HW6

Harris Temuri

4/5/2021

Problem 1

The PimaIndiansDiabetes2 [in mlbench package] data is a built in R dataset containing 9 variables and 768 cases. Your task is to use all the other 8 variables to predict the binary dependent variable 'diabetes' telling us whether the subject is diabetic or not (factor with 2 levels: neg and pos). You will split the data into 80% training and 20% testing, using seed = 123.

Solution

(a) Please split the data into 80% training and 20% testing using seed =123.

```
# Problem 1.1
# Split data into 80% training and 20% testing

set.seed(123)

training <- df$diabetes %>%
  createDataPartition(p=0.8, list = FALSE)

trainData <- df[training, ]
testData <- df[-training, ]
```

(b) Then you shall fit a logistic regression model with all the other 8 predictors using the training data.

```
# Problem 1.2
# Logistic Regression Fit

model <- glm(diabetes ~ ., data=trainData, family = binomial)
summary(model)
```

```
##
## Call:
## glm(formula = diabetes ~ ., family = binomial, data = trainData)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -2.5832  -0.6544  -0.3292   0.6248   2.5968
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -1.053e+01  1.440e+00  -7.317 2.54e-13 ***
## pregnant     1.005e-01  6.127e-02   1.640  0.10092
## glucose      3.710e-02  6.486e-03   5.719 1.07e-08 ***
## pressure    -3.876e-04  1.383e-02  -0.028  0.97764
## triceps      1.418e-02  1.998e-02   0.710  0.47800
```

```
## insulin      5.940e-04  1.508e-03   0.394  0.69371
## mass         7.997e-02  3.180e-02   2.515  0.01190 *
## pedigree     1.329e+00  4.823e-01   2.756  0.00585 **
## age          2.718e-02  2.020e-02   1.346  0.17840
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 398.80  on 313  degrees of freedom
## Residual deviance: 267.18  on 305  degrees of freedom
## AIC: 285.18
##
## Number of Fisher Scoring iterations: 5
```

(c) Please use this fitted model based on the training data to predict the response variable 'diabetes' (whether the subject is diabetic or not) for the testing data. Please generate the confusion matrix, and report:

```
# Predictions
probabilities <- model %>% predict(testData, type="response")
predictedClasses <- ifelse(probabilities > 0.5, "pos", "neg")

# Prediction accuracy
mean(predictedClasses == testData$diabetes)
```

```
## [1] 0.7564103
```

```
# Prediction error
mean(predictedClasses != testData$diabetes)
```

```
## [1] 0.2435897
```

```
# Confusion matrix
cm <- confusionMatrix(factor(predictedClasses), testData$diabetes, positive = "pos")
cm
```

```
## Confusion Matrix and Statistics
##
##           Reference
## Prediction neg pos
##        neg  44  11
##        pos   8  15
##
##                Accuracy : 0.7564
##                  95% CI : (0.646, 0.8465)
##     No Information Rate : 0.6667
##     P-Value [Acc > NIR] : 0.05651
##
##                   Kappa : 0.4356
##
##  Mcnemar's Test P-Value : 0.64636
##
##             Sensitivity : 0.5769
##             Specificity : 0.8462
##          Pos Pred Value : 0.6522
##          Neg Pred Value : 0.8000
```

```
##              Prevalence : 0.3333
##          Detection Rate : 0.1923
##    Detection Prevalence : 0.2949
##       Balanced Accuracy : 0.7115
##
##        'Positive' Class : pos
##
```

(i) The overall accuracy;

```
cm$overall[1]
```

```
##  Accuracy
## 0.7564103
```

(ii) The sensitivity (that is, the probability a subject is predicted to be diabetic given that he/she was in fact diabetic);

```
cm$byClass[1]
```

```
## Sensitivity
##   0.5769231
```

(iii) The specificity (that is, the probability a subject is predicted to be not diabetic given that he/she was in fact not diabetic).

```
cm$byClass[2]
```

```
## Specificity
##   0.8461538
```