Mehmet Duman

Solution to Homework #4—MTH 522

Problem 1 (Chapter 5 Exercises 5):

In Chapter 4, we used logistic regression to predict the probability of default using income and balance on the Default data set. We will now estimate the test error of this logistic regression model using the validation set approach. Do not forget to set a random seed before beginning your analysis.

(a) Fit a logistic regression model that uses income and balance to predict default.

```
> library(ISLR)
> summary(Default)

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


> attach(Default)
> set.seed(1)
> fit.glm = glm(default ˜ income + balance, data = Default, family = "binomial")
> summary(fit.glm)

Call:
glm(formula = default ˜ income + balance, family = "binomial",
data = Default)

Deviance Residuals:
Min       1Q   Median       3Q      Max
-2.4725  -0.1444  -0.0574  -0.0211   3.7245

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
```

(b) Using the validation set approach, estimate the test error of this model. In order to do this, you must perform the following steps:

(b) i. Split the sample set into a training set and a validation set.

```
> train = sample(dim(Default)[1], dim(Default)[1] / 2)
```

(b) ii. Fit a multiple logistic regression model using only the training observations.

```
> fit.glm = glm(default ˜ income + balance, data = Default, family = "binomial", subset = train)
> summary(fit.glm)

Call:
glm(formula = default ˜ income + balance, family = "binomial",
data = Default, subset = train)

Deviance Residuals:
Min       1Q   Median       3Q      Max
-2.3583  -0.1268  -0.0475  -0.0165   3.8116

Coefficients:
Estimate Std. Error z value Pr(>|z|)
(Intercept) -1.208e+01  6.658e-01 -18.148   <2e-16 ***
income       1.858e-05  7.573e-06   2.454   0.0141 *
balance      6.053e-03  3.467e-04  17.457   <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: 1457.0  on 4999  degrees of freedom
Residual deviance:  734.4  on 4997  degrees of freedom
AIC: 740.4

Number of Fisher Scoring iterations: 8
```

(b) iii. Obtain a prediction of default status for each individual in the validation set by computing the posterior probability of default for that individual, and classifying the individual to the default category if the posterior probability is greater than 0.5.

```
> probs = predict(fit.glm, newdata = Default[-train, ], type = "response")
> pred.glm = rep("No", length(probs))
> pred.glm[probs > 0.5] = "Yes"
```

(b) iv. Compute the validation set error, which is the fraction of the observations in the validation set that are misclassified.

```
> mean(pred.glm != Default[-train, ]$default)
[1] 0.0286
```

Using the validation set approach, the test error rate is 2.86%.

(c) Repeat the process in (b) three times, using three different splits of the observations into a training set and a validation set. Com- ment on the results obtained.

```
> train = sample(dim(Default)[1], dim(Default)[1] / 2)
>  fit.glm = glm(default ˜ income+ balance, data=Default, family="binomial", subset=train)
> probs = predict(fit.glm, newdata = Default[-train, ], type = "response")
>  pred.glm = rep("No", length(probs))
>  pred.glm[probs > 0.5] = "Yes"
> mean(pred.glm != Default[-train, ]$default)
[1] 0.0236


> train = sample(dim(Default)[1], dim(Default)[1] / 2)
>  fit.glm = glm(default ˜ income+ balance, data=Default, family="binomial", subset=train)
> probs = predict(fit.glm, newdata = Default[-train, ], type = "response")
>  pred.glm = rep("No", length(probs))
>  pred.glm[probs > 0.5] = "Yes"
> mean(pred.glm != Default[-train, ]$default)
[1] 0.028


> train = sample(dim(Default)[1], dim(Default)[1] / 2)
>  fit.glm = glm(default ˜ income+ balance, data=Default, family="binomial", subset=train)
> probs = predict(fit.glm, newdata = Default[-train, ], type = "response")
>  pred.glm = rep("No", length(probs))
>  pred.glm[probs > 0.5] = "Yes"
> mean(pred.glm != Default[-train, ]$default)
[1] 0.0268
```

Each time we get different test error rate. The average test error rate is 2.613333%.

(d) Now consider a logistic regression model that predicts the probability of default using income, balance, and a dummy variable for student. Estimate the test error for this model using the validation set approach. Comment on whether or not including a dummy variable for student leads to a reduction in the test error rate.

using :income, balance, student

```
> train = sample(dim(Default)[1], dim(Default)[1] / 2)
>  fit.glm=glm(default˜income+balance+student,data=Default,family="binomial",subset=train)
> probs = predict(fit.glm, newdata = Default[-train, ], type = "response")
>  pred.glm = rep("No", length(probs))
>  pred.glm[probs > 0.5] = "Yes"
> mean(pred.glm != Default[-train, ]$default)

[1] 0.0264
```

The test error rate is 2.64%, including a dummy variable for student didn’t leads to a reduction in the test error rate.

Problem 2 (Chapter 5 Exercises 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.

```
>   set.seed(1)
> attach(Default)
The following objects are masked from Default (pos = 3):

balance, default, income, student




> fit.glm = glm(default ˜ income + balance, data = Default, family = "binomial")
>  summary(fit.glm)

Call:
glm(formula = default ˜ income + balance, family = "binomial",
data = Default)

Deviance Residuals:
Min            1Q        Median            3Q           Max
-2.472542783  -0.144435943  -0.057366212  -0.021063920   3.724454436

Coefficients:
                   Estimate      Std. Error   z value   Pr(>|z|)
(Intercept) -1.15404684e+01  4.34756357e-01 -26.54468 < 2.22e-16 ***
income       2.08089755e-05  4.98516718e-06   4.17418 2.9906e-05 ***
balance      5.64710294e-03  2.27373142e-04  24.83628 < 2.22e-16 ***
---
Signif. codes:  0 *** 0.001 ** 0.01 * 0.05 . 0.1   1

(Dispersion parameter for binomial family taken to be 1)

Null deviance: 2920.649711  on 9999  degrees of freedom
Residual deviance: 1578.966270  on 9997  degrees of freedom
AIC: 1584.96627

Number of Fisher Scoring iterations: 8
```

The glm() functions estimated standard errors for the coefficients are 4.34756357e-01 , 4.98516718e-06 and 2.27373142e-04

(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) {
+ fit = glm(default ~ income + balance, data = data, family = "binomial", subset = index)
+ return (coef(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.

```
> library(boot)
> boot(Default, boot.fn, 300)

ORDINARY NONPARAMETRIC BOOTSTRAP


Call:
boot(data = Default, statistic = boot.fn, R = 300)


Bootstrap Statistics :
         original        bias     std. error
t1* -1.154047e+01 -6.310201e-02 4.424387e-01
t2*  2.080898e-05 -1.442808e-07 4.886839e-06
t3*  5.647103e-03  3.762129e-05 2.301732e-04
```

The boot.fn() function to estimates the standard errors of the logistic regression coefficients are 4.424387e-01, 4.886839e-06 and 2.301732e-04

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

The estimated standard errors obtained using the glm() function and using the bootstrap function are pretty close.

β_0 β_1 β_2

```
4.34756357e-01 , 4.98516718e-06  and    2.27373142e-04
4.424387e-01   , 4.886839e-06     and   2.301732e-04
```

Solution:

$$p_k(x) = \frac{\pi_k \frac{1}{\sqrt{2\pi\sigma}} \exp(-\frac{1}{2\sigma^2}(x-\mu_k)^2)}{\sum \pi_l \frac{1}{\sqrt{2\pi\sigma}} \exp(-\frac{1}{2\sigma^2}(x-\mu_l)^2)} \tag{1}$$