MULTIPLE LINEAR REGRESSION

1. SELECTION OF DATASET:

The dataset used here is 'Wine Quality Dataset'. The dataset is related to red variants of the Portuguese "Vinho Verde" wine. This dataset is to model wine quality based on physicochemical tests.

Dimension: 1599 rows and 12 columns

Attributes used: fixed acidity, volatile acidity, citric acid, residual sugar, chlorides, free sulfur dioxide, total sulfur dioxide, density, pH, sulphates, alcohol, quality (score between 0 and 10).

```
> redwine<- read.csv("C:/Users/AISHIKA/Desktop/R datasets/redwine.csv",header=TRUE)  #importing the dataset
> head(redwine)          #this command shows the first six rows of the dataset
  fixed.acidity volatile.acidity citric.acid residual.sugar chlorides free.sulfur.dioxide total.sulfur.dioxide density   pH
1           7.4             0.70        0.00            1.9     0.076                  11                   34  0.9978 3.51
2           7.8             0.88        0.00            2.6     0.098                  25                   67  0.9968 3.20
3           7.8             0.76        0.04            2.3     0.092                  15                   54  0.9970 3.26
4          11.2             0.28        0.56            1.9     0.075                  17                   60  0.9980 3.16
5           7.4             0.70        0.00            1.9     0.076                  11                   34  0.9978 3.51
6           7.4             0.66        0.00            1.8     0.075                  13                   40  0.9978 3.51
  sulphates alcohol quality
1      0.56     9.4       5
2      0.68     9.8       5
3      0.65     9.8       5
4      0.58     9.8       6
5      0.56     9.4       5
6      0.56     9.4       5
```

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

```
> str(redwine)          #this command gives the structure of the dataset
'data.frame':   1599 obs. of  12 variables:
 $ fixed.acidity       : num  7.4 7.8 7.8 11.2 7.4 7.4 7.9 7.3 7.8 7.5 ...
 $ volatile.acidity    : num  0.7 0.88 0.76 0.28 0.7 0.66 0.6 0.65 0.58 0.5 ...
 $ citric.acid         : num  0 0 0.04 0.56 0 0 0.06 0 0.02 0.36 ...
 $ residual.sugar      : num  1.9 2.6 2.3 1.9 1.9 1.8 1.6 1.2 2 6.1 ...
 $ chlorides           : num  0.076 0.098 0.092 0.075 0.076 0.075 0.069 0.065 0.073 0.071 ...
 $ free.sulfur.dioxide : num  11 25 15 17 11 13 15 15 9 17 ...
 $ total.sulfur.dioxide: num  34 67 54 60 34 40 59 21 18 102 ...
 $ density             : num  0.998 0.997 0.997 0.998 0.998 ...
 $ pH                  : num  3.51 3.2 3.26 3.16 3.51 3.51 3.3 3.39 3.36 3.35 ...
 $ sulphates           : num  0.56 0.68 0.65 0.58 0.56 0.56 0.46 0.47 0.57 0.8 ...
 $ alcohol             : num  9.4 9.8 9.8 9.8 9.4 9.4 9.4 10 9.5 10.5 ...
 $ quality             : int  5 5 5 6 5 5 5 7 7 5 ...
```

The structure of the data set gives a detailed description of the data types and the levels for the variables which have factors.

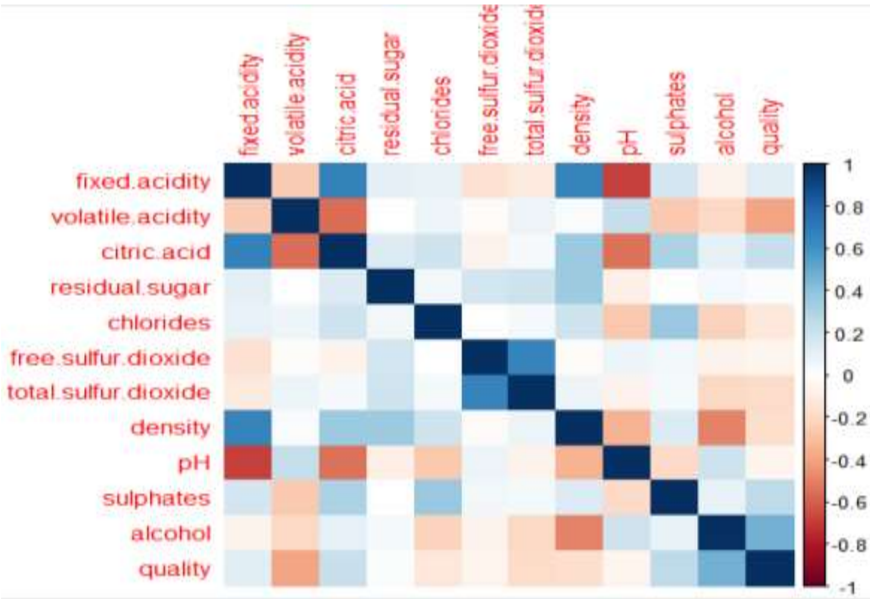
```
> any(is.na(redwine))    #to check whether there are any null values in the dataset
[1] FALSE
```

2. CORRELATION ANALYSIS OF THE DATASET:

```
> #The following libraries are used for Exploratory Data Analysis
> library(ggplot2)
> library(ggthemes)
> library(dplyr)
```

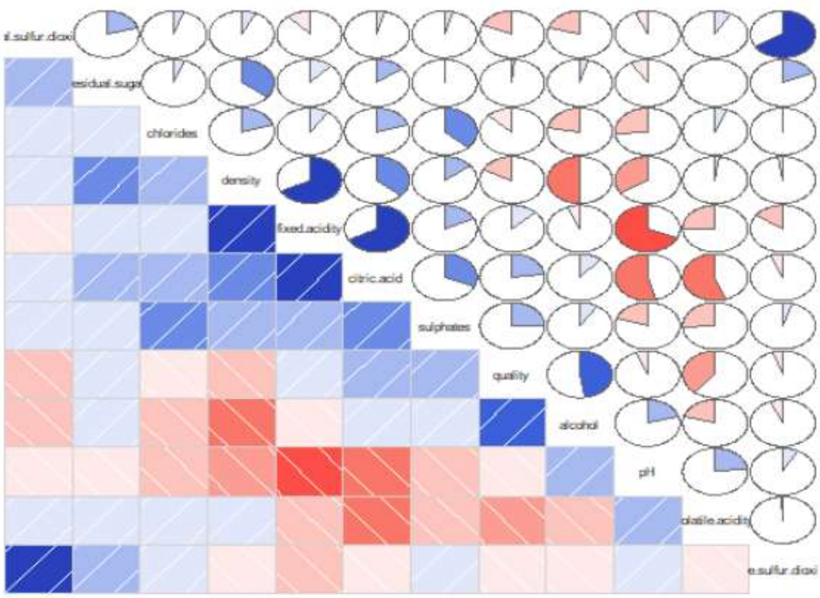
```
> #Grabbing only the numeric columns as we can't see correlation between the categorical variables
> num.cols<- sapply(redwine,is.numeric)
> #Filtering to numeric columns for correlation
> cor.data<-cor(redwine[ , num.cols])
> cor.data
                     fixed.acidity volatile.acidity citric.acid residual.sugar    chlorides free.sulfur.dioxide
fixed.acidity           1.00000000     -0.256130895  0.67170343    0.114776724  0.093705186        -0.153794193
volatile.acidity       -0.25613089      1.000000000 -0.55249568    0.001917882  0.061297772        -0.010503827
citric.acid             0.67170343     -0.552495685  1.00000000    0.143577162  0.203822914        -0.060978129
residual.sugar          0.11477672      0.001917882  0.14357716    1.000000000  0.055609535         0.187048995
chlorides               0.09370519      0.061297772  0.20382291    0.055609535  1.000000000         0.005562147
free.sulfur.dioxide    -0.15379419     -0.010503827 -0.06097813    0.187048995  0.005562147         1.000000000
total.sulfur.dioxide   -0.11318144      0.076470005  0.03553302    0.203027882  0.047400468         0.667666450
density                 0.66804729      0.022026232  0.36494718    0.355283371  0.200632327        -0.021945831
pH                     -0.68297819      0.234937294 -0.54190414   -0.085652422 -0.265026131         0.070377499
sulphates               0.18300566     -0.260986685  0.31277004    0.005527121  0.371260481         0.051657572
alcohol                -0.06166827     -0.202288027  0.10990325    0.042075437 -0.221140545        -0.069408354
quality                 0.12405165     -0.390557780  0.22637251    0.013731637 -0.128906560        -0.050656057
                     total.sulfur.dioxide     density          pH    sulphates     alcohol     quality
fixed.acidity                 -0.11318144  0.66804729 -0.68297819  0.183005664 -0.06166827  0.12405165
volatile.acidity               0.07647000  0.02202623  0.23493729 -0.260986685 -0.20228803 -0.39055778
citric.acid                    0.03553302  0.36494718 -0.54190414  0.312770044  0.10990325  0.22637251
residual.sugar                 0.20302788  0.35528337 -0.08565242  0.005527121  0.04207544  0.01373164
chlorides                      0.04740047  0.20063233 -0.26502613  0.371260481 -0.22114054 -0.12890656
free.sulfur.dioxide            0.66766645 -0.02194583  0.07037750  0.051657572 -0.06940835 -0.05065606
total.sulfur.dioxide           1.00000000  0.07126948 -0.06649456  0.042946836 -0.20565394 -0.18510029
density                        0.07126948  1.00000000 -0.34169933  0.148506412 -0.49617977 -0.17491923
pH                            -0.06649456 -0.34169933  1.00000000 -0.196647602  0.20563251 -0.05773139
sulphates                      0.04294684  0.14850641 -0.19664760  1.000000000  0.09359475  0.25139708
alcohol                       -0.20565394 -0.49617977  0.20563251  0.093594750  1.00000000  0.47616632
quality                       -0.18510029 -0.17491923 -0.05773139  0.251397079  0.47616632  1.00000000
```

```
> #For proper data visualization we use the 'corrgram' package and the 'corrplot' package
> library(corrgram)
> library(corrplot)
> #Now we perform the correlation plot and see what we can infer from that
> corrplot(cor.data, method='color')
```



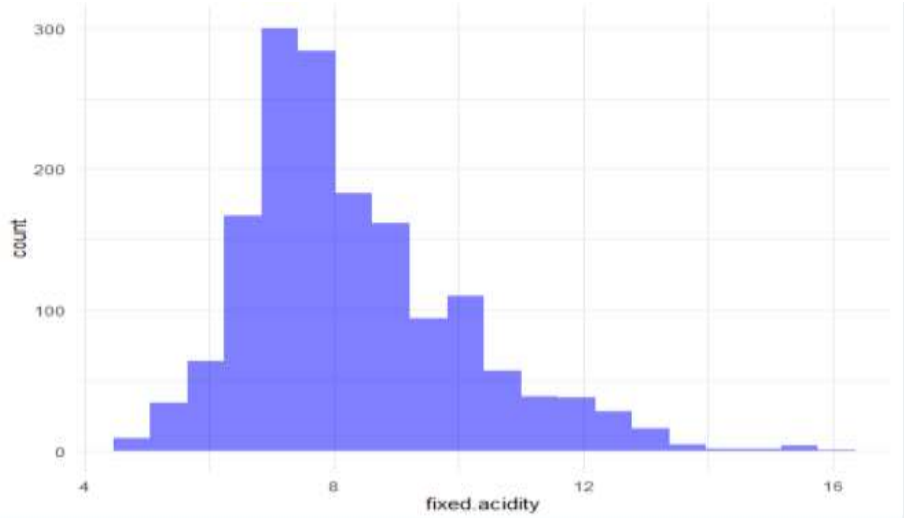
We can see from the above correlation plot that white indicates zero correlation, the shades from white to blue indicates positive correlation and from white to red indicates negative correlation. Positive correlation indicates that as the value of one variable increases, the value of the other variable also increases, i.e., directly proportional, while Negative correlation indicates that as the value of one variable increases the value of the other variable will decrease, i.e., indirectly proportional. We can see that citric.acid and fixed.acidity, density and fixed.acidity are highly correlated to each other. This means that higher the citric.acid, higher is the fixed.acidity and higher the density, higher is the fixed.acidity. We are going to predict the fixed.acidity based on the variables citric.acid and density. Here fixed.acidity is a dependent variable and citric.acid and density are independent variables.

```
> #Now we perform the correlogram and see what we can infer from that
> corrgram(redwine,order=TRUE, lower.panel=panel.shade, upper.panel=panel.pie, text.panel=panel.txt)
>
```



In R, correlograms are implemented using the ‘corrgram’ function. It shows the graph of the correlation matrix and is very useful for highlighting the most correlated values. The results of this plot can be interpreted in the same way as ‘corrplot’. Here blue is positive correlation and pink is negative correlation.

```
> #Histogram of the variable fixed.acidity
> ggplot(redwine,aes(x=fixed.acidity))+ geom_histogram(bins=20, alpha=0.5, fill='blue') + theme_minimal()
```



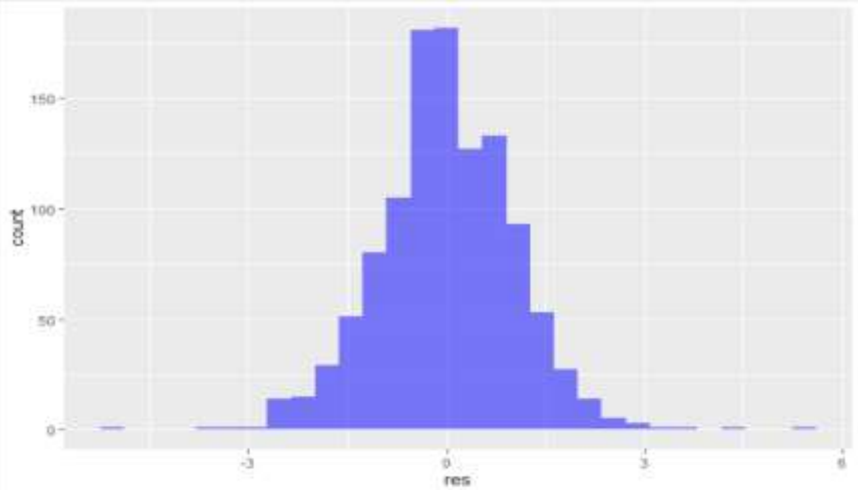
As we have to predict fixed.acidity so we draw a histogram of it. From the graph we can see that the higher amount of fixed.acidity is below the mean of fixed.acidity and there are much more values or observations beyond the mean value. So we can say that this graph is positively skewed.

3. MULTIPLE LINEAR REGRESSION:

```
> #We need to split our data into a training set and a testing set in order to test our accuracy, so we can do this using the caTools library
> library(caTools)
> #Spliting up the sample for training and testing and assigns boolean values to a new colum
> sample <- sample.split(redwine$fixed.acidity, SplitRatio = 0.70)
> #Training data
> train_data = subset(redwine, sample == TRUE)
> #Testing data
> test_data= subset(redwine, sample == FALSE)
> #Training the model
> model <- lm(fixed.acidity ~ citric.acid+density, train_data)
> summary(model)

Call:
lm(formula = fixed.acidity ~ citric.acid + density, data = train_data)

Residuals:
    Min      1Q  Median      3Q     Max 
-4.9697 -0.5808 -0.0218  0.6785  5.5298 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) -436.6703    17.4941  -24.96   <2e-16 ***
citric.acid    4.3776     0.1712   25.58   <2e-16 ***
density      445.2637    17.5692   25.34   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 1.026 on 1117 degrees of freedom
Multiple R-squared:  0.656,	Adjusted R-squared:  0.6554 
F-statistic:  1065 on 2 and 1117 DF,  p-value: < 2.2e-16
```

The residuals are the difference between the actual values of the variable we are predicting and predicted values from our regression.
The stars are for significance levels, with the number of asterisks displayed according to the p-value computed. *** indicates high significance. In this case, *** indicates that there is high significance between citric.acid and density with fixed.acidity.
The estimated coefficient is the value of slope calculated by the regression.
Standard Error of the Coefficient Estimate is the measure of the variability in the estimate for the coefficient.
t-value shows how many standard deviations the coefficient is far away from zero. Further it is away from zero, stronger the relationship between the variables.
p-value shows whether the overall model is significant or not. Coefficients having p-value less than the level of significance are said to be statistically significant. In this case we can say that the variables citric.acid and density are statistically significant for the prediction of fixed.acidity.
R-squared is an overall measure of the strength of association and adjusted R-squared gives a more proper estimate of the R-squared value for the population. Its value shows that 65.54 % of the variance in fixed.acidity can be predicted from the variables citric.acid and density.

```
> #Visualizing the model
> res<- residuals(model)      #Grabbing the residuals
> res<- as.data.frame(res)    #Converting the residuals to dataframe for ggplot
> head(res)
         res
2  0.6314392
3  0.3672841
4  1.0456898
5 -0.2138245
6 -0.2138245
8  1.1110194
> #Histogram of residuals
> ggplot(res, aes(res)) + geom_histogram(fill='blue', alpha=0.5)
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



We have to remove the negative values from here.

```
> #Predictions
> fixed.acidity.predictions <- predict(model,test_data)
> results<- cbind(fixed.acidity.predictions,test_data$fixed.acidity)
> colnames(results) <- c('predicted','real')
> results<- as.data.frame(results)
> results
```

	predicted	real
1	7.613825	7.4
7	7.253109	7.9
12	9.189746	7.5
13	6.055401	5.6
14	8.705211	7.8
16	8.801772	8.9
18	8.394277	8.1
20	9.445642	7.9
26	7.290127	6.3
45	6.988954	6.8
47	9.139964	7.7
54	8.833535	8.6
67	7.650092	7.5
70	7.120280	8.0
73	7.819939	7.7
75	10.244865	9.7
77	10.771673	8.8
78	7.079508	6.8
82	10.500010	7.8
83	9.315066	7.4
94	8.217673	7.7
97	7.034982	6.8
100	8.134627	8.1
102	8.081091	7.8
103	8.134627	8.1
111	7.822191	7.8
112	7.384436	8.4
113	7.428211	8.4
119	7.076505	8.8
120	7.118028	7.0
126	8.056085	9.0
129	7.601812	8.0
132	6.182223	5.6
133	6.182223	5.6
140	7.733139	7.8
144	6.895396	6.3
145	4.853189	5.2
146	9.576218	8.1
153	6.498413	7.5
156	9.229767	7.1
159	7.213087	7.1
161	6.899150	7.6
164	8.930095	7.4
165	8.885569	7.3
168	6.765571	7.3
171	7.477993	7.9
179	7.213087	7.0
184	8.177652	6.8
186	9.797348	8.9
190	8.569380	7.9
192	8.530109	6.4
194	7.909743	7.6
195	7.909743	7.6
197	8.481828	7.3
205	8.455061	7.0
211	9.372096	9.7
212	8.241438	8.0
215	7.785924	7.8
217	7.601812	8.7
225	9.033903	8.4
227	9.179235	8.9
228	8.493841	9.0
230	8.285214	6.9
233	8.578389	8.5
245	10.541794	15.0
251	9.851635	10.8
255	7.346666	7.1
259	10.495506	7.7
260	10.495015	10.0
261	7.930504	7.9
263	6.832361	8.0
266	9.444891	11.8
269	8.189664	6.9
272	10.647854	11.5
273	10.842967	10.9
277	8.189664	6.9

278	10.647854	11.5
290	11.358775	11.6
291	7.918752	8.7
293	9.333085	10.4
302	9.492421	11.1
308	10.164821	10.3
309	10.252372	10.3
318	8.517606	9.6
322	9.620254	9.3
326	10.136812	10.0
328	9.891657	10.3
331	10.504515	10.2
332	10.504515	10.2
337	8.247935	8.9
340	10.823707	12.5
342	9.447895	10.6
350	8.326246	9.1
353	7.435719	7.7
359	11.482594	11.9
360	11.927857	12.6
363	11.126383	11.9
367	12.462174	12.8
371	7.567797	6.9
372	8.875058	7.9
378	9.221509	9.4
382	12.193514	13.7
384	9.274294	8.3
387	8.885569	7.8
388	8.003300	8.3
393	9.979208	9.5
398	10.641848	11.5
401	8.413798	6.6
405	7.654597	7.7
408	11.304488	12.0
409	9.796598	10.4
412	9.546707	9.1
413	7.779917	7.1
415	9.280300	8.8
420	7.524772	6.8
423	7.465230	7.7
424	9.671277	10.5
428	9.022151	9.5
433	9.743813	11.9
438	11.165653	11.1
439	10.822957	10.4
440	7.199573	7.0
443	13.345193	15.6
447	11.041835	12.5
451	11.346762	11.9
462	8.754993	8.3
463	10.234354	11.0
464	10.410957	8.1
469	11.613920	11.4
470	6.543690	8.7
474	9.709797	9.9
476	9.109702	9.6
478	9.714302	10.4
480	8.362514	9.4
483	10.552795	10.6
491	9.062924	9.3
493	8.599641	8.9
495	7.462977	6.5
496	9.666772	10.7
498	8.480327	7.2
499	9.666772	10.7
501	8.975372	7.8
506	9.926423	10.2
508	11.001062	11.2
510	11.253205	13.3
514	10.192830	10.5
518	10.471250	10.4
521	9.936934	9.8
522	9.847881	7.6
523	9.803355	8.2
525	9.669776	9.2
527	9.224512	7.3
531	9.198755	9.1

535	9.644019	10.0
536	9.198755	9.1
538	8.397280	8.1
542	10.204092	9.5
546	9.936934	9.1
548	10.159566	10.6
549	10.471250	12.4
551	6.767073	6.8
552	8.753491	9.4
555	12.140989	15.5
559	10.827461	10.9
561	10.471250	12.7
563	9.936934	9.0
564	9.447144	7.6
565	11.673462	13.0
567	9.243282	8.7
577	8.664439	9.9
580	9.669776	10.6
582	10.827461	12.3
583	10.738409	11.7
586	8.753491	7.6
587	10.115039	11.1
589	5.948330	5.0
593	10.159566	9.0
596	10.299151	8.9
602	11.136894	13.2
605	8.450065	8.3
608	9.586729	8.8
611	10.200338	8.8
617	10.648605	9.6
618	10.024485	11.5
621	8.575386	8.3
622	8.531610	8.2
625	8.459826	6.8
631	8.662937	8.7
641	10.162569	9.9
646	8.318739	7.8
647	7.565544	7.3
649	8.303723	8.7
651	9.677283	10.7
652	10.133058	9.8
656	9.515695	9.7
660	7.345165	7.1
661	7.697622	7.2
664	9.449396	10.1
669	8.961858	10.0
671	7.818438	6.9
676	9.588231	9.3
678	8.139882	8.6
683	9.059921	8.5
691	7.079508	7.4
692	9.554966	9.2
693	9.890906	8.6
694	9.281802	9.0
704	9.447895	7.5
715	8.928594	9.9
722	8.842544	8.9
725	7.161053	7.5
729	7.345165	6.4
731	10.681119	9.5
738	8.682457	8.2
739	8.843295	9.0
742	8.700060	9.2
750	8.348353	7.3
753	8.205170	7.6
754	8.153991	8.3
755	10.038438	7.8
761	8.703762	9.0
762	8.564979	9.3
767	8.214774	8.8
772	8.751990	9.4
774	8.660685	7.9
782	8.012956	6.5
786	9.586729	9.9
787	9.586729	9.9
789	9.243282	10.0
796	8.953808	10.8

801	7.345113	7.2
802	7.807436	8.6
804	7.652345	7.7
806	7.476930	8.2
812	11.054494	12.9
818	8.406029	10.8
819	6.954834	7.1
821	6.839066	7.0
823	7.960275	6.7
824	7.960275	6.7
828	7.532071	7.1
835	8.369063	8.8
836	7.495001	7.6
845	8.781501	9.9
846	8.127922	6.4
848	8.269708	7.4
854	8.751135	9.3
857	8.751135	9.3
861	7.725088	7.2
863	8.337842	7.5
864	7.725088	7.2
868	7.097861	6.9
872	6.596371	6.9
875	9.110996	10.4
877	6.442781	7.1
880	7.633085	7.2
883	7.796873	8.4
894	7.580404	7.2
895	7.589309	7.2
897	7.824986	8.3
903	7.855664	7.4
905	7.164056	6.8
907	8.172396	7.2
915	7.321838	7.3
916	8.171541	8.6
917	6.731295	5.3
919	8.391274	8.4
921	9.260237	9.6
922	8.391274	8.4
923	6.910204	8.4
924	9.166784	6.8
927	9.085135	9.4
932	7.515116	7.4
934	7.515116	7.4
935	7.506210	6.6
936	8.222022	8.8
938	9.851583	12.0
939	7.389379	7.2
941	8.683490	9.6
943	9.605290	10.1
948	8.058672	8.3
952	8.058672	8.3
966	8.358552	8.3
970	7.513562	6.7
973	9.192593	10.1
981	8.900376	9.1
985	9.358091	12.2
989	7.319846	7.7
992	8.456614	7.1
993	7.045284	6.5
996	7.506909	7.7
999	8.095898	8.9
1003	7.882923	9.1
1004	6.788325	6.8
1005	8.095200	8.2
1007	7.882923	9.1
1009	8.336289	8.9
1013	7.486794	7.7
1016	8.817122	10.5
1019	5.791633	8.0
1021	9.888549	11.3
1023	7.135088	7.0
1026	7.800835	8.6
1029	7.669300	7.2
1030	6.713641	7.7
1032	6.473199	7.3
1034	7.126933	7.5

1035	8.219227	8.9
1048	7.707924	7.1
1053	5.508426	5.6
1058	8.726025	7.6
1064	9.501921	10.2
1067	6.085016	6.6
1070	8.789759	8.0
1071	8.328082	9.3
1075	8.988032	7.5
1076	9.053216	9.1
1082	8.524541	7.9
1085	8.504092	7.2
1086	6.846417	6.8
1087	9.177837	8.5
1089	9.777338	11.6
1090	9.777338	11.6
1093	6.667561	7.0
1094	8.455061	9.2
1095	7.451225	6.6
1097	7.451225	6.6
1098	8.769696	8.6
1102	7.050332	7.4
1103	6.324708	6.1
1105	7.232087	8.0
1113	7.698811	7.9
1115	6.418600	5.0
1117	6.994105	7.0
1121	6.675507	7.9
1122	5.708096	6.6
1127	5.946828	5.8
1128	5.828317	6.3
1129	9.325577	10.0
1132	6.578404	5.9
1133	7.098508	7.4
1135	8.006095	8.5
1136	8.071331	8.0
1140	8.593196	8.2
1142	8.320032	8.2
1143	6.198532	6.9
1145	7.765757	7.3
1148	9.236420	10.0
1155	6.563002	6.6
1158	5.997361	5.1
1159	8.570026	6.7
1160	9.264638	10.2
1162	8.476521	8.8
1166	9.152519	8.5
1168	8.199008	8.2
1171	8.599047	9.2
1178	5.556706	7.1
1180	8.252492	8.2
1182	8.151478	9.8
1188	6.994648	7.0
1190	7.102470	8.8
1194	6.594171	6.4
1196	7.922350	6.2
1204	8.881919	9.9
1209	8.532153	7.2
1210	7.928846	6.2
1211	6.445732	6.8
1215	8.950701	10.2
1217	8.512246	7.9
1218	7.577140	8.2
1224	9.039003	10.5
1227	6.908055	7.5
1228	8.659235	9.0
1230	8.607253	7.6
1232	6.949631	7.8
1233	8.607253	7.6
1235	5.262081	7.1
1236	8.088495	6.0
1241	7.215882	7.5
1244	8.799519	8.3
1247	6.682420	7.4
1248	7.888981	7.4
1252	7.558787	7.5
1255	7.199677	7.8

1256	5.822363	6.1
1262	6.086570	6.3
1263	9.036854	9.8
1267	6.870933	7.2
1277	8.411984	8.5
1278	6.972645	7.0
1279	8.167245	8.0
1280	7.985281	9.8
1282	7.509756	7.1
1283	6.896950	7.9
1286	9.597783	11.3
1287	7.527358	7.1
1292	7.419306	8.2
1294	7.132940	7.5
1295	7.419306	8.2
1299	4.991221	5.7
1301	6.117738	5.2
1305	7.509005	7.6
1311	8.003947	7.5
1312	6.088666	6.5
1314	7.544626	7.0
1317	5.289548	5.4
1325	7.328647	6.7
1328	7.328647	6.7
1331	8.021757	7.4
1338	6.135549	6.0
1342	6.623837	7.5
1349	6.885792	7.2
1350	6.447234	6.9
1355	6.801192	7.2
1357	7.301181	6.1
1358	8.076691	7.3
1363	9.026448	11.6
1365	6.454637	7.2
1366	7.696120	7.8
1369	8.462516	8.0
1373	9.147315	8.7
1374	8.279259	7.7
1383	7.833297	8.0
1385	7.437815	7.1
1389	8.694804	6.6
1392	7.828844	8.0
1396	7.895738	8.6
1399	7.604815	7.0
1402	7.820690	7.9
1409	7.943007	8.1
1413	8.924089	8.2
1420	7.553532	7.7
1426	9.055416	8.3
1429	6.901403	7.8
1432	8.080340	7.6
1433	6.099928	5.9
1450	7.599455	7.2
1451	7.634326	7.2
1454	8.728924	7.6
1457	5.454243	6.0
1461	7.758457	6.4
1462	7.097319	6.2
1469	8.783106	7.3
1470	7.498755	7.3
1480	10.454086	8.2
1484	8.480223	8.1
1486	7.539475	7.0
1494	8.110810	7.7
1502	8.268853	7.8
1505	9.329822	7.5
1512	7.486148	7.1
1513	6.934020	6.4
1514	6.106477	6.4
1515	8.809175	6.9
1517	8.053677	6.1
1523	8.053677	6.1
1524	7.595055	6.8
1530	7.198176	7.1
1531	6.511719	6.6
1534	8.646576	8.0
1535	6.521323	6.6

```
1539   6.930267  6.2
1545   8.472068  8.4
1546   7.949712  6.5
1554   7.547035  7.3
1556   7.173608  7.0
1557   7.147747  6.6
1559   9.089587  6.9
1560   8.048473  7.8
1566   6.405658  6.7
1567   9.142008  6.7
1569   7.510559  7.0
1576   8.002341  7.5
1578   7.566942  6.2
1579   7.201825  6.8
1582   6.324708  6.2
1586   8.266496  7.2
1598   7.101667  5.9
1599   8.642718  6.0
```

```
> #To remove negative predictions and replace it with 0
> to_zero <- function(x){
+   if(x <0) {
+     return(0)
+   }else{
+     return(x)
+   }
+ }
> results$predicted <- sapply(results$predicted,to_zero)
> results
      predicted real
1      7.613825  7.4
7      7.253109  7.9
12     9.189746  7.5
13     6.055401  5.6
14     8.705211  7.8
16     8.801772  8.9
18     8.394277  8.1
20     9.445642  7.9
26     7.290127  6.3
45     6.988954  6.8
47     9.139964  7.7
54     8.833535  8.6
67     7.650092  7.5
70     7.120280  8.0
73     7.819939  7.7
75    10.244865  9.7
77    10.771673  8.8
78     7.079508  6.8
82    10.500010  7.8
83     9.315066  7.4
94     8.217673  7.7
97     7.034982  6.8
100    8.134627  8.1
102    8.081091  7.8
103    8.134627  8.1
111    7.822191  7.8
112    7.384436  8.4
113    7.428211  8.4
119    7.076505  8.8
120    7.118028  7.0
126    8.056085  9.0
129    7.601812  8.0
132    6.182223  5.6
133    6.182223  5.6
140    7.733139  7.8
144    6.895396  6.3
145    4.853189  5.2
146    9.576218  8.1
153    6.498413  7.5
156    9.229767  7.1
159    7.213087  7.1
161    6.899150  7.6
164    8.930095  7.4
165    8.885569  7.3
168    6.765571  7.3
171    7.477993  7.9
179    7.213087  7.0
184    8.177652  6.8
```

186	9.797348	8.9
190	8.569380	7.9
192	8.530109	6.4
194	7.909743	7.6
195	7.909743	7.6
197	8.481828	7.3
205	8.455061	7.0
211	9.372096	9.7
212	8.241438	8.0
215	7.785924	7.8
217	7.601812	8.7
225	9.033903	8.4
227	9.179235	8.9
228	8.493841	9.0
230	8.285214	6.9
233	8.578389	8.5
245	10.541794	15.0
251	9.851635	10.8
255	7.346666	7.1
259	10.495506	7.7
260	10.495015	10.0
261	7.930504	7.9
263	6.832361	8.0
266	9.444891	11.8
269	8.189664	6.9
272	10.647854	11.5
273	10.842967	10.9
277	8.189664	6.9
278	10.647854	11.5
290	11.358775	11.6
291	7.918752	8.7
293	9.333085	10.4
302	9.492421	11.1
308	10.164821	10.3
309	10.252372	10.3
318	8.517606	9.6
322	9.620254	9.3
326	10.136812	10.0
328	9.891657	10.3
331	10.504515	10.2
332	10.504515	10.2
337	8.247935	8.9
340	10.823707	12.5
342	9.447895	10.6
350	8.326246	9.1
353	7.435719	7.7
359	11.482594	11.9
360	11.927857	12.6
363	11.126383	11.9
367	12.462174	12.8
371	7.567797	6.9
372	8.875058	7.9
378	9.221509	9.4
382	12.193514	13.7
384	9.274294	8.3
387	8.885569	7.8
388	8.003300	8.3
393	9.979208	9.5
398	10.641848	11.5
401	8.413798	6.6
405	7.654597	7.7
408	11.304488	12.0
409	9.796598	10.4
412	9.546707	9.1
413	7.779917	7.1
415	9.280300	8.8
420	7.524772	6.8
423	7.465230	7.7
424	9.671277	10.5
428	9.022151	9.5
433	9.743813	11.9
438	11.165653	11.1
439	10.822957	10.4
440	7.199573	7.0
443	13.345193	15.6
447	11.041835	12.5
451	11.346762	11.9

462	8.754993	8.3
463	10.234354	11.0
464	10.410957	8.1
469	11.613920	11.4
470	6.543690	8.7
474	9.709797	9.9
476	9.109702	9.6
478	9.714302	10.4
480	8.362514	9.4
483	10.552795	10.6
491	9.062924	9.3
493	8.599641	8.9
495	7.462977	6.5
496	9.666772	10.7
498	8.480327	7.2
499	9.666772	10.7
501	8.975372	7.8
506	9.926423	10.2
508	11.001062	11.2
510	11.253205	13.3
514	10.192830	10.5
518	10.471250	10.4
521	9.936934	9.8
522	9.847881	7.6
523	9.803355	8.2
525	9.669776	9.2
527	9.224512	7.3
531	9.198755	9.1
535	9.644019	10.0
536	9.198755	9.1
538	8.397280	8.1
542	10.204092	9.5
546	9.936934	9.1
548	10.159566	10.6
549	10.471250	12.4
551	6.767073	6.8
552	8.753491	9.4
555	12.140989	15.5
559	10.827461	10.9
561	10.471250	12.7
563	9.936934	9.0
564	9.447144	7.6
565	11.673462	13.0
567	9.243282	8.7
577	8.664439	9.9
580	9.669776	10.6
582	10.827461	12.3
583	10.738409	11.7
586	8.753491	7.6
587	10.115039	11.1
589	5.948330	5.0
593	10.159566	9.0
596	10.299151	8.9
602	11.136894	13.2
605	8.450065	8.3
608	9.586729	8.8
611	10.200338	8.8
617	10.648605	9.6
618	10.024485	11.5
621	8.575386	8.3
622	8.531610	8.2
625	8.459826	6.8
631	8.662937	8.7
641	10.162569	9.9
646	8.318739	7.8
647	7.565544	7.3
649	8.303723	8.7
651	9.677283	10.7
652	10.133058	9.8
656	9.515695	9.7
660	7.345165	7.1
661	7.697622	7.2
664	9.449396	10.1
669	8.961858	10.0
671	7.818438	6.9
676	9.588231	9.3
678	8.139882	8.6

683	9.059921	8.5
691	7.079508	7.4
692	9.554966	9.2
693	9.890906	8.6
694	9.281802	9.0
704	9.447895	7.5
715	8.928594	9.9
722	8.842544	8.9
725	7.161053	7.5
729	7.345165	6.4
731	10.681119	9.5
738	8.682457	8.2
739	8.843295	9.0
742	8.700060	9.2
750	8.348353	7.3
753	8.205170	7.6
754	8.153991	8.3
755	10.038438	7.8
761	8.703762	9.0
762	8.564979	9.3
767	8.214774	8.8
772	8.751990	9.4
774	8.660685	7.9
782	8.012956	6.5
786	9.586729	9.9
787	9.586729	9.9
789	9.243282	10.0
796	8.953808	10.8
801	7.345113	7.2
802	7.807436	8.6
804	7.652345	7.7
806	7.476930	8.2
812	11.054494	12.9
818	8.406029	10.8
819	6.954834	7.1
821	6.839066	7.0
823	7.960275	6.7
824	7.960275	6.7
828	7.532071	7.1
835	8.369063	8.8
836	7.495001	7.6
845	8.781501	9.9
846	8.127922	6.4
848	8.269708	7.4
854	8.751135	9.3
857	8.751135	9.3
861	7.725088	7.2
863	8.337842	7.5
864	7.725088	7.2
868	7.097861	6.9
872	6.596371	6.9
875	9.110996	10.4
877	6.442781	7.1
880	7.633085	7.2
883	7.796873	8.4
894	7.580404	7.2
895	7.589309	7.2
897	7.824986	8.3
903	7.855664	7.4
905	7.164056	6.8
907	8.172396	7.2
915	7.321838	7.3
916	8.171541	8.6
917	6.731295	5.3
919	8.391274	8.4
921	9.260237	9.6
922	8.391274	8.4
923	6.910204	8.4
924	9.166784	6.8
927	9.085135	9.4
932	7.515116	7.4
934	7.515116	7.4
935	7.506210	6.6
936	8.222022	8.8
938	9.851583	12.0
939	7.389379	7.2
941	8.683490	9.6

943	9.605290	10.1
948	8.058672	8.3
952	8.058672	8.3
966	8.358552	8.3
970	7.513562	6.7
973	9.192593	10.1
981	8.900376	9.1
985	9.358091	12.2
989	7.319846	7.7
992	8.456614	7.1
993	7.045284	6.5
996	7.506909	7.7
999	8.095898	8.9
1003	7.882923	9.1
1004	6.788325	6.8
1005	8.095200	8.2
1007	7.882923	9.1
1009	8.336289	8.9
1013	7.486794	7.7
1016	8.817122	10.5
1019	5.791633	8.0
1021	9.888549	11.3
1023	7.135088	7.0
1026	7.800835	8.6
1029	7.669300	7.2
1030	6.713641	7.7
1032	6.473199	7.3
1034	7.126933	7.5
1035	8.219227	8.9
1048	7.707924	7.1
1053	5.508426	5.6
1058	8.726025	7.6
1064	9.501921	10.2
1067	6.085016	6.6
1070	8.789759	8.0
1071	8.328082	9.3
1075	8.988032	7.5
1076	9.053216	9.1
1082	8.524541	7.9
1085	8.504092	7.2
1086	6.846417	6.8
1087	9.177837	8.5
1089	9.777338	11.6
1090	9.777338	11.6
1093	6.667561	7.0
1094	8.455061	9.2
1095	7.451225	6.6
1097	7.451225	6.6
1098	8.769696	8.6
1102	7.050332	7.4
1103	6.324708	6.1
1105	7.232087	8.0
1113	7.698811	7.9
1115	6.418600	5.0
1117	6.994105	7.0
1121	6.675507	7.9
1122	5.708096	6.6
1127	5.946828	5.8
1128	5.828317	6.3
1129	9.325577	10.0
1132	6.578404	5.9
1133	7.098508	7.4
1135	8.006095	8.5
1136	8.071331	8.0
1140	8.593196	8.2
1142	8.320032	8.2
1143	6.198532	6.9
1145	7.765757	7.3
1148	9.236420	10.0
1155	6.563002	6.6
1158	5.997361	5.1
1159	8.570026	6.7
1160	9.264638	10.2
1162	8.476521	8.8
1166	9.152519	8.5
1168	8.199008	8.2
1171	8.599047	9.2

1178	5.556706	7.1
1180	8.252492	8.2
1182	8.151478	9.8
1188	6.994648	7.0
1190	7.102470	8.8
1194	6.594171	6.4
1196	7.922350	6.2
1204	8.881919	9.9
1209	8.532153	7.2
1210	7.928846	6.2
1211	6.445732	6.8
1215	8.950701	10.2
1217	8.512246	7.9
1218	7.577140	8.2
1224	9.039003	10.5
1227	6.908055	7.5
1228	8.659235	9.0
1230	8.607253	7.6
1232	6.949631	7.8
1233	8.607253	7.6
1235	5.262081	7.1
1236	8.088495	6.0
1241	7.215882	7.5
1244	8.799519	8.3
1247	6.682420	7.4
1248	7.888981	7.4
1252	7.558787	7.5
1255	7.199677	7.8
1256	5.822363	6.1
1262	6.086570	6.3
1263	9.036854	9.8
1267	6.870933	7.2
1277	8.411984	8.5
1278	6.972645	7.0
1279	8.167245	8.0
1280	7.985281	9.8
1282	7.509756	7.1
1283	6.896950	7.9
1286	9.597783	11.3
1287	7.527358	7.1
1292	7.419306	8.2
1294	7.132940	7.5
1295	7.419306	8.2
1299	4.991221	5.7
1301	6.117738	5.2
1305	7.509005	7.6
1311	8.003947	7.5
1312	6.088666	6.5
1314	7.544626	7.0
1317	5.289548	5.4
1325	7.328647	6.7
1328	7.328647	6.7
1331	8.021757	7.4
1338	6.135549	6.0
1342	6.623837	7.5
1349	6.885792	7.2
1350	6.447234	6.9
1355	6.801192	7.2
1357	7.301181	6.1
1358	8.076691	7.3
1363	9.026448	11.6
1365	6.454637	7.2
1366	7.696120	7.8
1369	8.462516	8.0
1373	9.147315	8.7
1374	8.279259	7.7
1383	7.833297	8.0
1385	7.437815	7.1
1389	8.694804	6.6
1392	7.828844	8.0
1396	7.895738	8.6
1399	7.604815	7.0
1402	7.820690	7.9
1409	7.943007	8.1
1413	8.924089	8.2
1420	7.553532	7.7
1426	9.055416	8.3

```
1429   6.901403  7.8
1432   8.080340  7.6
1433   6.099928  5.9
1450   7.599455  7.2
1451   7.634326  7.2
1454   8.728924  7.6
1457   5.454243  6.0
1461   7.758457  6.4
1462   7.097319  6.2
1469   8.783106  7.3
1470   7.498755  7.3
1480  10.454086  8.2
1484   8.480223  8.1
1486   7.539475  7.0
1494   8.110810  7.7
1502   8.268853  7.8
1505   9.329822  7.5
1512   7.486148  7.1
1513   6.934020  6.4
1514   6.106477  6.4
1515   8.809175  6.9
1517   8.053677  6.1
1523   8.053677  6.1
1524   7.595055  6.8
1530   7.198176  7.1
1531   6.511719  6.6
1534   8.646576  8.0
1535   6.521323  6.6
1539   6.930267  6.2
1545   8.472068  8.4
1546   7.949712  6.5
1554   7.547035  7.3
1556   7.173608  7.0
1557   7.147747  6.6
1559   9.089587  6.9
1560   8.048473  7.8
1566   6.405658  6.7
1567   9.142008  6.7
1569   7.510559  7.0
1576   8.002341  7.5
1578   7.566942  6.2
1579   7.201825  6.8
1582   6.324708  6.2
1586   8.266496  7.2
1598   7.101667  5.9
1599   8.642718  6.0
```

```
> #Evaluating the prediction values by the method of MSE(Mean Squared Error)
> mse <- mean((results$real - results$predicted)^2)
> print(mse)
[1] 1.008759
> #Evaluating the prediction values by the method of RMSE(Root Mean Squared Error)
> mse^0.5
[1] 1.00437
> #Or we can just use the R-Squared Value for the model which gives the accuracy of the model
> SSE <- sum((results$predicted - results$real)^2)
> TSS <- sum( (mean(redwine$fixed.acidity) - results$real)^2)
> R2 <- 1-SSE/TSS
> R2
[1] 0.6607791
```

R-Squared (Coefficient of Determination) - This value lies between 0 and 1, and the higher it is, the better the model fits the data set.

Our main aim is to find those variables who give the lowest RMSE value and the highest R-Squared value.

The R-Squared for the training set is 66.07%. It means that the model can explain more than 66.07% of the variation.