5110_Final_Project

Bindu Latha Banisetti

4/25/2022

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.6     v dplyr   1.0.8
## v tidyr   1.1.4     v stringr 1.4.0
## v readr   2.1.2     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(modelr)
library(gridExtra)
```

```
##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine
```

```
library(randomForest)
```

```
## randomForest 4.7-1

## Type rfNews() to see new features/changes/bug fixes.

##
## Attaching package: 'randomForest'

## The following object is masked from 'package:gridExtra':
##
##     combine

## The following object is masked from 'package:dplyr':
##
##     combine
## The following object is masked from ’package:ggplot2’:
##
##     margin
```

```
library(glmnet)
```

```
## Loading required package: Matrix
```

```
##
## Attaching package: ’Matrix’
```

```
## The following objects are masked from ’package:tidyr’:
##
##     expand, pack, unpack
```

```
## Loaded glmnet 4.1-4
```

```
library(caret)
```

```
## Loading required package: lattice
```

```
##
## Attaching package: ’caret’
```

```
## The following object is masked from ’package:purrr’:
##
##     lift
```

```
library(broom)
```

```
##
## Attaching package: ’broom’
```

```
## The following object is masked from ’package:modelr’:
##
##     bootstrap
```

```
dir1 <- file.path("winequality-red.csv")
dir2 <- file.path("winequality-white.csv")
red_wine_df <- read.csv(dir1, header = TRUE, sep = ";")
white_wine_df <- read.csv(dir2, header = TRUE, sep = ";")

which(is.na(red_wine_df))
```

```
## integer(0)
```

```
which(is.na(white_wine_df))
```

```
## integer(0)
```

Stepwise Model Selection for red wine

```
#Partition of data
red_wine_std <- data.frame(scale(red_wine_df[1:11]))
red_wine_std$quality <- red_wine_df$quality
set.seed(10)
partition_rw <- resample_partition(red_wine_std,
                                p=c(train=0.5,
                                    valid=0.25,
                                    test=0.25))
```

```
#Function to calculate the RMSE of the predictors
step <- function(response, predictors, candidates, partition)
{
  rhs <- paste0(paste0(predictors, collapse="+"), "+", candidates)
  formulas <- lapply(paste0(response, "~", rhs), as.formula)
  rmses <- sapply(formulas,
                  function(fm) rmse(lm(fm, data=partition$train),
                                    data=partition$valid))
  names(rmses) <- candidates
  attr(rmses, "best") <- rmses[which.min(rmses)]
  rmses
}
```

```
model_rw <- NULL

preds <- "1"
cands <- c("fixed.acidity","volatile.acidity","citric.acid","residual.sugar",
           "chlorides","free.sulfur.dioxide","total.sulfur.dioxide","density",
           "pH","sulphates","alcohol")
s1 <- step("quality", preds, cands, partition_rw)

model_rw <- c(model_rw, attr(s1, "best"))
s1
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##            0.8515544            0.7855048            0.8398467
##       residual.sugar            chlorides  free.sulfur.dioxide
##            0.8578792            0.8472211            0.8532622
## total.sulfur.dioxide              density                   pH
##            0.8401893            0.8376948            0.8553145
##            sulphates              alcohol
##            0.8296731            0.7346630
## attr(,"best")
##  alcohol
## 0.734663
```

```
preds <- "alcohol"
cands <- c("fixed.acidity","volatile.acidity","citric.acid","residual.sugar",
           "chlorides","free.sulfur.dioxide","total.sulfur.dioxide","density",
           "pH","sulphates")
s1 <- step("quality", preds, cands, partition_rw)
```

```
model_rw <- c(model_rw, attr(s1, "best"))
s1
```

```
##        fixed.acidity     volatile.acidity          citric.acid 
##            0.7226665            0.6902853            0.7215572 
##       residual.sugar            chlorides  free.sulfur.dioxide 
##            0.7347169            0.7342695            0.7340596 
## total.sulfur.dioxide              density                   pH 
##            0.7308255            0.7322296            0.7188688 
##            sulphates 
##            0.7190395 
## attr(,"best")
## volatile.acidity 
##        0.6902853
```

```
preds <- c("alcohol","volatile.acidity")
cands <- c("fixed.acidity","citric.acid","residual.sugar",
           "chlorides","free.sulfur.dioxide","total.sulfur.dioxide","density",
           "pH","sulphates")
s1 <- step("quality", preds, cands, partition_rw)

model_rw <- c(model_rw, attr(s1, "best"))
s1
```

```
##        fixed.acidity          citric.acid       residual.sugar 
##            0.6878592            0.6903075            0.6911506 
##            chlorides  free.sulfur.dioxide total.sulfur.dioxide 
##            0.6902750            0.6890341            0.6862892 
##              density                   pH            sulphates 
##            0.6892822            0.6877442            0.6848121 
## attr(,"best")
## sulphates 
## 0.6848121
```

```
preds <- c("alcohol","volatile.acidity","sulphates")
cands <- c("fixed.acidity","citric.acid","residual.sugar",
           "chlorides","free.sulfur.dioxide","total.sulfur.dioxide","density",
           "pH")
s1 <- step("quality", preds, cands, partition_rw)

model_rw <- c(model_rw, attr(s1, "best"))
s1
```

```
##        fixed.acidity          citric.acid       residual.sugar 
##            0.6831365            0.6846930            0.6857594 
##            chlorides  free.sulfur.dioxide total.sulfur.dioxide 
##            0.6831365            0.6830200            0.6794726 
##              density                   pH 
##            0.6849305            0.6831672 
## attr(,"best")
## total.sulfur.dioxide 
##            0.6794726
```

```r
preds <- c("alcohol","volatile.acidity","sulphates","total.sulfur.dioxide")
cands <- c("fixed.acidity","citric.acid","residual.sugar",
           "chlorides","free.sulfur.dioxide","density",
           "pH")
s1 <- step("quality", preds, cands, partition_rw)

model_rw <- c(model_rw, attr(s1, "best"))
s1
```

```
##       fixed.acidity         citric.acid      residual.sugar           chlorides
##           0.6784623           0.6795727           0.6805401           0.6774864
## free.sulfur.dioxide             density                  pH
##           0.6806207           0.6796242           0.6776286
## attr(,"best")
## chlorides
## 0.6774864
```

```r
preds <- c("alcohol","volatile.acidity","sulphates","total.sulfur.dioxide","chlorides")
cands <- c("fixed.acidity","citric.acid","residual.sugar",
           "free.sulfur.dioxide","density",
           "pH")
s1 <- step("quality", preds, cands, partition_rw)

model_rw <- c(model_rw, attr(s1, "best"))
s1
```

```
##       fixed.acidity         citric.acid      residual.sugar free.sulfur.dioxide
##           0.6760903           0.6775808           0.6785556           0.6780041
##             density                  pH
##           0.6776890           0.6740840
## attr(,"best")
##       pH
## 0.674084
```

```r
preds <- c("alcohol","volatile.acidity","sulphates","total.sulfur.dioxide","chlorides","pH")
cands <- c("fixed.acidity","citric.acid","residual.sugar",
           "free.sulfur.dioxide","density")
s1 <- step("quality", preds, cands, partition_rw)

model_rw <- c(model_rw, attr(s1, "best"))
s1
```

```
##       fixed.acidity         citric.acid      residual.sugar free.sulfur.dioxide
##           0.6741904           0.6737470           0.6752222           0.6748061
##             density
##           0.6747926
## attr(,"best")
## citric.acid
##    0.673747
```

```
preds <- c("alcohol","volatile.acidity","sulphates","total.sulfur.dioxide",
           "chlorides","pH","citric.acid")
cands <- c("fixed.acidity","residual.sugar",
           "free.sulfur.dioxide","density")
s1 <- step("quality", preds, cands, partition_rw)

model_rw <- c(model_rw, attr(s1, "best"))
s1
```

```
##       fixed.acidity      residual.sugar free.sulfur.dioxide             density
##           0.6735421           0.6750824           0.6743678           0.6738133
## attr(,"best")
## fixed.acidity
##     0.6735421
```

```
preds <- c("alcohol","volatile.acidity","sulphates","total.sulfur.dioxide",
           "chlorides","pH","citric.acid", "fixed.acidity")
cands <- c("residual.sugar", "free.sulfur.dioxide","density")
s1 <- step("quality", preds, cands, partition_rw)

model_rw <- c(model_rw, attr(s1, "best"))
s1
```

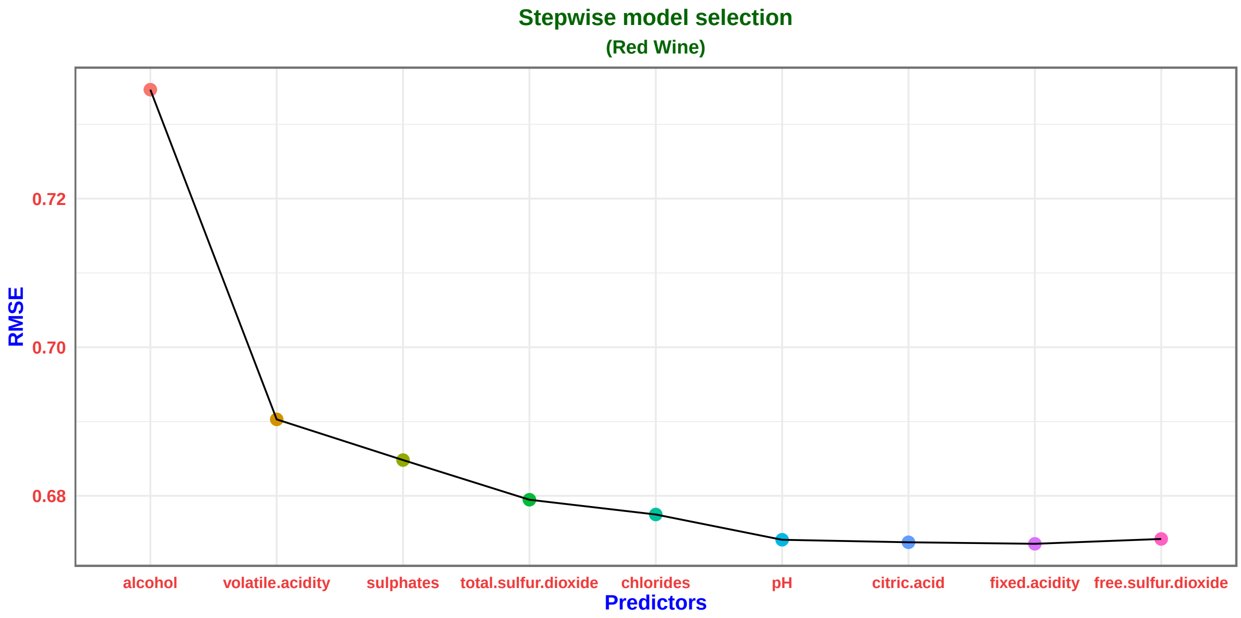
```
##      residual.sugar free.sulfur.dioxide             density
##           0.6749624           0.6741934           0.6742341
## attr(,"best")
## free.sulfur.dioxide
##           0.6741934
```

Model stopped improving at:

- fit = quality ~ alcohol + volatile acidity + sulphates + total sulfur dioxide + chlorides + pH + citric acid + fixed acidity

Visualizing how adding each variable affects the RMSE

```
step_model <- tibble(index=seq_along(model_rw),
                     variable=factor(names(model_rw), levels=names(model_rw)),
                     RMSE=model_rw)

ggplot(step_model, aes(y=RMSE)) +
  geom_point(aes(x=variable, color = variable), size =3) +
  geom_line(aes(x=index)) +
  theme_minimal()+
  labs(title = "Stepwise model selection",
               subtitle = "(Red Wine)",
               x = "Predictors",
               y = "RMSE")+
        theme(plot.title = element_text(hjust = 0.5,
                                        color = "darkgreen",
                                        face = "bold"),
              plot.subtitle = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
               axis.title = element_text(color = "blue",
                                         face = "bold",
                                         size = 12),
               axis.text.x = element_text(hjust = 0.5,
                                          vjust = 0.5,
                                          color = "brown2",
                                          face = "bold",
                                          size = 9),
               axis.text.y = element_text(color = "brown2",
                                          face = "bold",
                                          size = 10),
               panel.border = element_rect(colour = "grey45",
                                           fill=NA, size=1),
               legend.position = "none")
```



visualizing how each variable(including the predictors) affects the RMSE for Red wine

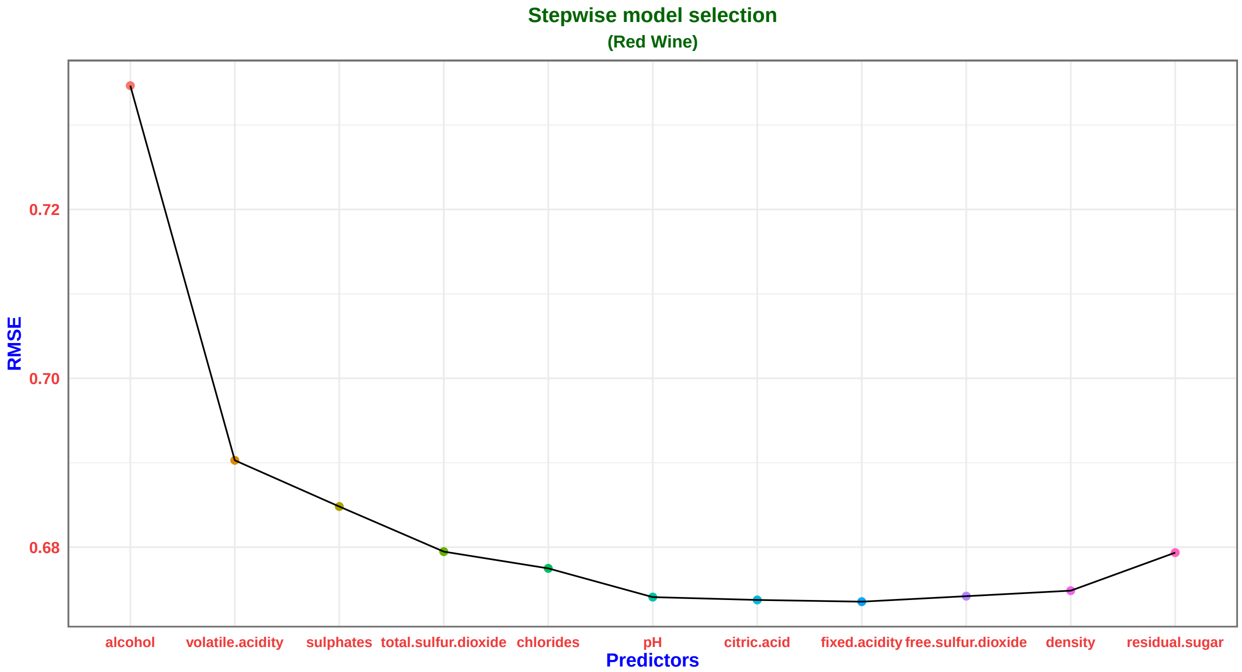
```
temp_model_rw <- model_rw
preds <- c("alcohol","volatile.acidity","sulphates","total.sulfur.dioxide",
           "chlorides","pH","citric.acid", "fixed.acidity", "free.sulfur.dioxide")
cands <- c("residual.sugar","density")
s1 <- step("quality", preds, cands, partition_rw)

temp_model_rw <- c(temp_model_rw, attr(s1, "best"))
s1
```

```
## residual.sugar        density
##      0.6754391      0.6748470
## attr(,"best")
##  density
## 0.674847
```

```
preds <- c("alcohol","volatile.acidity","sulphates","total.sulfur.dioxide",
           "chlorides","pH","citric.acid", "fixed.acidity", "free.sulfur.dioxide",
           "density")
cands <- c("residual.sugar")
s1 <- step("quality", preds, cands, partition_rw)

temp_model_rw <- c(temp_model_rw, attr(s1, "best"))
s1
```

```
## residual.sugar
##      0.6793522
## attr(,"best")
## residual.sugar
##      0.6793522
```

```
#Representation of RMSE for all the variables
step_model_temp <- tibble(index=seq_along(temp_model_rw),
                     variable=factor(names(temp_model_rw), levels=names(temp_model_rw)),
                     RMSE=temp_model_rw)

ggplot(step_model_temp, aes(y=RMSE)) +
  geom_point(aes(x=variable, color = variable), size = 2) +
  geom_line(aes(x=index)) +
  theme_minimal()+
  labs(title = "Stepwise model selection",
               subtitle = "(Red Wine)",
               x = "Predictors",
               y = "RMSE")+
        theme(plot.title = element_text(hjust = 0.5,
                                        color = "darkgreen",
                                        face = "bold"),
              plot.subtitle = element_text(hjust = 0.5,
                                        color = "darkgreen",
                                        face = "bold"),
              axis.title = element_text(color = "blue",
                                        face = "bold",
                                        size = 12),
              axis.text.x = element_text(hjust = 0.5,
                                        vjust = 0.5,
                                        color = "brown2",
                                        face = "bold",
                                        size = 9),
              axis.text.y = element_text(color = "brown2",
                                        face = "bold",
                                        size = 10),
              panel.border = element_rect(colour = "grey45",
                                        fill=NA, size=1),
              legend.position = "none")
```



Consider the following fits and extract the best fit model:

- fit1 <- quality ~ alcohol + volatile acidity + sulphates + total sulfur dioxide + chlorides + pH
- fit2 <- quality ~ alcohol + volatile acidity + sulphates + total sulfur dioxide + chlorides + pH + citric acid
- fit3 <- quality ~ alcohol + volatile acidity + sulphates + total sulfur dioxide + chlorides + pH + citric acid + fixed acidity

Cross Validation

```
set.seed(2020)
#partition_rw_train <- red_wine_std[-partition_rw$test$idx,]
#redwine_cv <- crossv_kfold(partition_rw_train, 5)
#redwine_cv

redwine_cv <- crossv_kfold(red_wine_std, 5)
redwine_cv
```

```
## # A tibble: 5 x 3
##   train                    test                   .id
##   <named list>             <named list>           <chr>
## 1 <resample [1,279 x 12]> <resample [320 x 12]> 1
## 2 <resample [1,279 x 12]> <resample [320 x 12]> 2
## 3 <resample [1,279 x 12]> <resample [320 x 12]> 3
## 4 <resample [1,279 x 12]> <resample [320 x 12]> 4
## 5 <resample [1,280 x 12]> <resample [319 x 12]> 5
```

```
#Calculating RMSE for each fold of data
cv_rw <- redwine_cv %>%
  mutate(fit = purrr::map(train,
                   ~ lm(quality ~ alcohol + volatile.acidity + sulphates +
                          total.sulfur.dioxide + chlorides + pH, data = .)),
         rmse = purrr::map2_dbl(fit, test, ~ rmse(.x, .y)))

cv_rw
```

```
## # A tibble: 5 x 5
##   train                  test                 .id   fit            rmse
##   <named list>           <named list>         <chr> <named list> <dbl>
## 1 <resample [1,279 x 12]> <resample [320 x 12]> 1     <lm>         0.673
## 2 <resample [1,279 x 12]> <resample [320 x 12]> 2     <lm>         0.659
## 3 <resample [1,279 x 12]> <resample [320 x 12]> 3     <lm>         0.587
## 4 <resample [1,279 x 12]> <resample [320 x 12]> 4     <lm>         0.670
## 5 <resample [1,280 x 12]> <resample [319 x 12]> 5     <lm>         0.666
```

```
#Average of RMSEs
mean(cv_rw$rmse)
```

```
## [1] 0.650961
```

Comparing models using CV

```
do_redwine_cv <- function(formula) {
  redwine_cv %>%
    mutate(fit = map(train,
                     ~ lm(formula, data = .)),
         rmse = map2_dbl(fit, test, ~ rmse(.x, .y)),
         rsq = map2_dbl(fit,test,~rsquare(.x,.y)),
         mae = map2_dbl(fit,test,~mae(.x,.y))) %>%
    summarize(cv_rmse = mean(rmse), cv_rsq = mean(rsq),
              cv_mae = mean(mae)) %>%
    return(c(cv_rmse, cv_rsq, cv_mae))
}
```

Calling the function

```
fit1_rmse <- do_redwine_cv(quality ~ alcohol + volatile.acidity + sulphates +
                total.sulfur.dioxide + chlorides + pH )

fit2_rmse <- do_redwine_cv(quality ~ alcohol + volatile.acidity + sulphates +
                total.sulfur.dioxide + chlorides + pH + citric.acid)

fit3_rmse <- do_redwine_cv(quality ~ alcohol + volatile.acidity + sulphates +
                total.sulfur.dioxide + chlorides + pH + citric.acid + fixed.acidity)

fit1_rmse
```

```
## # A tibble: 1 x 3
##   cv_rmse cv_rsq cv_mae
##     <dbl>  <dbl>  <dbl>
## 1   0.651  0.349  0.506
```

```
fit2_rmse
```

```
## # A tibble: 1 x 3
##   cv_rmse cv_rsq cv_mae
##     <dbl>  <dbl>  <dbl>
## 1   0.651  0.349  0.506
```

```
fit3_rmse
```

```
## # A tibble: 1 x 3
##   cv_rmse cv_rsq cv_mae
##     <dbl>  <dbl>  <dbl>
## 1   0.651  0.348  0.507
```

```
#Goodness of fit

gof_fs_rw <- lm( quality ~ alcohol + volatile.acidity + sulphates +
                total.sulfur.dioxide + chlorides + pH , red_wine_df)
glance(gof_fs_rw)
```

```
## # A tibble: 1 x 12
##   r.squared adj.r.squared sigma statistic   p.value    df logLik   AIC   BIC
##       <dbl>         <dbl> <dbl>     <dbl>     <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     0.357         0.355 0.649      147. 7.12e-149     6 -1573. 3163. 3206.
## # ... with 3 more variables: deviance <dbl>, df.residual <int>, nobs <int>
```

```
summary(gof_fs_rw)
```

```
##
## Call:
## lm(formula = quality ~ alcohol + volatile.acidity + sulphates +
##     total.sulfur.dioxide + chlorides + pH, data = red_wine_df)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.60575 -0.35883 -0.04806  0.46079  1.95643
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          4.2957316  0.3995603  10.751  < 2e-16 ***
## alcohol              0.2906738  0.0168108  17.291  < 2e-16 ***
## volatile.acidity    -1.0381945  0.1004270 -10.338  < 2e-16 ***
## sulphates            0.8886802  0.1100419   8.076 1.31e-15 ***
## total.sulfur.dioxide -0.0023721  0.0005064  -4.684 3.05e-06 ***
## chlorides           -2.0022839  0.3980757  -5.030 5.46e-07 ***
## pH                  -0.4351830  0.1160368  -3.750 0.000183 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Residual standard error: 0.6487 on 1592 degrees of freedom
## Multiple R-squared:  0.3572, Adjusted R-squared:  0.3548
## F-statistic: 147.4 on 6 and 1592 DF,  p-value: < 2.2e-16
```

Best fit model:

- quality ~ alcohol + volatile.acidity + sulphates + total.sulfur.dioxide + chlorides + pH

Stepwise Model Selection for white wine

```
#Partition of data
white_wine_std <- data.frame(scale(white_wine_df[1:11]))
white_wine_std$quality <- white_wine_df$quality
set.seed(10)
partition_ww <- resample_partition(white_wine_std,
                                p=c(train=0.5,
                                    valid=0.25,
                                    test=0.25))
```

```
model_ww <- NULL

preds <- "1"
cands <- c("fixed.acidity","volatile.acidity","citric.acid","residual.sugar",
           "chlorides","free.sulfur.dioxide","total.sulfur.dioxide","density",
           "pH","sulphates","alcohol")
s2 <- step("quality", preds, cands, partition_ww)

model_ww <- c(model_ww, attr(s2, "best"))
s2
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##            0.9046956            0.9040842            0.9115613
##       residual.sugar            chlorides  free.sulfur.dioxide
##            0.9083983            0.8910796            0.9115526
## total.sulfur.dioxide              density                   pH
##            0.8967295            0.8741188            0.9066175
##            sulphates              alcohol
##            0.9097914            0.8153880
## attr(,"best")
##  alcohol
## 0.815388
```

```
preds <- "alcohol"
cands <- c("fixed.acidity","volatile.acidity","citric.acid","residual.sugar",
           "chlorides","free.sulfur.dioxide","total.sulfur.dioxide","density",
           "pH","sulphates")
s2 <- step("quality", preds, cands, partition_ww)
```

```r
model_ww <- c(model_ww, attr(s2, "best"))
s2
```

```
##        fixed.acidity     volatile.acidity          citric.acid
##            0.8129728            0.8013749            0.8146745
##       residual.sugar            chlorides  free.sulfur.dioxide
##            0.8095793            0.8142981            0.8090157
## total.sulfur.dioxide              density                   pH
##            0.8153080            0.8141830            0.8140691
##            sulphates
##            0.8129943
## attr(,"best")
## volatile.acidity
##        0.8013749
```

```r
preds <- c("alcohol","volatile.acidity")
cands <- c("fixed.acidity","citric.acid","residual.sugar",
           "chlorides","free.sulfur.dioxide","total.sulfur.dioxide","density",
           "pH","sulphates")
s2 <- step("quality", preds, cands, partition_ww)

model_ww <- c(model_ww, attr(s2, "best"))
s2
```

```
##        fixed.acidity          citric.acid       residual.sugar
##            0.7986474            0.8022197            0.7901090
##            chlorides  free.sulfur.dioxide total.sulfur.dioxide
##            0.8008642            0.7956051            0.8001968
##              density                   pH            sulphates
##            0.7969973            0.8006140            0.7994550
## attr(,"best")
## residual.sugar
##       0.790109
```

```r
preds <- c("alcohol","volatile.acidity","residual.sugar")
cands <- c("fixed.acidity","citric.acid",
           "chlorides","free.sulfur.dioxide","total.sulfur.dioxide","density",
           "pH","sulphates")
s2 <- step("quality", preds, cands, partition_ww)

model_ww <- c(model_ww, attr(s2, "best"))
s2
```

```
##        fixed.acidity          citric.acid            chlorides
##            0.7866903            0.7912216            0.7899837
##  free.sulfur.dioxide total.sulfur.dioxide              density
##            0.7870588            0.7900478            0.7885526
##                   pH            sulphates
##            0.7880903            0.7879418
## attr(,"best")
## fixed.acidity
##     0.7866903
```

```r
preds <- c("alcohol","volatile.acidity","residual.sugar","fixed.acidity")
cands <- c("citric.acid","chlorides","free.sulfur.dioxide",
           "total.sulfur.dioxide","density","pH","sulphates")
s2 <- step("quality", preds, cands, partition_ww)

model_ww <- c(model_ww, attr(s2, "best"))
s2
```

```
##          citric.acid            chlorides  free.sulfur.dioxide 
##            0.7875729            0.7865390            0.7842173 
## total.sulfur.dioxide              density                   pH 
##            0.7865822            0.7873280            0.7861562 
##            sulphates 
##            0.7845328 
## attr(,"best")
## free.sulfur.dioxide 
##           0.7842173
```

```r
preds <- c("alcohol","volatile.acidity","residual.sugar","fixed.acidity",
           "free.sulfur.dioxide")
cands <- c("citric.acid","chlorides",
           "total.sulfur.dioxide","density","pH","sulphates")
s2 <- step("quality", preds, cands, partition_ww)

model_ww <- c(model_ww, attr(s2, "best"))
s2
```

```
##          citric.acid            chlorides total.sulfur.dioxide 
##            0.7850955            0.7840036            0.7837253 
##              density                   pH            sulphates 
##            0.7851699            0.7837374            0.7822481 
## attr(,"best")
## sulphates 
## 0.7822481
```

```r
preds <- c("alcohol","volatile.acidity","residual.sugar","fixed.acidity",
           "free.sulfur.dioxide","sulphates")
cands <- c("citric.acid","chlorides",
           "total.sulfur.dioxide","density","pH")
s2 <- step("quality", preds, cands, partition_ww)

model_ww <- c(model_ww, attr(s2, "best"))
s2
```

```
##          citric.acid            chlorides total.sulfur.dioxide 
##            0.7831977            0.7820405            0.7813755 
##              density                   pH 
##            0.7827787            0.7820393 
## attr(,"best")
## total.sulfur.dioxide 
##            0.7813755
```

```
preds <- c("alcohol","volatile.acidity","residual.sugar","fixed.acidity",
           "free.sulfur.dioxide","sulphates", "total.sulfur.dioxide")
cands <- c("citric.acid","chlorides",
           "density","pH")
s2 <- step("quality", preds, cands, partition_ww)

model_ww <- c(model_ww, attr(s2, "best"))
s2
```

```
## citric.acid   chlorides     density          pH
##   0.7822786   0.7811962   0.7824629   0.7810240
## attr(,"best")
##       pH
## 0.781024
```

```
preds <- c("alcohol","volatile.acidity","residual.sugar","fixed.acidity",
           "free.sulfur.dioxide","sulphates", "total.sulfur.dioxide","pH")
cands <- c("citric.acid","chlorides","density")
s2 <- step("quality", preds, cands, partition_ww)

model_ww <- c(model_ww, attr(s2, "best"))
s2
```

```
## citric.acid   chlorides     density
##   0.7819109   0.7808872   0.7861522
## attr(,"best")
## chlorides
## 0.7808872
```

```
preds <- c("alcohol","volatile.acidity","residual.sugar","fixed.acidity",
           "free.sulfur.dioxide","sulphates", "total.sulfur.dioxide",
           "pH","chlorides")
cands <- c("citric.acid","density")
s2 <- step("quality", preds, cands, partition_ww)

model_ww <- c(model_ww, attr(s2, "best"))
s2
```

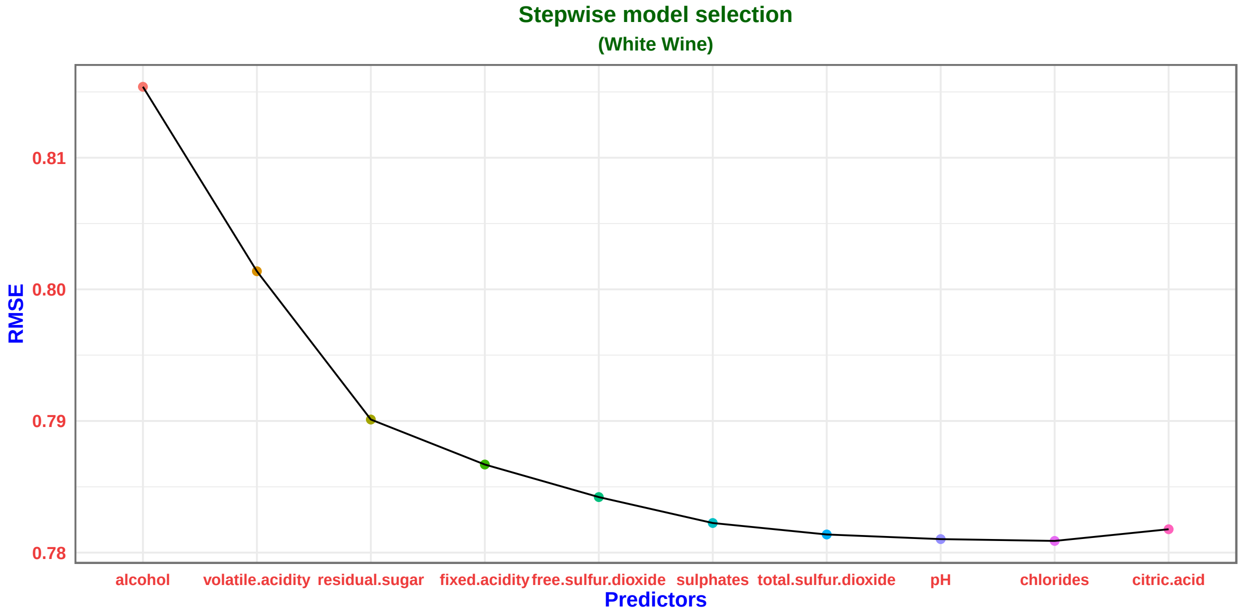
```
## citric.acid     density
##   0.7817750   0.7866456
## attr(,"best")
## citric.acid
##    0.781775
```

Model stopped improving at:

- fit = quality ~ alcohol + volatile acidity + residual sugar + fixed acidity + free sulfur dioxide + sulphates + total sulfur dioxide + pH + chlorides

Visualizing how adding each variable affects the RMSE.

```r
step_model_ww <- tibble(index=seq_along(model_ww),
                        variable=factor(names(model_ww), levels=names(model_ww)),
                        RMSE=model_ww)

ggplot(step_model_ww, aes(y=RMSE)) +
  geom_point(aes(x=variable, color = variable), size = 2) +
  geom_line(aes(x=index)) +
  theme_minimal()+
  labs(title = "Stepwise model selection",
               subtitle = "(White Wine)",
               x = "Predictors",
               y = "RMSE")+
        theme(plot.title = element_text(hjust = 0.5,
                                        color = "darkgreen",
                                        face = "bold"),
              plot.subtitle = element_text(hjust = 0.5,
                                        color = "darkgreen",
                                        face = "bold"),
              axis.title = element_text(color = "blue",
                                        face = "bold",
                                        size = 12),
              axis.text.x = element_text(hjust = 0.5,
                                        vjust = 0.5,
                                        color = "brown2",
                                        face = "bold",
                                        size = 9),
              axis.text.y = element_text(color = "brown2",
                                        face = "bold",
                                        size = 10),
              panel.border = element_rect(colour = "grey45",
                                        fill=NA, size=1),
              legend.position = "none")
```

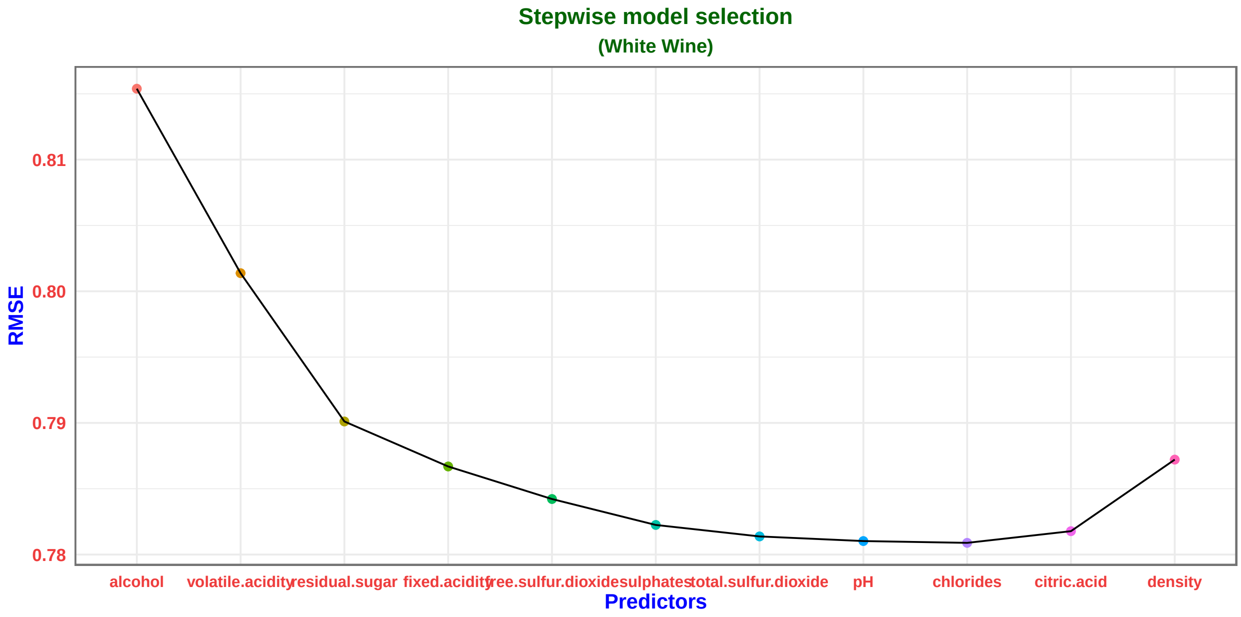


visualizing how each variable(including the predictors) affects the RMSE for Red wine

```
temp_model_ww <- model_ww
preds <- c("alcohol","volatile.acidity","residual.sugar","fixed.acidity",
           "free.sulfur.dioxide","sulphates", "total.sulfur.dioxide",
           "pH","chlorides", "citric.acid")
cands <- c("density")
s2 <- step("quality", preds, cands, partition_ww)

temp_model_ww <- c(temp_model_ww, attr(s2, "best"))
s2
```

```
##   density
## 0.7872114
## attr(,"best")
##   density
## 0.7872114
```

```
step_model_ww_temp <- tibble(index=seq_along(temp_model_ww),
                     variable=factor(names(temp_model_ww), levels=names(temp_model_ww)),
                     RMSE=temp_model_ww)

ggplot(step_model_ww_temp, aes(y=RMSE)) +
  geom_point(aes(x=variable, color = variable), size = 2) +
  geom_line(aes(x=index)) +
  theme_minimal()+
  labs(title = "Stepwise model selection",
               subtitle = "(White Wine)",
               x = "Predictors",
               y = "RMSE")+
        theme(plot.title = element_text(hjust = 0.5,
                                        color = "darkgreen",
                                        face = "bold"),
              plot.subtitle = element_text(hjust = 0.5,
                                        color = "darkgreen",
                                        face = "bold"),
              axis.title = element_text(color = "blue",
                                        face = "bold",
                                        size = 12),
              axis.text.x = element_text(hjust = 0.5,
                                        vjust = 0.5,
                                        color = "brown2",
                                        face = "bold",
                                        size = 9),
              axis.text.y = element_text(color = "brown2",
                                        face = "bold",
                                        size = 10),
              panel.border = element_rect(colour = "grey45",
                                        fill=NA, size=1),
              legend.position = "none")
```



Consider the following fits and extract the best fit model:

- fit1 <- quality ~ alcohol + volatile.acidity + residual.sugar + fixed.acidity + free.sulfur.dioxide + sulphates + total.sulfur.dioxide + pH
- fit2 <- quality ~ alcohol + volatile.acidity + residual.sugar + fixed.acidity + free.sulfur.dioxide + sulphates + total.sulfur.dioxide + pH + chlorides

Cross Validation

```
set.seed(2020)
#partition_ww_train <- white_wine_std[-partition_ww$test$idx,]
#whitewine_cv <- crossv_kfold(partition_ww_train, 5)
#whitewine_cv

whitewine_cv <- crossv_kfold(white_wine_std, 5)
whitewine_cv
```

```
## # A tibble: 5 x 3
##   train                    test                   .id
##   <named list>             <named list>           <chr>
## 1 <resample [3,918 x 12]> <resample [980 x 12]> 1
## 2 <resample [3,918 x 12]> <resample [980 x 12]> 2
## 3 <resample [3,918 x 12]> <resample [980 x 12]> 3
## 4 <resample [3,919 x 12]> <resample [979 x 12]> 4
## 5 <resample [3,919 x 12]> <resample [979 x 12]> 5
```

```
#Calculating RMSE for each fold of data
cv_rw <- whitewine_cv %>%
  mutate(fit = purrr::map(train,
                   ~ lm(quality ~ alcohol + volatile.acidity + residual.sugar +
                             fixed.acidity + free.sulfur.dioxide +
                             sulphates + total.sulfur.dioxide + pH, data = .)),
         rmse = purrr::map2_dbl(fit, test, ~ rmse(.x, .y)))

cv_rw
```

```
## # A tibble: 5 x 5
##   train                   test                  .id   fit           rmse
##   <named list>            <named list>          <chr> <named list> <dbl>
## 1 <resample [3,918 x 12]> <resample [980 x 12]> 1     <lm>         0.747
## 2 <resample [3,918 x 12]> <resample [980 x 12]> 2     <lm>         0.806
## 3 <resample [3,918 x 12]> <resample [980 x 12]> 3     <lm>         0.743
## 4 <resample [3,919 x 12]> <resample [979 x 12]> 4     <lm>         0.727
## 5 <resample [3,919 x 12]> <resample [979 x 12]> 5     <lm>         0.762
```

```
#Average of RMSEs
mean(cv_rw$rmse)
```

```
## [1] 0.7568375
```

Comparing models using CV

```
do_whitewine_cv <- function(formula) {
  whitewine_cv %>%
    mutate(fit = map(train,
                     ~ lm(formula, data = .)),
         rmse = map2_dbl(fit, test, ~ rmse(.x, .y)),
         rsq = map2_dbl(fit,test, ~rsquare(.x,.y)),
         mae = map2_dbl(fit,test,~mae(.x,.y))) %>%
    summarize(cv_rmse = mean(rmse), cv_rsq = mean(rsq),
              cv_mae = mean(mae)) %>%
    return(c(cv_rmse, cv_rsq, cv_mae))
}
```

Calling the function

```
fit1_rmse_ww <- do_whitewine_cv(quality ~ alcohol + volatile.acidity + residual.sugar +
                             fixed.acidity + free.sulfur.dioxide +
                             sulphates + total.sulfur.dioxide + pH)

fit2_rmse_ww <- do_whitewine_cv(quality ~ alcohol + volatile.acidity + residual.sugar +
                             fixed.acidity + free.sulfur.dioxide +
                             sulphates + total.sulfur.dioxide + pH + chlorides)

fit1_rmse_ww
```

```
## # A tibble: 1 x 3
##   cv_rmse cv_rsq cv_mae
##     <dbl>  <dbl>  <dbl>
## 1   0.757  0.269  0.588
```

```
fit2_rmse_ww
```

```
## # A tibble: 1 x 3
##   cv_rmse cv_rsq cv_mae
##     <dbl>  <dbl>  <dbl>
## 1   0.757  0.269  0.588
```

Best fit Model

- quality ~ alcohol + volatile.acidity + residual.sugar + fixed.acidity + free.sulfur.dioxide + sulphates + total.sulfur.dioxide + pH

```
fit_rw <- lm(quality ~ alcohol + volatile.acidity + sulphates + total.sulfur.dioxide + chlorides + pH, 

summary(fit_rw)
```

```
##
## Call:
## lm(formula = quality ~ alcohol + volatile.acidity + sulphates +
##     total.sulfur.dioxide + chlorides + pH, data = red_wine_std)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.60575 -0.35883 -0.04806  0.46079  1.95643
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           5.63602    0.01622 347.419  < 2e-16 ***
## alcohol               0.30976    0.01791  17.291  < 2e-16 ***
## volatile.acidity     -0.18590    0.01798 -10.338  < 2e-16 ***
## sulphates             0.15064    0.01865   8.076 1.31e-15 ***
## total.sulfur.dioxide -0.07803    0.01666  -4.684 3.05e-06 ***
## chlorides            -0.09424    0.01874  -5.030 5.46e-07 ***
## pH                   -0.06719    0.01791  -3.750 0.000183 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6487 on 1592 degrees of freedom
## Multiple R-squared:  0.3572, Adjusted R-squared:  0.3548
## F-statistic: 147.4 on 6 and 1592 DF,  p-value: < 2.2e-16
```

```
fit_ww <- lm(quality ~ alcohol + volatile.acidity + residual.sugar + fixed.acidity + free.sulfur.dioxid

summary(fit_ww)
```

```
##
## Call:
## lm(formula = quality ~ alcohol + volatile.acidity + residual.sugar +
##     fixed.acidity + free.sulfur.dioxide + sulphates + total.sulfur.dioxide +
##     pH, data = white_wine_std)
##
```

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8931 -0.4982 -0.0358  0.4644  3.1821
##
## Coefficients:
##                      Estimate Std. Error t value Pr(>|t|)
## (Intercept)           5.87791    0.01080 544.038  < 2e-16 ***
## alcohol               0.45433    0.01304  34.834  < 2e-16 ***
## volatile.acidity     -0.19887    0.01126 -17.665  < 2e-16 ***
## residual.sugar        0.13274    0.01283  10.349  < 2e-16 ***
## fixed.acidity        -0.04222    0.01220  -3.461 0.000542 ***
## free.sulfur.dioxide   0.08078    0.01427   5.662 1.58e-08 ***
## sulphates             0.04736    0.01108   4.273 1.97e-05 ***
## total.sulfur.dioxide -0.03866    0.01581  -2.445 0.014529 *
## pH                    0.02708    0.01240   2.184 0.029034 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.7561 on 4889 degrees of freedom
## Multiple R-squared:  0.2722, Adjusted R-squared:  0.2711
## F-statistic: 228.6 on 8 and 4889 DF,  p-value: < 2.2e-16
```

```r
#Data partition

#Redwine
set.seed(1)
train <- createDataPartition(red_wine_std$quality, p=0.6, list=FALSE)

table(red_wine_std$quality[train])
```

```
##
##   3   4   5   6   7   8
##   5  35 407 383 119  12
```

```r
reddf_train <- red_wine_std[as.integer(train),]
reddf_test <- red_wine_std[-as.integer(train),]

#Whitewine
set.seed(1)
train_w <- createDataPartition(white_wine_std$quality, p=0.6, list=FALSE)

table(white_wine_std$quality[train_w])
```

```
##
##    3    4    5    6    7    8    9
##   14  101  869 1319  518  113    5
```

```r
whitedf_train <- white_wine_std[as.integer(train_w),]
whitedf_test <- white_wine_std[-as.integer(train_w),]
```

Random Forest Regression

```
#Red wine
fit_rw_rf <- randomForest(quality ~ alcohol + volatile.acidity +
                            sulphates + total.sulfur.dioxide +
                            chlorides + pH, reddf_train,
                          mtry = 3,
                          importance = TRUE,
                          na.action = na.omit)
summary(fit_rw_rf)
```

```
##                 Length Class  Mode
## call              6    -none- call
## type              1    -none- character
## predicted       961    -none- numeric
## mse             500    -none- numeric
## rsq             500    -none- numeric
## oob.times       961    -none- numeric
## importance       12    -none- numeric
## importanceSD      6    -none- numeric
## localImportance   0    -none- NULL
## proximity         0    -none- NULL
## ntree             1    -none- numeric
## mtry              1    -none- numeric
## forest           11    -none- list
## coefs             0    -none- NULL
## y               961    -none- numeric
## test              0    -none- NULL
## inbag             0    -none- NULL
## terms             3    terms  call
```

```
fit_rw_rf_prediction<-predict(fit_rw_rf,newdata=reddf_test)
rf_rw_results<-as.data.frame(cbind(fit_rw_rf_prediction,reddf_test$quality))
colnames(rf_rw_results)<-c("prediction","real")
```

```
#White Wine - Random forest regression
fit_ww_rf <- randomForest(quality ~ alcohol + volatile.acidity +
                            residual.sugar + fixed.acidity +
                            free.sulfur.dioxide + sulphates +
                            total.sulfur.dioxide + pH, whitedf_train,
                          mtry = 3,
                          importance = TRUE,
                          na.action = na.omit)
fit_ww_rf
```

```
##
## Call:
##  randomForest(formula = quality ~ alcohol + volatile.acidity +       residual.sugar + fixed.acidity +
##                Type of random forest: regression
##                      Number of trees: 500
## No. of variables tried at each split: 3
##
```

```
##           Mean of squared residuals: 0.4264972
##                     % Var explained: 47.14
```

```
fit_ww_rf_prediction<-predict(fit_ww_rf,newdata=whitedf_test)
rf_ww_results<-as.data.frame(cbind(fit_ww_rf_prediction,whitedf_test$quality))
colnames(rf_ww_results)<-c("prediction","real")
```

Ridge Regression

```
#Ridge regression for red wine


ctrl <- trainControl(method="repeatedcv", number=10, repeats=10)

grd <- expand.grid(lambda=exp(seq(from=-7, to=-2, length.out=20)),
                   alpha=0)

set.seed(1)
fit_rw_ridge <- train(quality ~ alcohol + volatile.acidity +
                            sulphates + total.sulfur.dioxide +
                            chlorides + pH, data=reddf_train,
              method="glmnet",
              preProcess=c("center", "scale"),
              trControl=ctrl, tuneGrid=grd)
fit_rw_ridge
```

```
## glmnet
##
## 961 samples
##   6 predictor
##
## Pre-processing: centered (6), scaled (6)
## Resampling: Cross-Validated (10 fold, repeated 10 times)
## Summary of sample sizes: 866, 865, 865, 865, 865, 865, ...
## Resampling results across tuning parameters:
##
##   lambda       RMSE       Rsquared   MAE
##   0.000911882  0.6608113  0.3426984  0.5170648
##   0.001186388  0.6608113  0.3426984  0.5170648
##   0.001543529  0.6608113  0.3426984  0.5170648
##   0.002008180  0.6608113  0.3426984  0.5170648
##   0.002612707  0.6608113  0.3426984  0.5170648
##   0.003399216  0.6608113  0.3426984  0.5170648
##   0.004422489  0.6608113  0.3426984  0.5170648
##   0.005753800  0.6608113  0.3426984  0.5170648
##   0.007485879  0.6608113  0.3426984  0.5170648
##   0.009739369  0.6608113  0.3426984  0.5170648
##   0.012671232  0.6608113  0.3426984  0.5170648
##   0.016485680  0.6608113  0.3426984  0.5170648
##   0.021448399  0.6608113  0.3426984  0.5170648
##   0.027905057  0.6608113  0.3426984  0.5170648
##   0.036305375  0.6608116  0.3426984  0.5170653
##   0.047234459  0.6608796  0.3427288  0.5174779
##   0.061453549  0.6609935  0.3427710  0.5180766
##   0.079953042  0.6612124  0.3428183  0.5188810
##   0.104021477  0.6616069  0.3428619  0.5199677
##   0.135335283  0.6622773  0.3428978  0.5214328
##
## Tuning parameter 'alpha' was held constant at a value of 0
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were alpha = 0 and lambda = 0.02790506.
```

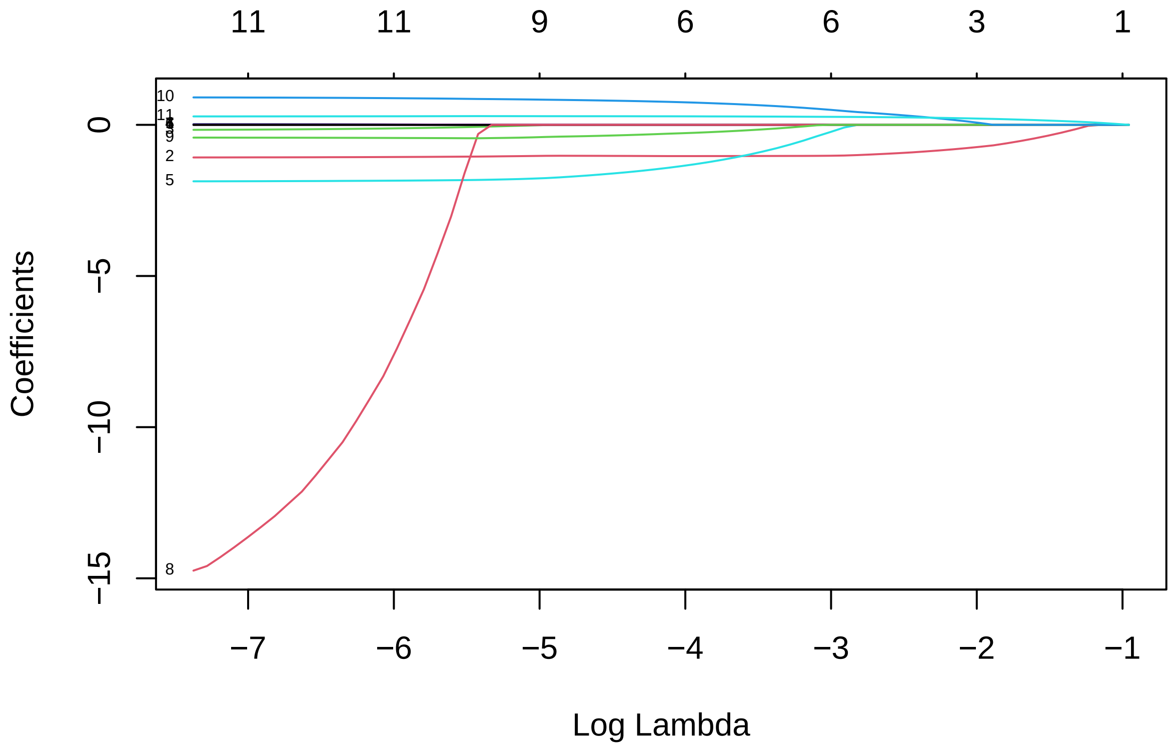
```
#Ridge regression for white wine


ctrl <- trainControl(method="repeatedcv", number=10, repeats=10)

grd <- expand.grid(lambda=exp(seq(from=-7, to=-2, length.out=20)),
                   alpha=0)

set.seed(1)
fit_ww_ridge <- train(quality ~ alcohol + volatile.acidity +
                            residual.sugar + fixed.acidity +
                            free.sulfur.dioxide + sulphates +
                            total.sulfur.dioxide + pH, data=whitedf_train,
              method="glmnet",
              preProcess=c("center", "scale"),
              trControl=ctrl, tuneGrid=grd)
fit_ww_ridge
```

```
## glmnet
##
## 2939 samples
##    8 predictor
##
## Pre-processing: centered (8), scaled (8)
## Resampling: Cross-Validated (10 fold, repeated 10 times)
## Summary of sample sizes: 2646, 2645, 2644, 2644, 2645, 2645, ...
## Resampling results across tuning parameters:
##
##   lambda       RMSE       Rsquared   MAE
##   0.000911882  0.7713956  0.2644750  0.5963063
##   0.001186388  0.7713956  0.2644750  0.5963063
##   0.001543529  0.7713956  0.2644750  0.5963063
##   0.002008180  0.7713956  0.2644750  0.5963063
##   0.002612707  0.7713956  0.2644750  0.5963063
##   0.003399216  0.7713956  0.2644750  0.5963063
##   0.004422489  0.7713956  0.2644750  0.5963063
##   0.005753800  0.7713956  0.2644750  0.5963063
##   0.007485879  0.7713956  0.2644750  0.5963063
##   0.009739369  0.7713956  0.2644750  0.5963063
##   0.012671232  0.7713956  0.2644750  0.5963063
##   0.016485680  0.7713956  0.2644750  0.5963063
##   0.021448399  0.7713956  0.2644750  0.5963063
##   0.027905057  0.7713956  0.2644750  0.5963063
##   0.036305375  0.7713956  0.2644750  0.5963063
##   0.047234459  0.7715800  0.2644133  0.5964922
##   0.061453549  0.7719637  0.2642660  0.5968524
##   0.079953042  0.7725760  0.2640283  0.5973675
##   0.104021477  0.7735260  0.2636500  0.5981261
##   0.135335283  0.7749544  0.2630627  0.5992021
##
## Tuning parameter 'alpha' was held constant at a value of 0
## RMSE was used to select the optimal model using the smallest value.
## The final values used for the model were alpha = 0 and lambda = 0.03630538.
```

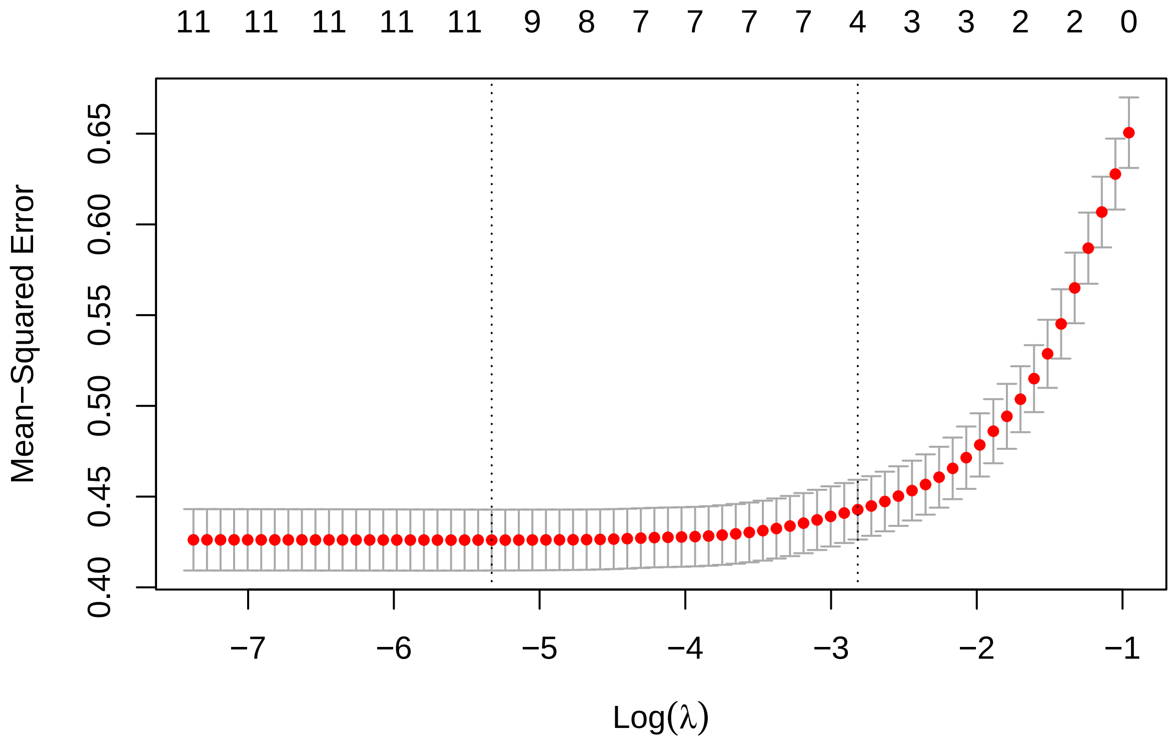
Lasso Regression

Lasso Regression

```
#Lasso regression for red wine
actual_quality_rw<-red_wine_df$quality
response_rw<-red_wine_df$quality
predictors_rw<-data.matrix(red_wine_df[,c("fixed.acidity","volatile.acidity","citric.acid",
                           "residual.sugar","chlorides","free.sulfur.dioxide",
                           "total.sulfur.dioxide","density","pH","sulphates",
                           "alcohol")])


red_ls_model<-glmnet(predictors_rw,response_rw,alpha=1)

coef_rw<-plot(red_ls_model,xvar="lambda",label=TRUE)
```



```
# Using k-fold cv to get best optimal lambda value
red_cv_ls_model<-cv.glmnet(predictors_rw,response_rw,alpha=1)

#produce plot of MSE values for each lambda
plot(red_cv_ls_model)
```



```
#find optimal lambda value that minimizes the MSE
best_lambda_rw<-red_cv_ls_model$lambda.min
best_lambda_rw
```

```
## [1] 0.004850794
```

```
best_model_ls_rw<-glmnet(predictors_rw,response_rw,alpha=1,lambda=best_lambda_rw)
coef(best_model_ls_rw)
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                s0
## (Intercept)           4.311058572
## fixed.acidity         0.001590786
## volatile.acidity     -1.045885442
## citric.acid          -0.056043048
## residual.sugar        0.004545097
## chlorides            -1.812127522
## free.sulfur.dioxide   0.003379133
## total.sulfur.dioxide -0.002981089
## density               .
## pH                   -0.437661346
## sulphates             0.850821345
## alcohol               0.287823262
```

```r
# To find the rsquare of our best model

#To obtain predicted values
predicted_quality_rw<-predict(best_model_ls_rw,s=best_lambda_rw,newx=predictors_rw)

#Finding the SST and SSE and Rsquare
sst_rw<-sum((actual_quality_rw-mean(actual_quality_rw))^2)
sse_rw<-sum((actual_quality_rw-predicted_quality_rw)^2)
rsq_ls_rw<-1-(sse_rw/sst_rw)
rsq_ls_rw
```

```
## [1] 0.3595207
```

```r
#Finding the RMSE and MAE
n_rw<-nrow(red_wine_df)
rmse_ls_rw<-sqrt(sum((actual_quality_rw-predicted_quality_rw)^2)/n_rw)
rmse_ls_rw
```

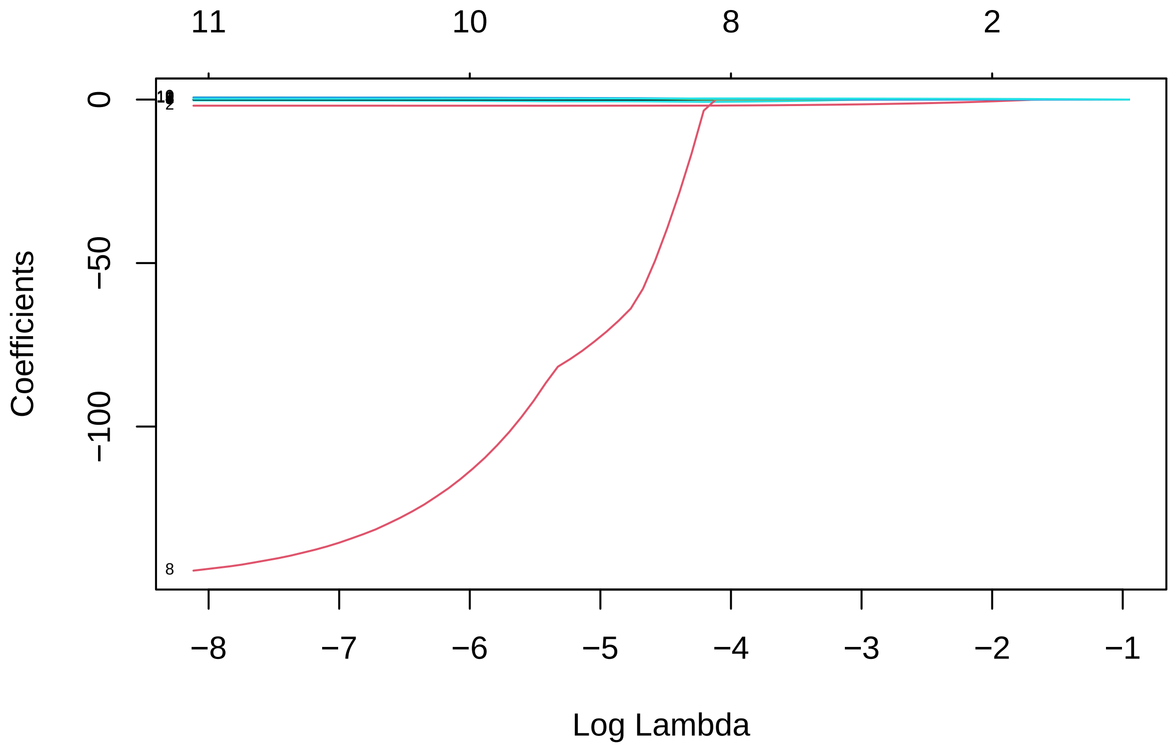
```
## [1] 0.6460953
```

```r
mae_ls_rw<-sum(abs(actual_quality_rw-predicted_quality_rw))/n_rw
mae_ls_rw
```

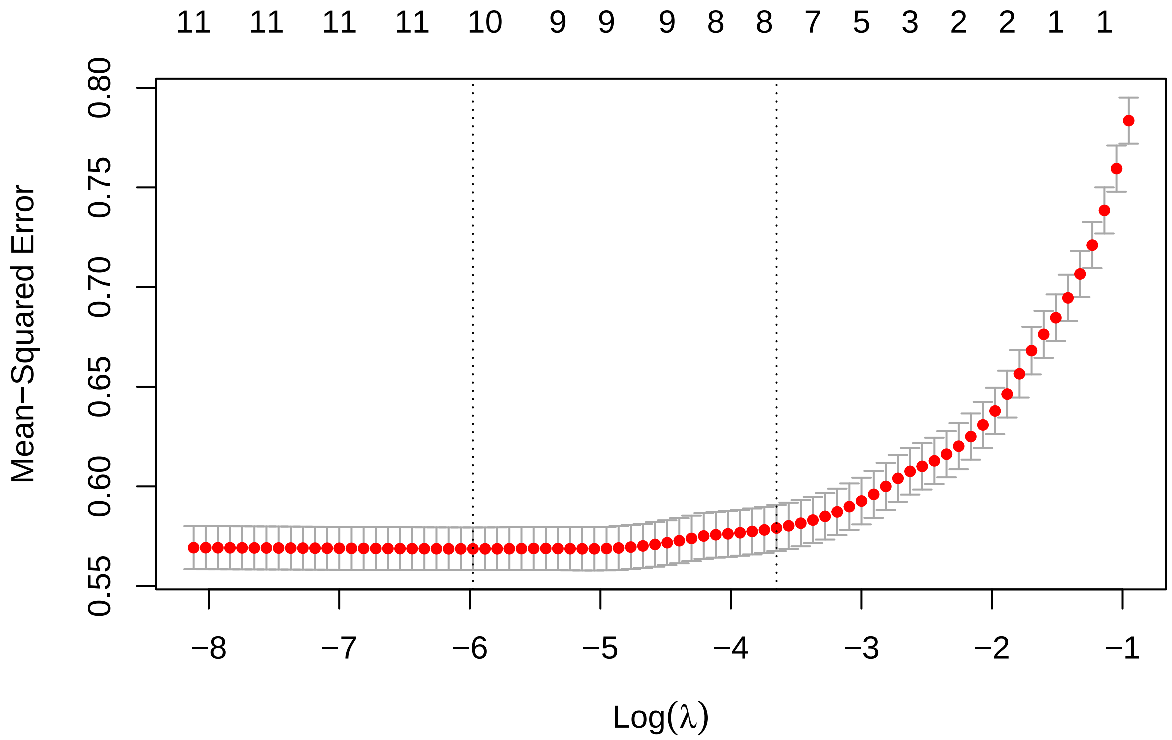
```
## [1] 0.5020461
```

```r
view(data.frame(RMSE=rmse_ls_rw,MAE=mae_ls_rw,R_Square=rsq_ls_rw))
```

```r
# Lasso Regression on White wine
response_ww<-white_wine_df$quality
actual_quality_ww<-white_wine_df$quality
predictors_ww<-data.matrix(white_wine_df[,c("fixed.acidity","volatile.acidity","citric.acid",
                            "residual.sugar","chlorides","free.sulfur.dioxide",
                            "total.sulfur.dioxide","density","pH","sulphates",
                            "alcohol")])


white_ls_model<-glmnet(predictors_ww,response_ww,alpha=1)

coef_ww<-plot(white_ls_model,xvar="lambda",label=TRUE)
```



```
# Using k-fold cv to get best optimal lambda value
library("glmnet")
white_cv_ls_model<-cv.glmnet(predictors_ww,response_ww,alpha=1)

plot(white_cv_ls_model)
```



```
#find optimal lambda value that minimizes the MSE
best_lambda_ww<-white_cv_ls_model$lambda.min
best_lambda_ww
```

```
## [1] 0.002537796
```

```
best_model_ls_ww<-glmnet(predictors_ww,response_ww,alpha=1,lambda=best_lambda_ww)
coef(best_model_ls_ww)
```

```
## 12 x 1 sparse Matrix of class "dgCMatrix"
##                                 s0
## (Intercept)           1.135530e+02
## fixed.acidity         3.015296e-02
## volatile.acidity     -1.875380e+00
## citric.acid           .
## residual.sugar        6.637850e-02
## chlorides            -3.934651e-01
## free.sulfur.dioxide   3.599613e-03
## total.sulfur.dioxide -2.344057e-04
## density              -1.129171e+02
## pH                    5.208469e-01
## sulphates             5.556340e-01
## alcohol               2.329256e-01
```

```
# To find the rsquare of our best model

#To obtain predicted values
predicted_quality_ww<-predict(best_model_ls_ww,s=best_lambda_ww,newx=predictors_ww)

#find the SST and SSE
sst_ww<-sum((actual_quality_ww-mean(actual_quality_ww))^2)
sse_ww<-sum((actual_quality_ww-predicted_quality_ww)^2)

rsq_ls_ww<-1-(sse_ww/sst_ww)
rsq_ls_ww
```

```
## [1] 0.2811733
```

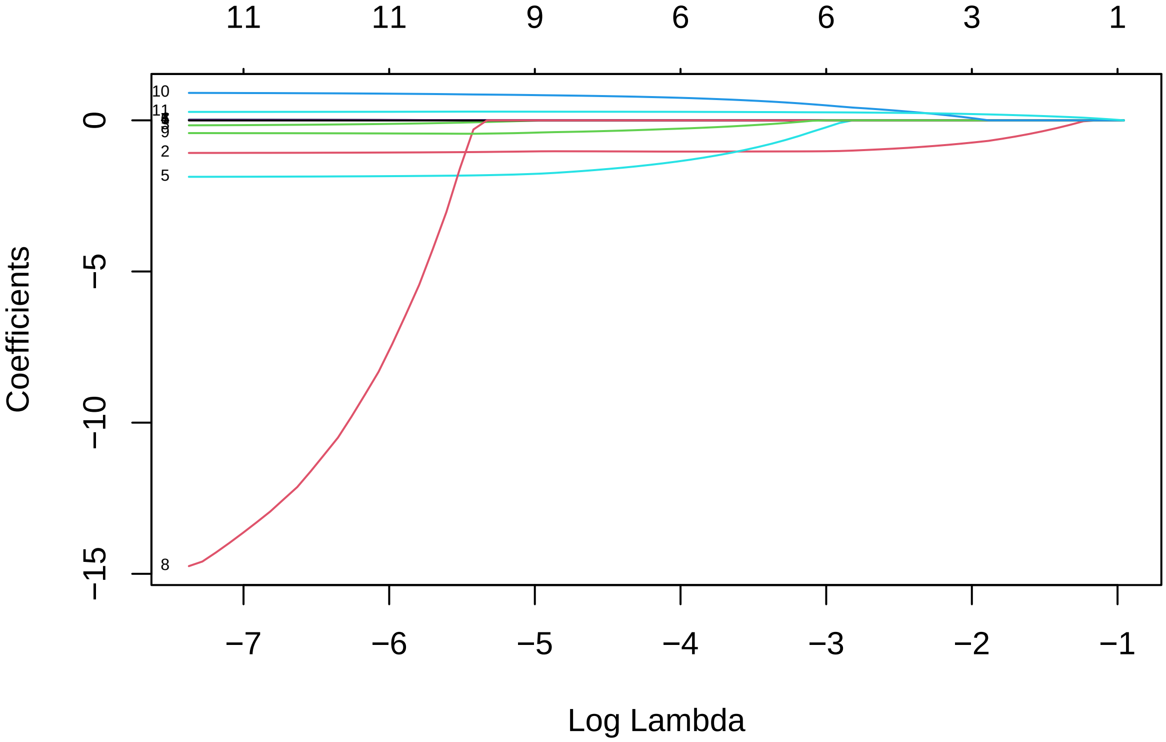
```
n_ww<-nrow(white_wine_df)
rmse_ls_ww<-sqrt(sum((actual_quality_ww-predicted_quality_ww)^2)/n_ww)
rmse_ls_ww
```

```
## [1] 0.7508001
```

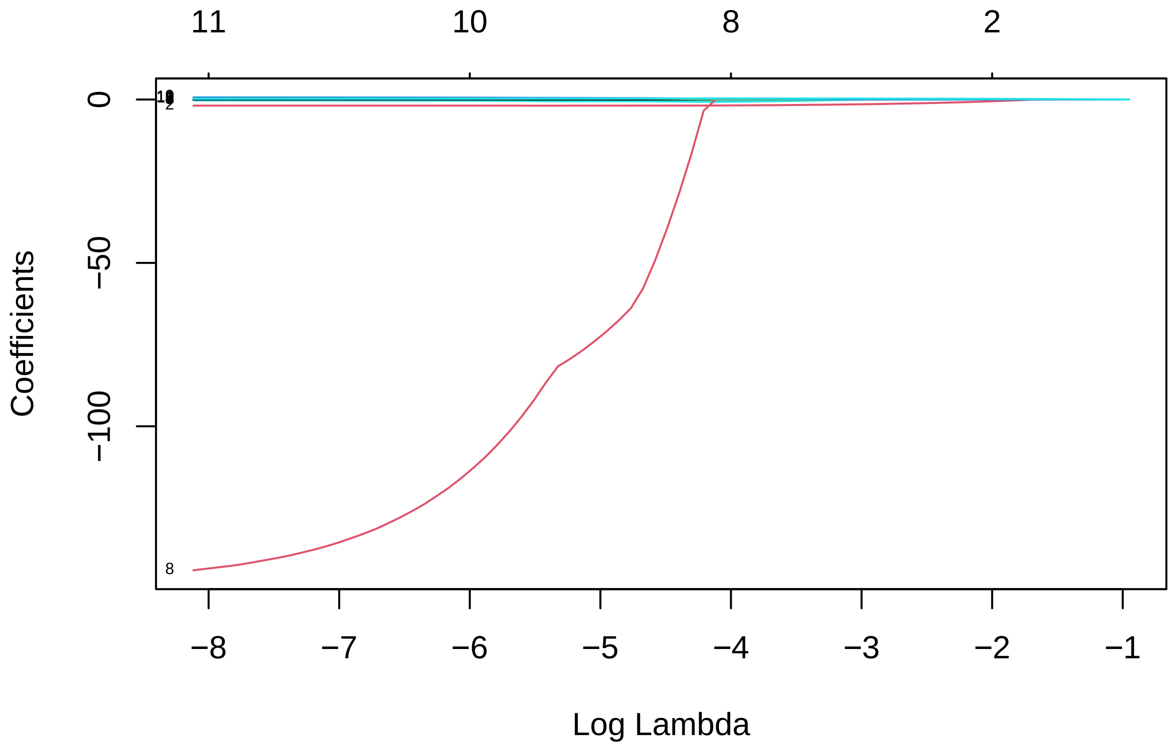
```
mae_ls_ww<-sum(abs(actual_quality_ww-predicted_quality_ww))/n_ww
mae_ls_ww
```

```
## [1] 0.5843081
```

```
ls_results_df <- tibble(wine_type = c("Red Wine","White wine"),
                             rmse = c(rmse_ls_rw,rmse_ls_ww),
                             mae = c(mae_ls_rw,rmse_ls_rw),
                             rsquare = c(rsq_ls_rw,rsq_ls_ww))
view(ls_results_df)

plot(red_ls_model,xvar="lambda",label=TRUE)
```



```
plot(white_ls_model,xvar="lambda",label=TRUE)
```



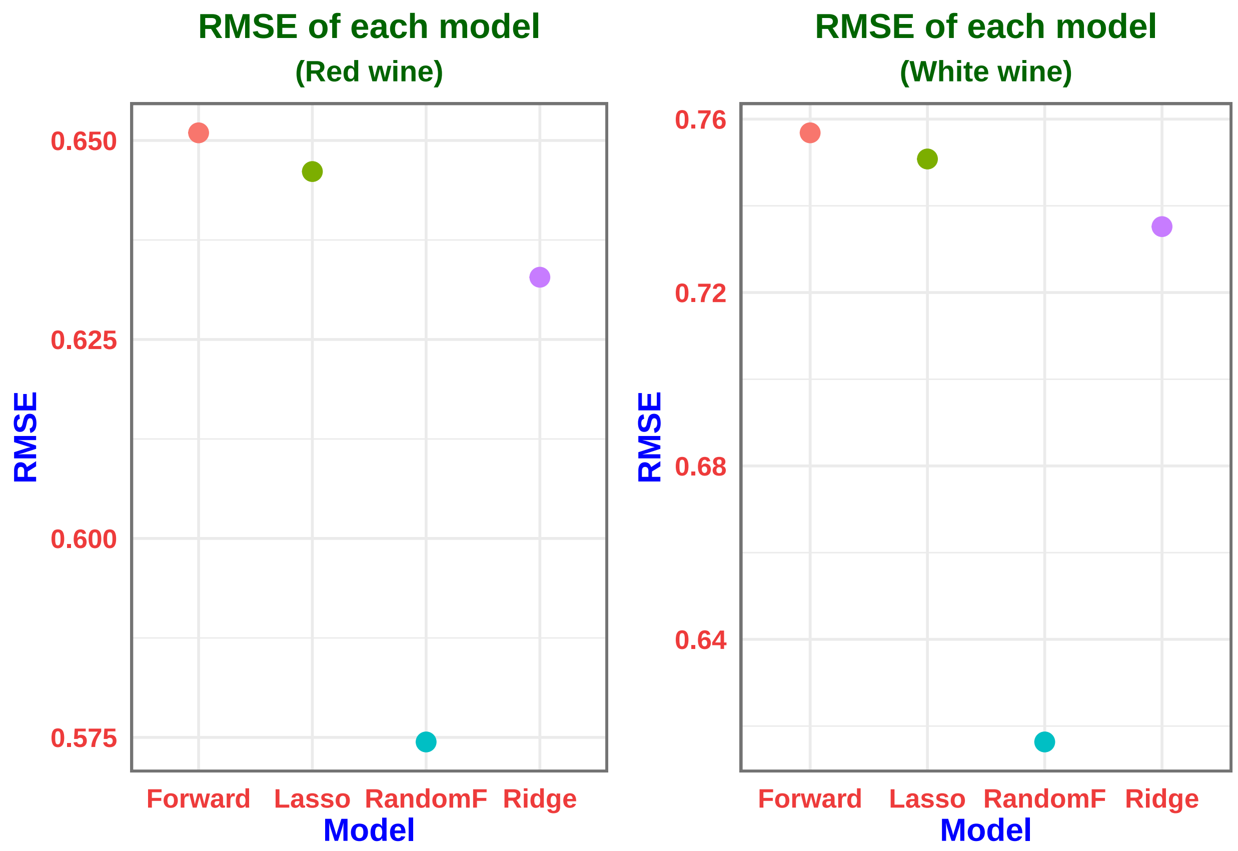
```
#grid.arrange(coef_rw,coef_ww,nrow=1)
```

```
fit_fs_rmse_rw<-as.numeric(fit1_rmse["cv_rmse"])
fit_fs_rsq_rw<-as.numeric(fit1_rmse["cv_rsq"])
fit_fs_rmse_ww<-as.numeric(fit1_rmse_ww["cv_rmse"])
fit_fs_rsq_ww<-as.numeric(fit1_rmse_ww["cv_rsq"])
fit_fs_mae_rw<-as.numeric(fit1_rmse["cv_mae"])
fit_fs_mae_ww<-as.numeric(fit1_rmse_ww["cv_mae"])

fit_rd_rmse_rw<-rmse(fit_rw_ridge,reddf_test)
fit_rd_rsq_rw<-rsquare(fit_rw_ridge,reddf_test)
fit_rd_rmse_ww<-rmse(fit_ww_ridge,whitedf_test)
fit_rd_rsq_ww<-rsquare(fit_ww_ridge,whitedf_test)
fit_rd_mae_rw<-mae(fit_rw_ridge,reddf_test)
fit_rd_mae_ww<-mae(fit_ww_ridge,whitedf_test)

fit_ls_rmse_rw<-rmse_ls_rw
fit_ls_rsq_rw<-rsq_ls_rw
fit_ls_rmse_ww<-rmse_ls_ww
fit_ls_rsq_ww<-rsq_ls_ww
fit_ls_mae_rw<-mae_ls_rw
fit_ls_mae_ww<-mae_ls_ww

fit_rf_rmse_rw<-rmse(fit_rw_rf,reddf_test)
fit_rf_rsq_rw<- rsquare(fit_rw_rf,reddf_test)
```

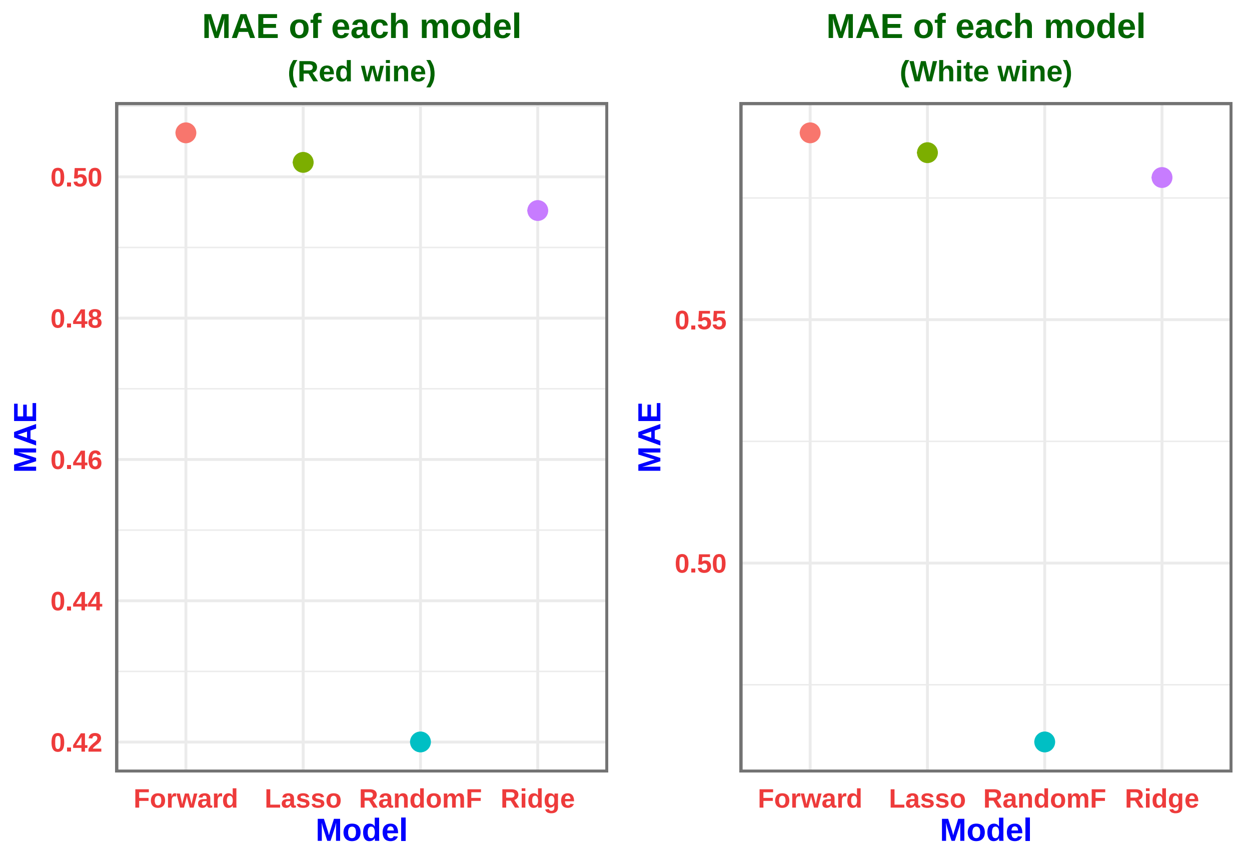
```
fit_rf_rmse_ww<-rmse(fit_ww_rf,whitedf_test)
fit_rf_rsq_ww<- rsquare(fit_ww_rf,whitedf_test)
fit_rf_mae_rw<-mae(fit_rw_rf,reddf_test)
fit_rf_mae_ww<-mae(fit_ww_rf,whitedf_test)
```

```
fs_results_df <- tibble(wine_type = c("Red Wine","White wine"),
                           rmse = c(fit_fs_rmse_rw,fit_fs_rmse_ww),
                           mae = c(fit_fs_mae_rw,fit_fs_mae_ww),
                           rsquare = c(fit_fs_rsq_rw,fit_fs_rsq_ww))
view(fs_results_df)


rd_results_df <- tibble(wine_type = c("Red Wine","White wine"),
                           rmse = c(fit_rd_rmse_rw,fit_rd_rmse_ww),
                           mae = c(fit_rd_mae_rw,fit_rd_mae_ww),
                           rsquare = c(fit_rd_rsq_rw,fit_rd_rsq_ww))
view(rd_results_df)

ls_results_df <- tibble(wine_type = c("Red Wine","White wine"),
                           rmse = c(fit_ls_rmse_rw,fit_ls_rmse_ww),
                           mae = c(fit_ls_mae_rw,fit_ls_mae_ww),
                           rsquare = c(fit_ls_rsq_rw,fit_ls_rsq_ww))
view(ls_results_df)


rf_results_df <- tibble(wine_type = c("Red Wine","White wine"),
                           rmse = c(fit_rf_rmse_rw,fit_rf_rmse_ww),
                           mae = c(fit_rf_mae_rw,fit_rf_mae_ww),
                           rsquare = c(fit_rf_rsq_rw,fit_rf_rsq_ww))
view(rf_results_df)
```

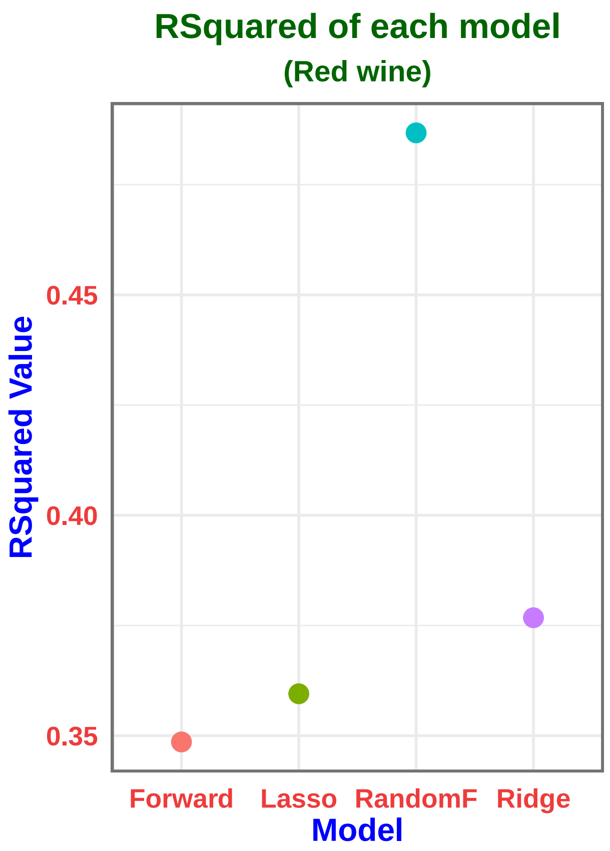
```
rmse_of_plots_rw<-data.frame(rmse_value=c(fit_fs_rmse_rw,fit_rd_rmse_rw,fit_ls_rmse_rw,fit_rf_rmse_rw),m
rmse_of_plots_ww<-data.frame(rmse_value=c(fit_fs_rmse_ww,fit_rd_rmse_ww,fit_ls_rmse_ww,fit_rf_rmse_ww),m


rsq_of_plots_rw<-data.frame(rsq_value=c(fit_fs_rsq_rw,fit_rd_rsq_rw,fit_ls_rsq_rw,fit_rf_rsq_rw),model=c
rsq_of_plots_ww<-data.frame(rsq_value=c(fit_fs_rsq_ww,fit_rd_rsq_ww,fit_ls_rsq_ww,fit_rf_rsq_ww),model=c

mae_of_plots_rw<-data.frame(mae_value=c(fit_fs_mae_rw,fit_rd_mae_rw,fit_ls_mae_rw,fit_rf_mae_rw),model=c
mae_of_plots_ww<-data.frame(mae_value=c(fit_fs_mae_ww,fit_rd_mae_ww,fit_ls_mae_ww,fit_rf_mae_ww),model=c

rmse_plot_rw<-ggplot(data=rmse_of_plots_rw)+
  geom_point(aes(x=model,y=rmse_value,color=model),size=3)+
  theme_minimal()+
  labs(x="Model",
       y="RMSE",
       title="RMSE of each model",
       subtitle="(Red wine)")+
   theme(plot.title = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                plot.subtitle = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                axis.title = element_text(color = "blue",
                                         face = "bold",
                                         size = 12),
                axis.text.x = element_text(hjust = 0.5,
                                         vjust = 0.5,
                                         color = "brown2",
                                         face = "bold",
                                         size = 10),
                axis.text.y = element_text(color = "brown2",
                                         face = "bold",
                                         size = 10),
                panel.border = element_rect(colour = "grey45",
                                         fill=NA, size=1),
        legend.position = "none")

rmse_plot_ww<-ggplot(data=rmse_of_plots_ww)+
  geom_point(aes(x=model,y=rmse_value,color=model),size=3)+
  theme_minimal()+
  labs(x="Model",
       y="RMSE",
       title="RMSE of each model",
       subtitle="(White wine)")+
   theme(plot.title = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                plot.subtitle = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                axis.title = element_text(color = "blue",
                                         face = "bold",
                                         size = 12),
                axis.text.x = element_text(hjust = 0.5,
                                         vjust = 0.5,
                                         color = "brown2",
                                         face = "bold",
                                         size = 10),
                axis.text.y = element_text(color = "brown2",
                                         face = "bold",
                                         size = 10),
                panel.border = element_rect(colour = "grey45",
                                         fill=NA, size=1),
        legend.position = "none")

rsq_plot_rw<-ggplot(data=rsq_of_plots_rw)+
  geom_point(aes(x=model,y=rsq_value,color=model),size=3)+
  theme_minimal()+
  labs(x="Model",
       y="RSquared Value",
       title="RSquared of each model",
       subtitle="(Red wine)")+
   theme(plot.title = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                plot.subtitle = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                axis.title = element_text(color = "blue",
                                         face = "bold",
                                         size = 12),
                axis.text.x = element_text(hjust = 0.5,
                                         vjust = 0.5,
                                         color = "brown2",
                                         face = "bold",
                                         size = 10),
                axis.text.y = element_text(color = "brown2",
                                         face = "bold",
                                         size = 10),
                panel.border = element_rect(colour = "grey45",
                                         fill=NA, size=1),
        legend.position = "none")

rsq_plot_ww<-ggplot(data=rsq_of_plots_ww)+
  geom_point(aes(x=model,y=rsq_value,color=model),size=3)+
  theme_minimal()+
  labs(x="Model",
       y="RSquared Value",
       title="RSquare of each model",
       subtitle="(White wine)")+
   theme(plot.title = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                plot.subtitle = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                axis.title = element_text(color = "blue",
                                         face = "bold",
                                         size = 12),
                axis.text.x = element_text(hjust = 0.5,
                                         vjust = 0.5,
                                         color = "brown2",
                                         face = "bold",
                                         size = 10),
                axis.text.y = element_text(color = "brown2",
                                         face = "bold",
                                         size = 10),
                panel.border = element_rect(colour = "grey45",
                                         fill=NA, size=1),
        legend.position = "none")

mae_plot_rw<-ggplot(data=mae_of_plots_rw)+
  geom_point(aes(x=model,y=mae_value,color=model),size=3)+
  theme_minimal()+
  labs(x="Model",
       y="MAE",
       title="MAE of each model",
       subtitle="(Red wine)")+
   theme(plot.title = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                plot.subtitle = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                axis.title = element_text(color = "blue",
                                         face = "bold",
                                         size = 12),
                axis.text.x = element_text(hjust = 0.5,
                                         vjust = 0.5,
                                         color = "brown2",
                                         face = "bold",
                                         size = 10),
                axis.text.y = element_text(color = "brown2",
                                         face = "bold",
                                         size = 10),
                panel.border = element_rect(colour = "grey45",
                                         fill=NA, size=1),
        legend.position = "none")

mae_plot_ww<-ggplot(data=mae_of_plots_ww)+
  geom_point(aes(x=model,y=mae_value,color=model),size=3)+
  theme_minimal()+
  labs(x="Model",
       y="MAE",
       title="MAE of each model",
       subtitle="(White wine)")+
   theme(plot.title = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                plot.subtitle = element_text(hjust = 0.5,
                                         color = "darkgreen",
                                         face = "bold"),
                axis.title = element_text(color = "blue",
                                         face = "bold",
                                         size = 12),
                axis.text.x = element_text(hjust = 0.5,
                                         vjust = 0.5,
                                         color = "brown2",
                                         face = "bold",
                                         size = 10),
                axis.text.y = element_text(color = "brown2",
                                         face = "bold",
                                         size = 10),
                panel.border = element_rect(colour = "grey45",
                                         fill=NA, size=1),
        legend.position = "none")


grid.arrange(rmse_plot_rw,rmse_plot_ww,nrow = 1)
```

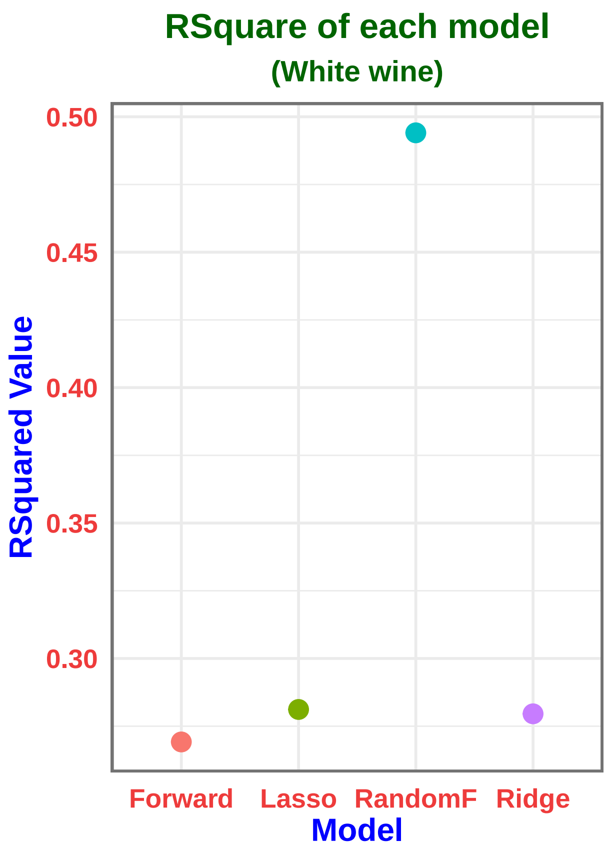


```
grid.arrange(mae_plot_rw,mae_plot_ww,nrow = 1)
```

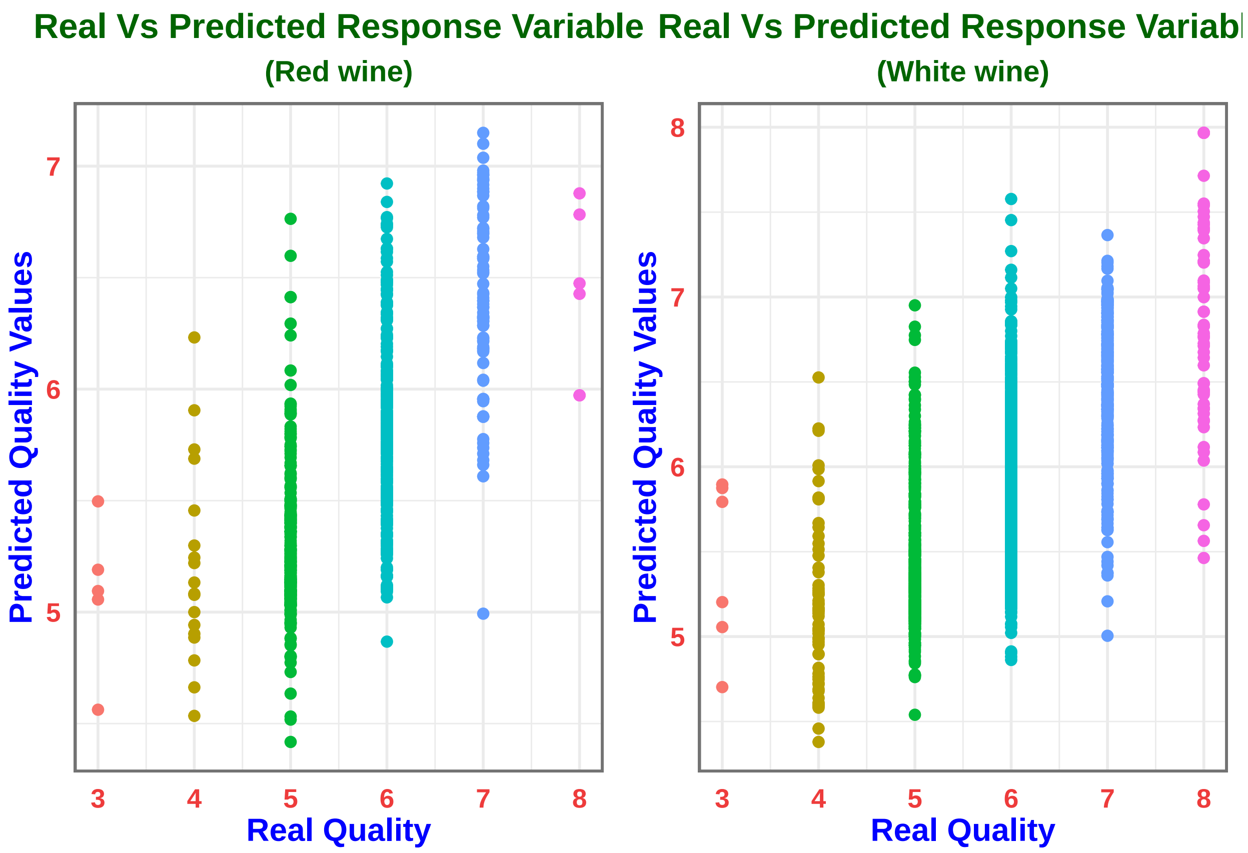


```r
grid.arrange(rsq_plot_rw,rsq_plot_ww,nrow = 1)
```



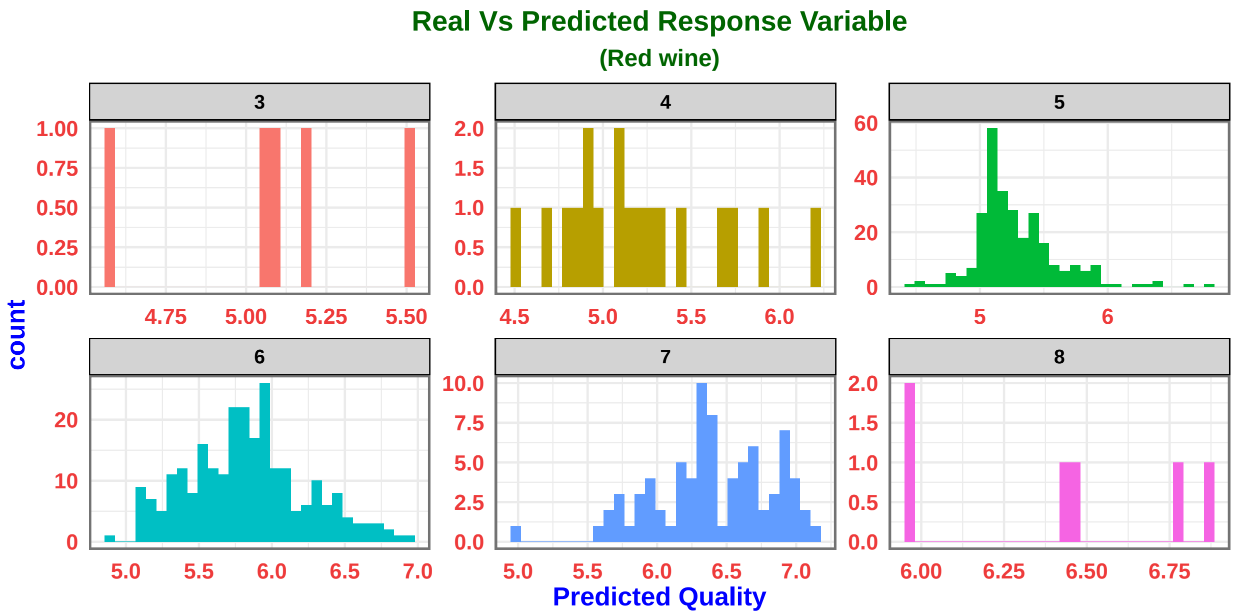


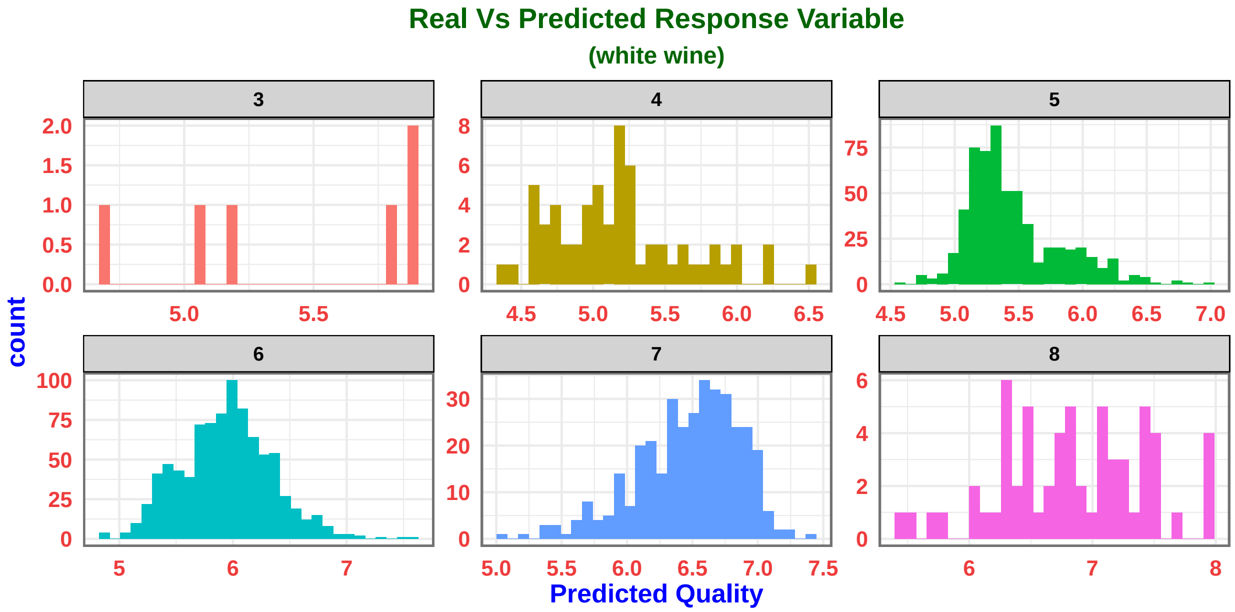
```
red_rf_results<-ggplot(data=rf_rw_results)+
  geom_point(aes(x=real,y=prediction,color=factor(real)))+
  theme_minimal()+
  labs(x="Real Quality",
       y="Predicted Quality Values",
       title="Real Vs Predicted Response Variable",
       subtitle="(Red wine)")+
   theme(plot.title = element_text(hjust = 0.5,
                                          color = "darkgreen",
                                          face = "bold"),
                plot.subtitle = element_text(hjust = 0.5,
                                          color = "darkgreen",
                                          face = "bold"),
                axis.title = element_text(color = "blue",
                                          face = "bold",
                                          size = 12),
                axis.text.x = element_text(hjust = 0.5,
                                          vjust = 0.5,
                                          color = "brown2",
                                          face = "bold",
                                          size = 10),
                axis.text.y = element_text(color = "brown2",
                                          face = "bold",
                                          size = 10),
                panel.border = element_rect(colour = "grey45",
                                          fill=NA, size=1),
         legend.position = "none")


white_rf_results<-ggplot(data=rf_ww_results)+
  geom_point(aes(x=real,y=prediction,color=factor(real)))+
  theme_minimal()+
  labs(x="Real Quality",
       y="Predicted Quality Values",
       title="Real Vs Predicted Response Variable",
       subtitle="(White wine)")+
   theme(plot.title = element_text(hjust = 0.5,
                                          color = "darkgreen",
                                          face = "bold"),
                plot.subtitle = element_text(hjust = 0.5,
                                          color = "darkgreen",
                                          face = "bold"),
                axis.title = element_text(color = "blue",
                                          face = "bold",
                                          size = 12),
                axis.text.x = element_text(hjust = 0.5,
                                          vjust = 0.5,
                                          color = "brown2",
                                          face = "bold",
                                          size = 10),
                axis.text.y = element_text(color = "brown2",
                                          face = "bold",
                                          size = 10),
                panel.border = element_rect(colour = "grey45",
                                          fill=NA, size=1),
         legend.position = "none")

grid.arrange(red_rf_results,white_rf_results,nrow=1)
```



```
red_rf_plot<-ggplot(data=rf_rw_results)+
  geom_histogram(aes(x= prediction,fill=factor(real)))+
  facet_wrap(~ factor(real),  scales="free")+
  theme_minimal()+
  labs(x="Predicted Quality",
       title="Real Vs Predicted Response Variable",
       subtitle="(Red wine)")+
   theme(plot.title = element_text(hjust = 0.5,
                                          color = "darkgreen",
                                          face = "bold"),
         plot.subtitle = element_text(hjust = 0.5,
                                    color = "darkgreen",
                                    face = "bold"),
         axis.title = element_text(color = "blue",
                                    face = "bold",
                                    size = 12),
         axis.text.x = element_text(hjust = 0.5,
                                    vjust = 0.5,
                                    color = "brown2",
                                    face = "bold",
                                    size = 10),
         axis.text.y = element_text(color = "brown2",
                                    face = "bold",
                                    size = 10),
         panel.border = element_rect(colour = "grey45",
                                    fill=NA, size=1),
         strip.background = element_rect(fill = "lightgrey"),
         strip.text = element_text(color = "black", face = "bold"),
         legend.position = "none")

white_rf_plot<-ggplot(data=rf_ww_results)+
  geom_histogram(aes(x= prediction,fill=factor(real)))+
  facet_wrap(~ factor(real),  scales="free")+
  theme_minimal()+
  labs(x="Predicted Quality",
       title="Real Vs Predicted Response Variable",
       subtitle="(white wine)")+
   theme(plot.title = element_text(hjust = 0.5,
                                          color = "darkgreen",
                                          face = "bold"),
          plot.subtitle = element_text(hjust = 0.5,
                                    color = "darkgreen",
                                    face = "bold"),
          axis.title = element_text(color = "blue",
                                    face = "bold",
                                    size = 12),
          axis.text.x = element_text(hjust = 0.5,
                                    vjust = 0.5,
                                    color = "brown2",
                                    face = "bold",
                                    size = 10),
          axis.text.y = element_text(color = "brown2",
                                    face = "bold",
                                    size = 10),
          panel.border = element_rect(colour = "grey45",
                                    fill=NA, size=1),
         strip.background = element_rect(fill = "lightgrey"),
         strip.text = element_text(color = "black", face = "bold"),
         legend.position = "none")

grid.arrange(red_rf_plot, white_rf_plot, nrow = 2)
```

```
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
## ‘stat_bin()‘ using ‘bins = 30‘. Pick better value with ‘binwidth‘.
```


Real Vs Predicted Response Variable
(Red wine)
count
Predicted Quality
3
1.00
0.75
0.50
0.25
0.00
4.75
5.00
5.25
5.50
4
2.0
1.5
1.0
0.5
0.0
4.5
5.0
5.5
6.0
5
60
40
20
0
5
6
6
20
10
0
5.0
5.5
6.0
6.5
7.0
7
10.0
7.5
5.0
2.5
0.0
5.0
5.5
6.0
6.5
7.0
8
2.0
1.5
1.0
0.5
0.0
6.00
6.25
6.50
6.75


Real Vs Predicted Response Variable
(white wine)
count
Predicted Quality
3
2.0
1.5
1.0
0.5
0.0
5.0
5.5
4
8
6
4
2
0
4.5
5.0
5.5
6.0
6.5
5
75
50
25
0
4.5
5.0
5.5
6.0
6.5
7.0
6
100
75
50
25
0
5
6
7
7
30
20
10
0
5.0
5.5
6.0
6.5
7.0
7.5
8
6
4
2
0
6
7
8