Analysis of Wine Quality Data Set

Introduction

This project uses the wine quality dataset from the UCI ML repository linked here. The dataset characterizes the relationship between wine quality and some chemical characteristics. I use regression model selection techniques to understand which properites have the greatest influence on quality measured by ratings from 0 to 10. I look at the data for red and white wines separately and then combine them for a visualization of principle components.

Data Summary

First I explore the data using some descriptive statistics, a group of boxplots of each attribute over all quality scores, pairwise scatterplots over each attribute and correlations over all variables.

Red Wine

```
redDat <- read.table("Datasets/winequality-red.csv",header=TRUE,sep=";")
#summarize
dim(redDat)
```

```
## [1] 1599   12
```

```
summary(redDat)
```

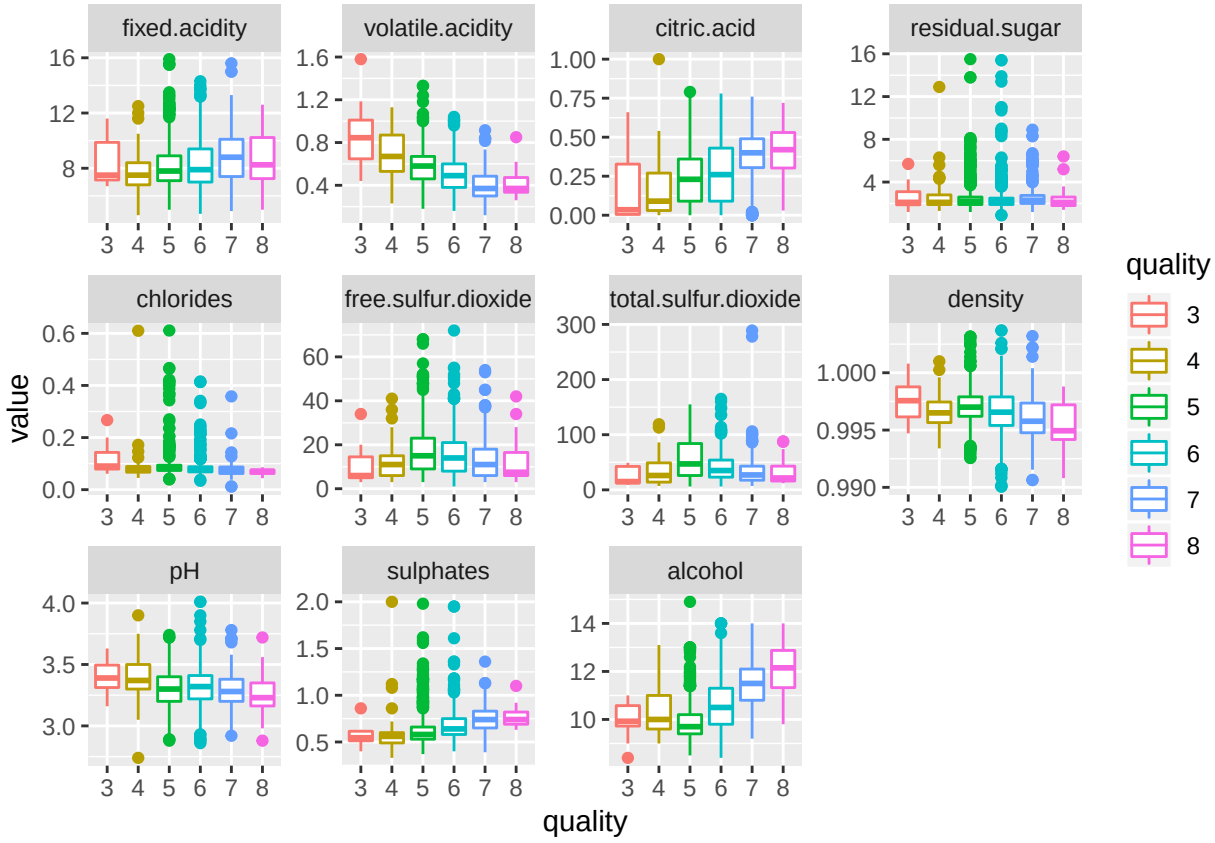
```
##  fixed.acidity   volatile.acidity  citric.acid    residual.sugar
##  Min.   : 4.60   Min.   :0.1200   Min.   :0.000   Min.   : 0.900
##  1st Qu.: 7.10   1st Qu.:0.3900   1st Qu.:0.090   1st Qu.: 1.900
##  Median : 7.90   Median :0.5200   Median :0.260   Median : 2.200
##  Mean   : 8.32   Mean   :0.5278   Mean   :0.271   Mean   : 2.539
##  3rd Qu.: 9.20   3rd Qu.:0.6400   3rd Qu.:0.420   3rd Qu.: 2.600
##  Max.   :15.90   Max.   :1.5800   Max.   :1.000   Max.   :15.500
##    chlorides       free.sulfur.dioxide total.sulfur.dioxide    density
##  Min.   :0.01200   Min.   : 1.00       Min.   :  6.00       Min.   :0.9901
##  1st Qu.:0.07000   1st Qu.: 7.00       1st Qu.: 22.00       1st Qu.:0.9956
##  Median :0.07900   Median :14.00       Median : 38.00       Median :0.9968
##  Mean   :0.08747   Mean   :15.87       Mean   : 46.47       Mean   :0.9967
##  3rd Qu.:0.09000   3rd Qu.:21.00       3rd Qu.: 62.00       3rd Qu.:0.9978
##  Max.   :0.61100   Max.   :72.00       Max.   :289.00       Max.   :1.0037
##        pH          sulphates         alcohol         quality
##  Min.   :2.740   Min.   :0.3300   Min.   : 8.40   Min.   :3.000
##  1st Qu.:3.210   1st Qu.:0.5500   1st Qu.: 9.50   1st Qu.:5.000
##  Median :3.310   Median :0.6200   Median :10.20   Median :6.000
##  Mean   :3.311   Mean   :0.6581   Mean   :10.42   Mean   :5.636
##  3rd Qu.:3.400   3rd Qu.:0.7300   3rd Qu.:11.10   3rd Qu.:6.000
##  Max.   :4.010   Max.   :2.0000   Max.   :14.90   Max.   :8.000
```

For the red wines there are 1,599 rows, 11 predictors and the quality score given by tasters between 1 and 10. The quality field's IQR is between 5 and 6, so most of the observations are within a narrow range. The

predictor measurements are distributed differently so the data should be normalized before performing any regularization.

```
#boxplots
redDatfact=redDat
redDatfact$quality=factor(redDatfact$quality)
ggplot(melt(redDatfact),aes(x=quality,y=value,colour=quality)) + geom_boxplot() + facet_wrap(~variable,
```

```
## Using quality as id variables
```

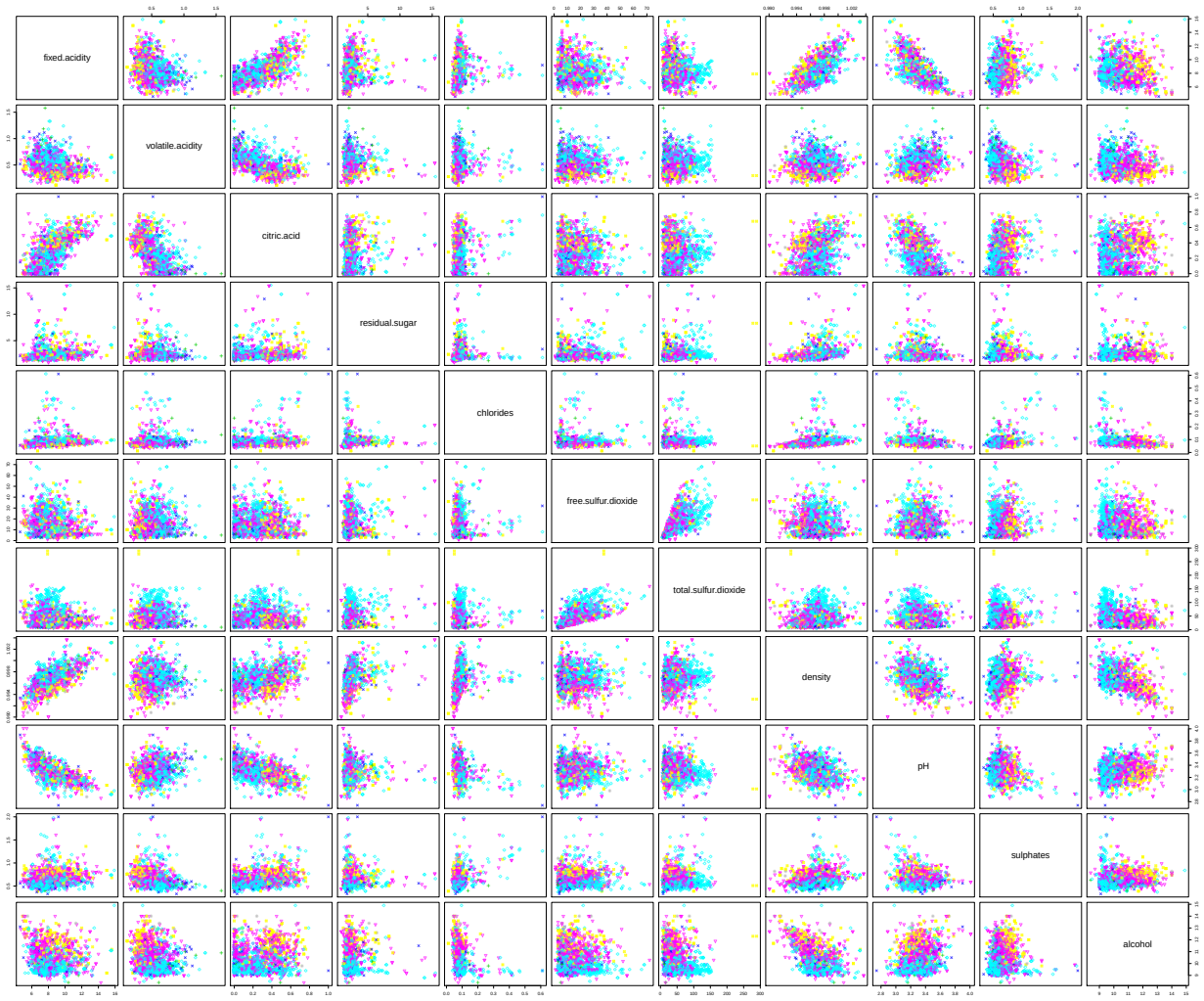



```
summary(redDatfact)[,"quality"]
```

```
##
## "3: 10   " "4: 53   " "5:681   " "6:638   " "7:199   " "8: 18   "
```

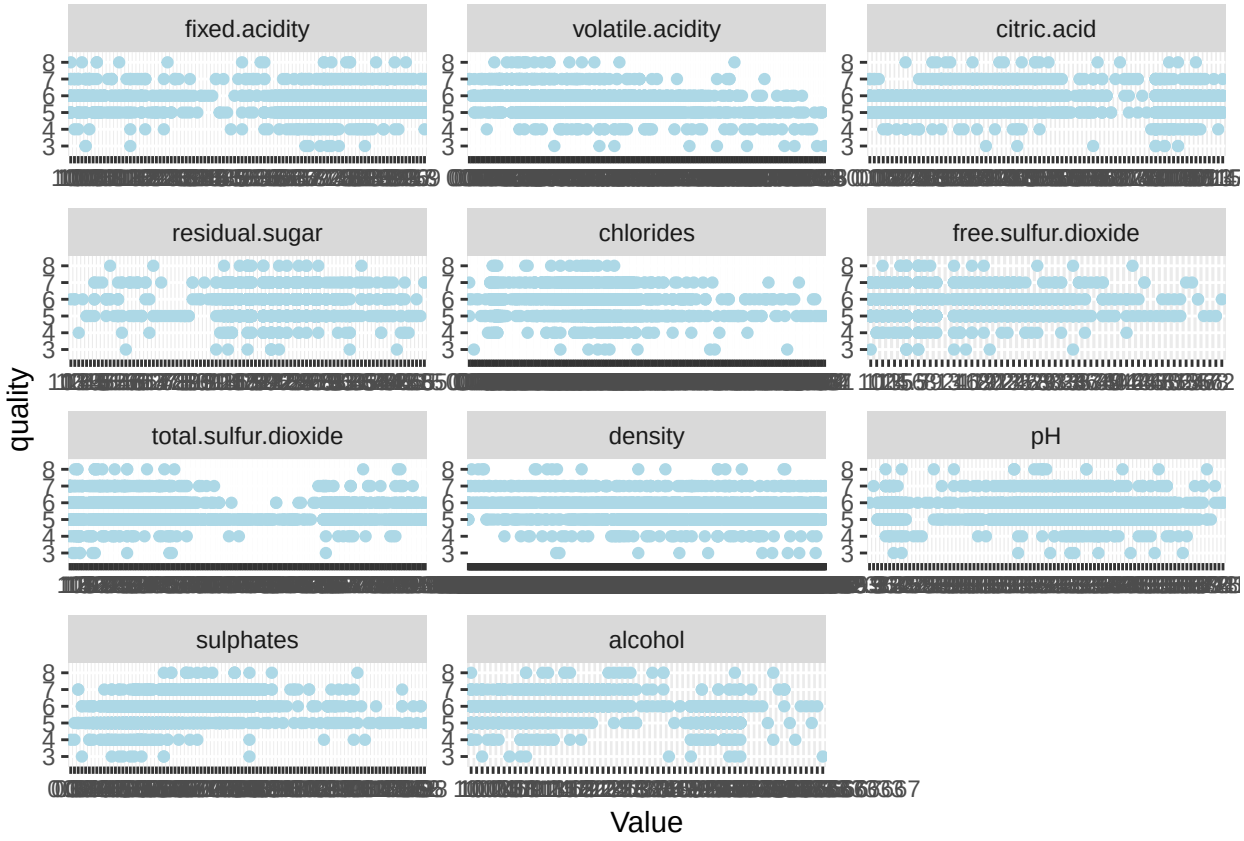
In the boxplots for each quality score; volatile acidity, citric acid, and alcohol seem to have the least amount of overlap between levels of rated wine quality suggesting that these will be good predictors. However one needs be cautious of overfitting at the lower and upper quality scores because of the much lower number of observations in this range.

```
#scatterplots
pairs(redDat[,-grep("quality",colnames(redDat))],col=alpha(redDat$quality,.6),pch=redDat$quality)
```



In these scatterplots, there seesm to be relationships between fixed acidity and density, fixed acidity and pH, fixed acidity and citric acid, citric acid and pH, citric acid and volatile acidity, residual sugar and density, free sulfur dioxide and total sulful dioxide, pH and density, and alcohol and density. Residual sugar, chlorides and sulphates all have data concentrated at the low end of the scatterplots. It may be worth trying a transformation of the data for these terms. For clustering of quality ratings, it's interesting that total sulfur dioxide has some very high values for quality 7. Other than that there does appear to be clustering but it's hard to judge if any two predictors are superior to others because there is so much overlap. Possibly some combination of free sulfur dioxide, total sulfur dioxide and alcohol will be effective.

```
#scatter with qual
pred=numeric;redDatScat=data.frame()
for ( pred in 1:(dim(redDat)[2]-1) ) {
redDatScat=rbind(redDatScat,cbind(colnames(redDat)[pred],redDat[,pred],redDat[,12]))
}
colnames(redDatScat)=c("Variable","Value","quality")
ggplot(redDatScat,aes(x=Value,y=quality)) + geom_point(colour="lightblue") + facet_wrap(~Variable,nrow=
```



When considering the scatterplots of each predictor against the outcome it is hard to draw any conclusions. It's possible that the large quantity of wines ranked in the 5 and 6 range obscures the signal. There are no obvious relationships and no strong indication of linearity or nonlinearity. There does seem to be a possible negative relationship with volatile acidity and a positive relationship with sulphates. There also appears to be some clusters at low end and high ends of residual sugar, fixed acidity, total sulfur dioxide, alcohol and others.

```
#correlations
signif(cor(redDat,method="pearson"),3) #linear
```

```
##                       fixed.acidity volatile.acidity citric.acid residual.sugar
## fixed.acidity               1.0000         -0.25600      0.6720        0.11500
## volatile.acidity           -0.2560          1.00000     -0.5520        0.00192
## citric.acid                 0.6720         -0.55200      1.0000        0.14400
## residual.sugar              0.1150          0.00192      0.1440        1.00000
## chlorides                   0.0937          0.06130      0.2040        0.05560
## free.sulfur.dioxide        -0.1540         -0.01050     -0.0610        0.18700
## total.sulfur.dioxide       -0.1130          0.07650      0.0355        0.20300
## density                     0.6680          0.02200      0.3650        0.35500
## pH                         -0.6830          0.23500     -0.5420       -0.08570
## sulphates                   0.1830         -0.26100      0.3130        0.00553
## alcohol                    -0.0617         -0.20200      0.1100        0.04210
## quality                     0.1240         -0.39100      0.2260        0.01370
##                       chlorides free.sulfur.dioxide total.sulfur.dioxide density
## fixed.acidity           0.09370            -0.15400               -0.1130  0.6680
## volatile.acidity        0.06130            -0.01050                0.0765  0.0220
## citric.acid             0.20400            -0.06100                0.0355  0.3650
```

```
## residual.sugar         0.05560              0.18700               0.2030  0.3550
## chlorides              1.00000              0.00556               0.0474  0.2010
## free.sulfur.dioxide    0.00556              1.00000               0.6680 -0.0219
## total.sulfur.dioxide   0.04740              0.66800               1.0000  0.0713
## density                0.20100             -0.02190               0.0713  1.0000
## pH                    -0.26500              0.07040              -0.0665 -0.3420
## sulphates              0.37100              0.05170               0.0429  0.1490
## alcohol               -0.22100             -0.06940              -0.2060 -0.4960
## quality               -0.12900             -0.05070              -0.1850 -0.1750
##                           pH sulphates alcohol quality
## fixed.acidity        -0.6830   0.18300 -0.0617  0.1240
## volatile.acidity      0.2350  -0.26100 -0.2020 -0.3910
## citric.acid          -0.5420   0.31300  0.1100  0.2260
## residual.sugar       -0.0857   0.00553  0.0421  0.0137
## chlorides            -0.2650   0.37100 -0.2210 -0.1290
## free.sulfur.dioxide   0.0704   0.05170 -0.0694 -0.0507
## total.sulfur.dioxide -0.0665   0.04290 -0.2060 -0.1850
## density              -0.3420   0.14900 -0.4960 -0.1750
## pH                    1.0000  -0.19700  0.2060 -0.0577
## sulphates            -0.1970   1.00000  0.0936  0.2510
## alcohol               0.2060   0.09360  1.0000  0.4760
## quality              -0.0577   0.25100  0.4760  1.0000
```

```
signif(cor(redDat,method="spearman"),3) #ranked (monotonic)
```

```
##                       fixed.acidity volatile.acidity citric.acid residual.sugar
## fixed.acidity                1.0000          -0.2780      0.6620         0.2210
## volatile.acidity            -0.2780           1.0000     -0.6100         0.0324
## citric.acid                  0.6620          -0.6100      1.0000         0.1760
## residual.sugar               0.2210           0.0324      0.1760         1.0000
## chlorides                    0.2510           0.1590      0.1130         0.2130
## free.sulfur.dioxide         -0.1750           0.0212     -0.0765         0.0746
## total.sulfur.dioxide        -0.0884           0.0941      0.0094         0.1450
## density                      0.6230           0.0250      0.3520         0.4220
## pH                          -0.7070           0.2340     -0.5480        -0.0900
## sulphates                    0.2130          -0.3260      0.3310         0.0383
## alcohol                     -0.0666          -0.2250      0.0965         0.1170
## quality                      0.1140          -0.3810      0.2130         0.0320
##                       chlorides free.sulfur.dioxide total.sulfur.dioxide density
## fixed.acidity          0.251000           -0.175000            -0.088400  0.6230
## volatile.acidity       0.159000            0.021200             0.094100  0.0250
## citric.acid            0.113000           -0.076500             0.009400  0.3520
## residual.sugar         0.213000            0.074600             0.145000  0.4220
## chlorides              1.000000            0.000805             0.130000  0.4110
## free.sulfur.dioxide    0.000805            1.000000             0.790000 -0.0412
## total.sulfur.dioxide   0.130000            0.790000             1.000000  0.1290
## density                0.411000           -0.041200             0.129000  1.0000
## pH                    -0.234000            0.116000            -0.009840 -0.3120
## sulphates              0.020800            0.045900            -0.000504  0.1610
## alcohol               -0.285000           -0.081400            -0.258000 -0.4620
## quality               -0.190000           -0.056900            -0.197000 -0.1770
##                            pH sulphates alcohol quality
## fixed.acidity         -0.70700  0.213000 -0.0666  0.1140
## volatile.acidity       0.23400 -0.326000 -0.2250 -0.3810
## citric.acid           -0.54800  0.331000  0.0965  0.2130
```

```
## residual.sugar       -0.09000  0.038300  0.1170  0.0320
## chlorides            -0.23400  0.020800 -0.2850 -0.1900
## free.sulfur.dioxide   0.11600  0.045900 -0.0814 -0.0569
## total.sulfur.dioxide -0.00984 -0.000504 -0.2580 -0.1970
## density              -0.31200  0.161000 -0.4620 -0.1770
## pH                    1.00000 -0.080300  0.1800 -0.0437
## sulphates            -0.08030  1.000000  0.2070  0.3770
## alcohol               0.18000  0.207000  1.0000  0.4790
## quality              -0.04370  0.377000  0.4790  1.0000
```

The predictor with the greatest correlation with quality is alcohol which is positive. After that is volatile acidity and then sulphates which is negative. Residual sugar, free sulphur dioxide and pH have low correlations with quality. Alcohol has a fairly low correlation with sulphates, but sulphates and volatile acidity have a higher correlation with eachother. Using Spearman correlation yields a significantly higher measure for sulphates and quality suggesting nonlinearity. Alcohol and sulfates also have higher correlation with eachother using spearman correlation.

White Wine

```
whiteDat <- read.table("Datasets/winequality-white.csv",header=TRUE,sep=";")
#summarize
dim(whiteDat)
```

```
## [1] 4898   12
```

```
summary(whiteDat)
```

```
##  fixed.acidity    volatile.acidity  citric.acid     residual.sugar
##  Min.   : 3.800   Min.   :0.0800   Min.   :0.0000   Min.   : 0.600
##  1st Qu.: 6.300   1st Qu.:0.2100   1st Qu.:0.2700   1st Qu.: 1.700
##  Median : 6.800   Median :0.2600   Median :0.3200   Median : 5.200
##  Mean   : 6.855   Mean   :0.2782   Mean   :0.3342   Mean   : 6.391
##  3rd Qu.: 7.300   3rd Qu.:0.3200   3rd Qu.:0.3900   3rd Qu.: 9.900
##  Max.   :14.200   Max.   :1.1000   Max.   :1.6600   Max.   :65.800
##    chlorides       free.sulfur.dioxide total.sulfur.dioxide    density
##  Min.   :0.00900   Min.   :  2.00      Min.   :  9.0        Min.   :0.9871
##  1st Qu.:0.03600   1st Qu.: 23.00      1st Qu.:108.0        1st Qu.:0.9917
##  Median :0.04300   Median : 34.00      Median :134.0        Median :0.9937
##  Mean   :0.04577   Mean   : 35.31      Mean   :138.4        Mean   :0.9940
##  3rd Qu.:0.05000   3rd Qu.: 46.00      3rd Qu.:167.0        3rd Qu.:0.9961
##  Max.   :0.34600   Max.   :289.00      Max.   :440.0        Max.   :1.0390
##        pH          sulphates         alcohol         quality
##  Min.   :2.720   Min.   :0.2200   Min.   : 8.00   Min.   :3.000
##  1st Qu.:3.090   1st Qu.:0.4100   1st Qu.: 9.50   1st Qu.:5.000
##  Median :3.180   Median :0.4700   Median :10.40   Median :6.000
##  Mean   :3.188   Mean   :0.4898   Mean   :10.51   Mean   :5.878
##  3rd Qu.:3.280   3rd Qu.:0.5500   3rd Qu.:11.40   3rd Qu.:6.000
##  Max.   :3.820   Max.   :1.0800   Max.   :14.20   Max.   :9.000
```

```
head(whiteDat)
```

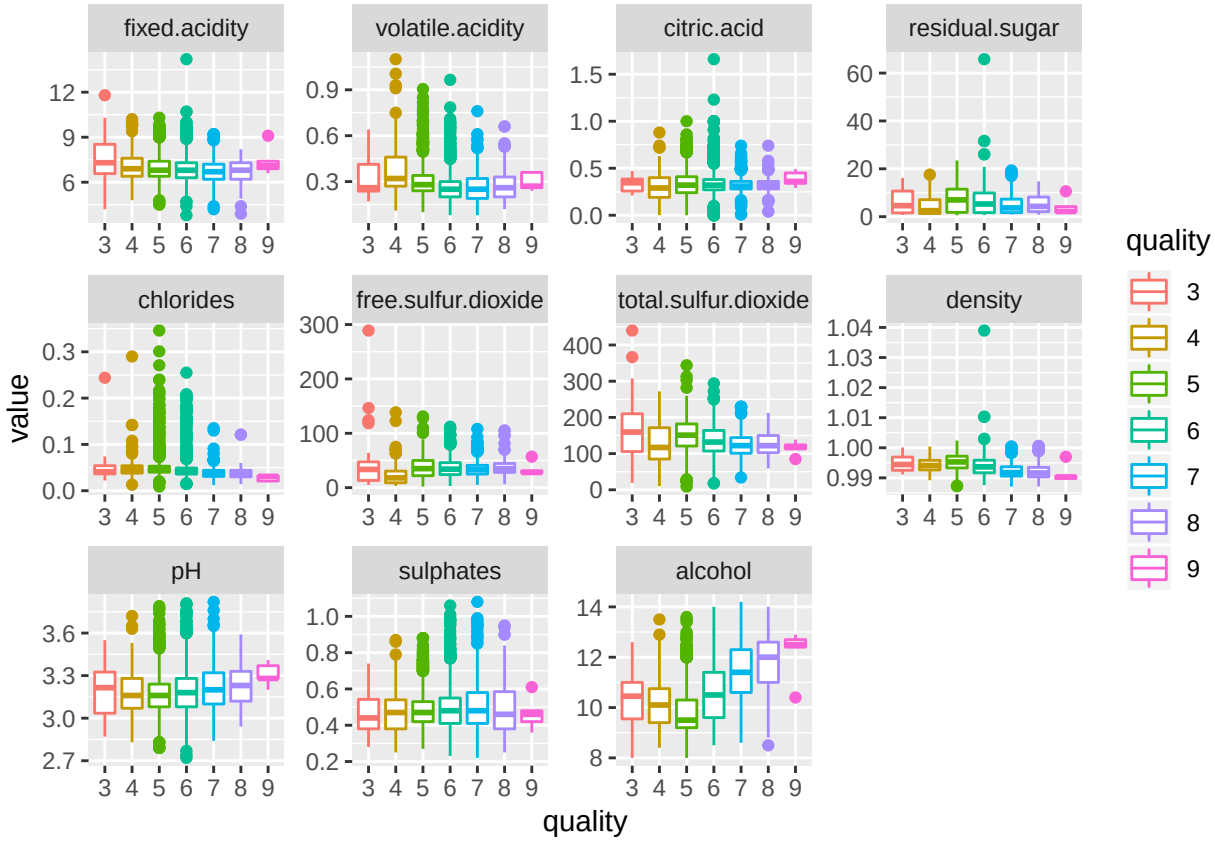
```
##   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1           7.0             0.27        0.36           20.7     0.045
## 2           6.3             0.30        0.34            1.6     0.049
## 3           8.1             0.28        0.40            6.9     0.050
## 4           7.2             0.23        0.32            8.5     0.058
```

```
## 5           7.2             0.23        0.32            8.5     0.058
## 6           8.1             0.28        0.40            6.9     0.050
##   free.sulfur.dioxide total.sulfur.dioxide density   pH sulphates alcohol
## 1                  45                  170  1.0010 3.00      0.45     8.8
## 2                  14                  132  0.9940 3.30      0.49     9.5
## 3                  30                   97  0.9951 3.26      0.44    10.1
## 4                  47                  186  0.9956 3.19      0.40     9.9
## 5                  47                  186  0.9956 3.19      0.40     9.9
## 6                  30                   97  0.9951 3.26      0.44    10.1
##   quality
## 1       6
## 2       6
## 3       6
## 4       6
## 5       6
## 6       6
```

The white wine dataset has the same predictors but larger number of observations. The quality IQR is also between 5 and 6.

```
#boxplots
whiteDatfact=whiteDat
whiteDatfact$quality=factor(whiteDatfact$quality)
ggplot(melt(whiteDatfact),aes(x=quality,y=value,colour=quality)) + geom_boxplot() + facet_wrap(~variable
```

```
## Using quality as id variables
```

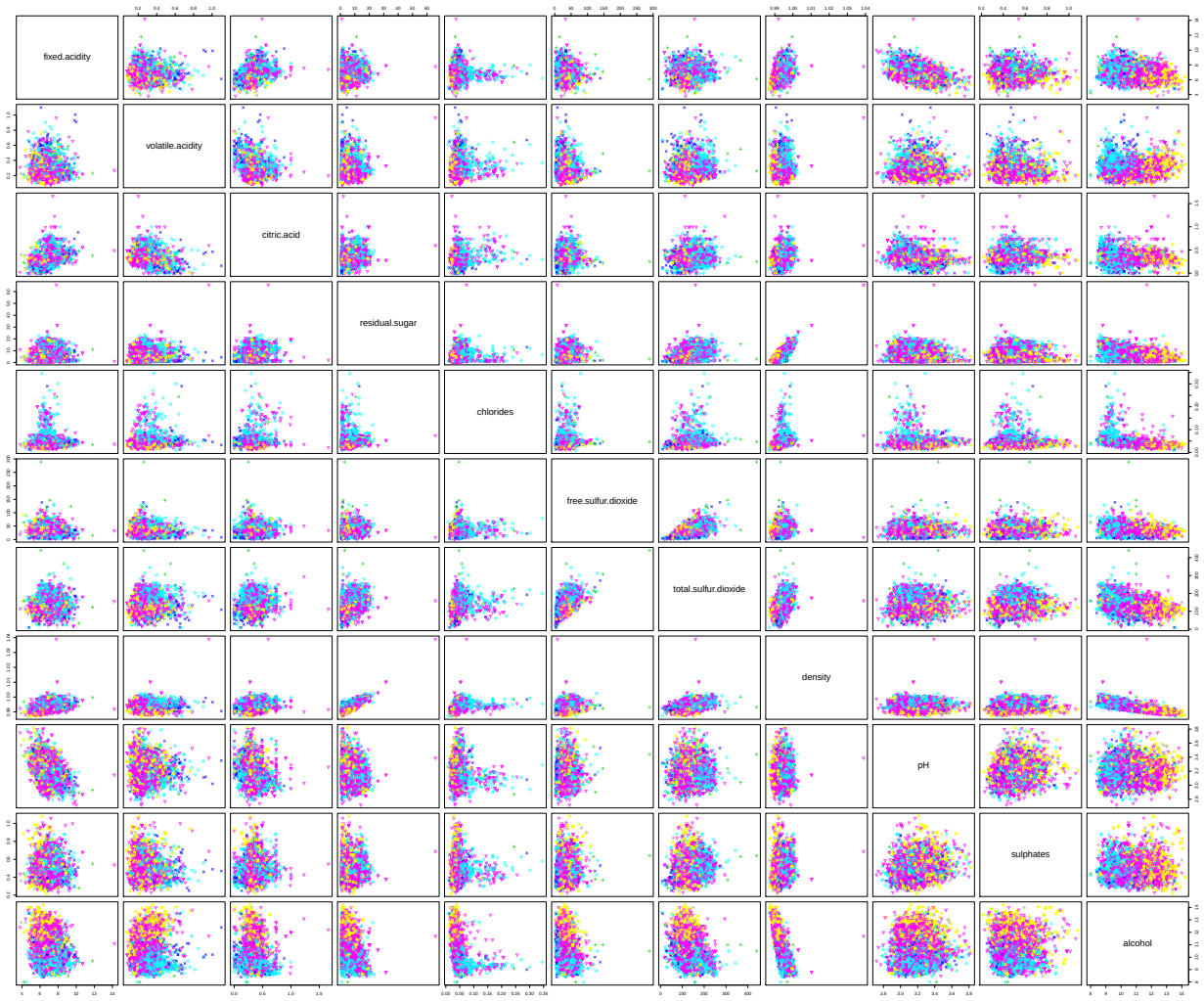


```
summary(whiteDatfact)[,"quality"]
```

```
##
## "3:  20  " "4: 163  " "5:1457  " "6:2198  " "7: 880  " "8: 175  " "9:   5  "
```

In the box plots sliced by quality scores, alcohol appears most distinct across the ratings. Density may also be a good predictor.

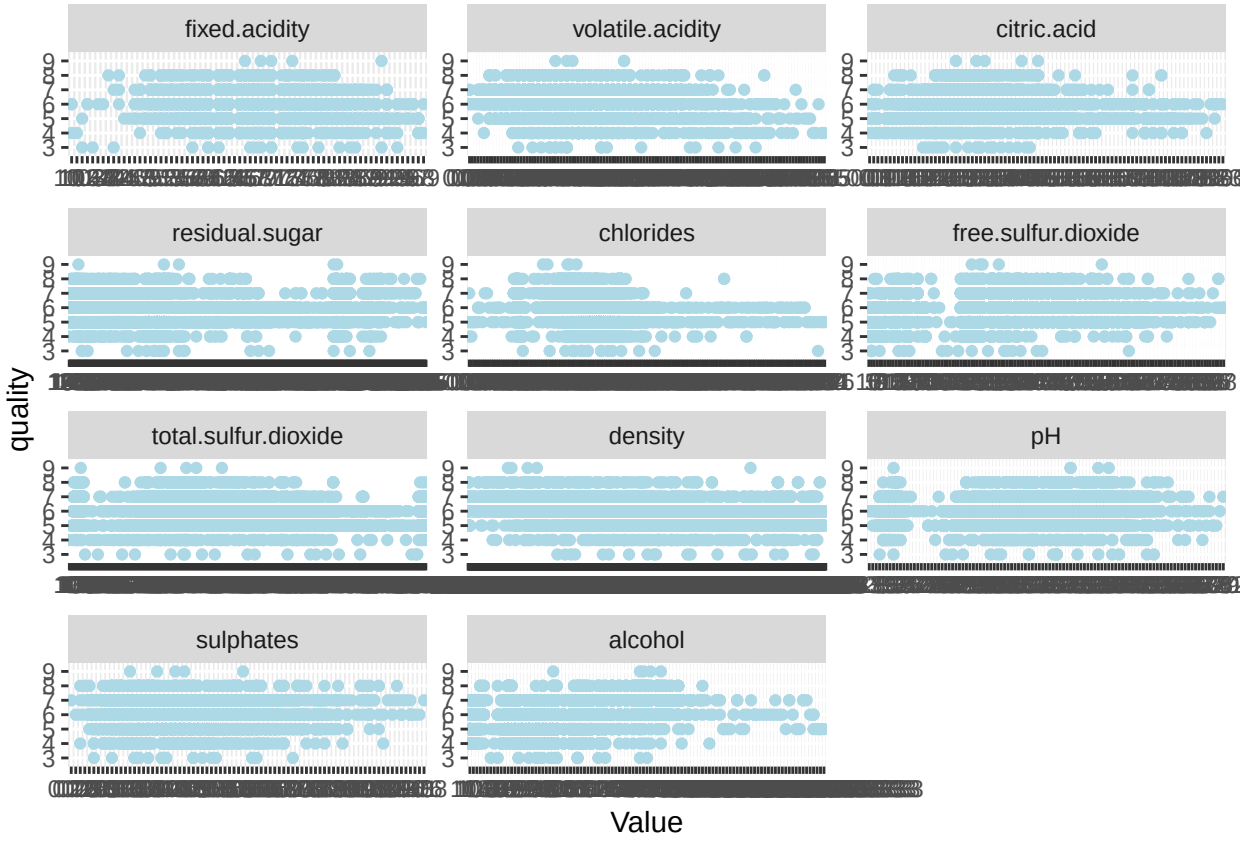
```
#scatterplots
pairs(whiteDat[,-grep("quality",colnames(whiteDat))],col=alpha(whiteDat$quality,.6),pch=whiteDat$quality
```



For the pairs scatterplot, fixed acidity and pH, residual sugar and density, free sulfur dioxide and total sulfur dioxide, density and fixed acidity, density and total sulfur dioxide, and density and alcohol seem to have a relationship. Alcohol seems to provide the best distinction of quality especially when paired with volatile acidity or density. Possibly alcohol with chlorides would be a good pair as well. Residual sugar has one large outlier, same with total sulfur dioxide and free sulfur dioxide. Density has two clear outliers. The chlorides data has a higher concentration at the lower end of the scatterplot. The higher variation of the y values at the lower x values suggests that log transformation may be useful. Although it seems more likely that there just happens to be a lot more data at the lower levels of x.

```
#scatter with qual
pred=numeric;whiteDatScat=data.frame()
```

```
for ( pred in 1:(dim(whiteDat)[2]-1) ) {
whiteDatScat=rbind(whiteDatScat,cbind(colnames(whiteDat)[pred],whiteDat[,pred],whiteDat[,12]))
}
colnames(whiteDatScat)=c("Variable","Value","quality")
ggplot(whiteDatScat,aes(x=Value,y=quality)) + geom_point(colour="lightblue") + facet_wrap(~Variable,nrow
```



Once again, it is hard to discern any meaningful relationships in the scatterplots of predictors compared to quality.

```
#correlations
signif(cor(whiteDat,method="pearson"),3) #linear
```

```
##                      fixed.acidity volatile.acidity citric.acid residual.sugar
## fixed.acidity               1.0000          -0.0227     0.28900         0.0890
## volatile.acidity           -0.0227           1.0000    -0.14900         0.0643
## citric.acid                 0.2890          -0.1490     1.00000         0.0942
## residual.sugar              0.0890           0.0643     0.09420         1.0000
## chlorides                   0.0231           0.0705     0.11400         0.0887
## free.sulfur.dioxide        -0.0494          -0.0970     0.09410         0.2990
## total.sulfur.dioxide        0.0911           0.0893     0.12100         0.4010
## density                     0.2650           0.0271     0.15000         0.8390
## pH                         -0.4260          -0.0319    -0.16400        -0.1940
## sulphates                  -0.0171          -0.0357     0.06230        -0.0267
## alcohol                    -0.1210           0.0677    -0.07570        -0.4510
## quality                    -0.1140          -0.1950    -0.00921        -0.0976
##                      chlorides free.sulfur.dioxide total.sulfur.dioxide density
## fixed.acidity           0.0231           -0.049400              0.09110  0.2650
```

```
## volatile.acidity        0.0705           -0.097000              0.08930  0.0271
## citric.acid             0.1140            0.094100              0.12100  0.1500
## residual.sugar          0.0887            0.299000              0.40100  0.8390
## chlorides               1.0000            0.101000              0.19900  0.2570
## free.sulfur.dioxide     0.1010            1.000000              0.61600  0.2940
## total.sulfur.dioxide    0.1990            0.616000              1.00000  0.5300
## density                 0.2570            0.294000              0.53000  1.0000
## pH                     -0.0904           -0.000618              0.00232 -0.0936
## sulphates               0.0168            0.059200              0.13500  0.0745
## alcohol                -0.3600           -0.250000             -0.44900 -0.7800
## quality                -0.2100            0.008160             -0.17500 -0.3070
##                              pH sulphates alcohol  quality
## fixed.acidity        -0.426000   -0.0171 -0.1210 -0.11400
## volatile.acidity     -0.031900   -0.0357  0.0677 -0.19500
## citric.acid          -0.164000    0.0623 -0.0757 -0.00921
## residual.sugar       -0.194000   -0.0267 -0.4510 -0.09760
## chlorides            -0.090400    0.0168 -0.3600 -0.21000
## free.sulfur.dioxide  -0.000618    0.0592 -0.2500  0.00816
## total.sulfur.dioxide  0.002320    0.1350 -0.4490 -0.17500
## density              -0.093600    0.0745 -0.7800 -0.30700
## pH                    1.000000    0.1560  0.1210  0.09940
## sulphates             0.156000    1.0000 -0.0174  0.05370
## alcohol               0.121000   -0.0174  1.0000  0.43600
## quality               0.099400    0.0537  0.4360  1.00000
```

```
signif(cor(whiteDat,method="spearman"),3) #ranked (monotonic)
```

```
##                      fixed.acidity volatile.acidity citric.acid residual.sugar
## fixed.acidity               1.0000         -0.04290      0.2980        0.10700
## volatile.acidity           -0.0429          1.00000     -0.1500        0.10900
## citric.acid                 0.2980         -0.15000      1.0000        0.02460
## residual.sugar              0.1070          0.10900      0.0246        1.00000
## chlorides                   0.0947         -0.00493      0.0327        0.22800
## free.sulfur.dioxide        -0.0245         -0.08120      0.0883        0.34600
## total.sulfur.dioxide        0.1130          0.11800      0.0932        0.43100
## density                     0.2700          0.01010      0.0914        0.78000
## pH                         -0.4180         -0.04520     -0.1460       -0.18000
## sulphates                  -0.0132         -0.01690      0.0798       -0.00384
## alcohol                    -0.1070          0.03400     -0.0292       -0.44500
## quality                    -0.0845         -0.19700      0.0183       -0.08210
##                      chlorides free.sulfur.dioxide total.sulfur.dioxide density
## fixed.acidity          0.09470            -0.02450               0.1130  0.2700
## volatile.acidity      -0.00493            -0.08120               0.1180  0.0101
## citric.acid            0.03270             0.08830               0.0932  0.0914
## residual.sugar         0.22800             0.34600               0.4310  0.7800
## chlorides              1.00000             0.16700               0.3750  0.5080
## free.sulfur.dioxide    0.16700             1.00000               0.6190  0.3280
## total.sulfur.dioxide   0.37500             0.61900               1.0000  0.5640
## density                0.50800             0.32800               0.5640  1.0000
## pH                    -0.05400            -0.00627              -0.0118 -0.1100
## sulphates              0.09390             0.05230               0.1580  0.0951
## alcohol               -0.57100            -0.27300              -0.4770 -0.8220
## quality               -0.31400             0.02370              -0.1970 -0.3480
##                           pH sulphates alcohol quality
## fixed.acidity        -0.41800  -0.01320 -0.1070 -0.0845
```

```
## volatile.acidity     -0.04520  -0.01690  0.0340 -0.1970
## citric.acid          -0.14600   0.07980 -0.0292  0.0183
## residual.sugar       -0.18000  -0.00384 -0.4450 -0.0821
## chlorides            -0.05400   0.09390 -0.5710 -0.3140
## free.sulfur.dioxide  -0.00627   0.05230 -0.2730  0.0237
## total.sulfur.dioxide -0.01180   0.15800 -0.4770 -0.1970
## density              -0.11000   0.09510 -0.8220 -0.3480
## pH                    1.00000   0.14000  0.1490  0.1090
## sulphates             0.14000   1.00000 -0.0449  0.0333
## alcohol               0.14900  -0.04490  1.0000  0.4400
## quality               0.10900   0.03330  0.4400  1.0000
```

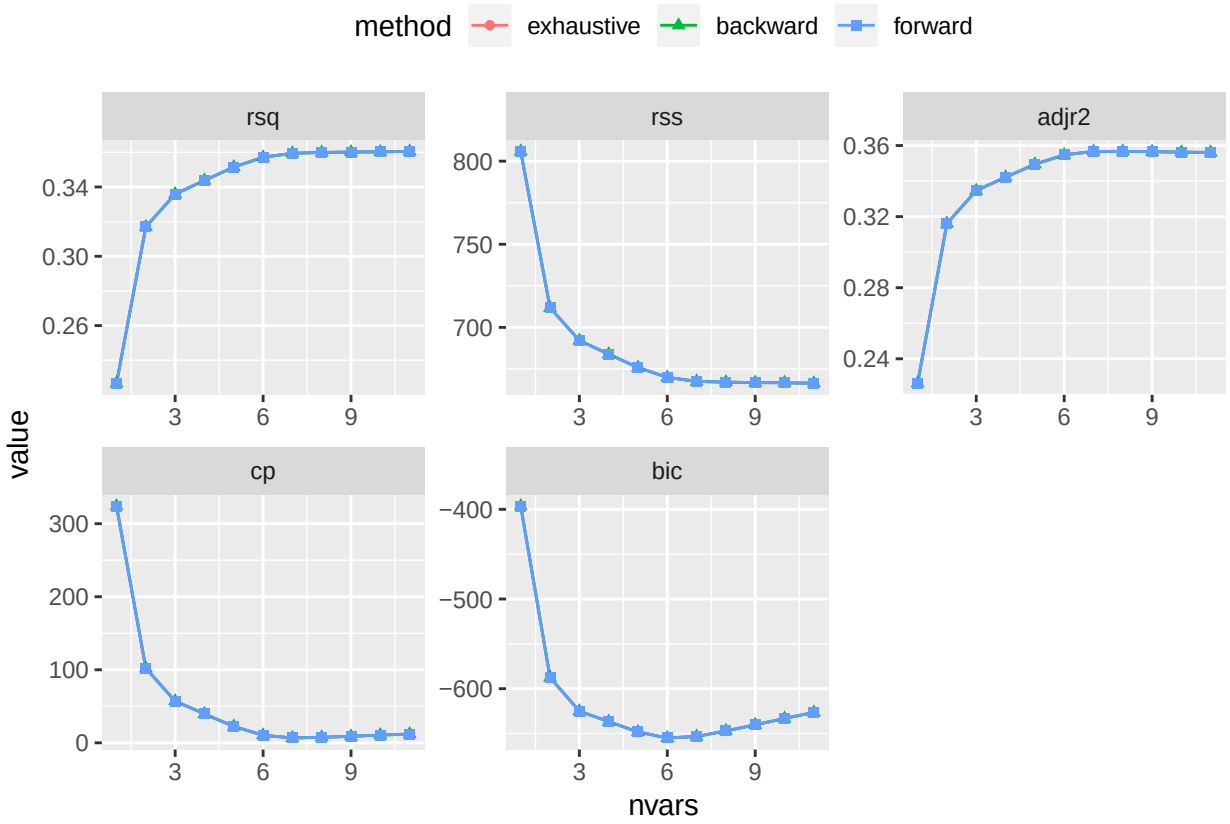
Alcohol is most correlated with quality. It has a positive correlation. the next highest is density which has a negative correlation with quality. After those two there is a drop off with chlorides and volatile acidity having greatest correlation. Citric acid and free sulfur dioxide have extremely low correlations with quality. Among themselves, alcohol and density have a strong negative correlation. Alcohol also correlates fairly strong with chlorides but weak with volatile acidity. There is a similar result with density which correlates more with chlorides but not much with volatile acidity. Using spearman correlation produces simlar results except with chlorides which correlates significantly more with quality suggesting nonlinearity.

Attribute Selection

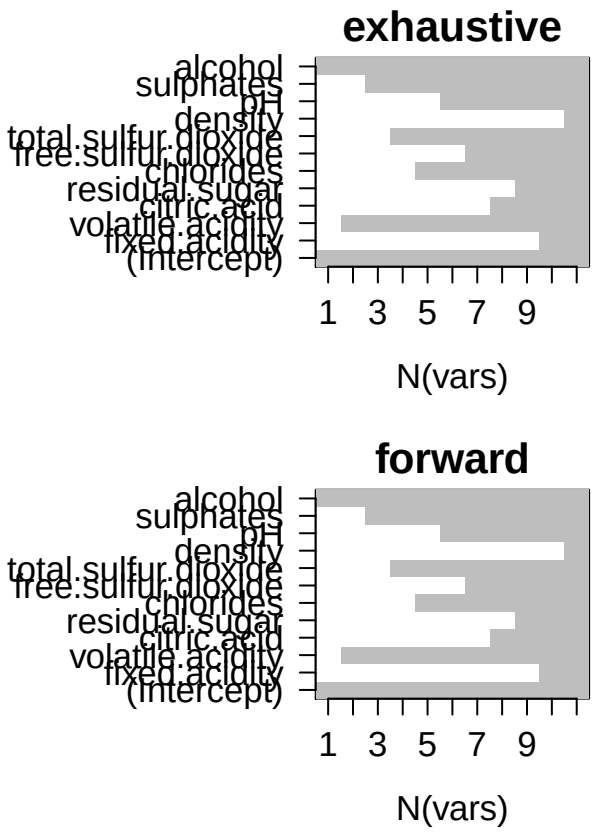
I use `regsubsets` from the `leaps` library to choose optimal sets of variables for modeling wine quality.

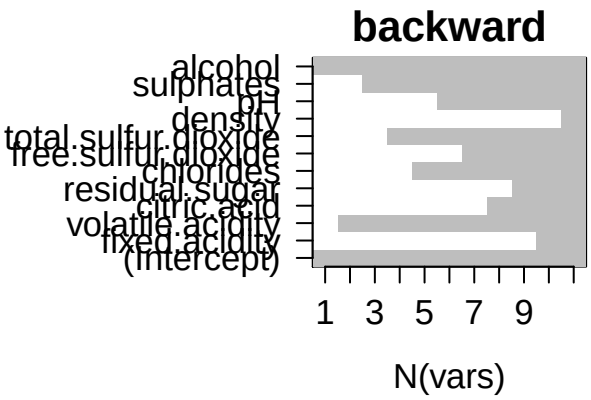
Red Wine

```
summaryMetrics <- NULL
whichAll <- list()
for ( myMthd in c("exhaustive", "backward", "forward") ) {
  rsRes <- regsubsets(quality~.,redDat,method=myMthd,nvmax=11)
  summRes <- summary(rsRes)
  whichAll[[myMthd]] <- summRes$which
  for ( metricName in c("rsq","rss","adjr2","cp","bic") ) {
    summaryMetrics <- rbind(summaryMetrics,
      data.frame(method=myMthd,metric=metricName,
                 nvars=1:length(summRes[[metricName]]),
                 value=summRes[[metricName]]))
  }
}
ggplot(summaryMetrics,aes(x=nvars,y=value,shape=method,colour=method)) + geom_path() + geom_point() + fa
```



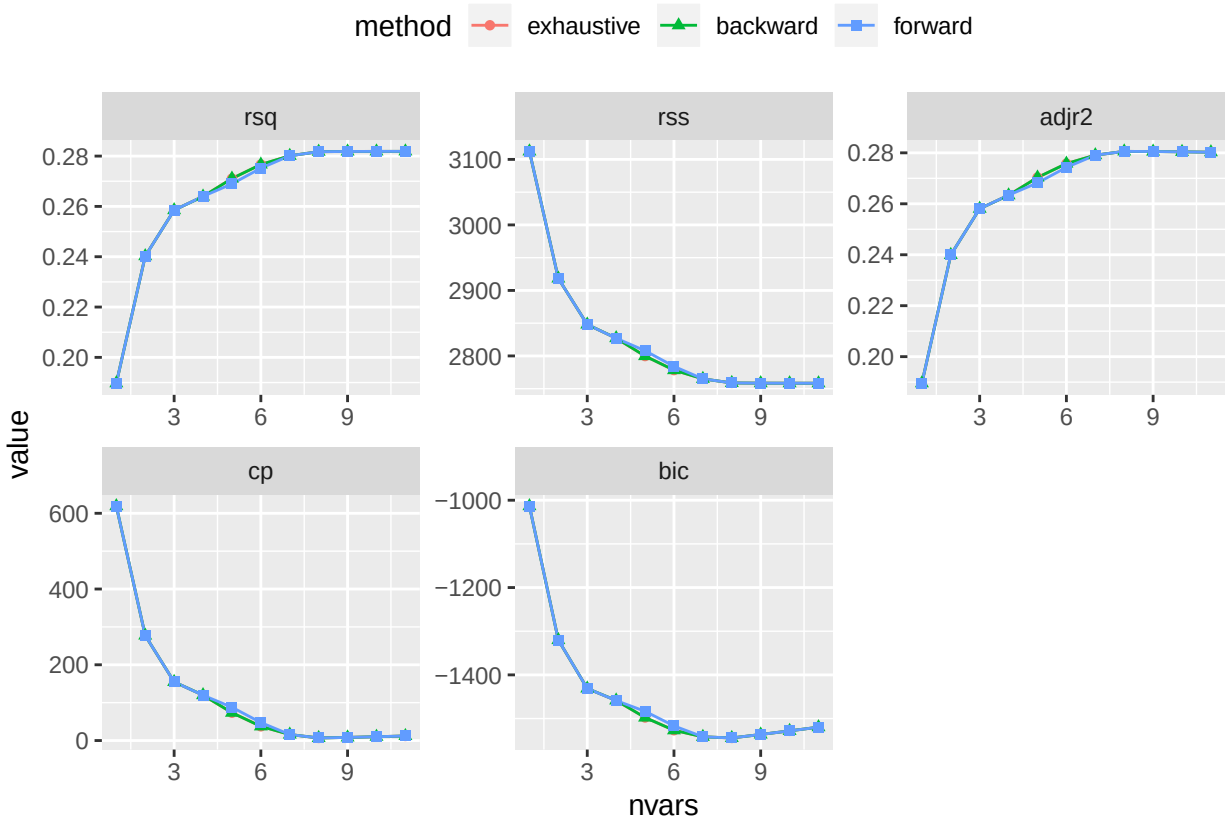

```
old.par <- par(mfrow=c(2,2),ps=16,mar=c(5,10,2,1))
for ( myMthd in names(whichAll) ) {
  image(1:nrow(whichAll[[myMthd]]),
        1:ncol(whichAll[[myMthd]]),
        whichAll[[myMthd]],xlab="N(vars)",ylab="",
        xaxt="n",yaxt="n",breaks=c(-0.5,0.5,1.5),
        col=c("white","gray"),main=myMthd)
  axis(1,1:nrow(whichAll[[myMthd]]),rownames(whichAll[[myMthd]]))
  axis(2,1:ncol(whichAll[[myMthd]]),colnames(whichAll[[myMthd]]),las=2)
}
par(old.par)
```



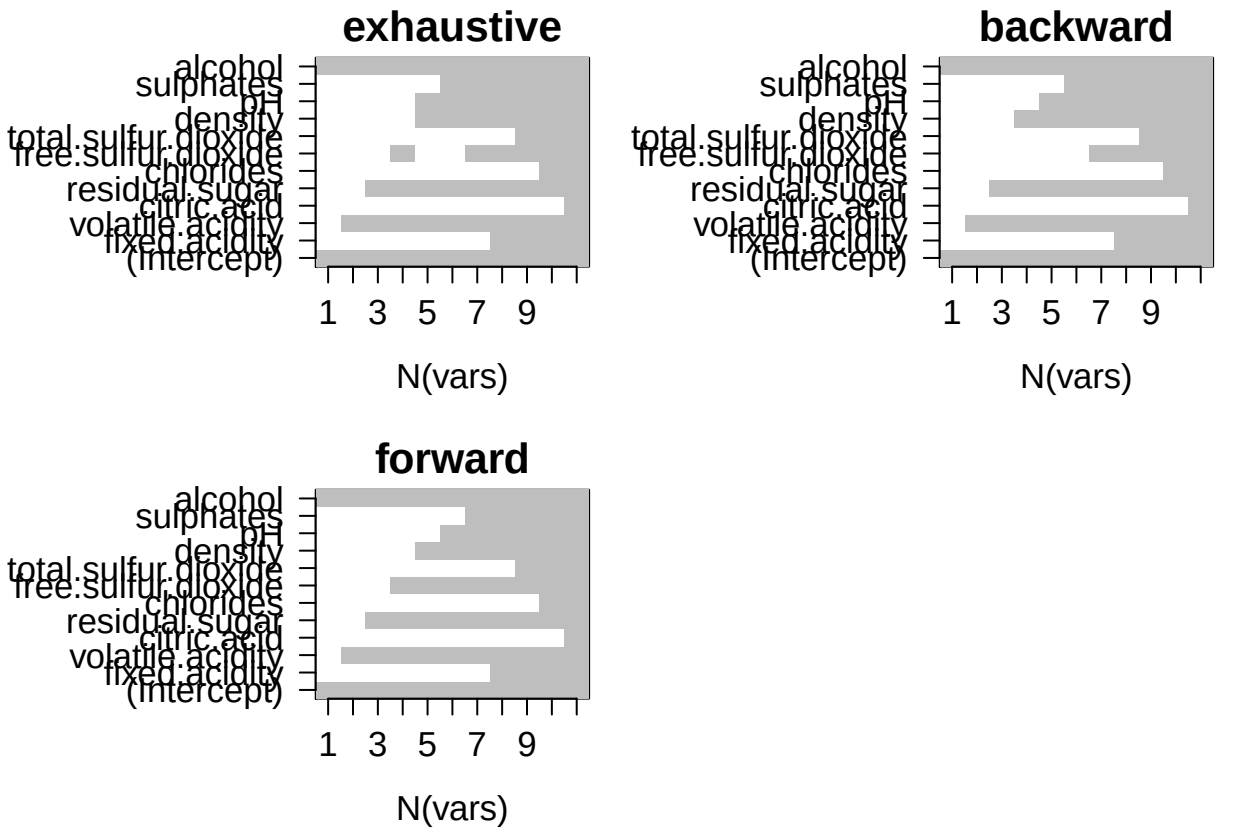


White Wine

```
summaryMetrics <- NULL
whichAll <- list()
for ( myMthd in c("exhaustive", "backward", "forward") ) {
  rsRes <- regsubsets(quality~.,whiteDat,method=myMthd,nvmax=11)
  summRes <- summary(rsRes)
  whichAll[[myMthd]] <- summRes$which
  for ( metricName in c("rsq","rss","adjr2","cp","bic") ) {
    summaryMetrics <- rbind(summaryMetrics,
      data.frame(method=myMthd,metric=metricName,
                 nvars=1:length(summRes[[metricName]]),
                 value=summRes[[metricName]]))
  }
}
ggplot(summaryMetrics,aes(x=nvars,y=value,shape=method,colour=method)) + geom_path() + geom_point() + fa
```




```
old.par <- par(mfrow=c(2,2),ps=16,mar=c(5,10,2,1))
for ( myMthd in names(whichAll) ) {
  image(1:nrow(whichAll[[myMthd]]),
        1:ncol(whichAll[[myMthd]]),
        whichAll[[myMthd]],xlab="N(vars)",ylab="",
        xaxt="n",yaxt="n",breaks=c(-0.5,0.5,1.5),
        col=c("white","gray"),main=myMthd)
  axis(1,1:nrow(whichAll[[myMthd]]),rownames(whichAll[[myMthd]]))
  axis(2,1:ncol(whichAll[[myMthd]]),colnames(whichAll[[myMthd]]),las=2)
}
par(old.par)
```



For the red wine dataset, all the selection methods performed similarly. The performance metrics get very flat at about 6 variables and sometimes get worse with more predictors. For the different selection methods, the same variables are chosen. First alcohol, then volatile acidity, then sulphates, total sulphur dioxide, chlorides, pH, free sulphur dioxide,citric acid, residual sugar, fixed acidity, and finally density. The red wine models performed better than the white wine models with a max adjusted R squared value at about .36 while the white wine models peak at about .28.

For the white wine dataset, the different selection methods also perform similarly, although it looks like backwards performs better with 5 and 6 variables. The performance metrics really flatten out at 7 terms and for some measures gets worse. Performance with white wine is lower than with red.

Each selection method chooses variables a little differently but they all pick alcohol, volatile acidity, and residual sugar for the first three. They also all take chlorides and citric acid last. After the initial three, exhaustive picks free sulphur dioxide, which is then dropped in favor of density and pH, followed by sulphates, free sulphur dioxide again and fixed acidity. The backwards selection method picks the first three, then density, pH, sulphates, free sulfur dioxide, fixed acidity, and total sulfur dioxide. Finally after the first three the forward method chooses free sulfur dioxide, density, pH, sulphates, fixed acidity, and total sulfur dioxide.

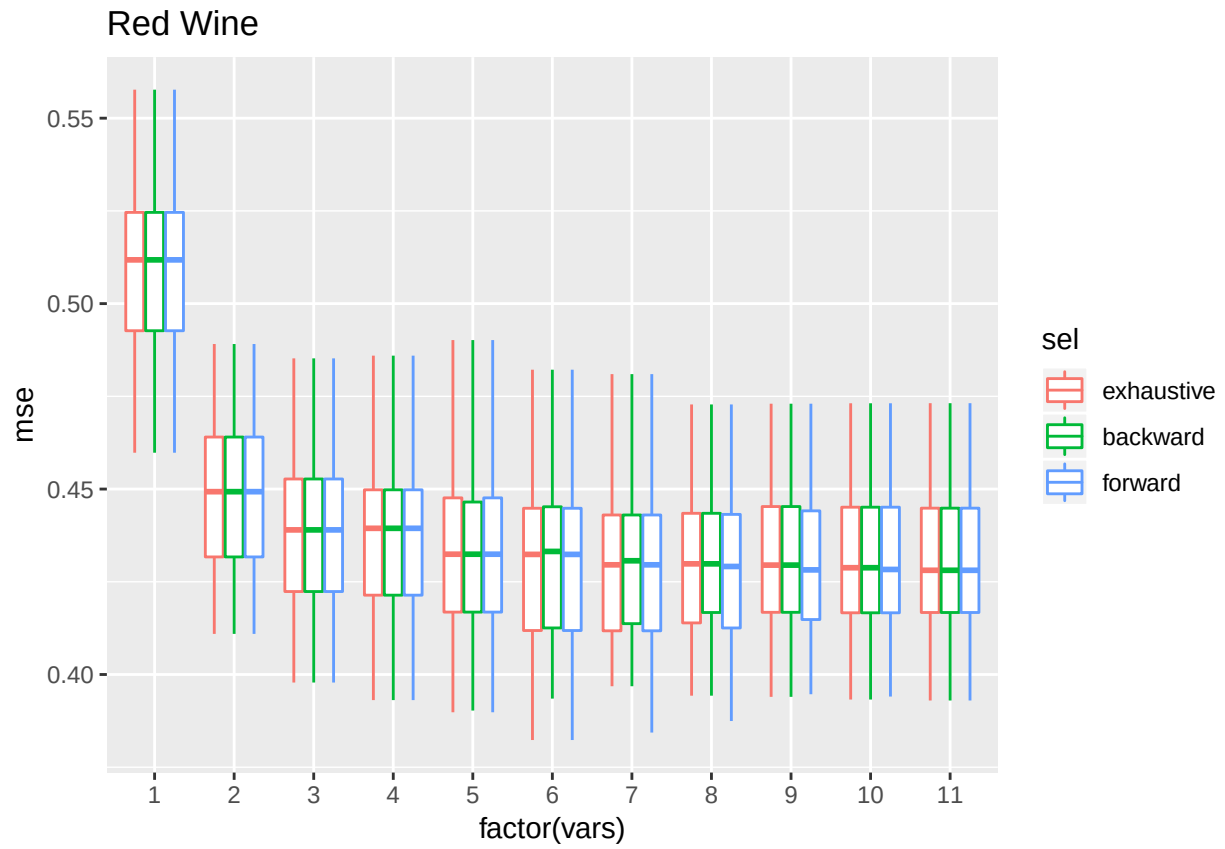
Alcohol and volatile acidity are both the most commonly chosen predictors for both red and white wines. Sulphates and total sulfur dioxide are important for the red models but not the white. Residual sugar is important for the white models but not the red ones.

Test models with resampling

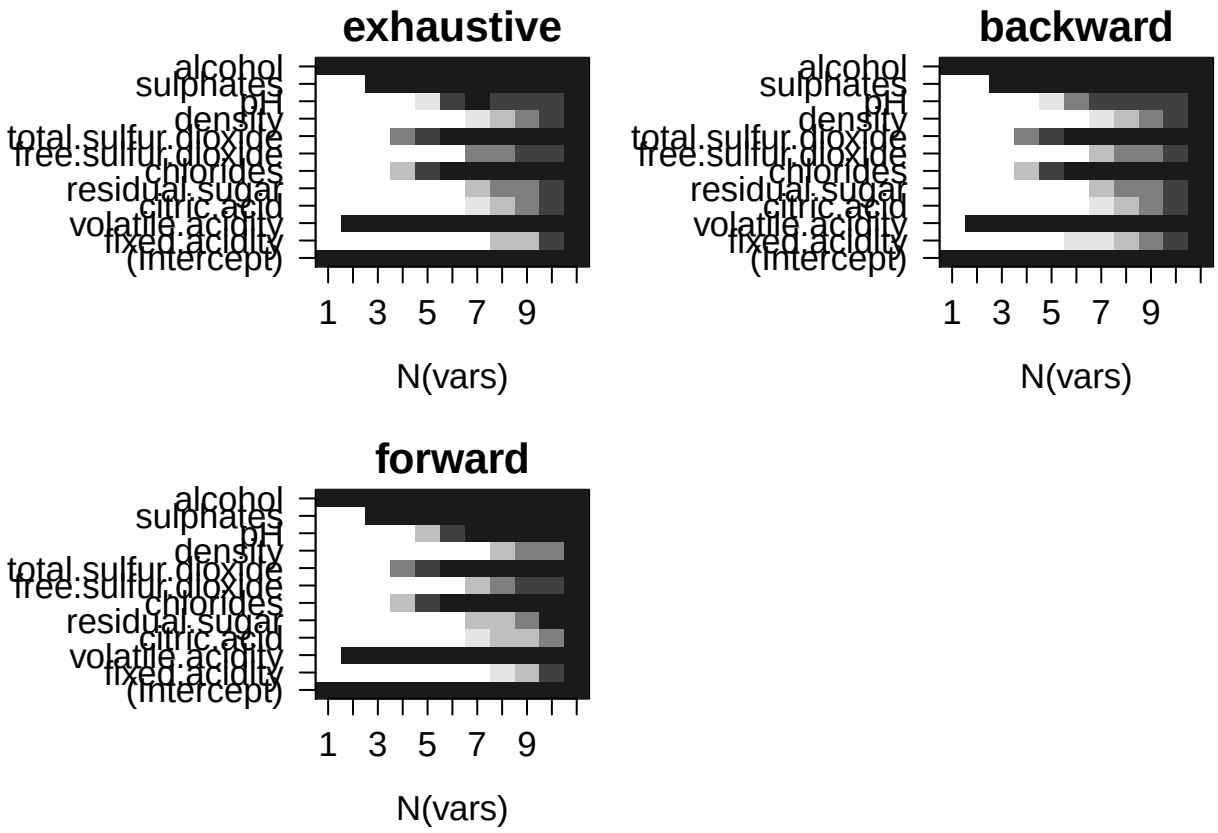
I split the data to fit and test models with different numbers of variables to find the optimal model.

Red Wine

```
dfTmp <- NULL
whichSum <- array(0,dim=c(11,12,3),
  dimnames=list(NULL,colnames(model.matrix(quality~.,redDat)),
      c("exhaustive", "backward", "forward")))
# Split data into training and test 30 times:
nTries <- 30
for ( iTry in 1:nTries ) {
  bTrain <- sample(rep(c(TRUE,FALSE),length.out=nrow(redDat)))
  # Try each method in regsubsets
  # to select best model of each size:
  for ( jSelect in c("exhaustive", "backward", "forward") ) {
    rsTrain <- regsubsets(quality~.,redDat[bTrain,],nvmax=11,method=jSelect)
    # Add up variable selections:
    whichSum[,,jSelect] <- whichSum[,,jSelect] + summary(rsTrain)$which
    # Calculate test error for each set of variables
    # using predict.regsubsets implemented above:
    for ( kVarSet in 1:11 ) {
      # make predictions:
      testPred <- predict(rsTrain,redDat[!bTrain,],id=kVarSet)
      # calculate MSE:
      mseTest <- mean((testPred-redDat[!bTrain,"quality"])^2)
      # add to data.frame for future plotting:
      dfTmp <- rbind(dfTmp,data.frame(sim=iTry,sel=jSelect,vars=kVarSet,
      mse=mseTest))
    }
  }
}
# plot MSEs by training/test, number of
# variables and selection method:
ggplot(dfTmp,aes(x=factor(vars),y=mse,colour=sel)) + ggtitle("Red Wine") + geom_boxplot()
```



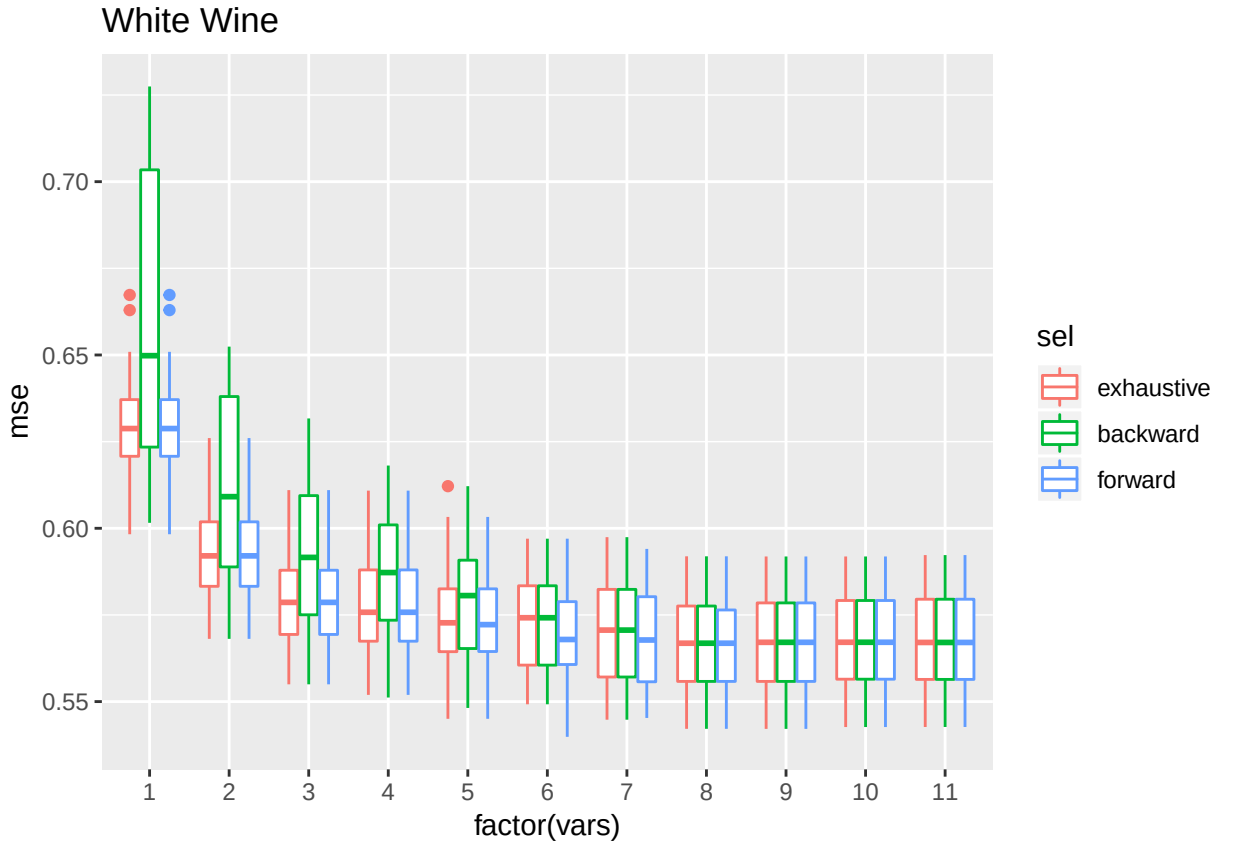

```
#average fraction of each variable inclusion in best model of every size
old.par <- par(mfrow=c(2,2),ps=16,mar=c(5,10,2,1))
for ( myMthd in dimnames(whichSum)[[3]] ) {
  tmpWhich <- whichSum[,,myMthd] / nTries
  image(1:nrow(tmpWhich),1:ncol(tmpWhich),tmpWhich,
        xlab="N(vars)",ylab="",xaxt="n",yaxt="n",main=myMthd,
        breaks=c(-0.1,0.1,0.25,0.5,0.75,0.9,1.1),
        col=c("white","gray90","gray75","gray50","gray25","gray10"))
  axis(1,1:nrow(tmpWhich),rownames(tmpWhich))
  axis(2,1:ncol(tmpWhich),colnames(tmpWhich),las=2)
}
par(old.par)
```



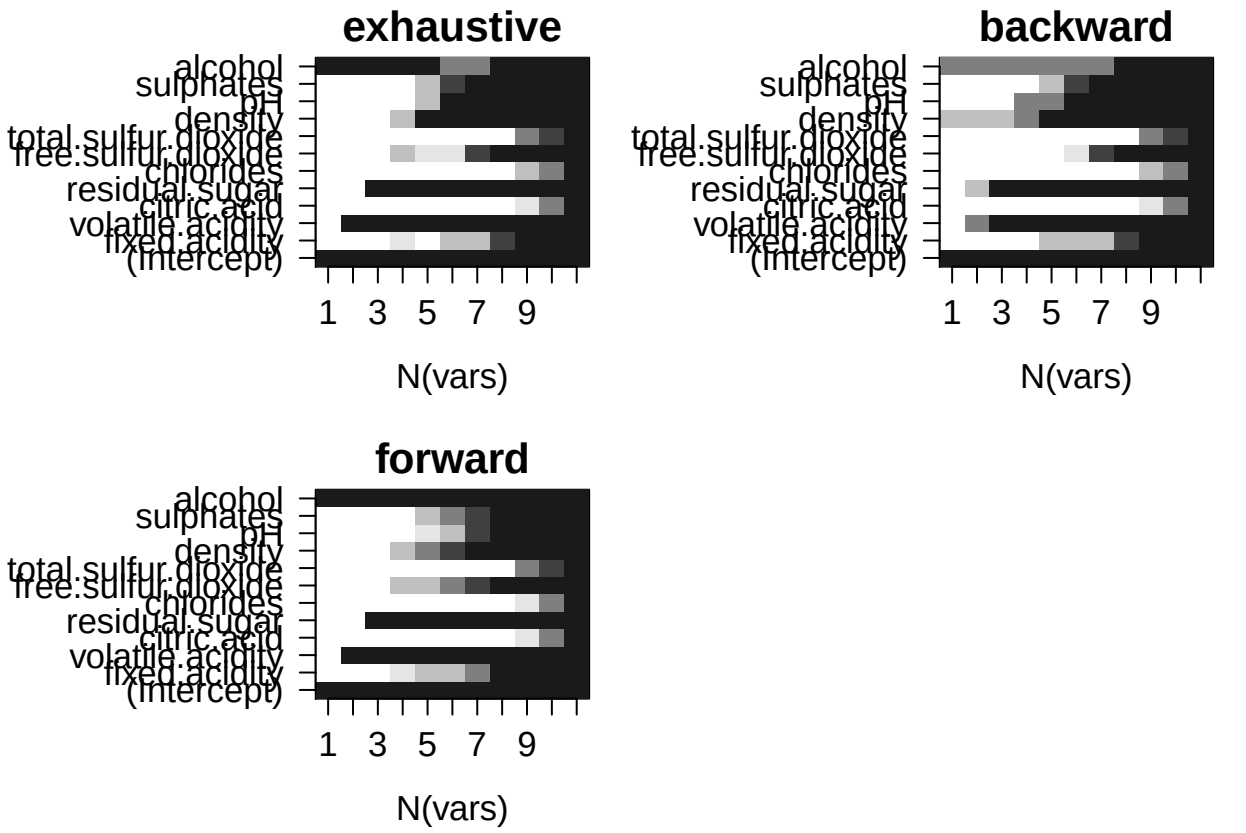
White Wine

```
dfTmp <- NULL
whichSum <- array(0,dim=c(11,12,3),
  dimnames=list(NULL,colnames(model.matrix(quality~.,whiteDat)),
      c("exhaustive", "backward", "forward")))
# Split data into training and test 30 times:
nTries <- 30
for ( iTry in 1:nTries ) {
  bTrain <- sample(rep(c(TRUE,FALSE),length.out=nrow(whiteDat)))
  # Try each method in regsubsets
  # to select best model of each size:
  for ( jSelect in c("exhaustive", "backward", "forward") ) {
    rsTrain <- regsubsets(quality~.,whiteDat[bTrain,],nvmax=11,method=jSelect)
    # Add up variable selections:
    whichSum[,,jSelect] <- whichSum[,,jSelect] + summary(rsTrain)$which
    # Calculate test error for each set of variables
    # using predict.regsubsets implemented above:
    for ( kVarSet in 1:11 ) {
      # make predictions:
      testPred <- predict(rsTrain,whiteDat[!bTrain,],id=kVarSet)
      # calculate MSE:
      mseTest <- mean((testPred-whiteDat[!bTrain,"quality"])^2)
      # add to data.frame for future plotting:
      dfTmp <- rbind(dfTmp,data.frame(sim=iTry,sel=jSelect,vars=kVarSet,
```

```
      mse=mseTest))
    }
  }
}
# plot MSEs by training/test, number of
# variables and selection method:
ggplot(dfTmp,aes(x=factor(vars),y=mse,colour=sel)) + ggtitle("White Wine") + geom_boxplot()
```




```
#average fraction of each variable inclusion in best model of every size
old.par <- par(mfrow=c(2,2),ps=16,mar=c(5,10,2,1))
for ( myMthd in dimnames(whichSum)[[3]] ) {
  tmpWhich <- whichSum[,,myMthd] / nTries
  image(1:nrow(tmpWhich),1:ncol(tmpWhich),tmpWhich,
        xlab="N(vars)",ylab="",xaxt="n",yaxt="n",main=myMthd,
        breaks=c(-0.1,0.1,0.25,0.5,0.75,0.9,1.1),
        col=c("white","gray90","gray75","gray50","gray25","gray10"))
  axis(1,1:nrow(tmpWhich),rownames(tmpWhich))
  axis(2,1:ncol(tmpWhich),colnames(tmpWhich),las=2)
}
par(old.par)
```



To estimate test error I randomly split the datasets 30 times into a training and test set. For red wine; exhaustive, backwards and forwards perform similarly. One to two variables has an appreciable affect. Moving to three variables may be worth considering but after that adding more provides very little improvement. The minimum median MSE occurs with around 6 variables, about 0.425. the choice of variables looks similar for all three methods. Alcohol, volatile acidity, and sulphates are the first three chosen and are very stable. These are the same choices as above. After three or four terms, the stability of the chosen optimal variable is poor.

For white wine, the backwards method performs worse with higher variance for the first five variables. this is interesting because backwards appeared to perform slightly better when training and testing against the entire dataset. Ignoring the backwards method, a three variable model is justified but going with four or more provides little reduction in MSE. The minimum median MSE is around 0.565. The optimal variables chosen also looks similar across methods and compared to the optimal variables chosen while using the entire dataset. Alcohol, volatile acidity and residual sugar are all fairly stable. After those three the rest are less stable. The fourth variable chosen is commonly density and then either free sulfur dioxide or fixed acidity.

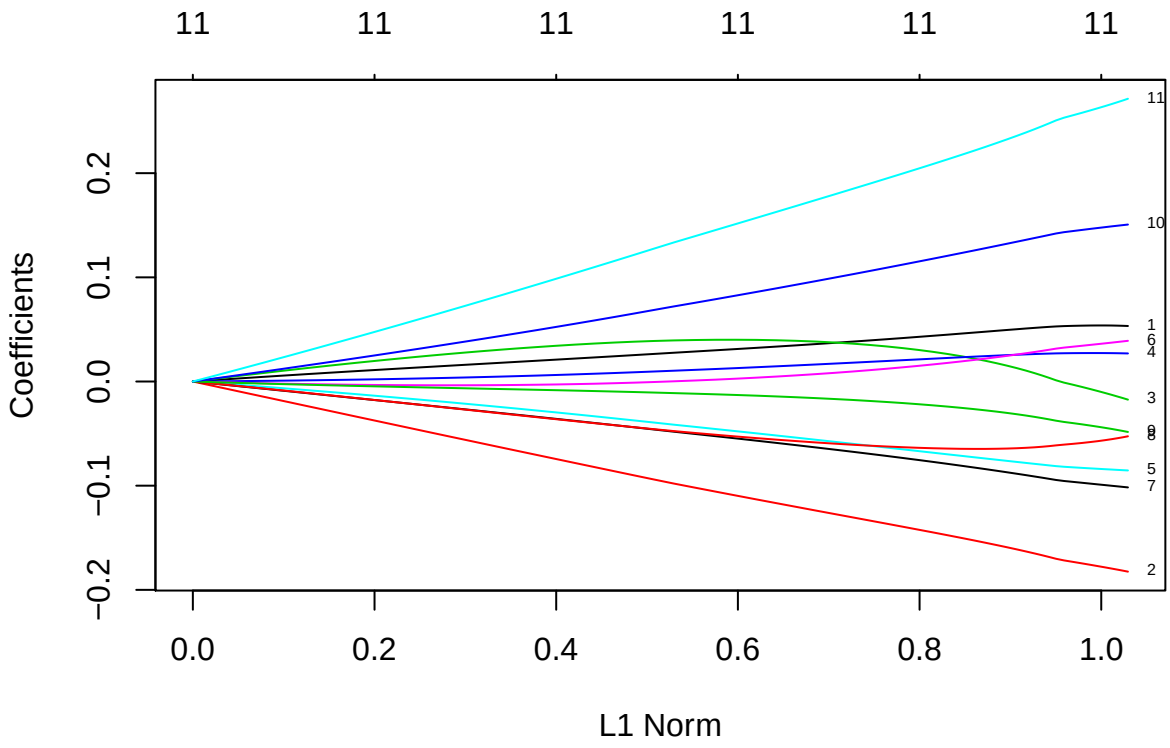
Model selection by regularized approaches

I try Ridge and Lasso regression to model quality of red and white wines and compare those models to the ones already produced.

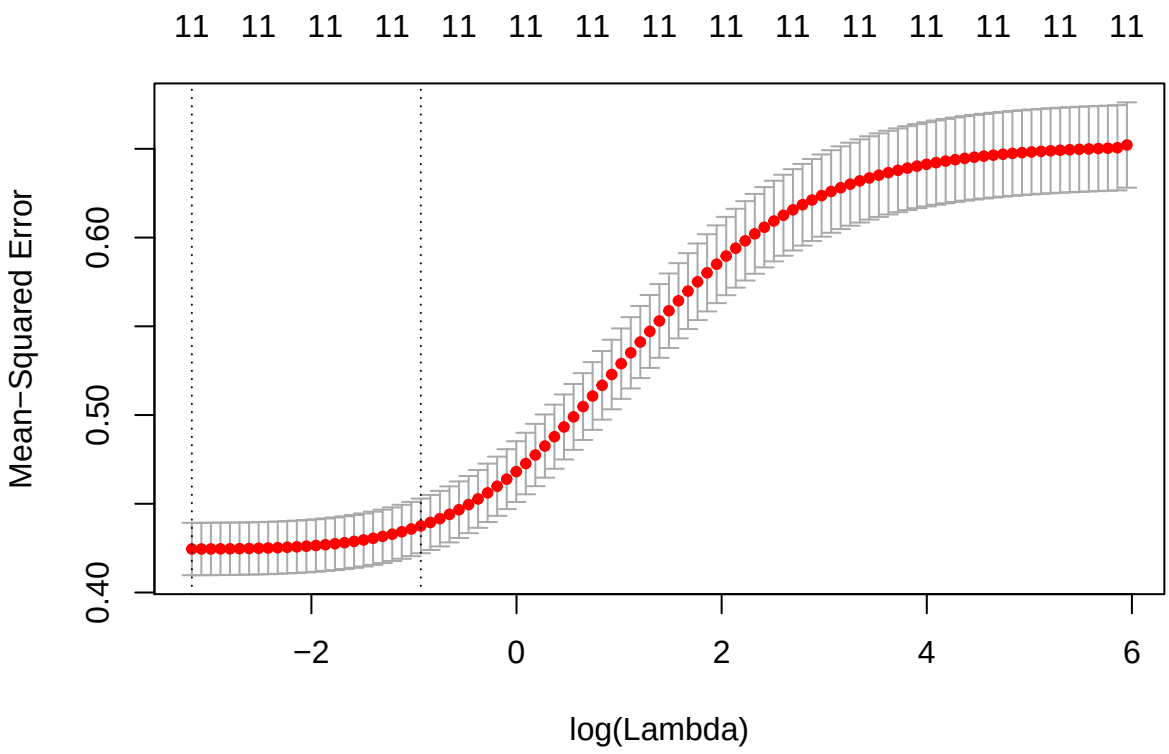
Red Wine

```
#scale and perform Ridge Regression
x <- scale(model.matrix(quality~.,redDat)[,-1])
y <- redDat[,"quality"]
```

```
ridgeRes <- glmnet(x,y,alpha=0)
plot(ridgeRes,label=TRUE)
```




```
cvRidgeRes <- cv.glmnet(x,y,alpha=0)
plot(cvRidgeRes)
```




```
cvRidgeRes$lambda.min
```

```
## [1] 0.04218973
```

```
predict(ridgeRes,type="coefficients",s=cvRidgeRes$lambda.min)
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                1
## (Intercept)           5.63602251
## fixed.acidity         0.05348478
## volatile.acidity     -0.18158331
## citric.acid          -0.01593318
## residual.sugar        0.02706594
## chlorides            -0.08511045
## free.sulfur.dioxide   0.03851676
## total.sulfur.dioxide -0.10117062
## density              -0.05352363
## pH                   -0.04748817
## sulphates             0.15014626
## alcohol               0.26981622
```

```
cvRidgeRes$lambda.1se
```

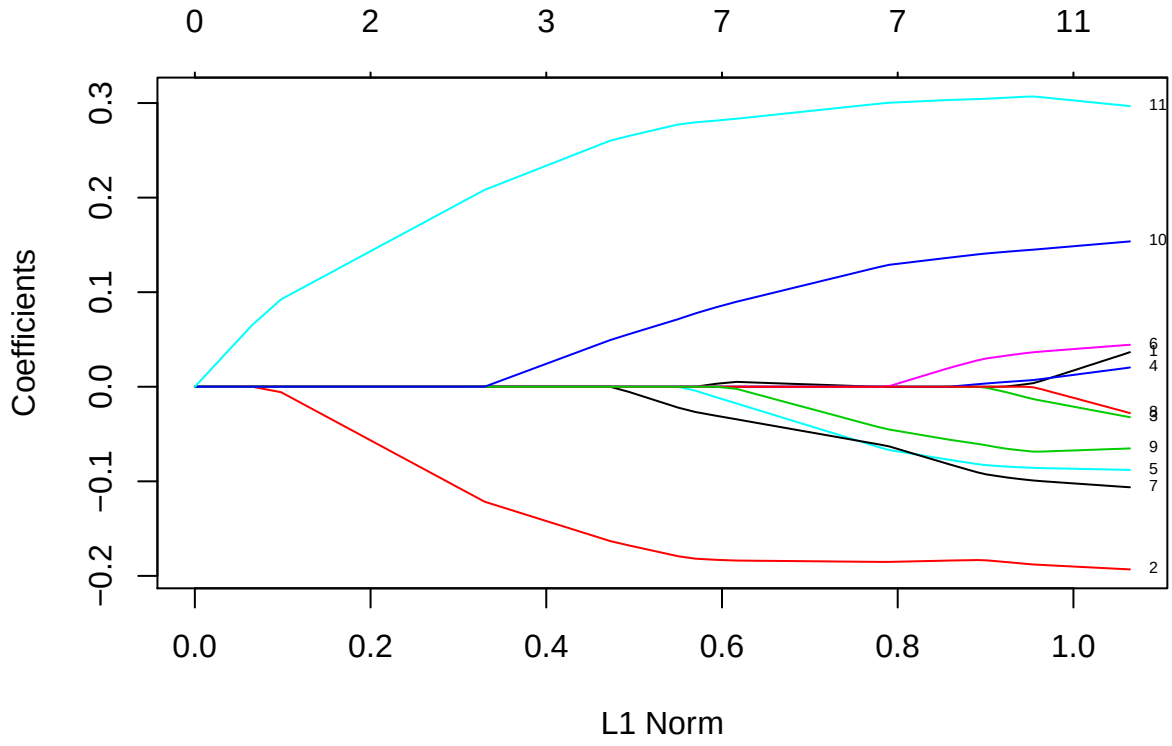
```
## [1] 0.3934628
```

```
predict(ridgeRes,type="coefficients",s=cvRidgeRes$lambda.1se)
```

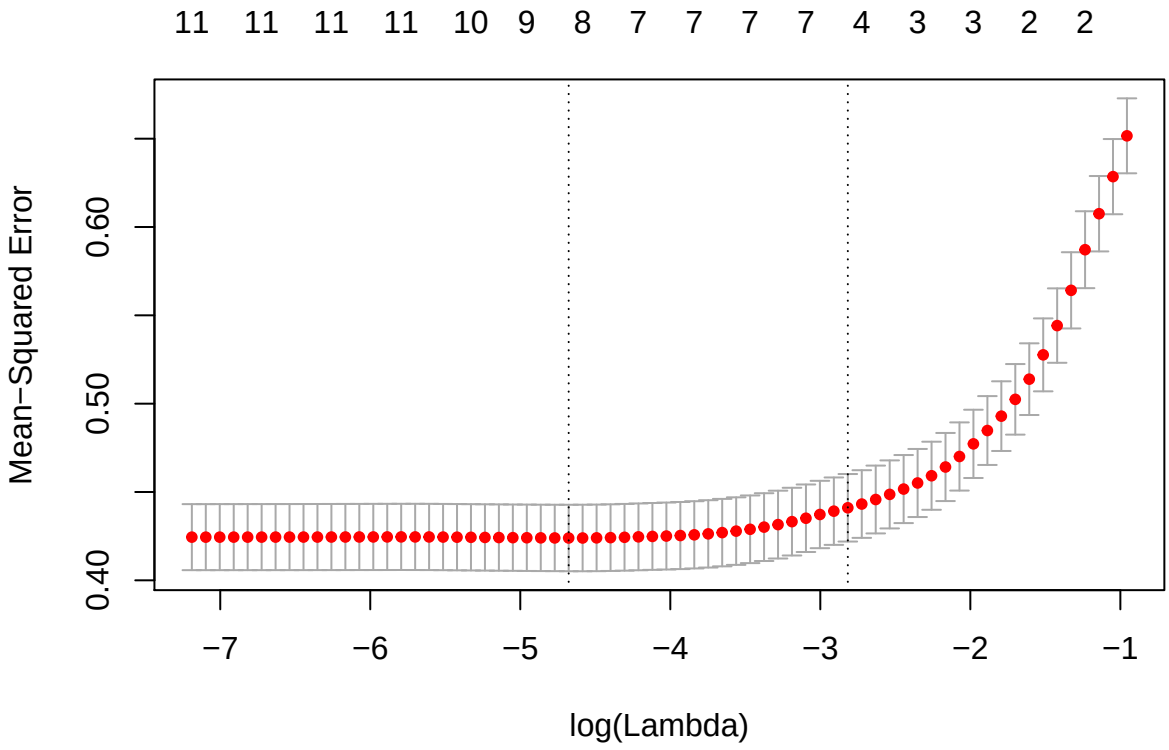
```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                1
```

```
## (Intercept)           5.63602251
## fixed.acidity         0.04060199
## volatile.acidity     -0.13649976
## citric.acid           0.03365799
## residual.sugar        0.01969343
## chlorides            -0.06340475
## free.sulfur.dioxide   0.01225852
## total.sulfur.dioxide -0.07146034
## density              -0.06239148
## pH                   -0.01965285
## sulphates             0.10922947
## alcohol               0.19490442
```

```
#Lasso Regression
lassoRes <- glmnet(x,y,alpha=1)
plot(lassoRes,label=TRUE)
```




```
cvLassoRes <- cv.glmnet(x,y,alpha=1)
plot(cvLassoRes)
```




```
cvLassoRes$lambda.min
```

```
## [1] 0.009303387
```

```
predict(lassoRes,type="coefficients",s=cvLassoRes$lambda.min)
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                1
## (Intercept)           5.636022514
## fixed.acidity         .
## volatile.acidity     -0.183567011
## citric.acid           .
## residual.sugar        0.001051982
## chlorides            -0.078990324
## free.sulfur.dioxide   0.023015005
## total.sulfur.dioxide -0.085284185
## density               .
## pH                   -0.057655868
## sulphates             0.137844436
## alcohol               0.303598249
```

```
cvLassoRes$lambda.1se
```

```
## [1] 0.05980285
```

```
predict(lassoRes,type="coefficients",s=cvLassoRes$lambda.1se)
```

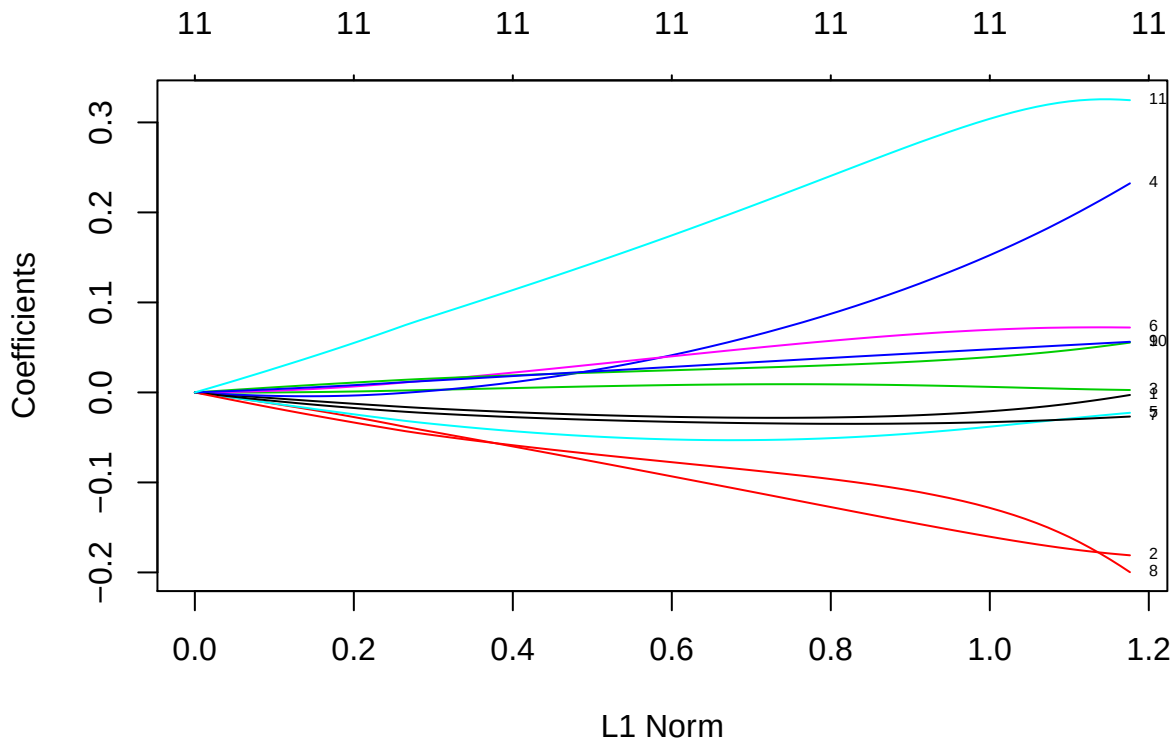
```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                               1
```

```
## (Intercept)           5.63602251
## fixed.acidity          .
## volatile.acidity     -0.17917726
## citric.acid            .
## residual.sugar         .
## chlorides              .
## free.sulfur.dioxide    .
## total.sulfur.dioxide -0.02201416
## density                .
## pH                     .
## sulphates              0.07143518
## alcohol                0.27725714
```

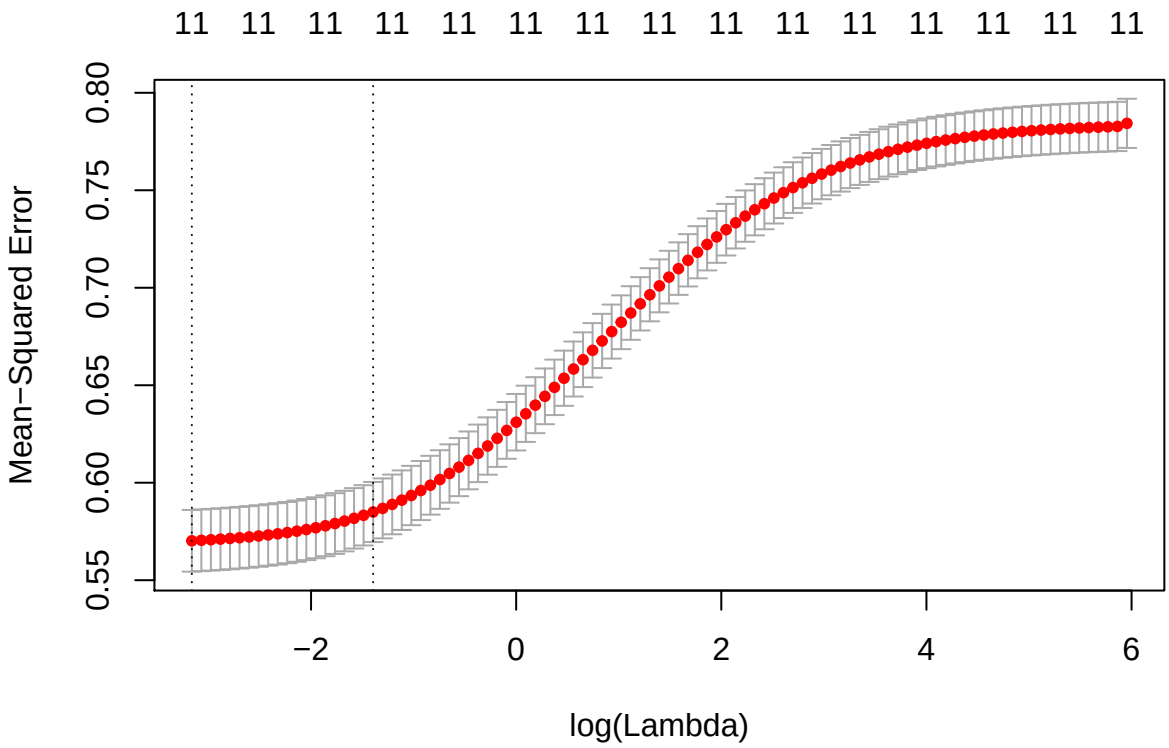
In the red wine dataset, previous work suggested three variables using alcohol, volatile acidity and sulphates. In ridge and lasso regression we also see the largest impract from alcohol, volatile acidity and sulphates. The least regularized model does not even include fixed acidity, citric acid or density. The more regularized model with 1se from minimum lambda includes 4 variables: alcohol, volatile acidity, sulphates, and total sulfur dioxide, with alcohol and volatile acidity having by far the biggest impact.

White Wine

```
#scale and perform Ridge Regression
x <- scale(model.matrix(quality~.,whiteDat)[,-1])
y <- whiteDat[,"quality"]
ridgeRes <- glmnet(x,y,alpha=0)
plot(ridgeRes,label=TRUE)
```



```
cvRidgeRes <- cv.glmnet(x,y,alpha=0)
plot(cvRidgeRes)
```




```
cvRidgeRes$lambda.min
```

```
## [1] 0.04233298
```

```
predict(ridgeRes,type="coefficients",s=cvRidgeRes$lambda.min)
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                1
## (Intercept)           5.877909351
## fixed.acidity        -0.004574642
## volatile.acidity     -0.180043857
## citric.acid           0.002826352
## residual.sugar        0.225992499
## chlorides            -0.023603324
## free.sulfur.dioxide   0.072200358
## total.sulfur.dioxide -0.027343604
## density              -0.192653624
## pH                    0.053876125
## sulphates             0.055639515
## alcohol               0.325315748
```

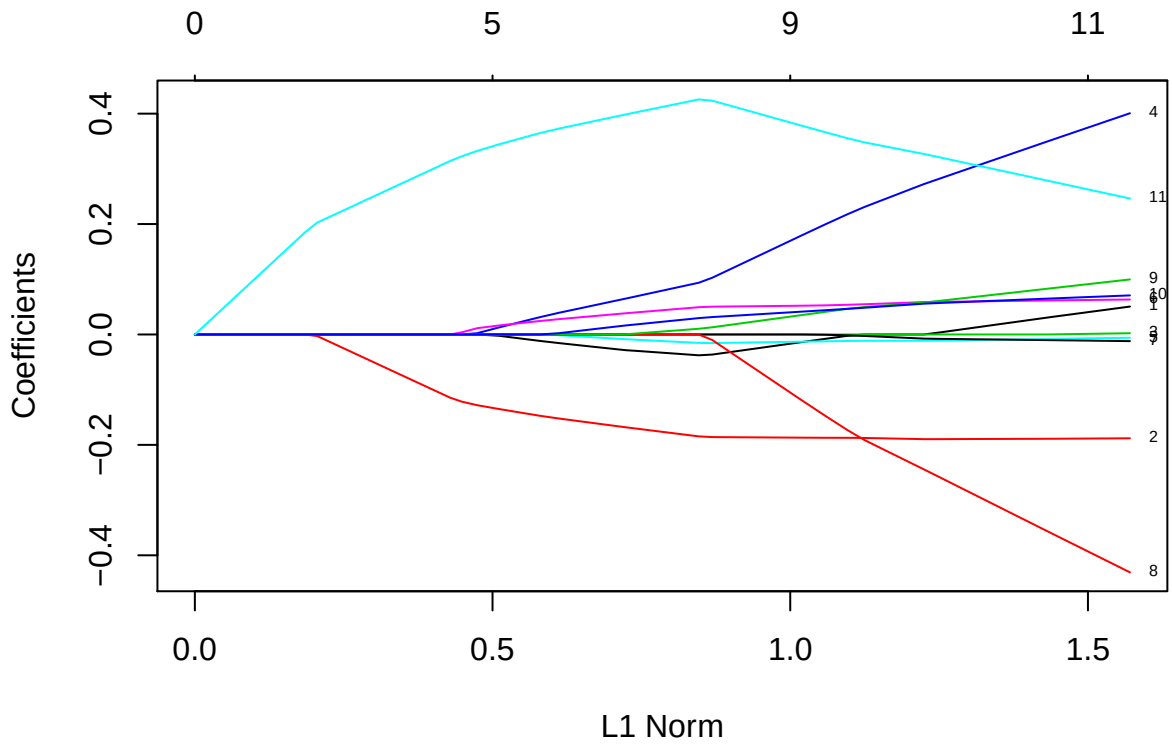
```
cvRidgeRes$lambda.1se
```

```
## [1] 0.2479452
```

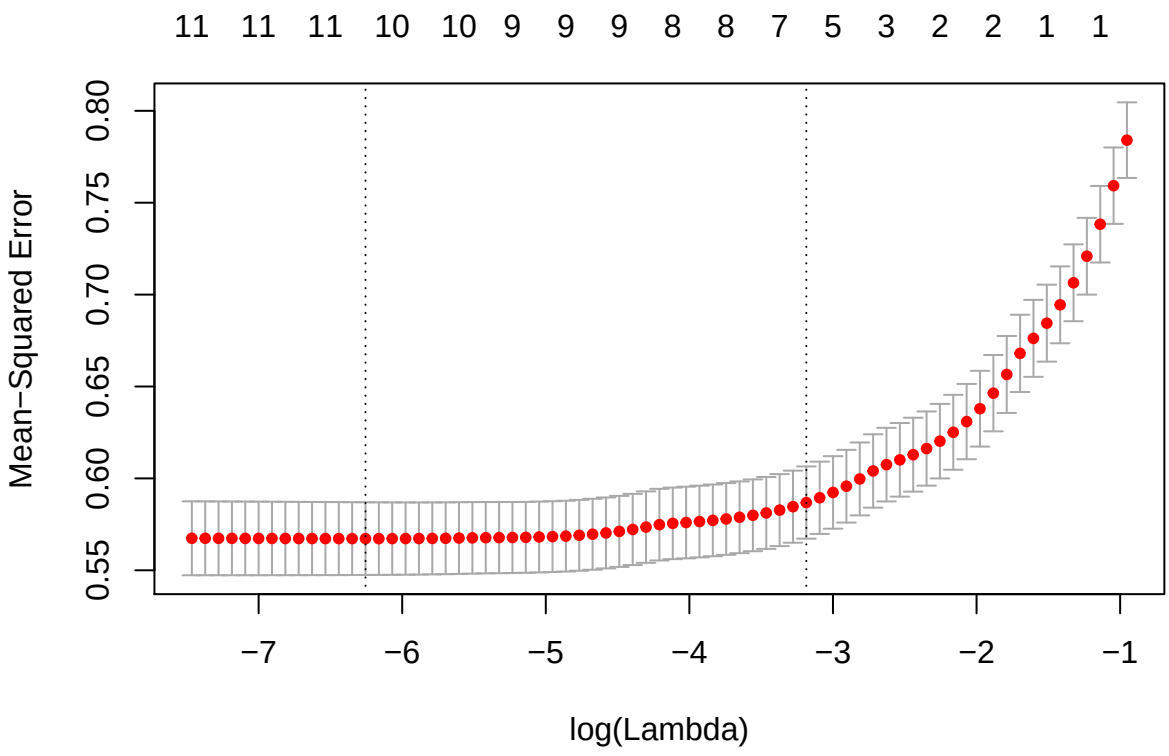
```r
predict(ridgeRes,type="coefficients",s=cvRidgeRes$lambda.1se)
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                1
## (Intercept)           5.877909351
## fixed.acidity        -0.026492357
## volatile.acidity     -0.139698974
## citric.acid           0.008364751
## residual.sugar        0.108370060
## chlorides            -0.047413341
## free.sulfur.dioxide   0.062598762
## total.sulfur.dioxide -0.034765220
## density              -0.105030267
## pH                    0.032826257
## sulphates             0.041838527
## alcohol               0.265053200
```

```r
#Lasso Regression
lassoRes <- glmnet(x,y,alpha=1)
plot(lassoRes,label=TRUE)
```




```r
cvLassoRes <- cv.glmnet(x,y,alpha=1)
plot(cvLassoRes)
```




```
cvLassoRes$lambda.min
```

```
## [1] 0.001919749
```

```
predict(lassoRes,type="coefficients",s=cvLassoRes$lambda.min)
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                1
## (Intercept)           5.8779093508
## fixed.acidity         0.0324636402
## volatile.acidity     -0.1889059142
## citric.acid           0.0001661752
## residual.sugar        0.3541844184
## chlorides            -0.0078047194
## free.sulfur.dioxide   0.0617424315
## total.sulfur.dioxide -0.0104050983
## density              -0.3630667647
## pH                    0.0843817179
## sulphates             0.0654326907
## alcohol               0.2757895966
```

```
cvLassoRes$lambda.1se
```

```
## [1] 0.04135975
```

```
predict(lassoRes,type="coefficients",s=cvLassoRes$lambda.1se)
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                1
```

```
## (Intercept)           5.877909351
## fixed.acidity        -0.019335318
## volatile.acidity     -0.156801699
## citric.acid           .
## residual.sugar        0.046586616
## chlorides            -0.003490220
## free.sulfur.dioxide   0.030823565
## total.sulfur.dioxide  .
## density               .
## pH                    .
## sulphates             0.006344567
## alcohol               0.380234758
```

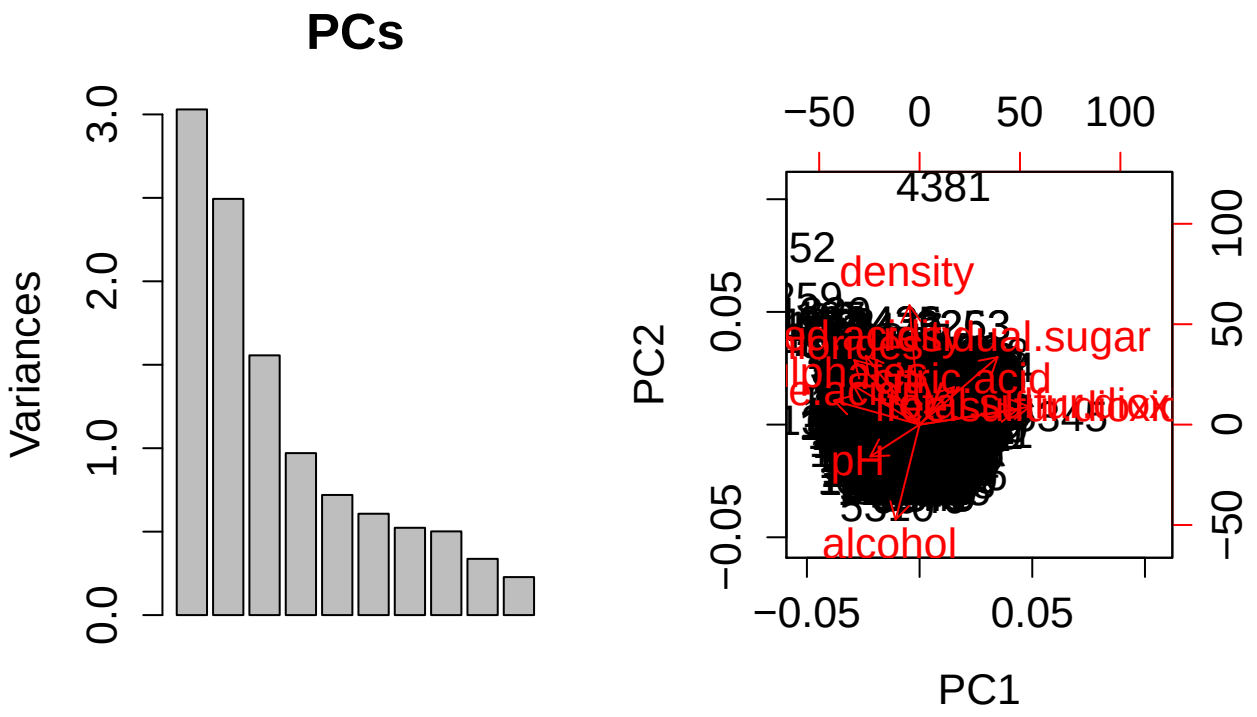
For the white wine dataset I had suggested using three variables: alcohol, volatile acidity and residual sugar. In ridge regression, using the lambda for minimum MSE; alcohol, residual sugar, density and volatile acidity have the largest impacts in that order. This differs a little from what I found before. However using the 1se lambda, I get alcohol, volatile acidity, residual sugar and density with the largest impacts which is more consistent with previous findings except that density has more of an impact than expected. Using lasso regression, the minimum MSE lambda uses 10 variables with residual sugar, density, alcohol and volatile acidity having the largest impacts. Again, this is not totally consistent with previous findings but is likely due to collinearity. Looking at the coefficients with the 1se lambda, there are 8 variables still in the model. Alcohol has by far the highest magnitude, then volatile acidity, residual sugar and free sulfur dioxide. Density drops out when using this lambda.

PCA

I combine the red and white wine data and perform principal component analysis to see if any structures emerge.

```
#join Red and White
wineDat=rbind(cbind(redDat,Wine="red"),cbind(whiteDat,Wine="yellow"))

#plot prcomp and biplot
old.par <- par(mfrow=c(1,2),ps=16)
plot(prcomp(scale(wineDat[,1:11])),main="PCs")
biplot(prcomp(scale(wineDat[,1:11])))
```




```
par(old.par)

#var rotations fo PC1 and PC2
sort(prcomp(scale(wineDat[,1:11]))$rotation[,1],decreasing=TRUE)
```

```
## total.sulfur.dioxide  free.sulfur.dioxide       residual.sugar
##           0.48741806           0.43091401           0.34591993
##          citric.acid              density              alcohol
##           0.15238844          -0.04493664          -0.10643712
##                   pH        fixed.acidity            chlorides
##          -0.21868644          -0.23879890          -0.29011259
##            sulphates     volatile.acidity
##          -0.29413517          -0.38075750
```

```
sort(prcomp(scale(wineDat[,1:11]))$rotation[,2],decreasing=TRUE)
```

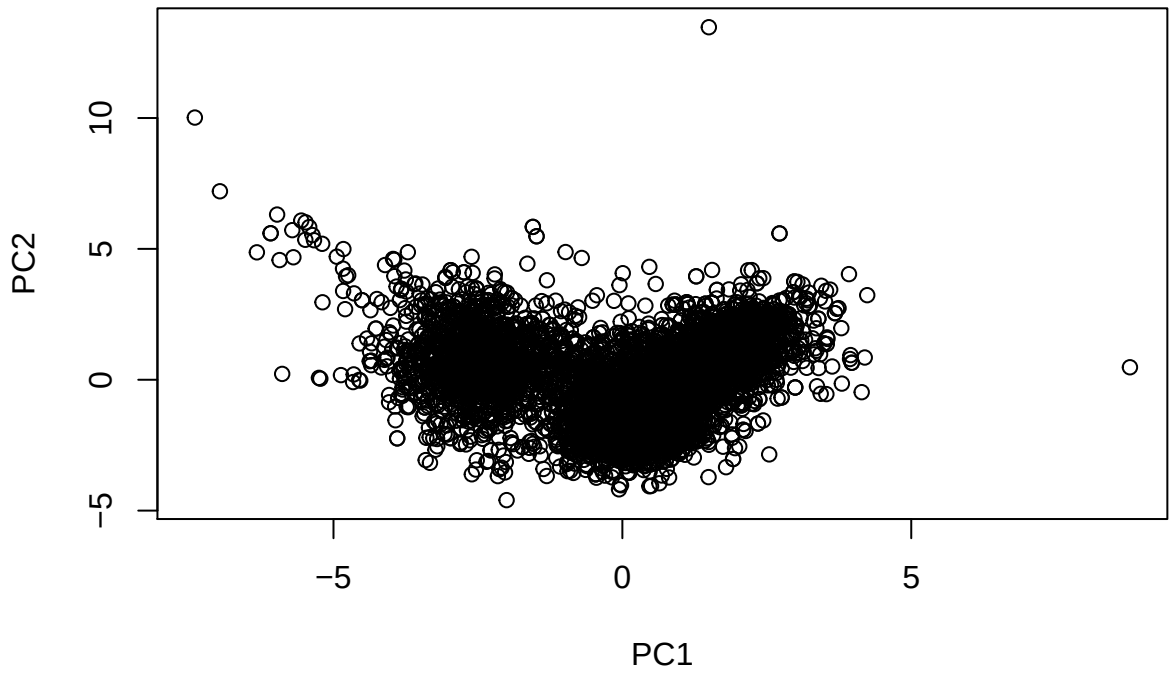
```
##              density        fixed.acidity       residual.sugar
##           0.58403734           0.33635454           0.32991418
##            chlorides            sulphates          citric.acid
##           0.31525799           0.19171577           0.18329940
##     volatile.acidity total.sulfur.dioxide  free.sulfur.dioxide
##           0.11754972           0.08726628           0.07193260
##                   pH              alcohol
##          -0.15586900          -0.46505769
```

```
sort(prcomp(scale(wineDat[,1:11]))$rotation[,3],decreasing=TRUE)
```

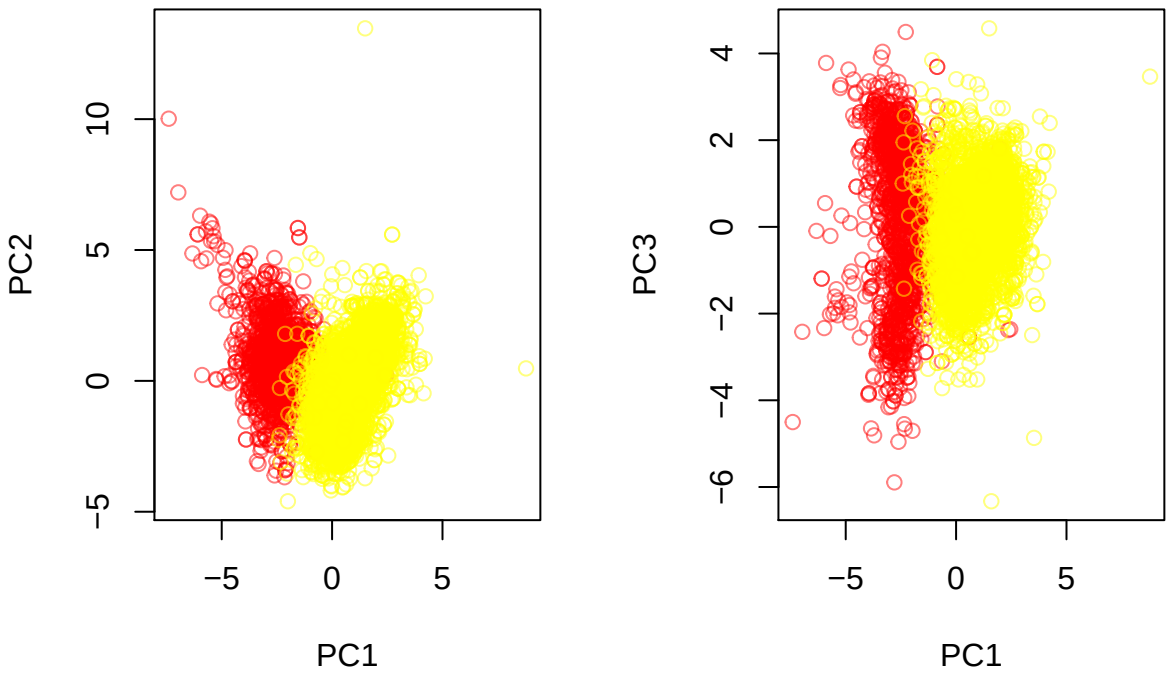
```
##                   pH     volatile.acidity              density
```

```
##           0.45532412           0.30725942           0.17560555
##       residual.sugar  free.sulfur.dioxide total.sulfur.dioxide
##           0.16468843           0.13422395           0.10746230
##            chlorides            sulphates              alcohol
##           0.01667910          -0.07004248          -0.26110053
##        fixed.acidity          citric.acid
##          -0.43430130          -0.59056967
```

```
#scatterplot PC1 vs PC2
plot(prcomp(wineDat[,1:11],scale=T)$x[,1:2])
```



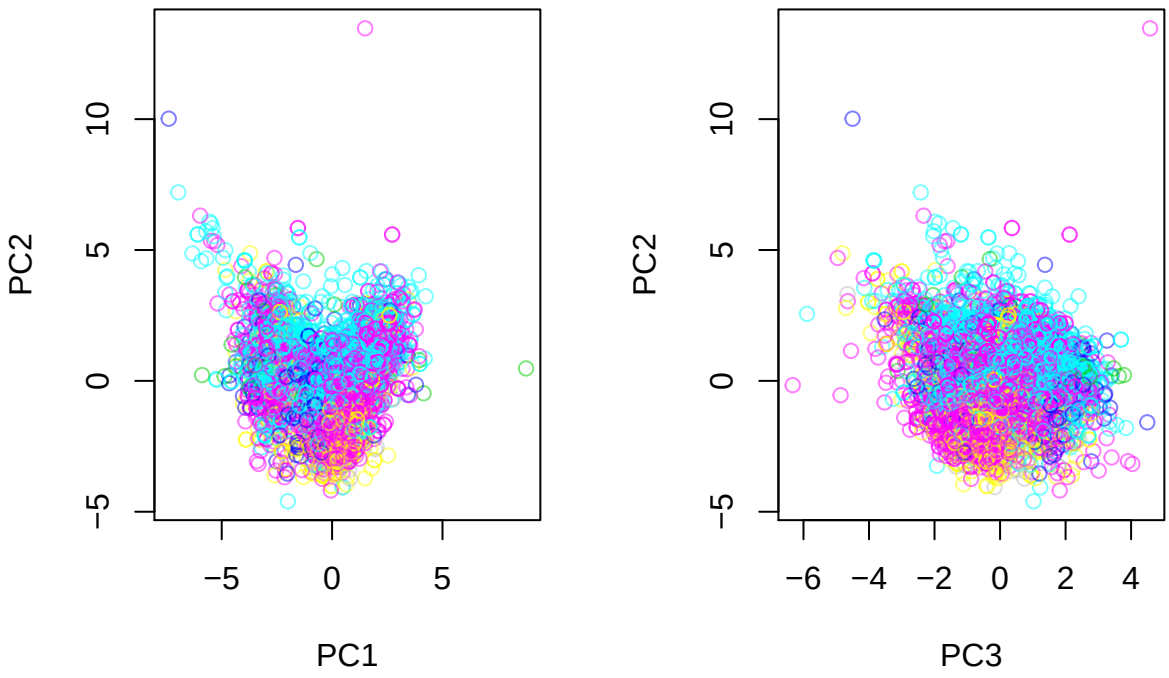

```
#color by wine type
old.par=par(mfrow=c(1,2))
plot(prcomp(wineDat[,1:11],scale=T)$x[,1:2],col=alpha(paste0(wineDat$Wine),.5))
plot(prcomp(wineDat[,1:11],scale=T)$x[,c(1,3)],col=alpha(paste0(wineDat$Wine),.5))
```




```
par(old.par)
```

PC1 is most closely aligned with free sulfur dioxide, total sulfur dioxide, and voltile acidity. This is where we see the most variation in wines. PC2 is most related to density and alcohol. Residual sugar contributes to both PC1 and PC2 about evenly. With PC1 and PC2, there does appear to be two possible clusters. If we color by red and white wines, we see that greater values of PC1 contains more of the white wines and lower values contains more of the reds. including PC3 may help even more by showing an elongated red wine cluster and a more compact white wine cluster.

```
#color by quality
old.par=par(mfrow=c(1,2))
plot(prcomp(wineDat[,1:11],scale=T)$x[,1:2],col=alpha(wineDat$quality,.5))
#legend("topright",legend=c(3:9),fill=alpha(c(3:9),.8))
plot(prcomp(wineDat[,1:11],scale=T)$x[,c(3,2)],col=alpha(wineDat$quality,.5))
```



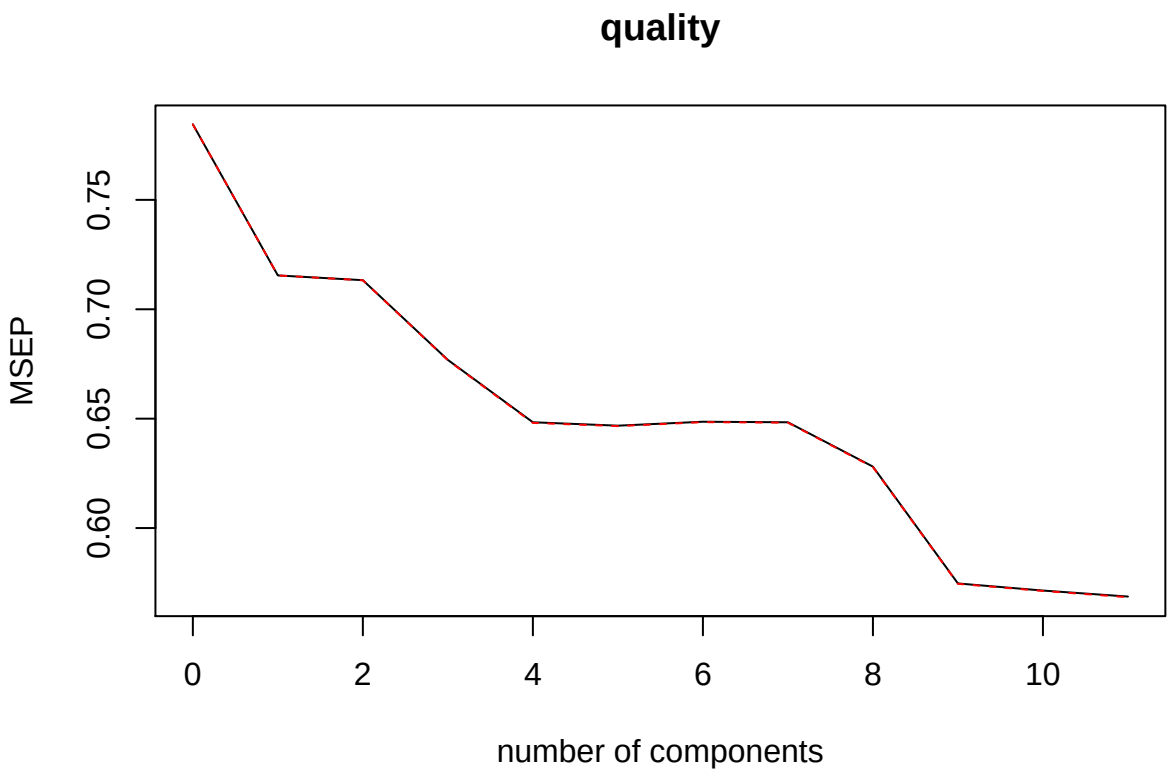

```
#legend("topright",legend=c(3:9),col=alpha(c(3:9),.8),pch="o")
par(old.par)
```

If we color the scatterplot by quality rating, there does appear to be some regionality but there is also a very large amount of overlap. It's hard to tell but PC2 seems to have a stronger impact on quality (which makes sense given that it is aligned with alcohol).

Wine quality model using principal components

For white wine, I use the principal components as predictors in a linear model of wine quality.

```
pcr.fit=pcr(quality~.,data=whiteDat,scale=TRUE,validation="CV")
validationplot(pcr.fit,val.type = "MSEP")
```




```
summary(pcr.fit)
```

```
## Data:    X dimension: 4898 11
##  Y dimension: 4898 1
## Fit method: svdpc
## Number of components considered: 11
##
## VALIDATION: RMSEP
## Cross-validated using 10 random segments.
##        (Intercept)  1 comps  2 comps  3 comps  4 comps  5 comps  6 comps
## CV          0.8857   0.8459   0.8446   0.8227   0.8052   0.8042   0.8054
## adjCV       0.8857   0.8458   0.8445   0.8226   0.8050   0.8041   0.8052
##        7 comps  8 comps  9 comps  10 comps  11 comps
## CV      0.8052   0.7925   0.7581    0.7559    0.7541
## adjCV   0.8051   0.7924   0.7580    0.7558    0.7539
##
## TRAINING: % variance explained
##          1 comps  2 comps  3 comps  4 comps  5 comps  6 comps  7 comps  8 comps
## X         29.293   43.614    54.72    63.98    72.83    81.36    87.97    93.42
## quality    8.818    9.156    13.92    17.56    17.78    17.78    17.83    20.39
##          9 comps  10 comps  11 comps
## X          97.18     99.81    100.00
## quality    27.14     27.58     28.19
```

```
head(sort(cvRidgeRes$cvm,decreasing = FALSE))
```

```
## [1] 0.5702208 0.5704762 0.5707561 0.5710632 0.5714057 0.5717835
```

The lowest adjusted cross validation error occurs when using all 11 principal components. Using all 11 gives an MSE of 0.569 which is similar to the Ridge regression lowest MSE. If instead of 11 components we use 3, the MSE is 0.677.