

# CMTH 642 Data Analytics: Advanced Methods

Assignment 3 (10%)

Rui Zhang

500736315

---

```
# INSERT YOUR CODE HERE.
```

```
wine <- read.csv("http://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-white.csv")
```

1. Import to R the following file: <http://archive.ics.uci.edu/ml/machine-learning-databases/wine-quality/winequality-white.csv> (The dataset is related to white Portuguese “Vinho Verde” wine. For more info: <https://archive.ics.uci.edu/ml/datasets/Wine+Quality>) (3 points)

```
# INSERT YOUR CODE HERE.
```

```
str(wine)
```

2. Check the datatypes of the attributes. (3 points)

```
## 'data.frame': 4898 obs. of 12 variables:
## $ fixed.acidity : num 7 6.3 8.1 7.2 7.2 8.1 6.2 7 6.3 8.1 ...
## $ volatile.acidity : num 0.27 0.3 0.28 0.23 0.23 0.28 0.32 0.27 0.3 0.22 ...
## $ citric.acid : num 0.36 0.34 0.4 0.32 0.32 0.4 0.16 0.36 0.34 0.43 ...
## $ residual.sugar : num 20.7 1.6 6.9 8.5 8.5 6.9 7 20.7 1.6 1.5 ...
## $ chlorides : num 0.045 0.049 0.05 0.058 0.058 0.05 0.045 0.045 0.049 0.044 ...
## $ free.sulfur.dioxide : num 45 14 30 47 47 30 30 45 14 28 ...
## $ total.sulfur.dioxide: num 170 132 97 186 186 97 136 170 132 129 ...
## $ density : num 1.001 0.994 0.995 0.996 0.996 ...
## $ pH : num 3 3.3 3.26 3.19 3.19 3.26 3.18 3 3.3 3.22 ...
## $ sulphates : num 0.45 0.49 0.44 0.4 0.4 0.44 0.47 0.45 0.49 0.45 ...
## $ alcohol : num 8.8 9.5 10.1 9.9 9.9 10.1 9.6 8.8 9.5 11 ...
## $ quality : int 6 6 6 6 6 6 6 6 6 6 ...
```

```
summary(wine)
```

```
## fixed.acidity volatile.acidity citric.acid residual.sugar
## Min. : 3.800 Min. :0.0800 Min. :0.0000 Min. : 0.600
## 1st Qu.: 6.300 1st Qu.:0.2100 1st Qu.:0.2700 1st Qu.: 1.700
```

```
## Median : 6.800 Median :0.2600 Median :0.3200 Median : 5.200
## Mean : 6.855 Mean :0.2782 Mean :0.3342 Mean : 6.391
## 3rd Qu.: 7.300 3rd Qu.:0.3200 3rd Qu.:0.3900 3rd Qu.: 9.900
## Max. :14.200 Max. :1.1000 Max. :1.6600 Max. :65.800
## chlorides free.sulfur.dioxide total.sulfur.dioxide density
## Min. :0.00900 Min. : 2.00 Min. : 9.0 Min. :0.9871
## 1st Qu.:0.03600 1st Qu.: 23.00 1st Qu.:108.0 1st Qu.:0.9917
## Median :0.04300 Median : 34.00 Median :134.0 Median :0.9937
## Mean :0.04577 Mean : 35.31 Mean :138.4 Mean :0.9940
## 3rd Qu.:0.05000 3rd Qu.: 46.00 3rd Qu.:167.0 3rd Qu.:0.9961
## Max. :0.34600 Max. :289.00 Max. :440.0 Max. :1.0390
## pH sulphates alcohol quality
## Min. :2.720 Min. :0.2200 Min. : 8.00 Min. :3.000
## 1st Qu.:3.090 1st Qu.:0.4100 1st Qu.: 9.50 1st Qu.:5.000
## Median :3.180 Median :0.4700 Median :10.40 Median :6.000
## Mean :3.188 Mean :0.4898 Mean :10.51 Mean :5.878
## 3rd Qu.:3.280 3rd Qu.:0.5500 3rd Qu.:11.40 3rd Qu.:6.000
## Max. :3.820 Max. :1.0800 Max. :14.20 Max. :9.000
```

```
# INSERT YOUR CODE HERE.
sum(is.na(wine))
```

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

```
## [1] 0
```

```
# no missing data
```

```
# INSERT YOUR CODE HERE.
wineNoEquality <- wine[, -grep("quality", names(wine))]
winecor <- cor(wineNoEquality)
winecor
```

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

```
## 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

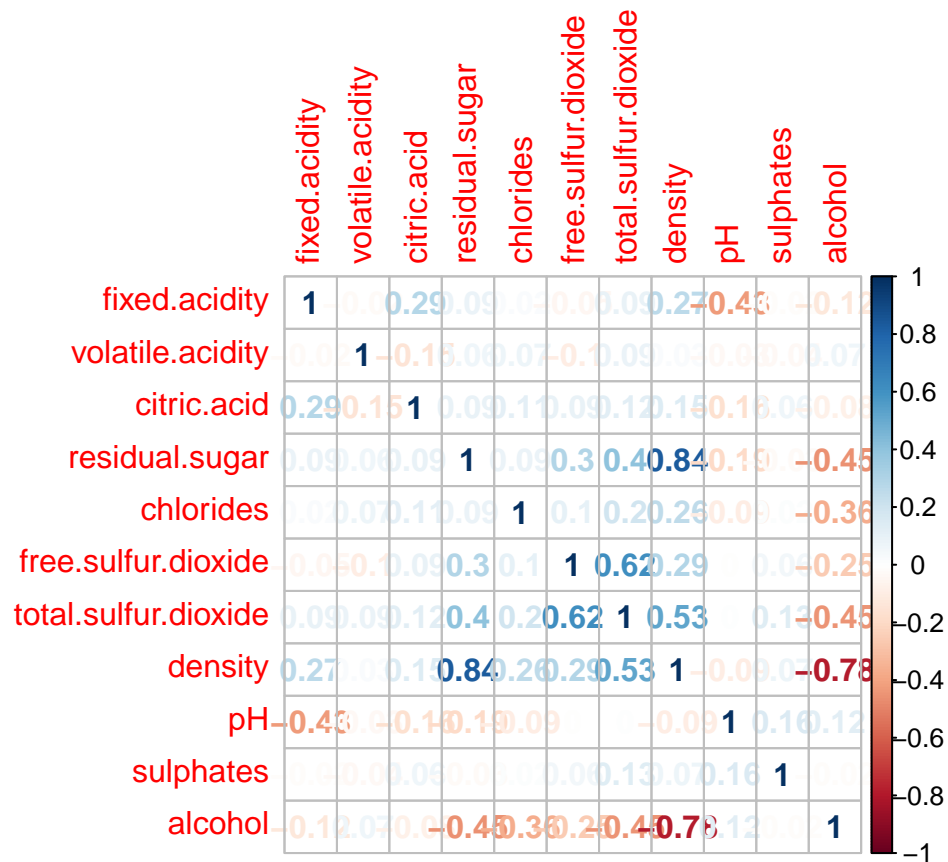
```
#install.packages("corrplot")
```

```
library(corrplot)
```

```
## Warning: package 'corrplot' was built under R version 4.0.2
```

```
## corrplot 0.84 loaded
```

```
corrplot(winecor, method="number")
```



```
# INSERT YOUR CODE
hist(wine$quality, ylim=c(0, 2500))
```

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



```
# INSERT YOUR CODE HERE.
wine$quality <- sapply(wine$quality, function(x) {
  if (x < 5) {
    return("low")
  } else if (x < 7) {
    return("medium")
  } else {
    return("high")
  }
})
wine$quality <- as.factor(wine$quality)
round(prop.table(table(wine$quality)) * 100, digits = 1)
```

6. Reduce the levels of rating for quality to three levels as high, medium and low. Assign the levels of 3 and 4 to level 0; 5 and 6 to level 1; and 7,8 and 9 to level 2. (10 points)

```
##
##   high   low medium
##   21.6   3.7  74.6
```

```
normalize <- function(x){
  return ((x - min(x)) / (max(x) - min(x)))
}
```

```
# INSERT YOUR CODE HERE.
wine[1:11] <- sapply(wine[1:11], normalize)
summary(wine)
```

7. Normalize the data set by using the following function: (12 points)

```
## 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    high :1060
## 1st Qu.:0.3364    1st Qu.:0.2209    1st Qu.:0.2419    low  : 183
## Median :0.4182    Median :0.2907    Median :0.3871    medium:3655
## 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
```

```
# INSERT YOUR CODE HERE.
set.seed(1)
train_index <- sample(1:nrow(wine), 0.7 * nrow(wine))
train.set <- wine[train_index,]
test.set <- wine[-train_index,]
```

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

```
# INSERT YOUR CODE HERE.
library("class")
library("gmodels")
```

9. Use the KNN algorithm to predict the quality of wine using its attributes. (12 points)

```
## Warning: package 'gmodels' was built under R version 4.0.2
```

```
train.set_new <- train.set[,-grep("quality", names(wine))]  
test.set_new <- test.set[,-grep("quality", names(wine))]  
wine_train_labels <- train.set$quality  
wine_test_labels <- test.set$quality  
wine_knn_prediction <- knn(train = train.set_new, test = test.set_new, cl= wine_train_labels, k = 3)  
head(wine_knn_prediction)
```

```
## [1] medium medium medium medium medium medium  
## Levels: high low medium
```

```
summary(wine_knn_prediction)
```

```
##      high      low medium  
##      290       22   1158
```

```
# INSERT YOUR CODE HERE.
```

```
ConfusionMatrix <- table(actual =wine_test_labels, predicted = wine_knn_prediction)  
ConfusionMatrix
```

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

```
##           predicted  
## actual    high low medium  
##   high    175   1   147  
##   low      6    3    34  
##   medium  109  18   977
```

```
CrossTable(x=wine_test_labels, y=wine_knn_prediction, prop.chisq=FALSE)
```

```
##  
##  
##      Cell Contents  
## |-----|  
## |                      N |  
## |      N / Row Total |  
## |      N / Col Total |  
## |      N / Table Total |  
## |-----|  
##  
##  
## Total Observations in Table:  1470  
##  
##  
##           | wine_knn_prediction  
## wine_test_labels |      high |      low |      medium | Row Total |
```

```
## -----|-----|-----|-----|-----|
##          high |      175 |         1 |      147 |      323 |
##          |      0.542 |      0.003 |      0.455 |      0.220 |
##          |      0.603 |      0.045 |      0.127 |      |
##          |      0.119 |      0.001 |      0.100 |      |
## -----|-----|-----|-----|
##          low  |         6 |         3 |        34 |        43 |
##          |      0.140 |      0.070 |      0.791 |      0.029 |
##          |      0.021 |      0.136 |      0.029 |      |
##          |      0.004 |      0.002 |      0.023 |      |
## -----|-----|-----|-----|
##          medium |      109 |        18 |      977 |      1104 |
##          |      0.099 |      0.016 |      0.885 |      0.751 |
##          |      0.376 |      0.818 |      0.844 |      |
##          |      0.074 |      0.012 |      0.665 |      |
## -----|-----|-----|-----|
##      Column Total |      290 |         22 |      1158 |      1470 |
##          |      0.197 |      0.015 |      0.788 |      |
## -----|-----|-----|-----|
##
##
```

```
# INSERT YOUR CODE HERE.
sum(diag(ConfusionMatrix))/nrow(test.set)
```

11. Evaluate the model performance by computing Accuracy, Sensitivity and Specificity. (12 points)

```
## [1] 0.7857143
```

This is the end of Assignment 3

Ceni Babaoglu, PhD