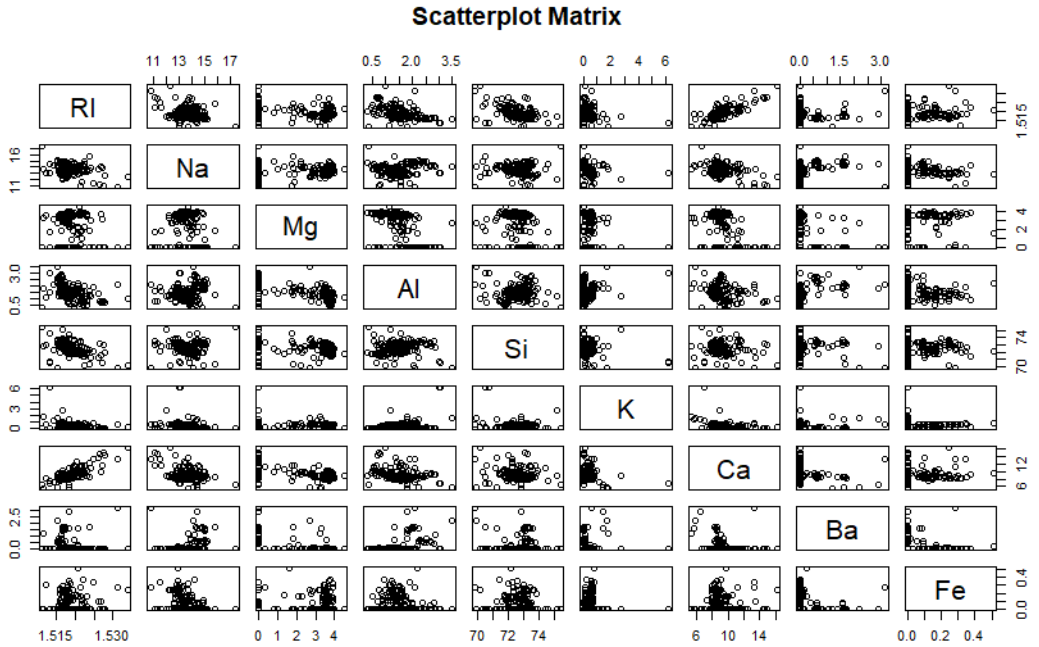
MA4790 Homework 1

Ian Boulis

3.1

a. Using visualizations, explore the predictor variables to understand their distributions as well as the relationships between predictors.

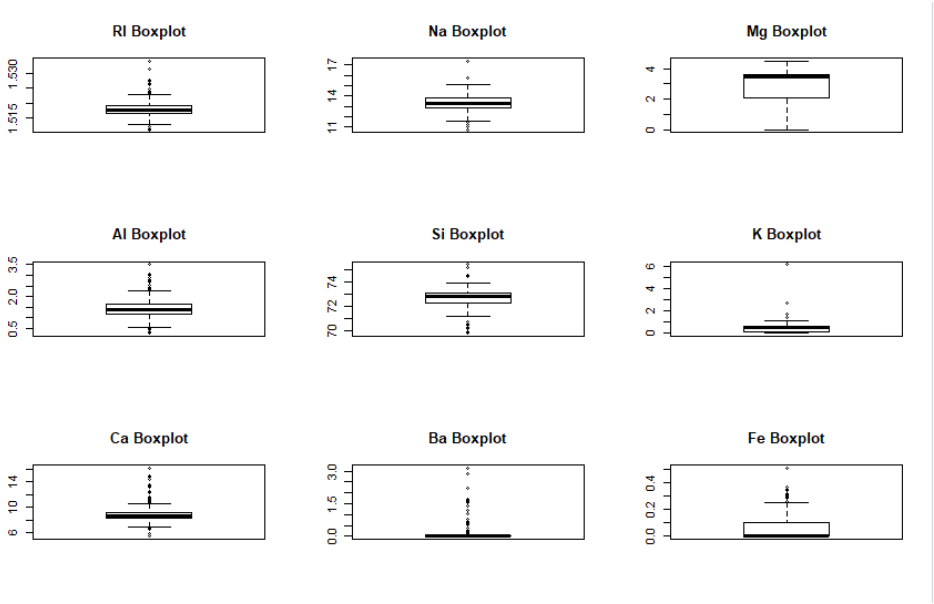
Using a scatterplot matrix will let give us a visual representation of which predictors are correlated, as well as how strong that correlation is.



Most of the predictors seem to be uncorrelated. With the exception of Ri and Si which could have a slightly negative correlation, and Ri and Ca which show an obvious positive correlation.

b. Do there appear to be any outliers in the data? Are any predictors skewed?

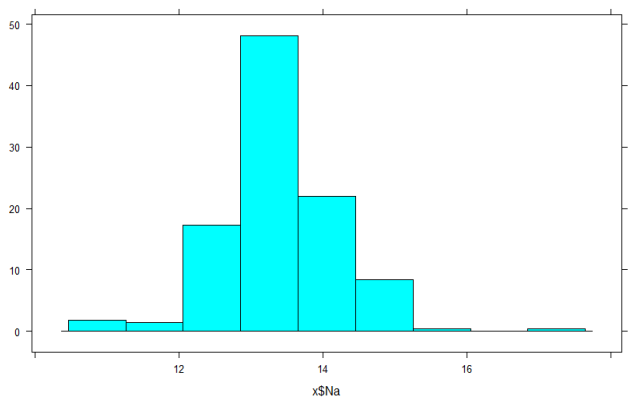
Using boxplots will very easily show us which data points are outliers.

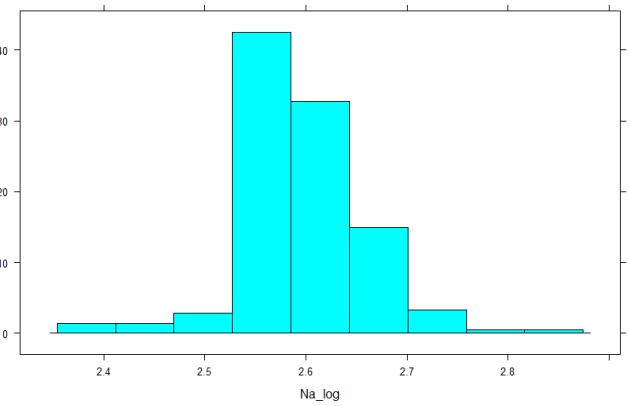


Predictor Variable	Skewness	Interpretation
RI	1.6	Heavily right skewed
Na	0.4	Symmetric
Mg	-1.1	Heavily left skewed
Ai	0.9	Moderately right skewed
Si	-0.7	Moderately left skewed
K	6.5	Heavily right skewed
Ca	2.0	Heavily right skewed
Ba	3.4	Heavily right skewed
Fe	1.7	Heavily right skewed

c. Are there any relevant transformations of one or more predictors that might improve the classification model?

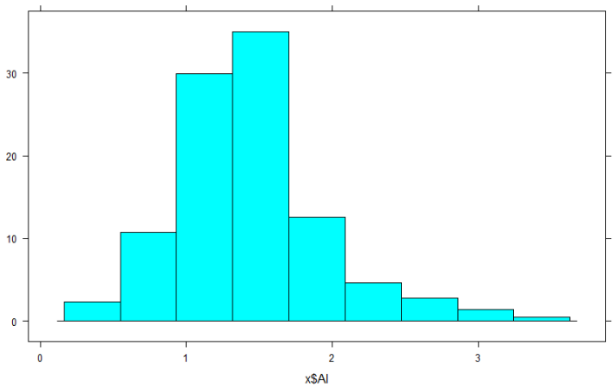
Calculating the Lambda value for the Na predictor resulted in a value close to 0, which is indicative of a log transformation. Here are the before and after histograms of Na.

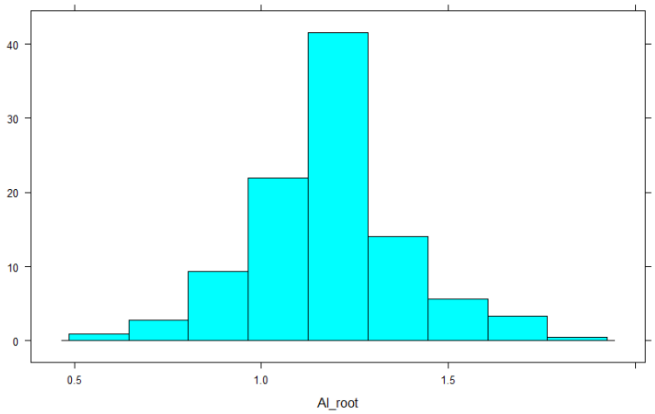




After applying the log transformation, the data is centered around a much lower number.

Calculating Lambda for the Al predictor we get a lambda equal near 0.5, which implies that a square root transformation is appropriate. These are the before and after histograms of Al.





After applying the transformation, the data is much more symmetric.

3.2

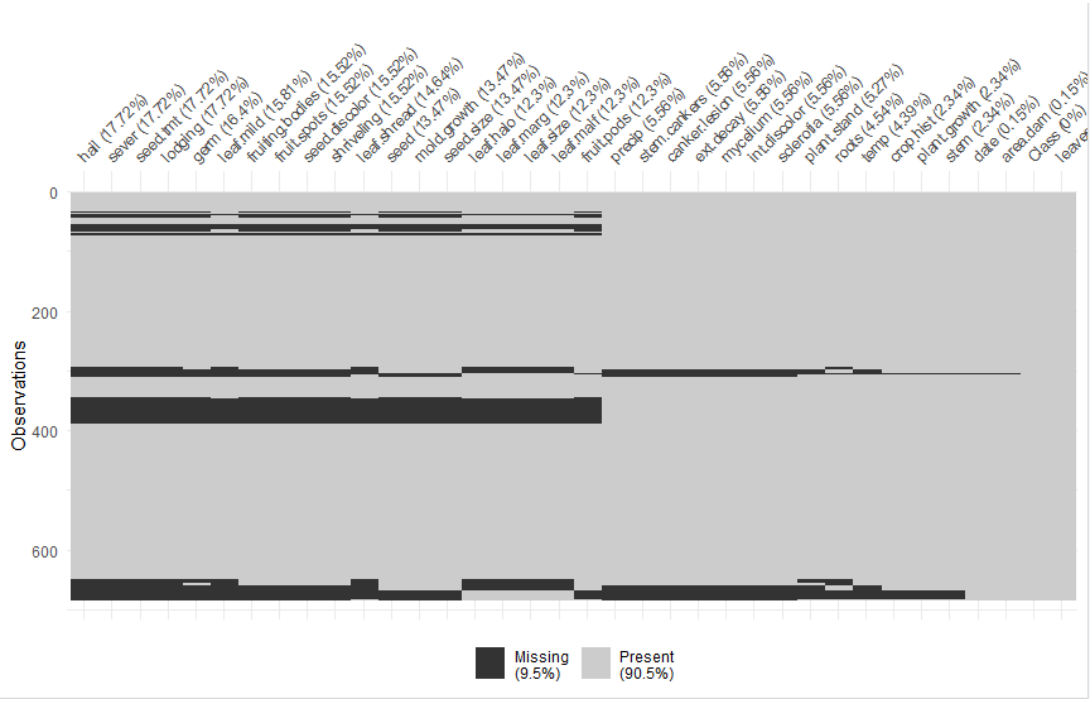
a. Investigate the frequency distribution for the categorial predictors. Are any of the distributions degenerate in the ways discussed earlier in the chapter?

Running the nearZeroVar function on the data set tells us that “leaf.mild”, “mycelium”, and “sclerotia” all have variances that are near zero. No variables in the set have a variance of zero.

```
> nearZeroVar(Soybean, names=TRUE)
[1] "leaf.mild" "mycelium"  "sclerotia"
```

b. Roughly 18% of the date are missing. Are there particular predictors that are more likely to be missing? Is the pattern of the missing data related to the classes?

The library naniar has a plethora of visualization of missing data points. Running this Vis_miss function will give us an easy visualization of the sorted missing data.



Referencing this with the raw data, there is an easy to see correlation between the missing data points and the location of the “phytophthora-rot” class, which implies that the missing data is class related, and the aforementioned class has the brunt of the missing data.

c. Develop a strategy for handling the missing date.

Since the missing data is class related and there is over 90% data that is not missing from that class, an imputation technique would most likely fill out the rest of the data set will pretty good accuracy. The method that we discussed in class (K-nearest neighbor) would be appropriate here.

3.3

b. Generally speaking, are there strong relationships between the predictor data? If so, how could correlations in the predictor set be reduced? Does this have a dramatic effect on the number of predictors available for modeling?

There does not appear to be any strong relationships between any pairs in the predictors. There are a handful that show possible positive relationships, mostly in the first set of 10 predictor plots. There are a lot of predictors in this set so I’m not going to include graphs of them in this write up but will include the visualization R code. You could perform PCA on this data set to reduce the amount of correlation in predictors. It also might be worth removing some predictors as well considering how large this set is and the removed predictors might not make a significant difference.

```
###3.1###

library(AppliedPredictiveModeling)

library(mlbench)

data(Glass)

str(Glass)

dim(Glass)

x <- Glass[,1:9]      #Compacting the data into a variable to make it easier to work with#

par(mfrow = c(3,3))

for (i in 1:ncol(x))  #Looping through the data so there is less code to write#

  pairs(x, main ="Scatterplot Matrix")#   #Scatterplot to show relationship between predictors

  boxplot(x[ ,i], main = paste(names(x[i]), "Boxplot")) #Boxplot to see for any outliers

library(e1071)

skewness(Glass$RI)    #Calculating skewness for each variable

skewness(Glass$Na)

skewness(Glass$Mg)

skewness(Glass$Al)

skewness(Glass$Si)

skewness(Glass$K)
```

```
skewness(Glass$Ca)

skewness(Glass$Ba)

skewness(Glass$Fe)

YY = scale(x)       #Scaling the data

head(colMeans(YY))

var(YY[,1:9])

library(caret)

NaTrans <- BoxCoxTrans(x$Na) #performing a transformation on the first predictor

NaTrans

histogram(x$Na)

Na_log <- log(x$Na)

histogram(Na_log)

GlassTrans <- BoxCoxTrans(x$Al) #performing a transformation on the second predictor

GlassTrans

histogram(x$Al)

Al_root <-sqrt(x$Al)

histogram(Al_root)
```

```
###3.2###

data(Soybean)

str(Soybean)

nearZeroVar(Soybean, names=TRUE)  #using nearZeroVar to see which predictors have a near zero variance

library(naniar)

vis_miss(Soybean, sort_miss = TRUE)    #visualization of missing data

###3.3###

library(caret)

data(BloodBrain)

x <- bbbDescr[,1:50]    #reassigning the data were working with into a single character variable

pairs(x[,1:10])         #Visualization for all the variables at the same time was pretty un-optimized so I split them

pairs(x[,10:20])        #up into groups of 10

pairs(x[,20:30])
```

```
pairs(x[,30:40])

pairs(x[,40:50])
```