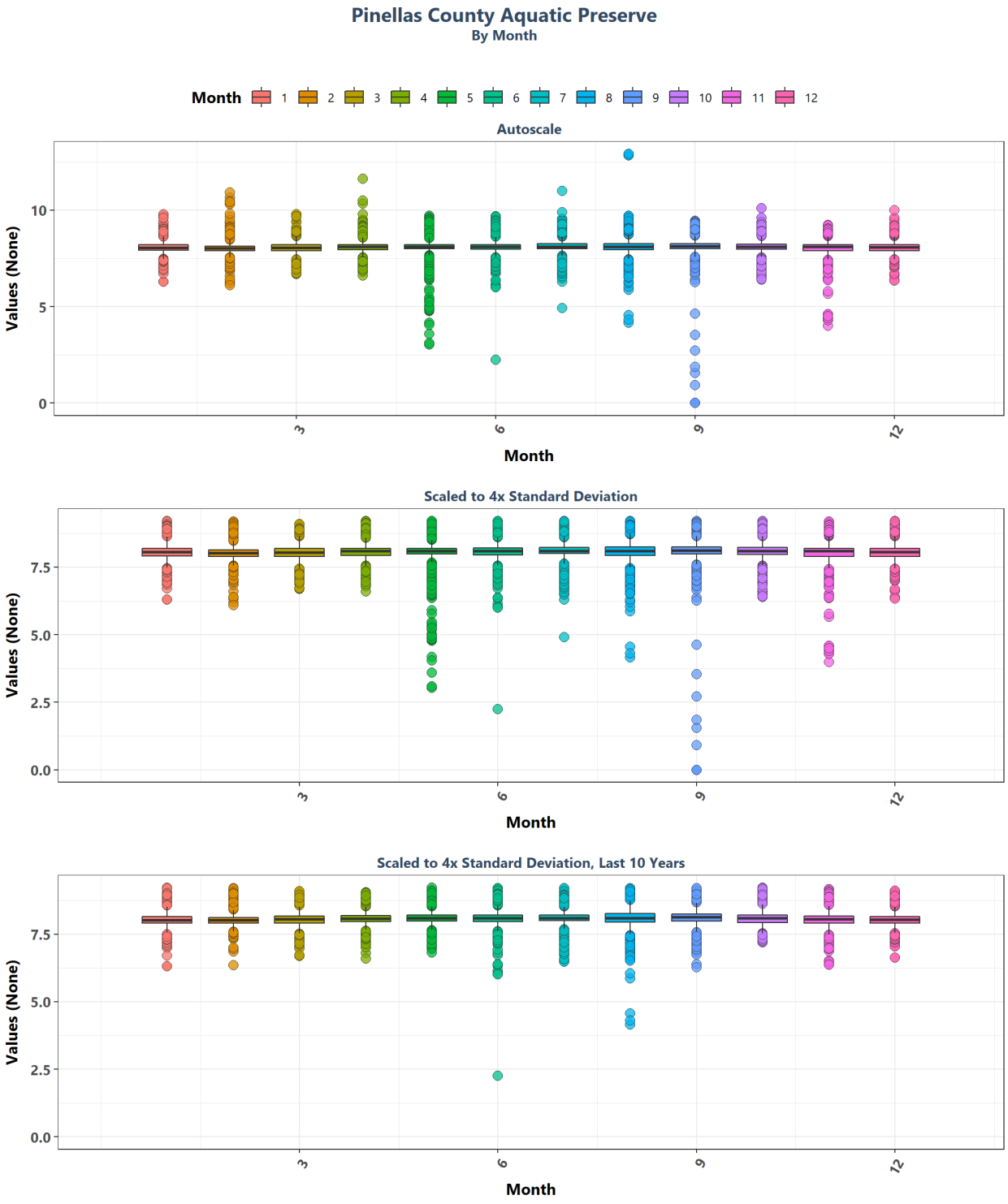

Pine Island Sound Aquatic Preserve
By Year
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year


Pine Island Sound Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year


Pine Island Sound Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

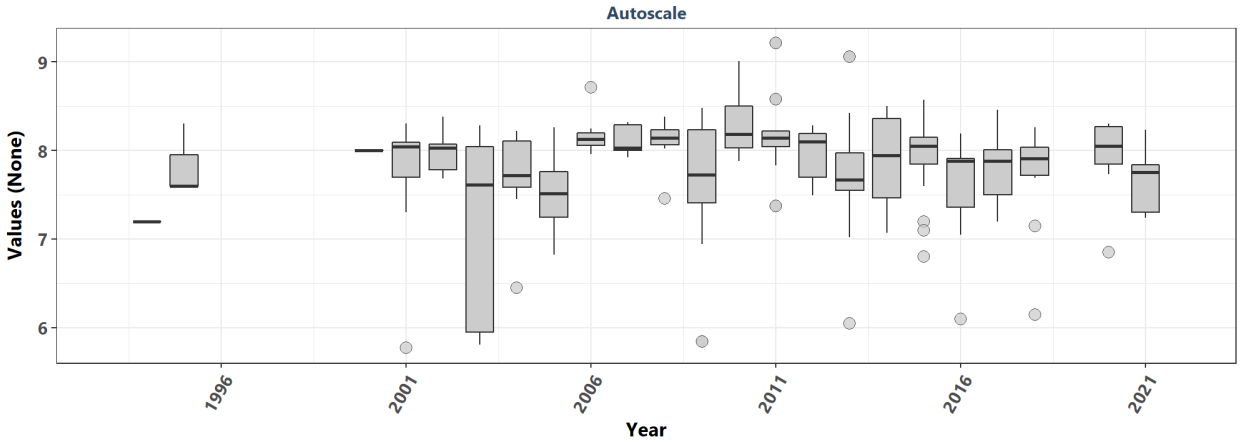

Pinellas County Aquatic Preserve
By Year
Autoscale
Values (None)
10
5
0
1962
1972
1982
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation
Values (None)
7.5
5.0
2.5
0.0
1962
1972
1982
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
7.5
5.0
2.5
0.0
2012
2014
2016
2018
2020
2022
Year

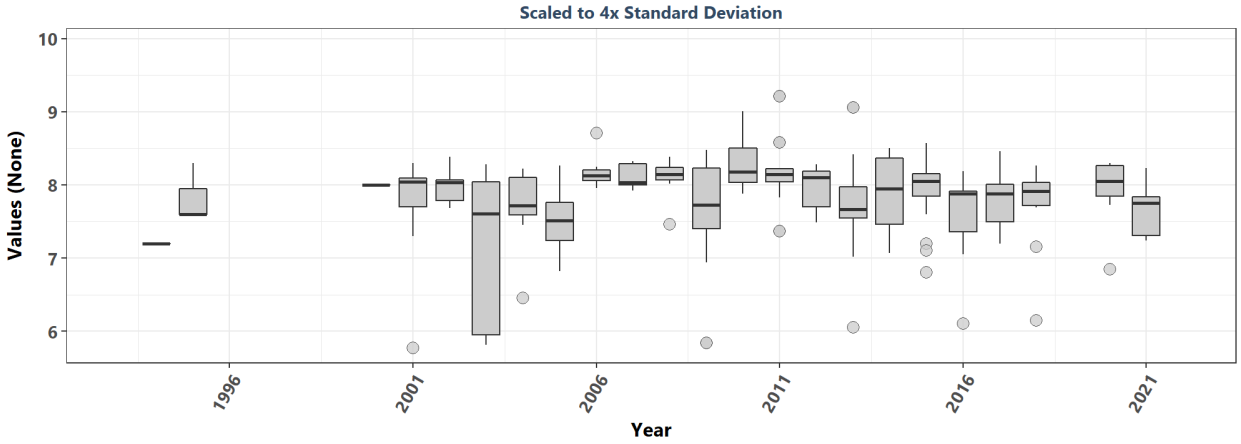

Pinellas County Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

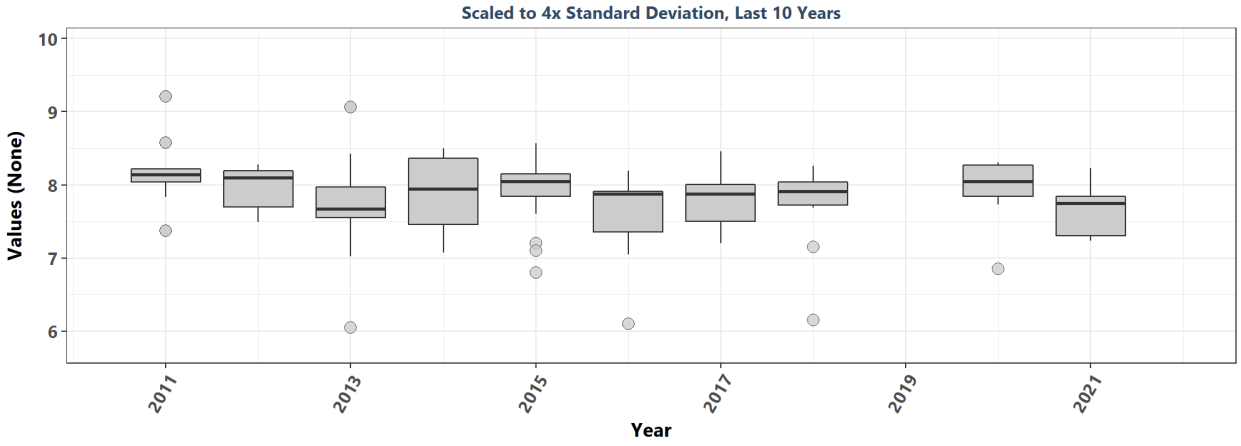

Pinellas County Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
10
5
0
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
7.5
5.0
2.5
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
7.5
5.0
2.5
0.0
3
6
9
12
Month

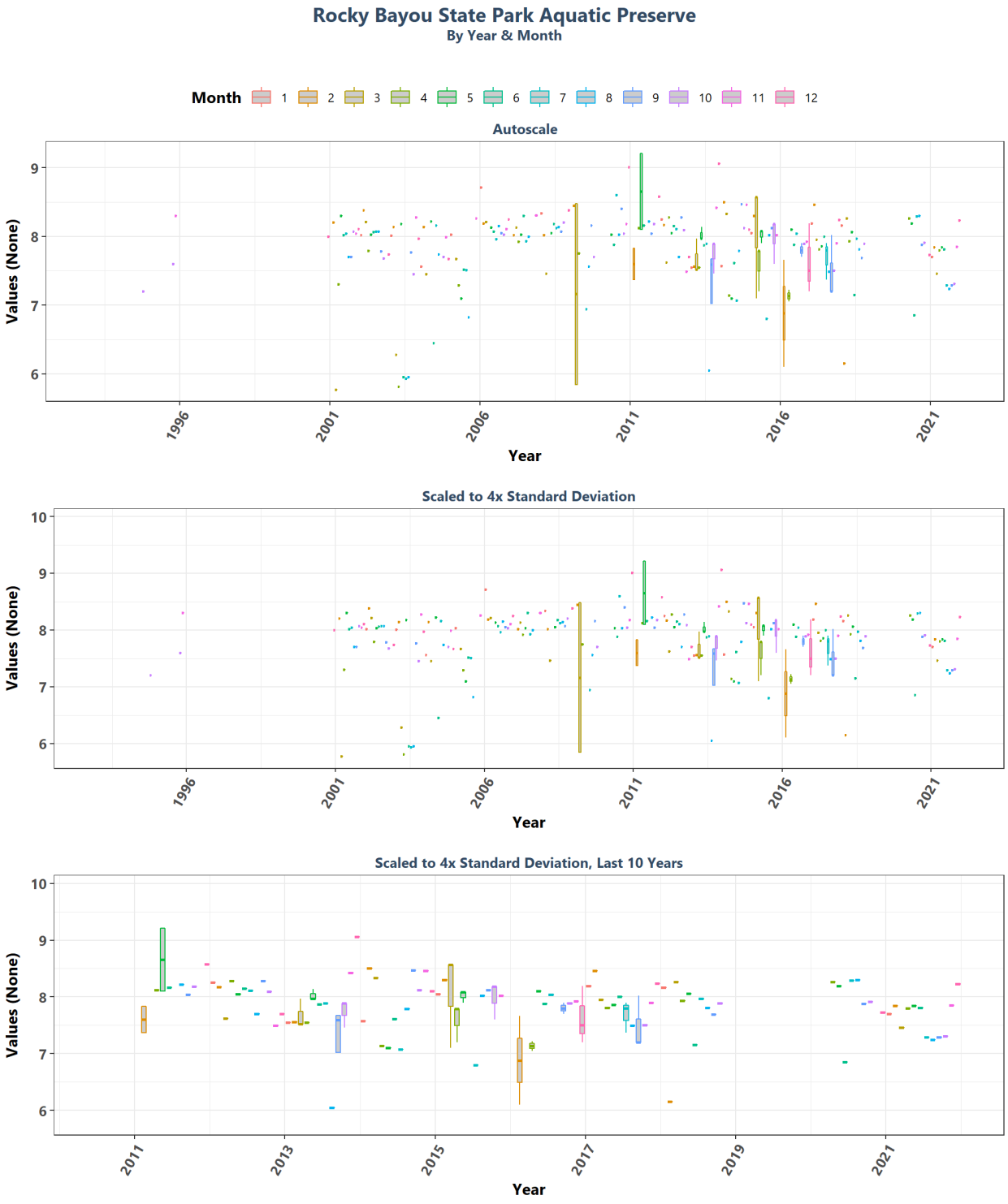
Rocky Bayou State Park Aquatic Preserve

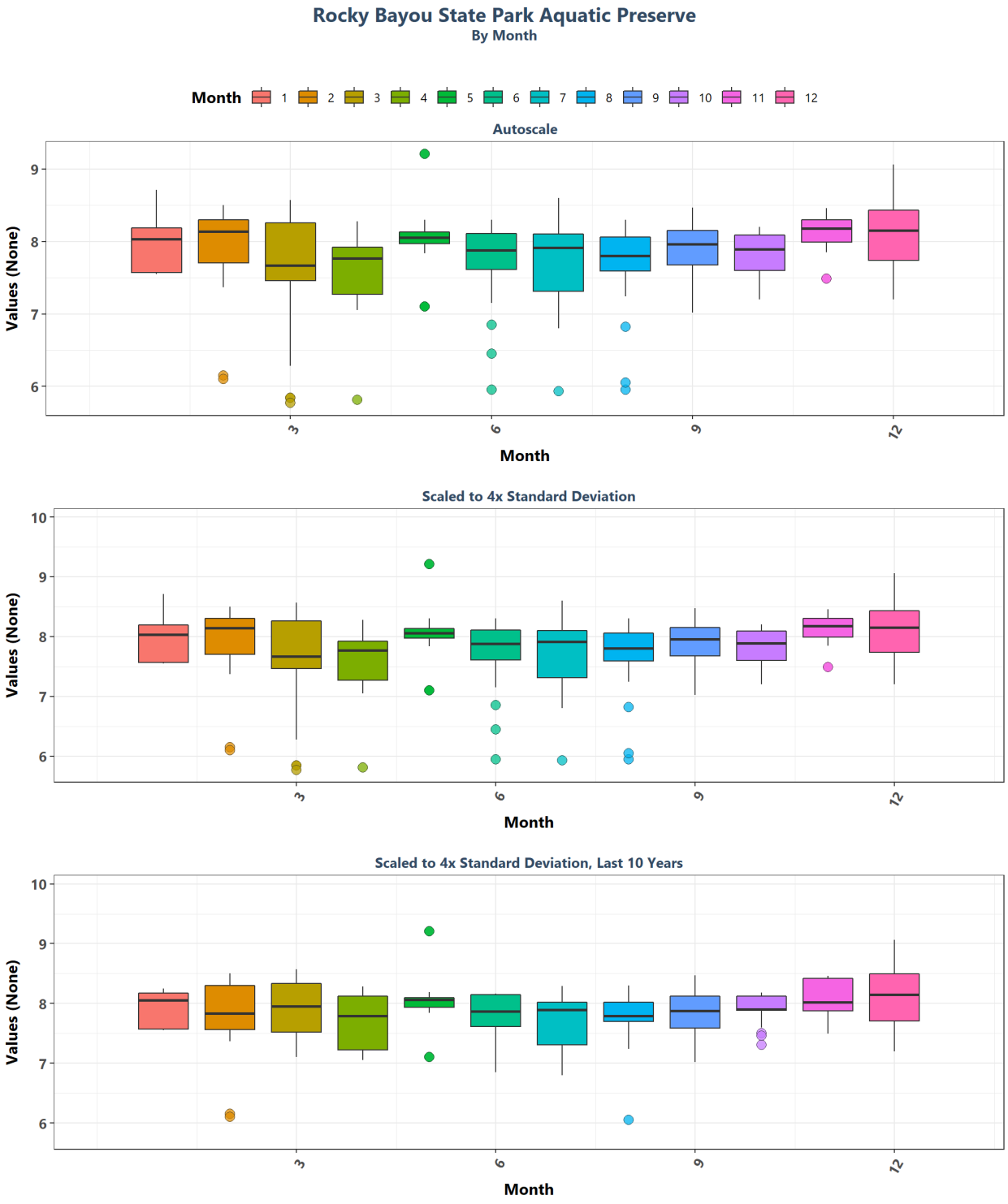
By Year





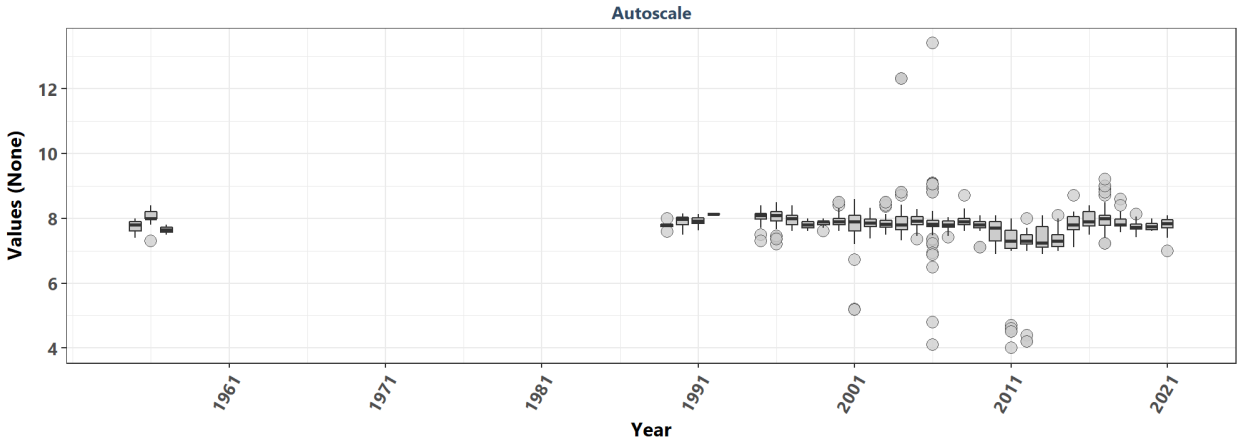


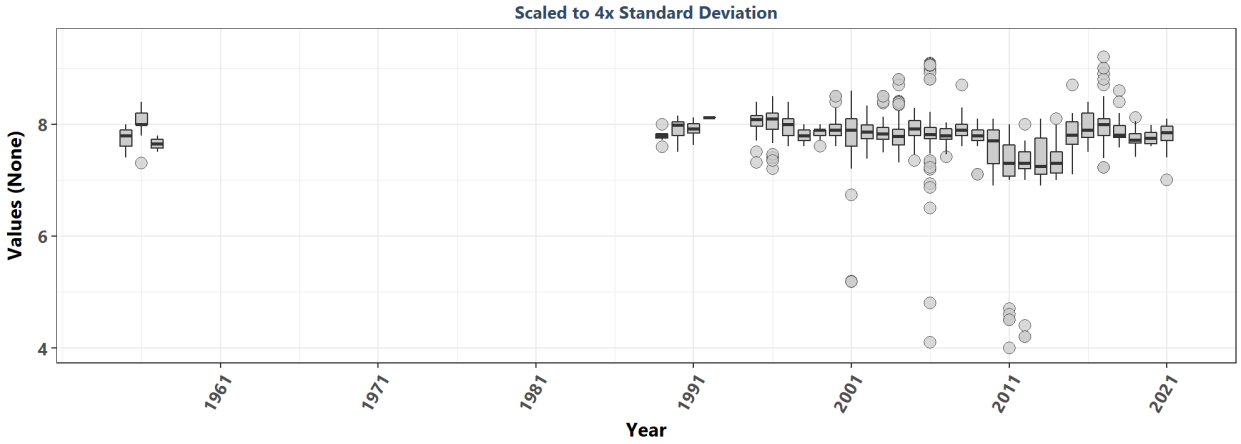

Rocky Bayou State Park Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

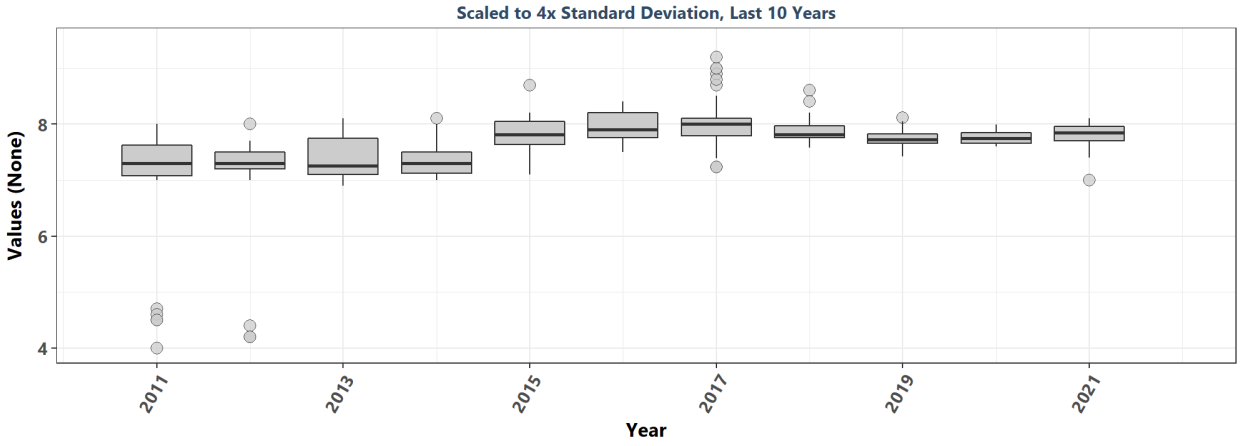

Rocky Bayou State Park Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

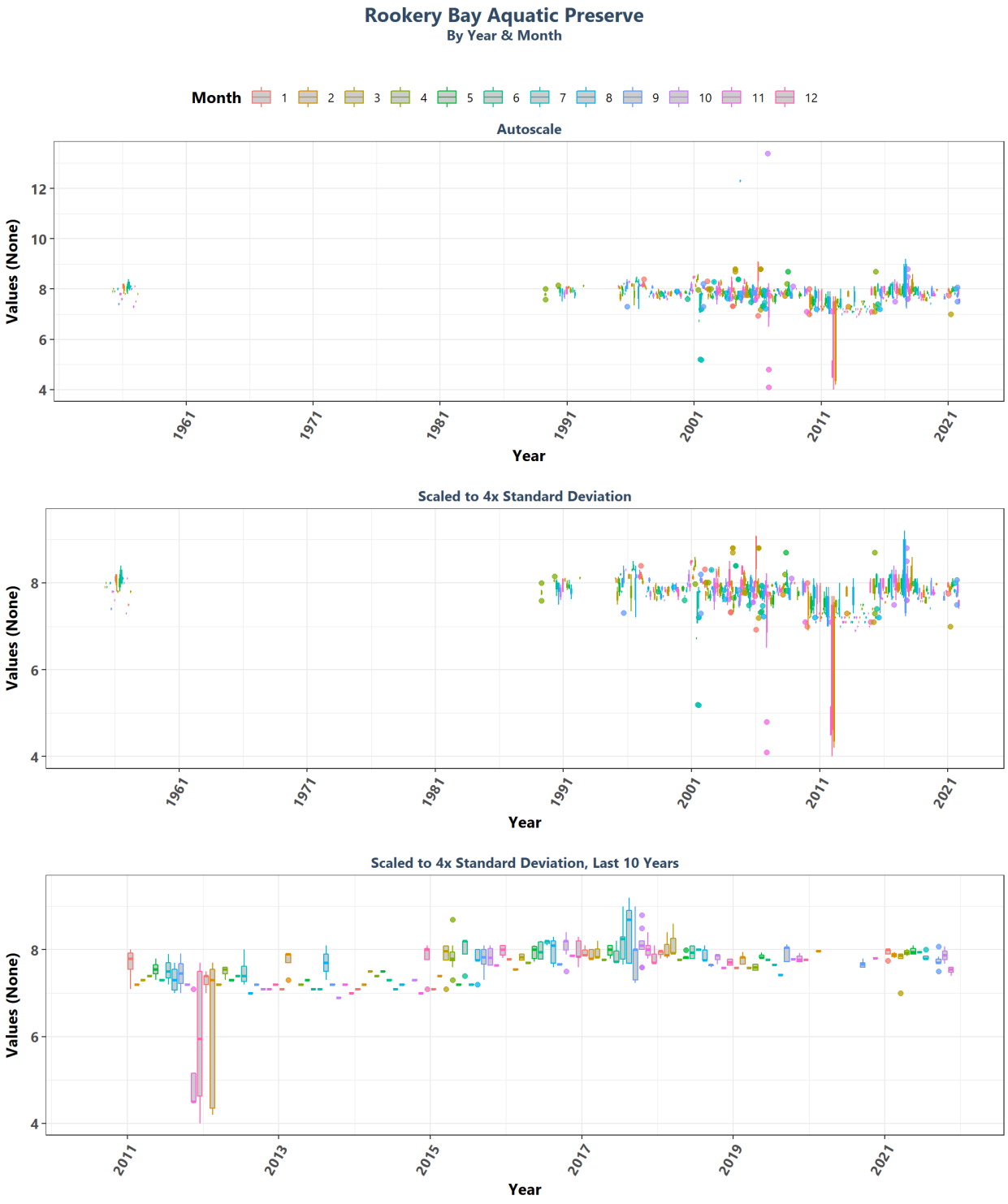
Rookery Bay Aquatic Preserve

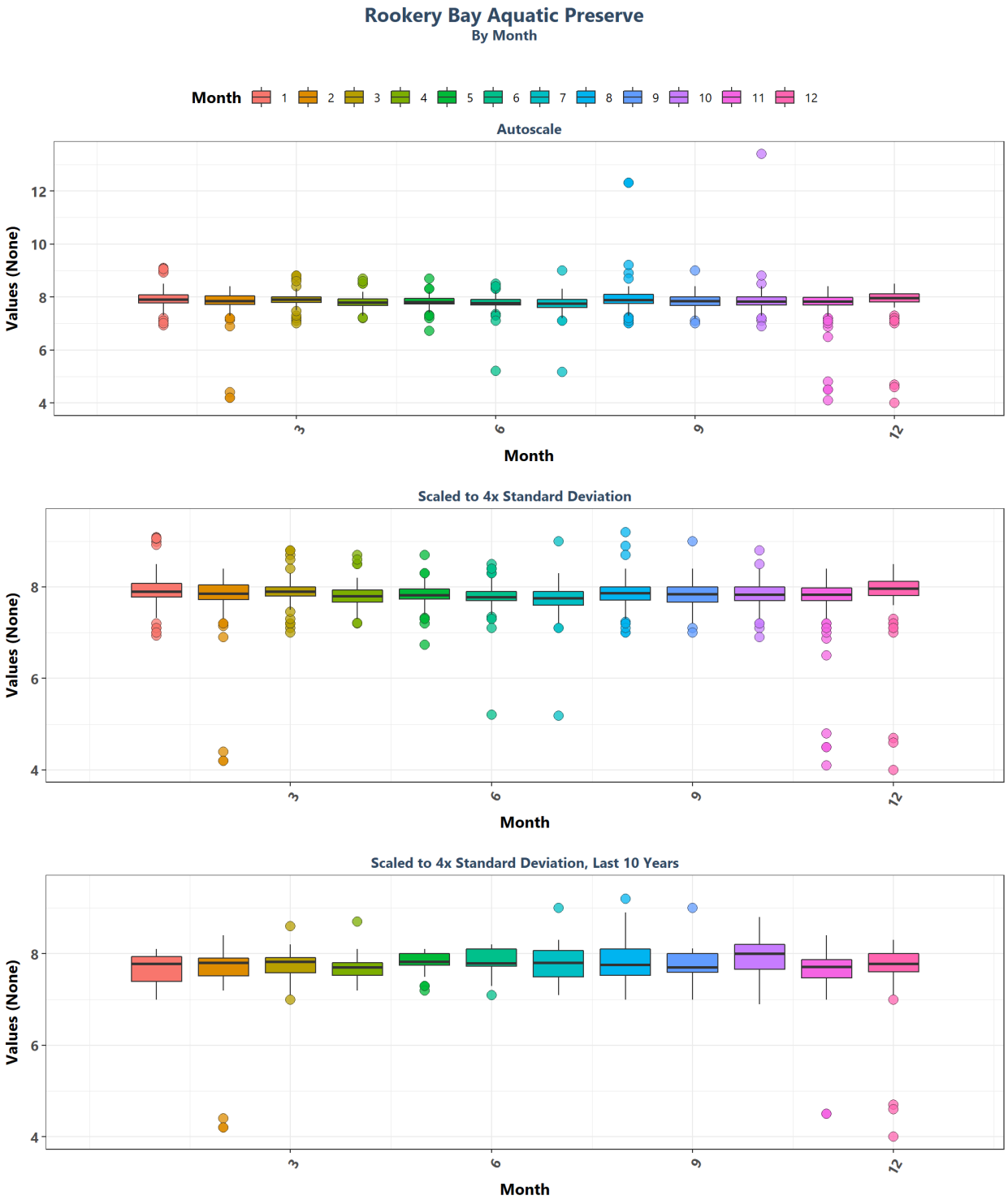
By Year

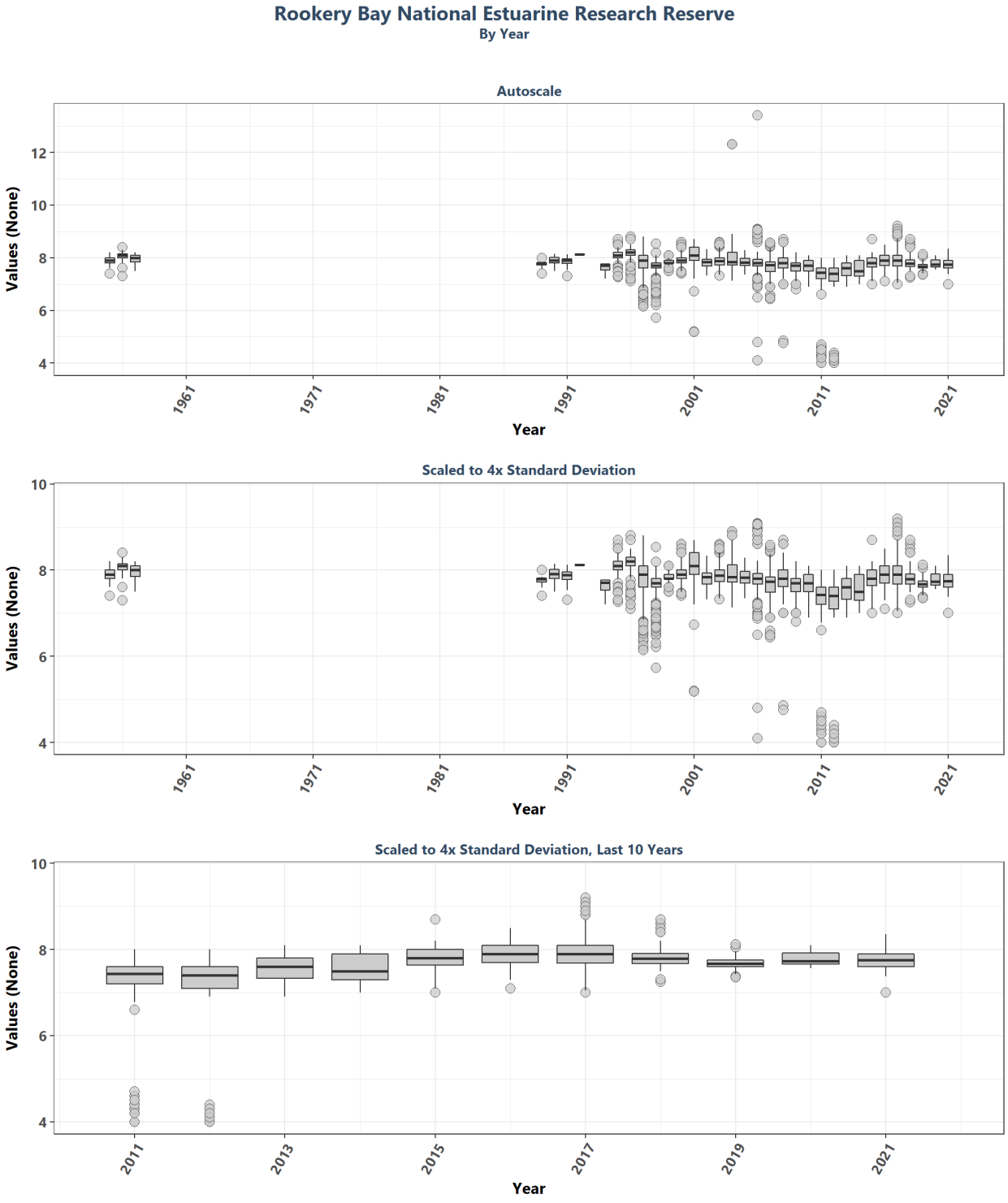


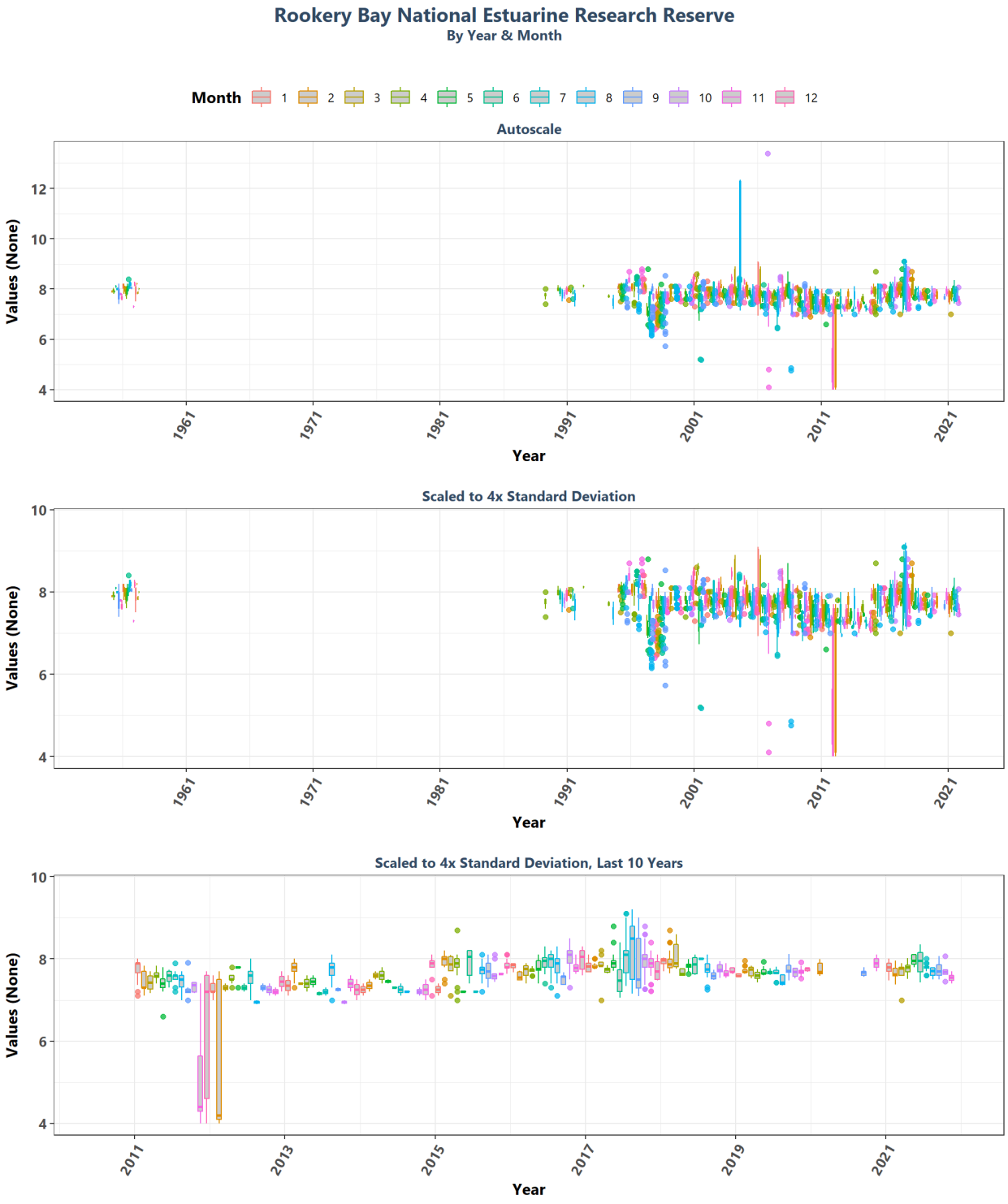


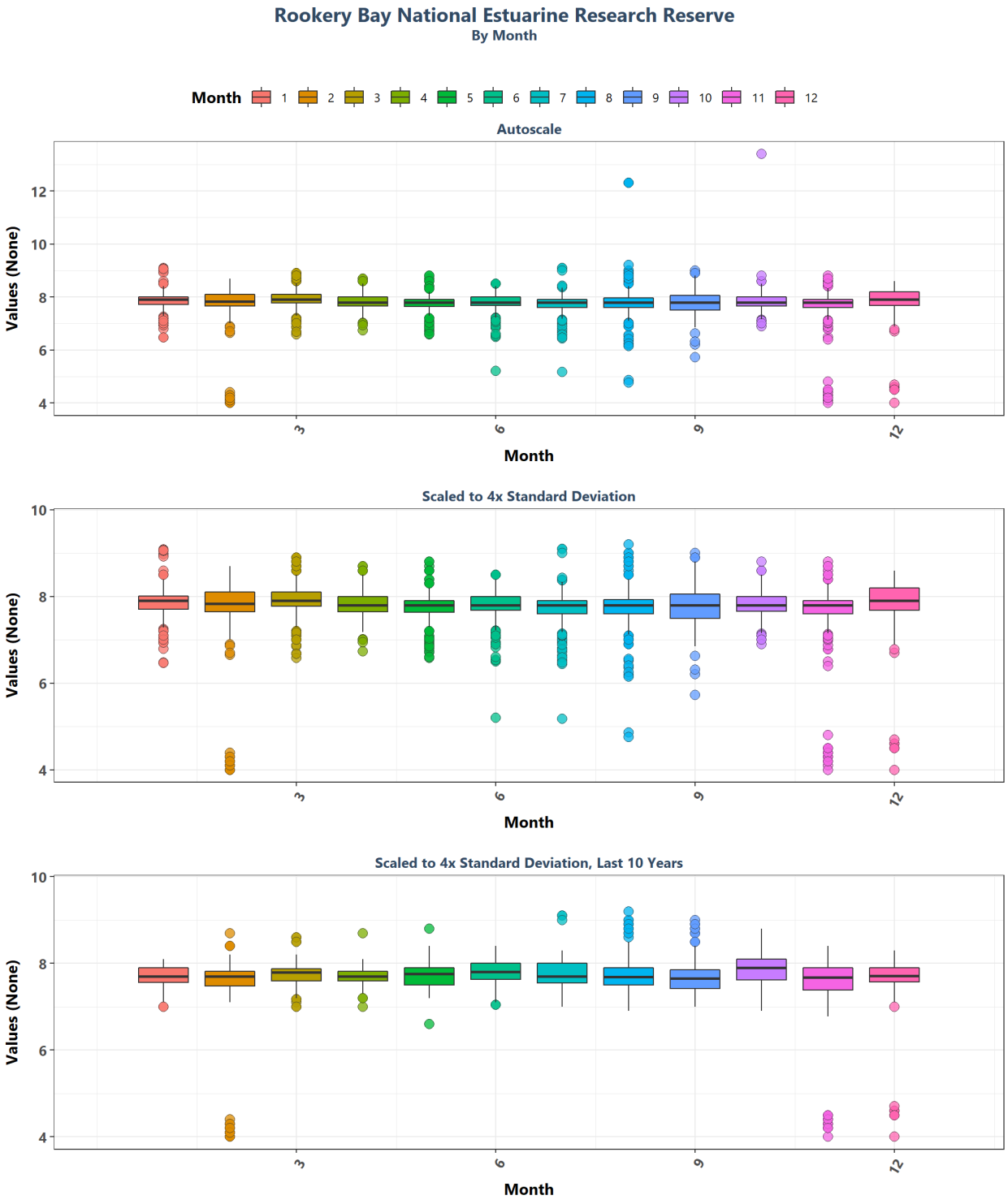



Rookery Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

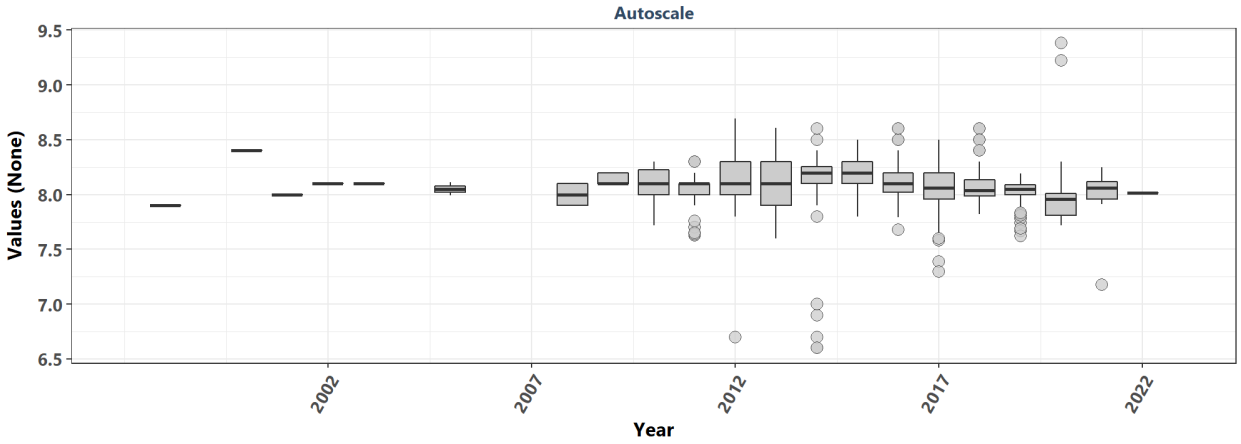

Rookery Bay Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

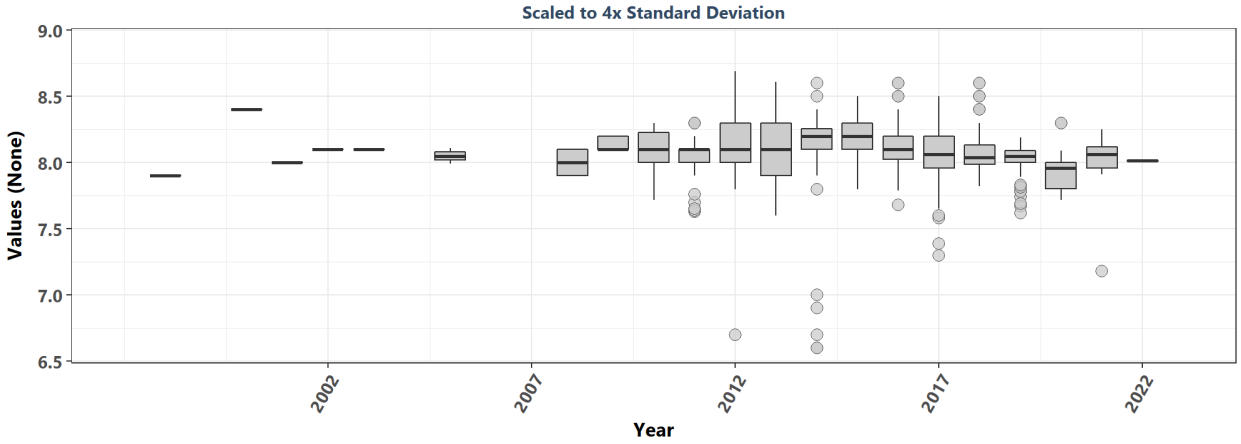

Rookery Bay National Estuarine Research Reserve
By Year
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

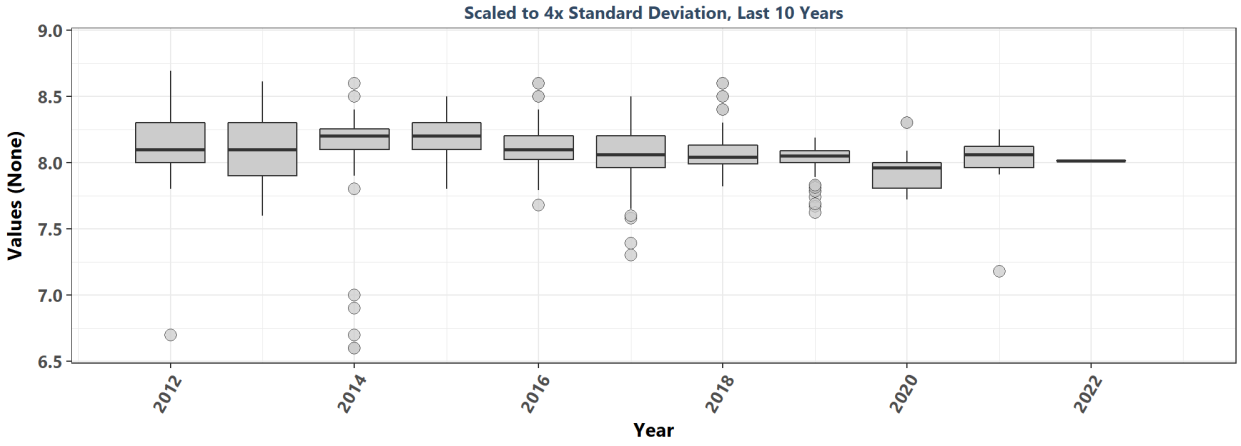

Rookery Bay National Estuarine Research Reserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

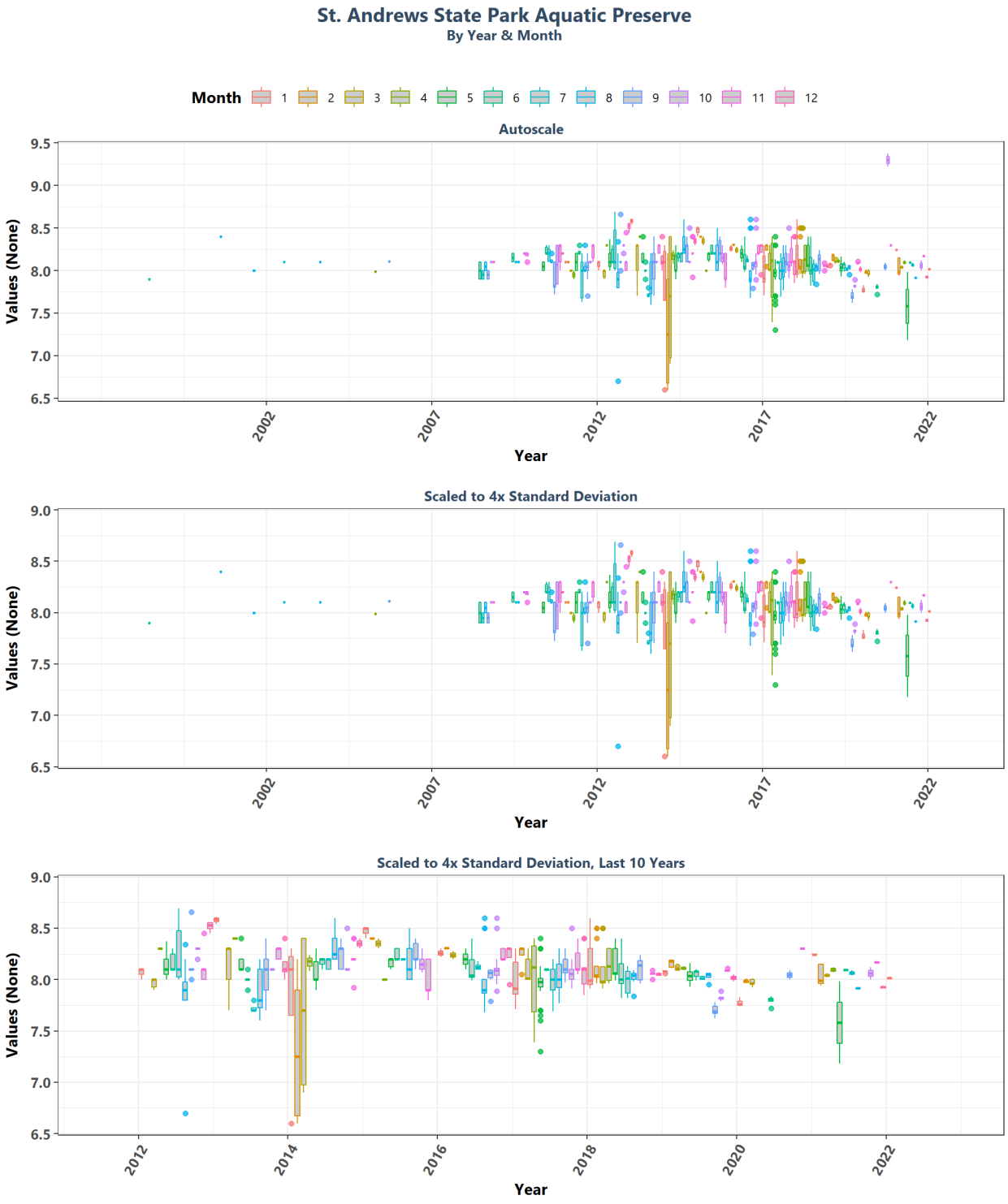

Rookery Bay National Estuarine Research Reserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

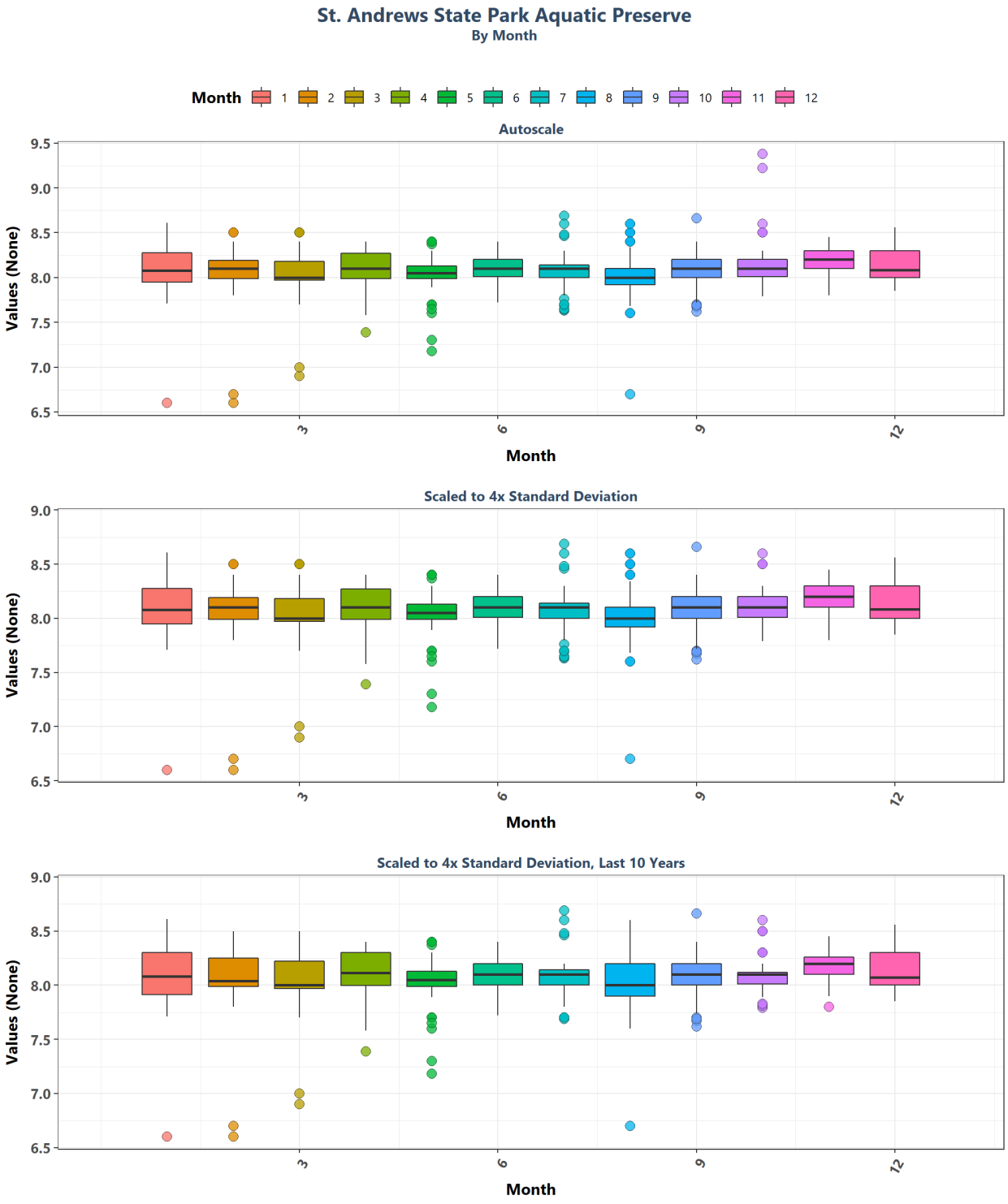
St. Andrews State Park Aquatic Preserve
By Year

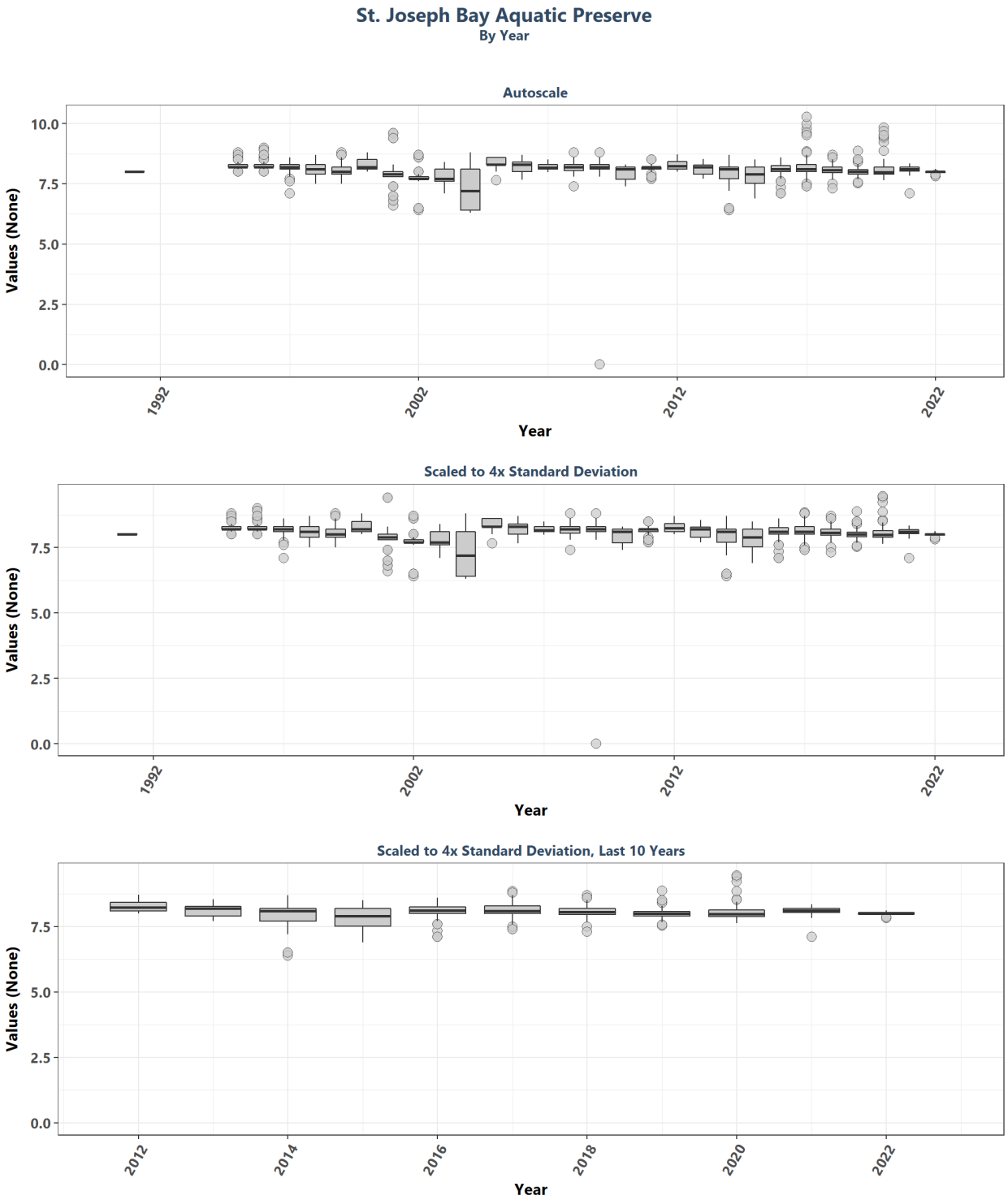


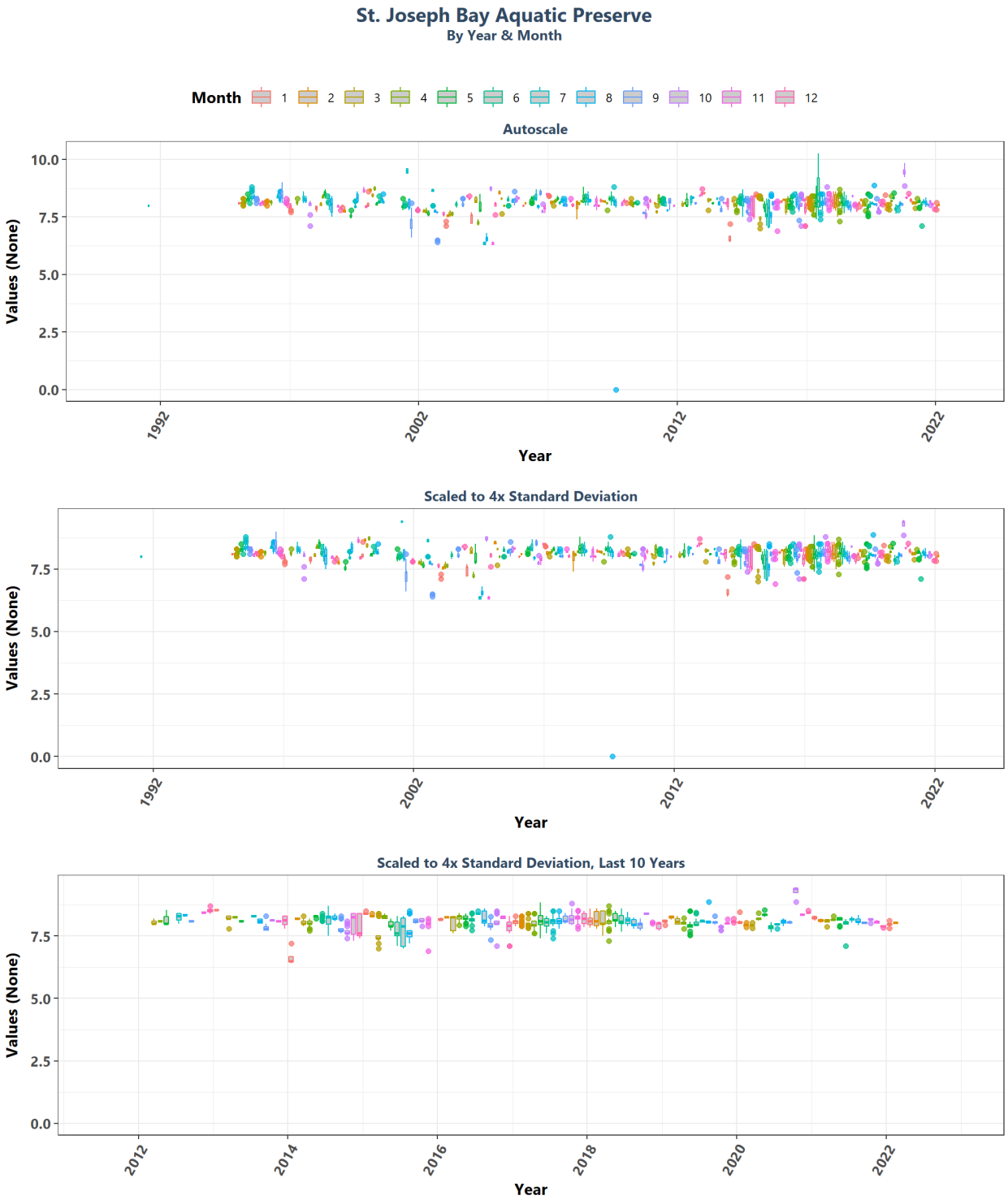


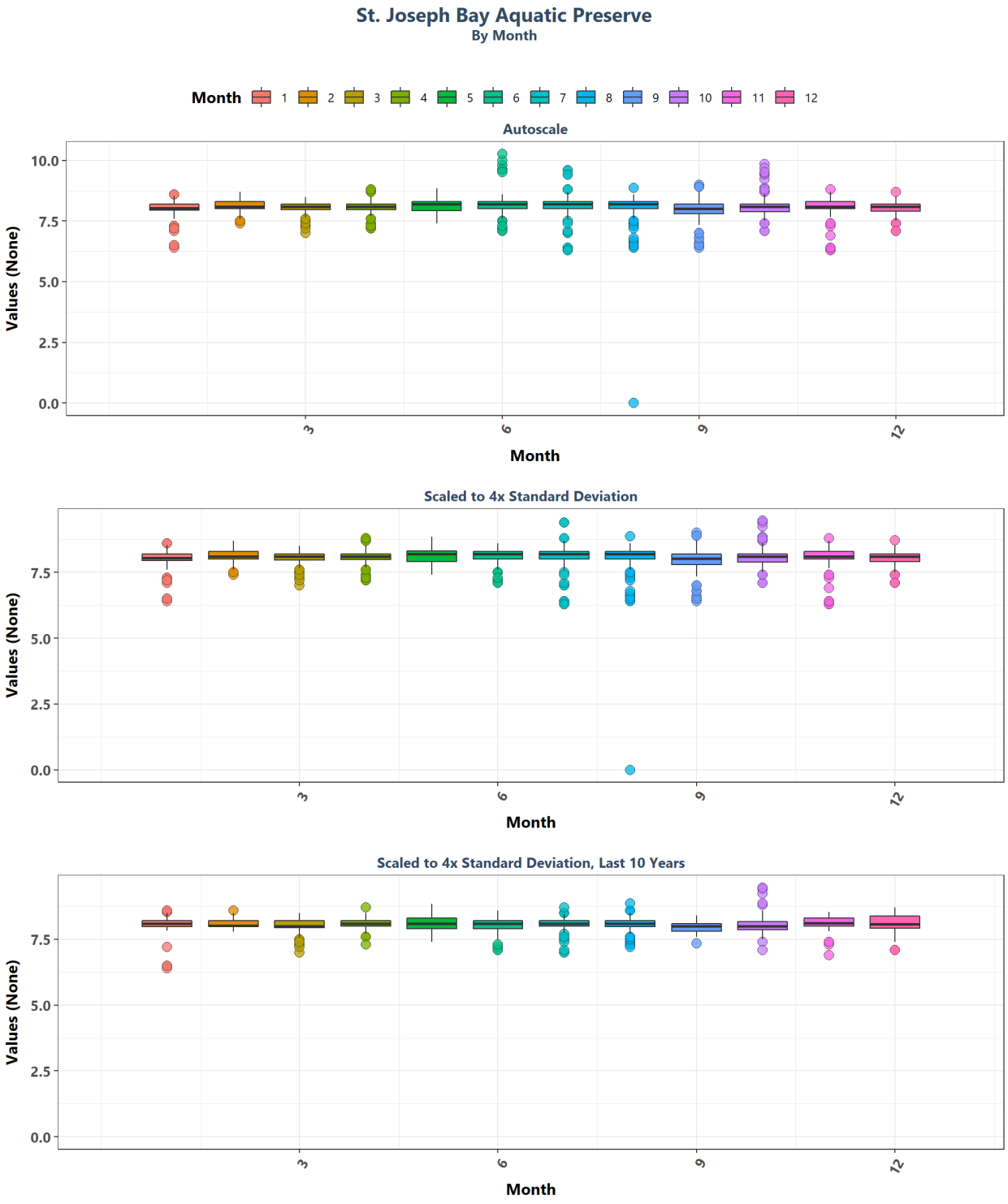


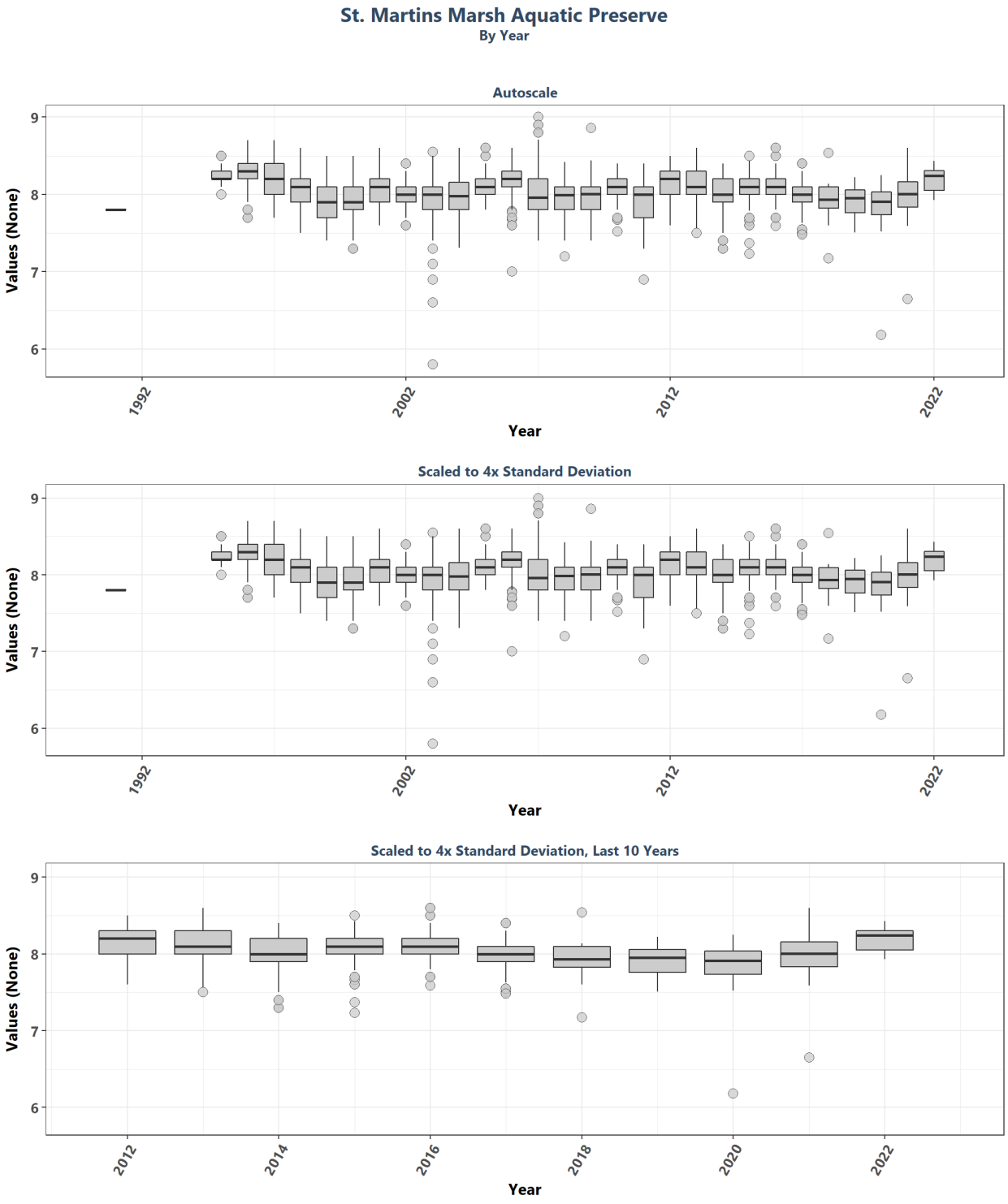

St. Andrews State Park Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

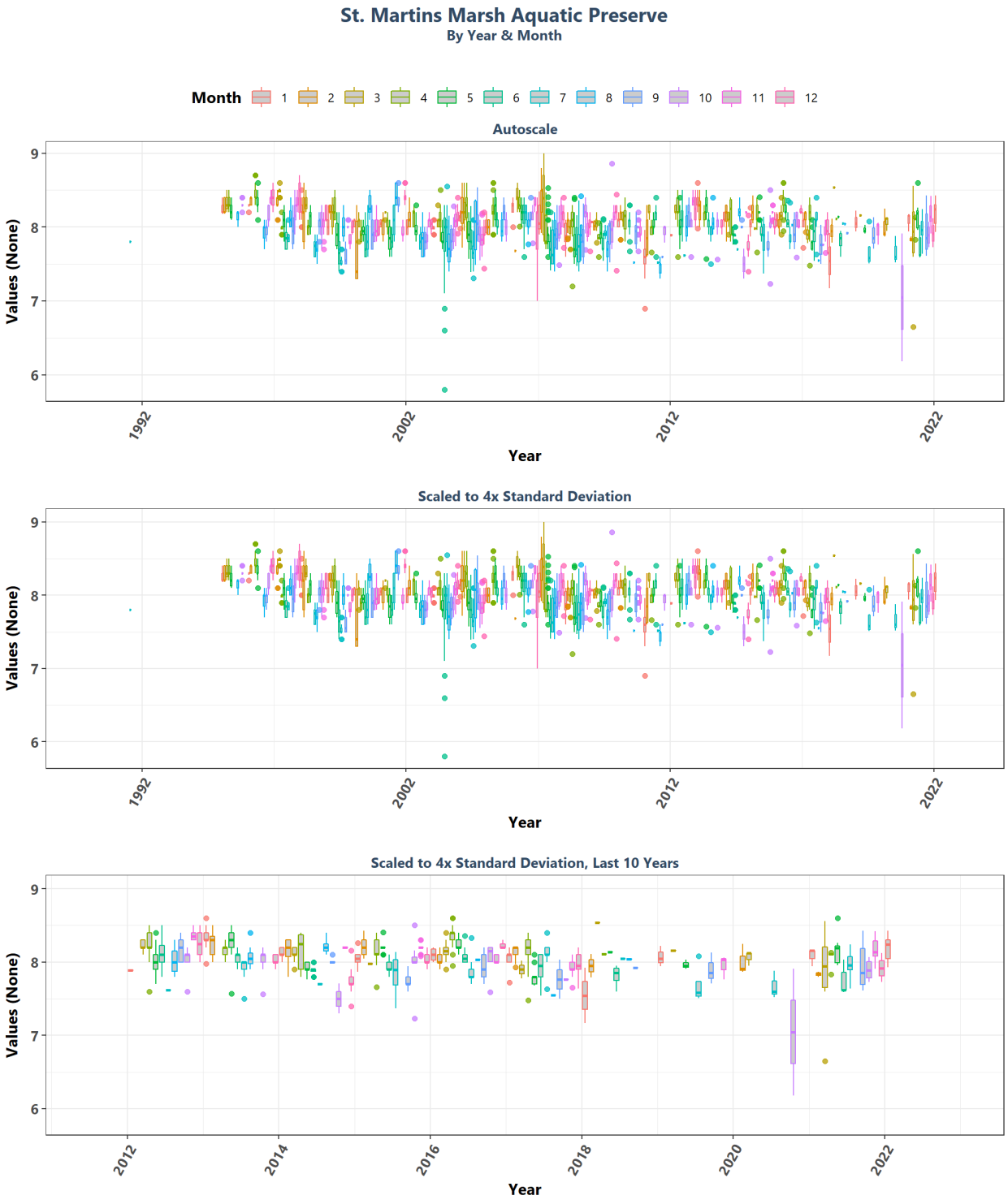

St. Andrews State Park Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
9.5
9.0
8.5
8.0
7.5
7.0
6.5
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
9.0
8.5
8.0
7.5
7.0
6.5
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
9.0
8.5
8.0
7.5
7.0
6.5
3
6
9
12
Month

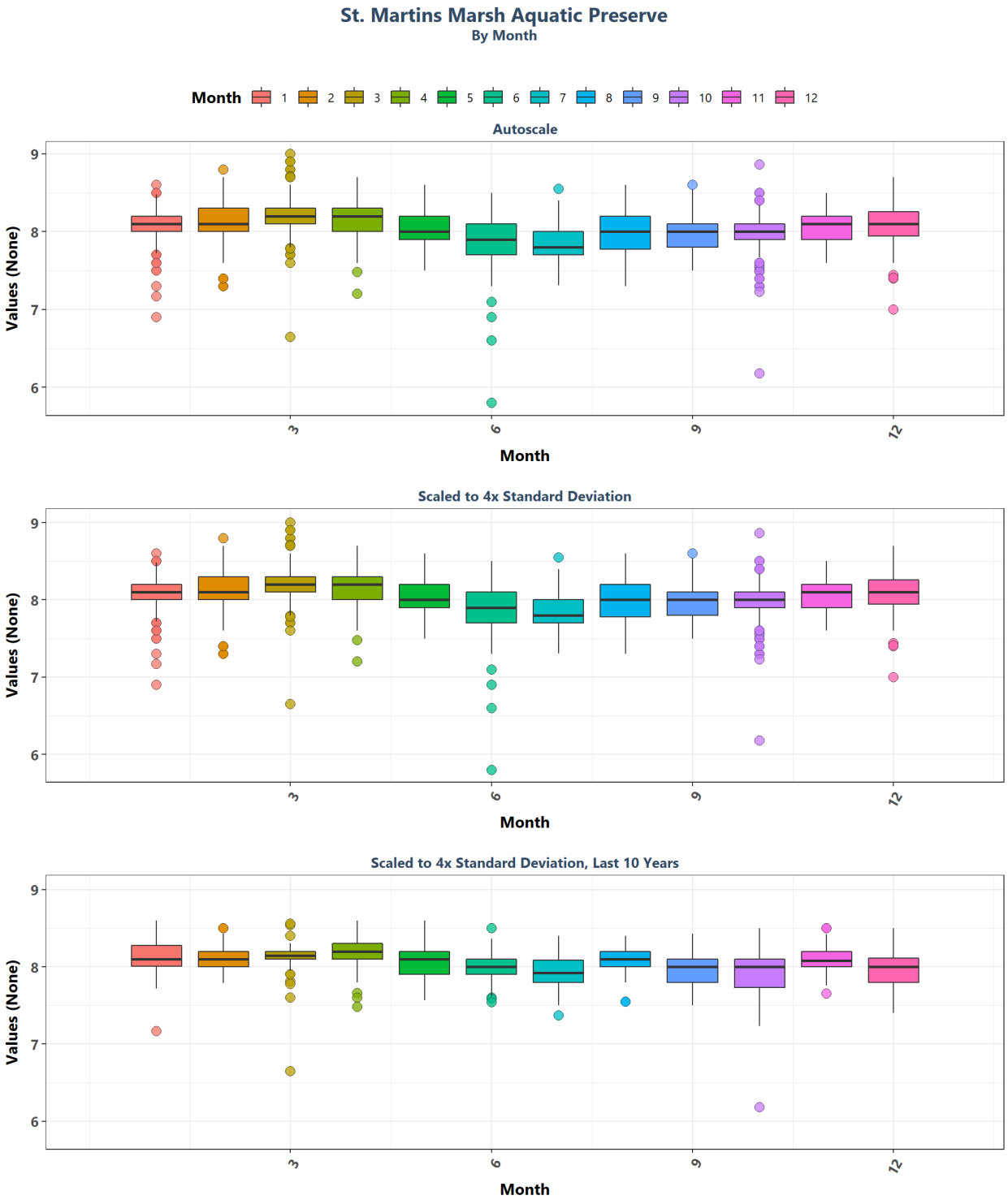

St. Joseph Bay Aquatic Preserve
By Year
Autoscale
Values (None)
10.0
7.5
5.0
2.5
0.0
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation
Values (None)
7.5
5.0
2.5
0.0
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
7.5
5.0
2.5
0.0
2012
2014
2016
2018
2020
2022
Year


St. Joseph Bay Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
10.0
7.5
5.0
2.5
0.0
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation
Values (None)
7.5
5.0
2.5
0.0
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
7.5
5.0
2.5
0.0
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 (None)
10.0
7.5
5.0
2.5
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
7.5
5.0
2.5
0.0
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
7.5
5.0
2.5
0.0
3
6
9
12
Month

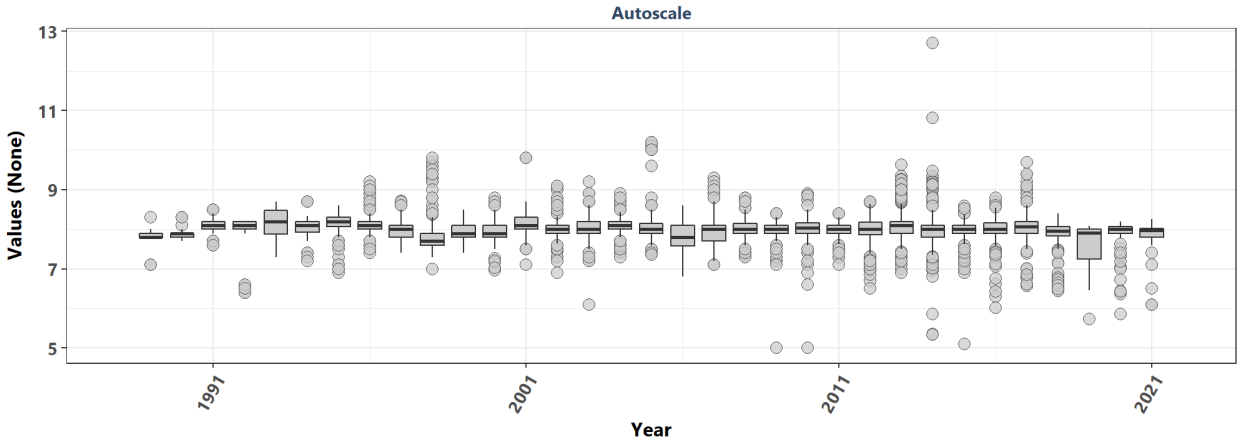

St. Martins Marsh Aquatic Preserve
By Year
Autoscale
Values (None)
9
8
7
6
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation
Values (None)
9
8
7
6
1992
2002
2012
2022
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
9
8
7
6
2012
2014
2016
2018
2020
2022
Year

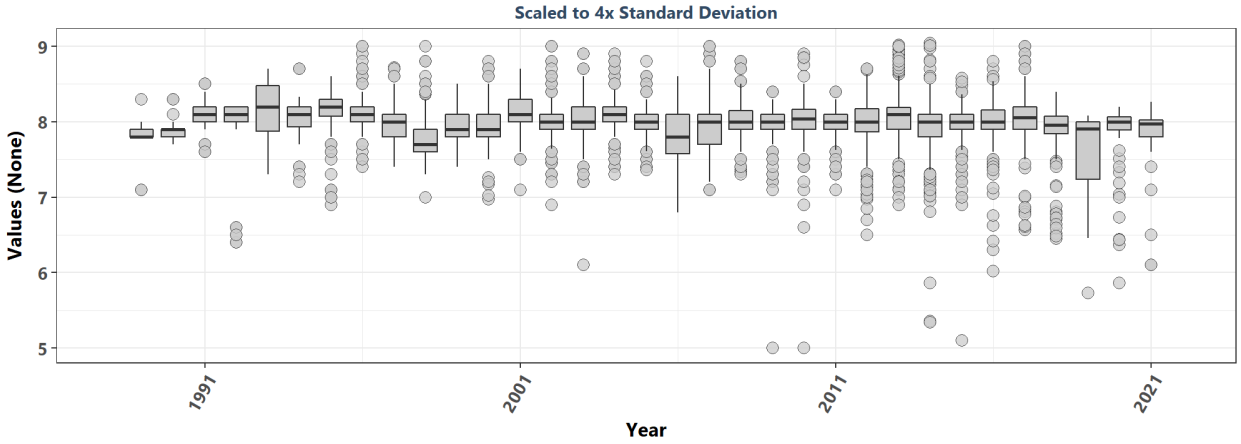

St. Martins Marsh Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

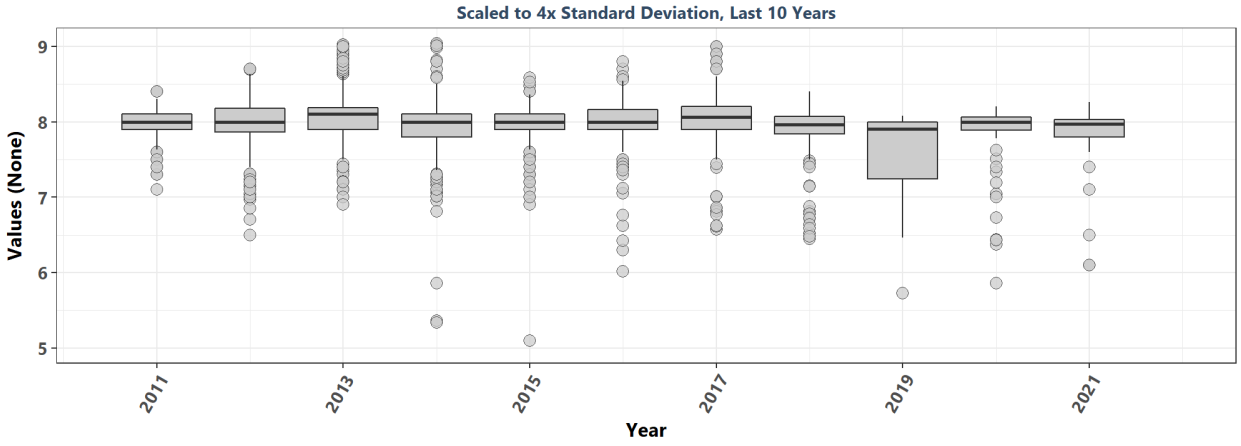

St. Martins Marsh Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

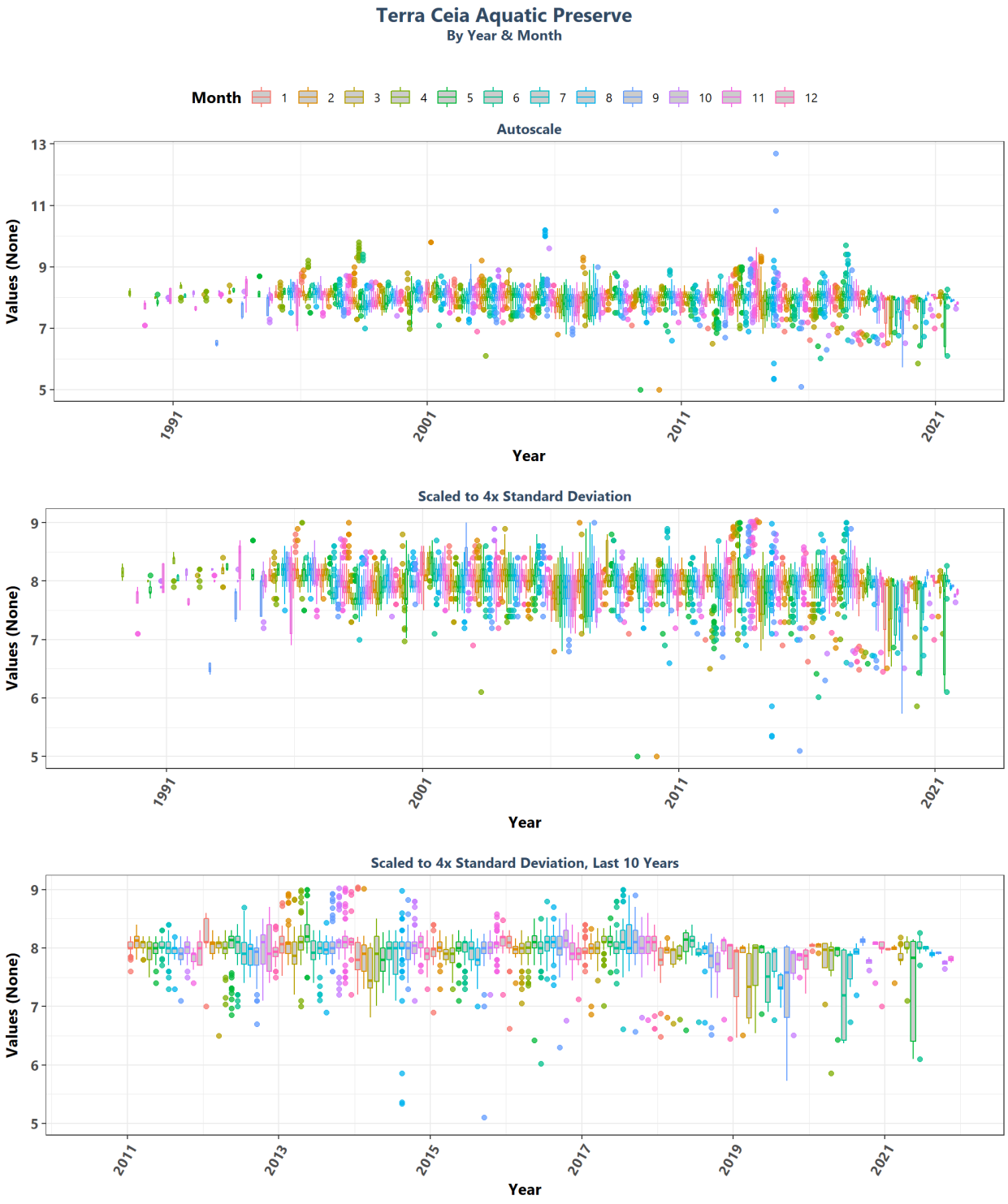
Terra Ceia Aquatic Preserve

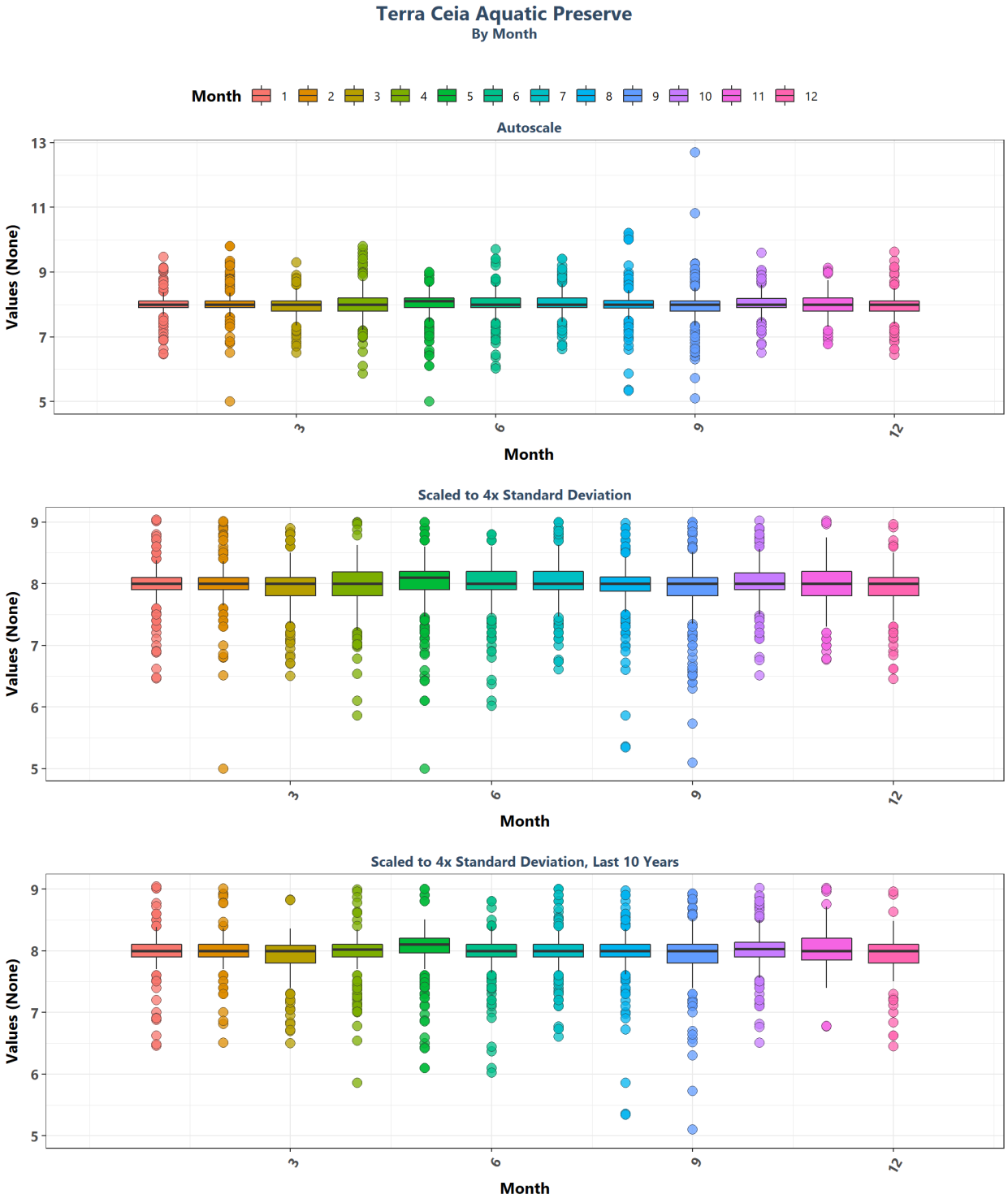
By Year





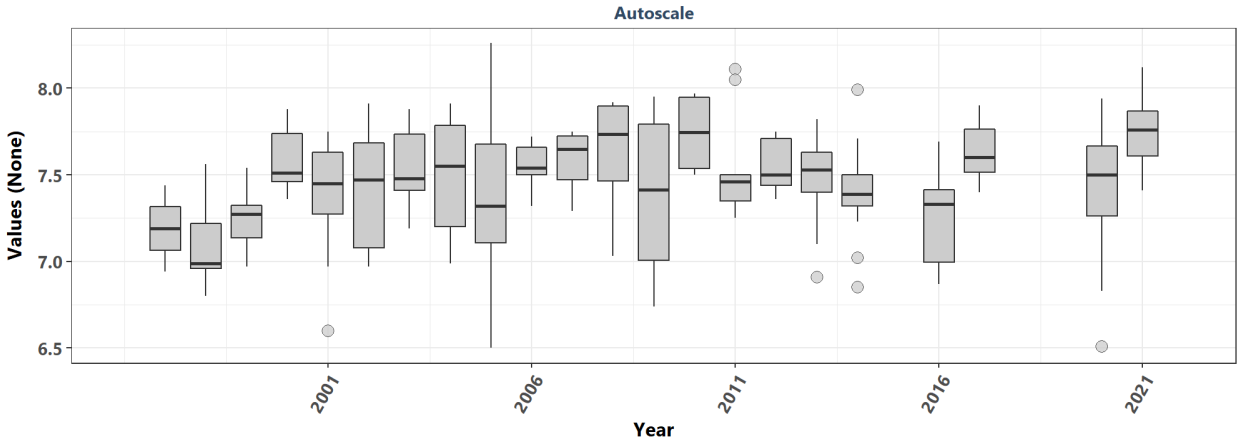


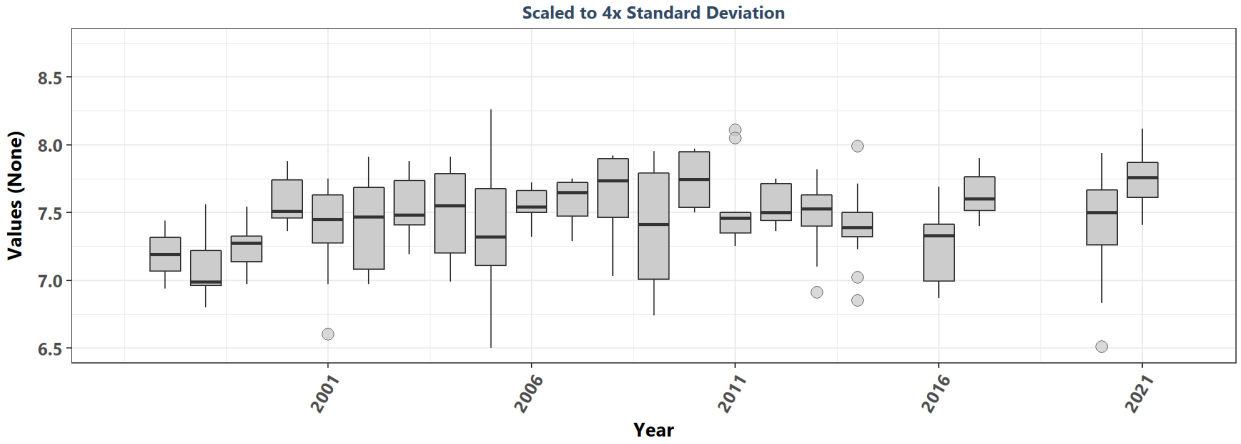

Terra Ceia Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year

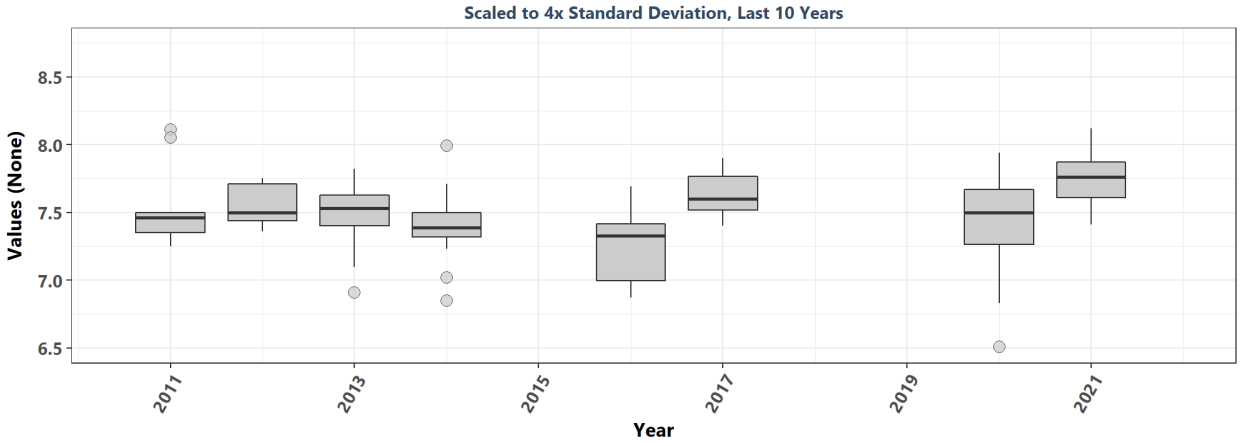

Terra Ceia Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Month
Scaled to 4x Standard Deviation
Values (None)
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Month

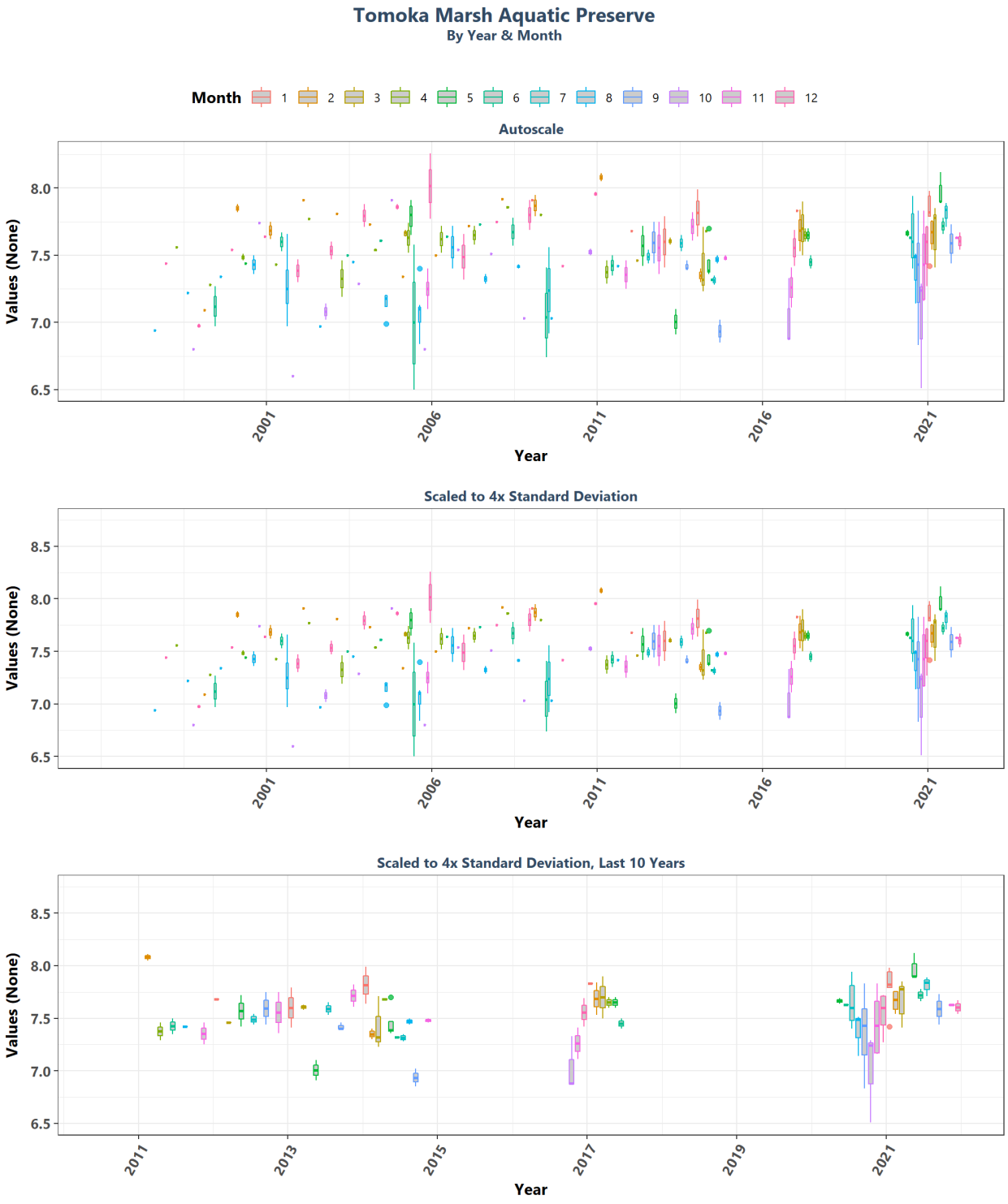
Tomoka Marsh Aquatic Preserve

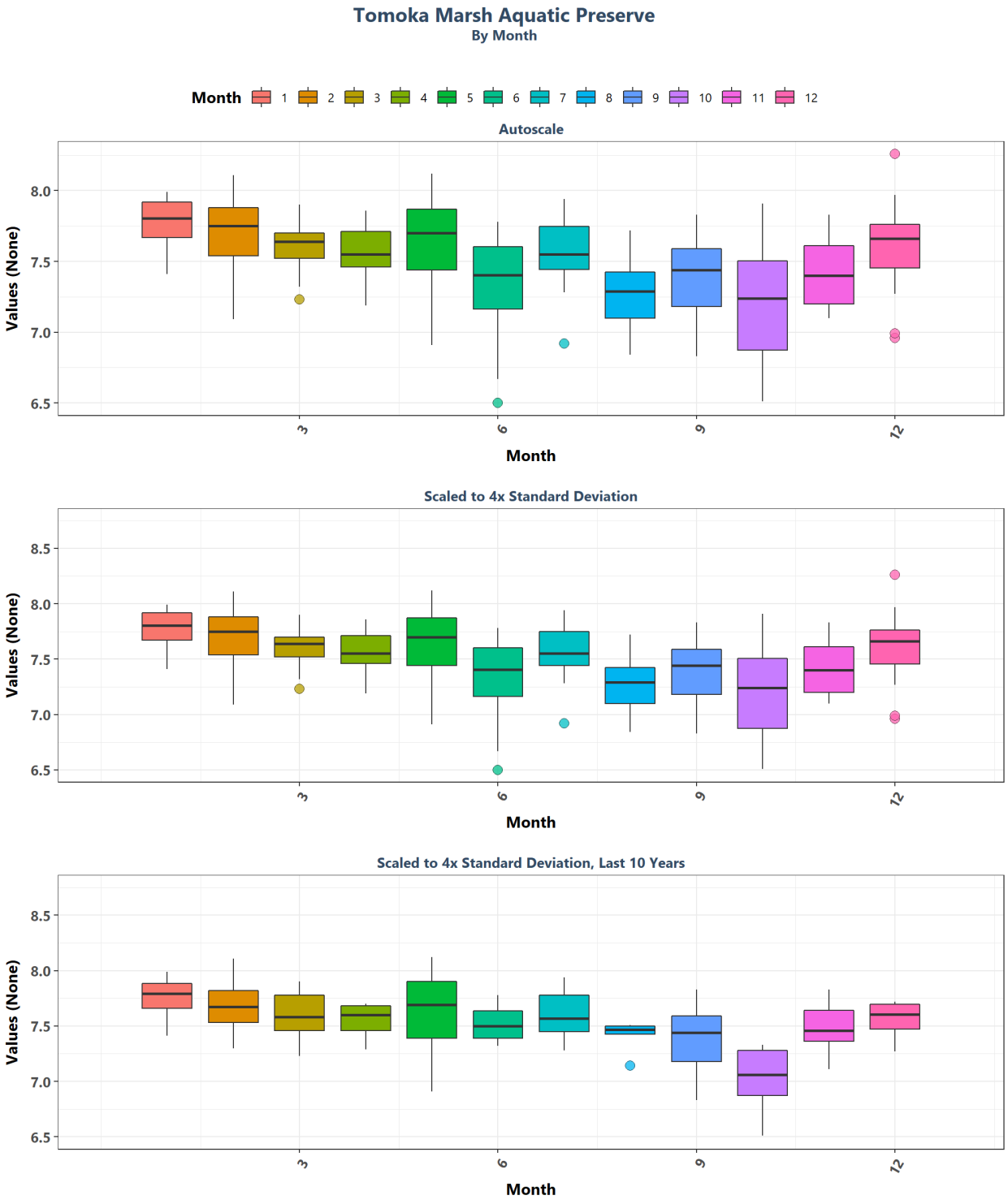
By Year





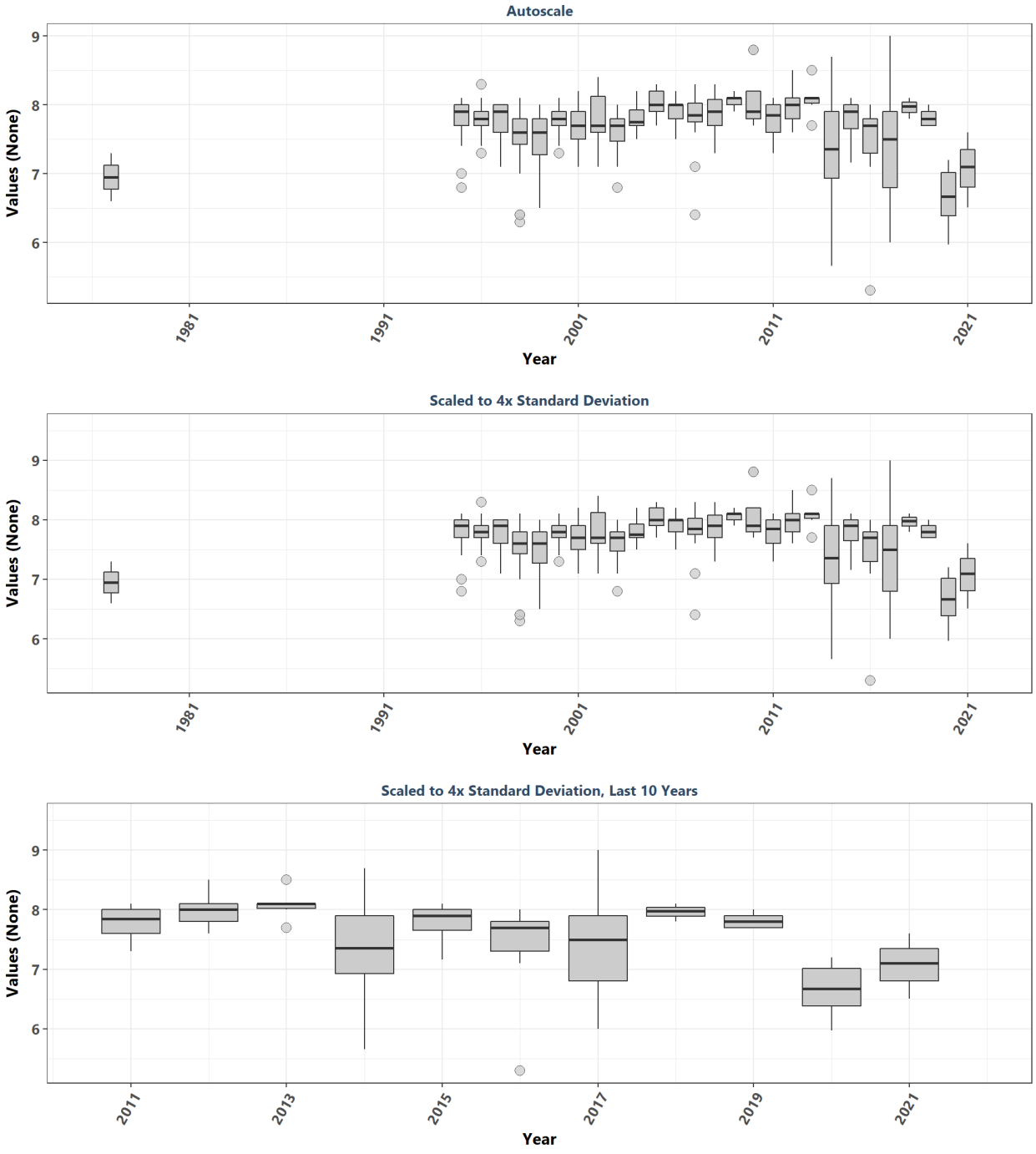


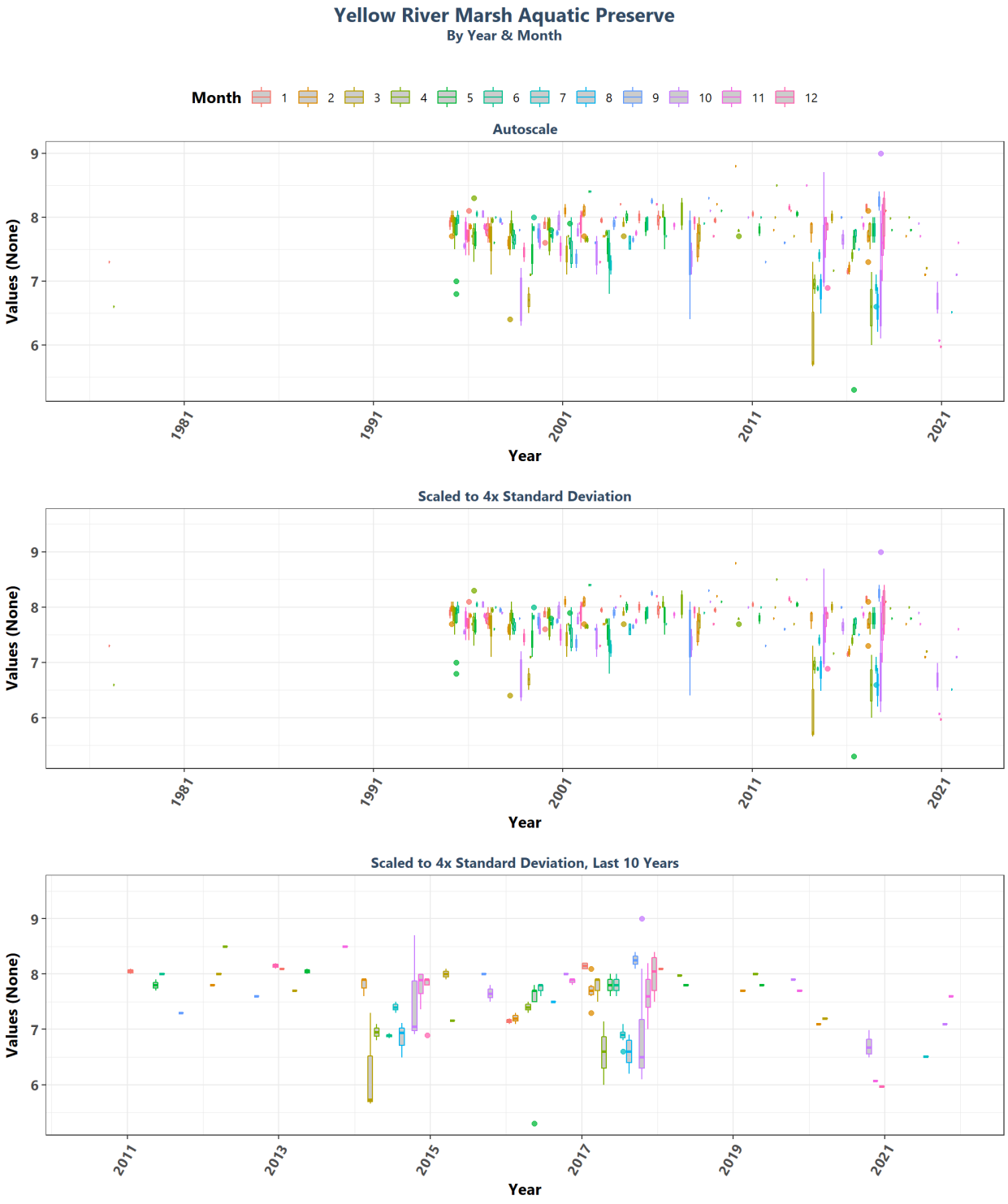

Tomoka Marsh Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
8.0
7.5
7.0
6.5
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation
Values (None)
8.5
8.0
7.5
7.0
6.5
2001
2006
2011
2016
2021
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
8.5
8.0
7.5
7.0
6.5
2011
2013
2015
2017
2019
2021
Year

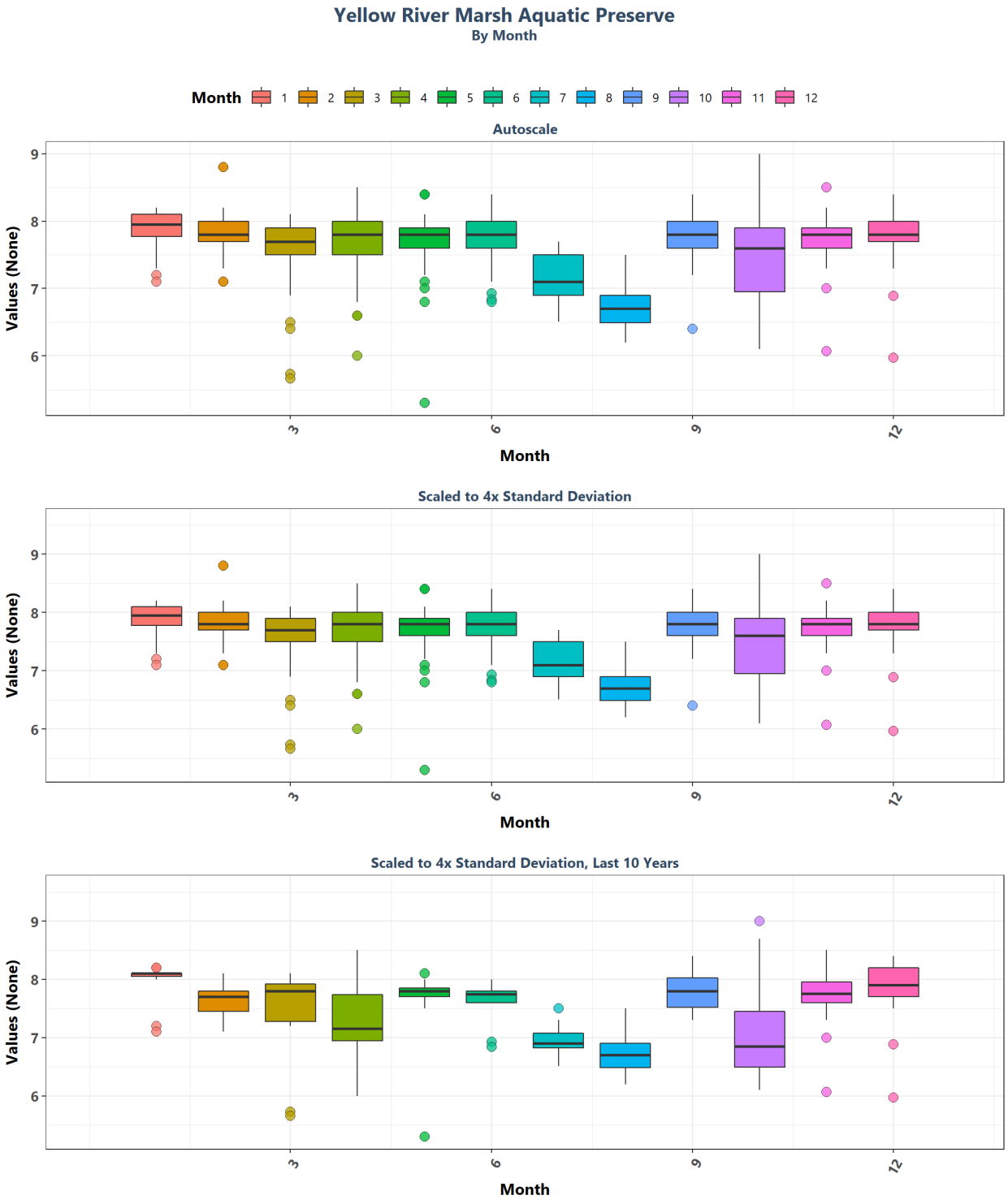

Tomoka Marsh Aquatic Preserve
By Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
8.0
7.5
7.0
6.5
3
6
9
12
Month
Scaled to 4x Standard Deviation
Values (None)
8.5
8.0
7.5
7.0
6.5
3
6
9
12
Month
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
8.5
8.0
7.5
7.0
6.5
3
6
9
12
Month

Yellow River Marsh Aquatic Preserve

By Year




Yellow River Marsh Aquatic Preserve
By Year & Month
Month 1 2 3 4 5 6 7 8 9 10 11 12
Autoscale
Values (None)
Year
Scaled to 4x Standard Deviation
Values (None)
Year
Scaled to 4x Standard Deviation, Last 10 Years
Values (None)
Year




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