CMTH642_Assignment_02

Tusaif Azmat

27/03/2022

R Markdown

QUESTIONS 1. Check the datatypes of the attributes. (3 points)

```
wine_df<-
read.csv("C:/Users/Zanara/Documents/Ryerson/Winter2022/CMTH642/CMTH642_winter
2022/A2/A2/winequality-white.csv",header= T,sep = ";")

head(wine_df)

##   fixed.acidity volatile.acidity citric.acid residual.sugar chlorides
## 1           7.0             0.27        0.36           20.7     0.045
## 2           6.3             0.30        0.34            1.6     0.049
## 3           8.1             0.28        0.40            6.9     0.050
## 4           7.2             0.23        0.32            8.5     0.058
## 5           7.2             0.23        0.32            8.5     0.058
## 6           8.1             0.28        0.40            6.9     0.050
##   free.sulfur.dioxide total.sulfur.dioxide density   pH sulphates alcohol
## 1                  45                  170  1.0010 3.00      0.45     8.8
## 2                  14                  132  0.9940 3.30      0.49     9.5
## 3                  30                   97  0.9951 3.26      0.44    10.1
## 4                  47                  186  0.9956 3.19      0.40     9.9
## 5                  47                  186  0.9956 3.19      0.40     9.9
## 6                  30                   97  0.9951 3.26      0.44    10.1
##   quality
## 1       6
## 2       6
## 3       6
## 4       6
## 5       6
## 6       6

#You could see the data types of each attribute under column names all double
except one integer, all numeric values

sapply(wine_df, class)

##        fixed.acidity     volatile.acidity          citric.acid
##            "numeric"            "numeric"            "numeric"
##       residual.sugar            chlorides  free.sulfur.dioxide
##            "numeric"            "numeric"            "numeric"
## total.sulfur.dioxide              density                   pH
##            "numeric"            "numeric"            "numeric"
##            sulphates              alcohol              quality
##            "numeric"            "numeric"            "integer"

#You could see the data types of each attribute
```

2. Are there any missing values in the dataset? (4 points)

```
which(is.na(wine_df))

## integer(0)

#There are no missing values in the dataset as seen below.
```

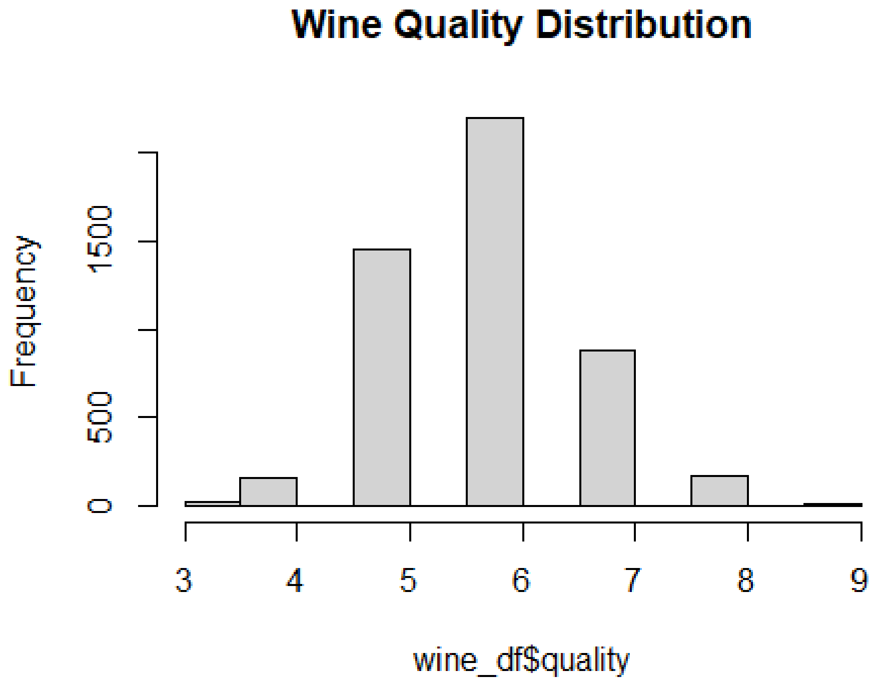
3. What is the correlation between the attributes other than Quality? (10 points)

```
cor(wine_df[-12])

##                      fixed.acidity volatile.acidity citric.acid
residual.sugar
## fixed.acidity           1.00000000      -0.02269729  0.28918070
0.08902070
## volatile.acidity       -0.02269729       1.00000000 -0.14947181
0.06428606
## citric.acid             0.28918070      -0.14947181  1.00000000
0.09421162
## residual.sugar          0.08902070       0.06428606  0.09421162
1.00000000
## chlorides               0.02308564       0.07051157  0.11436445
0.08868454
## free.sulfur.dioxide    -0.04939586      -0.09701194  0.09407722
0.29909835
## total.sulfur.dioxide    0.09106976       0.08926050  0.12113080
0.40143931
## density                 0.26533101       0.02711385  0.14950257
0.83896645
## pH                     -0.42585829      -0.03191537 -0.16374821    -
0.19413345
## sulphates              -0.01714299      -0.03572815  0.06233094    -
0.02666437
## alcohol                -0.12088112       0.06771794 -0.07572873    -
0.45063122
##                        chlorides free.sulfur.dioxide total.sulfur.dioxide
## fixed.acidity         0.02308564       -0.0493958591          0.091069756
## volatile.acidity      0.07051157       -0.0970119393          0.089260504
## citric.acid           0.11436445        0.0940772210          0.121130798
## residual.sugar        0.08868454        0.2990983537          0.401439311
## chlorides             1.00000000        0.1013923521          0.198910300
## free.sulfur.dioxide   0.10139235        1.0000000000          0.615500965
## total.sulfur.dioxide  0.19891030        0.6155009650          1.000000000
## density               0.25721132        0.2942104109          0.529881324
## pH                   -0.09043946       -0.0006177961          0.002320972
## sulphates             0.01676288        0.0592172458          0.134562367
## alcohol              -0.36018871       -0.2501039415         -0.448892102
##                           density            pH   sulphates     alcohol
## fixed.acidity        0.26533101 -0.4258582910 -0.01714299 -0.12088112
## volatile.acidity     0.02711385 -0.0319153683 -0.03572815  0.06771794
## citric.acid          0.14950257 -0.1637482114  0.06233094 -0.07572873
## residual.sugar       0.83896645 -0.1941334540 -0.02666437 -0.45063122
## chlorides            0.25721132 -0.0904394560  0.01676288 -0.36018871
## free.sulfur.dioxide  0.29421041 -0.0006177961  0.05921725 -0.25010394
## total.sulfur.dioxide 0.52988132  0.0023209718  0.13456237 -0.44889210
## density              1.00000000 -0.0935914935  0.07449315 -0.78013762
## pH                  -0.09359149  1.0000000000  0.15595150  0.12143210
## sulphates            0.07449315  0.1559514973  1.00000000 -0.01743277
## alcohol             -0.78013762  0.1214320987 -0.01743277  1.00000000
```

```
#A correlation is a number between -1 and +1 that measures the degree of
association between two Attributes (call them X and Y). A positive value for
the correlation implies a positive association. In this case large values of
X tend to be associated with large values of Y and small values of X tend to
be associated with small values of Y. A negative value for the correlation
implies a negative or inverse association. In this case large values of X
tend to be associated with small values of Y and vice versa.
#Following are the correlation values...of all attributes except quality.
```

4. Graph the frequency distribution of wine quality by using Quality. (10 points)

```
hist(wine_df$quality, main="Wine Quality Distribution")
```



5. Reduce the levels of rating for quality to two levels as Pass and Fail. Assign the levels of 3, 4 and 5 to level Fail; and 6, 7, 8 and 9 to level Pass. (10 points)

```
wine_df$quality<-as.factor(ifelse(wine_df$quality > 5,1,0))


table(wine_df$quality)

## 
##    0    1 
## 1640 3258

#fail = 0
#pass = 1
#I use zero for fail and one for pass and you could see below the values
```

6. Normalize the data set. (12 points)

```
normalize <- function(x) {
  return ((x-min(x))/(max(x)-min(x)))
}
#new normalized dataset created below
wine_df_new<-wine_df
wine_df_new[,-12] <- sapply(wine_df_new[,-12], normalize)
summary(wine_df_new)

##  fixed.acidity    volatile.acidity  citric.acid     residual.sugar  
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   Min.   :0.00000  
##  1st Qu.:0.2404   1st Qu.:0.1275   1st Qu.:0.1627   1st Qu.:0.01687  
##  Median :0.2885   Median :0.1765   Median :0.1928   Median :0.07055  
##  Mean   :0.2937   Mean   :0.1944   Mean   :0.2013   Mean   :0.08883  
##  3rd Qu.:0.3365   3rd Qu.:0.2353   3rd Qu.:0.2349   3rd Qu.:0.14264  
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000   Max.   :1.00000  
##    chlorides        free.sulfur.dioxide total.sulfur.dioxide    density       
##  Min.   :0.00000   Min.   :0.00000     Min.   :0.0000       Min.   
:0.00000  
##  1st Qu.:0.08012   1st Qu.:0.07317     1st Qu.:0.2297       1st 
Qu.:0.08892  
##  Median :0.10089   Median :0.11150     Median :0.2900       Median 
:0.12782  
##  Mean   :0.10912   Mean   :0.11606     Mean   :0.3001       Mean   
:0.13336  
##  3rd Qu.:0.12166   3rd Qu.:0.15331     3rd Qu.:0.3666       3rd 
Qu.:0.17332  
##  Max.   :1.00000   Max.   :1.00000     Max.   :1.0000       Max.   
:1.00000  
##        pH           sulphates         alcohol       quality 
##  Min.   :0.0000   Min.   :0.0000   Min.   :0.0000   0:1640  
##  1st Qu.:0.3364   1st Qu.:0.2209   1st Qu.:0.2419   1:3258  
##  Median :0.4182   Median :0.2907   Median :0.3871           
##  Mean   :0.4257   Mean   :0.3138   Mean   :0.4055           
##  3rd Qu.:0.5091   3rd Qu.:0.3837   3rd Qu.:0.5484           
##  Max.   :1.0000   Max.   :1.0000   Max.   :1.0000

#following are the normalized dataset values
```

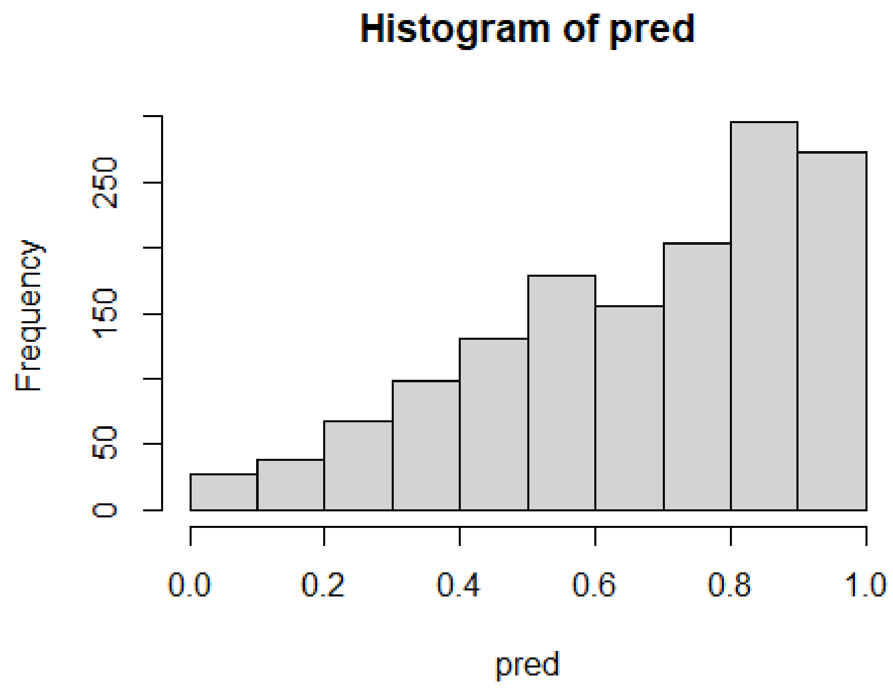
7. Divide the dataset to training and test sets. (12 points)

```
#I use the 70 30 split of dataset training and test sets
train_index = sample(1:nrow(wine_df_new),0.7*nrow(wine_df_new))
train.set= wine_df_new[train_index,]
test.set= wine_df_new[-train_index,]
```

8. Use the Logistic Regression algorithm to predict the quality of wine using its attributes. (12 points)

```
LR_model<-glm(formula =quality~.,data=train.set,family = "binomial")
summary(LR_model)

##
## Call:
## glm(formula = quality ~ ., family = "binomial", data = train.set)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.1817  -0.8704   0.4330   0.7926   2.6162
##
## Coefficients:
##                      Estimate Std. Error z value Pr(>|z|)
## (Intercept)            0.1633     0.4545   0.359  0.71939
## fixed.acidity          1.9322     0.9180   2.105  0.03531 *
## volatile.acidity      -6.8320     0.5063 -13.493  < 2e-16 ***
## citric.acid            0.7145     0.6116   1.168  0.24268
## residual.sugar        15.4507     2.1965   7.034 2.00e-12 ***
## chlorides              0.8175     0.6721   1.216  0.22385
## free.sulfur.dioxide    2.6850     0.9568   2.806  0.00501 **
## total.sulfur.dioxide  -0.2239     0.6184  -0.362  0.71734
## density              -23.8121     4.6769  -5.091 3.55e-07 ***
## pH                     1.9671     0.4885   4.027 5.65e-05 ***
## sulphates              2.1140     0.3793   5.573 2.50e-08 ***
## alcohol                3.3238     0.7173   4.634 3.59e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 4372.7  on 3427  degrees of freedom
## Residual deviance: 3417.7  on 3416  degrees of freedom
## AIC: 3441.7
##
## Number of Fisher Scoring iterations: 5

# Number of Fisher Scoring iterations: This is just a measure of how long it
took to fit your model. You can safely ignore it.

pred=predict(LR_model,type ='response',newdata = test.set)

predicted.quality<-ifelse(pred>=0.717189,1,0)
```

```
hist(pred)
```



```
# I decided to use the median to predict if the model can tell whether the
quality of a given wine will pass.
# Due to an imbalanced dataset we have clear problems with skewness in the
predicted variable of our model.
# This would affect the performance of the model by making it less accurate
at its prediction.
summary(pred)

##      Min.   1st Qu.    Median      Mean   3rd Qu.      Max. 
## 0.0001747 0.5031926 0.7165757 0.6660306 0.8730117 0.9934864
```

9. Display the confusion matrix to evaluate the model performance. (12 points)

```
c.matrix<-table(actual=test.set$quality,pred=predicted.quality)
c.matrix

##       pred
## actual   0   1
##      0 388 103
##      1 349 630

#The results are not quite convinced
```

10. Evaluate the model performance by computing Accuracy, Sensitivity and Specificity. (15 points)

```
TP=c.matrix["0","0"]
FP=c.matrix["1","0"]
FN=c.matrix["0","1"]
TN=c.matrix["1","1"]

#Accuracy
Accuracy=(TP+TN)/(TP+FN+FP+TN)
writeLines("Accuracy")

## Accuracy

Accuracy

## [1] 0.692517

#Sensitivity

Sensitivity=TP/(TP+FN)
writeLines("Sensitivity")

## Sensitivity

Sensitivity

## [1] 0.790224

#Specificity

Specificity=TN/(TN+FP)
writeLines("Specificity")

## Specificity

Specificity

## [1] 0.6435138
```

I obtained accuracy of the model 68.64 percent, Sensitivity of 82.07% and Specificity of 62.25% .

#This is the end of Assignment 2 ## R Markdown File