Assignment 4

SURESH KUMAR .R

2024-10-03

```
library(ISLR)
library(MASS)
library(class)
library(boot)
library(glmnet)
```

```
## Loading required package: Matrix
```

```
## Loaded glmnet 4.1-8
```

```
weekly=Weekly
```

Question 6:

We continue to consider the use of a logistic regression model to predict the probability of default using income and balance on the Default data set. In particular, we will now compute estimates for the standard errors of the income and balance logistic regression coefficients in two different ways: (1) using the bootstrap, and (2) using the standard formula for computing the standard errors in the glm ()function. Do not forget to set a random seed before beginning your analysis.

(a) Using the summary() and glm() functions, determine the estimated standard errors for the coefficients associated with income and balance in a multiple logistic regression model that uses both predictors.

```
Q6_a.fit=glm(default~income+balance,data = Default,family = binomial)
summary(Q6_a.fit)
```

```
##
## Call:
## glm(formula = default ~ income + balance, family = binomial,
##     data = Default)
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.154e+01  4.348e-01 -26.545  < 2e-16 ***
## income       2.081e-05  4.985e-06   4.174 2.99e-05 ***
## balance      5.647e-03  2.274e-04  24.836  < 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: 2920.6  on 9999  degrees of freedom
## Residual deviance: 1579.0  on 9997  degrees of freedom
## AIC: 1585
##
## Number of Fisher Scoring iterations: 8
```

- After seeing the summary balance has high standard error 0.0002274 and income has small compare to balance 0.000004985

(b) Write a function, boot.fn(), that takes as input the Default data set as well as an index of the observations, and that outputs the coefficient estimates for income and balance in the multiple logistic regression model.

```
boot.fn=function(data,index){
  q6_b.fit=glm(default~income+balance,data=data,family = binomial,subset =
index)
  return(coef(q6_b.fit))
}
```

(c) Use the boot() function together with your boot.fn() function to estimate the standard errors of the logistic regression coefficients for income and balance.

```
boot(Default,boot.fn,100)
```

```
##
## ORDINARY NONPARAMETRIC BOOTSTRAP
##
##
## Call:
## boot(data = Default, statistic = boot.fn, R = 100)
##
##
## Bootstrap Statistics :
##          original        bias     std. error
## t1* -1.154047e+01 -2.430725e-02 4.441540e-01
## t2*  2.080898e-05  2.275891e-07 4.520269e-06
## t3*  5.647103e-03  7.315510e-06 2.259176e-04
```

(d) Comment on the estimated standard errors obtained using the glm() function and using your bootstrap function.

```
0.000004985-0.000004255      #glm and bootstrap income
```

```
## [1] 7.3e-07
```

```
0.0002274-0.00022348       #glm and bootstrap balance
```

```
## [1] 3.92e-06
```

```
0.00000073
```

```
## [1] 7.3e-07
```

```
0.00000392
```

```
## [1] 3.92e-06
```

- For income the difference between glm and bootstrap is 0.00000073.
- For balance the difference between glm and bootstrap is 0.00000392.
- The estimated standard errors obtained by two methods there is difference.
- In bootstrap the standard error is low.
- after comparing glm and bootstrap the standard error is decreased in boostrap function. In glm function standard error is litte high. *

Question 7:

In Sections 5.3.2 and 5.3.3, we saw that the cv.glm() function can be used in order to compute the LOOCV test error estimate. Alternatively, one could compute those quantities using just the glm() and predict.glm() functions, and a for loop. You will now take this approach in order to compute the LOOCV error for a simple logistic regression model on the Weekly data set. Recall that in the context of classification problems, the LOOCV error is given in (5.4).

(a) Fit a logistic regressionmodel that predicts Direction using Lag1 and Lag2.

```
data("Weekly")

Q7_a.fit=glm(Direction~Lag1+Lag2,data=weekly,family = binomial)
summary(Q7_a.fit)
```

```
##
## Call:
## glm(formula = Direction ~ Lag1 + Lag2, family = binomial, data = weekly)
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.22122    0.06147   3.599 0.000319 ***
## Lag1        -0.03872    0.02622  -1.477 0.139672
## Lag2         0.06025    0.02655   2.270 0.023232 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1496.2  on 1088  degrees of freedom
## Residual deviance: 1488.2  on 1086  degrees of freedom
## AIC: 1494.2
##
## Number of Fisher Scoring iterations: 4
```

(b) Fit a logistic regressionmodel that predicts Direction using Lag1 and Lag2 using all but the first observation.

```
Q7_b.fit=glm(Direction~Lag1+Lag2,data = weekly[-1,],family = binomial)
```

(c) Use the model from (b) to predict the direction of the first observation.

```
Q7_probs=predict(Q7_b.fit,weekly[1,],type = "response")>0.5

error=0

acutal=weekly[1,]$Direction=="Up"

if(acutal!=Q7_probs)
  error=1
error
```

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

(d) Write a for loop from i = 1 to i = n, where n is the number of observations in the data set, that performs each of the following steps:

- i. Fit a logistic regression model using all but the ith observation to predict Direction using Lag1 and Lag2. *
- ii. Compute the posterior probability of the market moving up for the ith observation. *
- iii. Use the posterior probability for the ith observation in order to predict whether or not the market moves up. *
- iv. Determine whether or not an error was made in predicting the direction for the ith observation. If an error was made, then indicate this as a 1, and otherwise indicate it as a 0. *

```
error_Q8=rep(0,dim(weekly)[1])
for (i in 1:dim(weekly)[1]){
  fit_Q8=glm(Direction~Lag1+Lag2,data = weekly[-i,],family = binomial)
  prob_Q8=predict(fit_Q8,weekly[i,],type="response")>0.5
  Actual_Q8=weekly[i,]$Direction=="Up"
  if(Actual_Q8 != prob_Q8)
    error_Q8[i]=1
  }
sum(error_Q8)
```

```
## [1] 490
```

(e) Take the average of the n numbers obtained in (d)iv in order to obtain the LOOCV estimate for the test error. Comment on the results.

```
mean(error_Q8)
```

```
## [1] 0.4499541
```

- As i am seeing average n numbers test error is 0.44 i think it is high