Prepared by Richard Rennehan

NSCC Institute of Technology

DBAS3075 Introduction to Statistical Learning

Logistic regression assignment

April 4, 2018

# Questions and Answers

//Add references and r code to assignment 4

1. (1 pt) When using the glm() function to perform logistic regression, what must the family argument of the function be defined as?

* When using glm() to perform logistic regression, we define the family as binomial
* family=binomial

2. (1 pt) What are the dimensions of the Default data set?

* The Default data set has 10000 observations across 4 variables. This would make it 10000 by 4.

3. (1 pt) When using the predict() function to make predictions using a logistic regression model, what must the type argument of the function be defined as?

* The type argument must be defined as response
* (type=”response”)

4. (3 pts) Create a logistic regression model using the Default data set. Use default as the response variable, and use student, balance, and income as the input variables. When reporting the model, do not use scientific notation (e.g., 5.2 x 10^3). Do not round the results provided in the summary. Also be sure to indicate which value corresponds to 1 for the student indicator variable.

* *glm.fit=glm(default~student+balance+income, data=data, family=binomial)*

*summary(glm.fit)*

*contrasts(student)*

* The model is:
  + default = -10.87 – 0.6468(studentYes) + 0.005737(balance) + 0.000003033(income)
* The value ‘Yes’ corresponds to 1 for the student indicator variable. If the person is a student, -0.6468 is taken off the default. If the person is not a student, this does not apply.

5. (2 pts) Use the model from question 4 to predict the log-odds value for a student who has a $1000 balance and earns $30000. What is the resulting log-odds?

* default = -10.87 – 0.6468(1) + 0.005737(balance) + 0.000003033(income) = -5.68881
* This is the exponent number we will use in the probability equation

6. (3 pts) What is the probability of the student in question 5 defaulting on their credit card balance?

* We use this equation to determine probability: p(X) =
* Therefore:
* p(default) = = 0.003372%

7. (1 pts) True or False: In the logistic regression model, the coefficient of the balance variable means that the probability of defaulting on a loan increases by 0.005737 for every 1 dollar increase in balance.

* False. It increases the log odds by 0.005737 for every 1 dollar increase in balance. It does not affect the probability itself.

8. (3 points) Of the observations used to create the logistic regression model, approximately what percent of those observations does the logistic regression model correctly predict will have defaulted on their loan?

*glm.probs=predict(glm.fit, type="response")*

*contrasts(default)*

*glm.pred=rep("No",10000)*

*glm.pred[glm.probs>0.5]="Yes"*

*table(glm.pred, default)*

*predicted\_correctly= (9627+105)/10000*

* The model predicted 97.32% of observations correctly based on who defaulted on their loan.

# References

## R Code

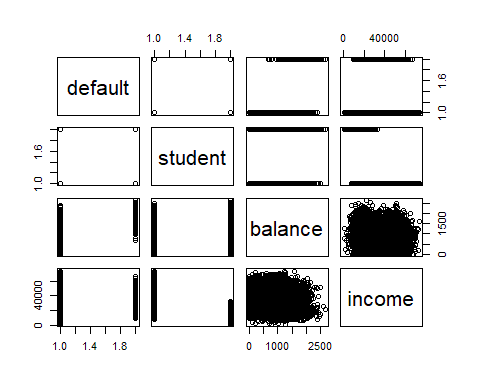
**library**("ISLR", lib.loc="~/R/win-library/3.4") *#gives us data sets from ISLR book*  
  
data = Default  
  
**names**(data)

## [1] "default" "student" "balance" "income"

**summary**(data)

## default student balance income   
## No :9667 No :7056 Min. : 0.0 Min. : 772   
## Yes: 333 Yes:2944 1st Qu.: 481.7 1st Qu.:21340   
## Median : 823.6 Median :34553   
## Mean : 835.4 Mean :33517   
## 3rd Qu.:1166.3 3rd Qu.:43808   
## Max. :2654.3 Max. :73554

**pairs**(data)



**cor**(data[,**-c**(1,2)])

## balance income  
## balance 1.0000000 -0.1522434  
## income -0.1522434 1.0000000

**attach**(data)  
  
glm.fit=**glm**(default**~**., data=data, family=binomial) *#Fitting of the logistic regression model*   
  
**summary**(glm.fit)

##   
## Call:  
## glm(formula = default ~ ., family = binomial, data = data)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.4691 -0.1418 -0.0557 -0.0203 3.7383   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.087e+01 4.923e-01 -22.080 < 2e-16 \*\*\*  
## studentYes -6.468e-01 2.363e-01 -2.738 0.00619 \*\*   
## balance 5.737e-03 2.319e-04 24.738 < 2e-16 \*\*\*  
## income 3.033e-06 8.203e-06 0.370 0.71152   
## ---  
## 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: 1571.5 on 9996 degrees of freedom  
## AIC: 1579.5  
##   
## Number of Fisher Scoring iterations: 8

**contrasts**(student)

## Yes  
## No 0  
## Yes 1

glm.probs=**predict**(glm.fit, type="response")  
  
glm.probs[1**:**100]

## 1 2 3 4 5   
## 1.428724e-03 1.122204e-03 9.812272e-03 4.415893e-04 1.935506e-03   
## 6 7 8 9 10   
## 1.989518e-03 2.333767e-03 1.086718e-03 1.638333e-02 2.080617e-05   
## 11 12 13 14 15   
## 1.065494e-05 1.127658e-02 8.079339e-05 7.082706e-04 1.198231e-02   
## 16 17 18 19 20   
## 1.127261e-04 2.217400e-05 2.168507e-04 3.725718e-04 1.091248e-02   
## 21 22 23 24 25   
## 8.251335e-05 4.984332e-03 9.416093e-03 8.294806e-04 1.780102e-03   
## 26 27 28 29 30   
## 2.765383e-03 8.626263e-04 8.124350e-02 7.330172e-04 6.411795e-03   
## 31 32 33 34 35   
## 3.835888e-04 1.373435e-04 3.057396e-04 4.127461e-03 3.630822e-02   
## 36 37 38 39 40   
## 5.352329e-02 1.164651e-03 1.589925e-04 1.510057e-03 9.086529e-04   
## 41 42 43 44 45   
## 3.206842e-03 2.704995e-03 1.578907e-02 3.719052e-05 1.465416e-01   
## 46 47 48 49 50   
## 2.340637e-04 2.629126e-02 2.390736e-02 2.435898e-03 2.860334e-03   
## 51 52 53 54 55   
## 5.492011e-04 1.604407e-02 5.746857e-02 1.872554e-03 2.215325e-05   
## 56 57 58 59 60   
## 1.698678e-02 1.551328e-02 1.293824e-01 4.284882e-02 3.543138e-04   
## 61 62 63 64 65   
## 1.774345e-03 1.207849e-03 2.100952e-05 9.419759e-02 6.319516e-03   
## 66 67 68 69 70   
## 3.571070e-02 8.915175e-03 2.173402e-05 3.064826e-03 1.157620e-03   
## 71 72 73 74 75   
## 6.459004e-04 2.612818e-04 6.132660e-02 8.295005e-02 2.810424e-03   
## 76 77 78 79 80   
## 7.841275e-04 8.881863e-04 1.422631e-03 3.227265e-05 6.660624e-05   
## 81 82 83 84 85   
## 2.399835e-03 2.555333e-04 3.093754e-04 5.597673e-04 1.524431e-04   
## 86 87 88 89 90   
## 4.267646e-04 3.412854e-03 2.635748e-05 4.649844e-03 1.734828e-03   
## 91 92 93 94 95   
## 4.358174e-02 6.259131e-02 7.581097e-03 1.874122e-04 2.882804e-04   
## 96 97 98 99 100   
## 2.451665e-03 3.658825e-04 8.853925e-03 8.897307e-05 6.369292e-05

**contrasts**(default)

## Yes  
## No 0  
## Yes 1

glm.pred=**rep**("No",10000)   
glm.pred[glm.probs**>**0.5]="Yes" *#If the chances of being defaulted are greater than half, assume the person represented is defaulted*  
  
**table**(glm.pred, default)

## default  
## glm.pred No Yes  
## No 9627 228  
## Yes 40 105

predicted\_correctly=(9627**+**105)**/**10000