Assignment 2

R Markdown

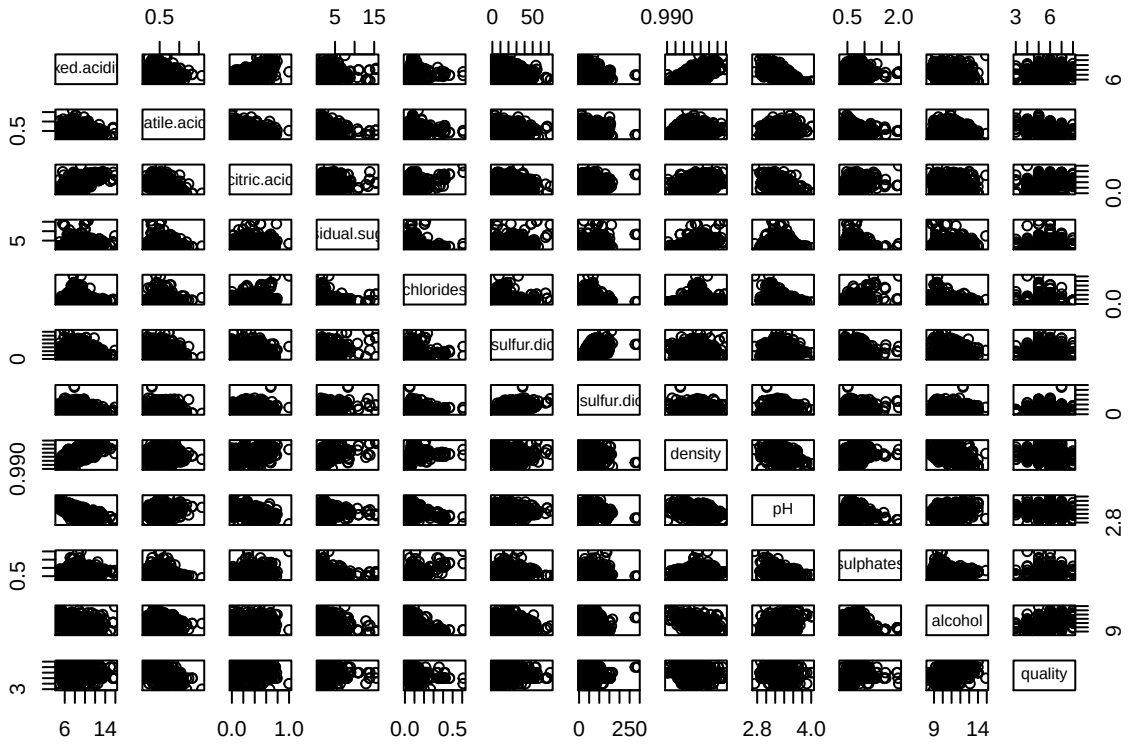
This is an R Markdown document. Markdown is a simple formatting syntax for authoring HTML, PDF, and MS Word documents. For more details on using R Markdown see http://rmarkdown.rstudio.com.

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:

```
redwine = read.csv("redwine.csv")
summary(redwine)
```

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

```
pairs(redwine)
```




```
cor(redwine)
```

```
##                       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
```

```
redwine$final_quality <- with(ifelse(quality>mean(quality), 1, 0), data=redwine)
```

```
glm.fit <- glm(final_quality ~ .-quality, data=redwine, family="binomial")
summary(glm.fit)
```

```
##
## Call:
## glm(formula = final_quality ~ . - quality, family = "binomial",
##     data = redwine)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -3.4025  -0.8387   0.3105   0.8300   2.3142
##
## Coefficients:
##                        Estimate Std. Error z value Pr(>|z|)
## (Intercept)           42.949948  79.473979   0.540  0.58890
## fixed.acidity          0.135980   0.098483   1.381  0.16736
## volatile.acidity      -3.281694   0.488214  -6.722 1.79e-11 ***
## citric.acid           -1.274347   0.562730  -2.265  0.02354 *
## residual.sugar         0.055326   0.053770   1.029  0.30351
## chlorides             -3.915713   1.569298  -2.495  0.01259 *
## free.sulfur.dioxide    0.022220   0.008236   2.698  0.00698 **
## total.sulfur.dioxide  -0.016394   0.002882  -5.688 1.29e-08 ***
## density              -50.932385  81.148745  -0.628  0.53024
## pH                    -0.380608   0.720203  -0.528  0.59717
## sulphates              2.795107   0.452184   6.181 6.36e-10 ***
## alcohol                0.866822   0.104190   8.320  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
```

```
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2209.0  on 1598  degrees of freedom
## Residual deviance: 1655.6  on 1587  degrees of freedom
## AIC: 1679.6
##
## Number of Fisher Scoring iterations: 4
```

```
glm.probs <- predict(glm.fit, type="response")
glm.preds <- ifelse(glm.probs>.5, "1", "0")
cm <- table(redwine$final_quality, glm.preds)
cm
```

```
##    glm.preds
##       0   1
##   0 549 195
##   1 214 641
```

```
set.seed(1)
rows <- sample(x=nrow(redwine), size=.75*nrow(redwine))
trainset <- redwine[rows, ]
testset <- redwine[-rows, ]
```

```
library(class)
set.seed(1)
sel.variables <- which(names(trainset)%in%c("fixed acidity", "volatile acidity", "citric acid", "residua

accuracies <- data.frame("k"=1:10, acc=NA)
for(k in 1:10){
  knn.pred <- knn(train=trainset[, sel.variables], test=testset[, sel.variables], cl=trainset$final_qual

  # test-error
  accuracies$acc[k]= round(sum(knn.pred!=testset$final_quality)/nrow(testset)*100,2)
}

accuracies
```

```
##     k   acc
## 1   1 25.75
## 2   2 29.25
## 3   3 28.50
## 4   4 28.25
## 5   5 29.00
## 6   6 30.00
## 7   7 28.75
## 8   8 28.75
## 9   9 28.50
## 10 10 29.00
```

```
set.seed(1)
rows <- sample(x=nrow(redwine), size=.8*nrow(redwine))
trainset <- redwine[rows, ]
testset <- redwine[-rows, ]
```

```
library(MASS)
lda.fit <- lda(final_quality ~.-quality, data=trainset)
lda.pred <- predict(lda.fit, testset)
```

```r
table(testset$final_quality, lda.pred$class)
```

```
##
##       0   1
##   0 105  27
##   1  47 141
```

```r
round(sum(lda.pred$class!=testset$final_quality)/nrow(testset)*100,2)
```

```
## [1] 23.12
```

```r
qda.fit <- qda(final_quality ~.-quality, data=trainset)
qda.pred <- predict(qda.fit, testset)
table(testset$final_quality, qda.pred$class)
```

```
##
##       0   1
##   0  90  42
##   1  36 152
```

```r
round(sum(qda.pred$class!=testset$final_quality)/nrow(testset)*100,2)
```

```
## [1] 24.38
```