Modern Applied Statistics exercises from ISLR

Yamuna Dhungana

Use set.seed(16489) in each exercise to make results reproducible. Do not use the `cv.glm` function or similar functions.

1. Question 5.4.3 on page 198

3 We now review k-fold cross-validation. a) Explain how k-fold cross-validation is implemented.

Let us say, n be the no of observation and K be the no of folds. The test set will be of the length n/k and The remaining length of data i.e. $n - n/k$ is training set data. The test data do not overlap. Error is then calculated for each K and then averaged. For example, we have the number of observations 1000, k is 5, Test set for each K is 1000/5 which is 200 without replacement. Now the error is calculated by the training data and then validated with the test data. After calculating the error for all k the error is averaged, and the final error is K-fold cross-validation.

(b) What are the advantages and disadvantages of k-fold crossvalidation relative to:

i. The validation set approach?

Advantage:

1. The Validation estimates of the test error can be highly variable, depending on precisely which observations are included in the training set and which observations are included in the validation set.
2. Validation set error rate may tend to overestimate the test error rate for the model fit on the entire data set.

Disadvantage:

1. Validation set approach is conceptually simple and easy to implement.

ii. LOOCV?

Advantage:

1. LOOCV requires fitting the statistical learning method n times. This has the potential to be computationally intensive.
2. k-fold CV often gives more accurate estimates of the test error rate than does LOOCV.

Disadvantage:

If we need to reduce bias, LOOCV should be preferred over k-fold CV because it tends to have less bias. So, there is a bias-variance trade-off associated with the choice of k in k-fold cross-validation. Generally, if we use $k = 5$ or $k = 10$ that yield test error rate estimates which suffer neither from excessively high bias nor from very high variance.

2. Question 5.4.5 on page 198 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.

```
##
## Call:
## glm(formula = default ~ balance + income, 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 ***
## balance      5.647e-03  2.274e-04  24.836  < 2e-16 ***
## income       2.081e-05  4.985e-06   4.174 2.99e-05 ***
## ---
## 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
```

2(a)

After fitting the model, we found out that both the balance and the income is statistically significant.

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

i.Split the sample set into a training set and a validation set.

```
##   No  Yes
## 6446  221
```

```
##   No  Yes
## 3221  112
```

b(i)

By using the set.set equal to 16489 as mentioned in question, we split the data into training set and validation set.

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

```
##
## Call:
## glm(formula = default ~ balance + income, family = binomial,
##     data = train.data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4339  -0.1410  -0.0569  -0.0213   3.5036
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.140e+01  5.319e-01 -21.440  < 2e-16 ***
```

```
## balance      5.629e-03  2.786e-04  20.207  < 2e-16 ***
## income       1.699e-05  6.108e-06   2.783  0.00539 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1940.4  on 6666  degrees of freedom
## Residual deviance: 1051.4  on 6664  degrees of freedom
## AIC: 1057.4
##
## Number of Fisher Scoring iterations: 8
```

A logistic model to predict the default status was made using the income and balance on the training set.

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.

The prediction of default status is dummy coded here. With 1 indicating the default status and 0 indicating non-default status.

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

Table 1: Validation set Error

	Error Rate
Accuracy	97.450
TPR	33.036
FPR	0.310

The test error was estimated based on the validation set. The accuracy of the model is 97.450 that means error rate was (100-97.450)= 2.55%. Likewise, true positive rate of the model is 33.036% and False positive rate is 0.310%.

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

```
##   No  Yes
## 8059  274
```

```
##   No  Yes
## 1608   59
```

```
##
## Call:
## glm(formula = default ~ balance + income, family = binomial,
##     data = train.data.1)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4426  -0.1418  -0.0554  -0.0203   3.7446
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.148e+01  4.774e-01 -24.042  < 2e-16 ***
```

```
## balance      5.678e-03  2.522e-04  22.514  < 2e-16 ***
## income       1.655e-05  5.438e-06   3.043  0.00234 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2410.2  on 8332  degrees of freedom
## Residual deviance: 1300.8  on 8330  degrees of freedom
## AIC: 1306.8
##
## Number of Fisher Scoring iterations: 8
```

```
##   No  Yes
## 6044  206
```

```
##   No  Yes
## 3623  127
```

```
##
## Call:
## glm(formula = default ~ balance + income, family = binomial,
##     data = train.data.2)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4402  -0.1398  -0.0567  -0.0214   3.4959
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.142e+01  5.505e-01 -20.753   <2e-16 ***
## balance      5.623e-03  2.879e-04  19.532   <2e-16 ***
## income       1.799e-05  6.334e-06   2.841   0.0045 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1811.07  on 6249  degrees of freedom
## Residual deviance:  982.76  on 6247  degrees of freedom
## AIC: 988.76
##
## Number of Fisher Scoring iterations: 8
```

```
##   No  Yes
## 5379  177
```

```
##   No  Yes
## 4288  156
```

```
##
## Call:
## glm(formula = default ~ balance + income, family = binomial,
##     data = train.data.3)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
```

```
## -2.4049  -0.1377  -0.0559  -0.0210   3.5252
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.133e+01  5.840e-01 -19.399   <2e-16 ***
## balance      5.628e-03  3.081e-04  18.264   <2e-16 ***
## income       1.406e-05  6.757e-06   2.081   0.0375 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1568.36  on 5555  degrees of freedom
## Residual deviance:  850.99  on 5553  degrees of freedom
## AIC: 856.99
##
## Number of Fisher Scoring iterations: 8
```

Table 2: Validation set Error

	Error Rate with 1st split	Error Rate with 2nd split	Error Rate with 3rd split
Accuracy	97.361	97.493	97.277
TPR	32.203	33.858	31.410
FPR	0.249	0.276	0.326

I have split the data into 3 sets. The set.seed is set as mentioned. For all the three splits the accuracy is diffenence with only 0.1 The TPR and FPR also has the similar output.

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.

```
##   No  Yes
## 4838  162
```

```
##   No  Yes
## 4829  171
```

```
##
## Call:
## glm(formula = default ~ balance + income + student, family = binomial,
##     data = train.data.d)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4293  -0.1383  -0.0550  -0.0208   3.5400
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.096e+01  7.113e-01 -15.406   <2e-16 ***
## balance      5.691e-03  3.286e-04  17.319   <2e-16 ***
## income       4.912e-06  1.196e-05   0.411    0.681
## studentYes  -3.520e-01  3.386e-01  -1.040    0.298
## ---
```

```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1429.88  on 4999  degrees of freedom
## Residual deviance:  778.45  on 4996  degrees of freedom
## AIC: 786.45
##
## Number of Fisher Scoring iterations: 8
```

Table 3: Validation set Error

	Error Rate
Accuracy	97.380
TPR	33.918
FPR	0.373

```
##   No  Yes
## 8059  274
```

```
##   No  Yes
## 1608   59
```

```
##
## Call:
## glm(formula = default ~ balance + income + student, family = binomial,
##     data = train.data.1d)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4386  -0.1393  -0.0539  -0.0198   3.7683
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.082e+01  5.415e-01 -19.986   <2e-16 ***
## balance      5.765e-03  2.571e-04  22.426   <2e-16 ***
## income      -8.136e-07  9.027e-06  -0.090   0.9282
## studentYes  -6.274e-01  2.594e-01  -2.419   0.0156 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 2410.2  on 8332  degrees of freedom
## Residual deviance: 1295.1  on 8329  degrees of freedom
## AIC: 1303.1
##
## Number of Fisher Scoring iterations: 8
```

```
##   No  Yes
## 6044  206
```

```
##   No  Yes
## 3623  127
```

```
##
```

```
## Call:
## glm(formula = default ~ balance + income + student, family = binomial,
##     data = train.data.2d)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.4322  -0.1386  -0.0556  -0.0212   3.5324
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.089e+01  6.343e-01 -17.171   <2e-16 ***
## balance      5.679e-03  2.913e-04  19.494   <2e-16 ***
## income       4.346e-06  1.058e-05   0.411    0.681
## studentYes  -4.856e-01  3.010e-01  -1.614    0.107
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1811.07  on 6249  degrees of freedom
## Residual deviance:  980.18  on 6246  degrees of freedom
## AIC: 988.18
##
## Number of Fisher Scoring iterations: 8

##   No  Yes
## 5379  177

##   No  Yes
## 4288  156

##
## Call:
## glm(formula = default ~ balance + income + student, family = binomial,
##     data = train.data.3d)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.3992  -0.1371  -0.0552  -0.0209   3.5556
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.089e+01  6.784e-01 -16.046   <2e-16 ***
## balance      5.676e-03  3.117e-04  18.211   <2e-16 ***
## income       2.667e-06  1.145e-05   0.233    0.816
## studentYes  -4.021e-01  3.255e-01  -1.235    0.217
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1568.36  on 5555  degrees of freedom
## Residual deviance:  849.48  on 5552  degrees of freedom
## AIC: 857.48
##
```

```
## Number of Fisher Scoring iterations: 8
```

Table 4: Validation set Error for clearification

	Error Rate with 1st split	Error Rate with 2nd split	Error Rate with 3rd split
Accuracy	97.241	97.467	97.322
TPR	30.508	32.283	32.051
FPR	0.311	0.248	0.303

Data partitioning was done as in the previous question with the set seed of 16489. With the variables, the data shows that it is statistically significant with both the balance and income. The accuracy of the model is 97.400% that means it has a test error of (100-97.380 =) 2.62%. Likewise, the true positive rate is 33.9% and, the False positive rate is 0.373%. These errors are similar to the error with three splits in the above table. The error rate for this model is 2.62. However, I do not argue that the slight change in the accuracy is because of the addition of the student variable. Hence, for clarification, I have repeated the same process as in question C and got the accuracy mentioned above in the table. The error showed a similar difference in question c with each other. However, if we compare the error with the same split of data the addition of the student variable, there is a slight decrease in the accuracy even it is 0.1. Hence, in conclusion, I can say that the student variable has decreased the accuracy of the model.

3. Question 5.4.7 page 200
   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.

```
## [1] 1089    9
```

```
##
## Call:
## glm(formula = Direction ~ Lag1 + Lag2, family = binomial, data = Weekly)
##
## Deviance Residuals:
##    Min      1Q  Median      3Q     Max
## -1.623  -1.261   1.001   1.083   1.506
##
## 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
```

The model did not pick the Lag1 variable as statistically significant.

b.Fit a logistic regression model that predicts Direction using Lag1 and Lag2 using all but the first observation.

```
##
## Call:
## glm(formula = Direction ~ Lag1 + Lag2, family = binomial, data = Weekly[-1,
##     ])
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.6258  -1.2617   0.9999   1.0819   1.5071
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  0.22324    0.06150   3.630 0.000283 ***
## Lag1        -0.03843    0.02622  -1.466 0.142683
## Lag2         0.06085    0.02656   2.291 0.021971 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1494.6  on 1087  degrees of freedom
## Residual deviance: 1486.5  on 1085  degrees of freedom
## AIC: 1492.5
##
## Number of Fisher Scoring iterations: 4
```

The model did not pick the Lag1 variable as statistically significant as before.

c.Use the model from (b) to predict the direction of the first observation. You can do this by predicting that the first observation will go up if P(Direction="Up"|Lag1, Lag2) > 0.5. Was this observation correctly classified?

```
##    1
## "UP"
```

```
## [1] Down
## Levels: Down Up
```

3c

First observation was classified as Up. It was a misclassification.

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:

e. 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.

(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.

Both question is answered combinedly

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

```
## [1] "Percent Error: 44.995"
```

Part d and e is answered combinedly. The loop was run 1089 times, since there were 1089 observations. 490 was misclassified. The overall error rate was 4.995%.

4. Write your own code (similar to Exercise #3 above) to estimate test error using k-fold cross validation for fitting a linear regression model of the form

$$mpg = \beta_0 + \beta_1 * X_1 + \beta_2 * X_1^2$$

from the **Auto** data in the **ISLR** library, with X_1 = horsepower. Use `echo = T` to show the code. Test this code with `k = 5` and `k = 30`. Discuss the computational trade-off between the two choices of `k`. Do not use the `cv.glm` function.

```
library(ISLR)
data(Auto)
# head(Auto)
AutoN <- dim(Auto)[1]
# k= 30
valueofK <- function(k){
# folds = AutoN/k
stopsoffolds <- c(0,round(c(1:k)*(AutoN/k)))
for (i in 1:k){
  (stopsoffolds[i+1]-stopsoffolds[i])
}
set.seed(16489)
randomizedindex=sample(1:AutoN,AutoN)
testMSE=rep(NA,k)
for(i in 1:k){
  autotestindex=randomizedindex[((stopsoffolds[i]+1):stopsoffolds[i+1])]
  autotrain=Auto[-autotestindex,]
  autotest=Auto[autotestindex,]

  automodel=glm(mpg~horsepower+I(horsepower^2),data=autotrain)
  pred=predict(automodel,newdata=autotest)

  testMSE[i]=sum((pred-autotest$mpg)^2)/dim(autotest)[1]
}
(testMSE)
sum(testMSE)/k
}

MSE_when_k_5 <- valueofK(5)

MSE_when_k_30  <- valueofK(30)



 finaltab <- cbind(MSE_when_k_5, MSE_when_k_30)
 knitr::kable(finaltab, digits = 3,
              caption = "K-fold cross validation with two different K")
```

Table 5: K-fold cross validation with two different K

MSE_when_k_5	MSE_when_k_30
19.108	19.203

Here, 5 and 30 folds were created using seed 16489. There are 392 observations in the dataset. Each time, 1 fold was held out for validation, and the other 5 folds were used to make the model. I have made the function for the calculation of the two different folds. The MSE for these folds is given in the table above. The MSE for 30 folds looks higher than the 5 folds. The MSE for all 5 folds is given as- 17.07849, 17.11979, 19.11285, 22.27647, 19.95182. The MSE of the 5 fold is 19.108 whereas, the MSE of the 30 fold is 19.203. By comparing these folds we can see that K- fold with 30 has comparatively little bias than the f- fold with 5. But, has slightly high variability.