Chapter 5 Problem 5

Andira Putri

September 20, 2018

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

```
library(ISLR)
data(Default)
log.reg=glm(default~income+balance,data=Default,family=binomial)
summary(log.reg) #just curious about the results
```

```
##
## 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 this, you must perform the following steps.

1. Split the data into a training set and validation set.
2. Fit a multiple logistic regression model using only the training observations.
3. 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.
4. Compute the validation set error.

```
#1. Split the data!
set.seed(1)
train = sample(10000,5000)
#2. Fit logistic reg. model using training data!
log.reg1=glm(default~income+balance,data=Default,family=binomial,subset=train)
#3. Predict default status1
```

```r
valid=Default[-train,]
log.prob1=predict(log.reg1,type="response")
log.pred1=rep("No",nrow(valid))
log.pred1[log.prob1>0.5]="Yes"
#4. Compute the test error!
table(log.pred1, valid$default)
```

```
##
## log.pred1   No  Yes
##       No  4756  166
##       Yes   77    1
```

```r
mean(valid$default != log.pred1)
```

```
## [1] 0.0486
```

c.) Repeat the process in (b) three times, using three different splits of the observations into a training set and validation set.

```r
#First repeat. Change seed each time.
set.seed(2)
train1 = sample(10000,5000)
log.reg1=glm(default~income+balance,data=Default,family=binomial,subset=train1)
valid1=Default[-train1,]
log.prob1=predict(log.reg1,type="response")
log.pred1=rep("No",nrow(valid1))
log.pred1[log.prob1>0.5]="Yes"
table(log.pred1, valid1$default)
```

```
##
## log.pred1   No  Yes
##       No  4759  165
##       Yes   72    4
```

```r
mean(valid1$default != log.pred1)
```

```
## [1] 0.0474
```

```r
#Second repeat
set.seed(3)
train = sample(10000,5000)
log.reg1=glm(default~income+balance,data=Default,family=binomial,subset=train)
valid=Default[-train,]
log.prob1=predict(log.reg1,type="response")
log.pred1=rep("No",nrow(valid))
log.pred1[log.prob1>0.5]="Yes"
table(log.pred1, valid$default)
```

```
##
## log.pred1   No  Yes
##       No  4764  153
##       Yes   80    3
```

```r
mean(valid$default != log.pred1)
```

```
## [1] 0.0466
```

```r
#Third repeat
set.seed(4)
train = sample(10000,5000)
log.reg1=glm(default~income+balance,data=Default,family=binomial,subset=train)
valid=Default[-train,]
log.prob1=predict(log.reg1,type="response")
log.pred1=rep("No",nrow(valid))
log.pred1[log.prob1>0.5]="Yes"
table(log.pred1, valid$default)
```

```
##
## log.pred1   No  Yes
##       No  4763  166
##       Yes   69    2
```

```r
mean(valid$default != log.pred1)
```

```
## [1] 0.047
```

The test error rates are similar among all iterations. Perhaps the validation set approach is a good method to use for this data set in particular.

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. Does including a dummy variable for `student` lead to a reduction in test error rate?

```r
#Create dummy variables
set.seed(1)
contrasts(Default$student)
```

```
##     Yes
## No    0
## Yes   1
```

```r
train=sample(10000,5000)
valid=Default[-train,]
log.reg2=glm(default~income+balance+student,data=Default,family=binomial,subset=train)
log.prob2=predict(log.reg2,type="response")
log.pred2=rep("No",nrow(valid))
log.pred2[log.prob2>0.5]="Yes"
table(log.pred2,valid$default)
```

```
##
## log.pred2   No  Yes
##       No  4756  166
##       Yes   77    1
```

```r
mean(valid$default !=log.pred2)
```

```
## [1] 0.0486
```

The `student` dummy variable did not impact the test error rate that much.