DaigleHomework4.R

mbair

Tue Oct 2 15:37:54 2018

Chris Daigle

Homework 4

```r
#



# Insurance data: You can find the insurance data from HuskyCT
#
# * Create a categorical variable lowcharge which equals 1 if insurance$charges
# < 7000 and equals 0 otherwise
#
# * Run the logit regression of this on age, sex, bmi, smoker, region
#
# * Split the data by choosing 1000 observations for training and by using the
# other observations for test.
#
# * Assess the accuracy of this model

rm(list = ls())

setwd('~/Git/MachineLearningAndBigDataWithR/Data')
dataName <- 'insurance.csv'
data <- read.csv(dataName, stringsAsFactors = FALSE)
str(data)
```

```
## 'data.frame':    1338 obs. of  7 variables:
##  $ age     : int  19 18 28 33 32 31 46 37 37 60 ...
##  $ sex     : chr  "female" "male" "male" "male" ...
##  $ bmi     : num  27.9 33.8 33 22.7 28.9 ...
##  $ children: int  0 1 3 0 0 0 1 3 2 0 ...
##  $ smoker  : chr  "yes" "no" "no" "no" ...
##  $ region  : chr  "southwest" "southeast" "southeast" "northwest" ...
##  $ charges : num  16885 1726 4449 21984 3867 ...
```

```r
data$lowCharge <- 0
data$lowCharge[data$charges < 7000] <- 1

trainSample <- sample(nrow(data), 1000, replace = F)

trainData <- data[trainSample,]
testData <- data[-trainSample,]
```

```r
glmTrain <-
  glm(lowCharge ~ age + sex + bmi + smoker + region, family = binomial, trainData)
summary(glmTrain)
```

```
##
## Call:
## glm(formula = lowCharge ~ age + sex + bmi + smoker + region,
##     family = binomial, data = trainData)
##
## Deviance Residuals:
##      Min        1Q    Median        3Q       Max
## -3.09351  -0.25434  -0.00004   0.26799   1.84985
##
## Coefficients:
##                   Estimate Std. Error z value Pr(>|z|)
## (Intercept)       7.692946   0.838083   9.179   <2e-16 ***
## age              -0.227917   0.015573 -14.635   <2e-16 ***
## sexmale           0.417487   0.241547   1.728   0.0839 .
## bmi               0.006851   0.021987   0.312   0.7554
## smokeryes       -21.769713 560.263560  -0.039   0.9690
## regionnorthwest   0.584863   0.343823   1.701   0.0889 .
## regionsoutheast   0.619393   0.358326   1.729   0.0839 .
## regionsouthwest   0.776245   0.358066   2.168   0.0302 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 1329.10  on 999  degrees of freedom
## Residual deviance:  459.18  on 992  degrees of freedom
## AIC: 475.18
##
## Number of Fisher Scoring iterations: 18
```

```r
glmProbs <- predict(glmTrain, testData, type = 'response')

glmPred <- rep(0, dim(testData)[1])
glmPred[glmProbs > 0.5] = 1

table(glmPred, testData$lowCharge)
```

```
##
## glmPred   0   1
##       0 205  18
##       1  12 103
```

```r
mean(glmPred == testData$lowCharge)
```

```
## [1] 0.9112426
```

```
# As about a 70:30 train/test split, the model predicts accurately at about 91%.
# This seems pretty good
```