Final Project

Matthew Rui and Rohit Suresh

Load Data

```
library(tidyverse)
library(tidymodels)
library(ggplot2)
library(pROC)
library(randomForest)
library(caret)

red <- read.csv("winequality-red.csv", sep=";")
red$color <- "red"
white <- read.csv("winequality-white.csv", sep=";")
white$color <- "white"

df <- rbind(red, white)
df <- na.omit(df)
df$quality <- as.factor(df$quality)
```

Introduction and data

Wine quality evaluations play an important role in numerous parts of the wine supply chain. They are important for quality assurance, market differentiation, and consumer guidance. Generally, a wine quality evaluation analyzes appearance, aroma, flavor, structure, and overall balance to generate a score from 1 to 10. Physicochemical properties of wine samples is readily available to most manufacturers, and quantifying the relationship between physiochemical properties and wine quality evaluations could be of great interest to manufacturers. It may help them with quality control, product development, and cost efficiency. [ADD CITATION]

The data for this work is from the UC Irvine Machine Learning Repository from the paper Cortez et al., 2009. [ADD CITATION] Rows with missing data were removed and quality was treated as a factor.

Datasets for white wine and red wine were provided separately from the source. In this work, we combined the data into one dataset, creating a new variable for the color of wine (color) in the process. [DEFINE VARIABLES]

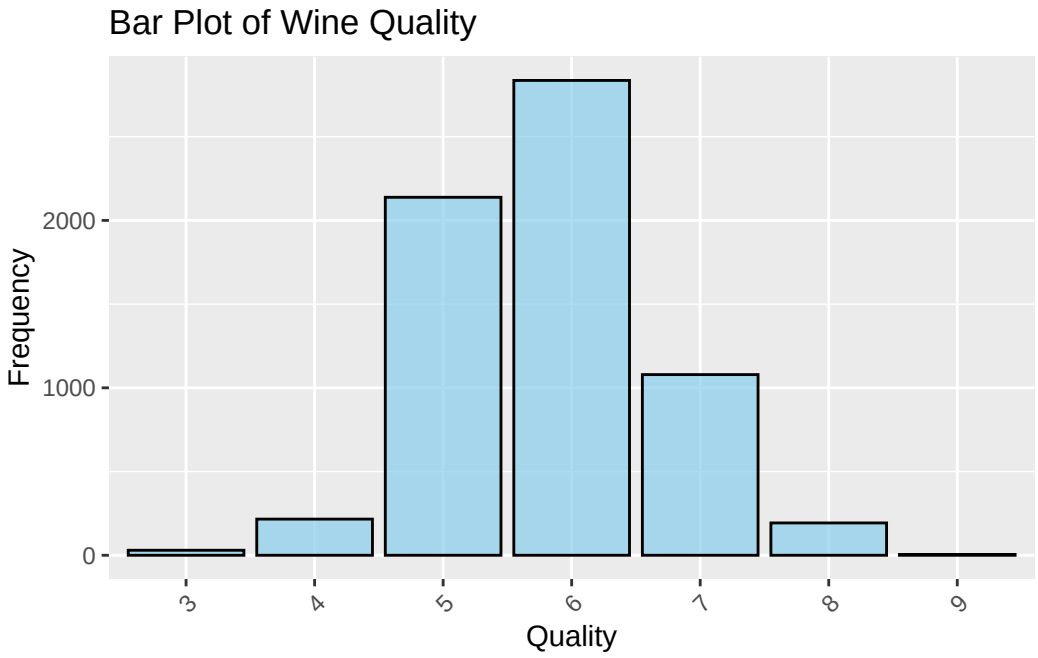
```
summary(df)
```

```
fixed.acidity    volatile.acidity  citric.acid     residual.sugar
Min.   : 3.800   Min.   :0.0800   Min.   :0.0000   Min.   : 0.600
1st Qu.: 6.400   1st Qu.:0.2300   1st Qu.:0.2500   1st Qu.: 1.800
Median : 7.000   Median :0.2900   Median :0.3100   Median : 3.000
Mean   : 7.215   Mean   :0.3397   Mean   :0.3186   Mean   : 5.443
3rd Qu.: 7.700   3rd Qu.:0.4000   3rd Qu.:0.3900   3rd Qu.: 8.100
Max.   :15.900   Max.   :1.5800   Max.   :1.6600   Max.   :65.800

  chlorides       free.sulfur.dioxide total.sulfur.dioxide    density
Min.   :0.00900   Min.   :  1.00      Min.   :  6.0        Min.   :0.9871
1st Qu.:0.03800   1st Qu.: 17.00      1st Qu.: 77.0        1st Qu.:0.9923
Median :0.04700   Median : 29.00      Median :118.0        Median :0.9949
Mean   :0.05603   Mean   : 30.53      Mean   :115.7        Mean   :0.9947
3rd Qu.:0.06500   3rd Qu.: 41.00      3rd Qu.:156.0        3rd Qu.:0.9970
Max.   :0.61100   Max.   :289.00      Max.   :440.0        Max.   :1.0390

      pH          sulphates         alcohol      quality     color
Min.   :2.720   Min.   :0.2200   Min.   : 8.00   3:  30   Length:6497
1st Qu.:3.110   1st Qu.:0.4300   1st Qu.: 9.50   4: 216   Class :character
Median :3.210   Median :0.5100   Median :10.30   5:2138   Mode  :character
Mean   :3.219   Mean   :0.5313   Mean   :10.49   6:2836
3rd Qu.:3.320   3rd Qu.:0.6000   3rd Qu.:11.30   7:1079
Max.   :4.010   Max.   :2.0000   Max.   :14.90   8: 193
                                                 9:   5
```

Red wines, on average, have lower quality scores than white wines on average. While this will result in a more complex model, it will more accurately picture the data in one model, which is why we chose to do this.

```
# Create a bar plot of quality
barplot_quality <- ggplot(df, aes(x = factor(quality))) +
  geom_bar(fill = "skyblue", color = "black", alpha = 0.7) +
  labs(title = "Bar Plot of Wine Quality", x = "Quality", y = "Frequency") +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
```

```r
# Plot the bar plot
print(barplot_quality)
```



To perform logistic regression, a quality threshold for high quality vs low quality was established after analyzing the distribution of quality scores.

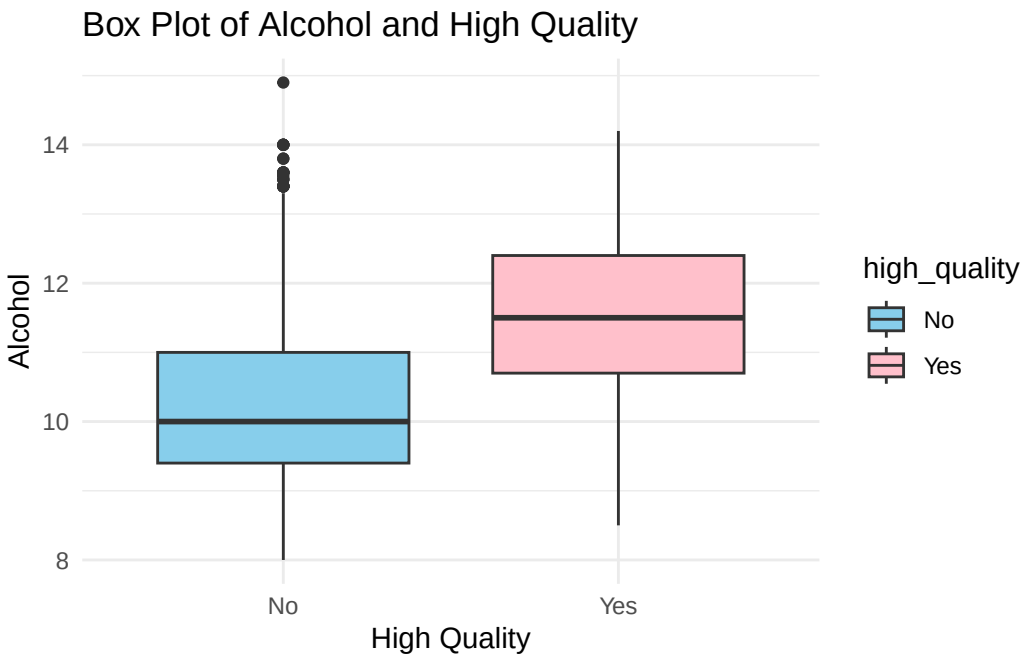
Samples with quality scores of 7 or greater were classified as high quality. [ADD REASONING]

```r
# Continuing, let's try to predict whether a wine is high quality (>= 7).
df$high_quality <- ifelse(df$quality %in% c("7", "8", "9"), 1, 0)

df$high_quality <- factor(df$high_quality, levels = c(0, 1), labels = c("No", "Yes"))
```

Alcohol could be a good predictor of whether a wine is high quality.

```r
# Create a box plot of alcohol and high_quality
boxplot_sulphates <- ggplot(df, aes(x = high_quality, y = alcohol, fill = high_quality)) +
  geom_boxplot() +
  labs(title = "Box Plot of Alcohol and High Quality", x = "High Quality", y = "Alcohol")
  scale_fill_manual(values = c("No" = "skyblue", "Yes" = "pink")) +  # Custom colors for t
  theme_minimal()
```

```r
# Plot the box plot
print(boxplot_sulphates)
```



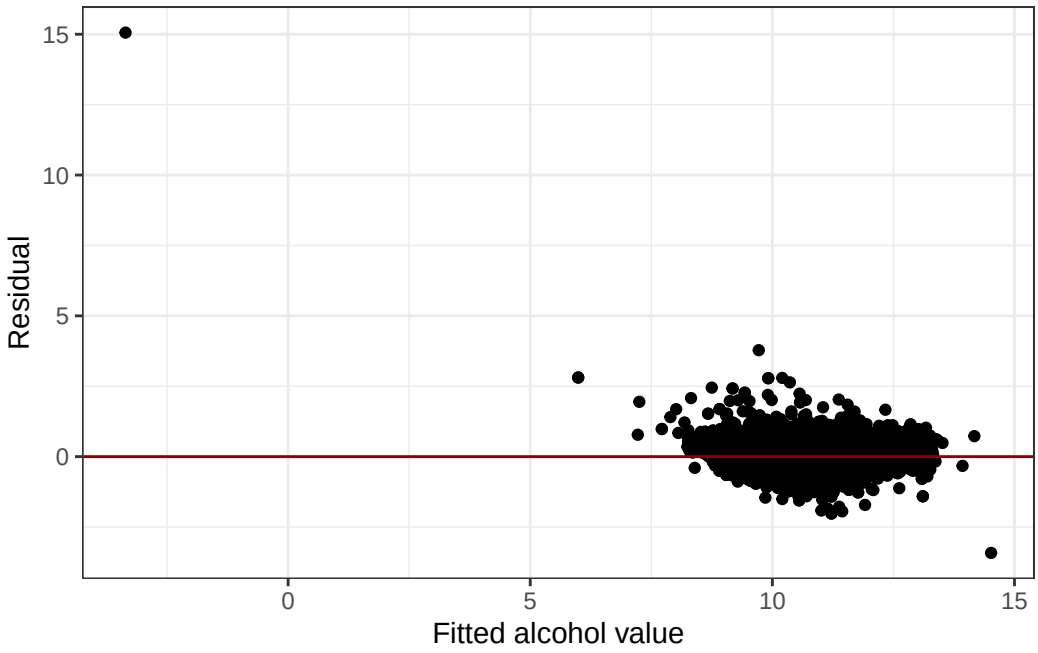
Linear Regression

```r
# Can we predict alcohol levels?

# Simple Linear Regression
simple_linear_regression <- lm(alcohol ~ ., data=df)
summary(simple_linear_regression)
```

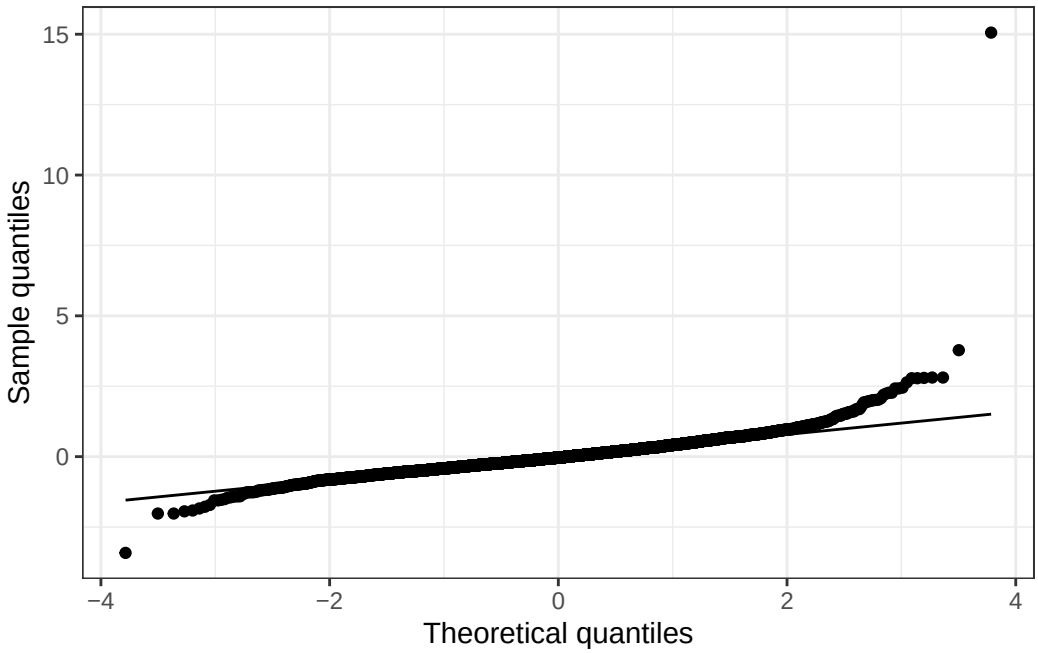
```
Call:
lm(formula = alcohol ~ ., data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-3.4176 -0.2901 -0.0354  0.2539 15.0574

Coefficients: (1 not defined because of singularities)
```

```
                       Estimate Std. Error  t value Pr(>|t|)
(Intercept)           6.471e+02  5.294e+00  122.242  < 2e-16 ***
fixed.acidity         5.176e-01  8.604e-03   60.156  < 2e-16 ***
volatile.acidity      7.930e-01  5.628e-02   14.091  < 2e-16 ***
citric.acid           5.334e-01  5.360e-02    9.951  < 2e-16 ***
residual.sugar        2.273e-01  2.914e-03   78.003  < 2e-16 ***
chlorides            -9.086e-01  2.264e-01   -4.013 6.07e-05 ***
free.sulfur.dioxide  -3.442e-03  5.206e-04   -6.612 4.09e-11 ***
total.sulfur.dioxide  1.253e-05  2.202e-04    0.057  0.95462
density              -6.534e+02  5.429e+00 -120.349  < 2e-16 ***
pH                    2.582e+00  5.266e-02   49.042  < 2e-16 ***
sulphates             9.768e-01  5.063e-02   19.293  < 2e-16 ***
quality4              2.818e-02  9.729e-02    0.290  0.77212
quality5             -2.740e-02  9.192e-02   -0.298  0.76562
quality6              1.478e-01  9.200e-02    1.607  0.10819
quality7              2.308e-01  9.313e-02    2.478  0.01324 *
quality8              2.891e-01  9.862e-02    2.932  0.00338 **
quality9             -2.937e-02  2.405e-01   -0.122  0.90279
colorwhite           -1.160e+00  3.605e-02  -32.180  < 2e-16 ***
high_qualityYes              NA         NA       NA       NA
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4964 on 6479 degrees of freedom
Multiple R-squared:  0.8273,    Adjusted R-squared:  0.8268
F-statistic:  1825 on 17 and 6479 DF,  p-value: < 2.2e-16
```

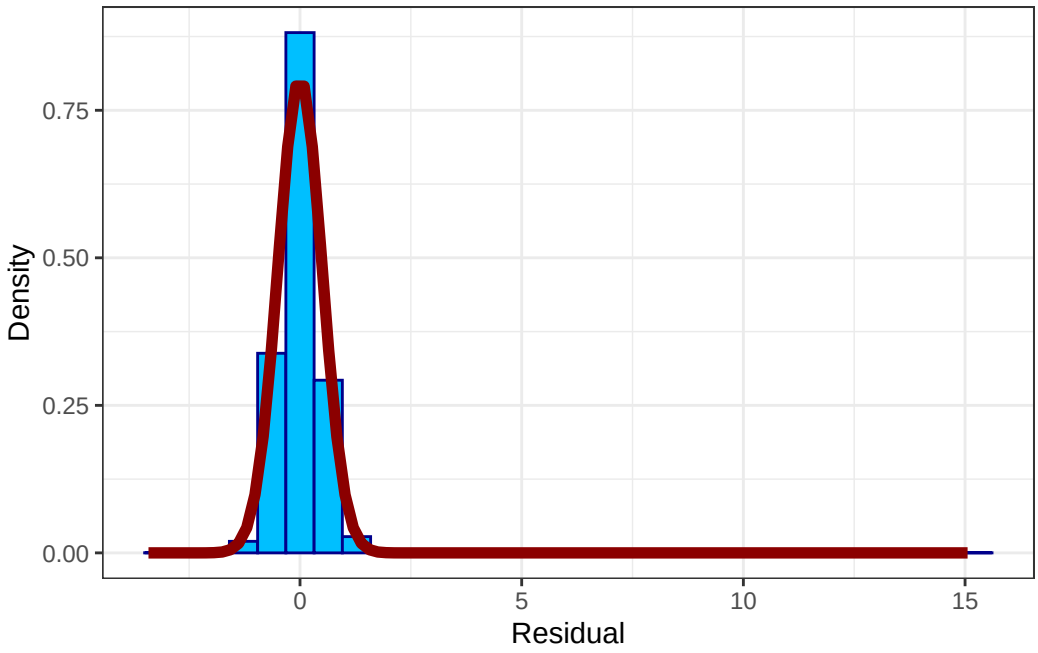
```r
# Residual Plot + Q-Q + Histogram

simple_linear_regression_aug <- augment(simple_linear_regression)
ggplot(simple_linear_regression_aug, aes(x = .fitted, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "darkred") +
  labs(x = "Fitted alcohol value", y = "Residual") +
  theme_bw()
```




```
ggplot(simple_linear_regression_aug, aes(sample = .resid)) +
  stat_qq() +
  stat_qq_line() +
  theme_bw() +
  labs(x = "Theoretical quantiles",
       y = "Sample quantiles")
```




```
ggplot(simple_linear_regression_aug, aes(x = .resid)) +
  geom_histogram(aes(y = ..density..),
                     fill = "deepskyblue", color = "darkblue") +
  stat_function(fun = dnorm,
                args = list(mean = mean(simple_linear_regression_aug$.resid),
                            sd = sd(simple_linear_regression_aug$.resid)),
                color = "darkred", lwd = 2) +
  labs(x = "Residual", y = "Density") +
  theme_bw()
```

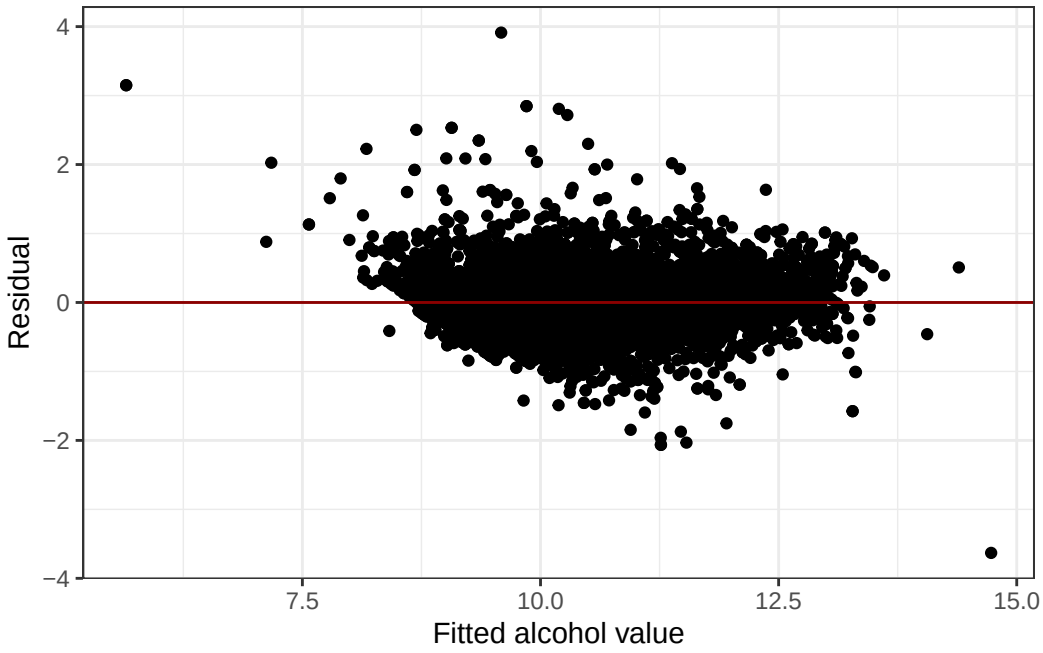



```r
# We have 1 clear outlier - what happens if we remove it?
residuals <- residuals(simple_linear_regression)
outliers <- df[abs(residuals) > 10, ]

df <- df[abs(residuals) <= 10, ]

simple_linear_regression_pruned <- lm(alcohol ~ ., data=df)
summary(simple_linear_regression_pruned)
```

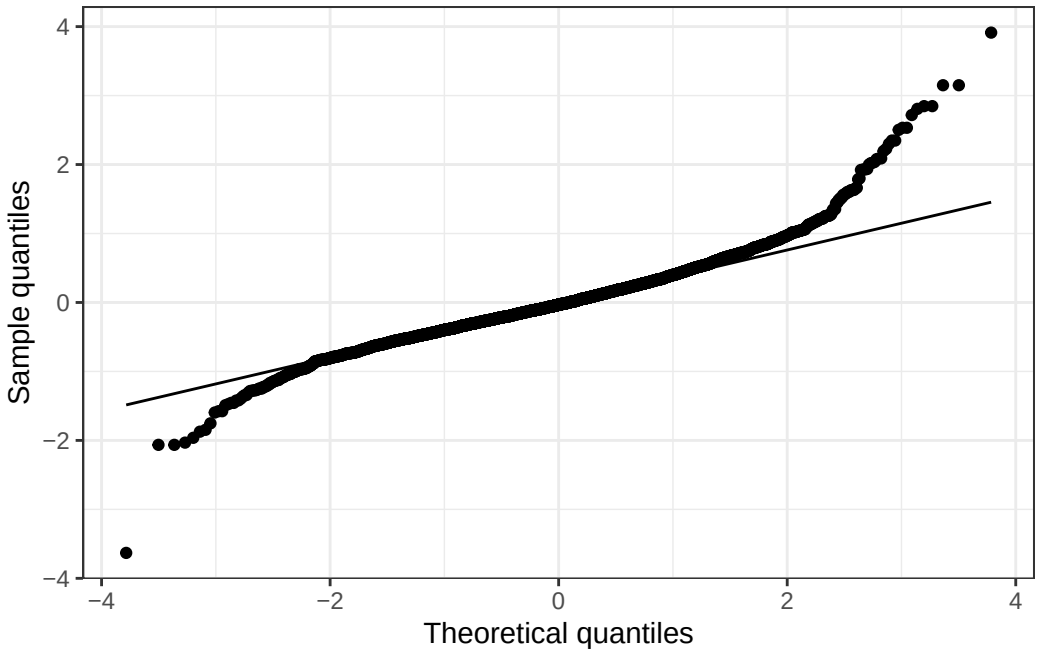
```
Call:
lm(formula = alcohol ~ ., data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-3.6313 -0.2780 -0.0325  0.2459  3.9129

Coefficients: (1 not defined because of singularities)
                      Estimate Std. Error  t value Pr(>|t|)
(Intercept)          6.781e+02  4.954e+00  136.868  < 2e-16 ***
fixed.acidity        5.441e-01  7.954e-03   68.398  < 2e-16 ***
volatile.acidity     6.257e-01  5.202e-02   12.029  < 2e-16 ***
```

```
citric.acid           4.417e-01  4.939e-02    8.943  < 2e-16 ***
residual.sugar        2.325e-01  2.685e-03   86.569  < 2e-16 ***
chlorides            -5.947e-01  2.086e-01   -2.852  0.00436 **
free.sulfur.dioxide  -3.050e-03  4.792e-04   -6.365 2.08e-10 ***
total.sulfur.dioxide  5.920e-04  2.033e-04    2.911  0.00361 **
density              -6.847e+02  5.079e+00 -134.821  < 2e-16 ***
pH                    2.615e+00  4.846e-02   53.952  < 2e-16 ***
sulphates             9.667e-01  4.659e-02   20.748  < 2e-16 ***
quality4              4.600e-02  8.953e-02    0.514  0.60739
quality5             -2.056e-02  8.458e-02   -0.243  0.80797
quality6              1.221e-01  8.466e-02    1.442  0.14942
quality7              1.810e-01  8.571e-02    2.112  0.03474 *
quality8              2.322e-01  9.076e-02    2.559  0.01053 *
quality9             -1.062e-01  2.213e-01   -0.480  0.63117
colorwhite           -1.305e+00  3.344e-02  -39.022  < 2e-16 ***
high_qualityYes              NA         NA       NA       NA
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.4568 on 6478 degrees of freedom
Multiple R-squared:  0.8537,    Adjusted R-squared:  0.8533
F-statistic:  2224 on 17 and 6478 DF,  p-value: < 2.2e-16
```

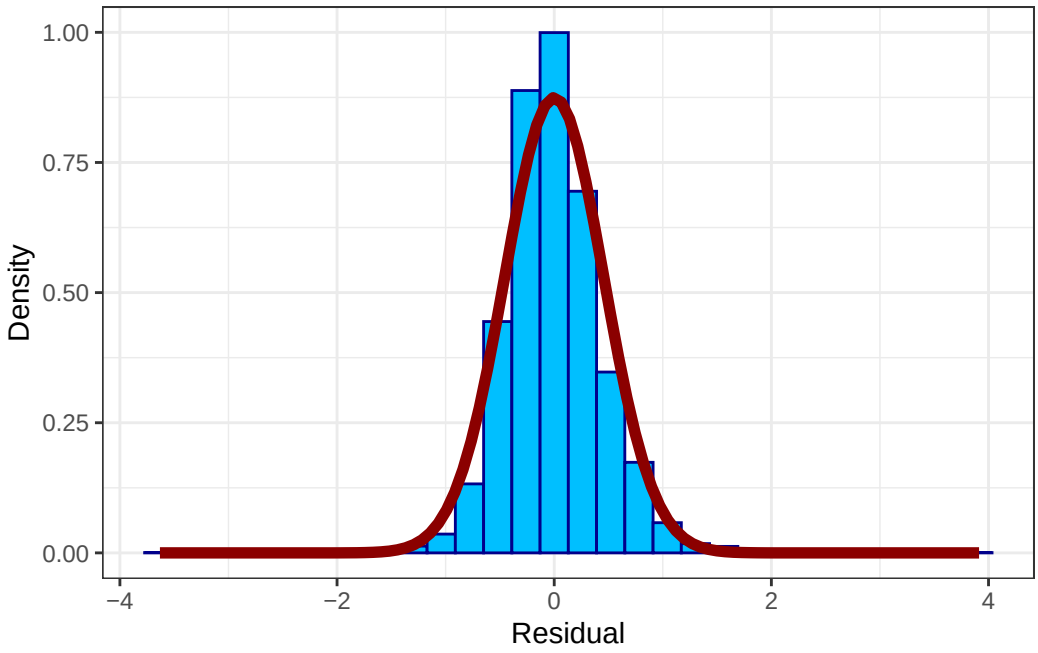
```r
simple_linear_regression_pruned_aug <- augment(simple_linear_regression_pruned)
ggplot(simple_linear_regression_pruned_aug, aes(x = .fitted, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "darkred") +
  labs(x = "Fitted alcohol value", y = "Residual") +
  theme_bw()
```




```
ggplot(simple_linear_regression_pruned_aug, aes(sample = .resid)) +
  stat_qq() +
  stat_qq_line() +
  theme_bw() +
  labs(x = "Theoretical quantiles",
       y = "Sample quantiles")
```




```
ggplot(simple_linear_regression_pruned_aug, aes(x = .resid)) +
  geom_histogram(aes(y = ..density..),
                     fill = "deepskyblue", color = "darkblue") +
  stat_function(fun = dnorm,
                args = list(mean = mean(simple_linear_regression_pruned_aug$.resid),
                            sd = sd(simple_linear_regression_pruned_aug$.resid)),
                color = "darkred", lwd = 2) +
  labs(x = "Residual", y = "Density") +
  theme_bw()
```



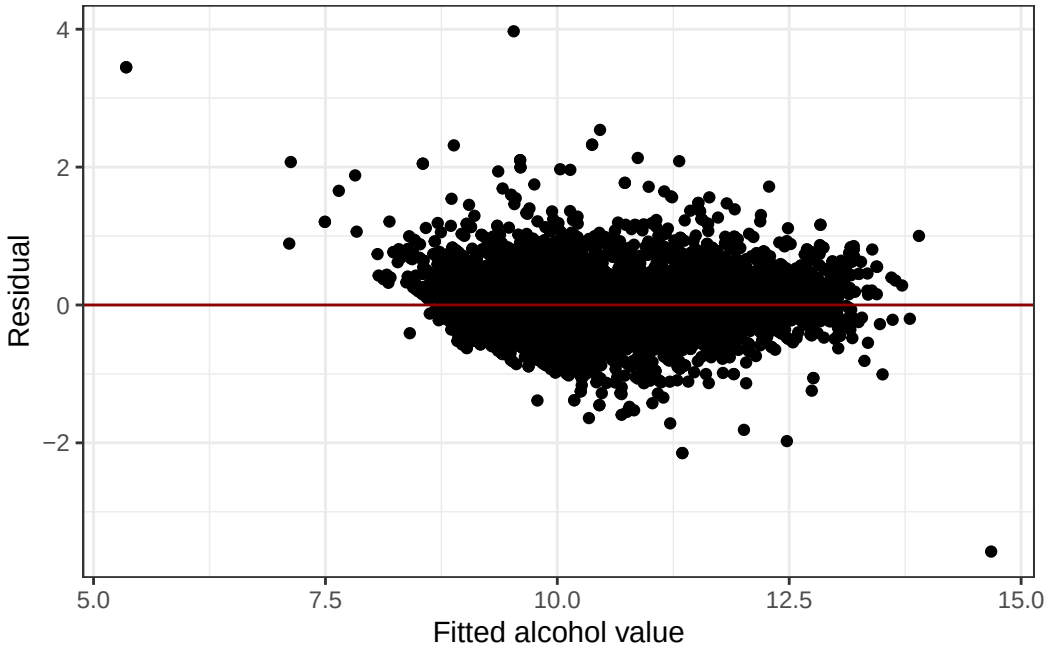

```r
# Ok, now what if we add interaction terms for color of wine?
linear_regression_interaction <- lm(alcohol ~ fixed.acidity + volatile.acidity + citric.ac
summary(linear_regression_interaction)
```

```
Call:
lm(formula = alcohol ~ fixed.acidity + volatile.acidity + citric.acid +
    residual.sugar + chlorides + free.sulfur.dioxide + total.sulfur.dioxide +
    density + pH + sulphates + quality + color + color * fixed.acidity +
    color * volatile.acidity + color * citric.acid + color *
    residual.sugar + color * chlorides + color * free.sulfur.dioxide +
    color * total.sulfur.dioxide + color * density + color *
    pH + color * sulphates + color * quality, data = df)

Residuals:
    Min      1Q  Median      3Q     Max
-3.5769 -0.2653 -0.0368  0.2284  3.9701

Coefficients: (1 not defined because of singularities)
                                Estimate Std. Error t value Pr(>|t|)
(Intercept)                    5.602e+02  9.997e+00  56.038  < 2e-16 ***
fixed.acidity                  4.887e-01  1.510e-02  32.364  < 2e-16 ***
```

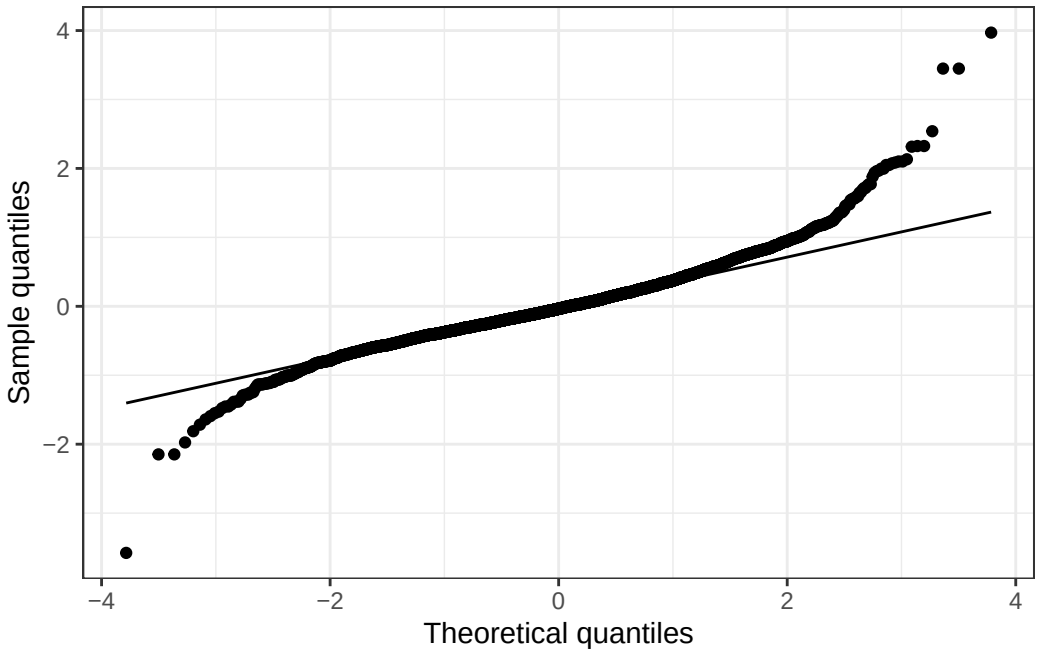
```
volatile.acidity                  5.670e-01  8.366e-02   6.777 1.33e-11 ***
citric.acid                       8.117e-01  9.826e-02   8.261  < 2e-16 ***
residual.sugar                    2.601e-01  8.977e-03  28.968  < 2e-16 ***
chlorides                        -9.280e-01  2.836e-01  -3.272 0.001074 **
free.sulfur.dioxide              -3.126e-03  1.465e-03  -2.133 0.032954 *
total.sulfur.dioxide             -1.205e-03  4.989e-04  -2.416 0.015731 *
density                          -5.697e+02  1.026e+01 -55.520  < 2e-16 ***
pH                                3.587e+00  1.118e-01  32.081  < 2e-16 ***
sulphates                         9.360e-01  7.698e-02  12.159  < 2e-16 ***
quality4                          1.815e-01  1.514e-01   1.199 0.230666
quality5                          2.918e-01  1.416e-01   2.061 0.039335 *
quality6                          5.614e-01  1.422e-01   3.948 7.98e-05 ***
quality7                          7.739e-01  1.463e-01   5.290 1.26e-07 ***
quality8                          1.105e+00  1.768e-01   6.247 4.44e-10 ***
quality9                         -2.331e-01  2.186e-01  -1.066 0.286288
colorwhite                        1.580e+02  1.149e+01  13.747  < 2e-16 ***
fixed.acidity:colorwhite          4.866e-02  1.803e-02   2.698 0.006991 **
volatile.acidity:colorwhite       2.878e-02  1.080e-01   0.267 0.789850
citric.acid:colorwhite           -5.329e-01  1.128e-01  -4.724 2.36e-06 ***
residual.sugar:colorwhite        -1.471e-02  9.451e-03  -1.556 0.119718
chlorides:colorwhite              1.453e+00  4.260e-01   3.412 0.000650 ***
free.sulfur.dioxide:colorwhite   -4.722e-04  1.545e-03  -0.306 0.759894
total.sulfur.dioxide:colorwhite   2.948e-03  5.458e-04   5.402 6.84e-08 ***
density:colorwhite               -1.561e+02  1.180e+01 -13.233  < 2e-16 ***
pH:colorwhite                    -1.179e+00  1.231e-01  -9.573  < 2e-16 ***
sulphates:colorwhite              2.827e-02  9.576e-02   0.295 0.767827
quality4:colorwhite              -1.659e-01  1.838e-01  -0.903 0.366713
quality5:colorwhite              -3.958e-01  1.726e-01  -2.293 0.021858 *
quality6:colorwhite              -5.873e-01  1.731e-01  -3.393 0.000695 ***
quality7:colorwhite              -7.826e-01  1.769e-01  -4.423 9.89e-06 ***
quality8:colorwhite              -1.076e+00  2.050e-01  -5.249 1.57e-07 ***
quality9:colorwhite                      NA         NA      NA       NA
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 0.436 on 6463 degrees of freedom
Multiple R-squared:  0.867, Adjusted R-squared:  0.8664
F-statistic:  1317 on 32 and 6463 DF,  p-value: < 2.2e-16
```

```
# Note that p > n for quality = 9 and color = white, hence we're unable to obtain estimate
```

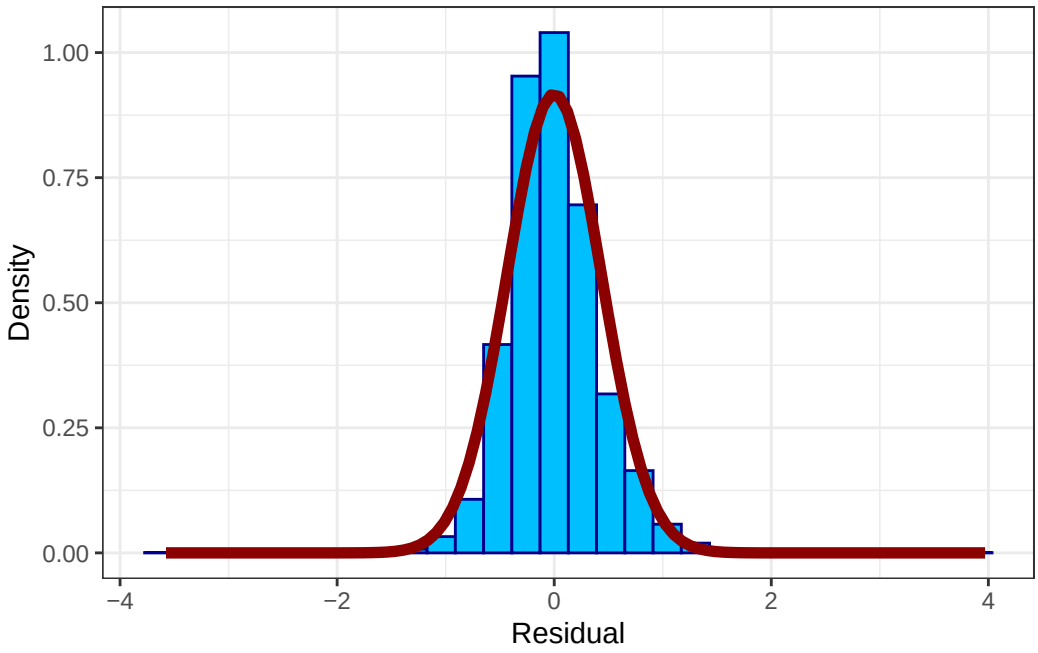
```r
# Residual Plot + Q-Q + Histogram Again
linear_regression_interaction_aug <- augment(linear_regression_interaction)
ggplot(linear_regression_interaction_aug, aes(x = .fitted, y = .resid)) +
  geom_point() +
  geom_hline(yintercept = 0, color = "darkred") +
  labs(x = "Fitted alcohol value", y = "Residual") +
  theme_bw()
```




```r
ggplot(linear_regression_interaction_aug, aes(sample = .resid)) +
  stat_qq() +
  stat_qq_line() +
  theme_bw() +
  labs(x = "Theoretical quantiles",
       y = "Sample quantiles")
```




```
ggplot(linear_regression_interaction_aug, aes(x = .resid)) +
  geom_histogram(aes(y = ..density..),
                     fill = "deepskyblue", color = "darkblue") +
  stat_function(fun = dnorm,
                args = list(mean = mean(linear_regression_interaction_aug$.resid),
                            sd = sd(linear_regression_interaction_aug$.resid)),
                color = "darkred", lwd = 2) +
  labs(x = "Residual", y = "Density") +
  theme_bw()
```



Classification w/ Logistic

```
# Continuing, let's try to predict whether a wine is high quality (>= 7, arbitrary cutoff)
df$high_quality <- ifelse(df$quality %in% c("7", "8", "9"), 1, 0)

logistic_model <- glm(high_quality ~ fixed.acidity + volatile.acidity + citric.acid + resi
summary(logistic_model)
```

```
Call:
glm(formula = high_quality ~ fixed.acidity + volatile.acidity +
    citric.acid + residual.sugar + chlorides + free.sulfur.dioxide +
    total.sulfur.dioxide + density + pH + sulphates + alcohol +
    color, family = "binomial", data = df)

Coefficients:
                       Estimate Std. Error z value Pr(>|z|)
(Intercept)           4.029e+02  6.585e+01   6.118 9.50e-10 ***
fixed.acidity         4.940e-01  6.711e-02   7.361 1.83e-13 ***
volatile.acidity     -3.663e+00  3.882e-01  -9.437  < 2e-16 ***
citric.acid          -2.488e-01  3.458e-01  -0.719 0.471841
```

```
residual.sugar        2.198e-01  2.627e-02   8.367  < 2e-16 ***
chlorides            -7.612e+00  2.498e+00  -3.048 0.002306 **
free.sulfur.dioxide   1.080e-02  2.953e-03   3.657 0.000255 ***
total.sulfur.dioxide -3.705e-03  1.335e-03  -2.776 0.005509 **
density              -4.238e+02  6.674e+01  -6.350 2.15e-10 ***
pH                    2.596e+00  3.614e-01   7.184 6.76e-13 ***
sulphates             2.459e+00  2.854e-01   8.615  < 2e-16 ***
alcohol               4.541e-01  8.084e-02   5.618 1.94e-08 ***
colorwhite           -7.747e-01  2.446e-01  -3.167 0.001540 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6439.2  on 6495  degrees of freedom
Residual deviance: 5078.8  on 6483  degrees of freedom
AIC: 5104.8

Number of Fisher Scoring iterations: 6
```

```r
# Honestly pretty good, let's now try again with interaction terms

logistic_model_interaction <- glm(high_quality ~ fixed.acidity + volatile.acidity + citric
summary(logistic_model_interaction)
```

```
Call:
glm(formula = high_quality ~ fixed.acidity + volatile.acidity +
    citric.acid + residual.sugar + chlorides + free.sulfur.dioxide +
    total.sulfur.dioxide + density + pH + sulphates + alcohol +
    color + color * fixed.acidity + color * volatile.acidity +
    color * citric.acid + color * residual.sugar + color * chlorides +
    color * free.sulfur.dioxide + color * total.sulfur.dioxide +
    color * density + color * pH + color * sulphates + color *
    alcohol, family = "binomial", data = df)

Coefficients:
                                Estimate Std. Error z value Pr(>|z|)
(Intercept)                    2.428e+02  1.081e+02   2.247 0.024660 *
fixed.acidity                  2.750e-01  1.253e-01   2.195 0.028183 *
volatile.acidity              -2.581e+00  7.843e-01  -3.291 0.000999 ***
```

```
citric.acid                          5.678e-01  8.385e-01   0.677 0.498313
residual.sugar                       2.395e-01  7.373e-02   3.248 0.001163 **
chlorides                           -8.816e+00  3.365e+00  -2.620 0.008788 **
free.sulfur.dioxide                  1.082e-02  1.223e-02   0.884 0.376469
total.sulfur.dioxide                -1.653e-02  4.894e-03  -3.378 0.000731 ***
density                             -2.578e+02  1.104e+02  -2.335 0.019536 *
pH                                   2.242e-01  9.984e-01   0.225 0.822327
sulphates                            3.750e+00  5.416e-01   6.924 4.39e-12 ***
alcohol                              7.533e-01  1.316e-01   5.724 1.04e-08 ***
colorwhite                           3.934e+02  1.433e+02   2.745 0.006045 **
fixed.acidity:colorwhite             2.772e-01  1.546e-01   1.793 0.072955 .
volatile.acidity:colorwhite         -1.204e+00  9.240e-01  -1.303 0.192587
citric.acid:colorwhite              -1.306e+00  9.295e-01  -1.405 0.160119
residual.sugar:colorwhite            5.573e-02  8.189e-02   0.681 0.496172
chlorides:colorwhite                -3.824e+00  5.088e+00  -0.752 0.452351
free.sulfur.dioxide:colorwhite      -2.176e-03  1.263e-02  -0.172 0.863207
total.sulfur.dioxide:colorwhite      1.626e-02  5.120e-03   3.176 0.001494 **
density:colorwhite                  -4.013e+02  1.459e+02  -2.750 0.005955 **
pH:colorwhite                        3.119e+00  1.086e+00   2.872 0.004074 **
sulphates:colorwhite                -1.582e+00  6.435e-01  -2.459 0.013944 *
alcohol:colorwhite                  -6.110e-01  1.740e-01  -3.511 0.000447 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6439.2  on 6495  degrees of freedom
Residual deviance: 5014.1  on 6472  degrees of freedom
AIC: 5062.1

Number of Fisher Scoring iterations: 6
```

```r
# Surprisingly, some of the predictor which were significant for the model w/o interaction

prob_logistic <- predict(logistic_model, type = "response")
roc_logistic <- roc(df$high_quality, prob_logistic)
optimal_logistic <- coords(roc_logistic, "best", ret = "threshold")[[1]]
print(paste("Optimal threshold:", optimal_logistic))
```

```
[1] "Optimal threshold: 0.186570303907223"
```

```r
optimal_threshold <- 0.5
df$predict_logistic <- ifelse(prob_logistic >= optimal_threshold, 'High Quality', 'Low Qua
table(df$predict_logistic, df$high_quality)
```

```
                0    1
High Quality  246  342
Low Quality  4973  935
```

```r
prob_logistic_interaction <- predict(logistic_model_interaction, type = "response")
roc_logistic_interaction <- roc(df$high_quality, prob_logistic_interaction)
optimal_logistic_interaction <- coords(roc_logistic_interaction, "best", ret = "threshold"
print(paste("Optimal threshold:", optimal_logistic_interaction))
```

```
[1] "Optimal threshold: 0.234879888142119"
```

```r
df$predict_logistic_interaction <- ifelse(prob_logistic_interaction >= optimal_threshold,
table(df$predict_logistic_interaction, df$high_quality)
```

```
                0    1
High Quality  248  372
Low Quality  4971  905
```

Logistic Model: - Sensitivity: 1183/(1183+94) = 0.9263899765 - Specificity: 2050/(2050+3169) = 0.3927955547 - Positive Predictive Value: 1183/(1183+3169) = 0.27182904411 - Negative Predictive Value: 2050/(2050+94) = 0.95615671641

Logistic Interaction Model: - Sensitivity: 1203/(1203+74) = 0.94205168363 - Specificity: 2091/(2091+3128) = 0.40065146579 - Positive Predictive Value: 1203/(1203+3128) = 0.27776495035 - Negative Predictive Value: 2091/(2091+74) = 0.96581986143

The interaction model is better in every metric!

Random Forest

```
include_cols <- c('fixed.acidity' , 'volatile.acidity' , 'citric.acid' , 'residual.sugar'
X <- df[, (names(df) %in% include_cols)]
y <- df$high_quality
k <- 10
ctrl <- trainControl(method = "cv", number = k, verboseIter = TRUE)
#rf_model <- train(x = X, y = y, method = "rf", trControl = ctrl)

#print(rf_model)
#df$rf_predictions <- predict(rf_model, X)
#df$rf_classification <- ifelse(df$rf_predictions > 0.5, "High Quality", "Low Quality")

#table(df$rf_classification, df$high_quality)
```

Random Forest Model: - Sensitivity: 1262/(1262+15) = 0.98825371965 - Specificity: 5219/(5219+0) = 1 - Positive Predictive Value: 1262/(1262+0) = 1 - Negative Predictive Value: 5219/(5219+15) = 0.99713412304