Assignment 1

Matt Hayden

3/31/2017

Introduction

In this assignment, we will perform a data quality check and an exploratory data analysis of the wine dataset. In this dataset, the goal is to determine the quality of the wine based on chemical and physical characteristics of the wine. The quality is graded by being put into a group of Class I, II, or III. The samples in this dataset come from 3 different cultivars in Italy.

Data Quality Check

First, it is important to understand whether or not the data we have is trustworthy. The Wine dataset has 178 observations from the 3 different cultivars, and there are 13 predictors variables available for us to predict the dependent variable, Class. These variables and descriptions are described in Figure 1 below. While there are no missing values, we need to determine if the values are trustworthy.

Figure 1

Name	Type	Description
Class	factor	Class of wine
Alcohol	numeric	Percentage of alcohol content
Malic.Acid	numeric	Malic acid content
Ash	numeric	Ash content
Alkalinity	numeric	Alkalinity
Magnesium	numeric	Magnesium content

Total.Phenols	numeric	Total phenols
Flavanoids	numeric	Flavanoids
Nonflavanoid.Phenols	numeric	Nonflavanoid.Phenols
Proanthocyanins	numeric	Proanthocyanins
Color.Intensity	numeric	Intensity of the wine color
Hue	numeric	Hue of the wine
OD280.OD315	numeric	OD280/OD315 of diluted wines
Proline	numeric	Proline

As an initial check of data quality, we look at basic statistics for the predictors:

- Alcohol content ranges between 11 - 14.8% alcohol. This appears normal.
- Several predictors have values many standard deviations from the mean (e.g Malic Acid, Ash). These variables will need to be investigated further to make sure the results are expected.
- Magnesium is the only variables with values more than 4 standard deviations from the mean.

Figure 2

	Mean	Min	Max	SD	Missing	Largest SD
Alcohol	13	11.03	14.83	0.81	0	2.43
Malic.Acid	2.34	0.74	5.8	1.12	0	3.1
Ash	2.37	1.36	3.23	0.27	0	3.67
Alcalinity	19.49	10.6	30	3.34	0	3.15
Magnesium	99.74	70	162	14.28	0	4.36
Total.Phenols	2.3	0.98	3.88	0.63	0	2.53
Flavanoids	2.03	0.34	5.08	1	0	3.05
Nonflavanoid.Phenols	0.36	0.13	0.66	0.12	0	2.4
Proanthocyanins	1.59	0.41	3.58	0.57	0	3.48
Color.Intensity	5.06	1.28	13	2.32	0	3.43
Hue	0.96	0.48	1.71	0.23	0	3.29
OD280.OD315	2.61	1.27	4	0.71	0	1.96
Proline	746.9	278	1680	314.9	0	2.96

We can get an overall look at the distributions of the variables by looking at the boxplots for each of the variables. The dots in Figures 3 and 4 reveal potential outliers. We also see that Malic Acid and OD280/OD315 have more pronounced skews than the other variables. Transformations are likely to be a good idea for those variables. Overall the data looks like it is decent shape, but we should take a closer look.

Figure 3 - Boxplots of Variables

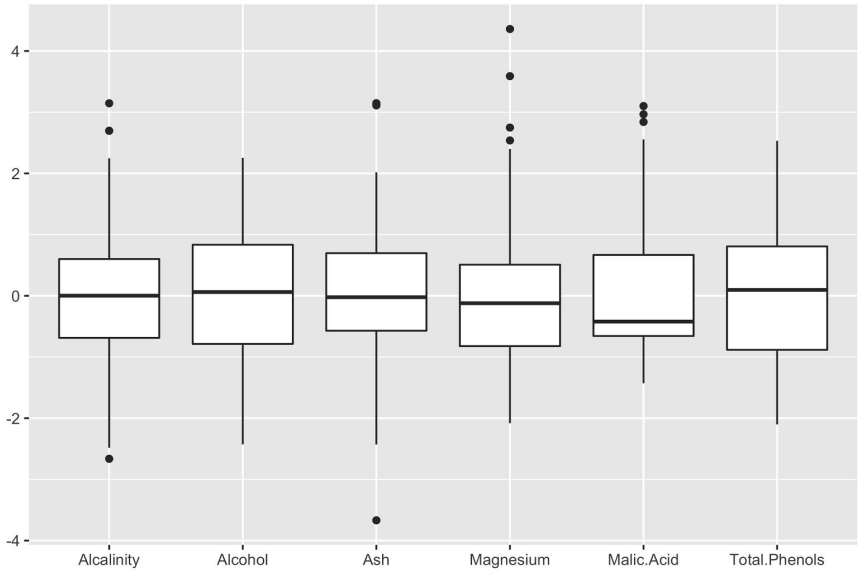
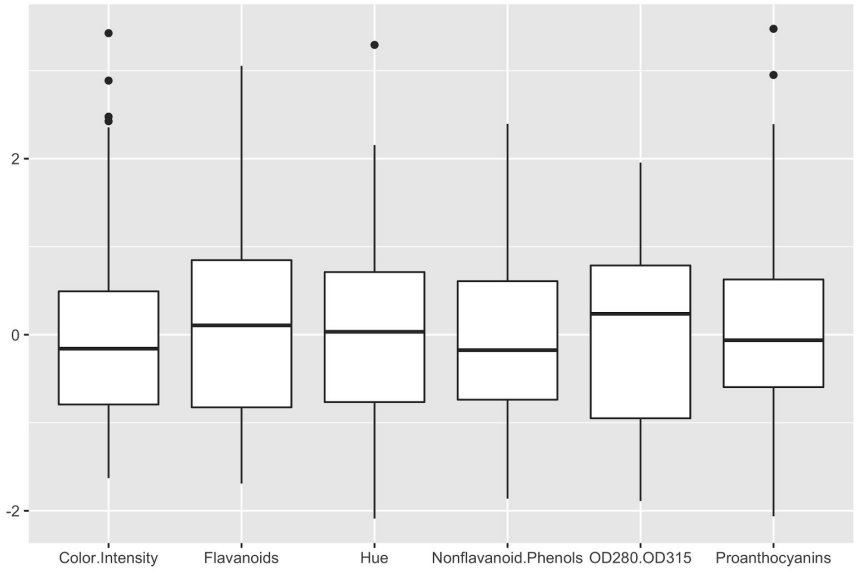
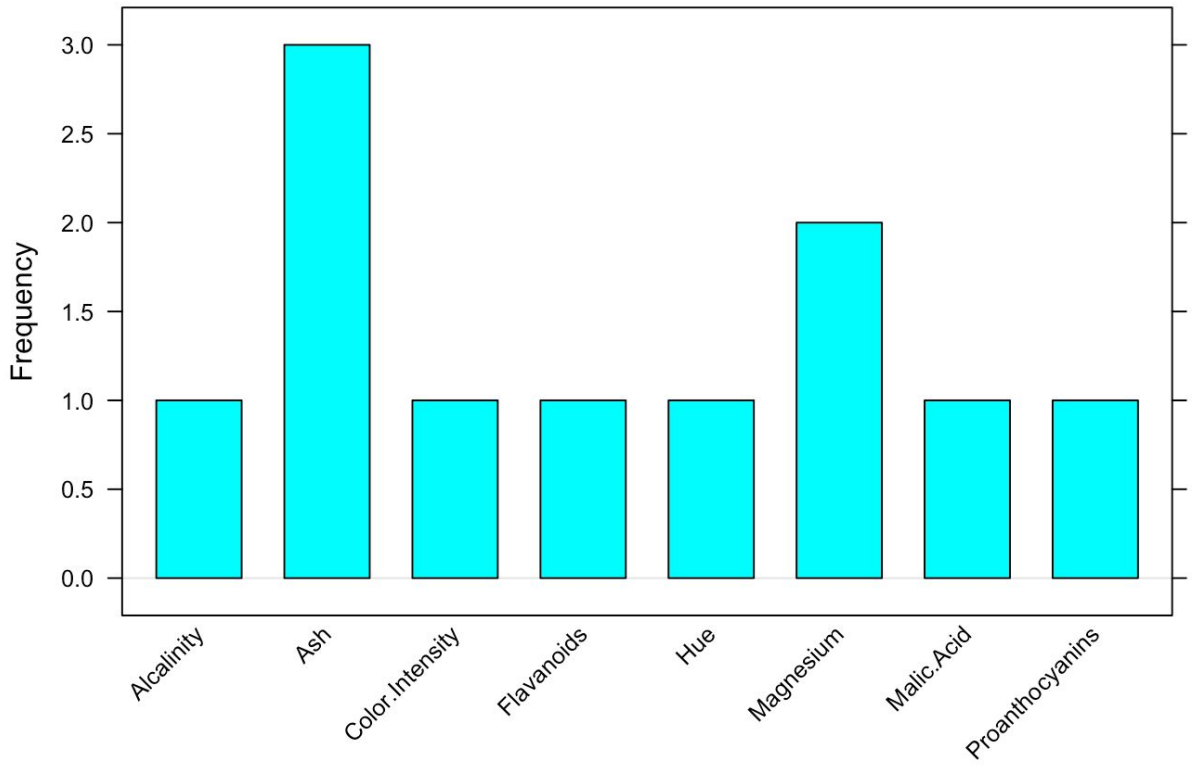


Figure 4 - Boxplots of Variables (cont)



In Figure 5 below, we are plotting the number of observations that variables have more than 2 standard deviations from the mean. Of the 178 observation, 56 of them fall outside of 2 standard deviations from the mean, and 17 of them have multiple variables that are more than 2 standard deviations. Looking at 3 standard deviations from the mean, there are only 10 observations, 1 of which has multiple variables more than 3 standard deviations from the mean. Given that their aren't many "extreme" observations, and not more than 4 standard deviations from the mean, it is probable that all of the data is acceptable to be included in a future analysis.

Figure 5 - Number of Values >2 Standard Deviations from Mean



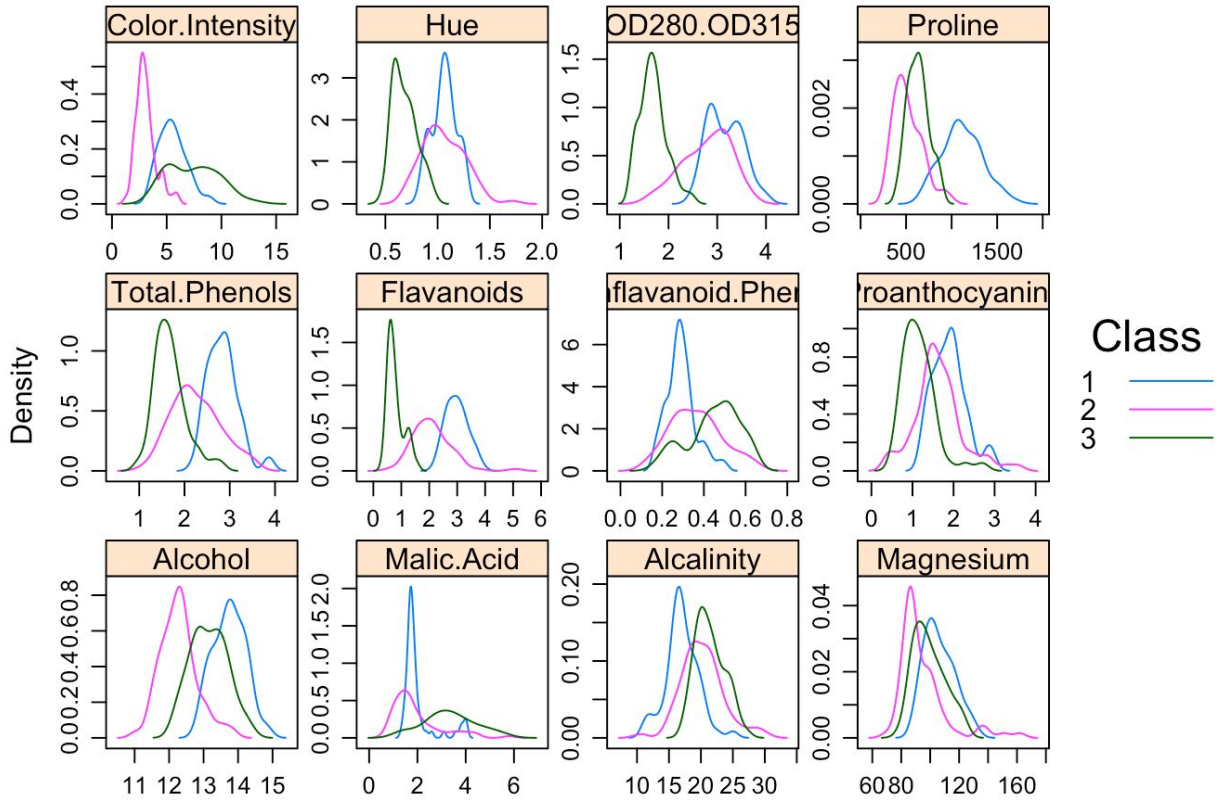
Exploratory Data Analysis

Now that the data is validated, we turn to an exploratory data analysis to understand the nature of the information we are looking at. First, in Figure 6, we are looking to see what the impact of Class is on the variables using density plots.

These plots show us that Class has an impact on almost all of the variables included in the study. None of the variables peak at the same value. And many of the variables have classes that

dominate value ranges, which is a good indicator that a machine learning algorithm can successfully determine the distinct classes. For example, Proline shows that values above 1000 are most likely in Class 1. And low Alcohol content indicates Class 2.

Figure 6 - Variable Density Plots



Next, we will look at the correlations between the variables to see what is related. This will help with addressing multi-collinearity issues in an analysis. We see the following variables with higher correlations that should be investigated further:

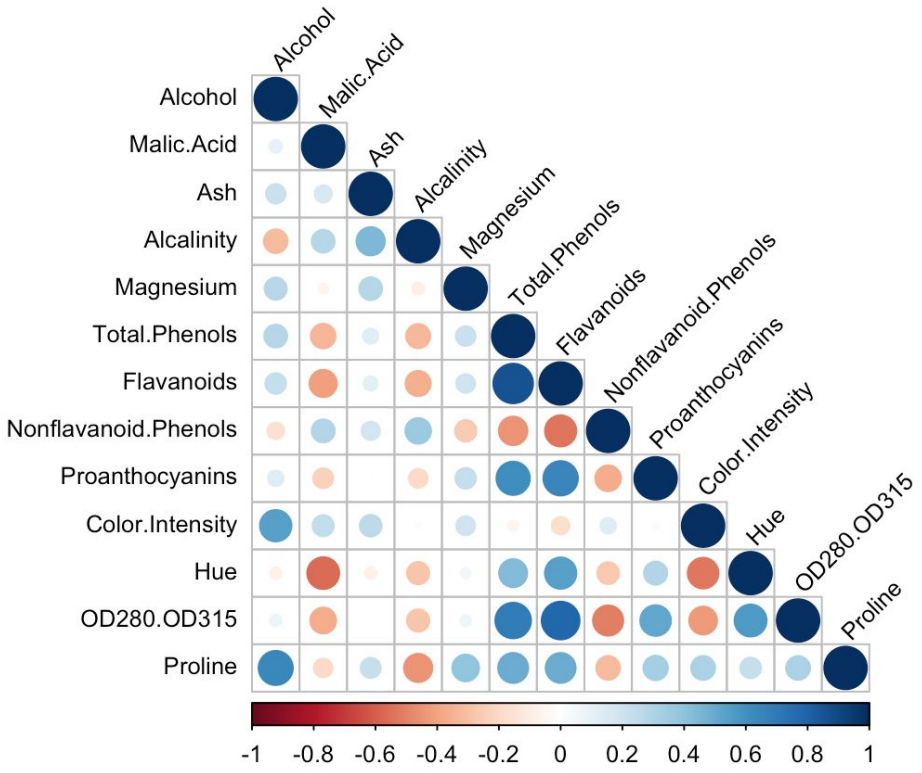
Positive Correlations

- Flavanoids and Total Phenols
- Flavanoids and OD280/OD315
- Alcohol and Proline

Negative Correlations

- Hue and Malic Acid

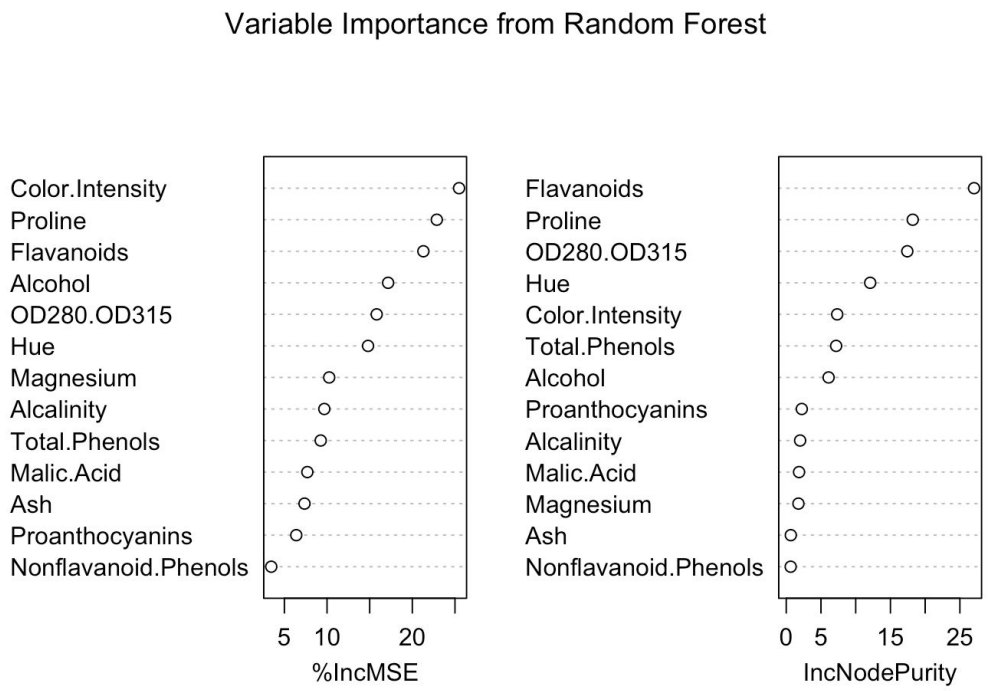
Figure 7 - Variable Correlation Matrix



Model-Based Exploratory Data Analysis

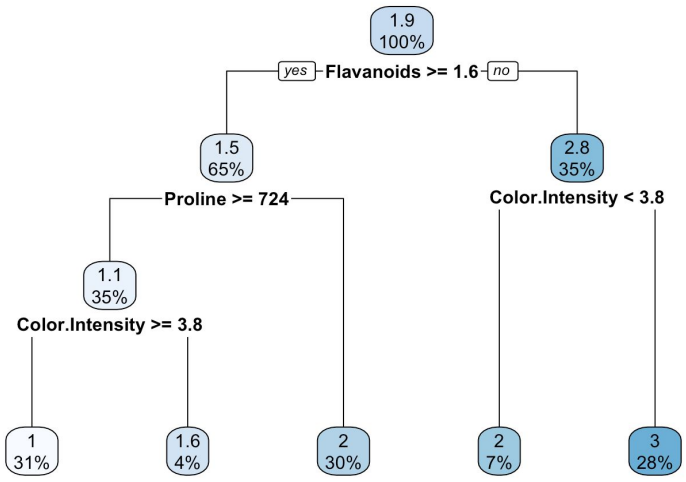
Doing basic model analysis of the Wine dataset can reveal some important truths within the data. By applying a random forest model, we are able to see which variables are more important in determining class. We see that Color Intensity, Proline, and Flavanoids account for the most improvements in mean squared error. Also, we see that Flavanoids have the best node purity. This is confirmed by looking at the density plots in Figure 6 above, that show 3 distinct Class curves with minimal overlap.

Figure 8



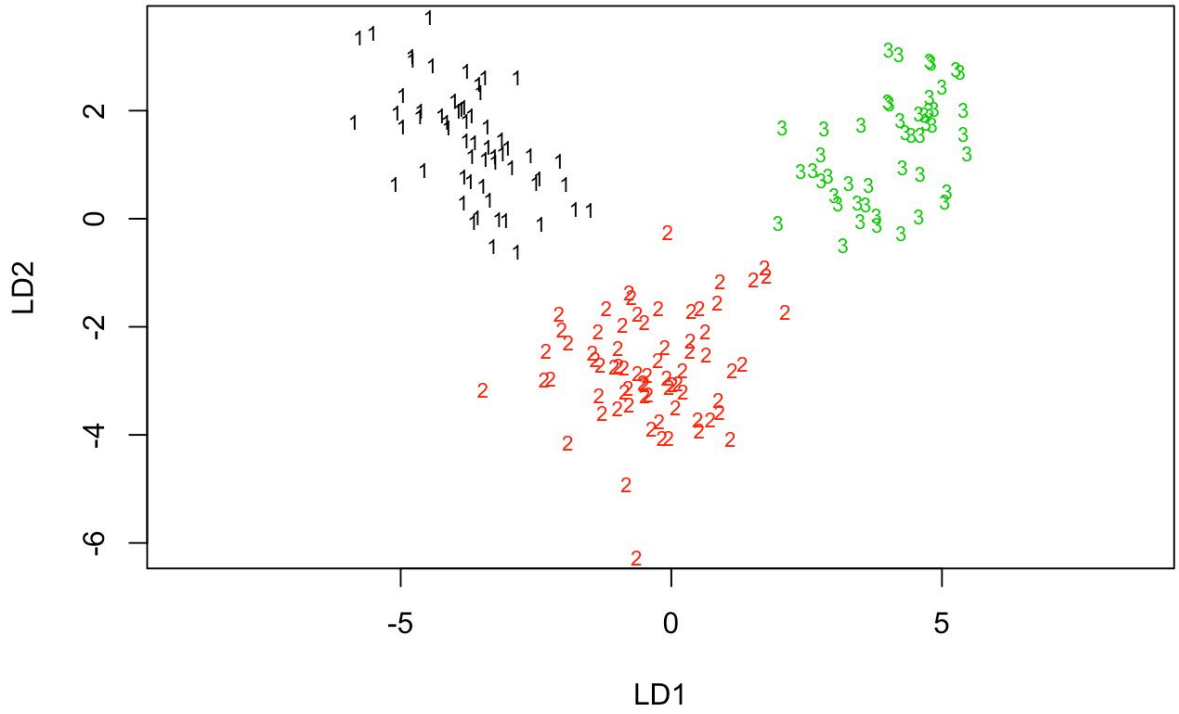
Next, we plot a decision tree to get a graphical display of which variables play the most important part in determining Class. The variables selected in this model correspond with the 4 variables with the highest node purity in the random forest model.

Figure 9 - Decision Tree Graph



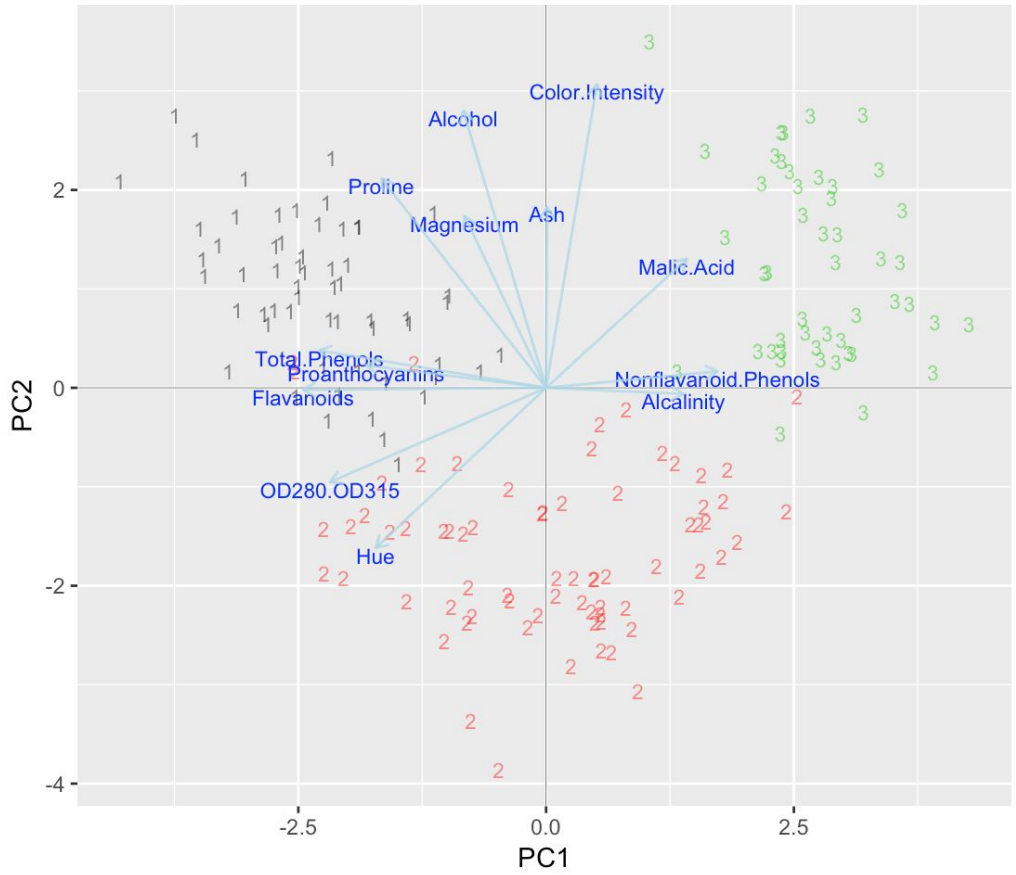
Next, performing a rudimentary Linear Discriminant Analysis (LDA), we are able to see that the modeling process identifies clear boundaries between the 3 classes. In LDA, the technique finds a combination of the predictors that gives maximum separation between the centers of the data while at the same time minimizing the variation within each group. The fact that we see clear boundaries means a modeling effort will find success.

Figure 10 - Linear Discriminant Analysis of LD1 vs LD2



Finally, we look at a Principal Component Analysis (PCA). This analysis shows us which variables explain the most variance in the the dataset. In PC1, Total Phenols and Flavanoids explain the most variance. In PC2, Color Intensity and Alcohol explain the most variance. These are likely to be important variables in a future analysis of this dataset.

Figure 11 - Principal Component Analysis of PC1 vs PC2



Conclusion

The results of the data quality check show that the data provided is in relatively trustworthy shape. Only 2 of the variables had a pronounced skew that would benefit from a variable transformation. Additionally, only 17 of the observations had multiple values past 2 standard deviations, and could be candidates for capping values or confirming their validity. Because there are only 178 observations, it will be import to salvage as many observations as possible.

Furthermore, the exploratory data analysis revealed the variables that are most likely to be useful in a predictive analysis. The tree based methods exposed Flavanoids, Proline, OD280/OD315, Hue, Alcohol, and Color Intensity as the most important variables. And the PCA analysis showed that Total Phenols is also important for explaining variance in the dataset. Given that LDA was successful in creating clear class boundaries, this initial analysis shows that modeling efforts will likely be successful for this use case.

Appendix - R Code

```
# Set code width to 60 to contain within PDF margins
knitr::opts_chunk$set(tidy = F, tidy.opts = list(width.cutoff = 60))

# Set all figures to be centered
knitr::opts_chunk$set(fig.align = "center")

# Set echo to off
knitr::opts_chunk$set(echo = F)

setwd('/Users/haydude/Google Drive/nu - 454 adv model/Assignment 1')
wine = read.csv('wine.data.csv',header = FALSE)
wine.variable.names = c('Class','Alcohol','Malic.Acid','Ash','Alcalinity',
                        'Magnesium','Total.Phenols','Flavanoids',
                        'Nonflavanoid.Phenols','Proanthocyanins',
                        'Color.Intensity','Hue','OD280.OD315','Proline')
colnames(wine) = wine.variable.names

wine.variable.types = apply(wine,2,class)
wine.variable.types[1] = 'factor'

wine.variable.descriptions = c('Class of wine',
                               'Percentage of alcohol content',
                               'Malic acid content',
                               'Ash content',
                               'Alcalinity',
                               'Magnesium content',
                               'Total phenols',
                               'Flavanoids',
                               'Nonflavanoid.Phenols',
                               'Proanthocyanins',
                               'Intensity of the wine color',
                               'Hue of the wine',
                               'OD280/OD315 of diluted wines',
                               'Proline')
wine.variable.table = data.frame(wine.variable.names, wine.variable.types, wine.variable.descriptions)
colnames(wine.variable.table) = c('Name','Type','Description')
```

```
library(pander)
temp = wine.variable.table
row.names(temp) = seq_along(1:dim(temp)[1])
pander(temp,justify = c('left', 'left', 'left'))

# missing values
checkmissing = table(is.na(wine)) # no missing values
checkmissing = which(is.na(wine), arr.ind = TRUE)

wine.means = sapply(wine[,-1],mean)
wine.mins = sapply(wine[,-1],min)
wine.maxs = sapply(wine[,-1],max)
wine.missing = apply(is.na(wine[,-1]),2,sum)
wine.sds = sapply(wine[,-1],sd)
wine.numbers = data.frame(wine.means, wine.mins, wine.maxs, wine.sds, wine.missing)
maxs = (wine.maxs - wine.means) / wine.sds
mins = (wine.means - wine.mins) / wine.sds
bigger = apply(data.frame(maxs,mins), 1, max)
wine.numbers = data.frame(wine.means, wine.mins, wine.maxs, wine.sds, wine.missing, bigger)
wine.numbers.names = c('Mean','Min','Max','SD','Missing','Largest SD')
colnames(wine.numbers) = wine.numbers.names
wine.numbers = round(wine.numbers,digits=2)
pander(wine.numbers)

library(lattice)
library(ggplot2)

wine.scaled = scale(wine)

st = stack(as.data.frame(wine.scaled[,2:7]))
ggplot(as.data.frame(st)) +
  geom_boxplot(aes(x = ind, y = values)) +
    theme(axis.text.x = element_text(angle=0)) +
      scale_x_discrete(name ="") + scale_y_continuous(name ="") +
         ggtitle("Figure 3 - Boxplots of Variables")

st = stack(as.data.frame(wine.scaled[,8:13]))
ggplot(as.data.frame(st)) +
```

```
  geom_boxplot(aes(x = ind, y = values)) +
    theme(axis.text.x = element_text(angle=0)) +
      scale_x_discrete(name ="") + scale_y_continuous(name ="") +
        ggtitle("Figure 4 - Boxplots of Variables (cont)")

wine.scaled.abs = abs(scale(wine))
wine.variable.outliers.indices = which(wine.scaled.abs[,-1] >= 3, arr.ind = TRUE)
wine.variable.outliers.indices[,2] = apply(wine.variable.outliers.indices,
                                    1,
                                    function(x) x[2] = wine.variable.names[x[2]+1])
wine.variable.outliers.table = table(wine.variable.outliers.indices[,2])
barchart(wine.variable.outliers.table, horizontal = FALSE,
        xlab="", ylab="Frequency",
        scales = list(x = list(rot = 45)))

outliers.withmultipleissues = sum(table(wine.variable.outliers.indices[,1]) > 1)

library(lattice)
density.plots = densityplot(~ Alcohol + Malic.Acid + Alcalinity + Magnesium + Total.Phenols + Flavanoids
+
                          Nonflavanoid.Phenols + Proanthocyanins + Color.Intensity + Hue + OD280.OD315 +
Proline,
                        data=wine, groups = Class, plot.points = FALSE, auto.key =
list(space="right",title="Class"),
                        scales= list(x="free",y="free"), xlab = '')

plot(density.plots)

#wine.correlations = cor(wine[, -1])

#wine.correlations.ordered = order.dendrogram(as.dendrogram(hclust(dist(wine.correlations))))
#levelplot(wine.correlations[wine.correlations.ordered,wine.correlations.ordered],
#         at = do.breaks(c(-1.01,1.01),20),
#         scales=list(x=list(rot=90)),
#         xlab='',ylab='',main='Variable Correlations')

library(corrplot)
corrplot(cor(wine[, wine.variable.names[-1]]),
```

```
      tl.col = "black", tl.cex = 0.8, tl.srt = 45,
      type="lower")

library(randomForest)
wine.rf = randomForest(Class~.,data=wine,mtry=4,importance=TRUE)
varImpPlot(wine.rf, main='Variable Importance from Random Forest')

library(rpart)
library(rpart.plot)
library(pander)
#library(rattle)
#fancyRpartPlot(rpart(wine$Class ~ ., data = wine), sub = "")
rpart.plot(rpart(wine$Class ~  .,  data  =  wine))

library(MASS)
# all predictors
wine.lda = lda(Class ~ .,data = as.data.frame(wine))
plot(wine.lda, col=wine$Class)

wine.wo.class = wine[,-1]
wine.pcr = prcomp(wine.wo.class, scale = T)
#biplot(wine.pcr, xlabs = wine[, "Class"])
#wine$Class = as.factor(wine$Class)


library(ggplot2)
PCbiplot <- function(dat, PC, x="PC1", y="PC2", colors=c('black', 'black', 'blue', 'lightblue')) {
  # PC being a prcomp object
  data <- data.frame(obsnames=wine$Class, PC$x)
  plot <- ggplot(data, aes_string(x=x, y=y)) + geom_text(alpha=.5, size=3, aes(label=obsnames),
color=data$obsnames)
  plot <- plot + geom_hline(aes(0), size=.1, yintercept = 0) + geom_vline(aes(0), size=.1, color=colors[2],
xintercept = 0)
  datapc <- data.frame(varnames=rownames(PC$rotation), PC$rotation)
  mult <- min(
      (max(data[,y]) - min(data[,y])/(max(datapc[,y])-min(datapc[,y]))),
      (max(data[,x]) - min(data[,x])/(max(datapc[,x])-min(datapc[,x])))
      )
  datapc <- transform(datapc,
          v1 = .7 * mult * (get(x)),
```

```
        v2 = .7 * mult * (get(y))
        )
  plot <- plot + coord_equal() + geom_text(data=datapc, aes(x=v1, y=v2, label=varnames), size = 3,
vjust=1, color=colors[3])
  plot <- plot + geom_segment(data=datapc, aes(x=0, y=0, xend=v1, yend=v2),
arrow=arrow(length=unit(0.2,"cm")), alpha=0.75, color=colors[4])
  plot
}

PCbiplot(wine, wine.pcr)
```