An introduction to statistical learning - Ch5 - Ex5

Thalles Quinaglia Liduares

23/03/2023

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.

Upload package

```
library(ISLR)
```

Upload data

```
Default<-ISLR::Default
```

```
set.seed(123)

# Split data into training and validation sets

train <- sample(nrow(Default), nrow(Default)/2)
train_data <- Default[train,]
valid_data <- Default[-train,]
```

Fit logistic regression model using income and balance as predictors

```
glm.fit <- glm(default ~ income + balance, data=train_data, family="binomial")
summary(glm.fit)
```

```
##
## Call:
## glm(formula = default ~ income + balance, family = "binomial",
##     data = train_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.2810  -0.1348  -0.0529  -0.0185   3.7767
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.194e+01  6.451e-01 -18.504  < 2e-16 ***
## income       2.210e-05  7.381e-06   2.995  0.00275 **
## balance      5.874e-03  3.362e-04  17.474  < 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: 1429.88  on 4999  degrees of freedom
## Residual deviance:  752.69  on 4997  degrees of freedom
## AIC: 758.69
##
## 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:

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

```
set.seed(1234)
train <- sample(nrow(Default), nrow(Default)/2)
train_data <- Default[train,]
valid_data <- Default[-train,]
```

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

```
set.seed(12345)
train <- sample(nrow(Default), nrow(Default)/2)
train_data <- Default[train,]
valid_data <- Default[-train,]
```

Fit logistic regression model using income and balance as predictors

```
glm.fit <- glm(default ~ income + balance, data=train_data, family="binomial")
summary(glm.fit)
```

```
##
## Call:
## glm(formula = default ~ income + balance, family = "binomial",
##     data = train_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5429  -0.1307  -0.0524  -0.0184   3.7461
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.218e+01  6.628e-01 -18.381  < 2e-16 ***
## income       2.998e-05  7.338e-06   4.085  4.4e-05 ***
## balance      5.800e-03  3.399e-04  17.066  < 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: 1402.61  on 4999  degrees of freedom
## Residual deviance:  747.05  on 4997  degrees of freedom
## AIC: 753.05
##
## Number of Fisher Scoring iterations: 8
```

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.

```
set.seed(321)
train <- sample(nrow(Default), nrow(Default)/2)
train_data <- Default[train,]
valid_data <- Default[-train,]

# Fit logistic regression model using income and balance as predictors

glm.fit <- glm(default ~ income + balance, data=train_data, family="binomial")

# Predict default status for each individual in the validation set

pred <- ifelse(predict(glm.fit, newdata=valid_data, type="response") > 0.5, "Yes", "No")
```

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

```
set.seed(1213)
train <- sample(nrow(Default), nrow(Default)/2)
train_data <- Default[train,]
valid_data <- Default[-train,]

# Fit logistic regression model using income and balance as predictors

glm.fit <- glm(default ~ income + balance, data=train_data, family="binomial")

# Predict default status for each individual in the validation set

pred <- ifelse(predict(glm.fit, newdata=valid_data, type="response") > 0.5, "Yes", "No")

# Compute validation set error

error <- mean(pred != valid_data$default)

error
```

```
## [1] 0.0284
```

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.

```
set.seed(123) # set random seed for reproducibility

# Split the data into two different training/validation sets
set1 <- sample(nrow(Default), nrow(Default)/2)
set2 <- sample(setdiff(1:nrow(Default), set1), nrow(Default)/2)
set3 <- setdiff(1:nrow(Default), c(set1, set2))
train_data1 <- Default[set1,]
train_data2 <- Default[set2,]
train_data3 <- Default[set3,]
valid_data1 <- Default[-set1,]
valid_data2 <- Default[-set2,]
valid_data3 <- Default[-set3,]

# Fit logistic regression model and compute validation set errors for each split
glm.fit1 <- glm(default ~ income + balance, data=train_data1, family="binomial")
pred1 <- ifelse(predict(glm.fit1, newdata=valid_data1, type="response") > 0.5, "Yes", "No")
error1 <- mean(pred1 != valid_data1$default)

glm.fit2 <- glm(default ~ income + balance, data=train_data2, family="binomial")
pred2 <- ifelse(predict(glm.fit2, newdata=valid_data2, type="response") > 0.5, "Yes", "No")
error2 <- mean(pred2 != valid_data2$default)



# Print validation set errors
cat("Validation Set Error 1:", error1, "\n")
```

```
## Validation Set Error 1: 0.0276
```

```
cat("Validation Set Error 2:", error2, "\n")
```

```
## Validation Set Error 2: 0.0254
```

```
set.seed(111)

# Split the data into training and validation sets
train_data <- Default[sample(nrow(Default), nrow(Default)/2),]
valid_data <- Default[-sample(nrow(Default), nrow(Default)/2),]

# Fit logistic regression model with income, balance, and student status as predictors
glm.fit <- glm(default ~ income + balance + student, data=train_data, family="binomial")

# Obtain predictions of default status for each individual in the validation set
pred <- ifelse(predict(glm.fit, newdata=valid_data, type="response") > 0.5, "Yes", "No")

# Compute validation set error
error <- mean(pred != valid_data$default)

# Print validation set error
cat("Validation Set Error:", error, "\n")
```

```
## Validation Set Error: 0.0266
```