Project

R Markdown

This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

When you click the **Knit** button a document will be generated that includes both content as well as the output of any embedded R code chunks within the document. You can embed an R code chunk like this:

Project

```r
library(plyr)
library(randomForest)
```

```
## randomForest 4.6-12
```

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

```r
library(tidyverse)
```

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

```
## v ggplot2 2.2.1     v purrr   0.2.4
## v tibble  1.4.1     v dplyr   0.7.4
## v tidyr   0.7.2     v stringr 1.2.0
## v readr   1.1.1     v forcats 0.2.0
```

```
## -- Conflicts ----------------------------------------------------------------------- tidyverse_confli
## x dplyr::arrange()   masks plyr::arrange()
## x dplyr::combine()   masks randomForest::combine()
## x purrr::compact()   masks plyr::compact()
## x dplyr::count()     masks plyr::count()
## x dplyr::failwith()  masks plyr::failwith()
## x dplyr::filter()    masks stats::filter()
## x dplyr::id()        masks plyr::id()
## x dplyr::lag()       masks stats::lag()
## x ggplot2::margin()  masks randomForest::margin()
## x dplyr::mutate()    masks plyr::mutate()
## x dplyr::rename()    masks plyr::rename()
## x dplyr::summarise() masks plyr::summarise()
## x dplyr::summarize() masks plyr::summarize()
```

```r
library(knitr)
library(gridExtra)
```

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

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

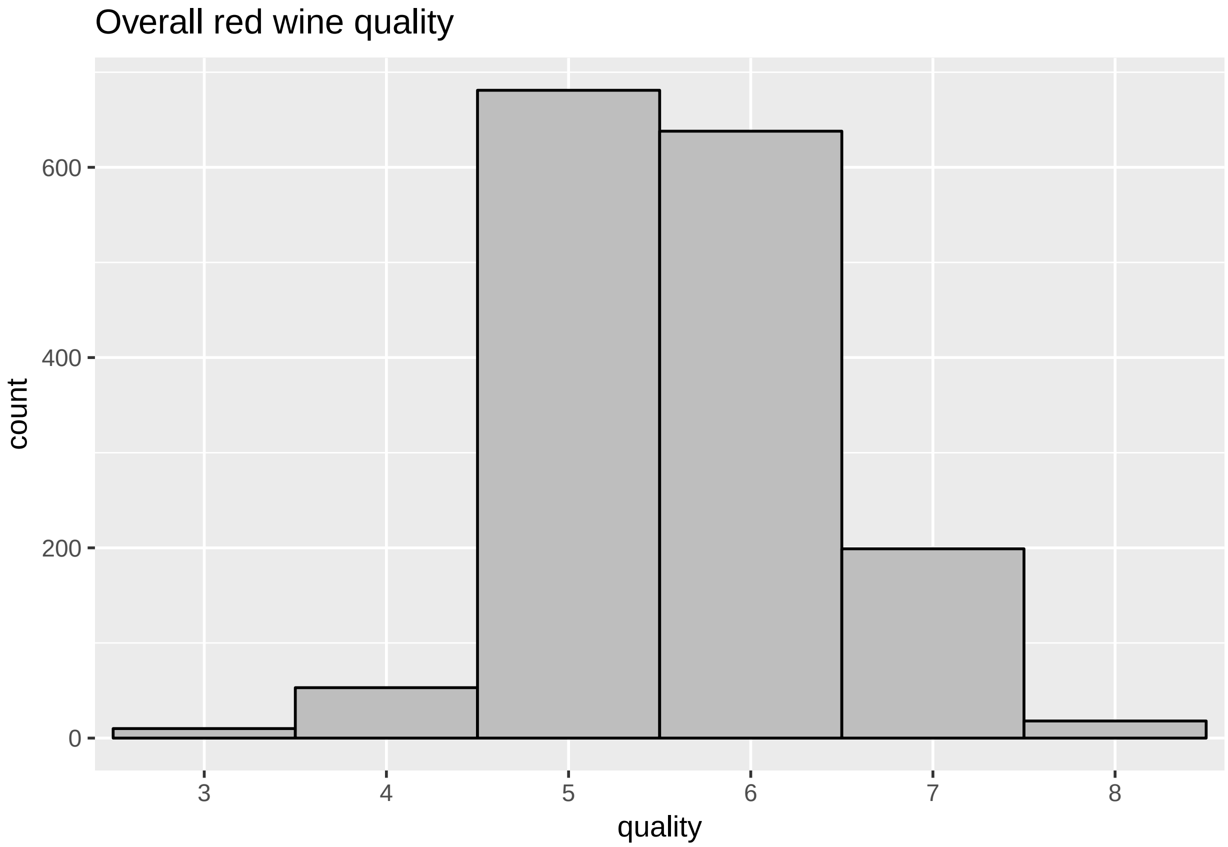
```
## The following object is masked from 'package:randomForest':
##
```

```
##     combine
```

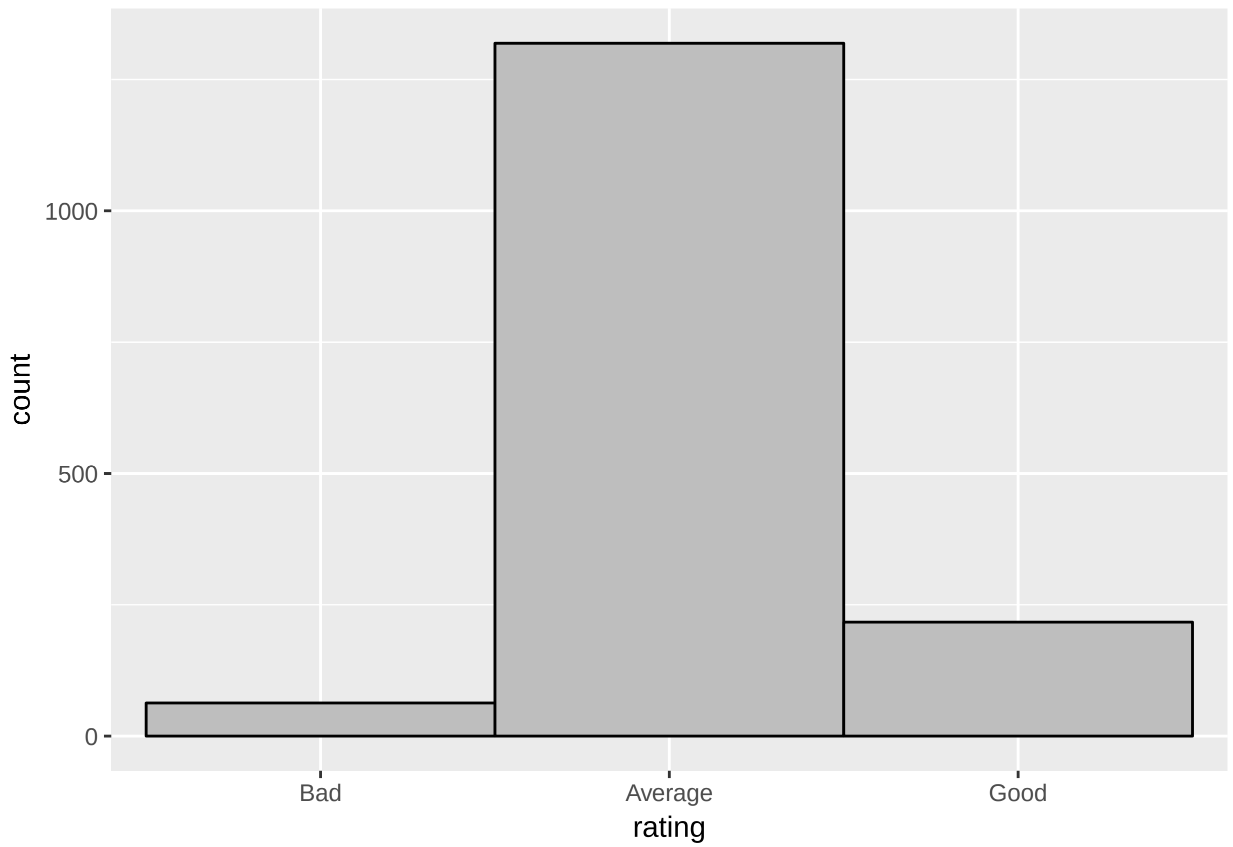
```r
library(rpart)
library(rpart.plot)
library(caret)
```

```
## Loading required package: lattice

##
## Attaching package: 'caret'

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

```r
dat <- read.csv("wineQualityReds.csv")
dat.rdforest <- dat # For random forest without new column rating

#Find correlation

#Creating 'Rating'
dat$quality <- factor(dat$quality, ordered = T)
dat$rating <- ifelse(dat$quality < 5, 'Bad', ifelse(
  dat$quality < 7, 'Average', 'Good'))
dat$rating <- ordered(dat$rating,
                      levels = c('Bad', 'Average', 'Good'))

#Plot Graph
ggplot(data = dat, aes(x = quality)) +geom_bar(width = 1, color = 'black',fill = I('gray'))+ggtitle("Ove
```



```
ggplot(data = dat, aes(x = rating)) +geom_bar(width = 1, color = 'black',fill = I('gray'))
```



```
cat("From graph we can see that there are a lot of wines with a quality of 5 and 6 as compared to the o
```

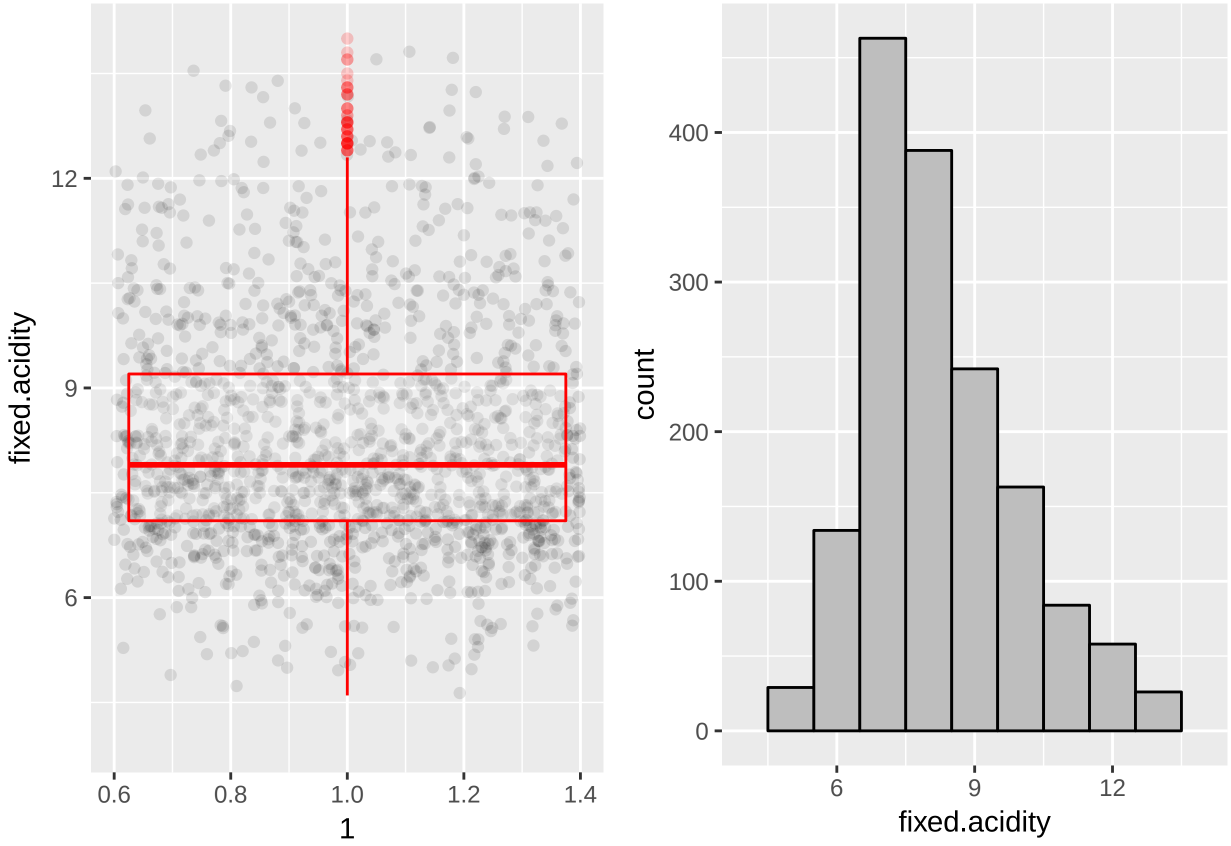
```
## From graph we can see that there are a lot of wines with a quality of 5 and 6 as compared to the othe
```

```
#Fixed Acidity
grid.arrange(ggplot(dat, aes( x = 1, y = fixed.acidity ) ) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ) +
               scale_y_continuous(lim = c(4,14)),
ggplot(data = dat, aes(x = fixed.acidity)) +
  geom_histogram(binwidth = 1, color = 'black',fill = I('gray')) +
  scale_x_continuous(lim = c(4,14)),ncol = 2)
```

```
## Warning: Removed 8 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 9 rows containing missing values (geom_point).
```

```
## Warning: Removed 8 rows containing non-finite values (stat_bin).
```



```
cat("From graph it look skew to the left and has mean around 8.")
```

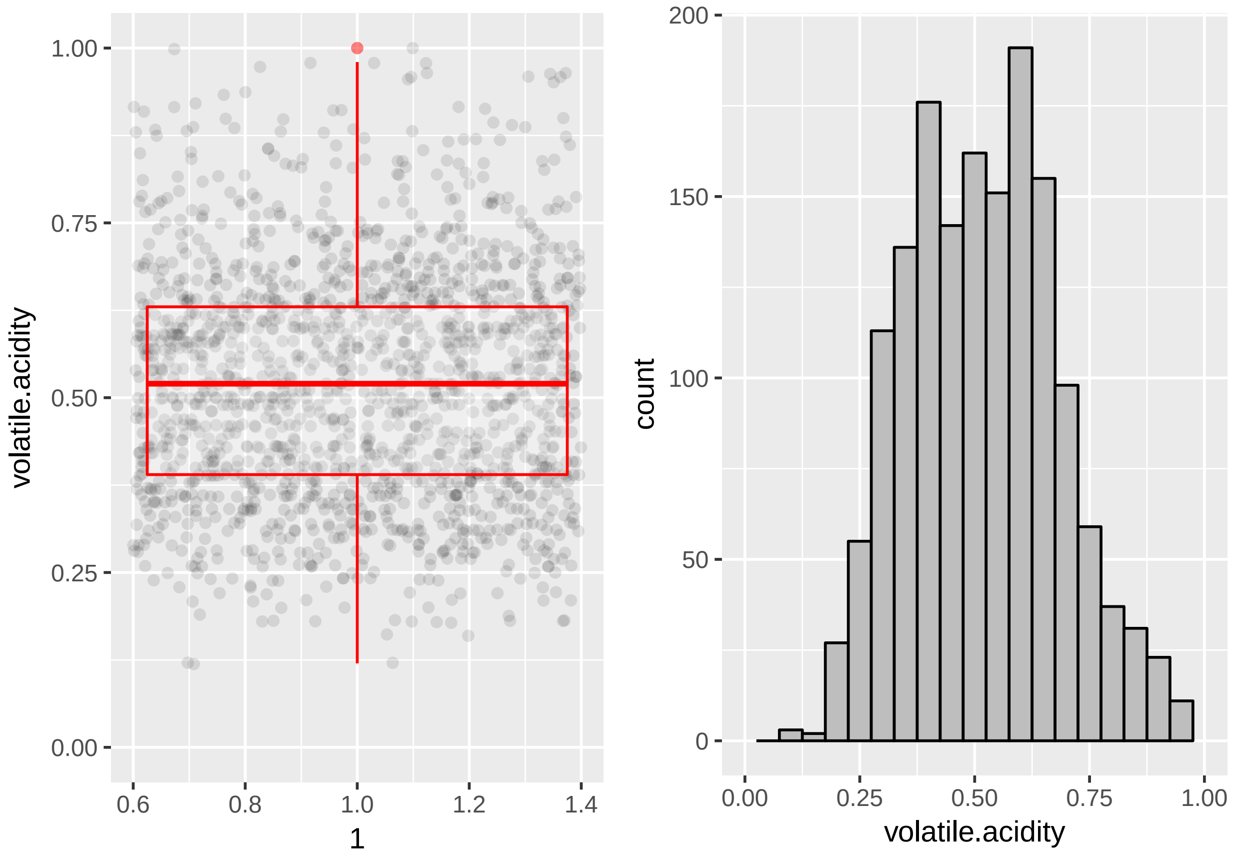
```
## From graph it look skew to the left and has mean around 8.
```

```
#Volatile Acidity
grid.arrange(ggplot(dat, aes( x = 1, y = volatile.acidity ) ) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ) +
               scale_y_continuous(lim = c(0,1)),
ggplot(data = dat, aes(x = volatile.acidity)) +
  geom_histogram(binwidth = 0.05, color = 'black',fill = I('gray')) +
  scale_x_continuous(lim = c(0,1)), ncol = 2)
```

```
## Warning: Removed 21 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 22 rows containing missing values (geom_point).
```

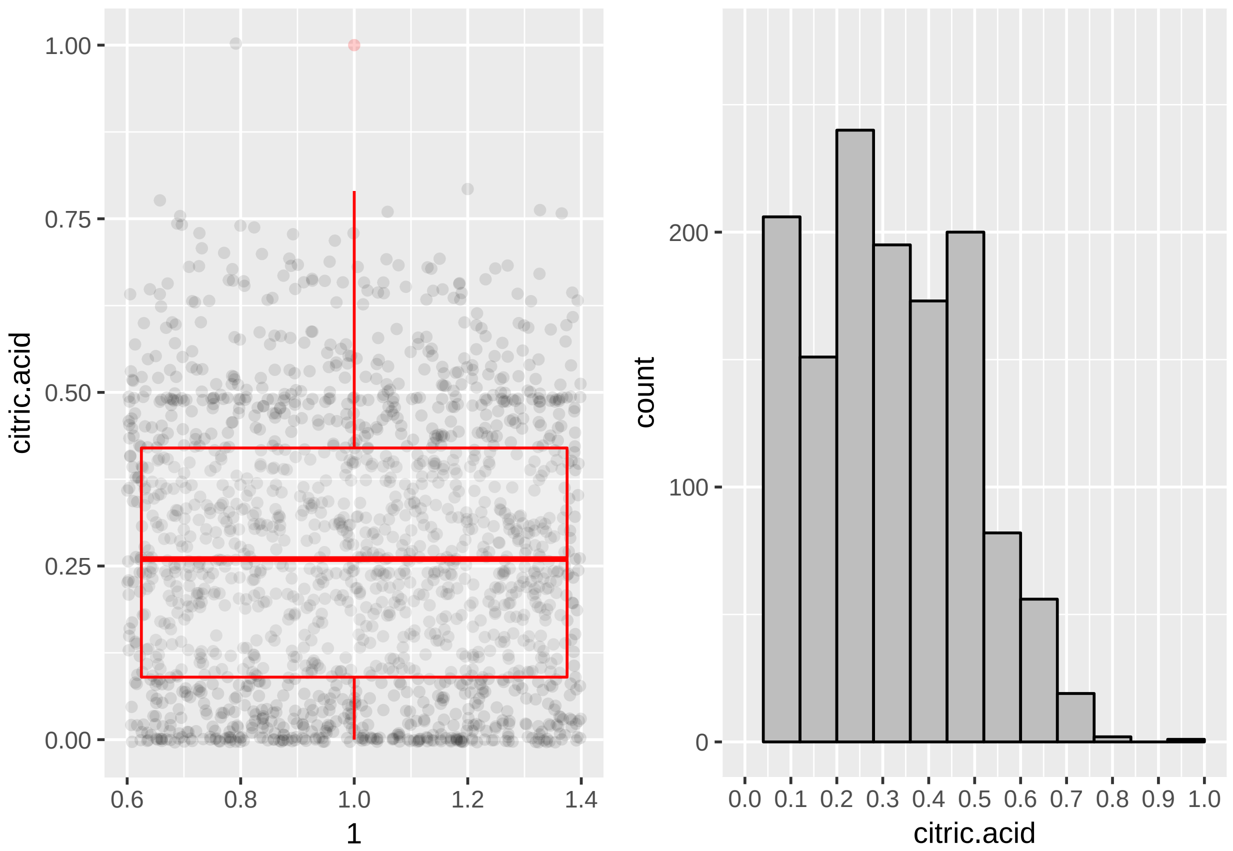
```
## Warning: Removed 21 rows containing non-finite values (stat_bin).
```



```r
cat("From graph has combine model and has peak around 0.6")
```

```
## From graph has combine model and has peak around 0.6
```

```r
#Citric Acid
grid.arrange(ggplot(dat, aes( x = 1, y = citric.acid )) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ),
ggplot(data = dat, aes(x = citric.acid)) +
  geom_histogram(binwidth = 0.08, color = 'black',fill = I('gray')) +
  scale_x_continuous(breaks = seq(0,1,0.1), lim = c(0,1)), ncol = 2)
```

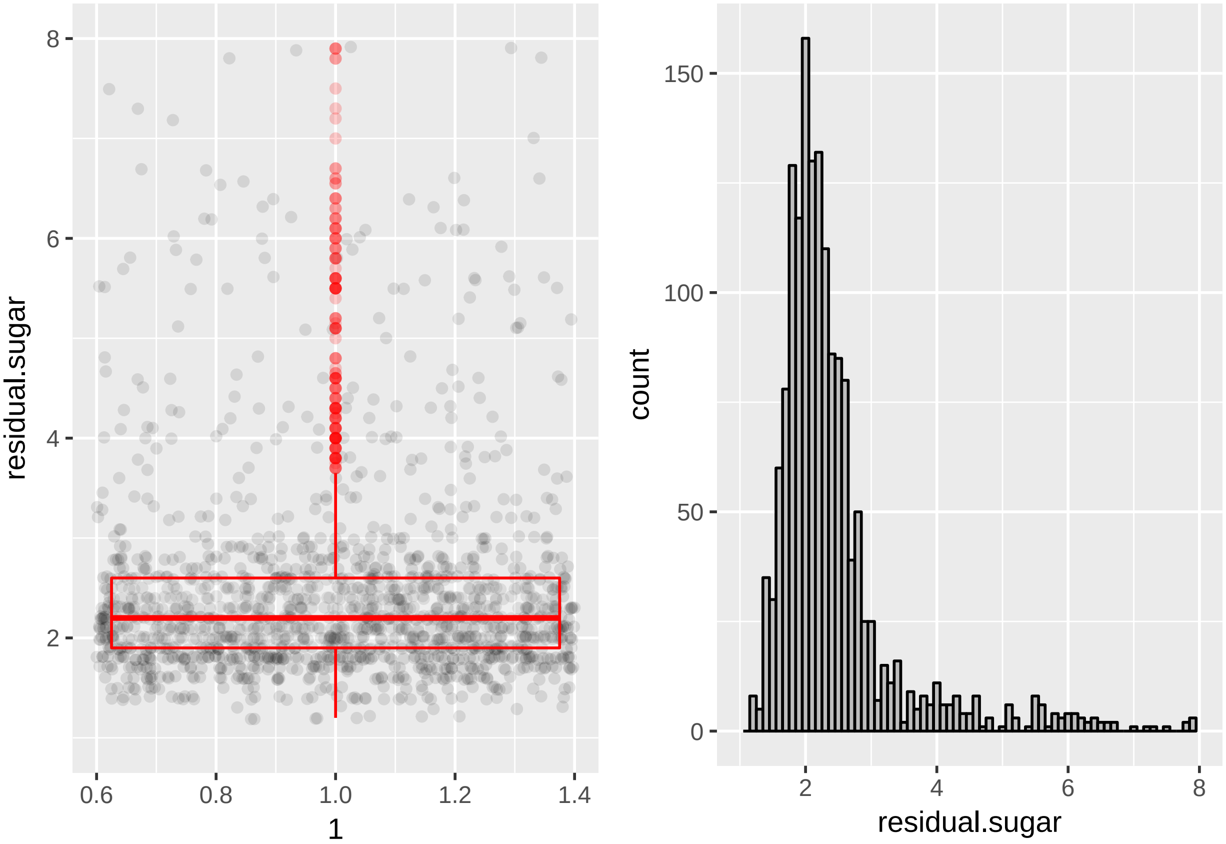


```
cat("From graph look similar rectangle on the left side.")
```

```
## From graph look similar rectangle on the left side.
```

```
#Residual Sugar
grid.arrange(ggplot(dat, aes( x = 1, y = residual.sugar )) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ) +
               scale_y_continuous(lim = c(1,8)),
ggplot(data = dat, aes(x = residual.sugar)) +
  geom_histogram(binwidth = 0.1, color = 'black',fill = I('gray')) +
  scale_x_continuous(lim = c(1,8)), ncol = 2)
```

```
## Warning: Removed 23 rows containing non-finite values (stat_boxplot).

## Warning: Removed 23 rows containing missing values (geom_point).

## Warning: Removed 23 rows containing non-finite values (stat_bin).
```

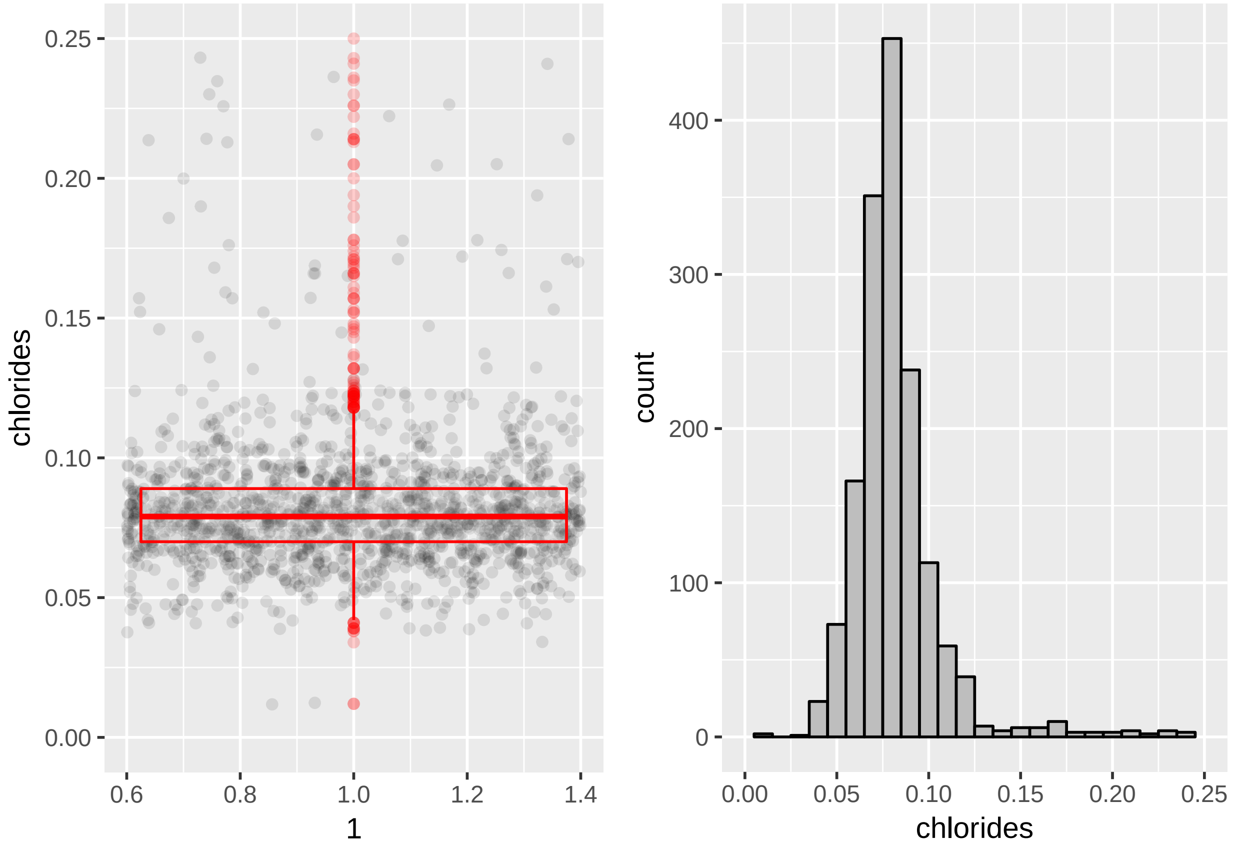


```
cat("The distribution of sugar has skew on the left")
```

```
## The distribution of sugar has skew on the left
```

```
#Chlorides
grid.arrange(ggplot(dat, aes( x = 1, y = chlorides )) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ) +
               scale_y_continuous(lim = c(0,0.25)),
ggplot(data = dat, aes(x = chlorides)) +
  geom_histogram(binwidth = 0.01, color = 'black',fill = I('gray')) +
  scale_x_continuous(lim = c(0,0.25)), ncol = 2)
```

```
## Warning: Removed 25 rows containing non-finite values (stat_boxplot).

## Warning: Removed 26 rows containing missing values (geom_point).

## Warning: Removed 25 rows containing non-finite values (stat_bin).
```

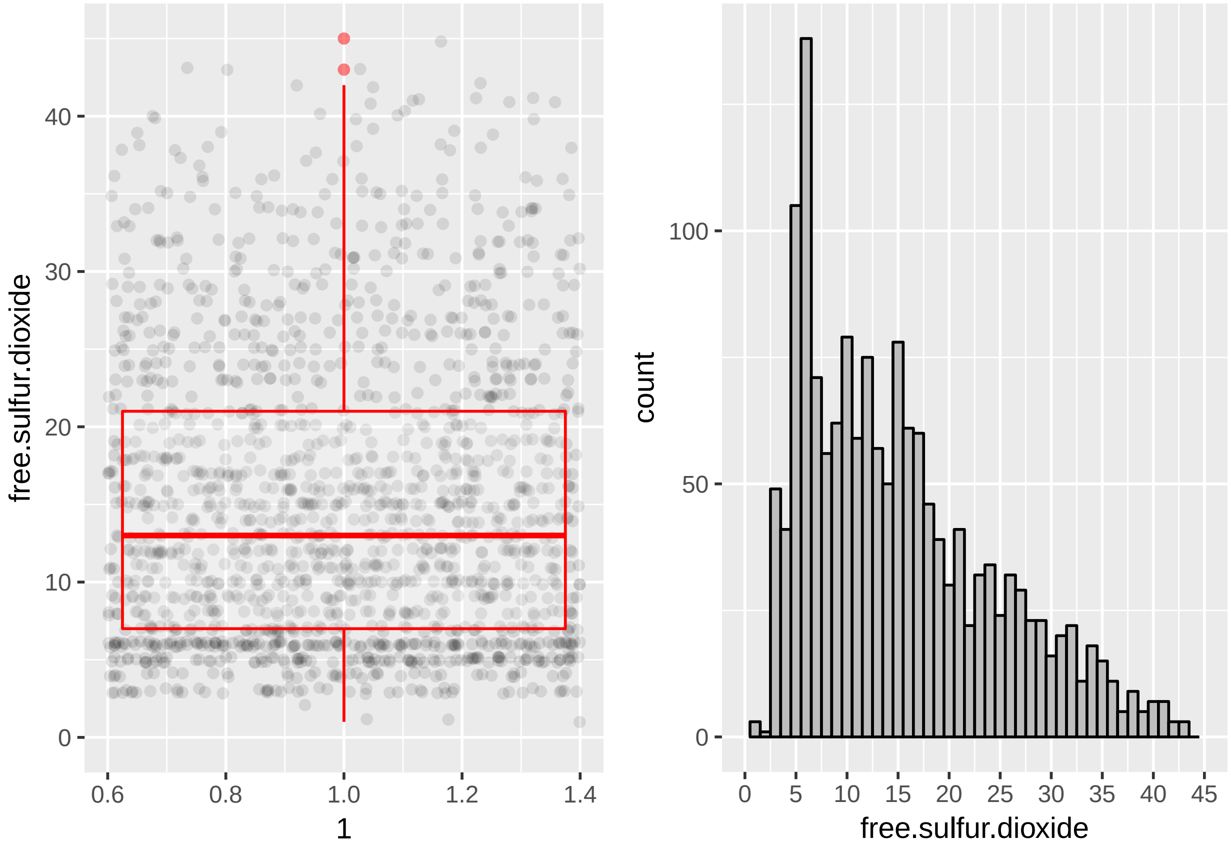


```
cat("Distribution has peak value around 0.7")
```

```
## Distribution has peak value around 0.7
```

```
#Free Sulfur Dioxide
grid.arrange(ggplot(dat, aes( x = 1, y = free.sulfur.dioxide )) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ) +
               scale_y_continuous(lim = c(0,45)),
ggplot(data = dat, aes(x = free.sulfur.dioxide)) +
  geom_histogram(binwidth = 1, color = 'black',fill = I('gray')) +
  scale_x_continuous(breaks = seq(0,80,5), lim = c(0,45)), ncol = 2)
```

```
## Warning: Removed 24 rows containing non-finite values (stat_boxplot).

## Warning: Removed 26 rows containing missing values (geom_point).

## Warning: Removed 24 rows containing non-finite values (stat_bin).
```

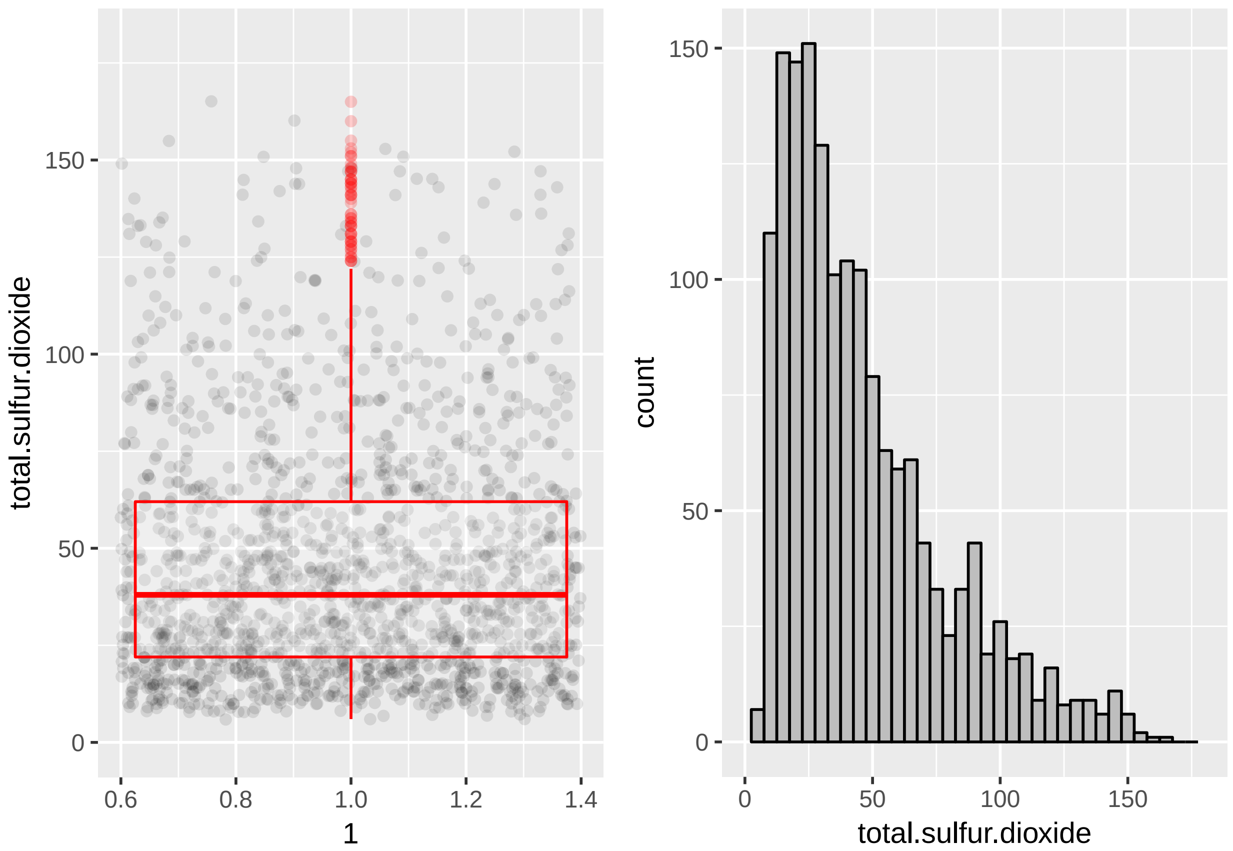


```r
cat("Distribution has peak value around 7")
```

```
## Distribution has peak value around 7
```

```r
#Total Sulfur Dioxide
grid.arrange(ggplot(dat, aes( x = 1, y = total.sulfur.dioxide )) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ) +
               scale_y_continuous(lim = c(0,180)),
ggplot(data = dat, aes(x = total.sulfur.dioxide)) +
  geom_histogram(binwidth = 5, color = 'black',fill = I('gray')) +
  scale_x_continuous(lim = c(0,180)), ncol = 2)
```

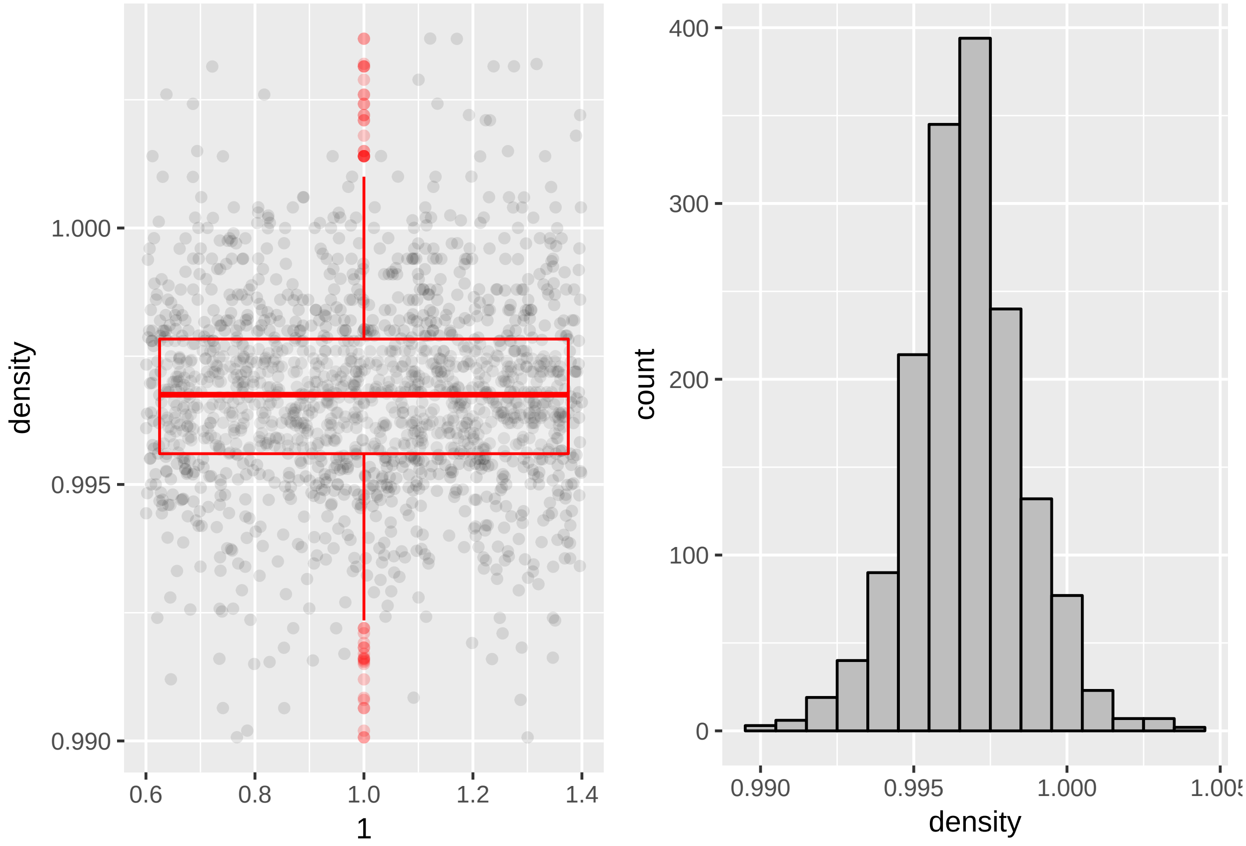
```
## Warning: Removed 2 rows containing non-finite values (stat_boxplot).

## Warning: Removed 2 rows containing missing values (geom_point).

## Warning: Removed 2 rows containing non-finite values (stat_bin).
```



```r
cat("Graph look skew on the left")
```

```
## Graph look skew on the left
```

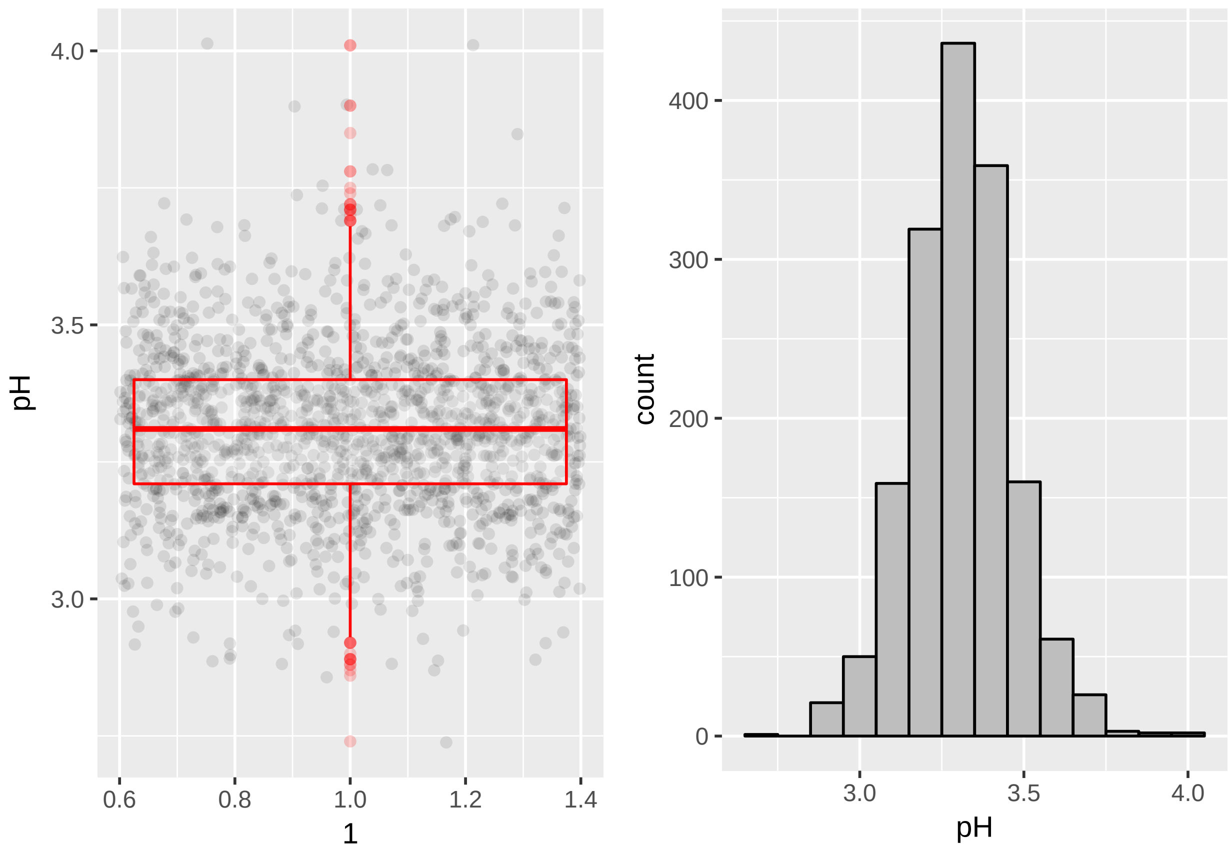
```r
#Density
grid.arrange(ggplot(dat, aes( x = 1, y = density)) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ),
ggplot(data = dat, aes(x = density)) +
  geom_histogram(binwidth = 0.001, color = 'black',fill = I('gray')), ncol = 2)
```



```
cat("Graph look normal distribution")
```

```
## Graph look normal distribution
```

```
#pH
grid.arrange(ggplot(dat, aes( x = 1, y = pH)) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ),
ggplot(data = dat, aes(x = pH)) +
  geom_histogram(binwidth = 0.1, color = 'black',fill = I('gray')), ncol = 2)
```



```
cat("Graph look normal distribution")
```

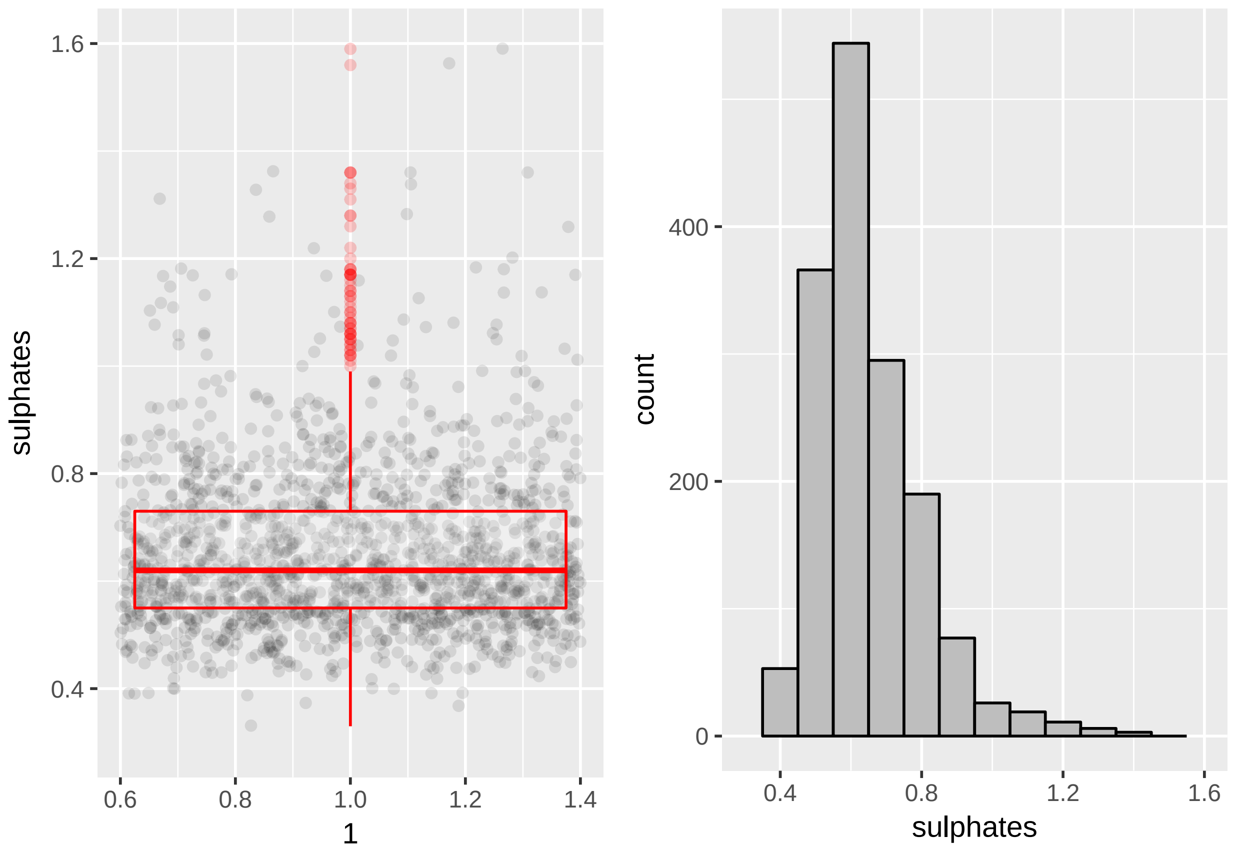
```
## Graph look normal distribution
```

```
#Sulphates
grid.arrange(ggplot(dat, aes( x = 1, y = sulphates)) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ) +
               scale_y_continuous(lim = c(0.3,1.6)),
ggplot(data = dat, aes(x = sulphates)) +
  geom_histogram(binwidth = 0.1, color = 'black',fill = I('gray')) +
  scale_x_continuous(lim = c(0.3,1.6)), ncol = 2)
```

```
## Warning: Removed 6 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 6 rows containing missing values (geom_point).
```

```
## Warning: Removed 6 rows containing non-finite values (stat_bin).
```

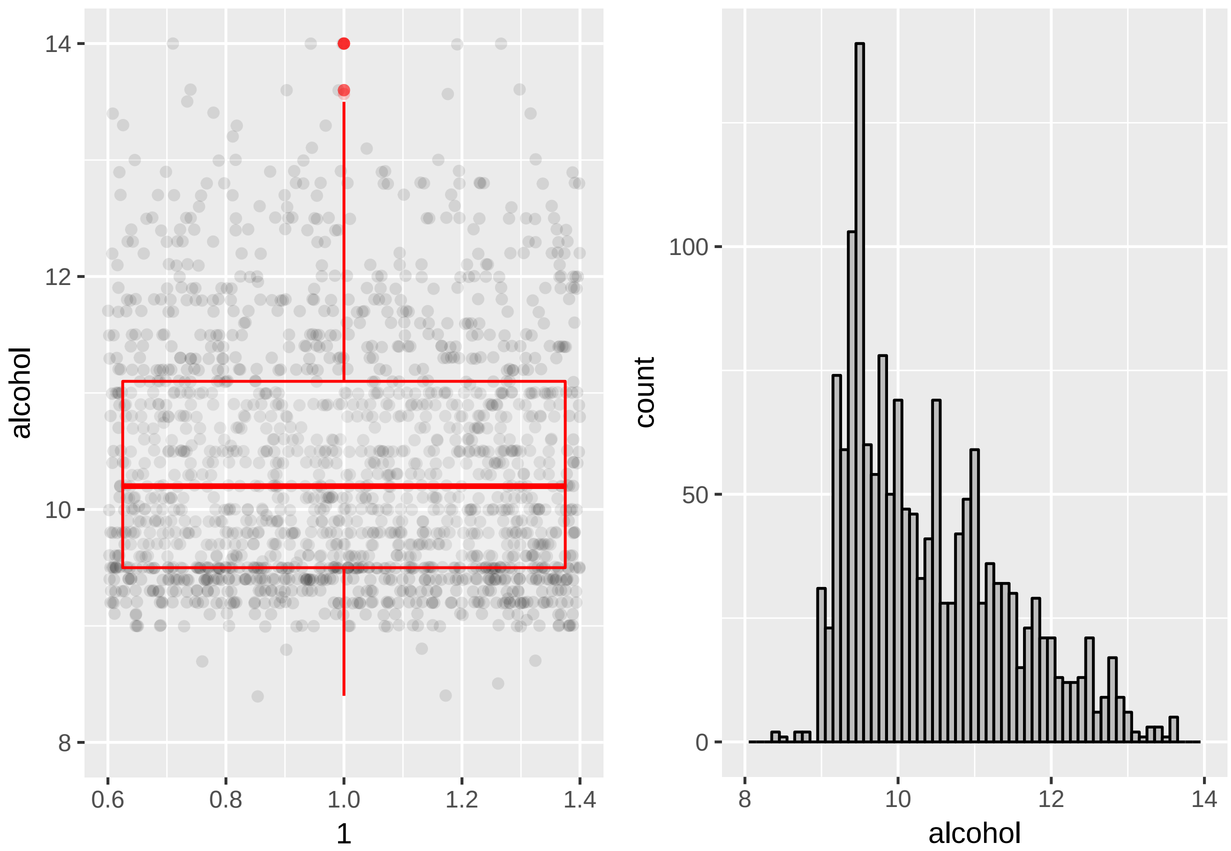


```
cat("Distribution has skew on the left")
```

```
## Distribution has skew on the left
```

```
#Alcohol
grid.arrange(ggplot(dat, aes( x = 1, y = alcohol)) +
               geom_jitter(alpha = 0.1 ) +
               geom_boxplot(alpha = 0.2, color = 'red' ) +
               scale_y_continuous(lim = c(8,14)),
ggplot(data = dat, aes(x = alcohol)) +
  geom_histogram(binwidth = 0.1, color = 'black',fill = I('gray')) +
  scale_x_continuous(lim = c(8,14)), ncol = 2)
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).

## Warning: Removed 3 rows containing missing values (geom_point).

## Warning: Removed 1 rows containing non-finite values (stat_bin).
```



```
cat("Distribution has skew on the left")
```

```
## Distribution has skew on the left
```

```
cat("Summary, after we look into graph for each feature we couldn't decide which one has effect to quali
```

```
## Summary, after we look into graph for each feature we couldn't decide which one has effect to quality
```

```
#Find correlation
cor(dat$fixed.acidity,as.numeric(dat$quality))
```

```
## [1] 0.1240516
```

```
cor(dat$volatile.acidity,as.numeric(dat$quality))
```

```
## [1] -0.3905578
```

```
cor(dat$citric.acid,as.numeric(dat$quality))
```

```
## [1] 0.2263725
```

```
cor(dat$residual.sugar,as.numeric(dat$quality))
```

```
## [1] 0.01373164
```

```
cor(dat$chlorides,as.numeric(dat$quality))
```

```
## [1] -0.1289066
```

```
cor(dat$free.sulfur.dioxide,as.numeric(dat$quality))
```

```
## [1] -0.05065606
```

```r
cor(dat$total.sulfur.dioxide,as.numeric(dat$quality))
```

```
## [1] -0.1851003
```

```r
cor(dat$density,as.numeric(dat$quality))
```

```
## [1] -0.1749192
```

```r
cor(dat$pH,as.numeric(dat$quality))
```

```
## [1] -0.05773139
```

```r
cor(dat$sulphates,as.numeric(dat$quality))
```

```
## [1] 0.2513971
```

```r
cor(dat$alcohol,as.numeric(dat$quality))
```

```
## [1] 0.4761663
```

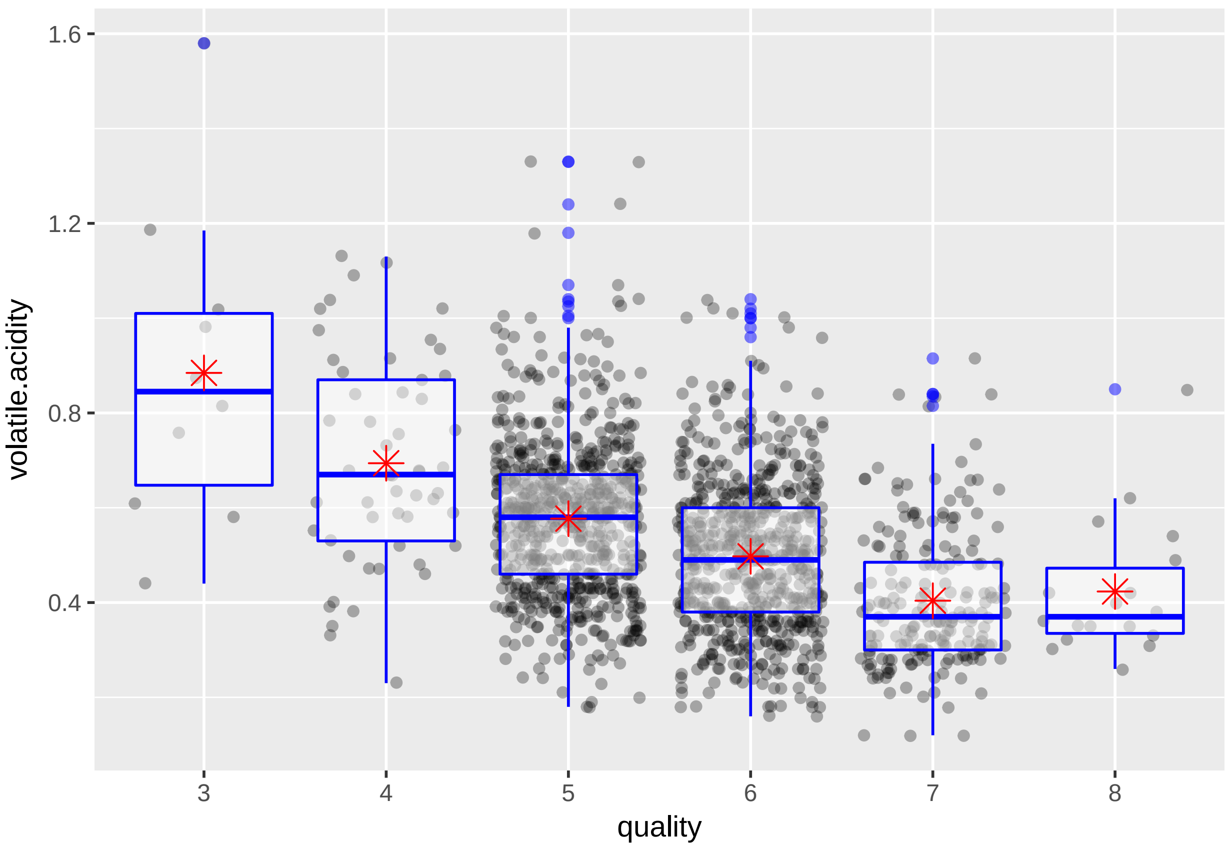
```r
cat("After we applied correlation we can see strong correlation about 4 feature that have signification
```

```
## After we applied correlation we can see strong correlation about 4 feature that have signification wi
```

```r
cat("First, we obtained correlation value between volatile.acidity and quality eqaul",cor(dat$volatile.a
```

```
## First, we obtained correlation value between volatile.acidity and quality eqaul -0.3905578 which has
```

```r
ggplot(data=dat, aes(x = quality, y = volatile.acidity)) +
  geom_jitter( alpha = .3) +
  geom_boxplot(alpha = .5,color = 'blue') +
  stat_summary(fun.y = "mean",
               geom = "point",
               color = "red",
               shape = 8,
               size = 4)
```



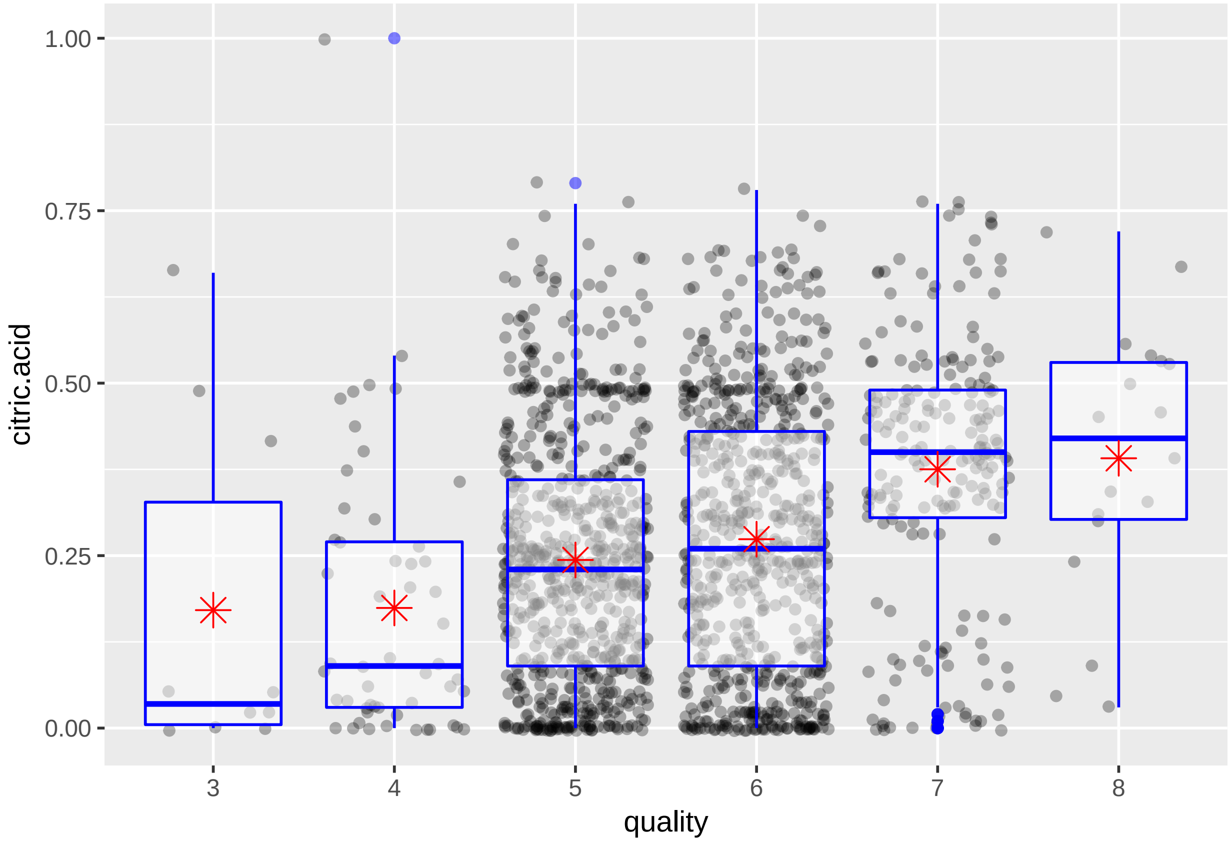
```
cat("From boxplot we can see that if we increase volatile acidity, quality will degrade")
```

```
## From boxplot we can see that if we increase volatile acidity, quality will degrade
```

```
cat("Second, we obtained another strong correlation value between citric.acid and quality eqaul",cor(da
```

```
## Second, we obtained another strong correlation value between citric.acid and quality eqaul 0.2263725
```

```
ggplot(data=dat, aes(x=quality, y=citric.acid)) +
  geom_jitter( alpha = .3) +
  geom_boxplot(alpha = .5,color = 'blue') +
  stat_summary(fun.y = "mean",
               geom = "point",
               color = "red",
               shape = 8,
               size = 4)
```



```r
cat("From boxplot we can see that if we increase volatile acidity, quality will increase too")
```

```
## From boxplot we can see that if we increase volatile acidity, quality will increase too
```

```r
cat("Third, we obtained another strong correlation value between sulphates and quality eqaul",cor(dat$su
```

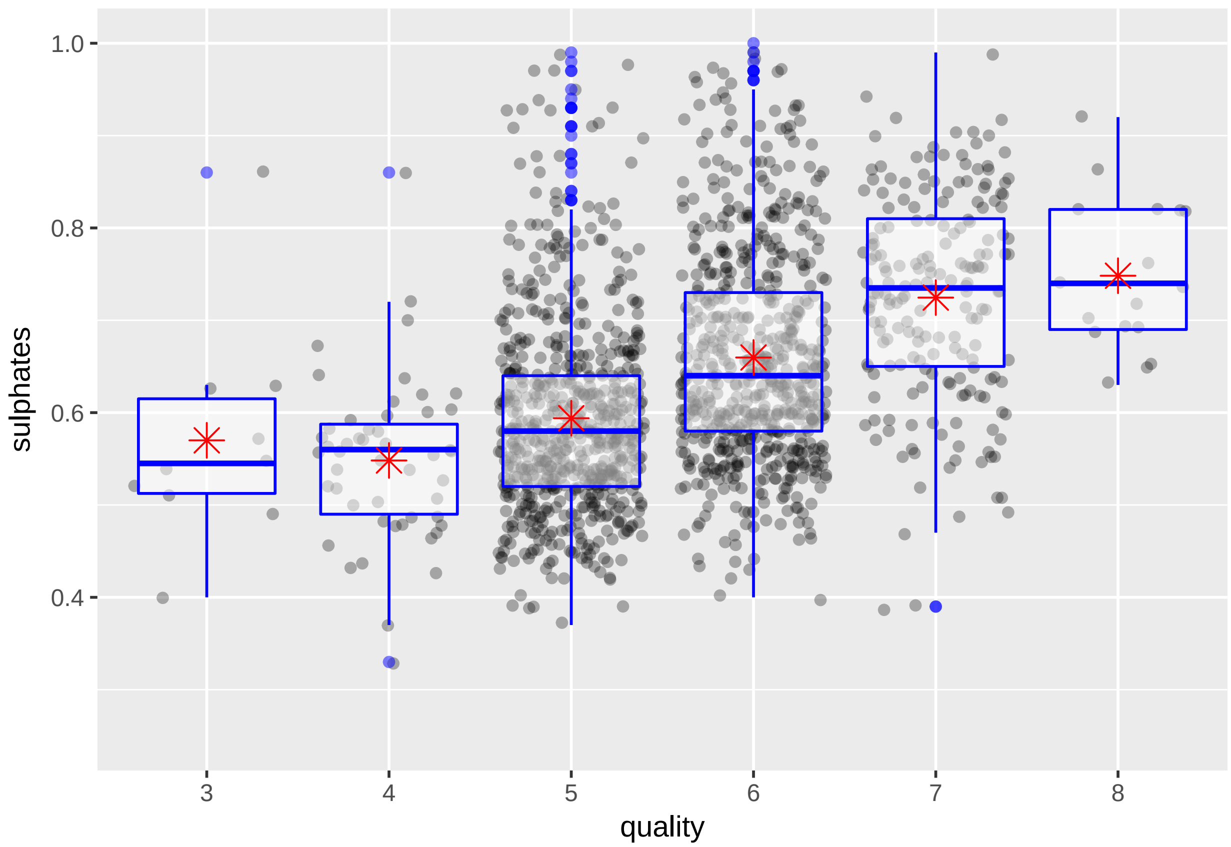
```
## Third, we obtained another strong correlation value between sulphates and quality eqaul 0.2513971
```

```r
ggplot(data=dat, aes(x=quality, y=sulphates)) +
  geom_jitter( alpha = .3) +
  geom_boxplot(alpha = .5,color = 'blue') +
  scale_y_continuous(lim = c(0.25,1)) +
  stat_summary(fun.y = "mean",
               geom = "point",
               color = "red",
               shape = 8,
               size = 4)
```

```
## Warning: Removed 58 rows containing non-finite values (stat_boxplot).
```

```
## Warning: Removed 58 rows containing non-finite values (stat_summary).
```

```
## Warning: Removed 59 rows containing missing values (geom_point).
```



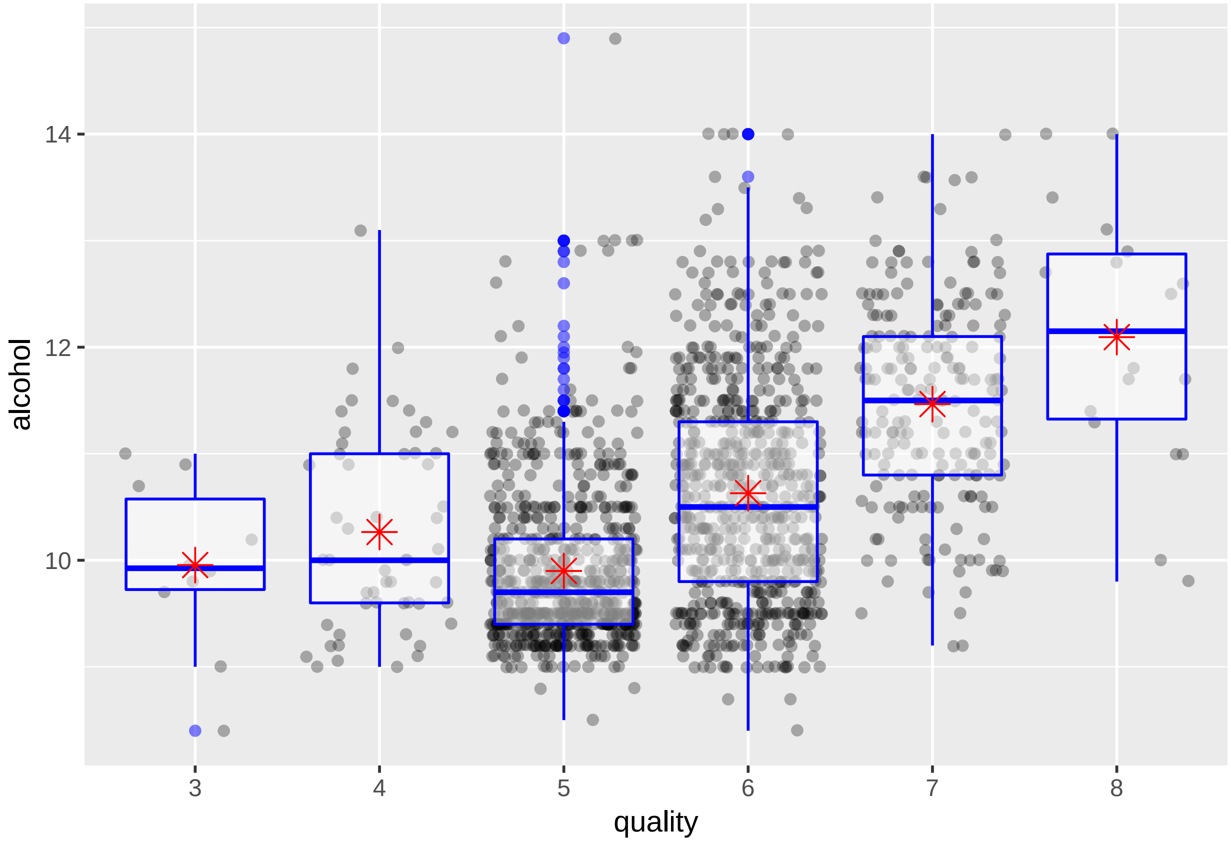
```
cat("From boxplot we can see that wine will have better quality if have strong sulphates")
```

```
## From boxplot we can see that wine will have better quality if have strong sulphates
```

```
cat("Fourth, we obtained another strong correlation value between alcohol and quality eqaul",cor(dat$al
```

```
## Fourth, we obtained another strong correlation value between alcohol and quality eqaul 0.4761663
```

```
ggplot(data=dat, aes(x=quality, y=alcohol)) +
  geom_jitter( alpha = .3) +
  geom_boxplot(alpha = .5,color = 'blue') +
  stat_summary(fun.y = "mean",
               geom = "point",
               color = "red",
               shape = 8,
               size = 4)
```



```
cat("From boxplot we can see that wine will have better quality if have strong alcohol too")
```

```
## From boxplot we can see that wine will have better quality if have strong alcohol too
```

```
cat("In summary, we can conclude that 1.Alcohol 2.Sulphates 3.Critic Acid 4.Volatile Acid are effect wi
```

```
## In summary, we can conclude that 1.Alcohol 2.Sulphates 3.Critic Acid 4.Volatile Acid are effect with
```

```
cat("We need to know which feature from four above that has the most effect with quality of wine respec
```

```
## We need to know which feature from four above that has the most effect with quality of wine respecti
```

```
set.seed(1234)
training_data <- sample_frac(dat)
test_data <- dat[ !dat$X %in% training_data$X, ]
m1 <- lm(as.numeric(quality) ~ alcohol, data = training_data)
m2 <- update(m1, ~ . + sulphates)
m3 <- update(m2, ~ . + volatile.acidity)
m4 <- update(m3, ~ . + citric.acid)
m4
```

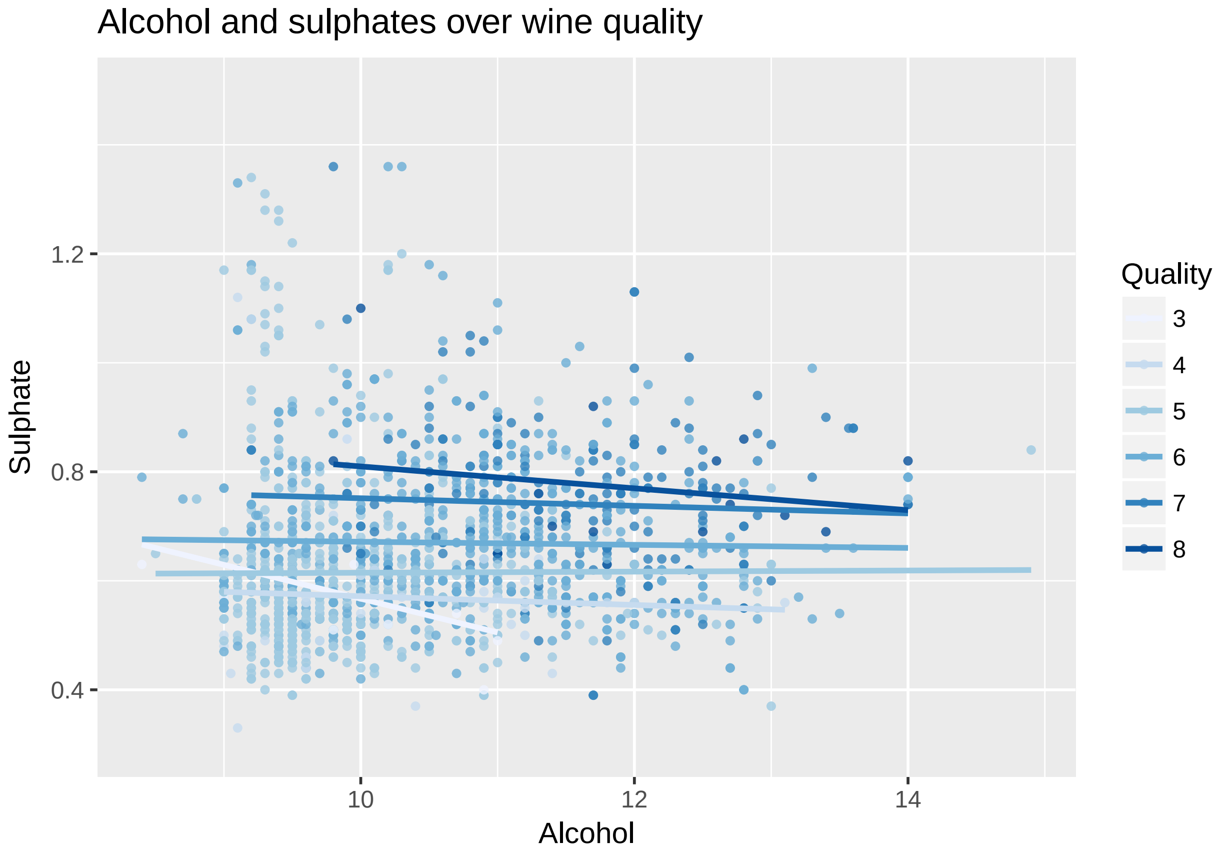
```
##
## Call:
## lm(formula = as.numeric(quality) ~ alcohol + sulphates + volatile.acidity +
##     citric.acid, data = training_data)
##
## Coefficients:
##      (Intercept)           alcohol          sulphates  volatile.acidity
##          0.64592           0.30908            0.69552          -1.26506
##      citric.acid
##         -0.07913
```

```
cat("From linear fit model we can see that alcoho and sulphates has the most effect with quality of win
```

```
## From linear fit model we can see that alcoho and sulphates has the most effect with quality of wine
```

```
ggplot(data = dat,
       aes(y = sulphates, x = alcohol,
           color = quality)) +
  geom_point(alpha = 0.8, size = 1) +
  geom_smooth(method = "lm", se = FALSE,size=1)  +
  scale_y_continuous(limits=c(0.3,1.5)) +
  ylab("Sulphate") +
  xlab("Alcohol") +
  scale_color_brewer(type='seq',
                     guide=guide_legend(title='Quality')) +
  ggtitle("Alcohol and sulphates over wine quality")
```

```
## Warning: Removed 8 rows containing non-finite values (stat_smooth).

## Warning: Removed 8 rows containing missing values (geom_point).
```



```
cat("From graph above we can conclude that better alcohol and sulphates will make a better quality of w
```

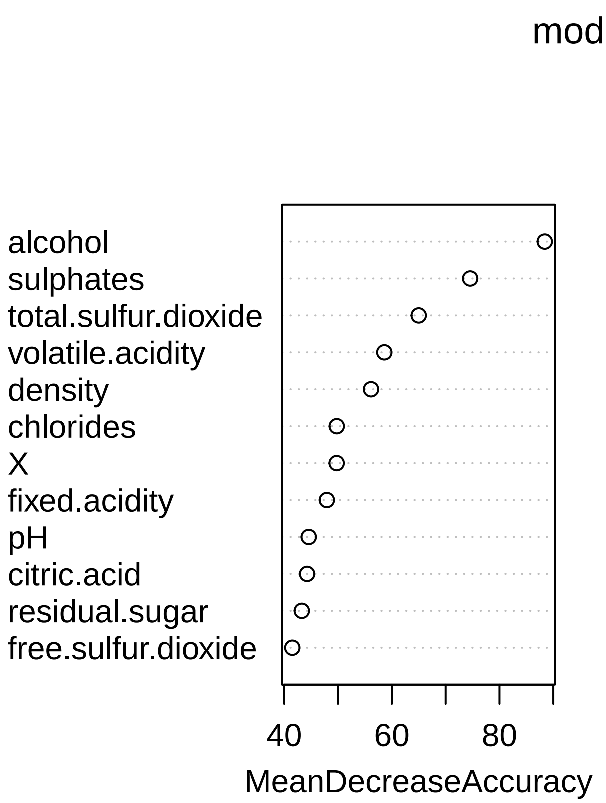
```
## From graph above we can conclude that better alcohol and sulphates will make a better quality of win
```

```
cat("Howeve, we still curious to know which one has the most effect between alcohol and sulphates. So, 
## Howeve, we still curious to know which one has the most effect between alcohol and sulphates. So, we
```

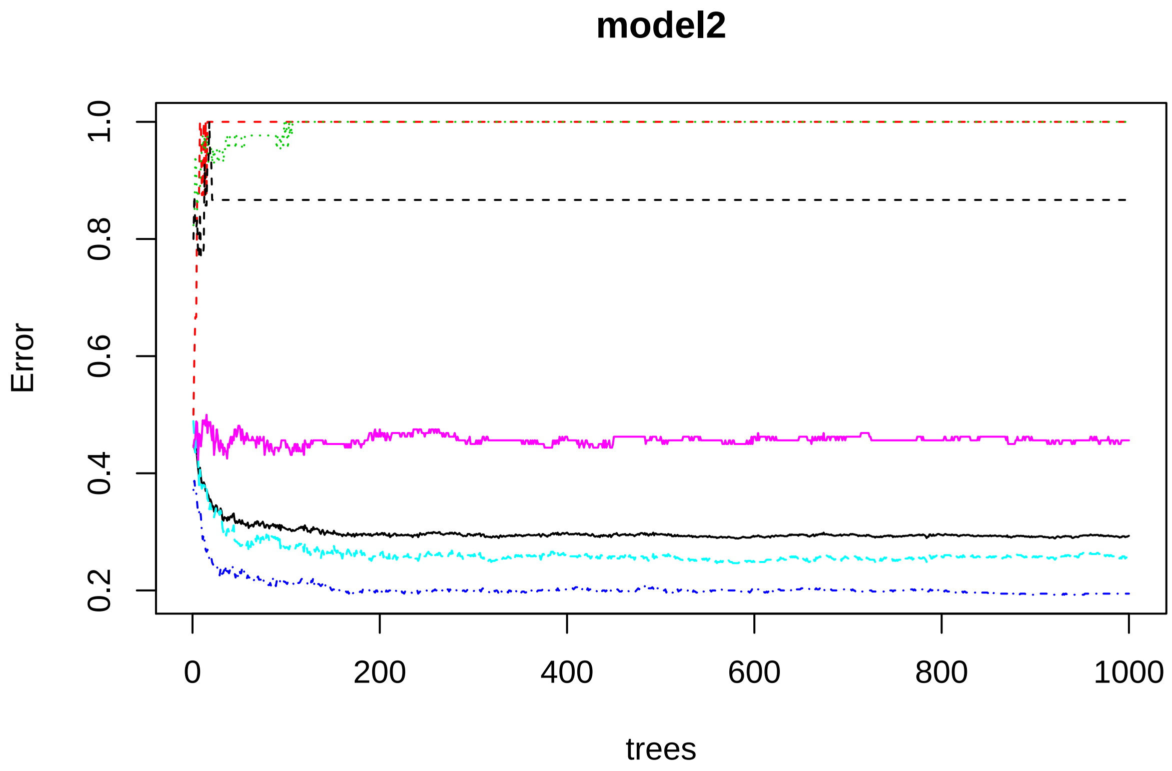
```r
#Random forest to find best feature for quality of wine
dat.rdforest$quality <- as.factor(dat.rdforest$quality)

index <- createDataPartition(dat.rdforest$quality, p=0.8, list=FALSE)
train <- dat.rdforest[index,]
test <- dat.rdforest[-index,]
model <- rpart(quality~., data=train)
prediction <- predict(model, test, type="class")
confusionMatrix(prediction, test$quality)
```

```
## Confusion Matrix and Statistics
##
##              Reference
## Prediction  3  4  5  6  7  8
##          3  0  0  0  0  0  0
##          4  0  0  0  0  0  0
##          5  2  6 97 44  4  1
##          6  0  4 34 70 24  0
##          7  0  0  5 13 11  2
##          8  0  0  0  0  0  0
##
## Overall Statistics
##
##                Accuracy : 0.5615
##                  95% CI : (0.505, 0.6169)
##     No Information Rate : 0.429
##     P-Value [Acc > NIR] : 1.44e-06
##
##                   Kappa : 0.2844
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: 3 Class: 4 Class: 5 Class: 6 Class: 7 Class: 8
## Sensitivity          0.000000  0.00000   0.7132   0.5512  0.28205 0.000000
## Specificity          1.000000  1.00000   0.6851   0.6737  0.92806 1.000000
## Pos Pred Value            NaN      NaN   0.6299   0.5303  0.35484      NaN
## Neg Pred Value       0.993691  0.96845   0.7607   0.6919  0.90210 0.990536
## Prevalence           0.006309  0.03155   0.4290   0.4006  0.12303 0.009464
## Detection Rate       0.000000  0.00000   0.3060   0.2208  0.03470 0.000000
## Detection Prevalence 0.000000  0.00000   0.4858   0.4164  0.09779 0.000000
## Balanced Accuracy    0.500000  0.50000   0.6992   0.6124  0.60505 0.500000
```

```r
model2<- randomForest(quality~., importance=TRUE, proximity=TRUE,train, ntree=1000)
varImpPlot(model2)
```


model2
alcohol
sulphates
total.sulfur.dioxide
volatile.acidity
density
chlorides
X
fixed.acidity
pH
citric.acid
residual.sugar
free.sulfur.dioxide
40
60
80
MeanDecreaseAccuracy
alcohol
sulphates
total.sulfur.dioxide
volatile.acidity
X
density
chlorides
pH
fixed.acidity
citric.acid
residual.sugar
free.sulfur.dioxide
0
40
80
MeanDecreaseGini

```
plot(model2)
```



```r
cat("We can see that Alcohol and Sulphates has the most effect to quality of wine according to correlat
```

```
## We can see that Alcohol and Sulphates has the most effect to quality of wine according to correlatio
```

```r
cat("To prove that random forest has better accuracy than decision tree. We can see as below.")
```

```
## To prove that random forest has better accuracy than decision tree. We can see as below.
```

```r
prediction2 <- predict(model2, test)
confusionMatrix(prediction2, test$quality)
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction   3   4   5   6   7   8
##          3   0   0   0   0   0   0
##          4   0   0   0   2   0   0
##          5   2   6 109  30   2   0
##          6   0   4  24  88  22   2
##          7   0   0   3   7  15   1
##          8   0   0   0   0   0   0
##
## Overall Statistics
##
##                Accuracy : 0.6688
##                  95% CI : (0.614, 0.7204)
##     No Information Rate : 0.429
##     P-Value [Acc > NIR] : < 2.2e-16
```

```
##
##                   Kappa : 0.458
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: 3 Class: 4 Class: 5 Class: 6 Class: 7 Class: 8
## Sensitivity          0.000000 0.000000   0.8015   0.6929  0.38462 0.000000
## Specificity          1.000000 0.993485   0.7790   0.7263  0.96043 1.000000
## Pos Pred Value            NaN 0.000000   0.7315   0.6286  0.57692      NaN
## Neg Pred Value       0.993691 0.968254   0.8393   0.7797  0.91753 0.990536
## Prevalence           0.006309 0.031546   0.4290   0.4006  0.12303 0.009464
## Detection Rate       0.000000 0.000000   0.3438   0.2776  0.04732 0.000000
## Detection Prevalence 0.000000 0.006309   0.4700   0.4416  0.08202 0.000000
## Balanced Accuracy    0.500000 0.496743   0.7902   0.7096  0.67252 0.500000
```

```
cat("The random forest give better accuracy than decition tree.")
```

```
## The random forest give better accuracy than decition tree.
```