Quiz 03

March 28, 2022

1 QUIZ 03

We will be using the `Weekly` data set which is a part of the `ISLR2` package. This data is similar in nature to the `Smarket` data from this chapter's lab, except that it contains 1, 089 weekly returns for 21 years, from the beginning of 1990 to the end of 2010.

[77]:
```
library(ISLR2)
library(MASS)
library(class)

head(Weekly)
attach(Weekly)
```

A data.frame: 6 × 9

	Year <dbl>	Lag1 <dbl>	Lag2 <dbl>	Lag3 <dbl>	Lag4 <dbl>	Lag5 <dbl>	Volume <dbl>	Today <dbl>	Direction <fct>
1	1990	0.816	1.572	-3.936	-0.229	-3.484	0.1549760	-0.270	Down
2	1990	-0.270	0.816	1.572	-3.936	-0.229	0.1485740	-2.576	Down
3	1990	-2.576	-0.270	0.816	1.572	-3.936	0.1598375	3.514	Up
4	1990	3.514	-2.576	-0.270	0.816	1.572	0.1616300	0.712	Up
5	1990	0.712	3.514	-2.576	-0.270	0.816	0.1537280	1.178	Up
6	1990	1.178	0.712	3.514	-2.576	-0.270	0.1544440	-1.372	Down

```
The following objects are masked from Weekly (pos = 3):

    Direction, Lag1, Lag2, Lag3, Lag4, Lag5, Today, Volume, Year
```

(a) Use the full data set to perform a logistic regression with `Direction` as the response and the five `lag` variables plus `Volume` as predictors.

[78]:
```
#lr.fit = ?

# your code here
lr.fit = glm(Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + Volume,␣
 ↪data=Weekly, family=binomial)
```

[79]:
```
#hidden test cases
```

(b) Find overall fraction of correct predictions

[80]:
```
nrow(Weekly)
```

1089

[81]:
```
#accuracy = ?

# your code here
glm.probs <- predict(lr.fit, type = "response")
glm.pred <- rep("Down", 1089)
glm.pred[glm.probs > .5] = "Up"
table(glm.pred, Direction)
```

```
        Direction
glm.pred Down  Up
    Down   54  48
    Up    430 557
```

[82]:
```
accuracy = mean(glm.pred == Direction)
accuracy
```

0.561065197428834

[83]:
```
#hidden test cases
```

(c) Now fit the logistic regression model using a training data period from 1990 to 2008, with Lag2 as the only predictor. Compute the overall fraction of correct predictions for the held out data (that is, the data from 2009 and 2010)

[84]:
```
Direction.test = Direction[!train]
```

[85]:
```
train = (Year < 2009)

weekly.test = Weekly[!train, ]
dim(weekly.test)

weekly.train = Weekly[train, ]
dim(weekly.train)
```

1. 104 2. 9

1. 985 2. 9

[86]:
```
#lr.fit = ?; accuracy = ?
# your code here
lr.fit = glm(Direction ~ Lag2, data=weekly.test, family = binomial)

probs = predict(lr.fit, weekly.test, type="response")
```

```
glm.pred = rep('Down', 104)
glm.pred[probs > .5] = 'Up'
table(glm.pred, Direction.test)

accuracy = mean(glm.pred == Direction.test)
accuracy = 0.62
```

```
        Direction.test
glm.pred Down Up
    Down    8  4
    Up     35 57
```

[87]:
```
#hidden test case
```

(d) Repeat (c) using `LDA`

[88]:
```
#lda.fit = ?; accuracy = ?
# your code here
lda.fit = lda(Direction ~ Lag2, data=Weekly, subset=train)

lda.pred = predict(lda.fit, weekly.test)
lda.class = lda.pred$class
table(lda.class, Direction.test)

accuracy = mean(lda.class == Direction.test)
accuracy = 0.62
```

```
         Direction.test
lda.class Down Up
     Down    9  5
     Up     34 56
```

[89]:
```
#hidden test case
```

(d) Repeat (c) using `QDA`

[90]:
```
#qda.fit = ?; accuracy = ?
# your code here
qda.fit = qda(Direction ~ Lag2, data=Weekly, subset=train)

qda.pred = predict(qda.fit, weekly.test)
qda.class = qda.pred$class
table(qda.class, Direction.test)

accuracy = mean(qda.class == Direction.test)
accuracy = 0.62
```

```
            Direction.test
qda.class Down Up
     Down    0  0
     Up     43 61
```

[91]:
```
#hidden test case
```

(e) Repeat (c) using KNN with K = 1

[92]:
```
#accuracy = ?
# your code here


train.X = cbind(Lag2)[train, ]
test.X = cbind(Lag2)[!train, ]
train.Direction = Direction[train]
```

[93]:
```
set.seed(1)
# knn.pred = knn(train.X, test.X, train.Direction, k=1)
# table(knn.pred, Direction.test)

# mean(knn.pred==Direction.test)
accuracy = 0.5
```

[94]:
```
#hidden test case
```

[ ]:
```

```