HW4 - Stat 4510/7510

Yang, Anton – #14405729

Instructions: Please list your name and student number clearly. In order to receive credit for a problem, your solution must show sufficient detail so that the grader can determine how you obtained your answer.

Submit a single pdf generated using R Markdown. All R code should be included, as well as all output produced. Upload your work to the Canvas course site.

Problem 1

In this problem, we will once again consider the data found in `iris.csv` from Homework 1.

Read in the iris data and change `variety` to a factor variable. We will investigate different classification techniques to predict the iris `variety`.

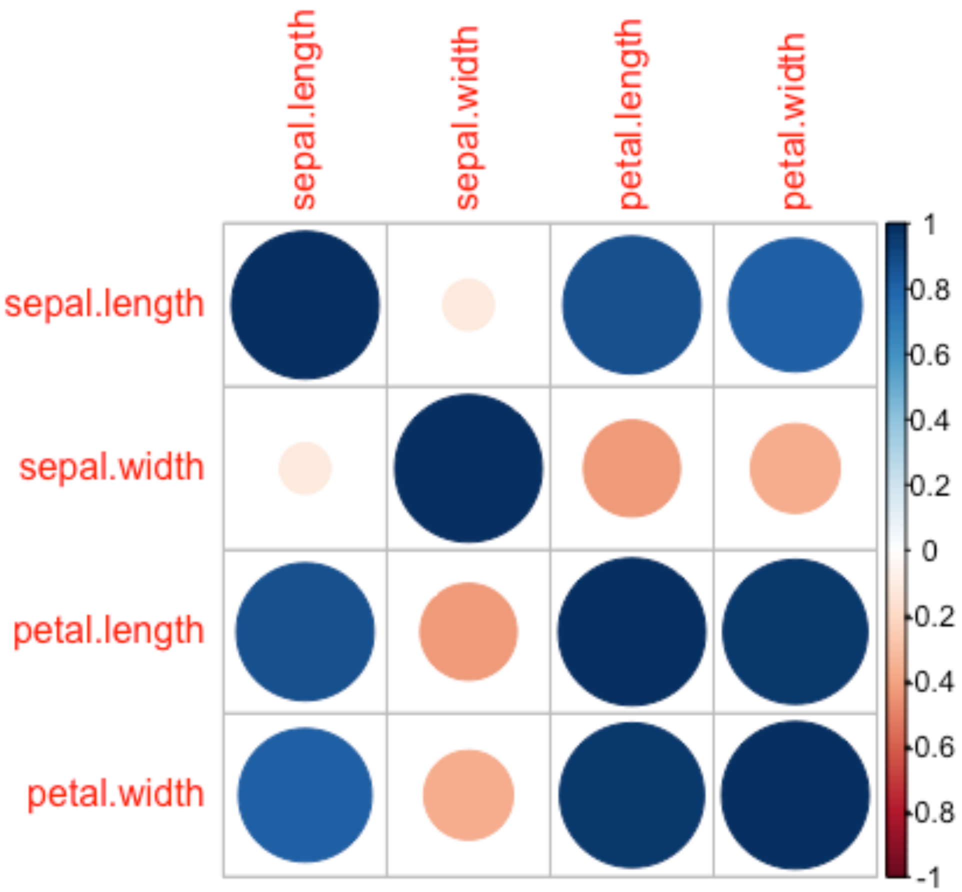
(a) Produce a correlation matrix for the quantitative variables. Which variables are strongly correlated?

```
library(corrplot)

## corrplot 0.92 loaded

library(class)
library(MASS)

data<-read.csv("iris.csv")
data$variety<-as.factor(data$variety)
quantitative<-data[, !names(data) %in% c("variety")]
correlation<-cor(quantitative)
corrplot(correlation)
```



According to the correlation matrix, petal length and sepal length, petal width and sepal length, petal length and petal width are all strongly correlated.

(b) Regardless of your answer to part (a), consider modeling the response variable `variety` using predictors `sepal.width` and `petal.width`. Split the data into a 50% training set and 50% test set. (Remember to set a seed of 1.) When defining these sets, include only the response and predictor variables.

```
set.seed(1)

split<-sample(1:nrow(data), size = 0.5*nrow(data))
training_set<-data[split,]
test_set<-data[-split,]
train_scale<-scale(training_set[,-which(names(training_set)=="variety")])
test_scale<-scale(test_set[,-which(names(test_set)=="variety")])
```

(c) For each of the models below, use the training data to predict `variety`. Print a confusion matrix for the test data, comparing the true test iris variety to the predicted test set iris variety. Calculate the test misclassification rate.

i. KNN: Use *K* nearest neighbors with `k=10`. (Remember to use `library(class)` before using `knn()`.)

ii. LDA (Remember to use `library(MASS)` for `lda()`.)

iii. QDA (Also requires `library(MASS)`, but if you used it for LDA, you don’t need to use it again.)

```
# KNN
knn<-knn(train = train_scale,
         test = test_scale,
         cl = training_set$variety,
         k = 10)
confusion_matrix_knn<-table(test_set$variety, knn)
print(confusion_matrix_knn)

##             knn
##              Setosa Versicolor Virginica
##   Setosa         22          0         0
##   Versicolor      0         26         4
##   Virginica       0          3        20

misclassification_knn<-1-
sum(diag(confusion_matrix_knn))/sum(confusion_matrix_knn)
print(misclassification_knn)

## [1] 0.09333333

#Linear Discriminant Analysis
#LDA is more stable than logistic regression when Y has classes that are well
separated. In addition, LDA is more stable than logistic regression when n is
small and the distribution the predictors X is approximately normal in each
class. Also, this distribution assume that the variances are all equal.
lda_model<-lda(variety~., data = training_set)
lda_prediction<-predict(lda_model, test_set, type="response")
confusion_matrix_lda<-table(test_set$variety, lda_prediction$class)
print(confusion_matrix_lda)

##
##              Setosa Versicolor Virginica
##   Setosa         22          0         0
##   Versicolor      0         29         1
##   Virginica       0          0        23

misclassification_lda<-1-
sum(diag(confusion_matrix_lda))/sum(confusion_matrix_lda)
print(misclassification_lda)

## [1] 0.01333333

#Quadratic Discriminant Analysis
#QDA is a more flexible approach, similar to LDA, but variance of each group
is estimated. The decision boundary is no longer linear, but quadratic.
qda_model<-qda(variety~., data = training_set)
qda_prediction<-predict(qda_model, test_set, type="response")
confusion_matrix_qda<-table(test_set$variety, qda_prediction$class)
print(confusion_matrix_qda)
```

```
##
##               Setosa Versicolor Virginica
##   Setosa          22          0         0
##   Versicolor       0         28         2
##   Virginica        0          0        23

misclassification_qda<-1-
sum(diag(confusion_matrix_qda))/sum(confusion_matrix_qda)
print(misclassification_qda)

## [1] 0.02666667
```

The misclassification of KNN is 0.09333, LDA is 0.01333, and QDA is 0.02667.

(d) Which of the models found in (c) seems to perform best?

Judging by the misclassification rate, the best model is Linear Discriminant Analysis by having a lowest misclassification rate of 0.0133.

(e) Why would we generally not use logistic regression in classification problem such as this one?

Logistic Regression would not be generally used because Logistic Regression is used to calculate the probability of belonging to a particular class. Since the Iris data are well separated classes, KNN, LDA, and QDA would be better for this classifying task.

Problem 2

The file `tumor.csv` was created from data compiled in the mid 1990s. Each record was generated from a digitized image of a fine needle aspirate (FNA) of a breast mass. They describe characteristics of the cell nuclei present in the image. It is of interest to classify the mass as benign (non-cancerous) or malignant (cancerous) based on a number of features which describe the mass. The columns of the dataset are as follows:

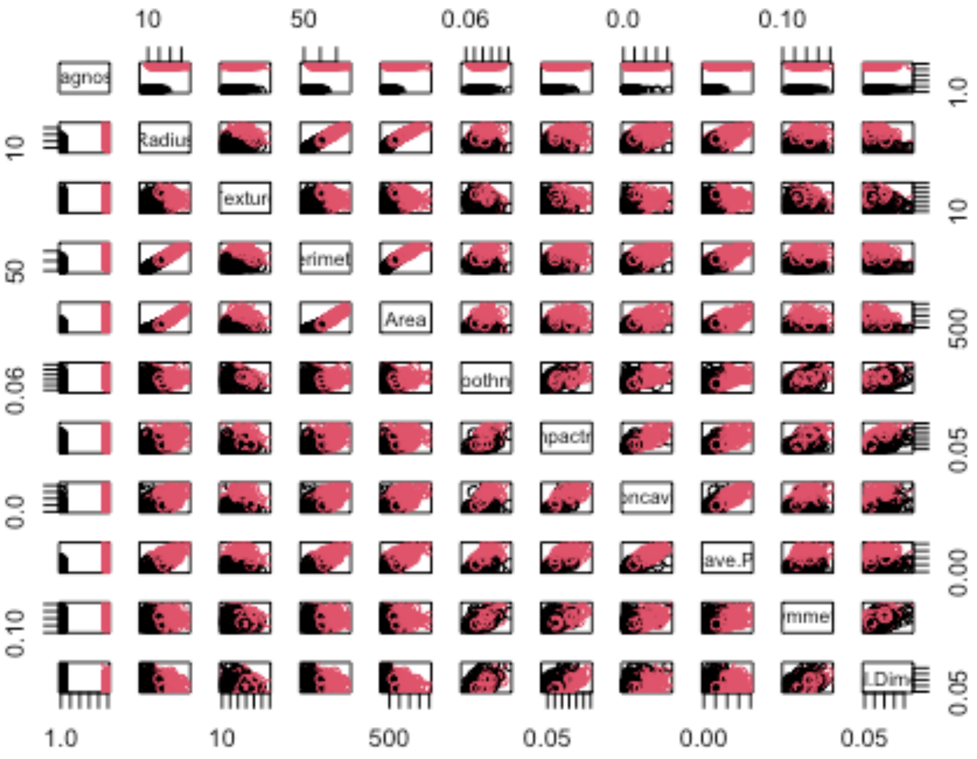
`Diagnosis` (Benign or Malignant)
`Radius` (mean of distances from center to points on the perimeter)
`Texture` (standard deviation of gray-scale values)
`Perimeter`
`Area`
`Smoothness` (local variation in radius lengths)
`Compactness` ($perimeter^2/area - 1.0$)
`Concavity` (severity of concave portions of the contour)
`Concave.Points` (number of concave portions of the contour)
`Symmetry`
`Fractal.Dimension` ("coastline approximation" - 1)

(a) Explore the data.

- Define `Diagnosis` as a factor.
- Produce a pairwise scatterplot matrix, colored by 'Diagnosis'.

- Which variables seem useful for predicting Diagnosis?

```
data<-read.csv("tumor.csv")
data$Diagnosis<-as.factor(data$Diagnosis)
pairs(data,
      col = as.numeric(data$Diagnosis))
```



It seems that radius, perimeter, area, and Concave.Points are useful for predicting Diagnosis.

(b) Split the data into an 80% training and 20% test set, being sure to set a seed of 1 for consistency.

```
set.seed(1)
split<-sample(1:nrow(data), size = 0.8*nrow(data))
training_set<-(data[split, ])
test_set<-(data[-split, ])
```

(c) Using the training data, fit a logistic regression model predicting the probability of a malignant tumor. Using the following predictors: Radius, Texture, Smoothness, Concavity, Symmetry, and Fractal.Dimension. Produce a summary of the model. Which of the selected variables are significant?

```
logistic_model<-
glm(Diagnosis~Radius+Texture+Smoothness+Concavity+Symmetry+Fractal.Dimension,
data=training_set,family="binomial")
summary(logistic_model)

##
## Call:
## glm(formula = Diagnosis ~ Radius + Texture + Smoothness + Concavity +
##     Symmetry + Fractal.Dimension, family = "binomial", data =
training_set)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.0661  -0.1446  -0.0294   0.0273   3.3081
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)       -36.39534    6.95305  -5.234 1.65e-07 ***
## Radius              1.11759    0.20488   5.455 4.90e-08 ***
## Texture             0.37403    0.06937   5.392 6.98e-08 ***
## Smoothness        116.94082   26.01787   4.495 6.97e-06 ***
## Concavity          22.00171    6.16189   3.571 0.000356 ***
## Symmetry           21.71386   11.81624   1.838 0.066117 .
## Fractal.Dimension -75.75657   66.58841  -1.138 0.255253
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 600.34  on 454  degrees of freedom
## Residual deviance: 125.70  on 448  degrees of freedom
## AIC: 139.7
##
## Number of Fisher Scoring iterations: 8
```

According to the summary, Intercept, Radius, Texture, Smoothness, Concavity, and Symmetry are significant.

(d) Produce a confusion matrix for the test data, using a probability threshold of 0.5 for classifying a tumor as Malignant. What is the total misclassification rate?

Note: In this week's R Tutorial video, the classifications for the logistic regression were 0 and 1. In this data, the classifications will be Benign and Malignant. If you plan to use the code in the tutorial as a guide, you'll need to replace the 0 with "Benign" and the 1 with "Malignant" (including the quotation marks). Note each word is Capitalized in data.

```
threshold<-0.5

logistic_prediction<-predict(logistic_model, test_set, type="response")
predicted_classes<-ifelse(logistic_prediction>=threshold, "Malignant",
"Benign")
```

```
confusion_matrix_log<-table(test_set$Diagnosis, predicted_classes)
print(confusion_matrix_log)
```

```
##            predicted_classes
##             Benign Malignant
##   Benign        69         2
##   Malignant      5        38
```

```
misclassification_log<-1-
sum(diag(confusion_matrix_log))/sum(confusion_matrix_log)
print(misclassification_log)
```

```
## [1] 0.06140351
```

The misclassification rate of logistic model is 0.06140351 if the threshold is 0.5