WineQuality

Suchitra

2/18/2017

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:

Variable definition:

Fixed acidity: Acid involved with wine that are non-volatile. Volatile acidity: Acetic acid in wine that at high levels leads to a taste of vinegar. Citric acid: Present in small quantities, add freshness and flavor to wine. Residual sugar: Sugra thats left after the fermentation stops. chlorides:The amount of salts in wine. Free sulphur dioxide:Prevents microbial growth and oxidation of wine. Total: Amount of free + bound forms of SO2. Density: Depends on the percent alcohol and sugar content. PH : Acidic or basic. Sulphates: Wine additive which ontributes to sulphur dioxide. Alcohol: Percent alcohol content in wine. Quality: Output variable.

Questions: 1. How is human tasting of wine related to the chemical properties.

Getting familiar with the data.

```
wine_data <- read.csv("wineQualityReds.csv", header = TRUE, sep=",")
summary(wine_data)
```

```
##        X          fixed.acidity   volatile.acidity  citric.acid
##  Min.   :   1.0   Min.   : 4.60   Min.   :0.1200   Min.   :0.000
##  1st Qu.: 400.5   1st Qu.: 7.10   1st Qu.:0.3900   1st Qu.:0.090
##  Median : 800.0   Median : 7.90   Median :0.5200   Median :0.260
##  Mean   : 800.0   Mean   : 8.32   Mean   :0.5278   Mean   :0.271
##  3rd Qu.:1199.5   3rd Qu.: 9.20   3rd Qu.:0.6400   3rd Qu.:0.420
##  Max.   :1599.0   Max.   :15.90   Max.   :1.5800   Max.   :1.000
##  residual.sugar     chlorides       free.sulfur.dioxide
##  Min.   : 0.900   Min.   :0.01200   Min.   : 1.00
##  1st Qu.: 1.900   1st Qu.:0.07000   1st Qu.: 7.00
##  Median : 2.200   Median :0.07900   Median :14.00
##  Mean   : 2.539   Mean   :0.08747   Mean   :15.87
##  3rd Qu.: 2.600   3rd Qu.:0.09000   3rd Qu.:21.00
##  Max.   :15.500   Max.   :0.61100   Max.   :72.00
##  total.sulfur.dioxide    density             pH          sulphates
##  Min.   :  6.00       Min.   :0.9901   Min.   :2.740   Min.   :0.3300
##  1st Qu.: 22.00       1st Qu.:0.9956   1st Qu.:3.210   1st Qu.:0.5500
##  Median : 38.00       Median :0.9968   Median :3.310   Median :0.6200
##  Mean   : 46.47       Mean   :0.9967   Mean   :3.311   Mean   :0.6581
##  3rd Qu.: 62.00       3rd Qu.:0.9978   3rd Qu.:3.400   3rd Qu.:0.7300
##  Max.   :289.00       Max.   :1.0037   Max.   :4.010   Max.   :2.0000
##     alcohol         quality
```

```
##  Min.   : 8.40   Min.   :3.000
##  1st Qu.: 9.50   1st Qu.:5.000
##  Median :10.20   Median :6.000
##  Mean   :10.42   Mean   :5.636
##  3rd Qu.:11.10   3rd Qu.:6.000
##  Max.   :14.90   Max.   :8.000
```

```
names(wine_data)
```

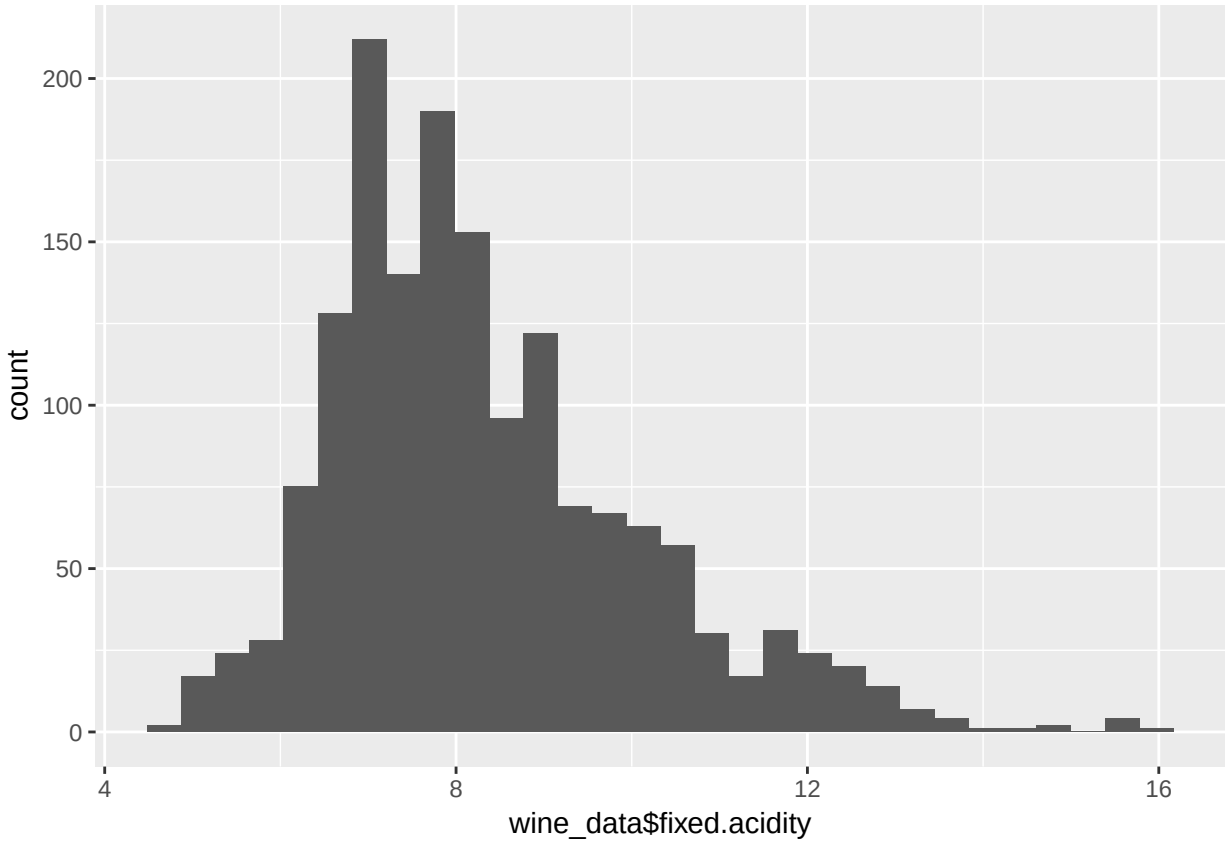
```
##  [1] "X"                    "fixed.acidity"        "volatile.acidity"
##  [4] "citric.acid"          "residual.sugar"       "chlorides"
##  [7] "free.sulfur.dioxide"  "total.sulfur.dioxide" "density"
## [10] "pH"                   "sulphates"            "alcohol"
## [13] "quality"
```

Variable Identification: All the variables are categorical or ordinal variables. the chemical properties are predictor variables and the outcome is the quality.

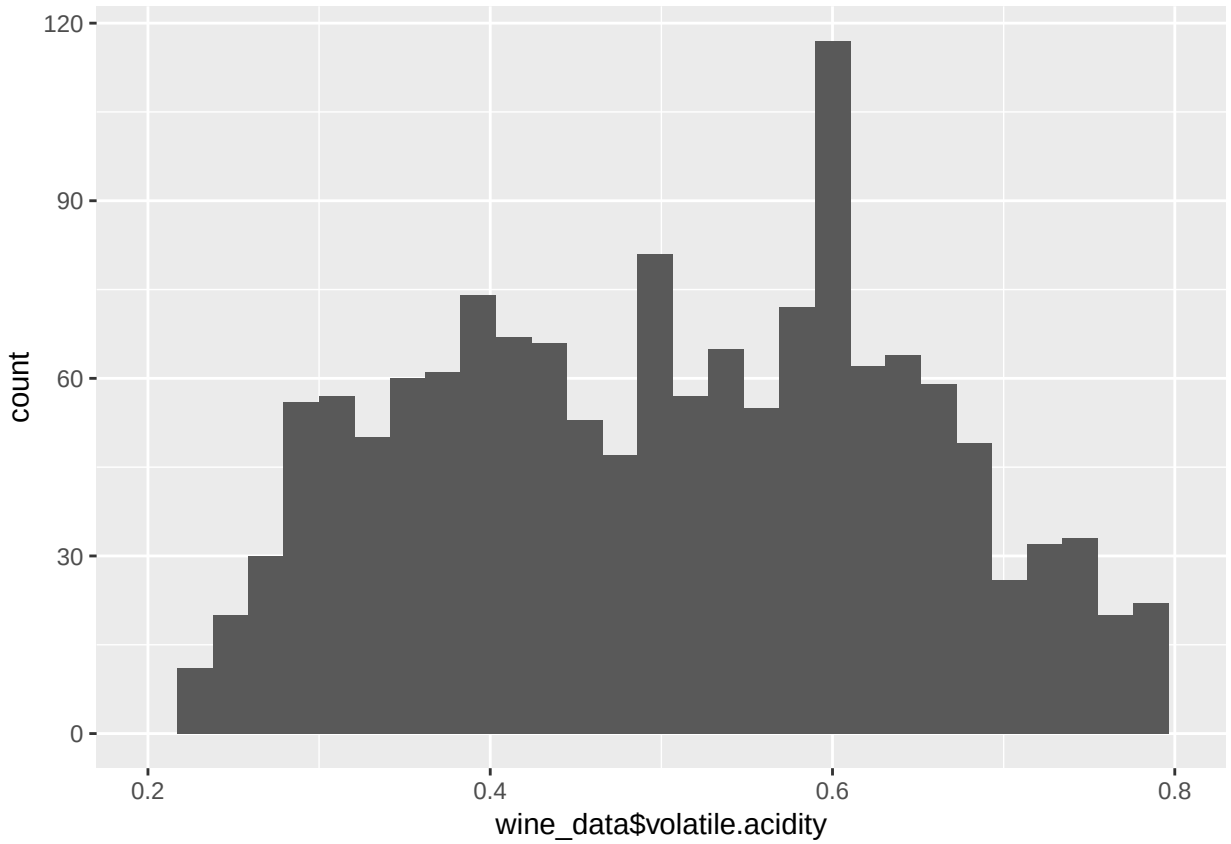
Univariate Data Analysis.

You can also embed plots, for example:

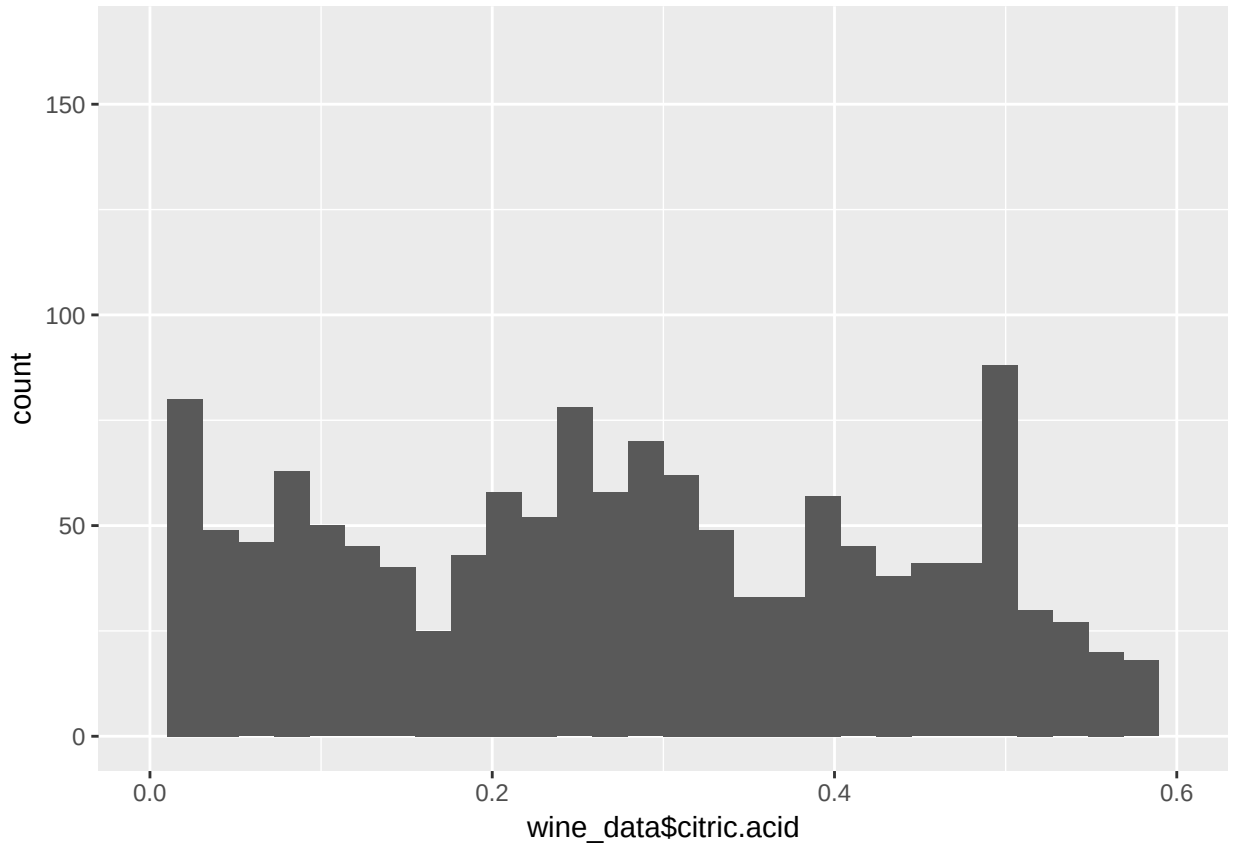
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 121 rows containing non-finite values (stat_bin).
## Warning: Removed 1 rows containing missing values (geom_bar).
```



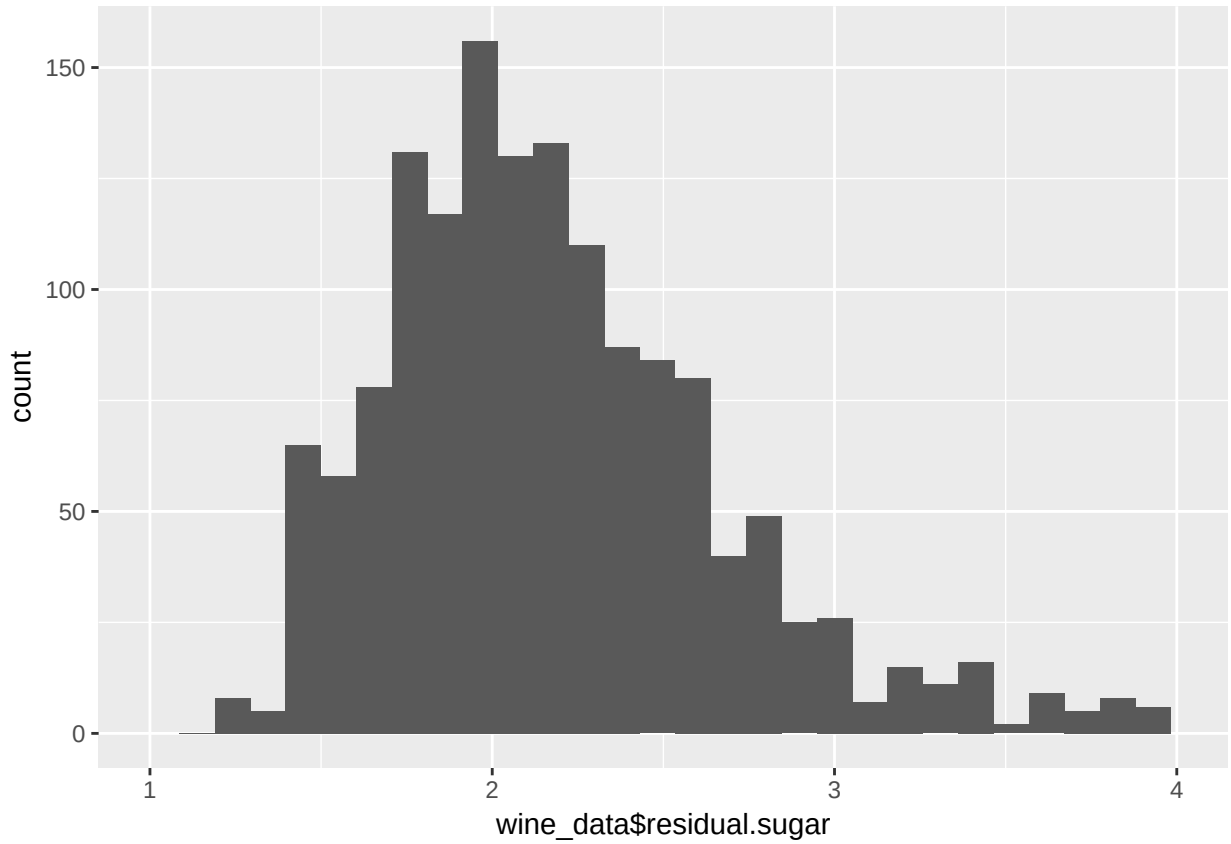

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 78 rows containing non-finite values (stat_bin).
```



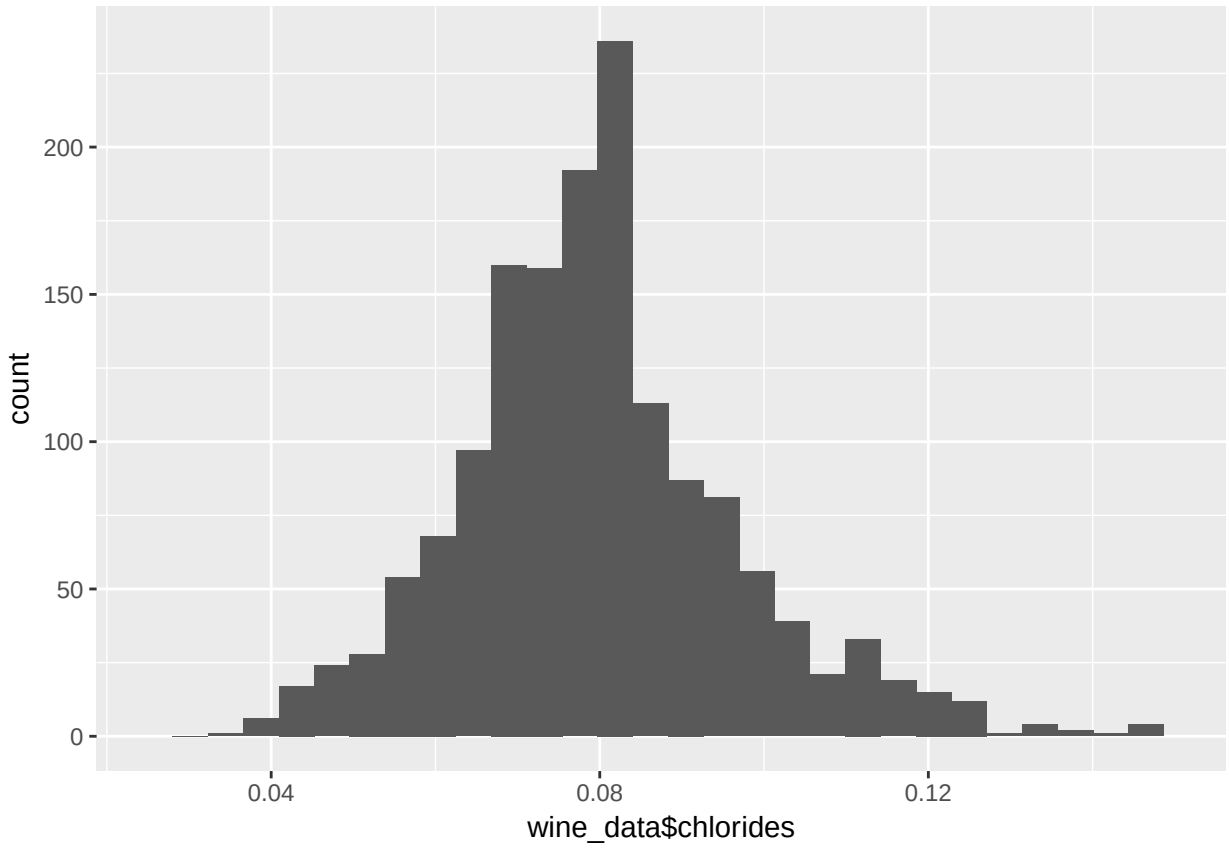

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## Warning: Removed 127 rows containing non-finite values (stat_bin).
```

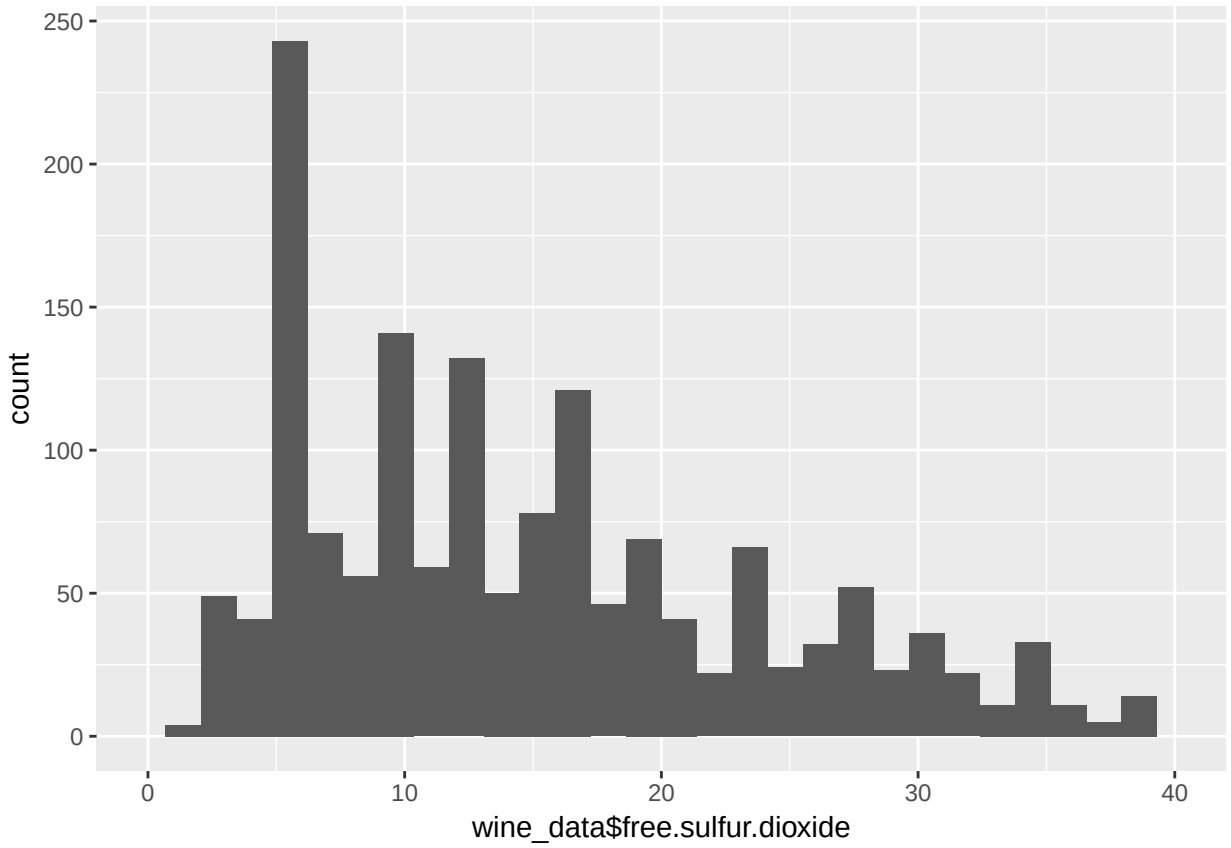
```
## Warning: Removed 1 rows containing missing values (geom_bar).
```



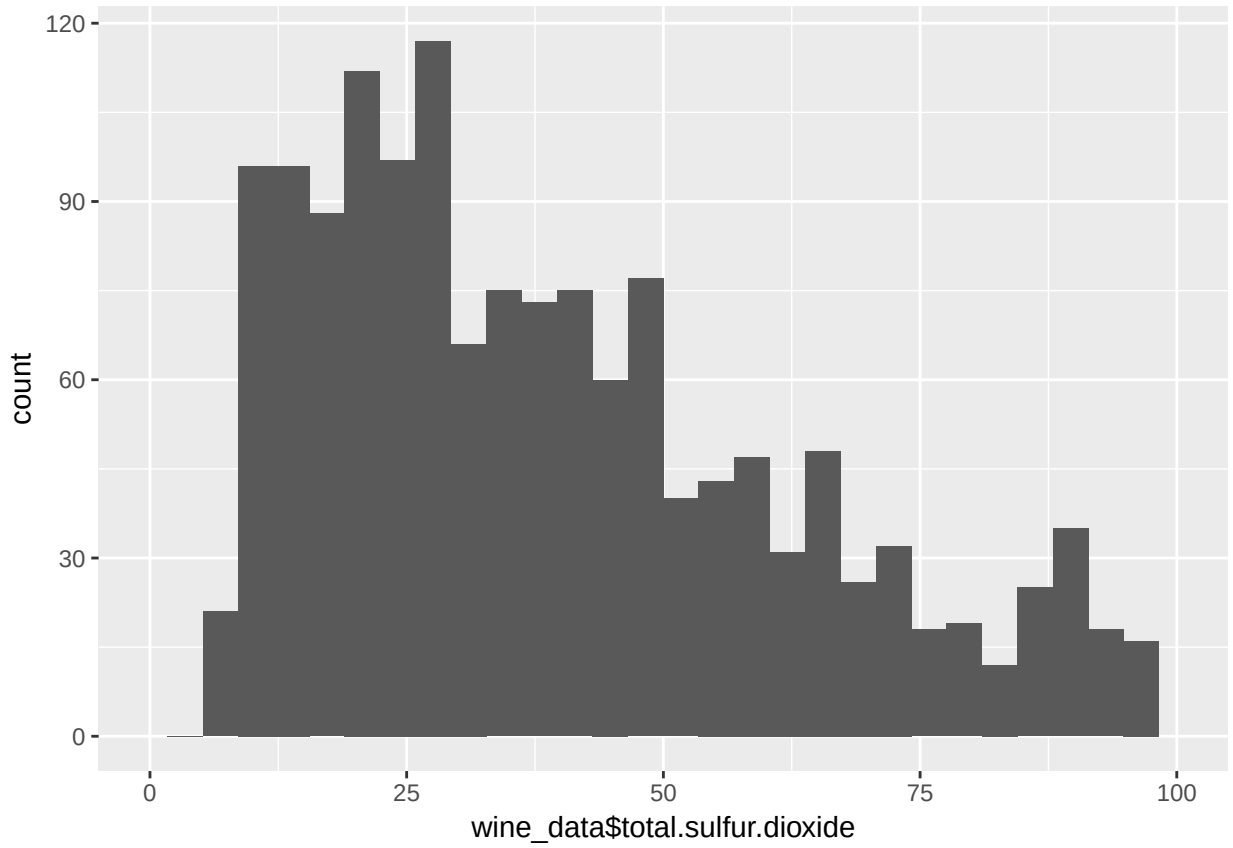

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 69 rows containing non-finite values (stat_bin).
## Warning: Removed 1 rows containing missing values (geom_bar).
```



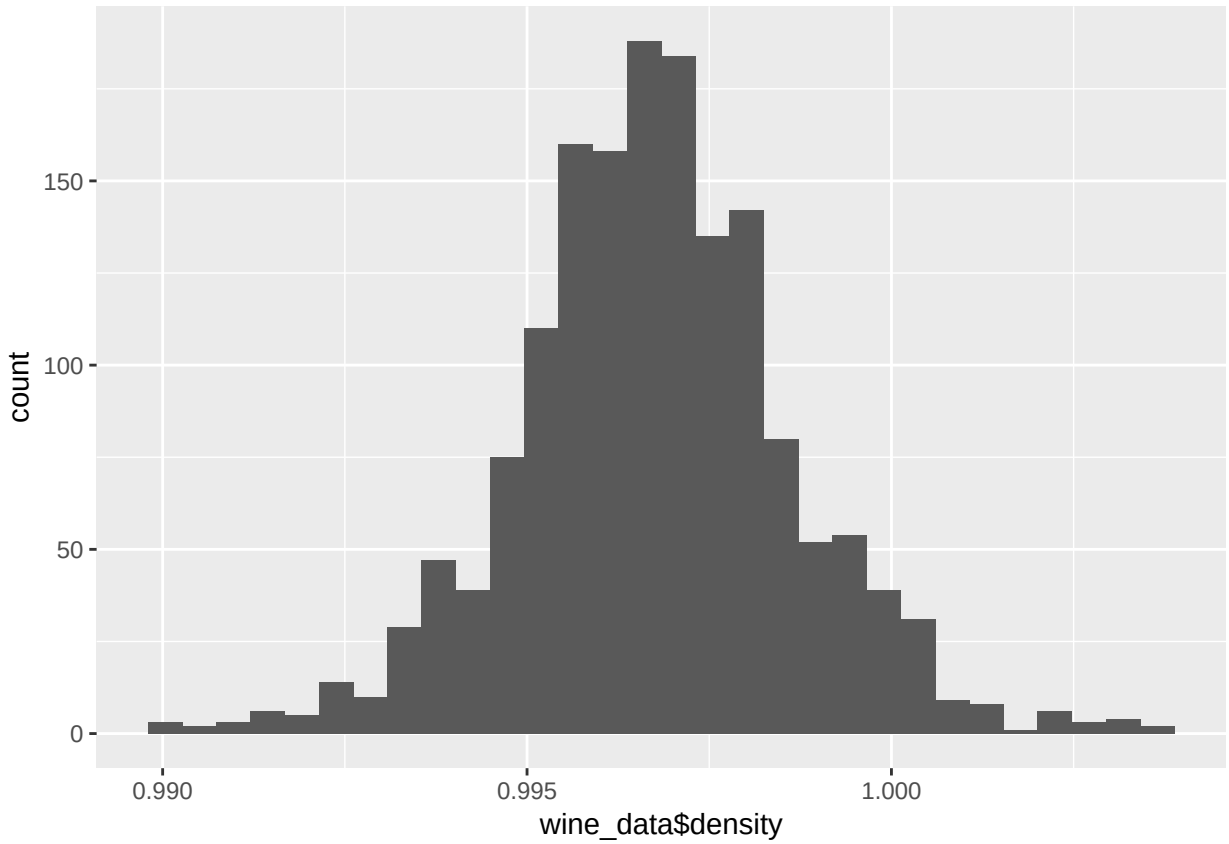

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 41 rows containing non-finite values (stat_bin).
```



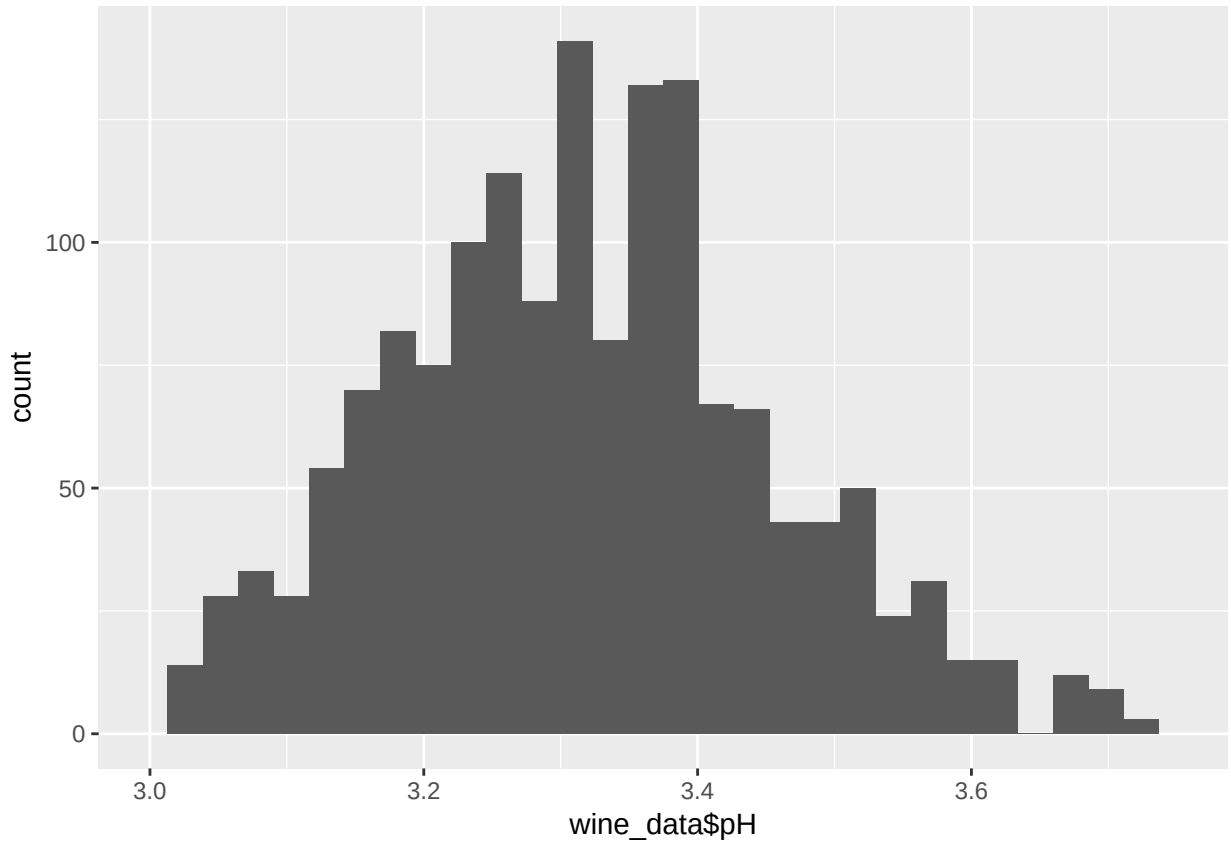

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 127 rows containing non-finite values (stat_bin).
```



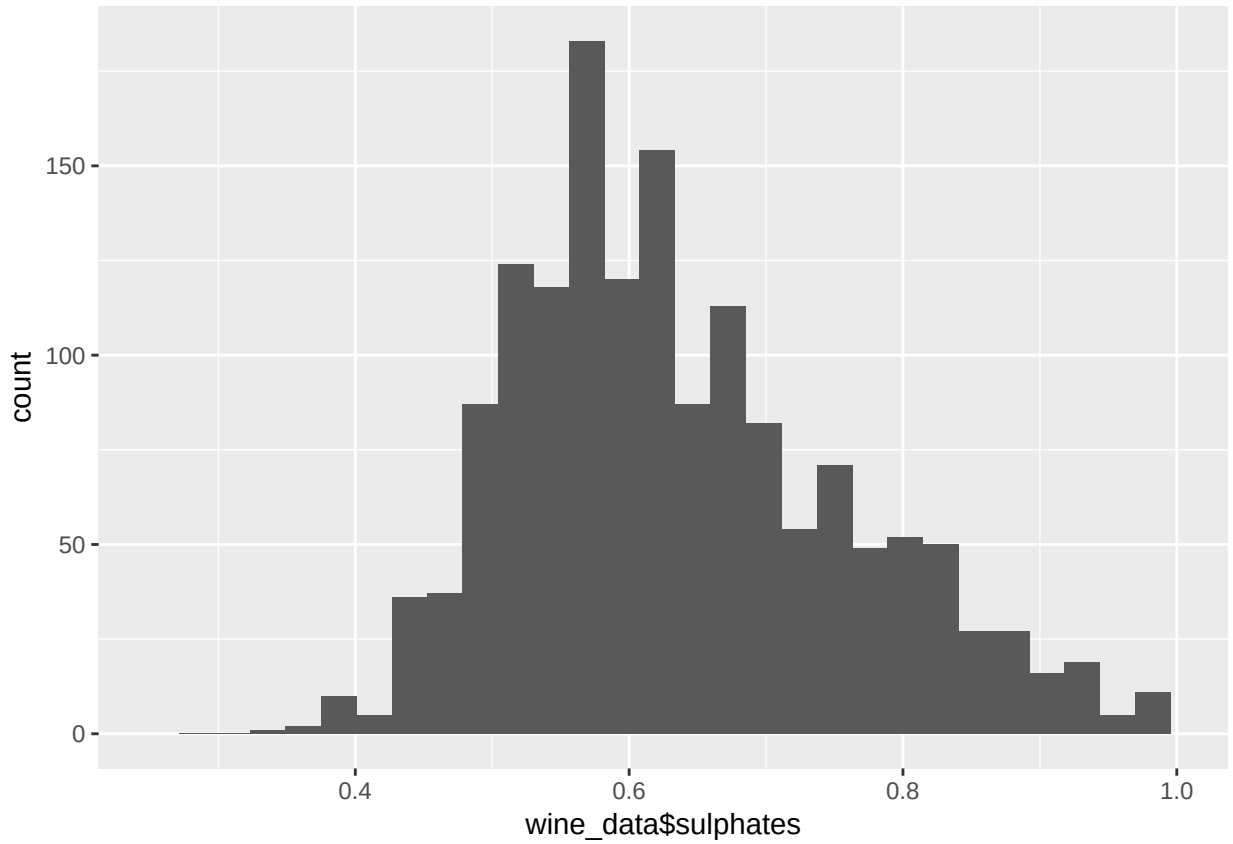

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



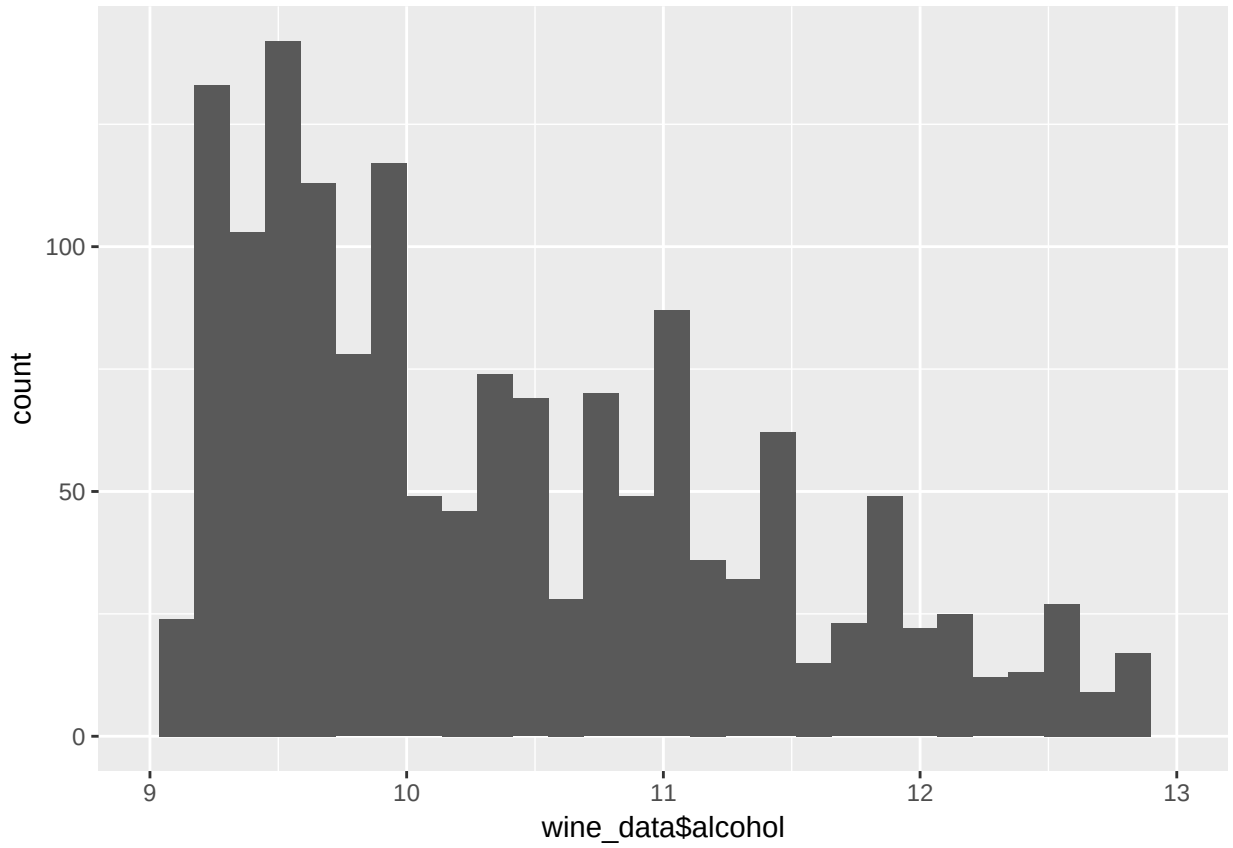

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 36 rows containing non-finite values (stat_bin).
```



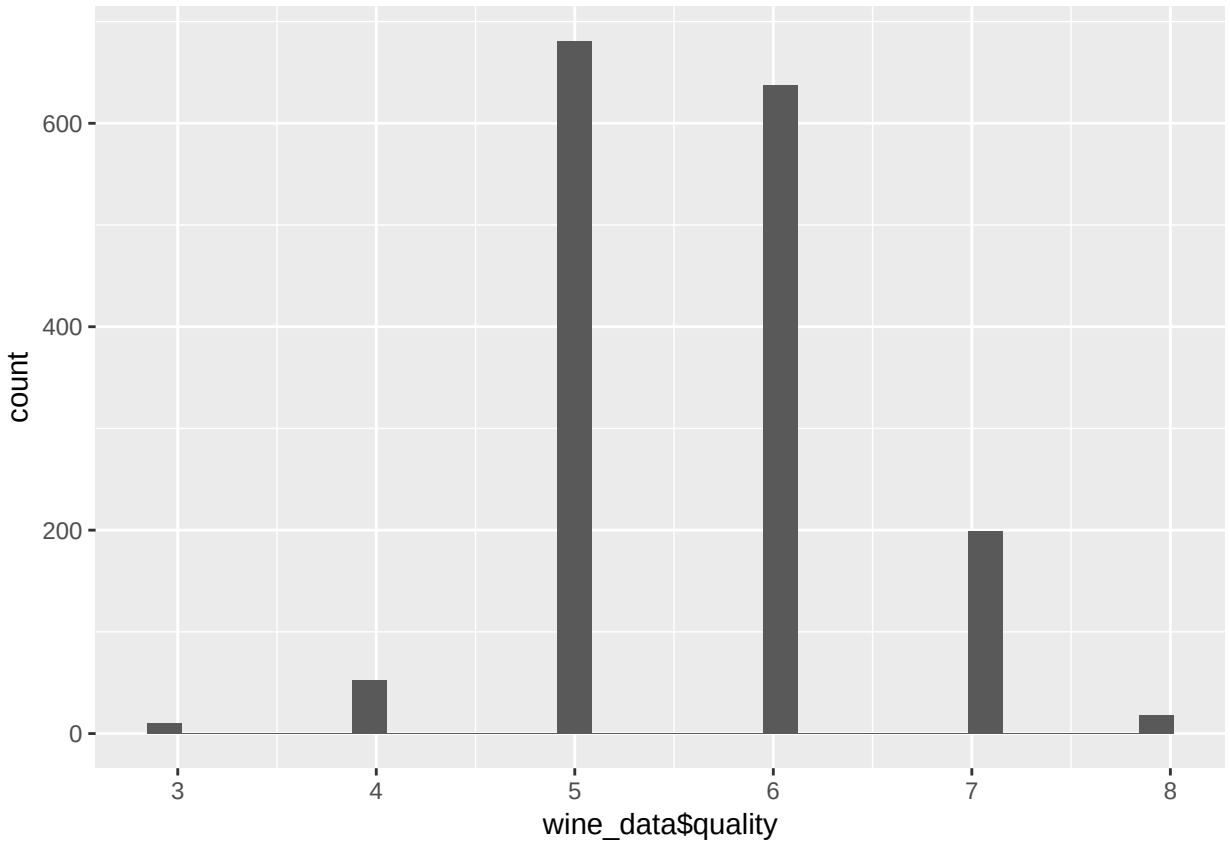

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 58 rows containing non-finite values (stat_bin).
## Warning: Removed 1 rows containing missing values (geom_bar).
```




```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 30 rows containing non-finite values (stat_bin).
## Warning: Removed 1 rows containing missing values (geom_bar).
```




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



There are outliers in fixed acidity.

Summary Statistics

```
summary(wine_data$fixed.acidity)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    4.60    7.10    7.90    8.32    9.20   15.90
```

```
summary(wine_data$volatile.acidity)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.1200  0.3900  0.5200  0.5278  0.6400  1.5800
```

```
summary(wine_data$citric.acid)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.000   0.090   0.260   0.271   0.420   1.000
```

```
summary(wine_data$residual.sugar)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   0.900   1.900   2.200   2.539   2.600  15.500
```

```
summary(wine_data$chlorides)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
## 0.01200 0.07000 0.07900 0.08747 0.09000 0.61100
```

```r
summary(wine_data$free.sulfur.dioxide)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    1.00    7.00   14.00   15.87   21.00   72.00
```

```r
summary(wine_data$total.sulfur.dioxide)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    6.00   22.00   38.00   46.47   62.00  289.00
```

```r
summary(wine_data$density)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.9901  0.9956  0.9968  0.9967  0.9978  1.0040
```

```r
summary(wine_data$pH)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   2.740   3.210   3.310   3.311   3.400   4.010
```

```r
summary(wine_data$sulphates)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.3300  0.5500  0.6200  0.6581  0.7300  2.0000
```

```r
summary(wine_data$alcohol)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##    8.40    9.50   10.20   10.42   11.10   14.90
```

```r
summary(wine_data$quality)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   3.000   5.000   6.000   5.636   6.000   8.000
```

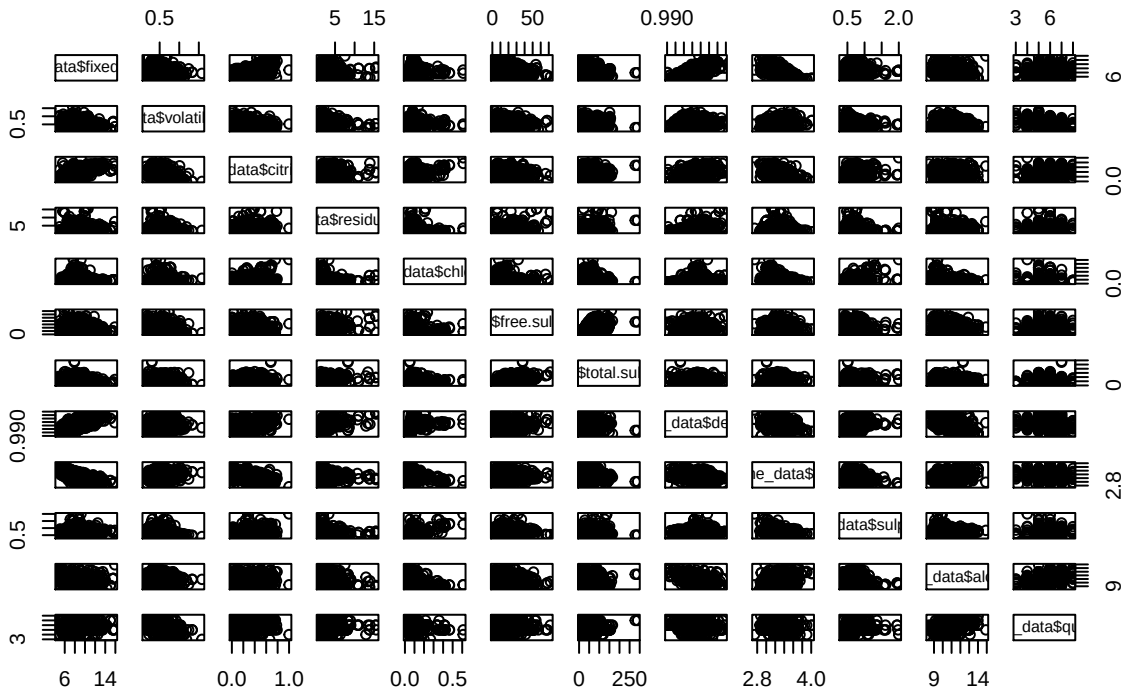
Correlation

```r
cor(x=wine_data[,2:13], y=wine_data[,2:13])
```

```
##                       fixed.acidity volatile.acidity citric.acid
## fixed.acidity           1.00000000     -0.256130895  0.67170343
## volatile.acidity       -0.25613089      1.000000000 -0.55249568
## citric.acid             0.67170343     -0.552495685  1.00000000
## residual.sugar          0.11477672      0.001917882  0.14357716
## chlorides               0.09370519      0.061297772  0.20382291
## free.sulfur.dioxide    -0.15379419     -0.010503827 -0.06097813
## total.sulfur.dioxide   -0.11318144      0.076470005  0.03553302
## density                 0.66804729      0.022026232  0.36494718
## pH                     -0.68297819      0.234937294 -0.54190414
## sulphates               0.18300566     -0.260986685  0.31277004
## alcohol                -0.06166827     -0.202288027  0.10990325
## quality                 0.12405165     -0.390557780  0.22637251
##                       residual.sugar    chlorides free.sulfur.dioxide
## fixed.acidity           0.114776724  0.093705186       -0.153794193
## volatile.acidity        0.001917882  0.061297772       -0.010503827
## citric.acid             0.143577162  0.203822914       -0.060978129
```

```
## residual.sugar          1.000000000  0.055609535         0.187048995
## chlorides               0.055609535  1.000000000         0.005562147
## free.sulfur.dioxide     0.187048995  0.005562147         1.000000000
## total.sulfur.dioxide    0.203027882  0.047400468         0.667666450
## density                 0.355283371  0.200632327        -0.021945831
## pH                     -0.085652422 -0.265026131         0.070377499
## sulphates               0.005527121  0.371260481         0.051657572
## alcohol                 0.042075437 -0.221140545        -0.069408354
## quality                 0.013731637 -0.128906560        -0.050656057
##                       total.sulfur.dioxide     density          pH
## fixed.acidity                -0.11318144  0.66804729 -0.68297819
## volatile.acidity              0.07647000  0.02202623  0.23493729
## citric.acid                   0.03553302  0.36494718 -0.54190414
## residual.sugar                0.20302788  0.35528337 -0.08565242
## chlorides                     0.04740047  0.20063233 -0.26502613
## free.sulfur.dioxide           0.66766645 -0.02194583  0.07037750
## total.sulfur.dioxide          1.00000000  0.07126948 -0.06649456
## density                       0.07126948  1.00000000 -0.34169933
## pH                           -0.06649456 -0.34169933  1.00000000
## sulphates                     0.04294684  0.14850641 -0.19664760
## alcohol                      -0.20565394 -0.49617977  0.20563251
## quality                      -0.18510029 -0.17491923 -0.05773139
##                         sulphates     alcohol     quality
## fixed.acidity         0.183005664 -0.06166827  0.12405165
## volatile.acidity     -0.260986685 -0.20228803 -0.39055778
## citric.acid           0.312770044  0.10990325  0.22637251
## residual.sugar        0.005527121  0.04207544  0.01373164
## chlorides             0.371260481 -0.22114054 -0.12890656
## free.sulfur.dioxide   0.051657572 -0.06940835 -0.05065606
## total.sulfur.dioxide  0.042946836 -0.20565394 -0.18510029
## density               0.148506412 -0.49617977 -0.17491923
## pH                   -0.196647602  0.20563251 -0.05773139
## sulphates             1.000000000  0.09359475  0.25139708
## alcohol               0.093594750  1.00000000  0.47616632
## quality               0.251397079  0.47616632  1.00000000
```

```
pairs(~wine_data$fixed.acidity+wine_data$volatile.acidity+wine_data$citric.acid+wine_data$residual.suga
```



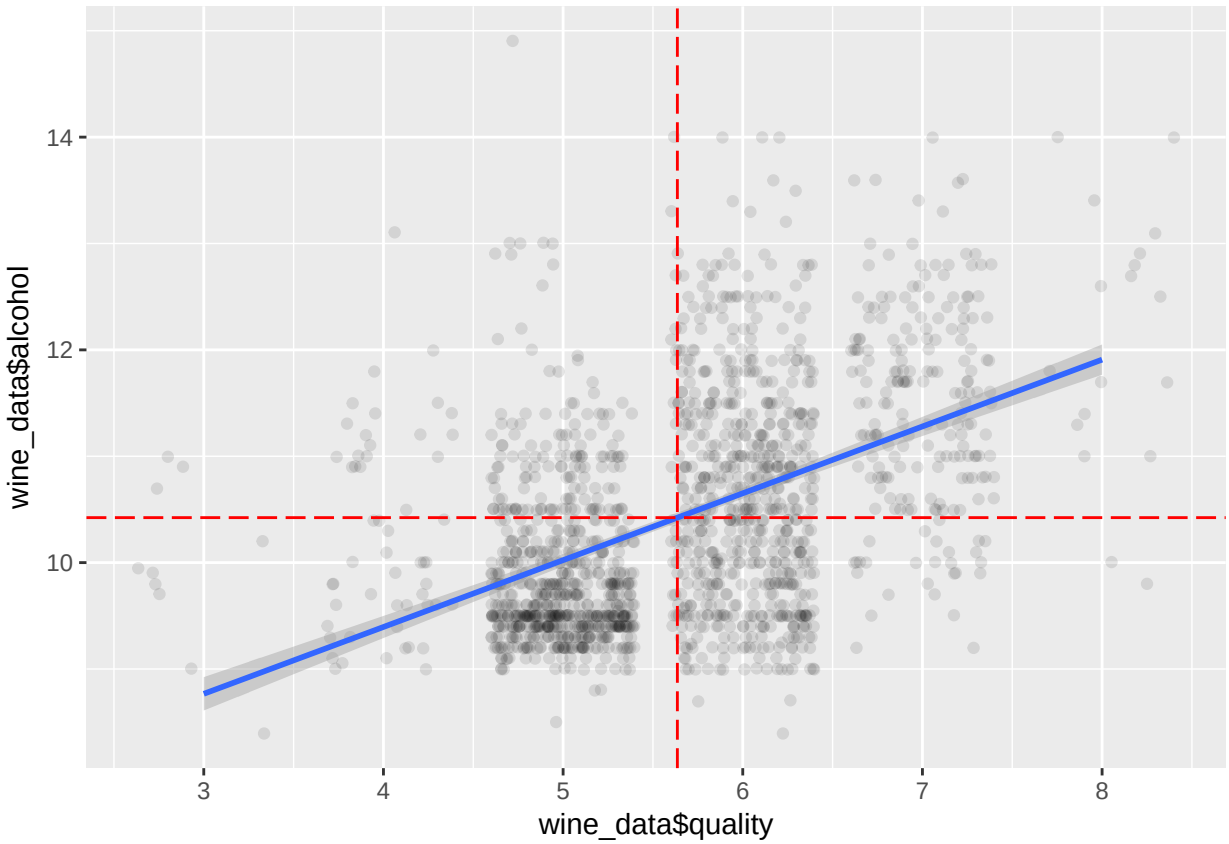
Quality of wine correlates strongly with alcohol, volatile acididty, suplhates, citric acid, as compared to the other variables. So we can consider these four variables to understand tehir association with wine quality scores.

Scatterplot matrix

Lets consider the highest correlated chemical properties.

Alcohol

```
ggplot(aes(x=wine_data$quality, y= wine_data$alcohol), data= wine_data) +
  geom_jitter(alpha=1/10)+
  scale_x_continuous(breaks = seq(0,8,1))+
  geom_smooth(method='lm', aes(group=1))+
  geom_hline(yintercept = mean(wine_data$alcohol), linetype="longdash", color='red')+
  geom_vline(xintercept = mean(wine_data$quality), linetype="longdash", color='red')
```



Relationship between alcohol and wine scores: From the scatterplot, we can see that the wine scores have a positive relationship with the percent alcohol content in wine. Wine scores in the higher range of 5 to 8, tend to have a higher alcohol content between 10 to 12.

Volatile acidity

```
tapply(wine_data$volatile.acidity, wine_data$quality, mean)
```

```
##         3         4         5         6         7         8
## 0.8845000 0.6939623 0.5770411 0.4974843 0.4039196 0.4233333
```

```
ggplot(aes(x= wine_data$quality, y=wine_data$volatile.acidity), data= wine_data)+
  geom_jitter(alpha=1/10)+
  geom_smooth(method='lm', aes(group=1))+
  geom_hline(yintercept = mean(wine_data$volatile.acidity), color='red', linetype='longdash')+
  geom_vline(xintercept = mean(wine_data$quality), color='red', linetype='longdash')
```

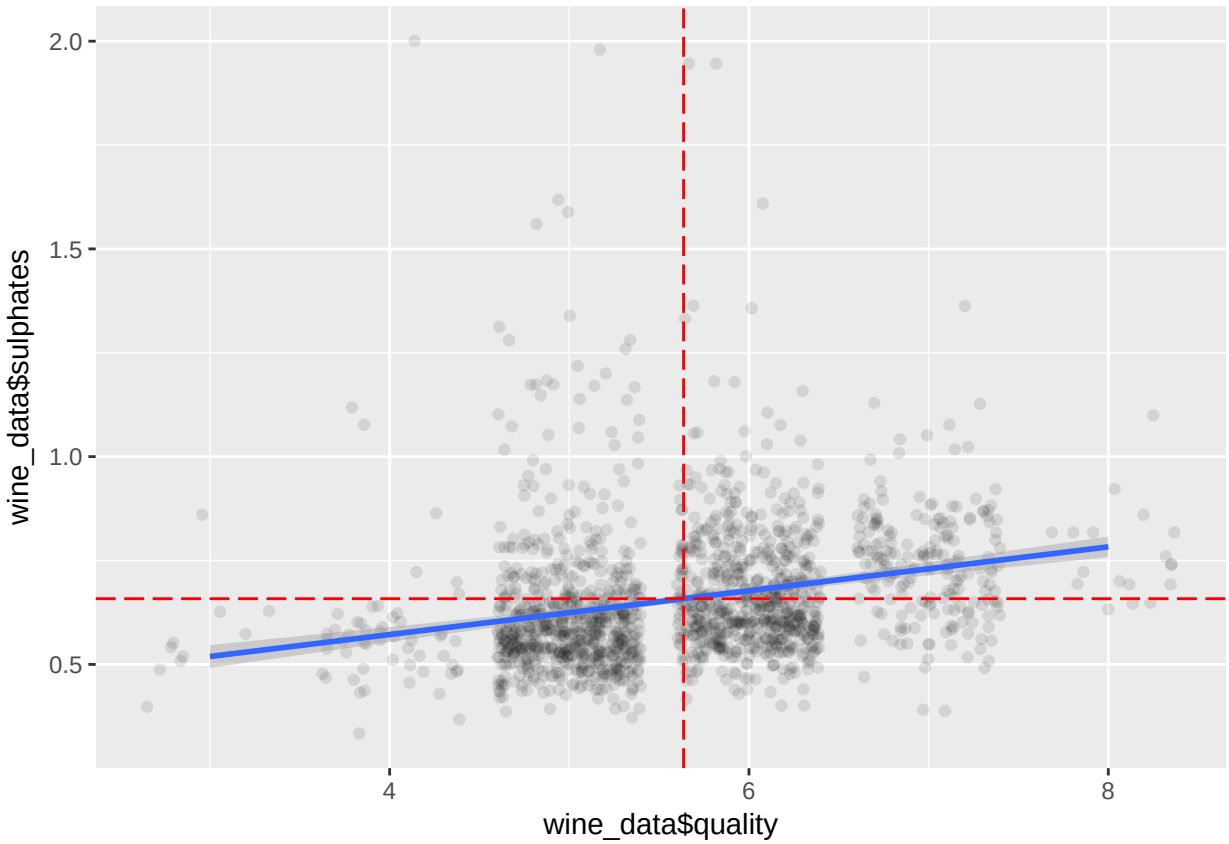
Relationship between volatile acidity and wine scores. From the pot, we can say that wines have a lower score for higher values of volatile acidity. This indicates that they have a negative association. This might be intuitive as volatile acidity in wine increases the acetiec acid content and leads to a more prominent vinegar taste. Presence of this taste is a not a good indication of a high quality wine.

Sulphates

```
tapply(wine_data$sulphates, wine_data$quality, mean)
```

```
##         3         4         5         6         7         8
## 0.5700000 0.5964151 0.6209692 0.6753292 0.7412563 0.7677778
```

```
ggplot(aes(x= wine_data$quality, y=wine_data$sulphates), data= wine_data)+
  geom_jitter(alpha=1/10)+
  geom_smooth(method='lm', aes(group=1))+
  geom_hline(yintercept = mean(wine_data$sulphates), color='red', linetype='longdash')+
  geom_vline(xintercept = mean(wine_data$quality), color='red', linetype='longdash')
```



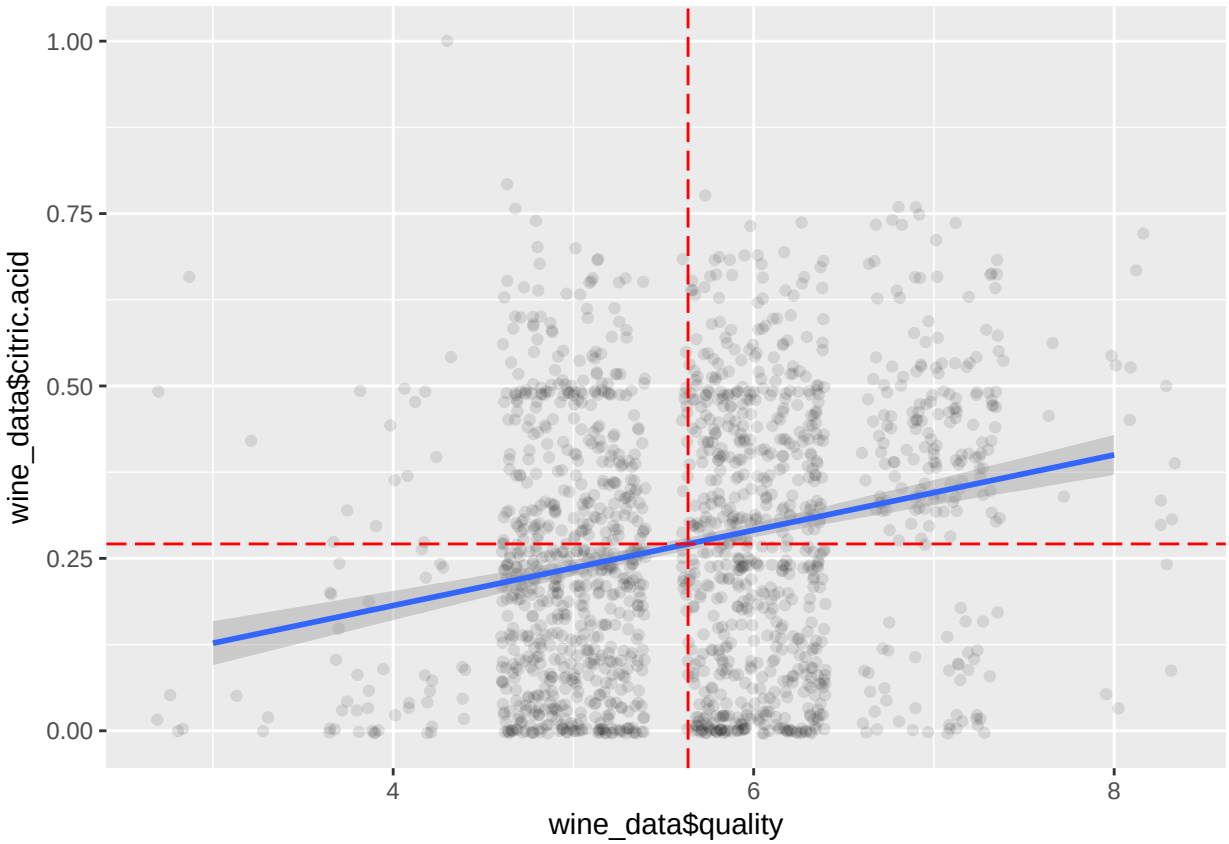
Relationship between suplhates and wine qulaity scores: There seems to a be a positive asscociation between wine quality scores and sulphates content in wine. As suplhates prevents microbial growth and oxidation of wine, it might be a good indication of a high quality wine. The wine quality score increases drastically with just a small increase in sulphates.

Citric acid

```
tapply(wine_data$citric.acid, wine_data$quality, mean)
```

```
##         3         4         5         6         7         8
## 0.1710000 0.1741509 0.2436858 0.2738245 0.3751759 0.3911111
```

```
ggplot(aes(x= wine_data$quality, y=wine_data$citric.acid), data= wine_data)+
  geom_jitter(alpha=1/10)+
  geom_smooth(method='lm', aes(group=1))+
  geom_hline(yintercept = mean(wine_data$citric.acid), color='red', linetype='longdash')+
  geom_vline(xintercept = mean(wine_data$quality), color='red', linetype='longdash')
```

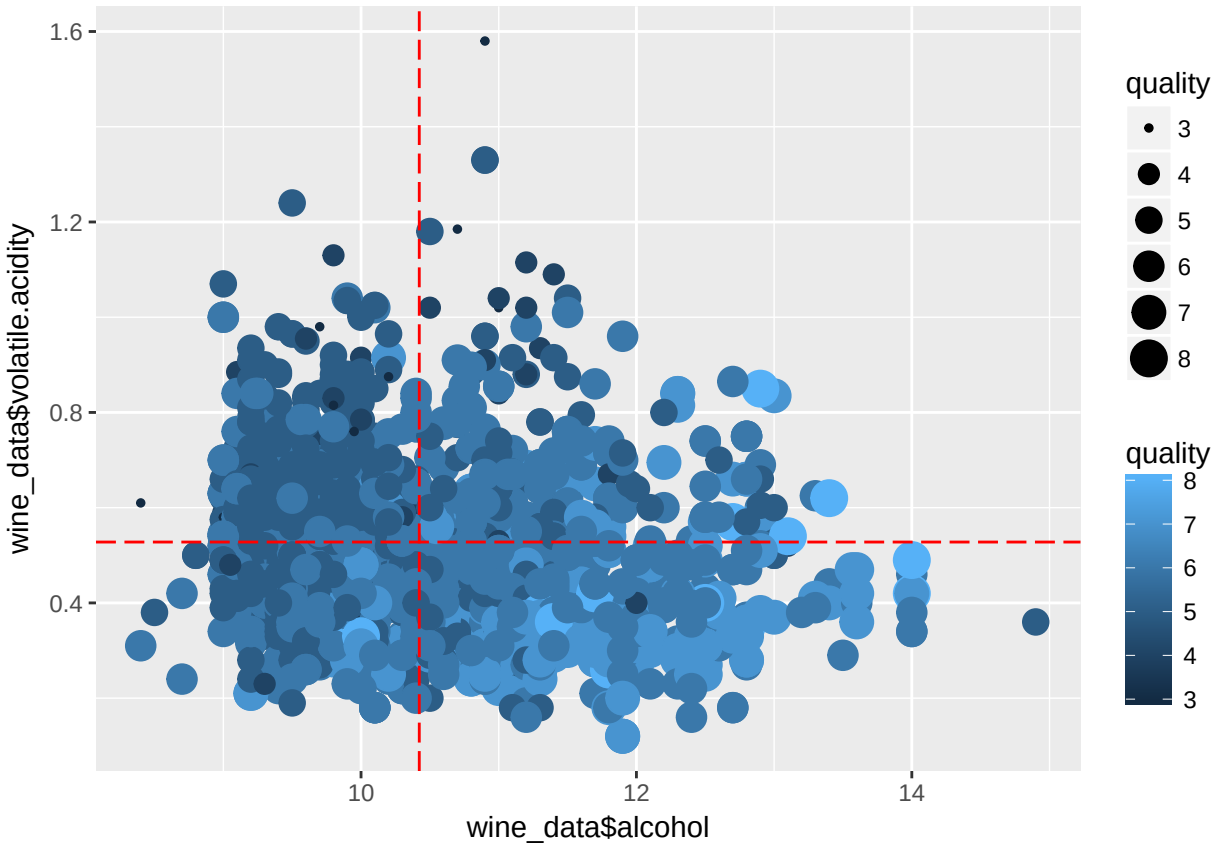


Realtionship between wine qulaity scores and citric acid content: There seems to be a positive relationship between the citric acid content and the wine quality scores. As citric acid brings freshness to the taste of wine, it might be indicative of a high qulaity wine.

Multivarate data analysis:

Volatile acidity vs alcohol

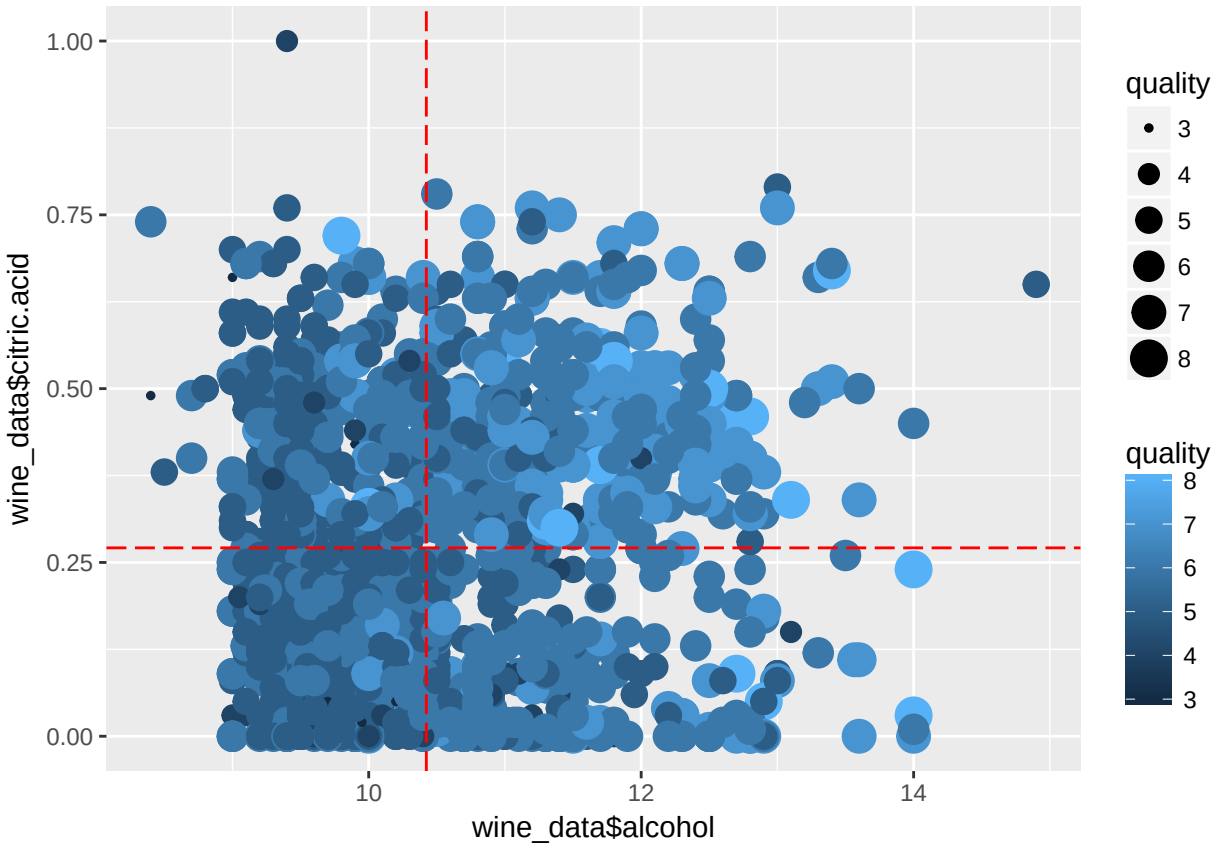
```
ggplot(aes(wine_data$alcohol, wine_data$volatile.acidity), data=wine_data)+
  geom_point(aes(color=quality, size=quality)) +
  geom_hline(yintercept = mean(wine_data$volatile.acidity), color='red', linetype='longdash')+
  geom_vline(xintercept = mean(wine_data$alcohol), color='red', linetype='longdash')
```



From the plot, we can see that wines having higher alcohol content and lower volatile sulphates have higher quality scores.

Citric acid vs alcohol

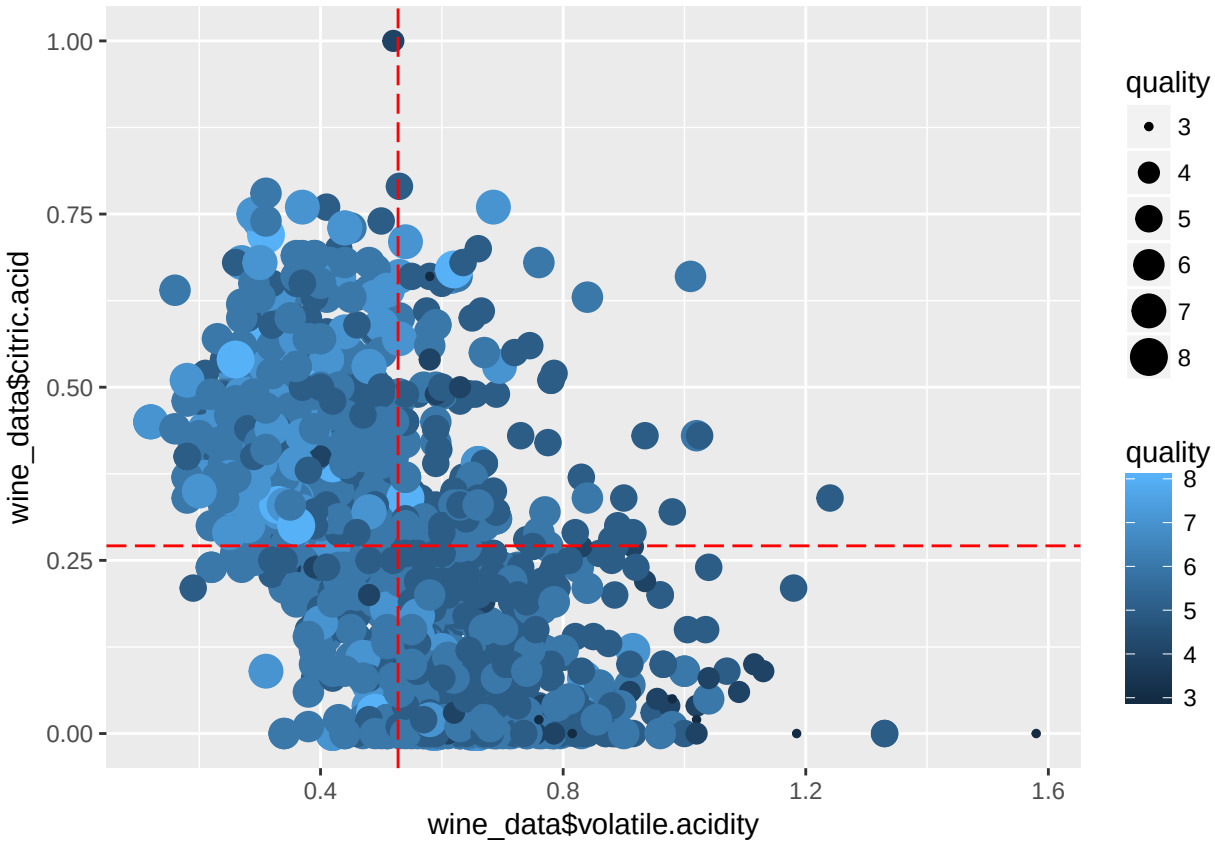
```
ggplot(aes(wine_data$alcohol, wine_data$citric.acid), data=wine_data)+
  geom_point(aes(color=quality, size=quality)) +
  geom_hline(yintercept = mean(wine_data$citric.acid), color='red', linetype='longdash')+
  geom_vline(xintercept = mean(wine_data$alcohol), color='red', linetype='longdash')
```



From the plot, we can see that wines have higher quality scores, when their alcohol and citric content are in the higher ranges.

Citric acid vs volatile acidity

```
ggplot(aes(wine_data$volatile.acidity, wine_data$citric.acid), data=wine_data)+
  geom_point(aes(color=quality, size=quality)) +
  geom_hline(yintercept = mean(wine_data$citric.acid), color='red', linetype='longdash')+
  geom_vline(xintercept = mean(wine_data$volatile.acidity), color='red', linetype='longdash')
```



From the plot we can see that wines having a higher level citric content and lower level volatile acid have higher quality scores.

Note that the `echo = FALSE` parameter was added to the code chunk to prevent printing of the R code that generated the plot.

Building a linear regression model.

```
quality_mdl <- lm(wine_data$quality ~ wine_data$alcohol +wine_data$volatile.acidity + wine_data$sulphat

summary(quality_mdl)
```

```
##
## Call:
## lm(formula = wine_data$quality ~ wine_data$alcohol + wine_data$volatile.acidity +
##     wine_data$sulphates + wine_data$citric.acid, data = wine_data)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.71408 -0.38590 -0.06402  0.46657  2.20393
##
## Coefficients:
##                             Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  2.64592    0.20106  13.160  < 2e-16 ***
## wine_data$alcohol            0.30908    0.01581  19.553  < 2e-16 ***
## wine_data$volatile.acidity -1.26506    0.11266 -11.229  < 2e-16 ***
```

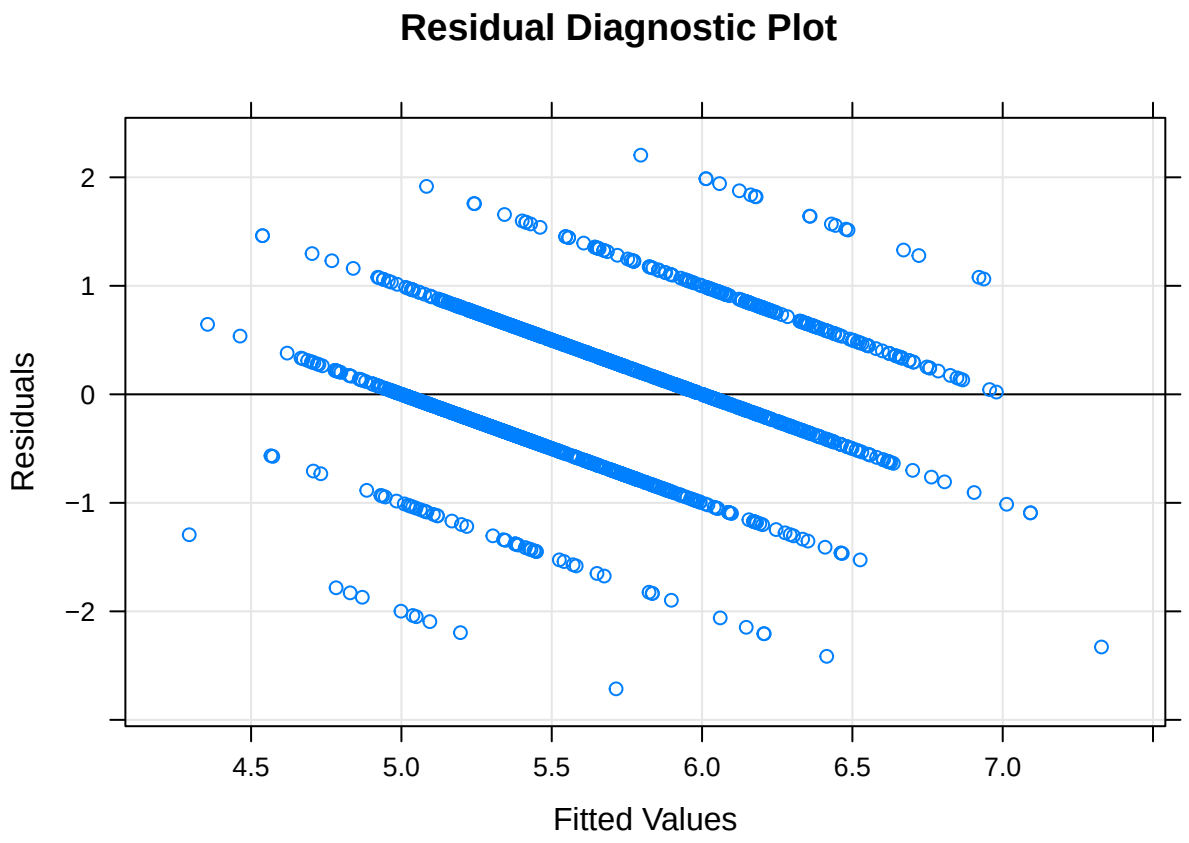
```
## wine_data$sulphates         0.69552    0.10311   6.746 2.12e-11 ***
## wine_data$citric.acid      -0.07913    0.10381  -0.762    0.446
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6588 on 1594 degrees of freedom
## Multiple R-squared:  0.3361, Adjusted R-squared:  0.3345
## F-statistic: 201.8 on 4 and 1594 DF,  p-value: < 2.2e-16
```

Intercept is 1.874 slope is 0.36084 Thus the formula is: quality= 0.36*alcohol + 1.874 Both the Pr values for the intercept and the slope is less than 0.05.The stars at the right of these value indicate the level of significance. p value implies that the observtion is statistically significant.

Testing the Linear regression assumptions with the help of the residual diagnostics

Residual analysis.

```
xyplot(resid(quality_mdl) ~ fitted(quality_mdl),
  xlab = "Fitted Values",
  ylab = "Residuals",
  main = "Residual Diagnostic Plot",
  panel = function(x, y, ...)
  {
    panel.grid(h = -1, v = -1)
    panel.abline(h = 0)
    panel.xyplot(x, y, ...)
  }
)
```




```
?xyplot
```

Prediction

```
#splitting data 75:25

#Computing sample_size of the train dataset
sample_size <- floor(0.7*nrow(wine_data))
#Load the train and test data
set.seed(100)
train_indices <- sample(seq_len(nrow(wine_data)), size = sample_size)


#Load the train and test dataset
train_data <- wine_data[train_indices,]
test_data <- wine_data[-train_indices,]

#Build a prediction model
linear1 <- lm(wine_data$quality ~ wine_data$alcohol+wine_data$sulphates+wine_data$citric.acid+wine_data$
summary(linear1)
```

```
##
## Call:
## lm(formula = wine_data$quality ~ wine_data$alcohol + wine_data$sulphates +
##     wine_data$citric.acid + wine_data$volatile.acidity, data = train_data)
```

```
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.71408 -0.38590 -0.06402  0.46657  2.20393
##
## Coefficients:
##                            Estimate Std. Error t value Pr(>|t|)
## (Intercept)                 2.64592    0.20106  13.160  < 2e-16 ***
## wine_data$alcohol           0.30908    0.01581  19.553  < 2e-16 ***
## wine_data$sulphates         0.69552    0.10311   6.746 2.12e-11 ***
## wine_data$citric.acid      -0.07913    0.10381  -0.762    0.446
## wine_data$volatile.acidity -1.26506    0.11266 -11.229  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.6588 on 1594 degrees of freedom
## Multiple R-squared:  0.3361, Adjusted R-squared:  0.3345
## F-statistic: 201.8 on 4 and 1594 DF,  p-value: < 2.2e-16
```

```
#Testing the prediction model
predicted=predict(linear1,test_data)
```

```
## Warning: 'newdata' had 480 rows but variables found have 1599 rows
```

```
head(predicted)
```

```
##        1        2        3        4        5        6
## 5.055201 5.034583 5.162360 5.679757 5.055201 5.105803
```

```
head(test_data$quality)
```

```
## [1] 5 5 5 5 5 5
```

```
#Calculating multiple R squared for test data
SSE <- sum((test_data$quality-predicted)^2)
```

```
## Warning in test_data$quality - predicted: longer object length is not a
## multiple of shorter object length
```

```
SST <-sum((test_data$quality-mean(test_data$quality))^2)
1-(SSE/SST)
```

```
## [1] -3.538116
```