Registration number= 2106227

R version 4.1.1 (2021-08-10) -- "Kick Things"

Copyright (C) 2021 The R Foundation for Statistical Computing

Platform: x86\_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.

You are welcome to redistribute it under certain conditions.

Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.

Type 'contributors()' for more information and

'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or

'help.start()' for an HTML browser interface to help.

Type 'q()' to quit R.

> setwd("D:/Statistical Methods/Test")

> ##=============## Question 1 ##=============##

>

> # Q1 1) Load data set, set 'rank' as factor variable

> ######### Write your code here ###############

>

> Binary.data<-read.csv('binary.csv',header = T)

> head(Binary.data)

admit gre gpa rank

1 0 380 3.61 3

2 1 660 3.67 3

3 1 800 4.00 1

4 1 640 3.19 4

5 0 520 2.93 4

6 1 760 3.00 2

> rank.factor=factor(Binary.data$rank)

> # display rank factor variable#############

>

>

>

> ##############################################

>

>

>

>

> # Q1 2)(i) training test split

> ######### Write your code here ###############

> library(boot)

> set.seed(227)

> train.index=sample(x=400,size=280)

> d.train=Binary.data[train.index,]

> d.test=Binary.data[-train.index,]

>

>

>

>

>

> ##############################################

>

>

>

> # Q1 2)(ii) fit logistic

> ######### Write your code here ###############

> glm.fit=glm(admit~.,data=d.train,family = binomial(link = "logit"))

> summary(glm.fit)

Call:

glm(formula = admit ~ ., family = binomial(link = "logit"), data = d.train)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.5020 -0.9081 -0.6623 1.2078 2.0915

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) -2.949897 1.245062 -2.369 0.017823 \*

gre 0.002569 0.001254 2.049 0.040469 \*

gpa 0.570657 0.368248 1.550 0.121225

rank -0.497293 0.146145 -3.403 0.000667 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 358.71 on 279 degrees of freedom

Residual deviance: 333.24 on 276 degrees of freedom

AIC: 341.24

Number of Fisher Scoring iterations: 4

>

>

>

>

>

> ##############################################

>

>

> # Q1 2)(iii) define cost functions

> ######### Write your code here ###############

> error.rate.fn=function(y,prediction){

+ d.predict=prediction>0.3

+ error.rate=mean(y!=d.predict)

+ return(error.rate)

+ }

>

>

>

>

>

>

>

>

>

> ##############################################

>

> # Q1 2)(iv) cross-validation

> ######### Write your code here ###############

> cv.fit=cv.glm(data = d.train,glmfit = glm.fit,cost=error.rate.fn,K=10)

> cv.val=cv.fit$delta[1]

> cv.val

[1] 0.425

>

>

> ##############################################

>

>

>

> # Q1 2)(v) repeat (iv)(v) with another threshold

> ######### Write your code here ###############

> error.rate.fn1=function(y,prediction){

+ d.predict1=prediction>0.5

+ error.rate1=mean(y!=d.predict1)

+ return(error.rate1)

+ }

> cv.fit1=cv.glm(data = d.train,glmfit = glm.fit,cost=error.rate.fn1,K=10)

> cv.val1=cv.fit1$delta[1]

> cv.val1

[1] 0.3321429

>

>

> ##############################################

>

>

>

> # Q1 3) test error

> ######### Write your code here ###############

> d.predict=predict(object=glm.fit,newdata = d.test,type='response')

> td.error=d.predict>0.3

> td.error1=d.predict>0.5

> error1=mean(td.error!=d.test$admit)

> error2=mean(td.error1!=d.test$admit)

> error1

[1] 0.4916667

> error2

[1] 0.2333333

>

> ##############################################

>

> # Q1 4) confusion table

> ######### Write your code here ###############

> cm1=table(d.test$admit,td.error)

>

> cm2=table(d.test$admit,td.error1)

> cm1

td.error

FALSE TRUE

0 38 50

1 9 23

> cm2

td.error1

FALSE TRUE

0 84 4

1 24 8

>

> ##############################################

>

>

> ##=========## End of Question 1 ##==========##

>

>

>

>

>

>

>

>

>

> ##=============## Question 2 ##=============##

>

> # Q2 1) Load dataset, report the size

> ######### Write your code here ###############

> Awards.data<-read.csv('Awards.csv',header = T)

> dim(Awards.data)

[1] 200 3

> head(Awards.data)

num\_awards prog final

1 0 Vocational Pass

2 0 General Pass

3 0 Vocational Pass

4 0 Vocational Pass

5 0 Vocational Pass

6 0 General Pass

>

>

>

>

>

> ##############################################

>

> ######## number of observations? #############

> # There are 200 observations in 'Awards' dataset

> #

> #

> #

> ##############################################

>

>

>

>

>

> # Q2 2) fit Poisson regression and analyze

> ######### Write your code here ###############

> pfit=glm(num\_awards~.,data = Awards.data,family = poisson(link="log"))

> summary(pfit)

Call:

glm(formula = num\_awards ~ ., family = poisson(link = "log"),

data = Awards.data)

Deviance Residuals:

Min 1Q Median 3Q Max

-1.8548 -1.0282 -0.5962 0.5075 2.5363

Coefficients:

Estimate Std. Error z value Pr(>|z|)

(Intercept) 0.5424 0.1200 4.519 6.21e-06 \*\*\*

progGeneral -1.1800 0.3587 -3.290 0.00100 \*\*

progVocational -0.9496 0.3214 -2.955 0.00313 \*\*

finalPass -1.0966 0.1935 -5.668 1.44e-08 \*\*\*

---

Signif. codes: 0 ‘\*\*\*’ 0.001 ‘\*\*’ 0.01 ‘\*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for poisson family taken to be 1)

Null deviance: 287.67 on 199 degrees of freedom

Residual deviance: 201.41 on 196 degrees of freedom

AIC: 385.47

Number of Fisher Scoring iterations: 6

>

> ##############################################

>

>

> ######### Comment on the result ##############

> # The intercept,progGeneral,progVocational and finalpass are significant under 5% significance level

> #

> #

> #

> ##############################################

>

>

>

>

>

>

>

> # Q2 3) Compute average for the listed cases

> ######### Write your code here ###############

> b.pois=coefficients(pfit)

> b.pois

(Intercept) progGeneral progVocational finalPass

0.5423742 -1.1799887 -0.9495939 -1.0966386

>

>

> # i)Pass in General program

> exp(b.pois[1]+b.pois[2]+b.pois[4])

(Intercept)

0.176532

>

> # ii)Distinction in General program

> exp(b.pois[1]+b.pois[2])

(Intercept)

0.5285518

>

>

> # iii)Pass in Academic program

> exp(b.pois[1]+b.pois[4])

(Intercept)

0.5744947

>

> # iv)Distinction in academic program

>

> exp(b.pois[1])

(Intercept)

1.720086

> # v)Pass in vocational program

>

> exp(b.pois[1]+b.pois[3]+b.pois[4])

(Intercept)

0.2222709

> # vi)Distinction in Vocational program

> exp(b.pois[1]+b.pois[3])

(Intercept)

0.665498

>

>

>

> ##############################################

>

> ######### Comment on the result ##############

> # On average the individuals distinction in academic program expects largest number of awards

> # and individuals that Pass in General program expects lowest number of awards

> #

> #

> ##############################################

>

>

>

>

> # Q2 4) Is the model better than the NULL model

> ######### Write your code here ###############

> 358.71-333.24

[1] 25.47

> 279-276

[1] 3

> 1-pchisq(25.47,3)

[1] 1.231286e-05

>

>

>

>

>

>

>

>

>

> ##############################################

>

> ######### Comment on the result ##############

> # The p value ids less that 0.5 thus the fitted model is significantly better than the null model under 5%significance level

> #

> #

> #

> ##############################################

>

> ##=========## End of Question 2 ##==========##

>