M3 Individual Coding Assignment

Leonard Genders

2025-01-26

### Part A.

**Use the Weekly dataset to perform a logistic regression with Direction as the response and the five lag variables plus Volume as predictors. Use the summary function to print the results.**

# Read in the data  
Weekly <- read.csv('Weekly.csv', stringsAsFactors = T)  
  
# View  
View(Weekly)  
  
# Logistic regression with Direction as the response, five lag variables and Volume  
# as predictors  
glm.fits <- glm(  
 Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 + Volume,  
 data = Weekly,  
 family = binomial # run logisitc regression  
)  
  
# Use the summary() function to print the results.  
summary(glm.fits)

##   
## Call:  
## glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 +   
## Volume, family = binomial, data = Weekly)  
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) 0.26686 0.08593 3.106 0.0019 \*\*  
## Lag1 -0.04127 0.02641 -1.563 0.1181   
## Lag2 0.05844 0.02686 2.175 0.0296 \*   
## Lag3 -0.01606 0.02666 -0.602 0.5469   
## Lag4 -0.02779 0.02646 -1.050 0.2937   
## Lag5 -0.01447 0.02638 -0.549 0.5833   
## Volume -0.02274 0.03690 -0.616 0.5377   
## ---  
## 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: 1486.4 on 1082 degrees of freedom  
## AIC: 1500.4  
##   
## Number of Fisher Scoring iterations: 4

**Do any of the predictors appear to be statistically significant? If so, which ones?**: Lag2 appears to be statistically significant (p = 0.0296), all other predictors are > .05 for logisitc regression with Direction as the response and the five lag variables plus Volume as predictors.

### Part B.

**Compute the confusion matrix and overall fraction of correct predictions. Explain what the confusion matrix is telling you about the types of mistakes made by logistic regression.**

# Using predict() function to predict the probability that the weekly stock   
# market returns will go up given the values of the above predictors  
glm.probs <- predict(glm.fits, type = "response") # type='response' to get probs  
  
# view first 15 results  
glm.probs[1:15] # begins at 0.6086249 for index 1 and 0.6248679 for index 15

## 1 2 3 4 5 6 7 8   
## 0.6086249 0.6010314 0.5875699 0.4816416 0.6169013 0.5684190 0.5786097 0.5151972   
## 9 10 11 12 13 14 15   
## 0.5715200 0.5554287 0.6092096 0.5370125 0.5125263 0.5858984 0.6248679

# confirm assumption that these probabilities correspond to the probability of   
# the weekly stock market returns going up by checking dummy var values  
contrasts(Weekly$Direction) # 0 = down, 1 = up

## Up  
## Down 0  
## Up 1

# convert the probabilities to classes and use .5 as the threshold for   
# class determination, < .5 = 0 (Down) and > .5 = 1 (Up)  
glm.pred <- rep("Down", 1089) # replicate str "Down" 1,250 times  
  
# transforms all elements within glm.probs with probability > .5 to str "Up"  
glm.pred[glm.probs > .5] = "Up"  
  
# use table() function to produce a confusion matrix IOT determine how many obs   
# are correctly or incorrectly classified  
table(glm.pred, Weekly$Direction) # diags indicate correct predictions

##   
## glm.pred Down Up  
## Down 54 48  
## Up 430 557

# 54 classifications correct for Down, 557 correct for Up  
# The model correctly predicted that the weekly stock market returns would go  
# up on 557 days and down on 54 days for a total of 611 correct predictions.  
(557 + 54) / 1089 # = 0.5610652

## [1] 0.5610652

# Computing the overall fraction of correct predictions.  
mean(glm.pred == Weekly$Direction) # = 0.5610652

## [1] 0.5610652

**Explain what the confusion matrix is telling you about the types of mistakes made by logistic regression:** The left side of the confusion matrix represents the predicted class and the top represents the true class. In this problem, there 54 True “Down” (True Negative - TN) weeks, 48 False “Down” (False Negative - FN) weeks, 430 False “Up” (False Positive - FP) weeks, and 537 True “Up” (True Positive - TP) weeks. The total number of negatives (N) is 102 and the total number of positives (P) is 987. The error rate for this example is (430 + 48) / 1,089 = 0.4389 or ~44% for determining the weekly stock market return trend. Of the 605 (48 + 557) times that the stock market returns actually went up, the model correctly predicted ~92% of these instances (557/605) - this correct prediction rate may encourage investors to be more optimistic, so looking at the other side, of the 484 times that the stock market returns actually went down, the model only correctly predicted this ~11% of the time which may be unacceptable in a business practice.

### Part C.

**Now fit the logistic regression model using a training data period from 1990 to 2008, with Lag2 as the only predictor.**

# 1900 to 2008 data for training data period  
train <- (Weekly$Year < 2009)  
  
# make test data for 2009 to 2010  
Weekly.2009 <- Weekly[!train, ] # remove training data from Weekly  
#check dimensions  
dim(Weekly.2009)

## [1] 104 9

Direction.2009 <- Weekly$Direction[!train]  
  
# train the model  
glm.fits <- glm(  
 Direction ~ Lag2,  
 data = Weekly,   
 family = binomial, subset = train)  
  
# test the model  
glm.probs <- predict(glm.fits, Weekly.2009, type = "response")  
  
# compute predictions for 2009-2010  
glm.pred <- rep("Down", 104)  
glm.pred[glm.probs > .5] <- "Up"  
  
# confusion matrix  
table(glm.pred, Direction.2009)

## Direction.2009  
## glm.pred Down Up  
## Down 9 5  
## Up 34 56

mean(glm.pred == Direction.2009) # 0.625

## [1] 0.625

mean(glm.pred != Direction.2009) # 0.375

## [1] 0.375

**Compute the confusion matrix and the overall fraction of correct predictions for the held out data (that is, the data from 2009 and 2010).** In Part C, there 9 TN weeks, 5 FN weeks, 34 FP weeks, and 56 TP weeks. The total number of negatives (N) is 14 and the total number of positives (P) is 90. The error rate for this example is (5 + 34) / 104 = 0.375 or ~38% for determining the weekly stock market return trend. Of the 61 (5 + 56) times that the stock market returns actually went up, the model correctly predicted ~92% of these instances (56/61) - this correct prediction rate may encourage investors to be more optimistic similar to Part B, so looking at the other side, of the 43 times that the stock market returns actually went down, the model only correctly predicted this ~21% of the time which may be unacceptable in a business practice. We can infer that simply guessing has an error rate of 62.5% and our model performs much better with an error rate of 37.5% for Part C. So we can conclude that our model will correctly classify whether weekly stock market returns in the Weekly dataset 62.5% of the time.