ST4061 – Computer Intensive Statistical Analytics II
2020-2021
In-class test 1B – (corrected Question 1)

NAME AND SURNAME: VIDHYA JANARTHANAN

STUDENT NUMBER: 120220137

PROGRAM: MSc Data Science and Analytics

INTRUCTIONS

- Provide your answers in this document, after each question.
- Paste the R code you used for each question item.
- **Save your files regularly.**
- **Email eric.w@ucc.ie for any question.**

Question 1

Load the following libraries and dataset into your R session, and split the dataset into a training set `x.train` (all observations prior to 2019) and a test set `x.test` (all 2019 observations) as follows:

```
library(class)
library(MASS)
x = read.csv(file="CA1_data.csv", stringsAsFactors=TRUE)
itrain = which(x$Year<2019)
x.train = x[itrain,]
x.test = x[-itrain,]
```

This dataset contains variables relating to a company's commercial activity between 2015 and 2019. The response variable of interest in this question is `x$Increase`, which indicates whether the stock for the company increases on that day. All covariates in the dataset except for `Year` are used as potential predictors.

(1) Fit a logistic regression model using all predictors except for `Year`. Provide the corresponding confusion matrix obtained for the test set `x.test`.

Your answer:

Confusion Matrix:

```
glm.p   No Yes
FALSE 28   4
TRUE   4  17
```

R code for (1):

```
> library(class)
> library(MASS)
>x=read.csv(file="D:/ML_2/CA1_data.csv",
stringsAsFactors=TRUE)
> itrain = which(x$Year<2019)
> x$Year = NULL
> x.train = x[itrain,]
> x.test = x[-itrain,]
> nrow(x.train)
[1] 210
> nrow(x.test)
[1] 53
> # train classifiers:
>glm.o=glm(Increase~.,data,x.train.out,
family=binomial(logit))
> glm.p = ( predict(glm.o, newdata=x.test,
type="response") > 0.5)
> tb.glm = table(glm.p,x.test$Increase)
> tb.glm
> acc.glm = sum(diag(tb.glm))/sum(tb.glm)
> acc.glm
[1] 0.8490566
```

(2) Perform linear discriminant analysis using all predictors except for `Year`. Provide the corresponding confusion matrix obtained for the test set `x.test`.

Your answer:

Confusion Matrix :

```
lda.p No Yes
No   26  5
Yes   6 16
```

R code for (2):

```
> library(class)
> library(MASS)
>x=read.csv(file="D:/ML_2/CA1_data.csv",
stringsAsFactors=TRUE)
> itrain = which(x$Year<2019)
> x$Year = NULL
> x.train = x[itrain,]
> x.test = x[-itrain,]
> nrow(x.train)
[1] 210
> nrow(x.test)
[1] 53
>
> # train classifiers:
> lda.o = lda(Increase~.,data = x.train)
> lda.p = predict(lda.o, newdata=x.test)$class
> tb.lda = table(lda.p,x.test$Increase)
> tb.lda

lda.p No Yes
No  26   5
Yes  6  16
> acc.lda = sum(diag(tb.lda))/sum(tb.lda)
> acc.lda
[1] 0.7924528
```

(3) Perform quadratic discriminant analysis using all predictors except for `Year`. Provide the corresponding confusion matrix obtained for the test set `x.test`.

Your answer:

Confusion Matrix:

```
qda.p No Yes
No    27  5
Yes    5  16
```

R code for (3):

```
> library(class)
> library(MASS)
>                   x=read.csv(file="D:/ML_2/CA1_data.csv",
stringsAsFactors=TRUE)
> itrain = which(x$Year<2019)
> x$Year = NULL
> x.train = x[itrain,]
> x.test = x[-itrain,]
> nrow(x.train)
[1] 210
> nrow(x.test)
[1] 53
> # train classifiers:
> qda.o = qda(Increase~.,data = x.train)
> qda.p = predict(qda.o, newdata=x.test)$class
> tb.qda = table(qda.p,x.test$Increase)
> tb.qda
qda.p No Yes
No    27   5
Yes    5  16
> acc.qda = sum(diag(tb.qda))/sum(tb.qda)
> acc.qda
[1] 0.8113208
```

(4) Compare the classifiers obtained in (2) and (3) and explain, using relevant output:

a) How they differ;
b) Why they differ.

(Note: If you did not manage to make your code work, you may still answer what you would expect to find for (a) and (b).)

Your answer:

a. The level of accuracy in QDA is 81% which is slightly higher compared to LDA .QDA predicted 27 cases whereas LDA predicted 26 cases.

b. LDA assumes in both classes the covariance matrix is same thus resulting in linear decision boundary whereas in QDA there is a separate covariance matrix for each class which results in quadratic decision boundary.

Question 2

A student is analyzing a dataset in R named `dat`, whose top 6 rows are shown below. This dataset comprises of 150 observations summarized by 7 covariates `x1`, …, `x7`, and a 3-level categorical response variable named `type`:

x1	x2	x3	x4	x5	x6	x7	type
5	86	68	28	30.2	0.364	24	class3
7	195	70	33	25.1	0.163	55	class2
5	77	82	41	35.8	0.156	35	class3
0	165	76	43	47.9	0.259	26	class2
0	107	60	25	26.4	0.133	23	class1
5	97	76	27	35.6	0.378	52	class3

(1) The student is applying the kth-nearest neighbours classifier to the above dataset, using the cross-validated value k=5, with the below R instruction:

```
set.seed(1)
itrain = sample(1:150, 100)
dat.train = dat[itrain, -8]
Y.train = dat[itrain, 8]
dat.test = dat[-itrain, -8]
knn.out = knn(dat.train, dat.test, Y.train, k=5)
```

What is the student doing wrong when calling `knn()`? Briefly explain your answer.

Your answer:
The values are not scaled according to the above code. If the scale of feature value varies then normalization is required as because the distance calculation is done using the feature scaled values in KNN.

(2) Briefly discuss whether logistic regression could be applied to this dataset, and why.

Your answer:
Logistic regression cannot be applied to the dataset as it expects only two output classes. Here we have 3 classes to predict which is done by different classification named Multinomial.