Homework Assignment 1

Matthew Xu

23 October 2020

```
# knit from directory
setwd("/Users/MatthewXu/Downloads/algae_data/")
library(readr)

knitr::opts_knit$set(root.dir = "/Users/MatthewXu/Downloads/algae_data/")
algae <- read_table2("algaeBloom.txt", col_names = c("season", "size", "speed", "mxPH",
    "mnO2", "Cl", "NO3", "NH4", "oPO4", "PO4", "Chla", "a1", "a2", "a3", "a4", "a5",
    "a6", "a7"), na = "XXXXXXX")

attach(algae)
library(dplyr)
```

1. "PROBLEM NUMBER ONE"

a. "PROBLEM NUMBER ONE, PART A"

```
# calculate size of each season
algae %>% group_by(season) %>% summarise(n = n())
```

```
## # A tibble: 4 x 2
##   season     n
##   <chr>  <int>
## 1 autumn    40
## 2 spring    53
## 3 summer    45
## 4 winter    62
```

b. "PROBLEM NUMBER ONE, PART B"

```
# calculate meand an variance among chemicals
algae %>% summarise(across(everything(), list(mean = mean, var = var), na.rm = TRUE))
```

```
## # A tibble: 1 x 36
##   season_mean season_var size_mean size_var speed_mean speed_var mxPH_mean
##         <dbl>      <dbl>     <dbl>    <dbl>      <dbl>     <dbl>     <dbl>
## 1          NA         NA        NA       NA         NA        NA      8.01
## # ... with 29 more variables: mxPH_var <dbl>, mnO2_mean <dbl>, mnO2_var <dbl>,
## #   Cl_mean <dbl>, Cl_var <dbl>, NO3_mean <dbl>, NO3_var <dbl>, NH4_mean <dbl>,
## #   NH4_var <dbl>, oPO4_mean <dbl>, oPO4_var <dbl>, PO4_mean <dbl>,
## #   PO4_var <dbl>, Chla_mean <dbl>, Chla_var <dbl>, a1_mean <dbl>,
## #   a1_var <dbl>, a2_mean <dbl>, a2_var <dbl>, a3_mean <dbl>, a3_var <dbl>,
## #   a4_mean <dbl>, a4_var <dbl>, a5_mean <dbl>, a5_var <dbl>, a6_mean <dbl>,
## #   a6_var <dbl>, a7_mean <dbl>, a7_var <dbl>
```

Yes, by looking at glimpse(algae) there appears to be "XXXXXXX" values, indicating there are missing values. It appears that by looking at the summary statistics, the higher the mean, the higher the variance for each chemical. The mean of mnO2 is bigger than NO3 ($9.11 > 3.28$) but the variance of NO3 is larger than that of mnO2 ($14.261 > 5.71$). This is possbily due to the presence of missing values as well as frequency of missing values among the variables. The mangitudes differ based on chemicals as well as other attribues such as missing elements.

c. "PROBLEM NUMBER ONE, PART C"

```
# Median of Chemicals
print("Median of Chemicals")
```

```
## [1] "Median of Chemicals"
```

```
median(algae$mxPH, na.rm = TRUE)
```

```
## [1] 8.06
```

```
median(algae$mnO2, na.rm = TRUE)
```

```
## [1] 9.8
```

```
median(algae$Cl, na.rm = TRUE)
```

```
## [1] 32.73
```

```
median(algae$NO3, na.rm = TRUE)
```

```
## [1] 2.675
```

```r
median(algae$NH4, na.rm = TRUE)
```

```
## [1] 103.1665
```

```r
median(algae$oPO4, na.rm = TRUE)
```

```
## [1] 40.15
```

```r
median(algae$PO4, na.rm = TRUE)
```

```
## [1] 103.2855
```

```r
median(algae$Chla, na.rm = TRUE)
```

```
## [1] 5.475
```

```r
# Median Absolute Deviation of Chemicals
print("Median Absolute Deviation of Chemicals")
```

```
## [1] "Median Absolute Deviation of Chemicals"
```

```r
mad(algae$mxPH, na.rm = TRUE)
```

```
## [1] 0.504084
```

```r
mad(algae$mnO2, na.rm = TRUE)
```

```
## [1] 2.053401
```

```r
mad(algae$mnO2, na.rm = TRUE)
```

```
## [1] 2.053401
```

```r
mad(algae$Cl, na.rm = TRUE)
```

```
## [1] 33.24953
```

```r
mad(algae$NO3, na.rm = TRUE)
```

```
## [1] 2.172009
```

```r
mad(algae$NH4, na.rm = TRUE)
```

```
## [1] 111.6175
```

```r
mad(algae$oPO4, na.rm = TRUE)
```

```
## [1] 44.04582
```

```r
mad(algae$PO4, na.rm = TRUE)
```

```
## [1] 122.3212
```

```r
mad(algae$Chla, na.rm = TRUE)
```

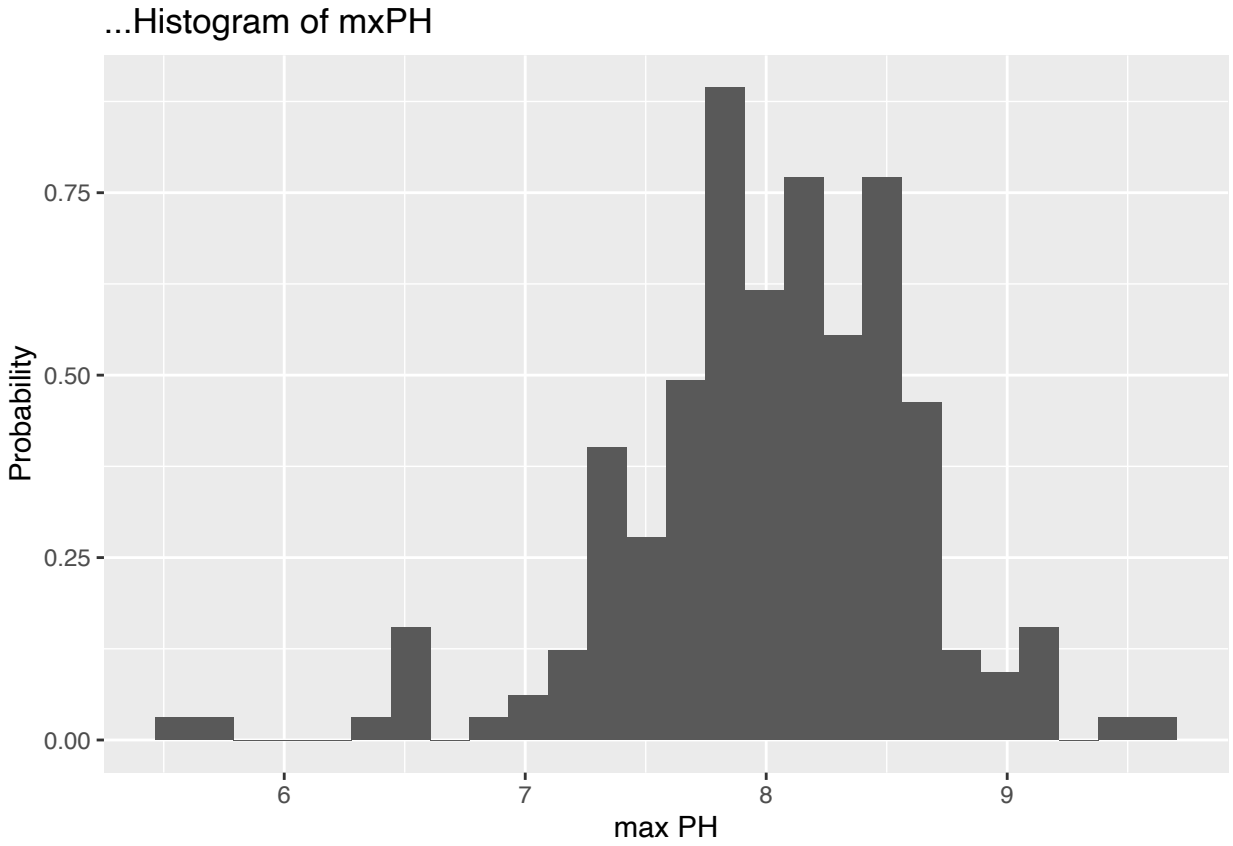
```
## [1] 6.6717
```

For chemicals mnO2, Cl, and NO3, there appears to be a higher median than MAD. For the other chemicals, the MAD is higher than the median. For PH, the median is higher than its MAD. In addition, the means and variances margins between them are much larger than that between MAD and median. The variances for certain chemcials are extremely large. This could be due to the frequency of missing values and the influences they have on error when calculating these staistics. This may also be due to the compositions of each chemical.

2. "PROBLEM NUMBER TWO"

a. "PROBLEM NUMBER TWO, PART A"

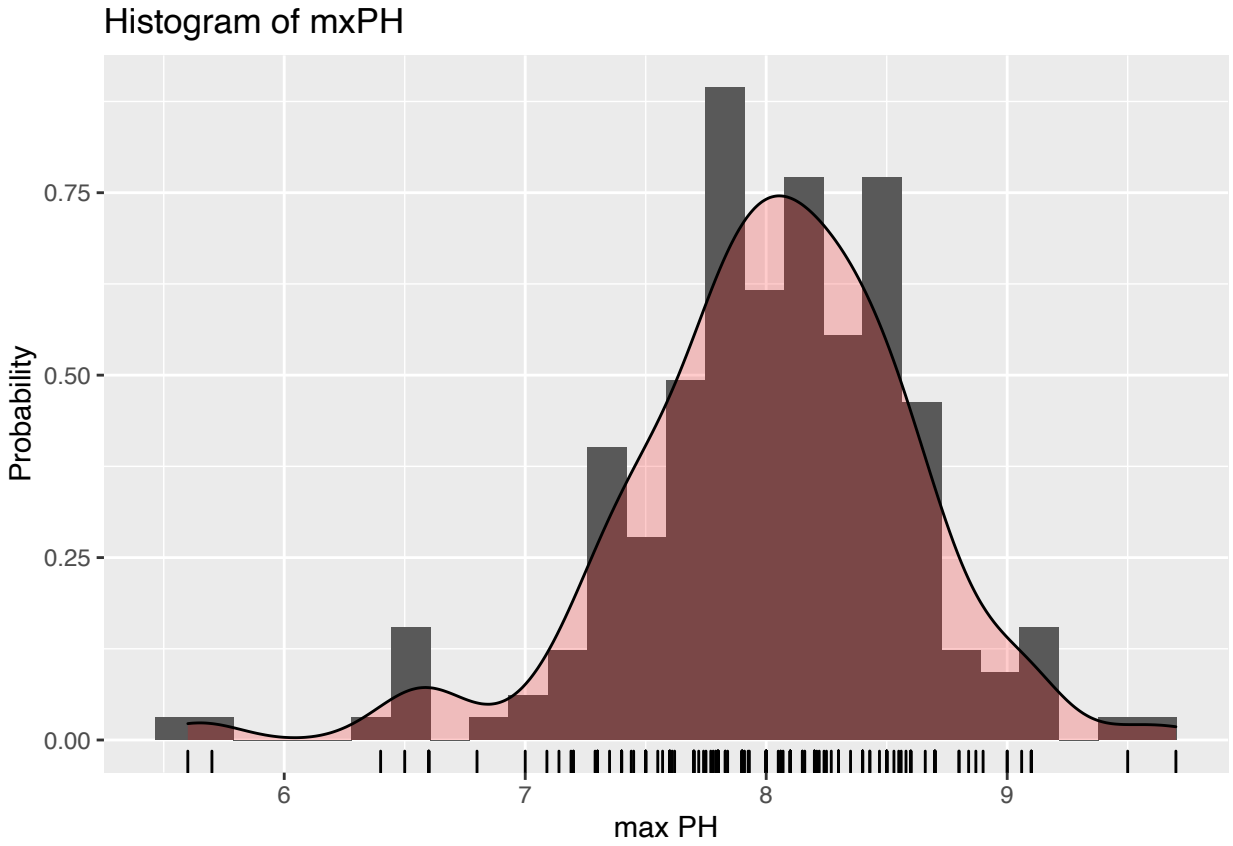
```r
library(ggplot2)
# using gg plot max PH. using algae (max ph is column 4) must elimiate NA values
# when calculating bin width
p1 <- ggplot(algae[, 4], aes(x = mxPH)) + geom_histogram(aes(y = ..density..), binwidth = densi
    4])$mxPH)$bw) + labs(title = "'Histogram of mxPH", y = "Probability", x = "max PH")
p1
```



There appears to be a of a negative skew, but also has symmetric properties as well.

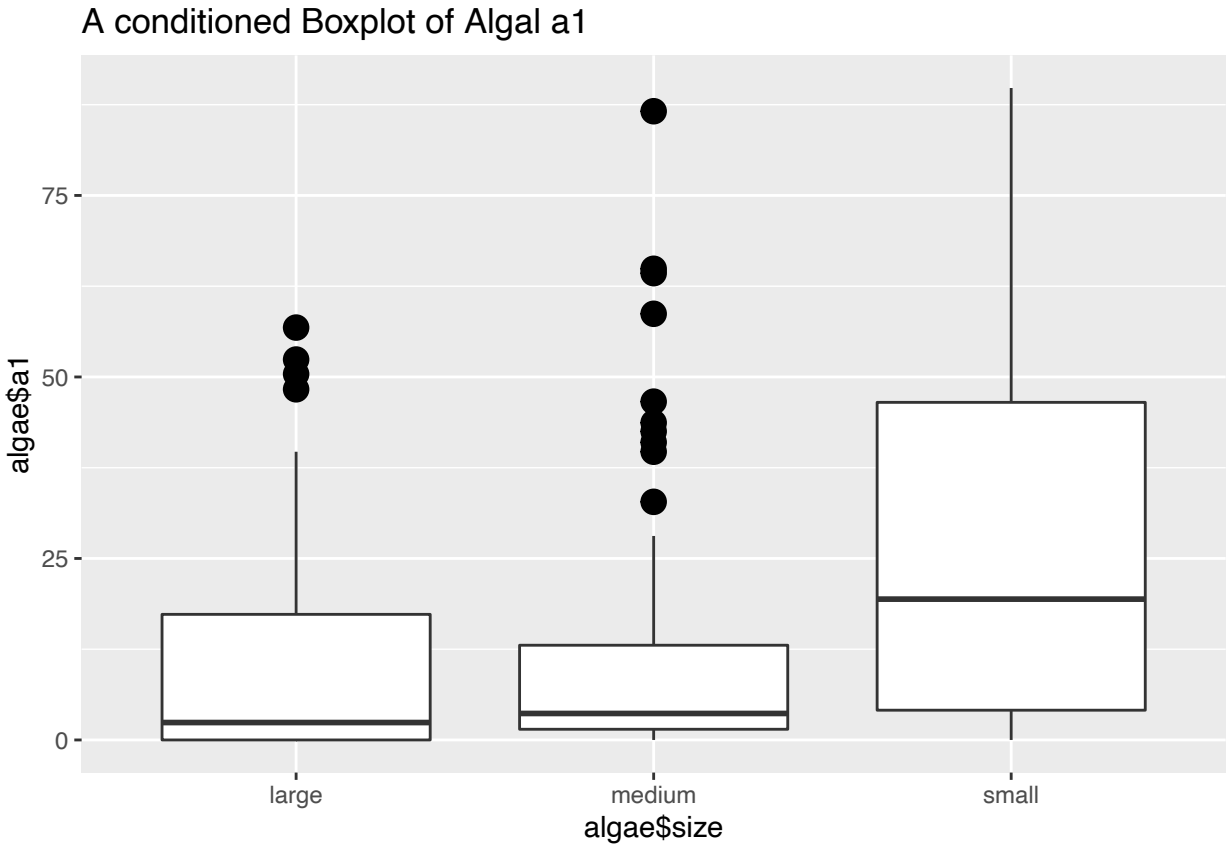
b. "PROBLEM NUMBER TWO, PART B"

```
# Adding additions of density curve and rugs
p2 <- ggplot(algae[, 4], aes(x = mxPH)) + geom_histogram(aes(y = ..density..), binwidth = densi
    4])$mxPH)$bw) + geom_density(fill = "red", alpha = 0.2) + labs(title = "Histogram of mxPH"
    y = "Probability", x = "max PH")
p2 + geom_rug()
```



c. "PROBLEM NUMBER TWO, PART C"

```
# plotting boxplot of a1~size
p3 <- ggplot(algae, aes(x = algae$size, y = algae$a1)) + geom_boxplot(outlier.colour = "black"
    outlier.size = 4) + labs(title = "A conditioned Boxplot of Algal a1")
p3
```

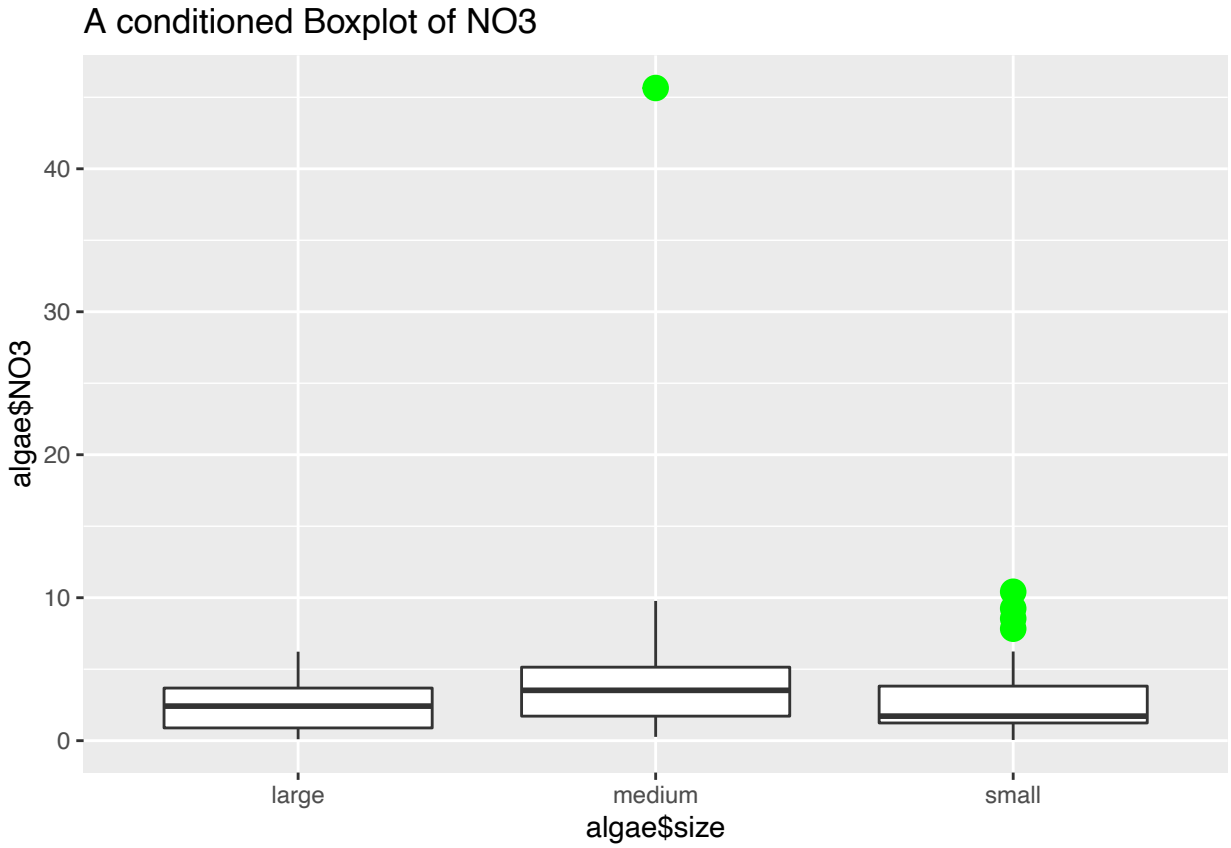


d. "PROBLEM NUMBER TWO, PART D"

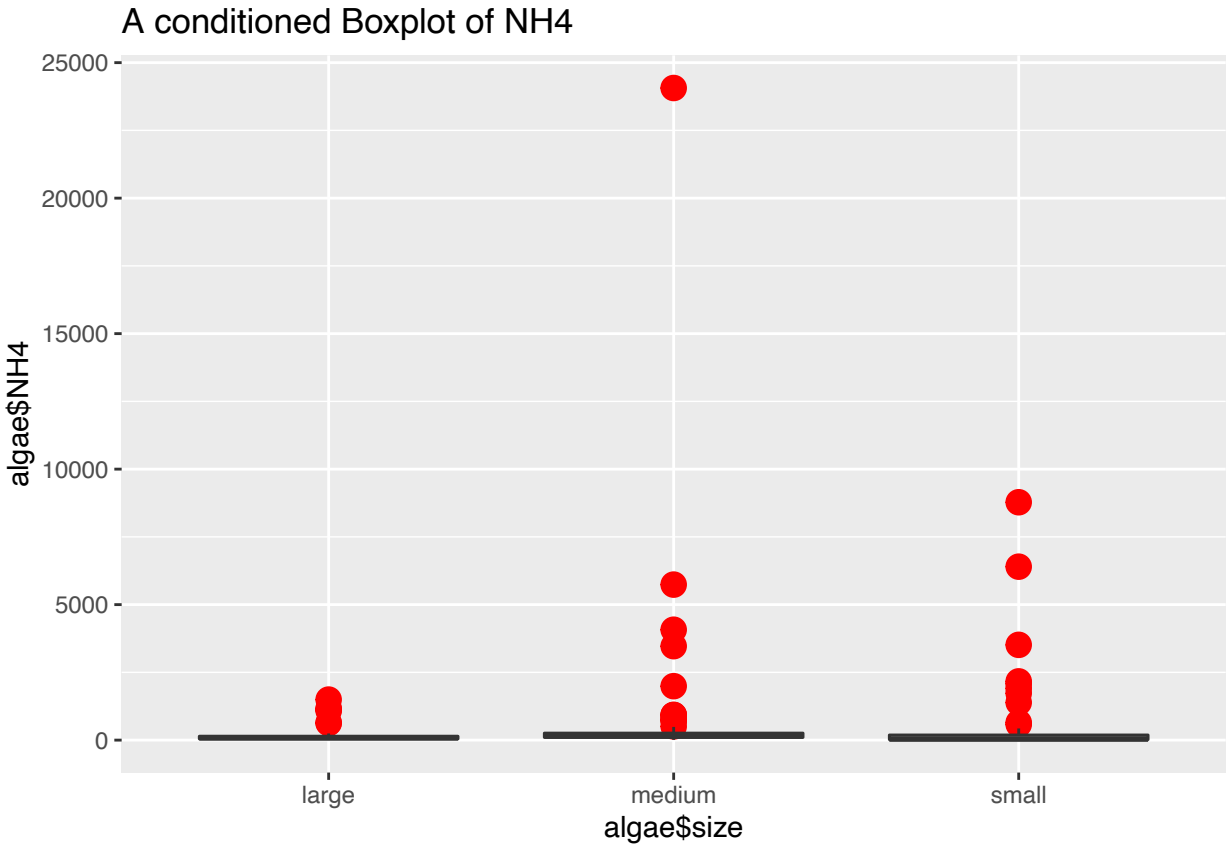
```
# libraries to test for outliers
library(car)
library(outliers)

# plot for NO3~size
p4 <- ggplot(algae, aes(x = algae$size, y = algae$NO3)) + geom_boxplot(outlier.colour = "green"
    outlier.size = 4) + labs(title = "A conditioned Boxplot of NO3")

p4
```




```
# plot for NH4~size
p5 <- ggplot(algae, aes(x = algae$size, y = algae$NH4)) + geom_boxplot(outlier.colour = "red",
    outlier.size = 4) + labs(title = "A conditioned Boxplot of NH4")
p5
```




```
# outleirs based on IQR criteria where any value above Q3 + 1.5IQR or below Q1 -
# 1.5IQR is an outlier to use htis test must omit outliers
boxplot.stats(na.omit(algae)$NO3)$out
```

```
## [1] 10.416  9.773  9.715 45.650
```

```
boxplot.stats(na.omit(algae)$NH4)$out
```

```
##  [1]   578.000  8777.600  1729.000  3515.000  6400.000  1911.000   647.570
##  [8]  1386.250  2082.850  2167.370   737.500   914.000  5738.330  4073.330
## [15]   758.750   931.833   723.667  3466.660   920.000  1990.160 24064.000
## [22]  1131.660  1495.000   643.000   627.273  1168.000  1081.660
```

```
# outlier test package for extreme cases
outlierTest(lm(algae$NO3 ~ algae$size))
```

```
##     rstudent unadjusted p-value Bonferroni p
## 153 18.89171         7.6474e-46   1.5142e-43
```

```r
outlierTest(lm(algae$NH4 ~ algae$size))
```

```
##      rstudent unadjusted p-value Bonferroni p
## 153 23.247639         5.5153e-58   1.0920e-55
## 20   4.455082         1.4152e-05   2.8021e-03
```

According to the IQR criteria, where an outlier is defined as any value above Q3 + 1.5IQR or below Q1 - 1.5IQR using the function boxplots.stats, there are outliers at the values indicated. Using the function outlierTest, with multiple outliers on both appearing on the tests. There also appears to be outliers in both plots, with the 153rd observation appearing in both.

e. "PROBLEM NUMBER TWO, PART E"

Mean of NO3 was 3.28 and NH4 is 501.3 and variance: 14.26 and 3851585 Medians are 2.67 and 103.16 and MAD: 2.17 and 111.675

Outliers have higher variance from the rest of the data points. The computation of the mean and variance uses outliers, having outlier influence. On the other hand, MAD and median are better and more robust because they are less sensitive to outliers. This explains why the means and variances of NO3 and NH4 are so different from their respective medians and MADs.

3. "PROBLEM NUMBER THREE"

a. "PROBLEM NUMBER THREE, PART A"

```r
# frequency on NA
sum(is.na(algae))
```

```
## [1] 33
```

```r
# finding frequency of each chemical
colSums(is.na(algae))
```

```
## season   size  speed   mxPH   mnO2     Cl    NO3    NH4   oPO4    PO4   Chla
##      0      0      0      1      2     10      2      2      2      2     12
##     a1     a2     a3     a4     a5     a6     a7
##      0      0      0      0      0      0      0
```

There are 33 total observations that contain missing values. The only variables that contain missing values are mxPH (1), mn02(2), Cl(10), NO3(2), NH4(2),oPo4(2), Po4(2), Chla(12)

b. "PROBLEM NUMBER THREE, PART B"

```
# filter out NA values
algae.del1 <- algae %>% filter(complete.cases(.))
# observation frequency
nrow(algae.del1)
```

```
## [1] 184
```

There are 184 observations in algae.del

c. "PROBLEM NUMBER THREE, PART C"

```
# mutate algae to replace NA with median values
algae.med <- algae %>% mutate_at(vars("mxPH", "mnO2", "Cl", "NO3", "NH4", "oPO4",
    "PO4", "Chla"), ~ifelse(is.na(.), median(., na.rm = TRUE), .))

nrow(algae.med)
```

```
## [1] 200
```

```
algae.med[c(48, 62, 199), ]
```

```
## # A tibble: 3 x 18
##   season size  speed  mxPH  mnO2    Cl   NO3   NH4  oPO4   PO4  Chla    a1    a2
##   <chr>  <chr> <chr> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
## 1 winter small low    8.06  12.6   9    0.23   10    5      6   1.1   35.5   0
## 2 summer small medi~  6.4    9.8  32.7  2.68  103.  40.2   14   5.48  19.4   0
## 3 winter large medi~  8      7.6  32.7  2.68  103.  40.2  103.  5.48   0    12.5
## # ... with 5 more variables: a3 <dbl>, a4 <dbl>, a5 <dbl>, a6 <dbl>, a7 <dbl>
```

There are 200 observation like the origin data set, NA values replaced with medians. 48th, 62th and 199th observations are shown above.

d. "PROBLEM NUMBER THREE, PART D"

```
# finding correlations using cor() function
xmat <- algae.med %>% select("mxPH", "mnO2", "Cl", "NO3", "NH4", "oPO4", "PO4", "Chla")
xmat
```

```
## # A tibble: 200 x 8
##     mxPH  mnO2    Cl    NO3   NH4  oPO4   PO4  Chla
##    <dbl> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl> <dbl>
##  1  8      9.8  60.8  6.24  578   105   170   50
##  2  8.35   8    57.8  1.29  370   429.  559.   1.3
##  3  8.1   11.4  40.0  5.33  347.  126.  187.  15.6
```

```
##  4  8.07   4.8  77.4  2.30   98.2  61.2 139.   1.4
##  5  8.06   9    55.4 10.4   234.   58.2  97.6 10.5
##  6  8.25  13.1  65.8  9.25  430    18.2  56.7 28.4
##  7  8.15  10.3  73.2  1.54  110    61.2 112.   3.2
##  8  8.05  10.6  59.1  4.99  206.   44.7  77.4  6.9
##  9  8.7    3.4  22.0  0.886 103.   36.3  71    5.54
## 10  7.93   9.9   8    1.39    5.8  27.2  46.6  0.8
## # ... with 190 more rows
```

```
cor(xmat)
```

```
##              mxPH        mnO2          Cl         NO3         NH4       oPO4
## mxPH  1.00000000 -0.16793588  0.13348318 -0.12637570 -0.08905891  0.1604940
## mnO2 -0.16793588  1.00000000 -0.27790470  0.09853221 -0.08731331 -0.4150941
## Cl    0.13348318 -0.27790470  1.00000000  0.22532102  0.07450448  0.3927796
## NO3  -0.12637570  0.09853221  0.22532102  1.00000000  0.72152844  0.1450640
## NH4  -0.08905891 -0.08731331  0.07450448  0.72152844  1.00000000  0.2277842
## oPO4  0.16049404 -0.41509407  0.39277958  0.14506398  0.22778417  1.0000000
## PO4   0.18976104 -0.48641358  0.45668016  0.16988077  0.20913887  0.9132424
## Chla  0.38915072 -0.16571514  0.15158609  0.14342461  0.09447493  0.1307048
##              PO4        Chla
## mxPH  0.1897610  0.38915072
## mnO2 -0.4864136 -0.16571514
## Cl    0.4566802  0.15158609
## NO3   0.1698808  0.14342461
## NH4   0.2091389  0.09447493
## oPO4  0.9132424  0.13070484
## PO4   1.0000000  0.26920346
## Chla  0.2692035  1.00000000
```

```
# prediction 28th observation
prediction <- predict(lm(PO4 ~ oPO4, data = algae))  #from PIAZZA this is algae not algae.med
prediction[28]
```

```
##       29
## 76.51663
```

76.51663 is the predicted value for obervation 28 using algae dataset

d. "PROBLEM NUMBER THREE, PART D" Similarily to the example given in lecture 2 about the bullet holes on crashed planes, the idea that the true missing values is always better than replacement values such as the median or correlation. It about how and why the data is missing, just like the bullet patterns on the planes, rather than a potential subsitute on where they will be.

4. "PROBLEM NUMBER FOUR"

a. "PROBLEM NUMBER FOUR, PART A"

```
# Specify we want a 5-fold CV
nfold = 5
# cut: divides all training observations into 5 intervals; labels = FALSE
# instructs R to use integers to code different intervals randomize using
# sample()
set.seed(66)
folds = cut(1:nrow(algae.med), breaks = nfold, labels = FALSE) %>% sample()
folds
```

```
##   [1] 3 3 5 4 1 4 5 3 3 2 1 4 1 4 1 2 3 3 5 5 3 2 1 3 5 2 4 3 5 2 1 4 4 2 4 3 4
##  [38] 4 3 1 2 4 1 5 4 2 5 2 2 1 2 5 4 3 5 1 5 1 1 2 2 2 2 1 4 2 3 4 4 1 3 4 4 5
##  [75] 4 5 1 2 2 3 1 5 5 1 1 1 4 5 2 3 1 4 3 5 1 2 3 4 5 5 1 1 5 5 5 3 5 4 4 3 3
## [112] 5 2 3 4 1 3 2 3 5 5 5 4 1 2 3 3 5 2 3 2 1 2 3 4 4 3 2 3 1 2 1 5 5 2 1 1 4
## [149] 4 2 5 3 4 5 1 2 1 4 2 3 2 3 3 1 5 4 3 5 4 1 1 4 2 4 4 1 1 5 4 3 2 3 3 1 2
## [186] 2 1 3 5 5 4 5 1 2 3 5 2 5 4 2
```

b. "PROBLEM NUMBER FOUR, PART B"

```
library(plyr)
do.chunk <- function(chunkid, chunkdef, dat){ # function argument

  train = (chunkdef != chunkid)

  Xtr = dat[train,1:11] # get training set
  Ytr = dat[train,12] # get true response values in trainig set

  Xvl = dat[!train,1:11] # get validation set
  Yvl = dat[!train,12] # get true response values in validation set

  lm.a1 <- lm(a1~., data = dat[train,1:12])
  predYtr = predict(lm.a1) # predict training values
  predYvl = predict(lm.a1,Xvl) # predict validation values

  data.frame(fold = chunkid,
  train.error = mean((predYtr - Ytr$a1)^2), # compute and store trainin error UPDATED FROM PIA
  val.error = mean((predYvl - Yvl$a1)^2)) # compute and store test error UPDATED FROM PIAZZA
}

set.seed(66)
# Loop through different number of neighbors
# print out errors
tmp = ldply(1:nfold, do.chunk, chunkdef = folds, dat = algae.med)
print(tmp)
```

```
##   fold train.error val.error
## 1    1    290.3775  285.3887
## 2    2    240.6154  506.5678
## 3    3    296.3188  256.5233
## 4    4    280.9803  400.1096
## 5    5    299.8153  257.5973
```

5. "PROBLEM NUMBER FIVE"

```
setwd("/Users/MatthewXu/Downloads/algae_data/")

knitr::opts_knit$set(root.dir = "/Users/MatthewXu/Downloads/algae_data/")
algae.Test <- read_table2("algaeTest.txt", col_names = c("season", "size", "speed",
    "mxPH", "mnO2", "Cl", "NO3", "NH4", "oPO4", "PO4", "Chla", "a1"), na = c("XXXXXXX"))

lm.a1 <- lm(a1 ~ ., data = algae.med[1:12])
predYvl = predict(lm.a1, algae.Test)  # predict validation values

tmp2 <- data.frame(val.error = mean((predYvl - algae.Test$a1)^2))  # compute and store test er

# from piazza, using mean to find error in both types of error we are only
# concerned with test error for this question on both data sets algae.Test and
# algae.med
cat("algae.med mean val error", mean(tmp$val.error))
```

```
## algae.med mean val error 341.2373
```

```
cat(" algae.Test mean val error", mean(tmp2$val.error))
```

```
##  algae.Test mean val error 250.1794
```

The mean val error is smaller on the test model than that of algae.med, which is what we want, meaning the model is getting better when analyzing both datasets.

6. "PROBLEM NUMBER SIX"

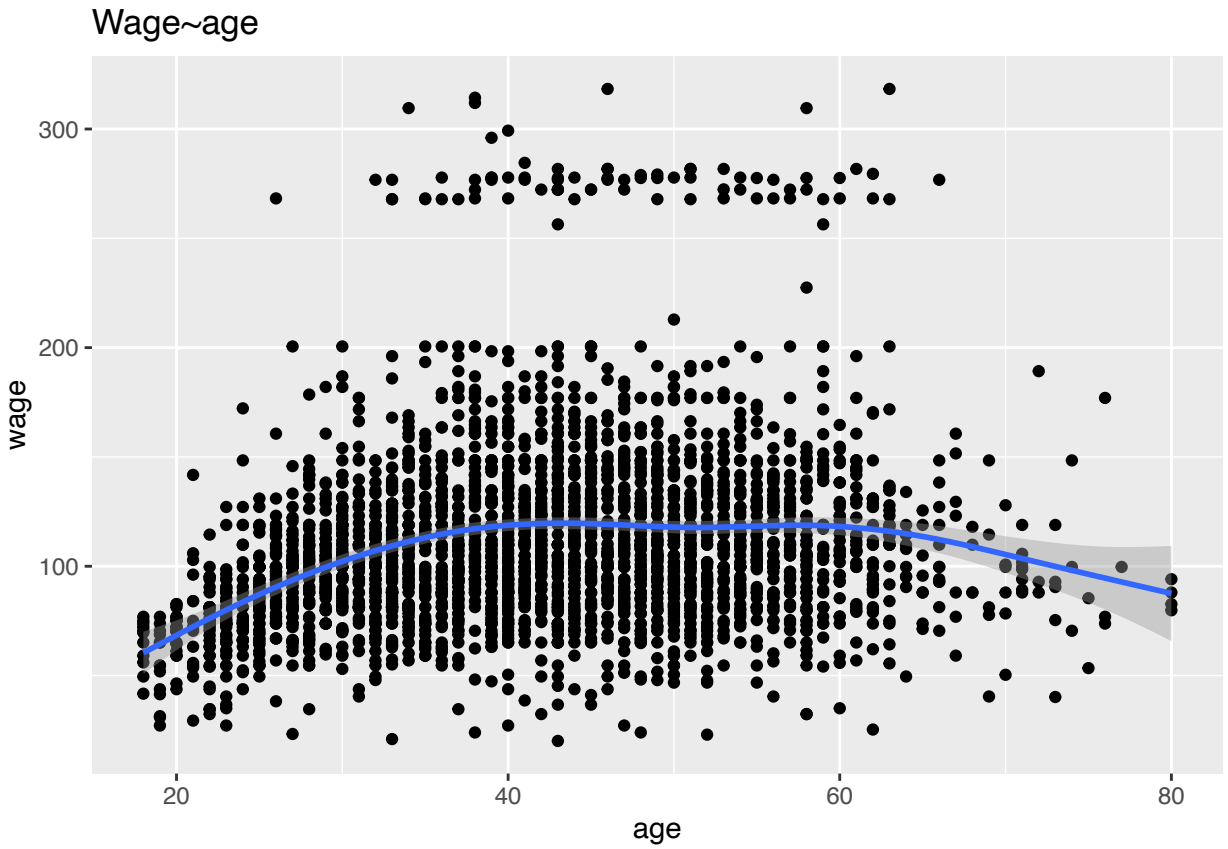
```
library(ISLR)
head(Wage)
```

```
##        year age           maritl     race       education             region
## 231655 2006  18 1. Never Married 1. White    1. < HS Grad 2. Middle Atlantic
```

```
## 86582  2004  24 1. Never Married 1. White 4. College Grad 2. Middle Atlantic
## 161300 2003  45       2. Married 1. White 3. Some College 2. Middle Atlantic
## 155159 2003  43       2. Married 3. Asian 4. College Grad 2. Middle Atlantic
## 11443  2005  50      4. Divorced 1. White      2. HS Grad 2. Middle Atlantic
## 376662 2008  54       2. Married 1. White 4. College Grad 2. Middle Atlantic
##              jobclass         health health_ins  logwage      wage
## 231655  1. Industrial      1. <=Good      2. No 4.318063  75.04315
## 86582  2. Information 2. >=Very Good      2. No 4.255273  70.47602
## 161300  1. Industrial      1. <=Good     1. Yes 4.875061 130.98218
## 155159 2. Information 2. >=Very Good     1. Yes 5.041393 154.68529
## 11443  2. Information      1. <=Good     1. Yes 4.318063  75.04315
## 376662 2. Information 2. >=Very Good     1. Yes 4.845098 127.11574
```

a. "PROBLEM NUMBER SIX, PART A"

```
library(ggplot2)
ggplot(Wage, aes(age, wage)) + geom_point() + geom_smooth() + labs(title = "Wage~age")
```



Wages appear to be lowest at the earliest ages and highest ages. This is expected as most people earn their wages in society in the middle of their lives. They may be too oung to work earlier and retired later in life.

b. "PROBLEM NUMBER SIX, PART B"

```r
Wage_wage <- Wage$wage
Wage_age <- Wage$age

do.chunk2 <- function(chunkid, chunkdef, dat, p){ # function argument

  train = (chunkdef != chunkid)

  Xtr = dat[train,1:10] # get training set
  Ytr = dat[train,11] # get true response values in trainig set

  Xvl = dat[!train,1:10] # get validation set
  Yvl = dat[!train,11] # get true response values in validation set

  #poly() cannot take in degree zero, need to manually lm a intercept only model
  if (p != 0){
    lm.a2 <- lm(Wage_wage ~ poly(Wage_age, p, raw=FALSE), data = dat[train,1:11])
  }else{
    lm.a2 <- lm(Wage_wage ~ 1, data = dat[train,1:11]) #intercept-only model
  }

  predYtr = predict(lm.a2) # predict training values
  predYvl = predict(lm.a2,Xvl) # predict validation values

  data.frame(fold = chunkid, degree = p,
  train.error = mean((predYtr - Ytr)^2), # compute and store trainin error
  val.error = mean((predYvl - Yvl)^2))# compute and store test error

}
#cut: divides all training observations into 5 intervals
folds2 = cut(1:nrow(Wage), breaks=nfold, labels=FALSE) %>% sample()
#empty dataframe to update poly values
res = data.frame(degree=integer(), fold=integer(), train.error=double(), val.error=double())

#loop to add a polynomials errors to empty dataframe
for(i in 0:10){
    poly <- ldply(1:nfold, do.chunk2, chunkdef = folds2, dat = Wage, p = i)
    res <- rbind(res, poly)
}

#output mean of errors by degree
res %>%
    group_by(degree) %>%
    summarise_at(.vars=c("train.error", "val.error"), mean)
```

```
## # A tibble: 11 x 3
##    degree train.error val.error
##     <int>       <dbl>     <dbl>
```

```
##  1      0       1772.     1741.
##  2      1       1835.     1804.
##  3      2       1903.     1875.
##  4      3       1906.     1880.
##  5      4       1908.     1880.
##  6      5       1909.     1880.
##  7      6       1911.     1882.
##  8      7       1912.     1882.
##  9      8       1912.     1882.
## 10      9       1913.     1884.
## 11     10       1913.     1884.
```

c. "PROBLEM NUMBER SIX, PART C"

```
library(ggplot2)

# plot errors of polynomials
g <- ggplot(res, aes(x = degree, y = Value), title) + geom_line(aes(y = res$train.error,
    colour = "train.error")) + geom_line(aes(y = res$val.error, colour = "val.error")) +
    labs(title = "Plot of Train and Val Error")
g
```

The graph is depicted this way as all values are pin pointed per degree and ggplot draws a line through them at each degree. As you can see, looking at the direction and slopes that the errors,

it appears to converge beyond degree = 3, indicating that this may be the best model, given its number of predictors as well that are possible to include (flexibility).