Week4\_Assignment

Mohit Mehndiratta

18/08/2021

## Question 1

### (a) - Load and attach the “Default” dataset and view the first few rows. Investigate the structure/class of each variable using str() function.

We will be loading the Default dataset directly from ISLR library.

library(ISLR)  
attach(Default)  
head(Default)

## default student balance income  
## 1 No No 729.5265 44361.625  
## 2 No Yes 817.1804 12106.135  
## 3 No No 1073.5492 31767.139  
## 4 No No 529.2506 35704.494  
## 5 No No 785.6559 38463.496  
## 6 No Yes 919.5885 7491.559

dim(Default)

## [1] 10000 4

str(Default)

## 'data.frame': 10000 obs. of 4 variables:  
## $ default: Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...  
## $ student: Factor w/ 2 levels "No","Yes": 1 2 1 1 1 2 1 2 1 1 ...  
## $ balance: num 730 817 1074 529 786 ...  
## $ income : num 44362 12106 31767 35704 38463 ...

Here we can see that, there are a total of 10,000 observations of 4 variables. Out of which “default” and “student” are categorical/factor variables and “balance”, “income” are numerical variables.

Since, default is the categorical variable we can use it in our logistic regression model.

### (b) - Fit a Logistic Regression Model for default in terms of balance and give the model.

model1 <- glm(default~balance, data = Default, family = binomial)  
summary(model1)

##   
## Call:  
## glm(formula = default ~ balance, family = binomial, data = Default)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.2697 -0.1465 -0.0589 -0.0221 3.7589   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.065e+01 3.612e-01 -29.49 <2e-16 \*\*\*  
## balance 5.499e-03 2.204e-04 24.95 <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: 2920.6 on 9999 degrees of freedom  
## Residual deviance: 1596.5 on 9998 degrees of freedom  
## AIC: 1600.5  
##   
## Number of Fisher Scoring iterations: 8

The model1 above, is our logistic regression model for default in terms of balance. Upon looking at the summary we can see that value of the **intercept** is **-10.65** and the **slope** is **0.005499**. Therefore our equation for this model will be:

=

After replacing the intercept and slope value w.r.t to default vs balance model:

=

This is the probability that a person with the balance X would be in default.

### (c) - Give the probability that a person with balance 500 will be in default.

So Given that,

balance = 500, by replacing this in our model equation we can find out the probability.

balance = 500  
t = -10.65 + (0.005499 \* balance) # this is our exponential's superscript.  
# probability  
probabilityOfDefault = exp(t) / (1 + exp(t))  
probabilityOfDefault

## [1] 0.000370421

From the output, we can see that **0.000370421** is the probability of a person with balance 500 to be in a default.

### (d) - Fit a Logistic Regression Model for default in terms of all the three predictors.

model2 <- glm(default~balance+student+income, data = Default, family = binomial)  
summary(model2)

##   
## Call:  
## glm(formula = default ~ balance + student + income, family = binomial,   
## data = Default)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.4691 -0.1418 -0.0557 -0.0203 3.7383   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -1.087e+01 4.923e-01 -22.080 < 2e-16 \*\*\*  
## balance 5.737e-03 2.319e-04 24.738 < 2e-16 \*\*\*  
## studentYes -6.468e-01 2.363e-01 -2.738 0.00619 \*\*   
## income 3.033e-06 8.203e-06 0.370 0.71152   
## ---  
## 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: 1571.5 on 9996 degrees of freedom  
## AIC: 1579.5  
##   
## Number of Fisher Scoring iterations: 8

## Question 2

### (a) Import the “Heart” dataset and view the first few rows. Investigate the structure/class of each variable.

heartDS <- read.csv("../datasets/heart.csv")  
attach(heartDS)  
head(heartDS)

## X Age Sex ChestPain RestBP Chol Fbs RestECG MaxHR ExAng Oldpeak Slope Ca  
## 1 1 63 1 typical 145 233 1 2 150 0 2.3 3 0  
## 2 2 67 1 asymptomatic 160 286 0 2 108 1 1.5 2 3  
## 3 3 67 1 asymptomatic 120 229 0 2 129 1 2.6 2 2  
## 4 4 37 1 nonanginal 130 250 0 0 187 0 3.5 3 0  
## 5 5 41 0 nontypical 130 204 0 2 172 0 1.4 1 0  
## 6 6 56 1 nontypical 120 236 0 0 178 0 0.8 1 0  
## Thal AHD  
## 1 fixed 0  
## 2 normal 1  
## 3 reversable 1  
## 4 normal 0  
## 5 normal 0  
## 6 normal 0

dim(heartDS)

## [1] 303 15

str(heartDS)

## 'data.frame': 303 obs. of 15 variables:  
## $ X : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ Age : int 63 67 67 37 41 56 62 57 63 53 ...  
## $ Sex : int 1 1 1 1 0 1 0 0 1 1 ...  
## $ ChestPain: chr "typical" "asymptomatic" "asymptomatic" "nonanginal" ...  
## $ RestBP : int 145 160 120 130 130 120 140 120 130 140 ...  
## $ Chol : int 233 286 229 250 204 236 268 354 254 203 ...  
## $ Fbs : int 1 0 0 0 0 0 0 0 0 1 ...  
## $ RestECG : int 2 2 2 0 2 0 2 0 2 2 ...  
## $ MaxHR : int 150 108 129 187 172 178 160 163 147 155 ...  
## $ ExAng : int 0 1 1 0 0 0 0 1 0 1 ...  
## $ Oldpeak : num 2.3 1.5 2.6 3.5 1.4 0.8 3.6 0.6 1.4 3.1 ...  
## $ Slope : int 3 2 2 3 1 1 3 1 2 3 ...  
## $ Ca : int 0 3 2 0 0 0 2 0 1 0 ...  
## $ Thal : chr "fixed" "normal" "reversable" "normal" ...  
## $ AHD : int 0 1 1 0 0 0 1 0 1 1 ...

By looking at the output we can see that heart data set consists of 303 observations of 15 variables.

We can see that our target variable “AHD” is the integer variable to we need to make it a factor variable because this a classification model.

heartDS$AHD = as.factor(heartDS$AHD)  
str(heartDS)

## 'data.frame': 303 obs. of 15 variables:  
## $ X : int 1 2 3 4 5 6 7 8 9 10 ...  
## $ Age : int 63 67 67 37 41 56 62 57 63 53 ...  
## $ Sex : int 1 1 1 1 0 1 0 0 1 1 ...  
## $ ChestPain: chr "typical" "asymptomatic" "asymptomatic" "nonanginal" ...  
## $ RestBP : int 145 160 120 130 130 120 140 120 130 140 ...  
## $ Chol : int 233 286 229 250 204 236 268 354 254 203 ...  
## $ Fbs : int 1 0 0 0 0 0 0 0 0 1 ...  
## $ RestECG : int 2 2 2 0 2 0 2 0 2 2 ...  
## $ MaxHR : int 150 108 129 187 172 178 160 163 147 155 ...  
## $ ExAng : int 0 1 1 0 0 0 0 1 0 1 ...  
## $ Oldpeak : num 2.3 1.5 2.6 3.5 1.4 0.8 3.6 0.6 1.4 3.1 ...  
## $ Slope : int 3 2 2 3 1 1 3 1 2 3 ...  
## $ Ca : int 0 3 2 0 0 0 2 0 1 0 ...  
## $ Thal : chr "fixed" "normal" "reversable" "normal" ...  
## $ AHD : Factor w/ 2 levels "0","1": 1 2 2 1 1 1 2 1 2 2 ...

So now we can see that AHD is a factor variable now.

### (b) Fit a Logistic Regression Model for AHD in terms of Sex, ChestPain and Ca.

model3 <- glm(AHD~Sex+ChestPain+Ca, data = heartDS, family = binomial)  
summary(model3)

##   
## Call:  
## glm(formula = AHD ~ Sex + ChestPain + Ca, family = binomial,   
## data = heartDS)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -2.8837 -0.6232 -0.3069 0.5411 2.0180   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)   
## (Intercept) -0.7915 0.3355 -2.359 0.018317 \*   
## Sex 1.4917 0.3527 4.229 2.34e-05 \*\*\*  
## ChestPainnonanginal -2.2402 0.3835 -5.841 5.19e-09 \*\*\*  
## ChestPainnontypical -2.2521 0.4518 -4.985 6.21e-07 \*\*\*  
## ChestPaintypical -1.9776 0.5556 -3.559 0.000372 \*\*\*  
## Ca 1.1473 0.2005 5.723 1.05e-08 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 412.73 on 298 degrees of freedom  
## Residual deviance: 263.02 on 293 degrees of freedom  
## (4 observations deleted due to missingness)  
## AIC: 275.02  
##   
## Number of Fisher Scoring iterations: 5

## Question 3

### (a) Load and attach the “Smarket” dataset and view the first few rows. Investigate the structure/class of each variable.

attach(Smarket)  
head(Smarket)

## Year Lag1 Lag2 Lag3 Lag4 Lag5 Volume Today Direction  
## 1 2001 0.381 -0.192 -2.624 -1.055 5.010 1.1913 0.959 Up  
## 2 2001 0.959 0.381 -0.192 -2.624 -1.055 1.2965 1.032 Up  
## 3 2001 1.032 0.959 0.381 -0.192 -2.624 1.4112 -0.623 Down  
## 4 2001 -0.623 1.032 0.959 0.381 -0.192 1.2760 0.614 Up  
## 5 2001 0.614 -0.623 1.032 0.959 0.381 1.2057 0.213 Up  
## 6 2001 0.213 0.614 -0.623 1.032 0.959 1.3491 1.392 Up

dim(Smarket)

## [1] 1250 9

str(Smarket)

## 'data.frame': 1250 obs. of 9 variables:  
## $ Year : num 2001 2001 2001 2001 2001 ...  
## $ Lag1 : num 0.381 0.959 1.032 -0.623 0.614 ...  
## $ Lag2 : num -0.192 0.381 0.959 1.032 -0.623 ...  
## $ Lag3 : num -2.624 -0.192 0.381 0.959 1.032 ...  
## $ Lag4 : num -1.055 -2.624 -0.192 0.381 0.959 ...  
## $ Lag5 : num 5.01 -1.055 -2.624 -0.192 0.381 ...  
## $ Volume : num 1.19 1.3 1.41 1.28 1.21 ...  
## $ Today : num 0.959 1.032 -0.623 0.614 0.213 ...  
## $ Direction: Factor w/ 2 levels "Down","Up": 2 2 1 2 2 2 1 2 2 2 ...

Here we can see that, there are a total of 1250 observations of 9 variables.

Since, Direction is the categorical variable we can use it in our logistic regression model as a target variable.

### (b) Fit a Logistic Regression Model for Direction in terms of the Predictors Lag1, Lag2, Lag3, Lag4, Lag5 and Volume.

model4 <- glm(Direction~Lag1+Lag2+Lag3+Lag4+Lag5+Volume, data = Smarket, family = binomial)  
summary(model4)

##   
## Call:  
## glm(formula = Direction ~ Lag1 + Lag2 + Lag3 + Lag4 + Lag5 +   
## Volume, family = binomial, data = Smarket)  
##   
## Deviance Residuals:   
## Min 1Q Median 3Q Max   
## -1.446 -1.203 1.065 1.145 1.326   
##   
## Coefficients:  
## Estimate Std. Error z value Pr(>|z|)  
## (Intercept) -0.126000 0.240736 -0.523 0.601  
## Lag1 -0.073074 0.050167 -1.457 0.145  
## Lag2 -0.042301 0.050086 -0.845 0.398  
## Lag3 0.011085 0.049939 0.222 0.824  
## Lag4 0.009359 0.049974 0.187 0.851  
## Lag5 0.010313 0.049511 0.208 0.835  
## Volume 0.135441 0.158360 0.855 0.392  
##   
## (Dispersion parameter for binomial family taken to be 1)  
##   
## Null deviance: 1731.2 on 1249 degrees of freedom  
## Residual deviance: 1727.6 on 1243 degrees of freedom  
## AIC: 1741.6  
##   
## Number of Fisher Scoring iterations: 3

### (c) Explain how you would calculate the probability (the Direction = “up”) for a given set of X (predictor) values.

So to find the probability for Direction “Up”, we need to find out that if our model has dummy variable for the same. To do so, we can use contrasts function

contrasts(Direction)

## Up  
## Down 0  
## Up 1

Here, we can confirm that R has a dummy variable with 1 for “Up”.

Now, we can check that coefficients of our model and create our model equation to find the probability of Direction “Up”.

coef(model4)

## (Intercept) Lag1 Lag2 Lag3 Lag4 Lag5   
## -0.126000257 -0.073073746 -0.042301344 0.011085108 0.009358938 0.010313068   
## Volume   
## 0.135440659

Now substituting the value of coefficients into our equation for the model,

=

By using the above equation we can calculate the probability of Direction “Up”.

### (d) Construct the Misclassification Martrix

for this we need to create a prediction class for the misclassification matrix first. For prediction class we first need to make predictions.

predicted\_prob <- predict(model4, type = "response")

From above we can see all the predictions of our model. Now we need to make a prediction class out of it for Up and Down. We will create a prediction class with the condition that if a probability is greater than 0.5 then we will take it as “Up”.

predicted\_class <- rep("Down", nrow(Smarket))  
predicted\_class[predicted\_prob > 0.5] = "Up"  
  
table(predicted\_class, Direction)

## Direction  
## predicted\_class Down Up  
## Down 145 141  
## Up 457 507

From the above table we can see that this is our misclassification matrix.

### (e) Calculate the Prob(Misclassification) or Misclassification Rate

Misclassification rate can be calculated as the number of items falsely predicted by our model prediction. From above table we can see that we falsely predicted **141** as **Down** and **457** as **Up**.

Therefore, using this misclassification rate would be:

misclassificationRate <- (141 + 457) / (145 + 141 + 457 + 507)  
misclassificationRate

## [1] 0.4784

From this we can find out that the misclassification rate is 47.84%.

### (f) Calculate the False Positive rate

From the misclassification matrix, false positive is 457. That is 457 items predicted as Up while in actual they were Down.

Therefore, using this false positive rate would be:

falsePositiveRate <- 457 / (145 + 457)  
falsePositiveRate

## [1] 0.7591362

From above output, we can see that false positive rate is 75.91%

### (g) Calculate the False Negative rate

From the misclassification matrix, false negative is 141. That is 141 items predicted as Down while in actual they were Up

Therefore, using this false negative rate would be:

falseNegativeRate <- 141 / (141 + 507)  
falseNegativeRate

## [1] 0.2175926

From above output, we can see that false negative rate is 21.75%