Wine Data Analysis

Tate Welty

Data from UCI Machine learning Repository

```
library(SuperLearner)  #used for advanced modeling
```

Load in Libraries

```
## Loading required package: nnls
```

```
## Loading required package: gam
```

```
## Warning: package ’gam’ was built under R version 4.0.5
```

```
## Loading required package: splines
```

```
## Loading required package: foreach
```

```
## Loaded gam 1.20
```

```
## Super Learner
```

```
## Version: 2.0-28
```

```
## Package created on 2021-05-04
```

```
library(car)  #allows for Variance Inflation factors
```

```
## Warning: package ’car’ was built under R version 4.0.5
```

```
## Loading required package: carData
```

```
#getwd()  #gets working directory.  Put files in here
wine_red<-read.csv(’winequality-red.csv’, sep=";")
wine_white<-read.csv(’winequality-white.csv’, sep=";")

#create a combined dataframe
wine_all<-rbind(wine_red,wine_white)
wine_all=cbind(c(rep(1,dim(wine_red)[1]),rep(0,dim(wine_white)[1])),wine_all)
colnames(wine_all)[1]<-’Red Wine’
```

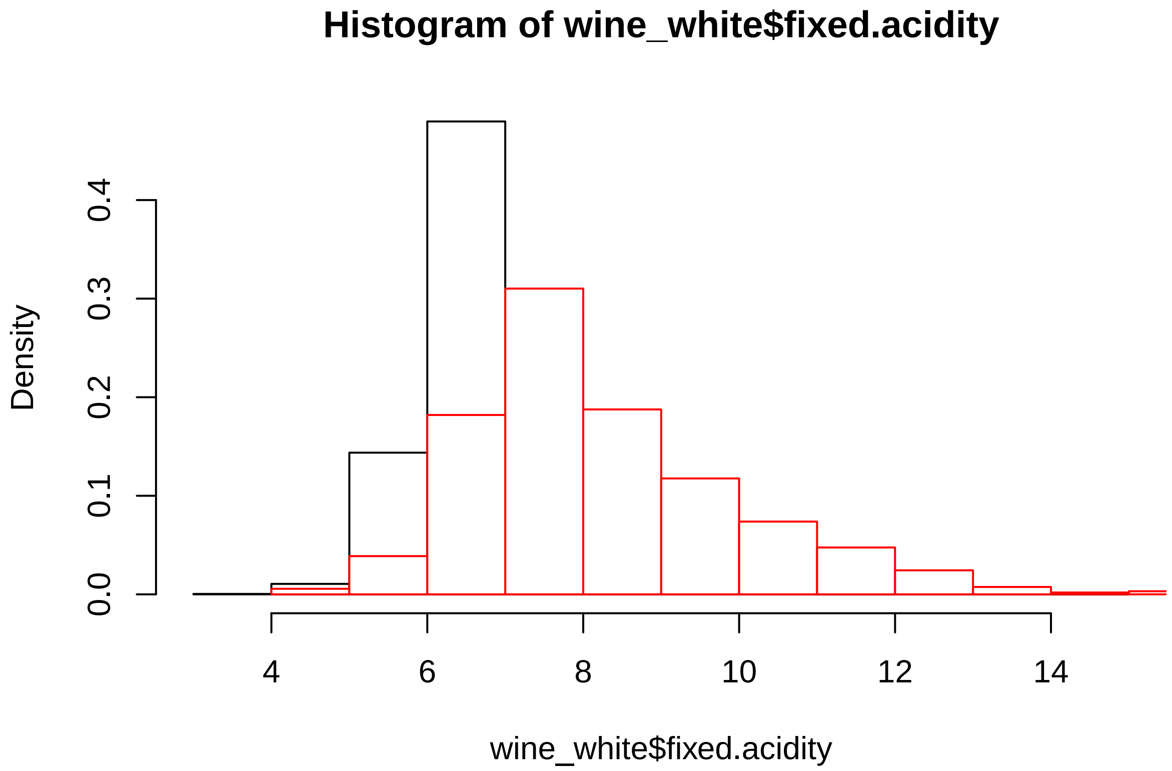
Importing Data

```
names(wine_white)
```

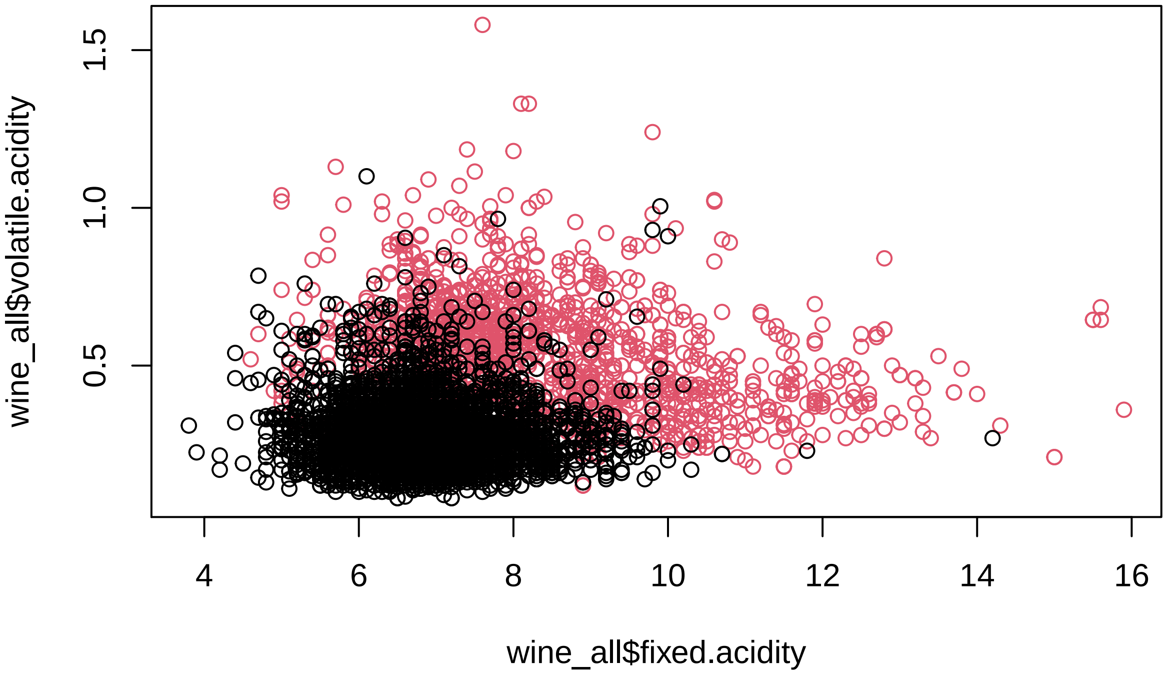
Looking at data

```
##  [1] "fixed.acidity"        "volatile.acidity"     "citric.acid"
##  [4] "residual.sugar"       "chlorides"            "free.sulfur.dioxide"
##  [7] "total.sulfur.dioxide" "density"              "pH"
## [10] "sulphates"            "alcohol"              "quality"
```

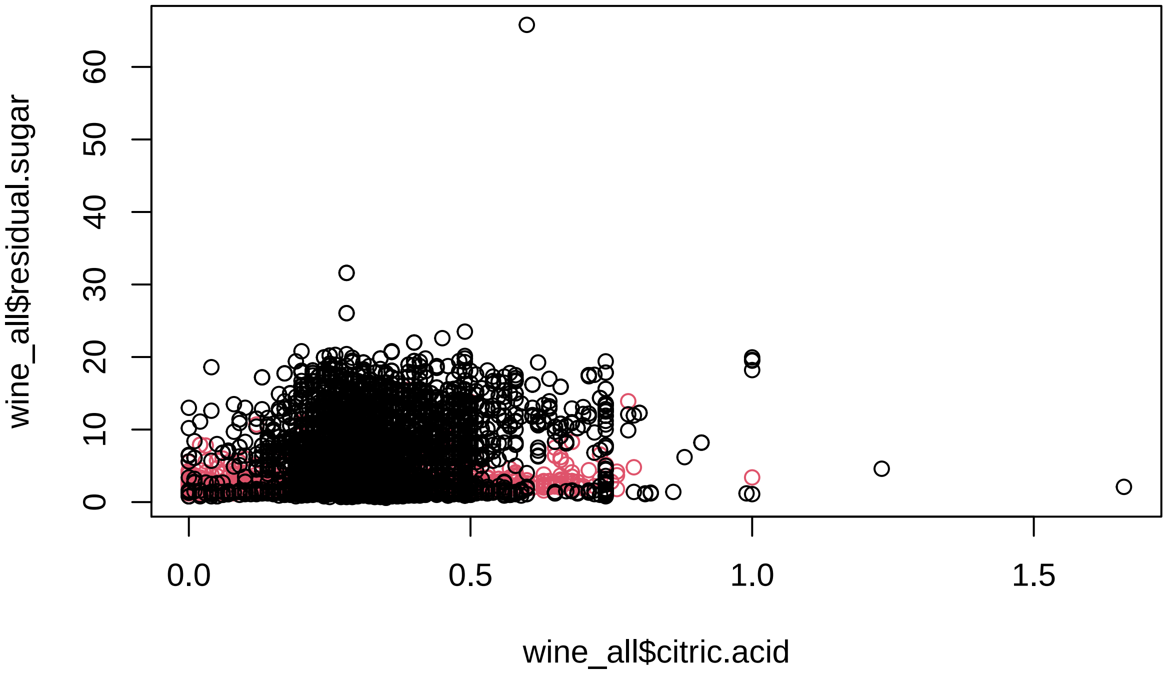
```
hist(wine_white$fixed.acidity, freq=F, col=’white’, border=’black’)
hist(wine_red$fixed.acidity, freq=F, col=’white’, border = ’red’, add=T)
```



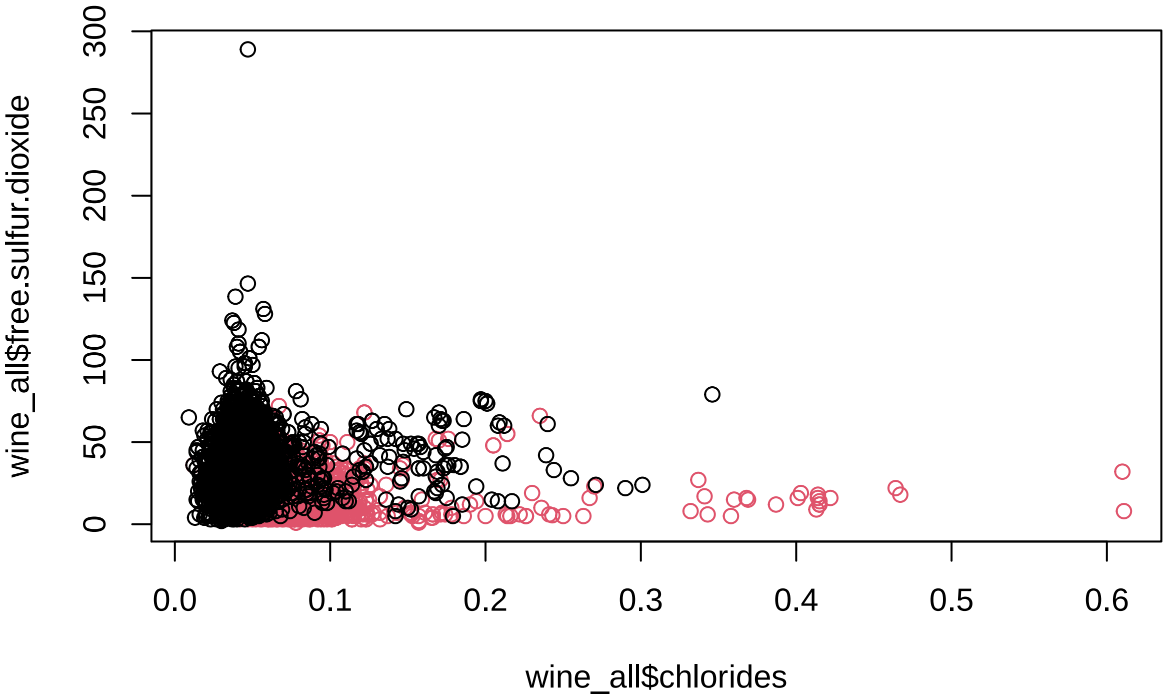
```
plot(wine_all$fixed.acidity,wine_all$volatile.acidity, col=(wine_all$‘Red Wine‘+1))
```



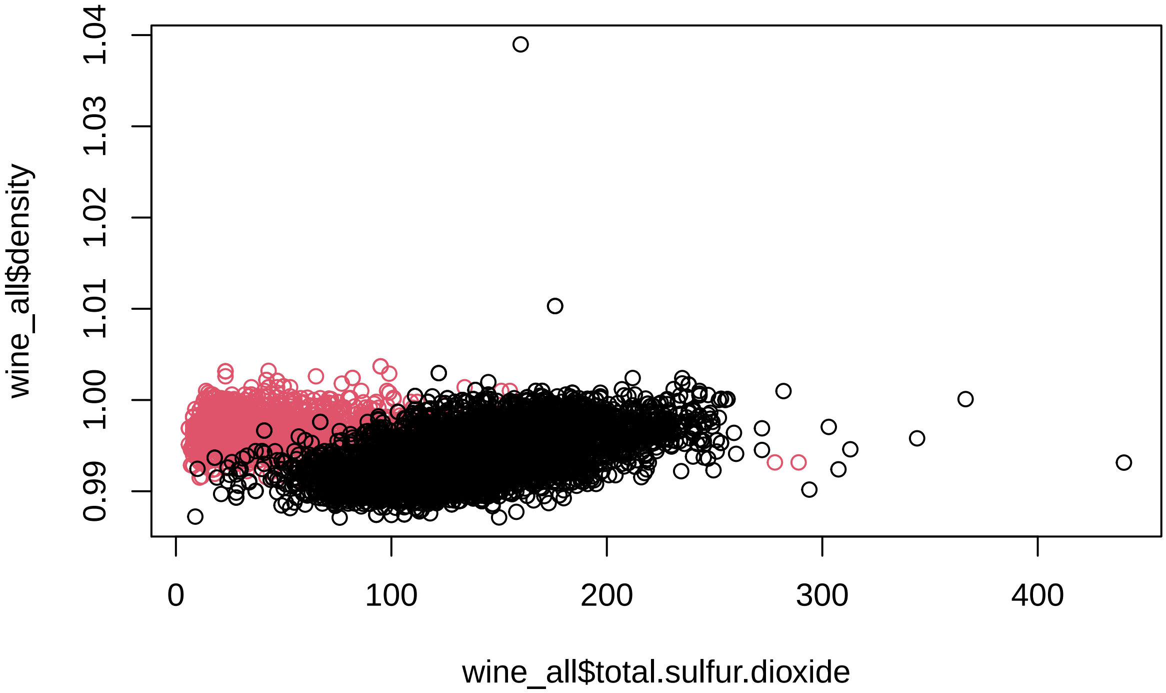
```
plot(wine_all$citric.acid,wine_all$residual.sugar, col=(wine_all$`Red Wine`+1))
```



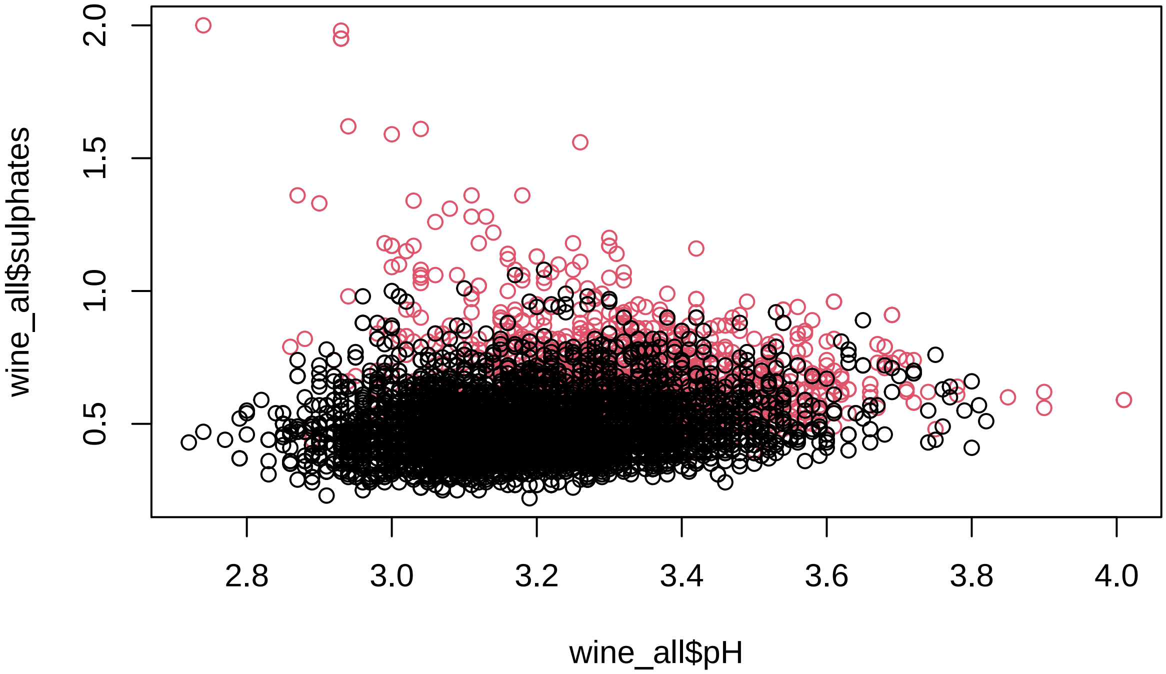
```
plot(wine_all$chlorides,wine_all$free.sulfur.dioxide, col=(wine_all$`Red Wine`+1))
```



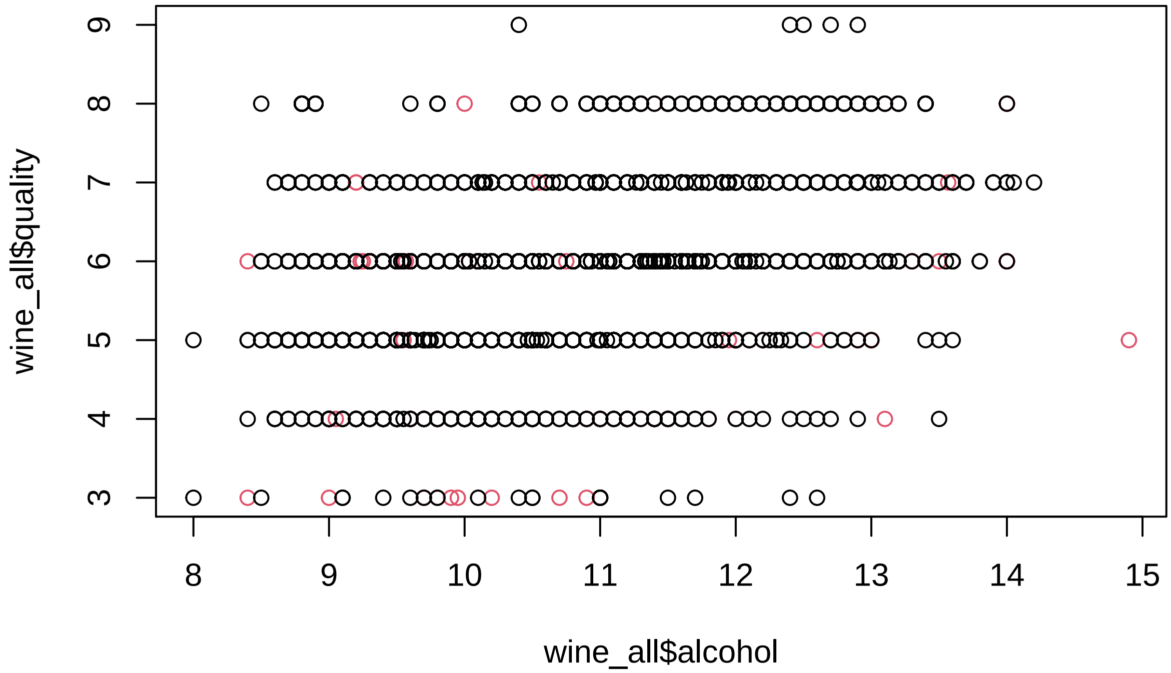
```
plot(wine_all$total.sulfur.dioxide,wine_all$density, col=(wine_all$`Red Wine`+1))
```



```
plot(wine_all$pH,wine_all$sulphates, col=(wine_all$`Red Wine`+1))
```



```
plot(wine_all$alcohol,wine_all$quality, col=(wine_all$`Red Wine`+1))
```



As we can see above, red wine has less total sulfur dioxide than white wine does.

```
model1<-lm(wine_all$`Red Wine`~.,data=wine_all)
summary(model1)
```

Variable significance

```
##
## Call:
## lm(formula = wine_all$`Red Wine` ~ ., data = wine_all)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.2778 -0.0908 -0.0045  0.0831  1.4417
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          -1.355e+02  2.591e+00 -52.310  < 2e-16 ***
## fixed.acidity        -4.927e-02  3.392e-03 -14.526  < 2e-16 ***
## volatile.acidity      4.781e-01  1.720e-02  27.791  < 2e-16 ***
## citric.acid          -1.283e-01  1.732e-02  -7.408 1.45e-13 ***
## residual.sugar       -5.300e-02  1.127e-03 -47.007  < 2e-16 ***
## chlorides             7.653e-01  7.236e-02  10.576  < 2e-16 ***
## free.sulfur.dioxide   2.756e-03  1.642e-04  16.788  < 2e-16 ***
## total.sulfur.dioxide -2.942e-03  6.054e-05 -48.605  < 2e-16 ***
## density               1.364e+02  2.644e+00  51.591  < 2e-16 ***
## pH                   -1.721e-01  1.969e-02  -8.741  < 2e-16 ***
## sulphates             1.155e-01  1.668e-02   6.923 4.86e-12 ***
## alcohol               1.182e-01  3.709e-03  31.865  < 2e-16 ***
## quality               1.719e-02  2.701e-03   6.367 2.06e-10 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1599 on 6484 degrees of freedom
## Multiple R-squared:  0.8624, Adjusted R-squared:  0.8622
## F-statistic:  3388 on 12 and 6484 DF,  p-value: < 2.2e-16
```

For a linear model like the one above, all variables appear to be significant at all confidence levels. This means we have good predictors.

```
vif(model1)
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##             4.911189             2.037955             1.608690
##       residual.sugar            chlorides  free.sulfur.dioxide
##             7.308546             1.632490             2.156281
## total.sulfur.dioxide              density                   pH
##             2.974040            15.964831             2.545764
##            sulphates              alcohol              quality
##             1.565737             4.970044             1.412703
```

Density has a VIF above 10, so there is potential collinearity there, but otherwise looks good.

```


#set testing and training data
set.seed(123)

train_obs=sample(nrow(wine_all), .75*nrow(wine_all))

x_train=wine_all[train_obs,]

x_test=wine_all[-train_obs,]

y_train=as.numeric(wine_all$‘Red Wine‘[train_obs])

y_test=as.numeric(wine_all$‘Red Wine‘[-train_obs])
```

```


model_lm1<-lm(wine_all$‘Red Wine‘~.,data=wine_all)
pred = predict(model_lm1, x_test, onlySL = TRUE)
prediction_lm1=ifelse(pred>.5,1,0)
table(prediction_lm1,y_test)
```

Models

```
##                 y_test
## prediction_lm1    0    1
##               0 1232    8
##               1    0  385
```

Above is a table of linear model. The left has our prediction, and the top has the actual values. As we can see we have 8 datapoints that we misinterpret. This is out of 1625 which is very good.

```
#use superlearner to do the same thing
model_lm = SuperLearner(Y = y_train,
                               X = x_train,
                               family = binomial(),
                               SL.library = c('SL.lm'))
```

```
pred = predict(model_lm, x_test, onlySL = TRUE)
pred_binary=ifelse(pred$pred>.5,1,0)
table(pred_binary,y_test)
```

```
##            y_test
## pred_binary    0    1
##           0 1232    0
##           1    0  393
```

Now we have a linear model, but utilizing SuperLearner. Superlearner does a lot of additional optimizations with the datasets. We can see above that there are 0 datapoints misidentified.