Goodyear_WorkEx

MoWater Goodyear Team

6/23/2020

```
library(tidyverse); theme_set(theme_minimal())
theme_update(panel.grid.minor = element_blank())
library(lubridate)
library(rcartocolor)
library(RColorBrewer)
library(viridis)
library(scales)
library(rstatix)
library(dplyr)
library(ggpubr)
library(leaps) #For Best Subset
library(plotly) #for 3D plots
library(fields)
library(here) #Optional for loading files.
#   If library not install, call it by here::here(). If installed, just here().
library(webshot) #for knitting html output into pdf
```

Authors:
Ivan Ko
Blake Loosley
Lauren Varnado

MoWater

Goodyear Artificial Wetland Project

1. Setting Important Dates

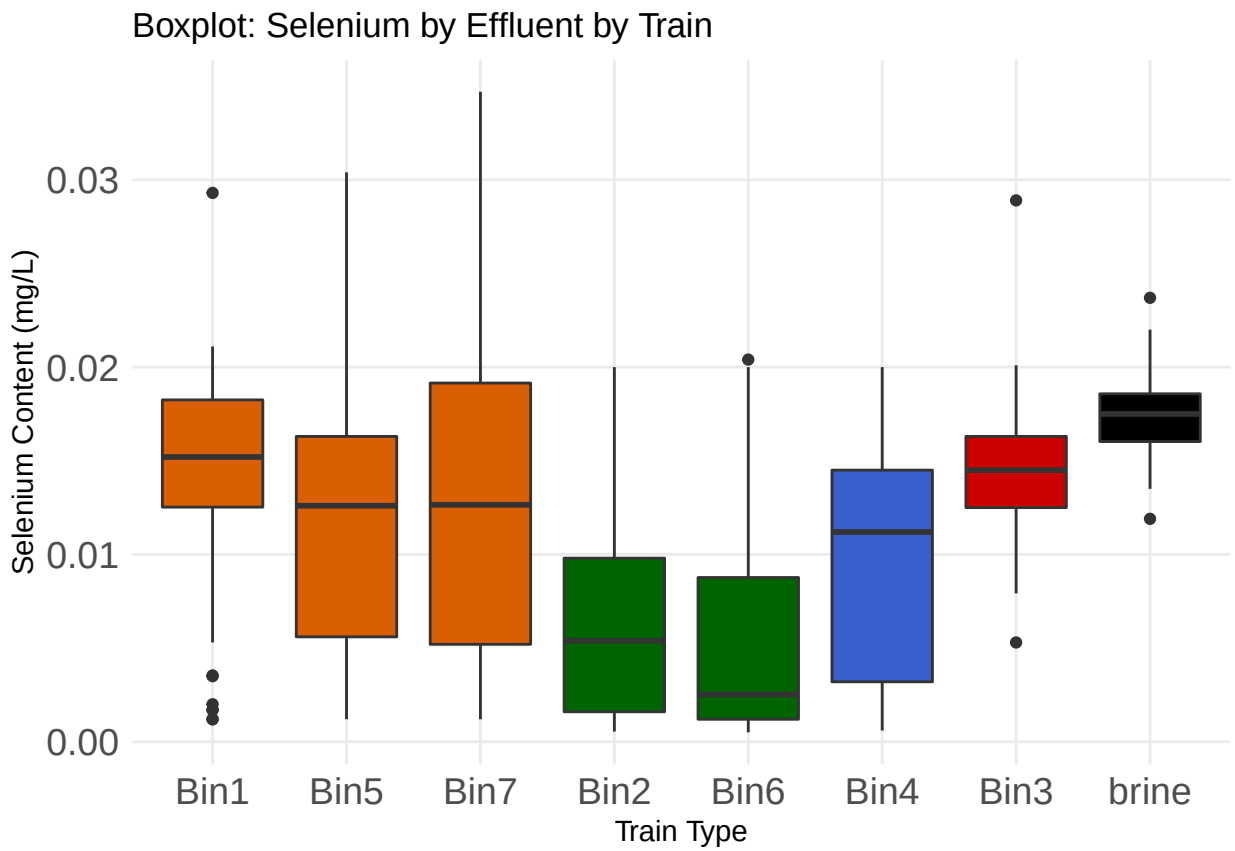
```
#train change date: relevant for bin 2 and 4.
#Before the change, bin 2 is train 3,
#This can be our starting date since we will only be ignoring 8 months of data.
trainChangeDate <- ymd( "2011-06-15")

#unstable periods
unstablePeriodStart <- ymd( "2014-04-01")
unstablePeriodEnd <- ymd( "2016-01-01") #rough est. according to Katie (stakeholder)

#Note: 2015-04-01 may be set to 2015-01-01 because the data doesn't look right.
#There's a spike in data around Jan 2015 that should be grouped with the next
# performance period,hence this choice.

#set periods: most bins have different perfmance periods!
```

```
#bin1, 5, 6, 7
bin1567Period1End <- ymd( "2012-03-01")
bin1567Period2End <- ymd( "2015-04-01")
bin1567Period3End <- ymd( "2017-04-01")
bin1567Periods <- c(bin1567Period1End, bin1567Period2End, bin1567Period3End)

#periods for bin2
bin2Period1End <- ymd( "2015-04-01")
bin2Period2End <- ymd( "2017-04-01")
bin2Periods <- c(bin2Period1End, bin2Period2End)

#periods for bin3, bin4
bin34Period1End <- ymd( "2015-04-01")
bin34Period2End <- ymd( "2016-12-01")
bin34Periods <- c(bin34Period1End, bin34Period2End)
```

2. Exploratory Analysis

2.1 Boxplot on Bin Selenium level

```
#different boxplot grouped by train type
dfTest <- dfDataSel
dfTest$ID <- factor(dfTest$ID , levels=c("Bin1", "Bin5", "Bin7", "Bin2",
                                         "Bin6", "Bin4", "Bin3", "brine"))

boxSelTCGroup <- dfTest %>%
    ggplot(aes(x = ID, y = Selenium)) +
    geom_boxplot(fill = c("#D95F02", "#D95F02", "#D95F02",
                          "darkgreen", "darkgreen", "royalblue3",
                          "red3", "black")) +
    xlab("Train Type") +
    ylab("Selenium Content (mg/L)") +
    labs(title= "Boxplot: Selenium by Effluent by Train") +
    theme(legend.position = "none", axis.text=element_text(size=14))

boxSelTCGroup
```



In this Boxplot, it is shown that Bin 3 and Bin 1 have a small range with most of the data occurring well above the Selenium threshold. However, there are a couple outliers that produce more successful Selenium concentrations. Additionally, Bin 2 and Bin 6 have the most consistently low Selenium concentration values compared to the other bins. In other words, Bins 2 and 6 appear to be the only bins that are skewed towards higher values whereas the other bins are skewed toward the lower values. At face value, it appears that Bins 2 and 6 seem to be the best for removing Selenium since they have the lowest medians.

2.2 Estimated Marginal Means Testing

```
#ANOVA
#Uses library(emmeans)
#test for homogeneity; compares the behavior (slope) with the addition of covariate

#-- by Bin --

#COD vs sel
anoSelvCOD <- anova_test(Selenium ~ COD * ID, data = dfDataSel)
```

```
## Warning: NA detected in rows: 1,3,5,9,10,11,13,15,17,26,28,30,31,32,34,36,55,57,59,63,64,65,66,68,70
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```r
get_anova_table(anoSelvCOD)
```

```
## ANOVA Table (type II tests)
##
##   Effect DFn DFd      F        p p<.05   ges
## 1    COD   1 280 22.756 2.96e-06     * 0.075
## 2     ID   7 280 22.157 8.94e-24     * 0.356
## 3 COD:ID   7 280  1.734 1.01e-01       0.042
```

```r
# maybe covariant! p = .1

#COD + Temp vs sel
anoSelvCODT <- anova_test(Selenium ~ (COD + Temp..Celsius) * ID, data = dfDataSel)
```

```
## Warning: NA detected in rows: 1,3,4,5,6,7,8,9,10,11,12,13,14,15,16,17,18,19,20,21,23,24,25,26,27,28,
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```r
get_anova_table(anoSelvCODT)
```

```
## ANOVA Table (type II tests)
##
##             Effect DFn DFd      F     p p<.05   ges
## 1              COD   1  15 16.130 0.001     * 0.518
## 2    Temp..Celsius   1  15  8.628 0.010     * 0.365
## 3               ID   7  15  5.374 0.003     * 0.715
## 4           COD:ID   7  15  1.970 0.128       0.479
## 5 Temp..Celsius:ID   7  15  1.323 0.306       0.382
```

```r
# maybe covariant! p = .306

#pH vs sel
anoSelvpH <- anova_test(Selenium ~ pH * ID, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,23,24,25,27,29,31,35,36
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```r
get_anova_table(anoSelvpH)
```

```
## ANOVA Table (type II tests)
##
##   Effect DFn DFd      F        p p<.05   ges
## 1     pH   1  73 15.174 2.15e-04     * 0.172
## 2     ID   7  73  5.147 8.57e-05     * 0.330
## 3  pH:ID   7  73  0.957 4.69e-01       0.084
```

```r
# is covariant! p = 0.469

#T vs sel
anoSelvT <- anova_test(Selenium ~ Temp..Celsius * ID, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,23,24,25,27,29,31,35,36
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```r
get_anova_table(anoSelvT)
```

```
## ANOVA Table (type II tests)
##
##             Effect DFn DFd     F        p p<.05   ges
## 1    Temp..Celsius   1  73 0.245 0.622000       0.003
## 2               ID   7  73 5.008 0.000114     * 0.324
## 3 Temp..Celsius:ID   7  73 0.975 0.456000       0.085
```

```r
# is covariant! p = 0.45

#DO vs sel
anoSelvDO <- anova_test(Selenium ~ DO.mg.L * ID, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,22,23,24,25,26,27,28,29
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```r
get_anova_table(anoSelvDO)
```

```
## ANOVA Table (type II tests)
##
##       Effect DFn DFd     F     p p<.05   ges
## 1    DO.mg.L   1  23 0.969 0.335       0.040
## 2         ID   7  23 0.796 0.599       0.195
## 3 DO.mg.L:ID   7  23 0.698 0.673       0.175
```

```r
# highest is covariant! p = 0.67

#Nit vs Sel
anoSelvNit <- anova_test(Selenium ~ Nitrate * ID, data = dfDataSel)
```

```
## Warning: NA detected in rows: 1,17,32,34,55,85,87,138,140,162,188,190,211,241,243,252,264,276,294,295
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```r
get_anova_table(anoSelvNit)
```

```
## ANOVA Table (type II tests)
##
##       Effect DFn DFd       F        p p<.05   ges
## 1    Nitrate   1 369 381.624 7.33e-59     * 0.508
## 2         ID   7 369   2.041 4.90e-02     * 0.037
## 3 Nitrate:ID   7 369   6.851 1.15e-07     * 0.115
```

```r
#P is low so it's bad = = 1.15e-7

#Phosphorus vs Sel
anoSelvPho <- anova_test(Selenium ~ Phosphorus * ID, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,5,6,9,10,11,13,15,17,26,28,30,32,34,36,38,41,45,47,48,52,57,59,60,63
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```r
get_anova_table(anoSelvPho)
```

```
## ANOVA Table (type II tests)
```

```
##
##          Effect DFn DFd      F        p p<.05   ges
## 1    Phosphorus   1 257 17.468 4.01e-05     * 0.064
## 2            ID   7 257 13.535 7.08e-15     * 0.269
## 3 Phosphorus:ID   7 257  1.388 2.10e-01       0.036
```

```
# maybe is covariant! p = 0.21

#-- by veg --

# DO test but for veg type
anoSelvDOVeg <- anova_test(Selenium ~ DO.mg.L * Veg, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,22,23,24,25,26,27,28,29
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvDOVeg)
```

```
## ANOVA Table (type II tests)
##
##        Effect DFn DFd     F     p p<.05   ges
## 1     DO.mg.L   1  31 3.128 0.087       0.092
## 2         Veg   3  31 1.334 0.281       0.114
## 3 DO.mg.L:Veg   3  31 0.132 0.941       0.013
```

```
# p = 0.986

#pH test but for veg type
anoSelvpHVeg <- anova_test(Selenium ~ pH * Veg, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,23,24,25,27,29,31,35,36
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvpHVeg)
```

```
## ANOVA Table (type II tests)
##
##   Effect DFn DFd      F        p p<.05   ges
## 1     pH   1  81 13.391 0.000449     * 0.142
## 2    Veg   3  81  4.696 0.004000     * 0.148
## 3 pH:Veg   3  81  1.364 0.260000       0.048
```

```
# p = 0.187!

#COD test but for veg type
anoSelvCODVeg <- anova_test(Selenium ~ COD * Veg, data = dfDataSel)
```

```
## Warning: NA detected in rows: 1,3,5,9,10,11,13,15,17,26,28,30,31,32,34,36,55,57,59,63,64,65,66,68,70
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvCODVeg)
```

```
## ANOVA Table (type II tests)
##
```

```
##    Effect DFn DFd      F        p p<.05   ges
## 1     COD   1 288 20.080 1.07e-05     * 0.065
## 2     Veg   3 288 28.879 2.36e-16     * 0.231
## 3 COD:Veg   3 288  0.309 8.19e-01       0.003
```

```
# p > 0.7

#T test but for veg type
anoSelvTVeg <- anova_test(Selenium ~ Temp..Celsius * Veg, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,23,24,25,27,29,31,35,36
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvTVeg)
```

```
## ANOVA Table (type II tests)
##
##              Effect DFn DFd     F     p p<.05   ges
## 1     Temp..Celsius   1  81 0.237 0.628       0.003
## 2               Veg   3  81 5.226 0.002     * 0.162
## 3 Temp..Celsius:Veg   3  81 0.812 0.491       0.029
```

```
# p = 0.35

#Nit test but for veg type
anoSelvNitVeg <- anova_test(Selenium ~ Nitrate * Veg, data = dfDataSel)
```

```
## Warning: NA detected in rows: 1,17,32,34,55,85,87,138,140,162,188,190,211,241,243,252,264,276,294,295
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvNitVeg)
```

```
## ANOVA Table (type II tests)
##
##        Effect DFn DFd       F        p p<.05   ges
## 1     Nitrate   1 377 475.129 9.54e-69     * 0.558
## 2         Veg   3 377   2.855 3.70e-02     * 0.022
## 3 Nitrate:Veg   3 377   7.566 6.31e-05     * 0.057
```

```
# p = 0.02

#Phosphorus test but for media type
anoSelvPhoVeg <- anova_test(Selenium ~ Phosphorus * Veg, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,5,6,9,10,11,13,15,17,26,28,30,32,34,36,38,41,45,47,48,52,57,59,60,63
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvPhoVeg)
```

```
## ANOVA Table (type II tests)
##
##           Effect DFn DFd      F        p p<.05   ges
## 1     Phosphorus   1 265 19.993 1.15e-05     * 0.070
## 2            Veg   3 265 17.951 1.23e-10     * 0.169
```

```
## 3 Phosphorus:Veg   3 265  1.387 2.47e-01       0.015
```

```
# p = 0.04

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

#-- by train --

# DO test but for train type
anoSelvDOTrain <- anova_test(Selenium ~ DO.mg.L * TrainGroup, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,22,23,24,25,26,27,28,29
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvDOTrain)
```

```
## ANOVA Table (type II tests)
##
##               Effect DFn DFd     F     p p<.05   ges
## 1            DO.mg.L   1  29 1.308 0.262       0.043
## 2         TrainGroup   4  29 1.417 0.253       0.163
## 3 DO.mg.L:TrainGroup   4  29 0.931 0.459       0.114
```

```
# p > 0.459!

#pH test but for train type
anoSelvpHTrain <- anova_test(Selenium ~ pH * TrainGroup, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,23,24,25,27,29,31,35,36
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvpHTrain)
```

```
## ANOVA Table (type II tests)
##
##          Effect DFn DFd      F        p p<.05   ges
## 1            pH   1  79 14.140 3.24e-04     * 0.152
## 2    TrainGroup   4  79  8.326 1.17e-05     * 0.297
## 3 pH:TrainGroup   4  79  0.736 5.70e-01       0.036
```

```
# p = 0.57!

#COD test but for train type
anoSelvCODTrain <- anova_test(Selenium ~ COD * TrainGroup, data = dfDataSel)
```

```
## Warning: NA detected in rows: 1,3,5,9,10,11,13,15,17,26,28,30,31,32,34,36,55,57,59,63,64,65,66,68,70
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvCODTrain)
```

```
## ANOVA Table (type II tests)
##
##           Effect DFn DFd      F        p p<.05   ges
## 1            COD   1 286 20.400 9.20e-06     * 0.067
```

```
## 2     TrainGroup   4 286 35.636 3.73e-24     * 0.333
## 3 COD:TrainGroup   4 286  2.205 6.90e-02       0.030
```

```
# p > 0.069

#T test but for train type
anoSelvTTrain <- anova_test(Selenium ~ Temp..Celsius * TrainGroup, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,23,24,25,27,29,31,35,36
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvTTrain)
```

```
## ANOVA Table (type II tests)
##
##                      Effect DFn DFd     F        p p<.05   ges
## 1            Temp..Celsius   1  79 0.191 6.64e-01       0.002
## 2               TrainGroup   4  79 8.752 6.69e-06     * 0.307
## 3 Temp..Celsius:TrainGroup   4  79 1.390 2.45e-01       0.066
```

```
# p = 0.24

#Nit test but for train type
anoSelvNitTrain <- anova_test(Selenium ~ Nitrate * TrainGroup, data = dfDataSel)
```

```
## Warning: NA detected in rows: 1,17,32,34,55,85,87,138,140,162,188,190,211,241,243,252,264,276,294,295
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvNitTrain)
```

```
## ANOVA Table (type II tests)
##
##                Effect DFn DFd       F        p p<.05   ges
## 1             Nitrate   1 375 387.700 8.91e-60     * 0.508
## 2          TrainGroup   4 375   2.842 2.40e-02     * 0.029
## 3 Nitrate:TrainGroup   4 375  11.896 4.14e-09     * 0.113
```

```
# p = 4.14e-9

#Phosphorus test but for train type
anoSelvPhoTrain <- anova_test(Selenium ~ Phosphorus * TrainGroup, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,5,6,9,10,11,13,15,17,26,28,30,32,34,36,38,41,45,47,48,52,57,59,60,63
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvPhoTrain)
```

```
## ANOVA Table (type II tests)
##
##                    Effect DFn DFd      F        p p<.05   ges
## 1            Phosphorus   1 263 14.289 1.94e-04     * 0.052
## 2            TrainGroup   4 263 21.613 1.97e-15     * 0.247
## 3 Phosphorus:TrainGroup   4 263  1.925 1.07e-01       0.028
```

```
# p = 0.1

#--- by Media ---

# DO test but for media type
anoSelvDOMedia <- anova_test(Selenium ~ DO.mg.L * MediaType, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,22,23,24,25,26,27,28,29
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvDOMedia)
```

```
## ANOVA Table (type II tests)
##
##              Effect DFn DFd     F     p p<.05   ges
## 1           DO.mg.L   1  29 2.176 0.151       0.070
## 2         MediaType   4  29 0.208 0.932       0.028
## 3 DO.mg.L:MediaType   4  29 0.332 0.854       0.044
```

```
# p > 0.825!

#pH test but for media type
anoSelvpHMedia <- anova_test(Selenium ~ pH * MediaType, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,23,24,25,27,29,31,35,36
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvpHMedia)
```

```
## ANOVA Table (type II tests)
##
##         Effect DFn DFd      F        p p<.05   ges
## 1           pH   1  79 14.647 0.000258     * 0.156
## 2    MediaType   4  79  4.160 0.004000     * 0.174
## 3 pH:MediaType   4  79  0.417 0.796000       0.021
```

```
# p = 0.81.

#COD test but for media type
anoSelvCODMedia <- anova_test(Selenium ~ COD * MediaType, data = dfDataSel)
```

```
## Warning: NA detected in rows: 1,3,5,9,10,11,13,15,17,26,28,30,31,32,34,36,55,57,59,63,64,65,66,68,70
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvCODMedia)
```

```
## ANOVA Table (type II tests)
##
##          Effect DFn DFd      F        p p<.05   ges
## 1           COD   1 286 28.314 2.09e-07     * 0.090
## 2     MediaType   4 286 16.786 2.26e-12     * 0.190
## 3 COD:MediaType   4 286  0.510 7.28e-01       0.007
```

```
# p > 0.95

#T test but for media type
anoSelvTMedia <- anova_test(Selenium ~ Temp..Celsius * MediaType, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,4,5,6,7,8,9,10,11,12,13,14,15,16,18,19,20,21,23,24,25,27,29,31,35,36
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvTMedia)
```

```
## ANOVA Table (type II tests)
##
##                     Effect DFn DFd     F     p p<.05     ges
## 1           Temp..Celsius   1  79 0.070 0.791       0.00089
## 2               MediaType   4  79 4.295 0.003     * 0.17900
## 3 Temp..Celsius:MediaType   4  79 1.528 0.202       0.07200
```

```
# p = 0.115

#Nit test but for media type
anoSelvNitMedia <- anova_test(Selenium ~ Nitrate * MediaType, data = dfDataSel)
```

```
## Warning: NA detected in rows: 1,17,32,34,55,85,87,138,140,162,188,190,211,241,243,252,264,276,294,295
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvNitMedia)
```

```
## ANOVA Table (type II tests)
##
##              Effect DFn DFd       F        p p<.05   ges
## 1           Nitrate   1 375 535.451 3.16e-74     * 0.588
## 2         MediaType   4 375   2.462 4.50e-02     * 0.026
## 3 Nitrate:MediaType   4 375   3.710 6.00e-03     * 0.038
```

```
# p = 0.99!!!

#Phosphorus test but for media type
anoSelvPhoMedia <- anova_test(Selenium ~ Phosphorus * MediaType, data = dfDataSel)
```

```
## Warning: NA detected in rows: 3,5,6,9,10,11,13,15,17,26,28,30,32,34,36,38,41,45,47,48,52,57,59,60,63
## Removing this rows before the analysis.
```

```
## Coefficient covariances computed by hccm()
```

```
get_anova_table(anoSelvPhoMedia)
```

```
## ANOVA Table (type II tests)
##
##                  Effect DFn DFd      F        p p<.05   ges
## 1           Phosphorus   1 263 26.177 6.01e-07     * 0.091
## 2            MediaType   4 263 11.927 6.40e-09     * 0.154
## 3 Phosphorus:MediaType   4 263  1.386 2.39e-01       0.021
```

```
# p = 0.33
```

```
dfDataSel$MediaType <- as.character(dfDataSel$MediaType)
emmeans_test(Selenium ~ MediaType, covariate = Temp..Celsius, p.adjust.method = "bonferroni", data = df
```

```
## # A tibble: 10 x 8
##    .y.      group1 group2    df statistic        p   p.adj p.adj.signif
##  * <chr>    <chr>  <chr>  <dbl>     <dbl>    <dbl>   <dbl> <chr>
##  1 Selenium brine  GW        83     3.58  0.000578 0.00578 **
##  2 Selenium brine  MM        83     0.982 0.329    1       ns
##  3 Selenium brine  PM        83     3.10  0.00265  0.0265  *
##  4 Selenium brine  Soil      83     1.69  0.0955   0.955   ns
##  5 Selenium GW     MM        83    -2.60  0.0111   0.111   ns
##  6 Selenium GW     PM        83    -1.36  0.178    1       ns
##  7 Selenium GW     Soil      83    -1.71  0.0904   0.904   ns
##  8 Selenium MM     PM        83     1.88  0.0638   0.638   ns
##  9 Selenium MM     Soil      83     0.757 0.451    1       ns
## 10 Selenium PM     Soil      83    -0.829 0.410    1       ns
```

```
emmeans_test(Selenium ~ MediaType, p.adjust.method = "bonferroni", data = dfDataSel)
```

```
## # A tibble: 10 x 8
##    .y.      group1 group2    df statistic        p    p.adj p.adj.signif
##  * <chr>    <chr>  <chr>  <dbl>     <dbl>    <dbl>    <dbl> <chr>
##  1 Selenium brine  GW       411    6.78   4.17e-11 4.17e-10 ****
##  2 Selenium brine  MM       411    2.91   3.80e- 3 3.80e- 2 *
##  3 Selenium brine  PM       411    8.51   3.19e-16 3.19e-15 ****
##  4 Selenium brine  Soil     411    4.24   2.80e- 5 2.80e- 4 ***
##  5 Selenium GW     MM       411   -3.88   1.20e- 4 1.20e- 3 **
##  6 Selenium GW     PM       411   -0.0572 9.54e- 1 1.00e+ 0 ns
##  7 Selenium GW     Soil     411   -2.43   1.54e- 2 1.54e- 1 ns
##  8 Selenium MM     PM       411    4.85   1.74e- 6 1.74e- 5 ****
##  9 Selenium MM     Soil     411    1.38   1.68e- 1 1.00e+ 0 ns
## 10 Selenium PM     Soil     411   -2.99   3.00e- 3 3.00e- 2 *
```

```
dfDataSel$Veg <- as.character(dfDataSel$Veg)
emmeans_test(Selenium ~ Veg, covariate = Temp..Celsius, data = dfDataSel)
```

```
## # A tibble: 6 x 8
##   .y.      group1    group2       df statistic        p   p.adj p.adj.signif
## * <chr>    <chr>     <chr>     <dbl>     <dbl>    <dbl>   <dbl> <chr>
## 1 Selenium brine     VegType_A    84     2.30  0.0242   0.145   ns
## 2 Selenium brine     VegType_B    84     3.65  0.000458 0.00275 **
## 3 Selenium brine     VegType_C    84     3.13  0.00244  0.0146  *
## 4 Selenium VegType_A VegType_B    84     2.34  0.0215   0.129   ns
## 5 Selenium VegType_A VegType_C    84     1.54  0.127    0.764   ns
## 6 Selenium VegType_B VegType_C    84    -0.956 0.342    1       ns
```

```
emmeans_test(Selenium ~ Veg, p.adjust.method = "bonferroni", data = dfDataSel)
```

```
## # A tibble: 6 x 8
##   .y.      group1    group2       df statistic        p    p.adj p.adj.signif
## * <chr>    <chr>     <chr>     <dbl>     <dbl>    <dbl>    <dbl> <chr>
## 1 Selenium brine     VegType_A   412      5.57 4.59e- 8 2.75e- 7 ****
## 2 Selenium brine     VegType_B   412      9.91 6.37e-21 3.82e-20 ****
## 3 Selenium brine     VegType_C   412      8.40 7.30e-16 4.38e-15 ****
## 4 Selenium VegType_A VegType_B   412      6.94 1.57e-11 9.43e-11 ****
## 5 Selenium VegType_A VegType_C   412      4.67 4.03e- 6 2.42e- 5 ****
```

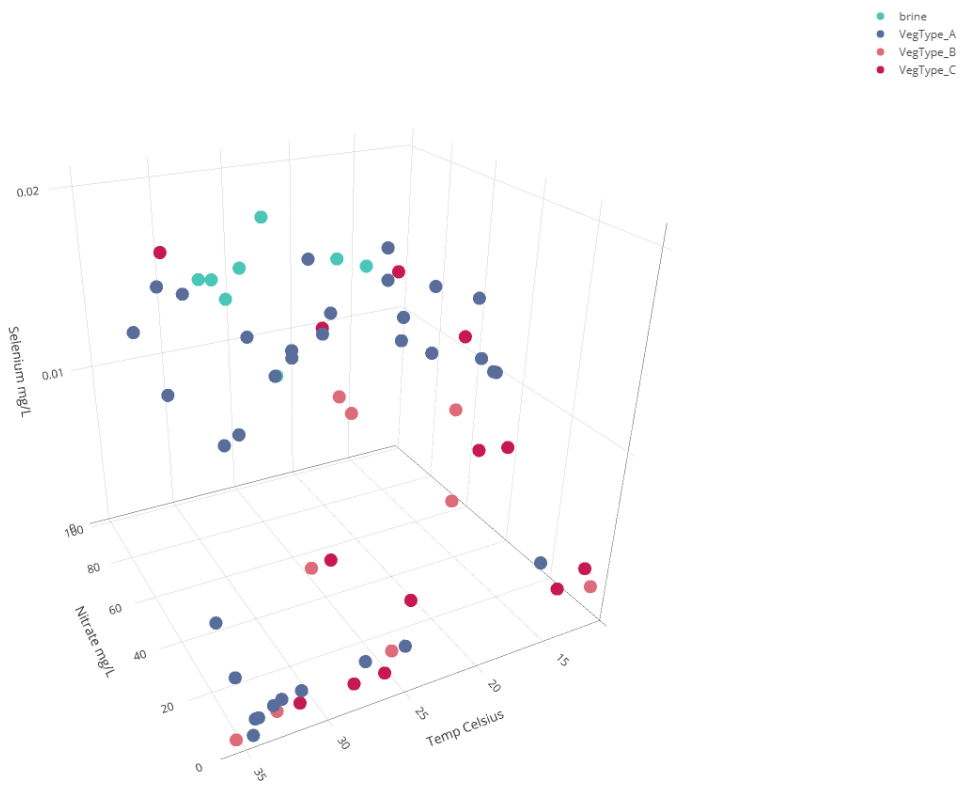
```
## 6 Selenium VegType_B VegType_C   412     -2.93 3.53e- 3 2.12e- 2 *
```

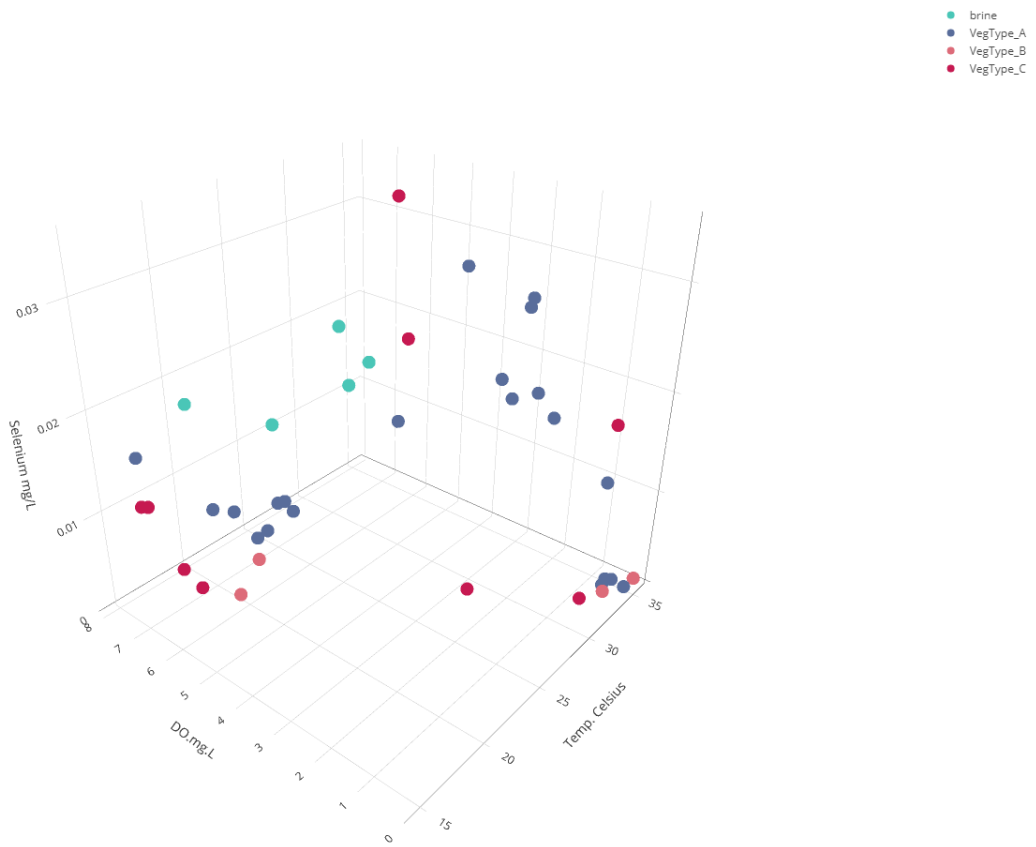
```
dfDataSel$ID <- as.character(dfDataSel$ID)
BinControlTemp <- emmeans_test(Selenium ~ ID, covariate = Temp..Celsius, p.adjust.method = "bonferroni"
BinContorlNone <- emmeans_test(Selenium ~ ID, p.adjust.method = "bonferroni", data = dfDataSel)
```

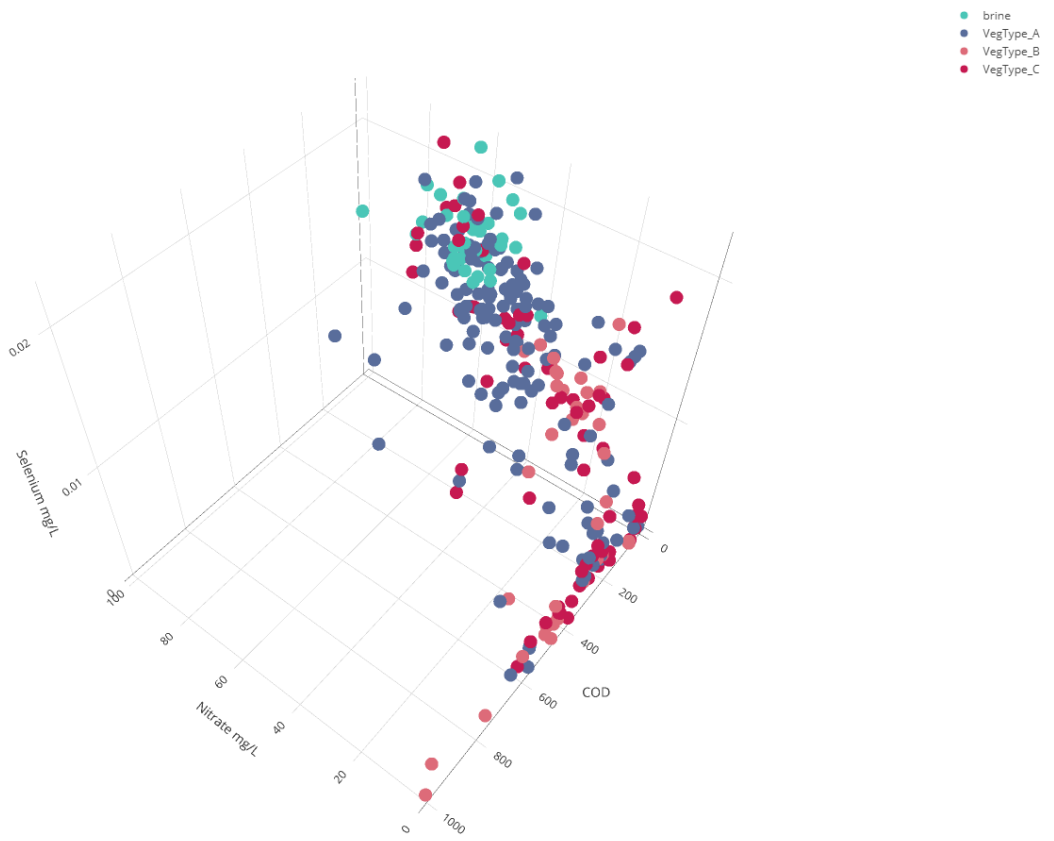
Estimated Marginal Means tests were run on available categorical variables to examine the effect of di
Primary findings from these tests showed that when controlling for temperature, there are no significan

2.3 3D Plots

```
#NOTE: This code is NOT run here when knitting due to the html output issue.

#Set color here
colorsScale <- c('#4AC6B7', '#1972A4', '#965F8A', '#FF7070', '#C61951')

#Temp vs Nit on Veg
fig <- plot_ly(dfT, x = ~Nitrate, y = ~Temp..Celsius,
               z = ~Selenium, color = ~Veg, colors = colorsScale)
fig <- fig %>% add_markers()
fig <- fig %>% layout(scene = list(xaxis = list(title = 'Nitrate mg/L'),
                                   yaxis = list(title = 'Temp Celsius'),
                                   zaxis = list(title = 'Selenium mg/L')))
fig

#---

#Temp vs DO on Veg
figTDOV <- plot_ly(dfT, x = ~Temp..Celsius, y = ~DO.mg.L,
                   z = ~Selenium, color = ~Veg, colors = colorsScale)
figTDOV <- figTDOV %>% add_markers()
figTDOV <- figTDOV %>% layout(scene = list(xaxis = list(title = 'Temp. Celsius'),
                                           yaxis = list(title = 'DO.mg.L'),
                                           zaxis = list(title = 'Selenium mg/L')))
figTDOV

#---

#Temp vs COD
figTCV <- plot_ly(dfT, x = ~Temp..Celsius, y = ~COD,
                  z = ~Selenium, color = ~Veg, colors = colorsScale)
figTCV <- figTCV %>% add_markers()
figTCV <- figTCV %>% layout(scene = list(xaxis = list(title = 'Temp. Celsius'),
                                         yaxis = list(title = 'COD mg/L'),
                                         zaxis = list(title = 'Selenium mg/L')))
figTCV
```

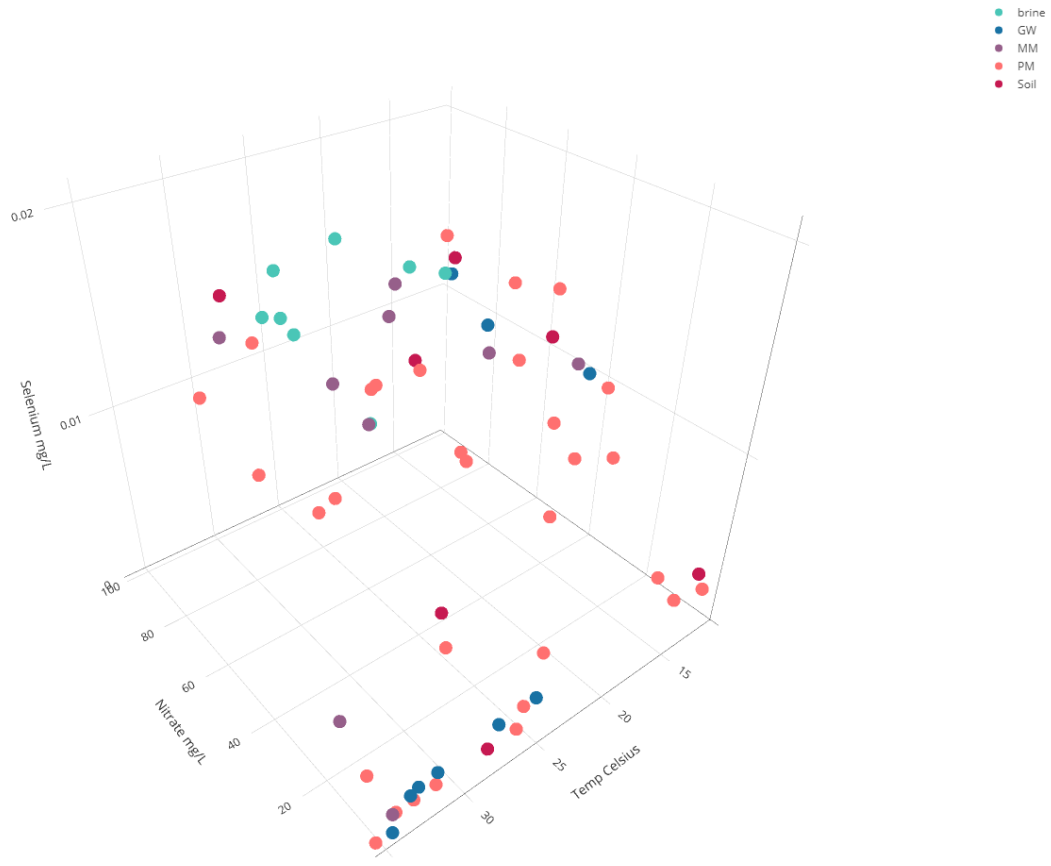

brine
VegType_A
VegType_B
VegType_C
0.02
0.01
Selenium mg/L
180
80
60
40
20
0
Nitrate mg/L
35
30
25
20
15
Temp Celsius

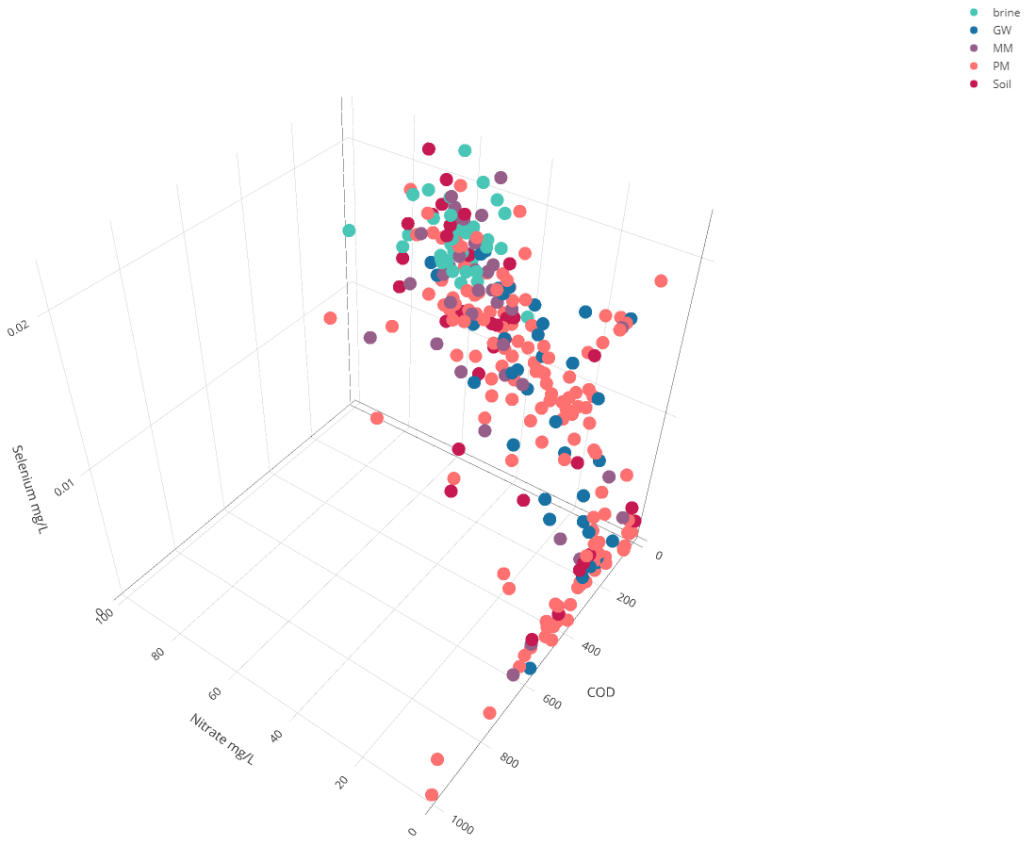

brine
VegType_A
VegType_B
VegType_C
Selenium mg/L
0.03
0.02
0.01
DO.mg.L
∞
7
6
5
4
3
2
1
0
Temp.celsius
35
30
25
20
15



When high Temperature is coupled with low Nitrate or DO, Selenium tends to be low. But COD doesn't have a clear correlation with Selenium level.

```
#Temp vs Nit on Media
figM <- plot_ly(dfT, x = ~Nitrate, y = ~Temp..Celsius,
                z = ~Selenium, color = ~MediaType, colors = colorsScale)
figM <- figM %>% add_markers()
figM <- figM %>% layout(scene = list(xaxis = list(title = 'Nitrate mg/L'),
                                     yaxis = list(title = 'Temp Celsius'),
                                     zaxis = list(title = 'Selenium mg/L')))
figM

#---

#Nit vs COD on Media
figNCM <- plot_ly(dfT, x = ~Nitrate, y = ~COD,
                  z = ~Selenium, color = ~MediaType, colors = colorsScale)
figNCM <- figNCM %>% add_markers()
figNCM <- figNCM %>% layout(scene = list(xaxis = list(title = 'Nitrate mg/L'),
                                         yaxis = list(title = 'COD'),
                                         zaxis = list(title = 'Selenium mg/L')))
figNCM
```


brine
GW
MM
PM
Soil
0.02
0.01
Selenium mg/L
100
80
60
40
20
Nitrate mg/L
15
20
25
30
Temp Celsius



In general, Media Type seems to be affected by variables in a similar way to Vegetation except for Soil type. Soil Type Media is more resistant to changes in the environment than other Media Types.

```
#Diff Nit and Temp on Veg
figNTVD <- plot_ly(dfD, x = ~Nitrate, y = ~Temp..Celsius,
                   z = ~diff_Selenium, color = ~Veg, colors = colorsScale)
figNTVD <- figNTVD %>% add_markers()
figNTVD <- figNTVD %>% layout(scene = list(xaxis = list(title = 'Nitrate mg/L'),
                                      yaxis = list(title = 'Temp Celsius'),
                                      zaxis = list(title = 'Difference Selenium mg/L')))
figNTVD

#---

#Diff Nit and Temp on Media
figNTMD <- plot_ly(dfD, x = ~Nitrate, y = ~Temp..Celsius,
                   z = ~diff_Selenium, color = ~MediaType, colors = colorsScale)
figNTMD <- figNTMD %>% add_markers()
figNTMD <- figNTMD %>% layout(scene = list(xaxis = list(title = 'Nitrate mg/L'),
                                         yaxis = list(title = 'Temp Celsius'),
                                         zaxis = list(title = 'Difference Selenium mg/L')))
figNTMD

#---
```

```
#Diff Nit vs COD on Media
figNCMD <- plot_ly(dfD, x = ~Nitrate, y = ~COD,
                   z = ~diff_Selenium, color = ~MediaType, colors = colorsScale)
figNCMD <- figNCMD %>% add_markers()
figNCMD <- figNCMD %>% layout(scene = list(xaxis = list(title = 'Nitrate mg/L'),
                                           yaxis = list(title = 'COD mg/L'),
                                           zaxis = list(title = 'Difference Selenium mg/L')))
figNCMD
```

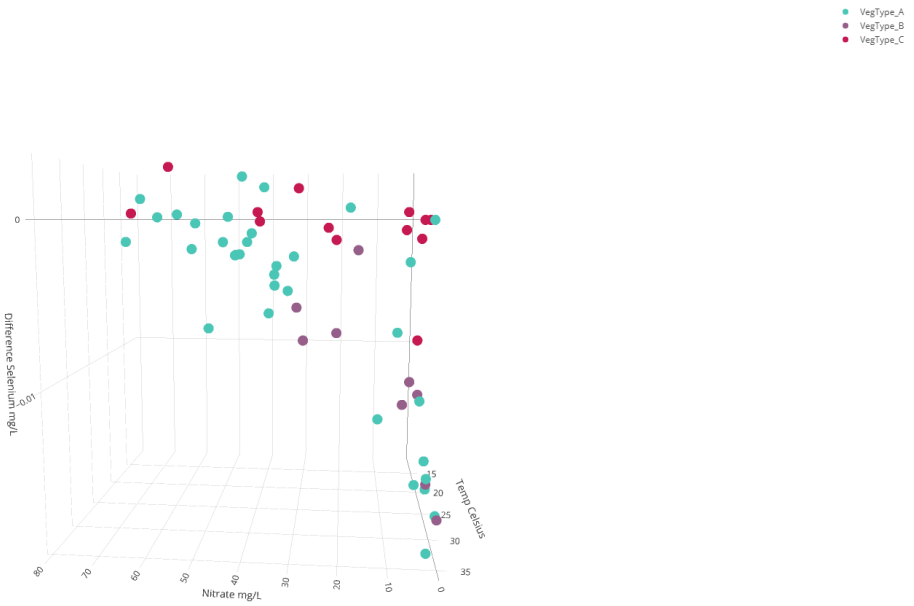


Figure 1: Diff Selenium Nit vs Temp on Veg

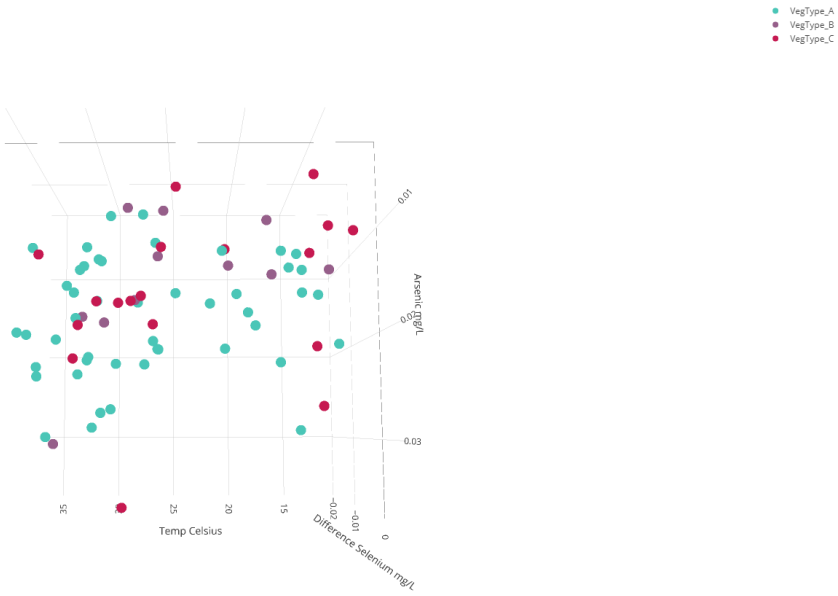
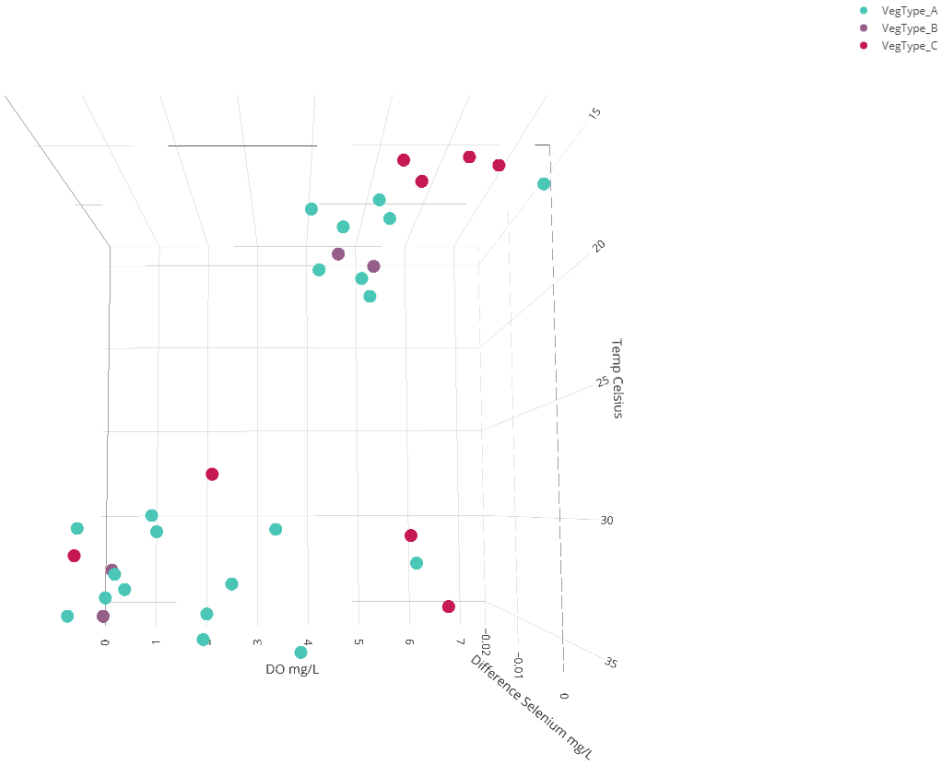


Figure 2: Diff Selenium Nit vs Temp on Media



The change in Selenium level when comparing Temp and Nitrate is consistent with our findings above. The next figure shows that Arsenic doesn't have a relationship with Temp. However, Arsenic, similar to Nitrate, positively correlates with Selenium. Lower Arsenic at higher Temp generally results in more Selenium reduction. And finally, DO negatively correlates with Temp that higher Temp is associated with lower DO. Low DO by itself isn't as strong a predictor of Selenium reduction, but when coupled with high Temp, the relationship is stronger.

2.4 Best Subset Regression

```
GetLeapTable <- function(leapSummaryIn){
    result <- cbind(leapSummaryIn$adjr2, leapSummaryIn$cp, leapSummaryIn$bic)
    return(result)
}

GetMinMax <- function(leapSummaryIn){
    result <- data.frame(
        Adj.R2 = which.max(leapSummaryIn$adjr2),
        CP = which.min(leapSummaryIn$cp),
        BIC = which.min(leapSummaryIn$bic)
    )
    return(result)
}

#--- using long data with fewer variables ---

#using dfCLong as dataset

leapsResultL <- regsubsets(Selenium ~ Nitrate + COD + Phosphorus + Arsenic +
                               Veg + MediaType,
                           data = dfCLong, nvmax = 5)
# view results
leapSummaryL <- summary(leapsResultL)
leapTableL <- GetLeapTable(leapSummaryL)
minMaxLeapL <- GetMinMax(leapSummaryL)

minMaxLeapL
```

```
##   Adj.R2 CP BIC
## 1      5  5   3
```

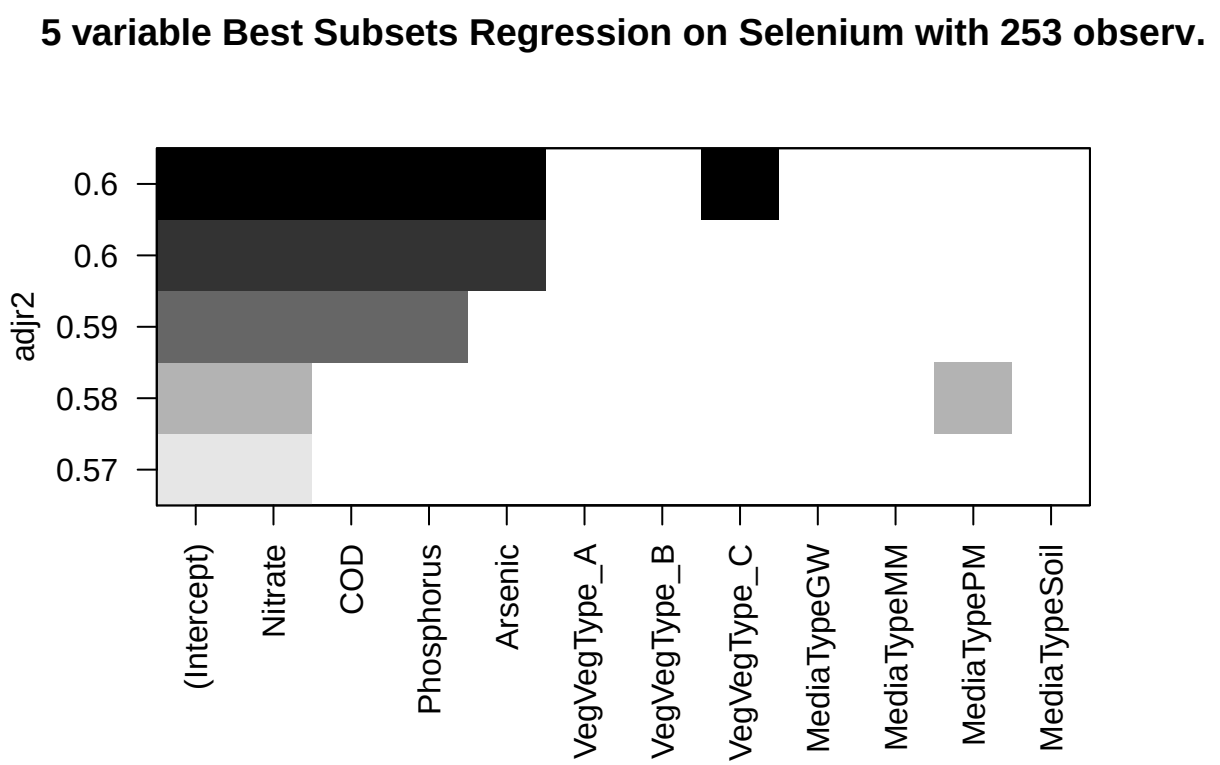
```
#minMaxLeap result
#max adj.r2     min cp      min bic
#5              5           3
#---
leapTableL
```

```
##           [,1]      [,2]      [,3]
## [1,] 0.5706988 21.488922 -203.8751
## [2,] 0.5798578 16.663516 -204.8078
## [3,] 0.5886022 12.143186 -205.6097
## [4,] 0.5958105  8.623054 -205.5666
## [5,] 0.6022738  5.601003 -205.1338
```

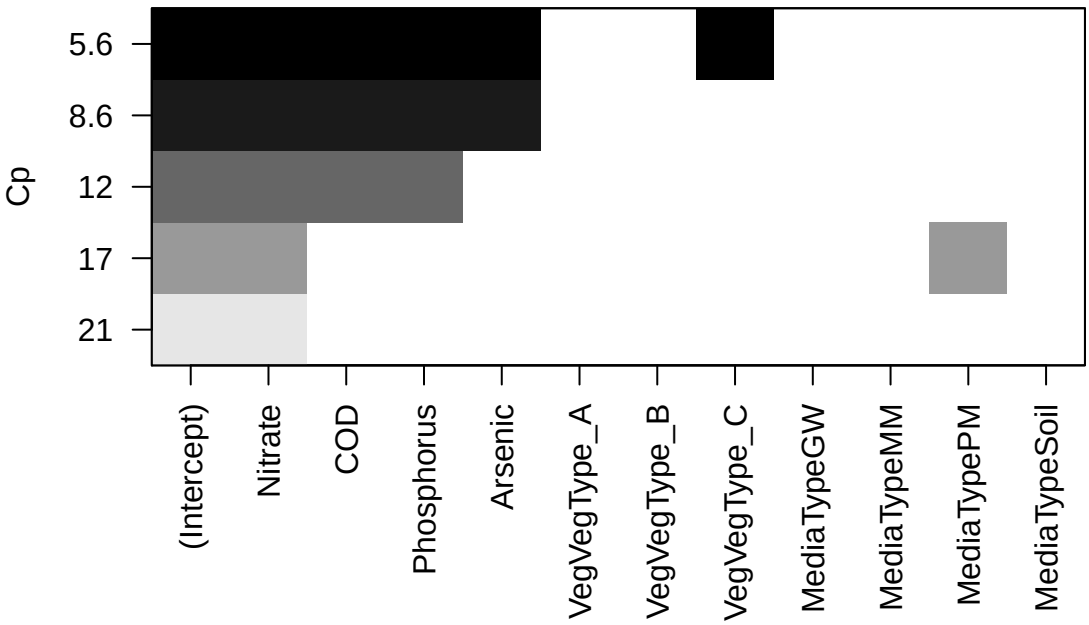
```
# [3,] 0.589 12.143 -205.610
# [5,] 0.602  5.601 -205.134
#---
leapSummaryL
```

```
## Subset selection object
## Call: regsubsets.formula(Selenium ~ Nitrate + COD + Phosphorus + Arsenic +
##     Veg + MediaType, data = dfCLong, nvmax = 5)
## 11 Variables  (and intercept)
##               Forced in Forced out
## Nitrate           FALSE      FALSE
## COD               FALSE      FALSE
## Phosphorus        FALSE      FALSE
## Arsenic           FALSE      FALSE
## VegVegType_A      FALSE      FALSE
## VegVegType_B      FALSE      FALSE
## VegVegType_C      FALSE      FALSE
## MediaTypeGW       FALSE      FALSE
## MediaTypeMM       FALSE      FALSE
## MediaTypePM       FALSE      FALSE
## MediaTypeSoil     FALSE      FALSE
## 1 subsets of each size up to 5
## Selection Algorithm: exhaustive
##          Nitrate COD Phosphorus Arsenic VegVegType_A VegVegType_B VegVegType_C
## 1  ( 1 ) "*"     " " " "        " "     " "          " "          " "
## 2  ( 1 ) "*"     " " " "        " "     " "          " "          " "
## 3  ( 1 ) "*"     "*" "*"        " "     " "          " "          " "
## 4  ( 1 ) "*"     "*" "*"        "*"     " "          " "          " "
## 5  ( 1 ) "*"     "*" "*"        "*"     " "          " "          "*"
##          MediaTypeGW MediaTypeMM MediaTypePM MediaTypeSoil
## 1  ( 1 ) " "         " "         " "         " "
## 2  ( 1 ) " "         " "         "*"         " "
## 3  ( 1 ) " "         " "         " "         " "
## 4  ( 1 ) " "         " "         " "         " "
## 5  ( 1 ) " "         " "         " "         " "
```

```
#[3,] Nit, COD, Phosphorus
#[5,] Nit, COD, Phosphorus, Arsenic, Veg Type C

# plot a table of models showing variables in each model.
# models are ordered by the selection statistic.
plot(leapsResultL, scale = "adjr2",
     main = "5 variable Best Subsets Regression on Selenium with 253 observ.")
```




```
plot(leapsResultL, scale = "Cp",
     main = "5 variable Best Subsets Regression on Selenium with 253 observ.")
```




```
dfCLong$VegTypeCTrue <- dfCLong$Veg == "VegType_C"
FinalSubsetsModel <- lm(Selenium ~ Nitrate + COD + Phosphorus + Arsenic + VegTypeCTrue, data = dfCLong)
summary(FinalSubsetsModel)
```

```
##
## Call:
## lm(formula = Selenium ~ Nitrate + COD + Phosphorus + Arsenic +
##     VegTypeCTrue, data = dfCLong)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0094384 -0.0022832 -0.0001965  0.0016127  0.0146796
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       4.449e-03  7.320e-04   6.078 4.58e-09 ***
## Nitrate           1.890e-04  1.339e-05  14.116  < 2e-16 ***
## COD               6.007e-06  1.973e-06   3.044 0.002586 **
## Phosphorus       -1.561e-03  4.179e-04  -3.736 0.000232 ***
## Arsenic           4.078e-02  1.731e-02   2.356 0.019261 *
## VegTypeCTrueTRUE -1.357e-03  6.051e-04  -2.243 0.025797 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.003845 on 247 degrees of freedom
## Multiple R-squared:  0.6102, Adjusted R-squared:  0.6023
```

```
## F-statistic: 77.32 on 5 and 247 DF,  p-value: < 2.2e-16
```

```
Best subset regression was used on the data to get the best variable model for predicting which variable
```

The procedure tests every combination of possible predictors and gives suggestions on subsets based on certain criteria. In particular, adjusted R2 and Mallow's Criterion. One of the primary problems in creating models with this data was a lack of complete observations. The interesting dynamic with this project was finding a comfortable balance between testing all the predictors or having enough observations to get more accurate models. Field data was excluded from the model selection process since in most cases it reduced the degrees of freedom to 18, resulting in over-fit models. The most successful model contained the following predictor variables: Nitrate Content, COD, Phosphorus Content, Arsenic Content, and a dummy variable showing whether or not the Vegetation Type from the Bin held Vegetation Type C. This subset of predictors was then mapped onto Selenium and had the highest possible Adjusted R2, meaning this model accounted for the most variability of the data when compared to other models that best subsets tested. One other interesting result from this test was the impact of the categorical variables for Vegetation type and Media type. Two separate subsets procedures were run that only tested the impact of vegetation type and media type individually. Results showed that Vegetation Type C has the most significant impact on Selenium for Vegetation and Peat Moss has the most significant impact on Selenium for Media. When both Vegetation and Media are included in the procedure Vegetation Type C plays a bigger role in predicting Selenium than Peat Moss does as seen in Figure 4.