7: Lab - Generalized Linear Models

Gretchen Barbera

Fall 2024

Objectives

1. Answer questions on M5/A5
2. Answer questions on M6 - GLMs
3. Practice more application GLM to real datasets

Set up

```
install.packages("agricolae")
library(tidyverse)
library(agricolae)
library(here)
here()
```

```
## [1] "/home/guest/EDE_Fall2024"
```

```
EPAair <- read.csv(here(
  "Data/Processed_KEY/EPAair_O3_PM25_NC1819_Processed.csv"),
  stringsAsFactors = TRUE)
# Set date to date format
EPAair$Date <- as.Date(EPAair$Date, format = "%Y-%m-%d")

Litter <- read.csv(here(
  "Data/Processed_KEY/NEON_NIWO_Litter_mass_trap_Processed.csv"),
  stringsAsFactors = TRUE)
# Set date to date format
Litter$collectDate <- as.Date(Litter$collectDate , format = "%Y-%m-%d")

# Set theme
mytheme <- theme_classic(base_size = 14) +
  theme(axis.text = element_text(color = "black"),
        legend.position = "top")
theme_set(mytheme)
```

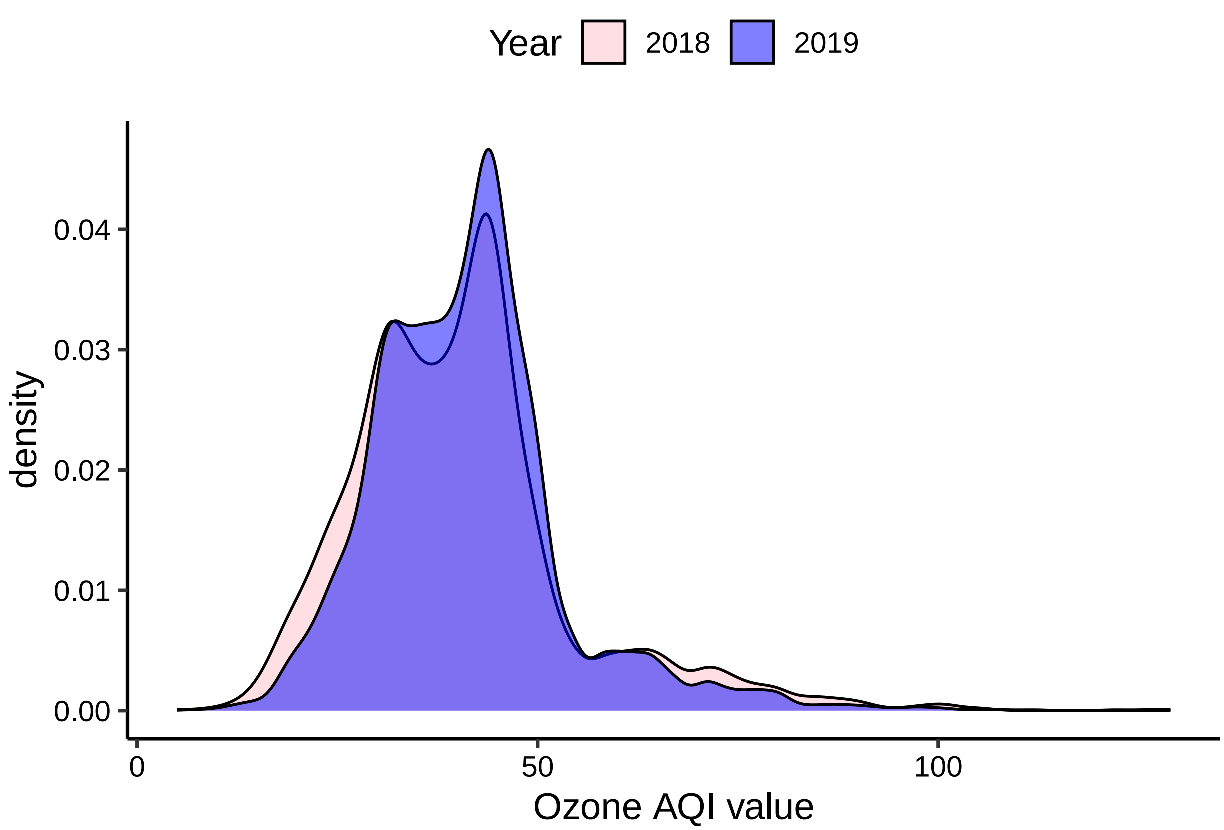
Visualization and interpretation challenge

Create three plots, each with appropriately formatted axes and legends. Choose a non-default color palette.

1. geom_density of ozone divided by year (distinguish between years by adding transparency to the geom_density layer).
2. geom_boxplot of ozone divided by year. Add letters representing a significant difference between 2018 and 2019 (hint: stat_summary).
3. geom_violin of ozone divided by year, with the 0.5 quantile marked as a horizontal line. Add letters representing a significant difference between 2018 and 2019.

```
#Exercise 1:

#1

Ozone.density <- ggplot(EPAair, aes(x = Ozone,
                                  fill= as.factor(Year)))  +
  geom_density(alpha=0.5) +
 scale_fill_manual(values = c("pink", "blue")) +
  labs(x= "Ozone AQI value",
       y= "density",
       fill="Year")+
  mytheme
print(Ozone.density)
```

```
## Warning: Removed 2146 rows containing non-finite outside the scale range
## (`stat_density()`).
```

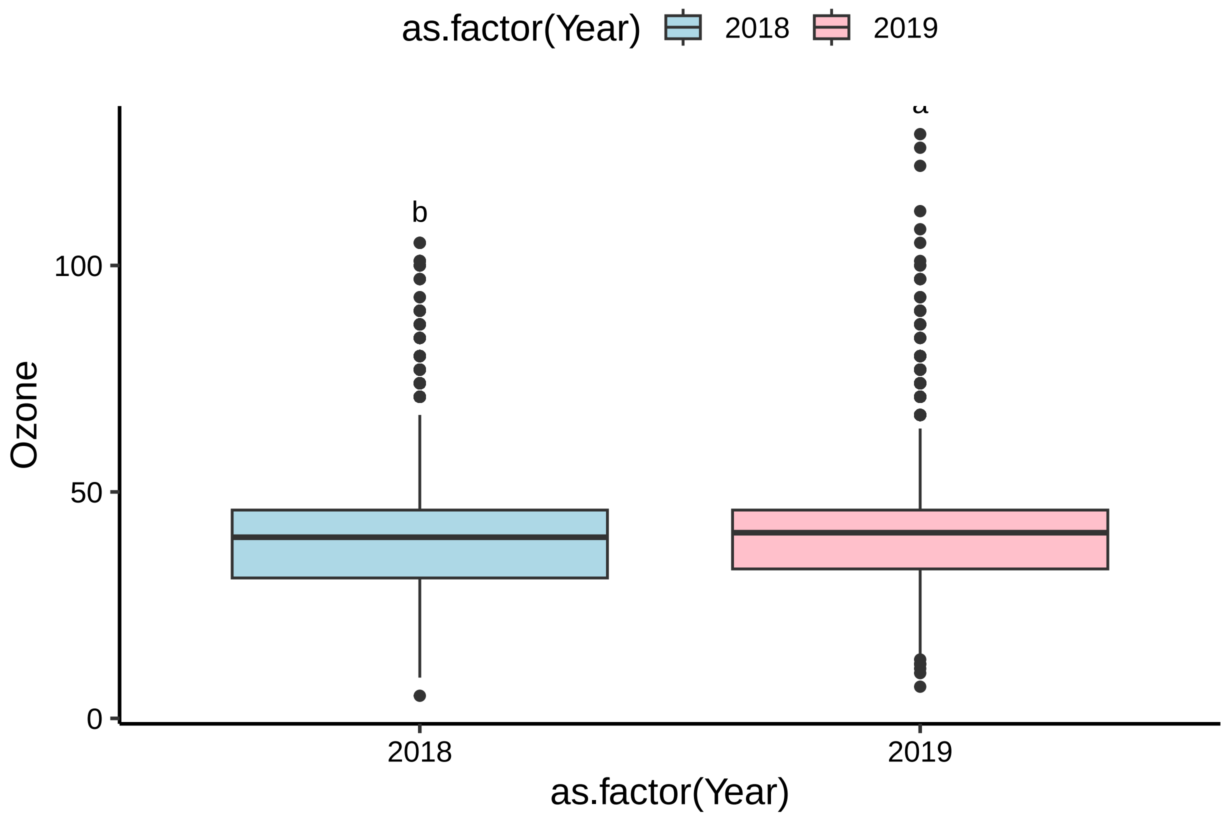


```r
#2.

O3.boxplot <-
  ggplot(EPAair, aes(x=as.factor(Year),
                     y=Ozone))+
  geom_boxplot(aes(fill=as.factor(Year)))+
  stat_summary(geom = "text",
               fun = max,
               vjust= -1,
               size=4,
               label=c("b","a"))+
  scale_fill_manual(values = c("2018"= "lightblue", "2019"= "pink"))
  labs(x="",
       y="Ozone, AQI Value") +
  ylim(0,150)+
  mytheme
```

```
## NULL
```

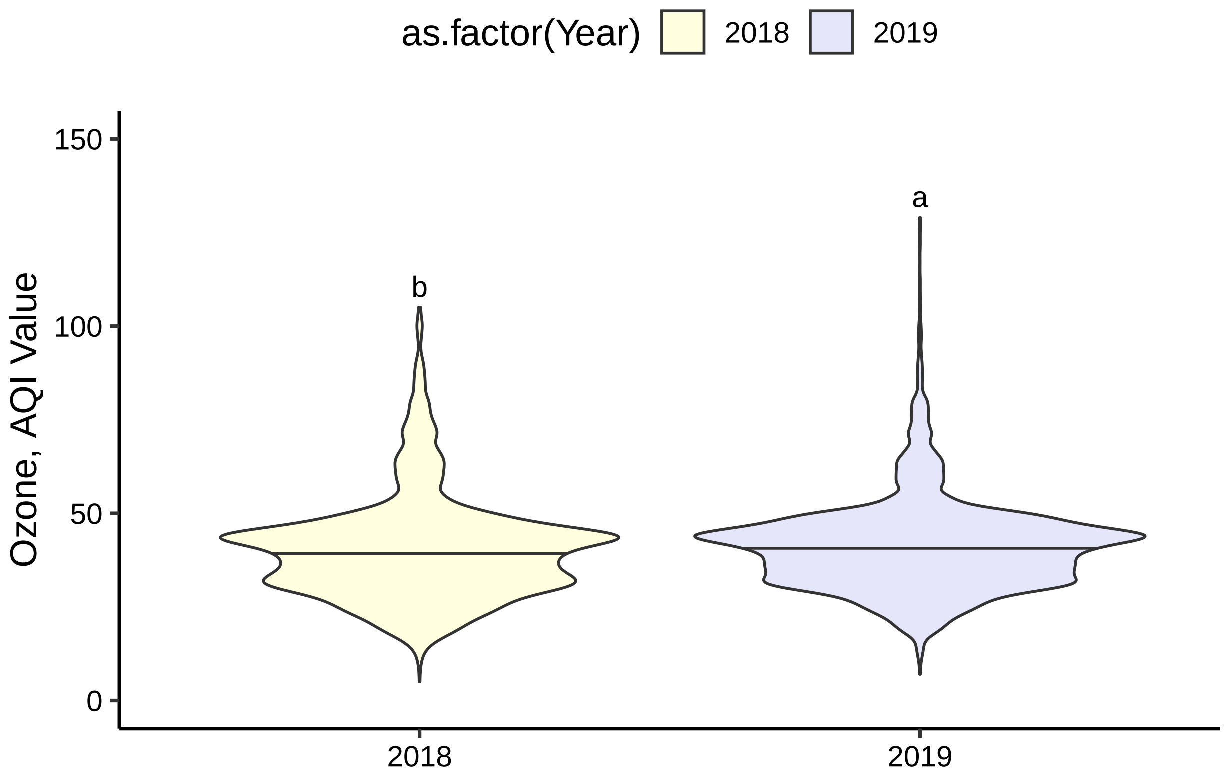
```r
print(O3.boxplot)
```

```
## Warning: Removed 2146 rows containing non-finite outside the scale range
## (`stat_boxplot()`).

## Warning: Removed 2146 rows containing non-finite outside the scale range
## (`stat_summary()`).
```




```
#3

o3.violin <-
  ggplot(EPAair, aes(x=as.factor(Year),
                     y=Ozone, fill = as.factor(Year)))+
  geom_violin(draw_quantiles =c(0.5))+
  stat_summary(geom = "text",
               fun = max,
               vjust=-0.5,
               size=4,
               label=c("b","a"))+
   scale_fill_manual(values = c("2018"= "lightyellow", "2019"= "lavender")) +
  labs(x="",
       y="Ozone, AQI Value") +
  ylim(0,150)

print(o3.violin)
```

```
## Warning: Removed 2146 rows containing non-finite outside the scale range
## (`stat_ydensity()`).
## Removed 2146 rows containing non-finite outside the scale range
## (`stat_summary()`).
```




```
#i adjusted the vjust so i was able to see the a and b values
```

Linear Regression

Important components of the linear regression are the correlation and the R-squared value. The **correlation** is a number between -1 and 1, describing the relationship between the variables. Correlations close to -1 represent strong negative correlations, correlations close to zero represent weak correlations, and correlations close to 1 represent strong positive correlations. The **R-squared value** is the correlation squared, becoming a number between 0 and 1. The R-squared value describes the percent of variance accounted for by the explanatory variables.

For the NTL-LTER dataset, can we predict PM2.5 from Ozone?

```
#Exercise 2: Run a linear regression PM2.5 by Ozone.
#Find the p-value and R-squared value.


PM2.5_ozone <-
  lm(EPAair$PM2.5~EPAair$Ozone)

PM2.5_Ozone <- lm(PM2.5~Ozone, data = EPAair)
summary(PM2.5_Ozone)
```

```
##
## Call:
```

```
## lm(formula = PM2.5 ~ Ozone, data = EPAair)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -37.204  -8.931  -0.613   8.463  48.473
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 15.63824    0.55556   28.15   <2e-16 ***
## Ozone        0.38384    0.01298   29.58   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.06 on 5774 degrees of freedom
##   (3200 observations deleted due to missingness)
## Multiple R-squared:  0.1316, Adjusted R-squared:  0.1314
## F-statistic: 874.9 on 1 and 5774 DF,  p-value: < 2.2e-16
```

```r
summary(PM2.5_ozone)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$Ozone)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -37.204  -8.931  -0.613   8.463  48.473
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  15.63824    0.55556   28.15   <2e-16 ***
## EPAair$Ozone  0.38384    0.01298   29.58   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.06 on 5774 degrees of freedom
##   (3200 observations deleted due to missingness)
## Multiple R-squared:  0.1316, Adjusted R-squared:  0.1314
## F-statistic: 874.9 on 1 and 5774 DF,  p-value: < 2.2e-16
```

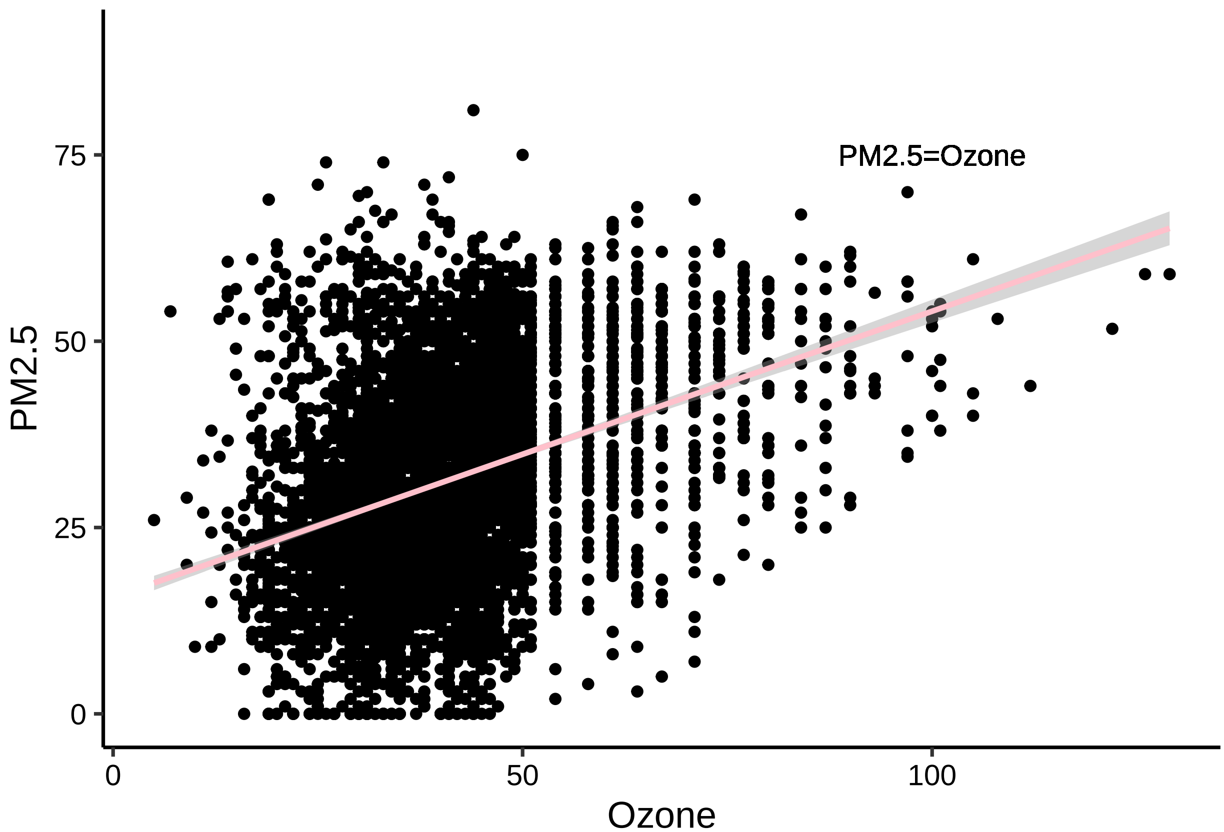
```r
print(PM2.5_ozone)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$Ozone)
##
## Coefficients:
##  (Intercept)  EPAair$Ozone
##      15.6382        0.3838
```

```r
#p-value = less than 0.05 so it is significant
#r-squared value = 0.1314 which means 13% of the variability
```

```r
#in the dependent variable
#indicates a relatively weak connection between PM2.5 and the Ozone

#Exercise 3: Build a scatterplot. Add a line and standard error
#for the linear regression. Add the regression equation to the plot

model <- lm(PM2.5 ~ Ozone, data = EPAair)


PM2.5_ozone.plot <-  ggplot(EPAair, aes(x=Ozone,
                   y= PM2.5)) +
  geom_point()+
  geom_smooth(method ="lm",col= "pink", se=TRUE)+
  geom_text(x=100,
            y= 75,
            label= expression("PM2.5=Ozone"))

print(PM2.5_ozone.plot)
```

```
## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 3200 rows containing non-finite outside the scale range
## (`stat_smooth()`).

## Warning: Removed 3200 rows containing missing values or values outside the scale range
## (`geom_point()`).

## Warning in is.na(x): is.na() applied to non-(list or vector) of type
## 'expression'
```




```
sum(is.na(EPAair$Ozone))
```

```
## [1] 2146
```

```
sum(is.na(EPAair$PM2.5))
```

```
## [1] 1054
```

```
#I got a warning message about my NA and infinite numbers
#I looked at them and they didn't impact my scatterplot as much
#so I will keep them in
#the se= FALSE got rid of the confidence interval around my regression
```

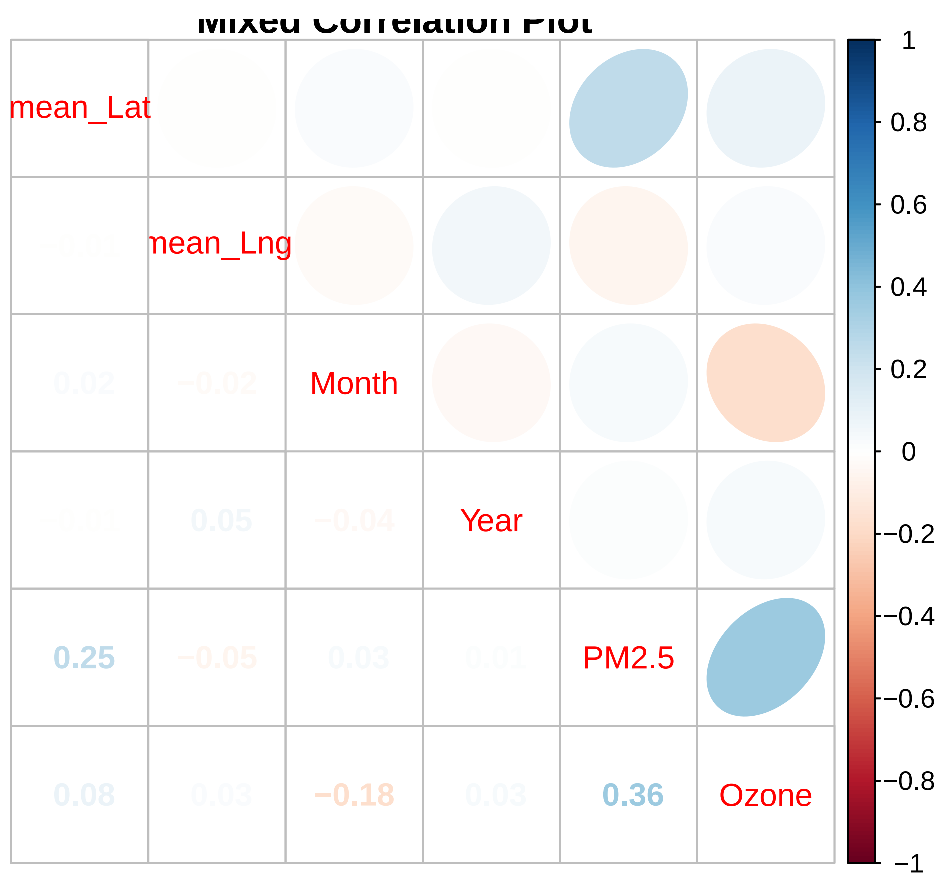
AIC to select variables

What other variables can we add to improve model?

```
#Exercise 4: Build correlation plots and identify more
#possible explanatory variables to add to the regression.

library(corrplot)
```

```
## corrplot 0.95 loaded
```

```r
library(dplyr)
library(ggplot2)


EPAair.subset <- EPAair %>%
  select(mean_Lat:Ozone) %>%
  na.omit()
#looking at the variables mean_Lat, mean_Lng, Month, Year, PM2.5, and Ozone


EPAair.corr <- cor(EPAair.subset)
corrplot.mixed(EPAair.corr, upper= "ellipse",
               title= "Mixed Correlation Plot")
```




```r
#based on the correlation plot, more explanatory variable would be
#PM2.5 and mean_Lat because they have a decently dark ellipse



#Exercise 5: Choose a model by AIC in a Stepwise Algorithm.
#Do the results from AIC match the variables you selected on Exercise 4?



OzoneAll.reg <- lm(data = EPAair, PM2.5 ~ Ozone +
```

```r
                      mean_Lat + mean_Lng + Month + Year)
print(OzoneAll.reg)
```

```
##
## Call:
## lm(formula = PM2.5 ~ Ozone + mean_Lat + mean_Lng + Month + Year,
##     data = EPAair)
##
## Coefficients:
## (Intercept)        Ozone     mean_Lat     mean_Lng        Month         Year
##   -909.9344       0.3823       6.5242      -0.5006       0.4660       0.3221
```

```r
#shows that ozone pollution levels could be dependent on the mean_lat
#and the month
#this shows that ozone and PM2.5 have a weaker correlation than
#PM2.5 and the mean_lat do

#Exercise 6: Run another regression using the variables selected on Exercise 6.
#Compare r-squared value with the one from Exercise 2.

PM2.5_Lat <-
  lm(EPAair$PM2.5~EPAair$mean_Lat)

summary(PM2.5_Lat)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$mean_Lat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.018 -10.568  -1.138   9.539  57.036
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -248.2368    11.6823  -21.25   <2e-16 ***
## EPAair$mean_Lat    7.8055     0.3274   23.84   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.08 on 7920 degrees of freedom
##   (1054 observations deleted due to missingness)
## Multiple R-squared:  0.06698,    Adjusted R-squared:  0.06686
## F-statistic: 568.6 on 1 and 7920 DF,  p-value: < 2.2e-16
```

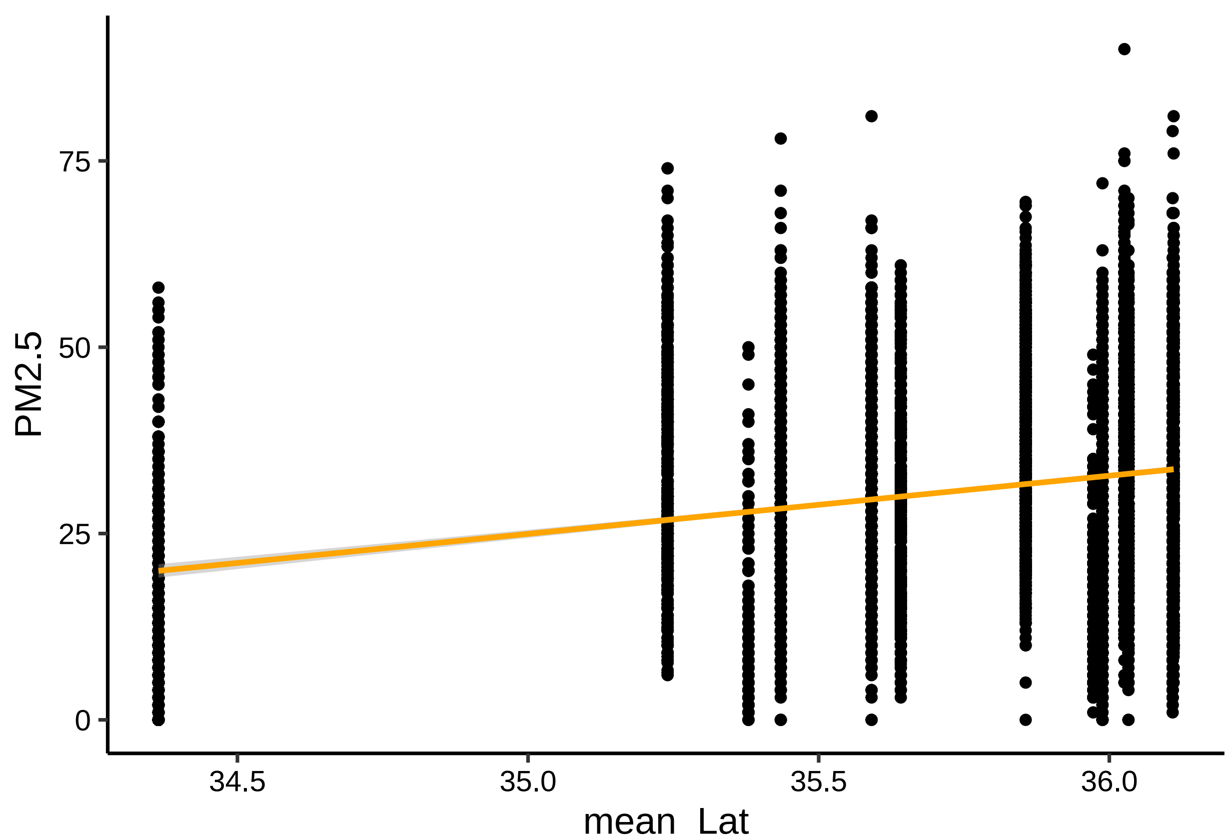
```r
print(PM2.5_Lat)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$mean_Lat)
##
## Coefficients:
```

```
##      (Intercept)  EPAair$mean_Lat
##        -248.237            7.805
```

```
PM2.5_Lat.plot <- ggplot(EPAair, aes(
  x= mean_Lat,
  y= PM2.5
)) +
  geom_point()+
  geom_smooth(method = "lm", col="orange", se=TRUE) +
  geom_text(x=100,
            y=75,
            label=expression("PM2.5=mean_Lat"))

print(PM2.5_Lat.plot)
```

```
## `geom_smooth()` using formula = 'y ~ x'

## Warning: Removed 1054 rows containing non-finite outside the scale range
## (`stat_smooth()`).

## Warning: Removed 1054 rows containing missing values or values outside the scale range
## (`geom_point()`).

## Warning in is.na(x): is.na() applied to non-(list or vector) of type
## 'expression'
```



```r
summary(PM2.5_ozone)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$Ozone)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -37.204  -8.931  -0.613   8.463  48.473
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  15.63824    0.55556   28.15   <2e-16 ***
## EPAair$Ozone  0.38384    0.01298   29.58   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 13.06 on 5774 degrees of freedom
##   (3200 observations deleted due to missingness)
## Multiple R-squared:  0.1316, Adjusted R-squared:  0.1314
## F-statistic: 874.9 on 1 and 5774 DF,  p-value: < 2.2e-16
```

```r
summary(PM2.5_Lat)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$mean_Lat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.018 -10.568  -1.138   9.539  57.036
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -248.2368    11.6823  -21.25   <2e-16 ***
## EPAair$mean_Lat    7.8055     0.3274   23.84   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.08 on 7920 degrees of freedom
##   (1054 observations deleted due to missingness)
## Multiple R-squared:  0.06698,    Adjusted R-squared:  0.06686
## F-statistic: 568.6 on 1 and 7920 DF,  p-value: < 2.2e-16
```

```r
#looking that the summaries of the datasets I made based on the PM2.5
#and ozone and the PM2.5 and the mean_Lat
#from the summary, I can see that they have p values smaller than
#0.05 which means they are statistically significant meaning there is a
#strong association between the PM2.5 levels and ozone/mean_Lat respectively
#mean_lat has a smaller r squared value which means that it does not
#explain much of the variability in PM2.5 which means there are other
#factors that have a more significant influence on the PM2.5 levels collected
```

```r
print(PM2.5_ozone)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$Ozone)
##
## Coefficients:
##  (Intercept)  EPAair$Ozone
##      15.6382        0.3838
```

```r
summary(PM2.5_Lat)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$mean_Lat)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.018 -10.568  -1.138   9.539  57.036
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## (Intercept)     -248.2368    11.6823  -21.25   <2e-16 ***
## EPAair$mean_Lat    7.8055     0.3274   23.84   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.08 on 7920 degrees of freedom
##   (1054 observations deleted due to missingness)
## Multiple R-squared:  0.06698,    Adjusted R-squared:  0.06686
## F-statistic: 568.6 on 1 and 7920 DF,  p-value: < 2.2e-16
```

```r
print(PM2.5_Lat)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$mean_Lat)
##
## Coefficients:
##     (Intercept)  EPAair$mean_Lat
##        -248.237            7.805
```

```r
print(PM2.5_ozone)
```


```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$Ozone)
##
## Coefficients:
##  (Intercept)  EPAair$Ozone
##      15.6382        0.3838
```

```r
#I am going to try and see if month is a better variable


PM2.5_month <-
  lm(EPAair$PM2.5~EPAair$Month)

summary(PM2.5_month)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$Month)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.77 -10.66  -1.17  10.08  59.23
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  29.74112    0.34520  86.157   <2e-16 ***
## EPAair$Month  0.08580    0.04699   1.826   0.0679 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.57 on 7920 degrees of freedom
##   (1054 observations deleted due to missingness)
## Multiple R-squared:  0.0004209,  Adjusted R-squared:  0.0002947
## F-statistic: 3.335 on 1 and 7920 DF,  p-value: 0.06786
```

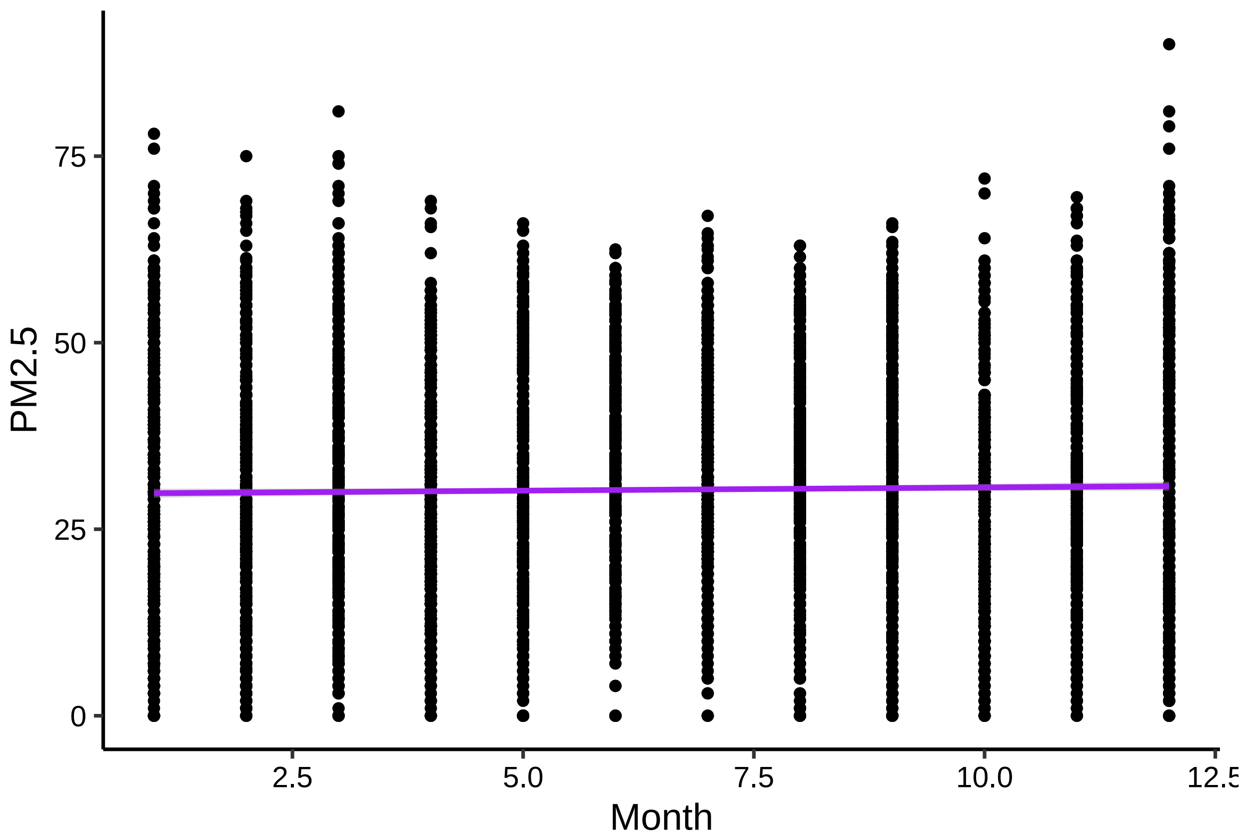
```r
print(PM2.5_month)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$Month)
##
## Coefficients:
##  (Intercept)  EPAair$Month
##      29.7411        0.0858
```

```r
PM2.5_Month.plot <- ggplot(EPAair, aes(
  x= Month,
  y= PM2.5
)) +
  geom_point()+
  geom_smooth(method = "lm", col="purple", se=TRUE) +
  geom_text(x=100,
            y=75,
            label=expression("PM2.5=month"))

print(PM2.5_Month.plot)
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 1054 rows containing non-finite outside the scale range
## (`stat_smooth()`).

## Warning: Removed 1054 rows containing missing values or values outside the scale range
## (`geom_point()`).

## Warning in is.na(x): is.na() applied to non-(list or vector) of type
## 'expression'
```




```r
summary(PM2.5_month)
```

```
##
## Call:
## lm(formula = EPAair$PM2.5 ~ EPAair$Month)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -30.77 -10.66  -1.17  10.08  59.23
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  29.74112    0.34520  86.157   <2e-16 ***
## EPAair$Month  0.08580    0.04699   1.826   0.0679 .
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 14.57 on 7920 degrees of freedom
##   (1054 observations deleted due to missingness)
## Multiple R-squared:  0.0004209,  Adjusted R-squared:  0.0002947
## F-statistic: 3.335 on 1 and 7920 DF,  p-value: 0.06786
```

Litter Exercise

```
# Wrangle the data
Litter.Totals <- Litter %>%
  group_by(plotID, collectDate, nlcdClass) %>%
  summarise(dryMass = sum(dryMass))
```

```
## 'summarise()' has grouped output by 'plotID', 'collectDate'. You can override
## using the '.groups' argument.
```

```
# Format ANOVA as aov
Litter.Totals.anova <- aov(data = Litter.Totals, dryMass ~ plotID)
summary(Litter.Totals.anova)
```

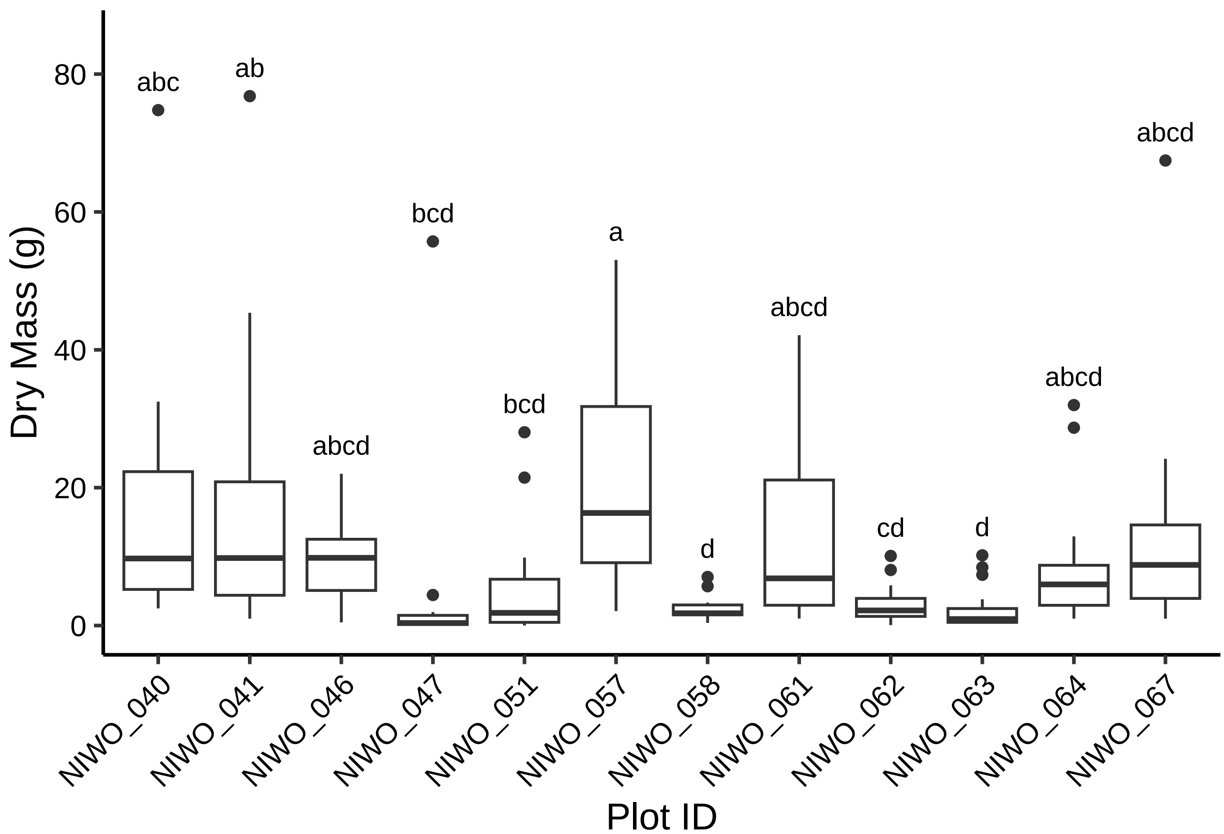
```
##              Df Sum Sq Mean Sq F value   Pr(>F)
## plotID       11   7584   689.5   4.813 1.45e-06 ***
## Residuals   198  28363   143.2
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
# Extract groupings for pairwise relationships
Litter.Totals.groups <- HSD.test(Litter.Totals.anova, "plotID", group = TRUE)
Litter.Totals.groups$groups
```

```
##            dryMass groups
## NIWO_057 20.685833      a
## NIWO_041 16.979063     ab
## NIWO_040 15.680000    abc
## NIWO_061 13.186111   abcd
## NIWO_067 12.565938   abcd
## NIWO_046  9.956176   abcd
## NIWO_064  8.015789   abcd
## NIWO_051  5.668750    bcd
## NIWO_047  4.476333    bcd
## NIWO_062  3.047632     cd
## NIWO_058  2.398421      d
## NIWO_063  2.393889      d
```

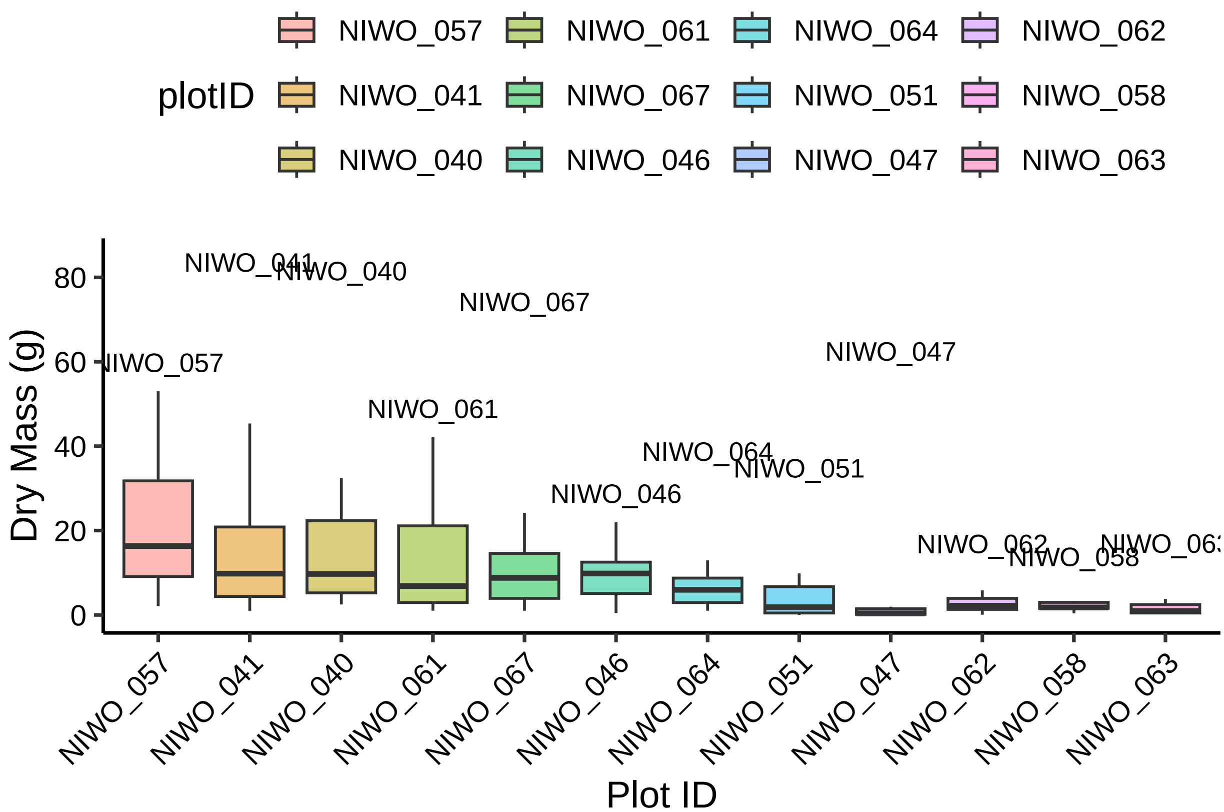
```
Litter.Totals <- Litter.Totals %>%
  mutate( treatgroups = Litter.Totals.groups$groups[plotID,2])

# Graph the results
Litter.Totals.plot <- ggplot(Litter.Totals, aes(x = plotID, y = dryMass)) +
```

```
  geom_boxplot() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  stat_summary(geom = "text", fun = max, vjust = -1, size = 3.5,
               label = c("abc", "ab", "abcd", "bcd", "bcd", "a",
                         "d", "abcd", "cd", "d", "abcd", "abcd")) +
  labs(x = "Plot ID", y = "Dry Mass (g)") +
  ylim(0, 85)
print(Litter.Totals.plot)
```




```
#Exercise 7: Improve the plot

ordered_plotID <- rownames(Litter.Totals.groups$groups)
#ordering based on the means collected from the litter data previous

Litter.Totals <- Litter.Totals %>%
  mutate(plotID = factor(plotID, levels = ordered_plotID))
#changed the plotID to a factor and ordered the plots so they would show up in
#The plot based on their means and not based on their plotID



Litter.Totals.plot <- ggplot(Litter.Totals, aes(x = plotID, y = dryMass)) +
  geom_boxplot(aes(fill = plotID), outlier.shape = NA, alpha = 0.5) +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  stat_summary(geom = "text", fun = max, vjust = -1, size = 3.5,
```

```r
              label = rownames(Litter.Totals.groups$groups)) +
  labs(x = "Plot ID", y = "Dry Mass (g)") +
  ylim(0, 85)

print(Litter.Totals.plot)
```




```r
#creating a boxplot with the reordered data so it looks nice. Now they are
#rainbow and they are in decending order based on the significant mean drymass
#in grams
```