Assignment 3

Linear Regression

Jan Izquierdo, Laura Llorente

2023-10-31

A researcher is interested in how variables, such as GRE (Graduate Record Exam scores), GPA (grade point average), prestige of the undergraduate institution, gender and month of birth affect admission into graduate school. The response variable, admitted or non-admitted is a binary variable (1: admitted, 0: Not). The dataset has a binary response (outcome, dependent) variable called admit. There are five predictor variables: gre, gpa, rank, gender and month. We will treat the variables gre, gpa and month as continuous. The variable rank takes on the values 1 through 4; Institutions with a rank of 1 have the highest prestige, while those with a rank of 4 have the lowest. The gender takes values 0 and 1 to indicate male and female, respectively. We will apply logistic regression to a data set of predictor variables on admission. For 10 individuals in the database the admission information is missing. Here we study the relationship between "admit" and the other variables by means of logistic regression, to see if we can predict the admission on the basis of the predictor variables. "Order" is not a predictor variable, is just used as identifier of the row position. .

Setting libraries

```
#install.packages("ggplot2")
#install.packages("readxl")
#install.packages("lmtest")
library(readxl)
library(lmtest)
library(ggplot2)
```

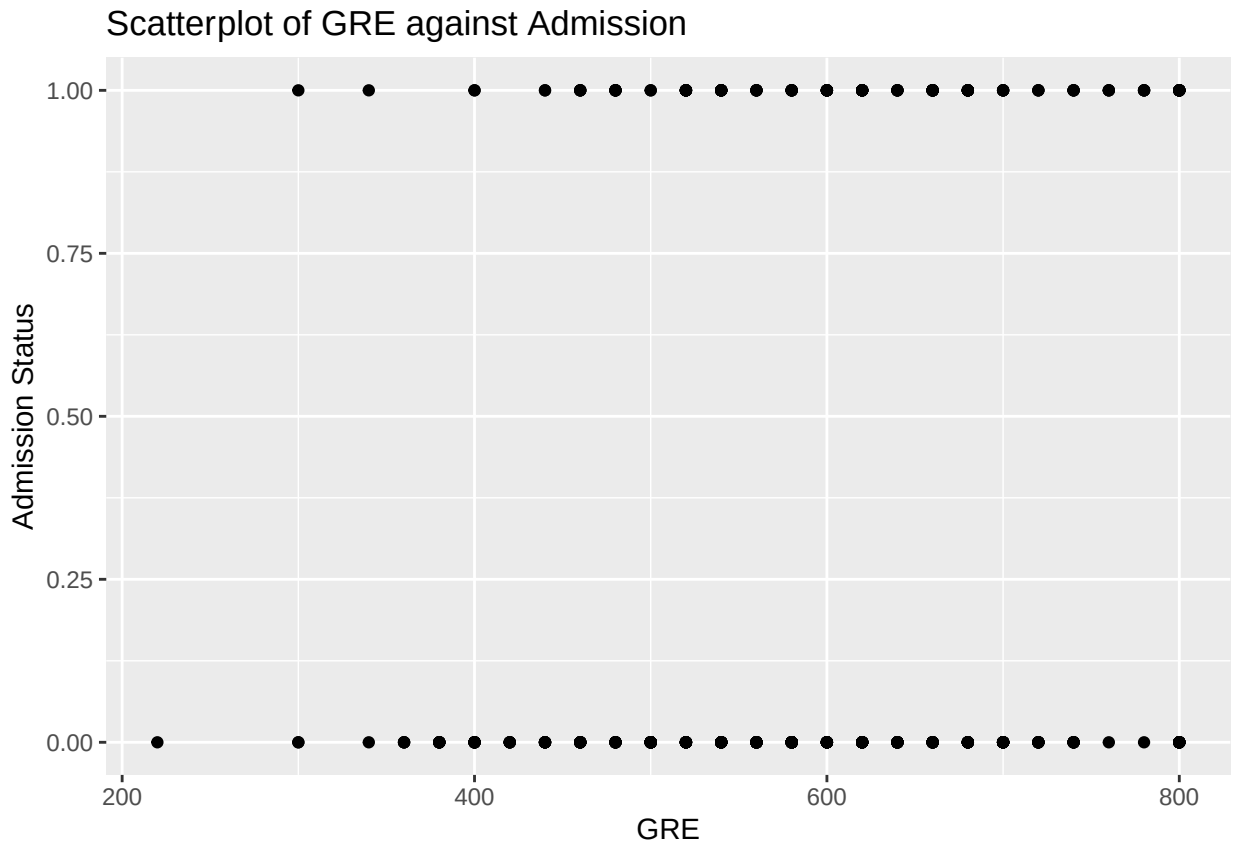
1 **Read the file binary_v2_missing.xlsx into the R environment.** .

```
data <- read_excel("./assignment_data/binary_v2_missing.xlsx")
```

2 **Make a scatterplot of gre against admit. Do you think there is any relationship between the two variables? What is the sample size? Can you think of alternative ways to better visualize the relationship between the variables?** .

```
ggplot(data, aes(x = gre, y = admit)) +
  geom_point() +
  labs(title = "Scatterplot of GRE against Admission",
       x = "GRE",
       y = "Admission Status")
```

```
## Warning: Removed 10 rows containing missing values (geom_point).
```




```
#Sample size
n<-nrow(data)
logistic_model <- lm(admit ~ gre, data = data)
```

```
## We think that there is no visible relationship between the variables. An alternative way would be
## to do an histogram, or other plots and using linear models between both variables
```

```
## Sample size is 400
```

***3* Do a logistic regression of admit on gre. Write down the estimated logit.** .

```
logistic_model <- glm(admit ~ gre, data = data, family = binomial(link= "logit"))
co<-coef(logistic_model)
```

```
## The summary of the logistic regression is
```

```
##
## Call:
## glm(formula = admit ~ gre, family = binomial(link = "logit"),
##     data = data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
```

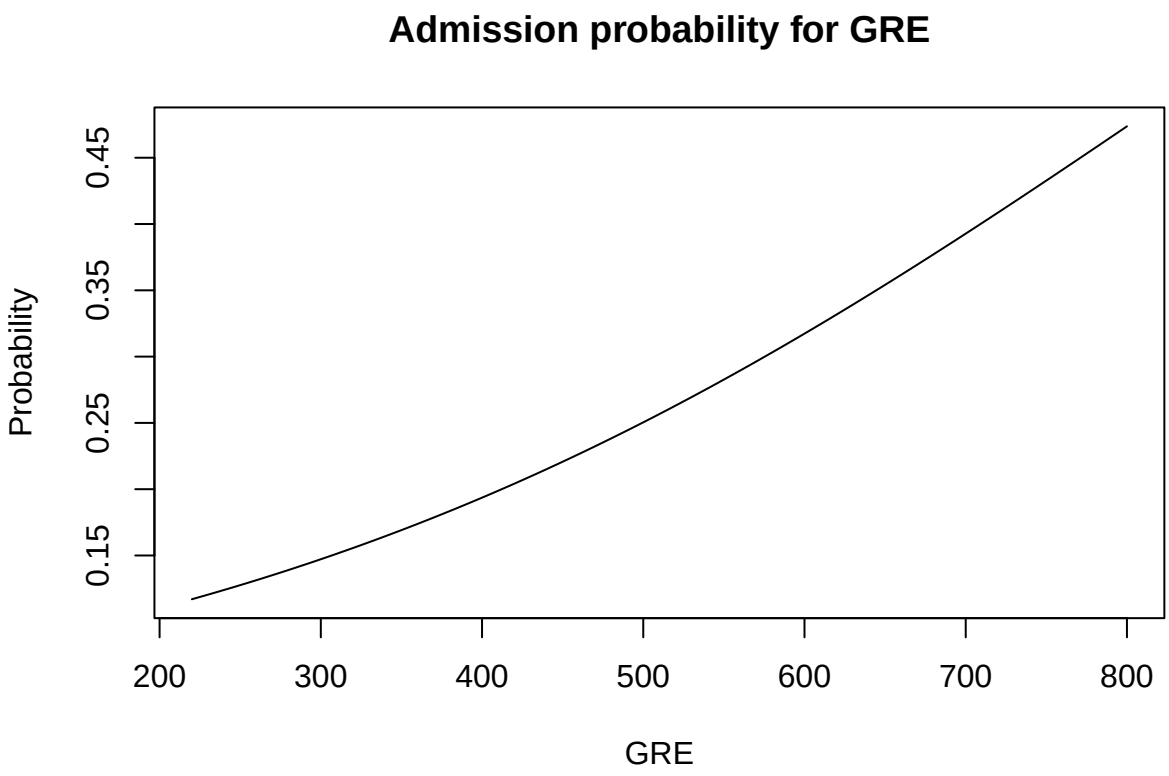
```
## -1.1330  -0.8980  -0.7593   1.3598   1.9576
##
## Coefficients:
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept) -2.747876   0.615465  -4.465 8.02e-06 ***
## gre          0.003303   0.001001   3.300 0.000967 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 486.21  on 389  degrees of freedom
## Residual deviance: 474.82  on 388  degrees of freedom
##   (10 observations deleted due to missingness)
## AIC: 478.82
##
## Number of Fisher Scoring iterations: 4
```

```
## The estimated logit
```

```
## 1.003309 gle
```

***4* How does gre affects the response variable? Interpret the regression coefficient for gre. Apply transformations as you consider adequate.** .

```
ratio<-exp(coef(logistic_model))
#possibles values for gre
gre_val<- data.frame(gre=seq(max(data$gre), min(data$gre)))
gre_val$prob<-predict(logistic_model, newdata = gre_val, type = "response")
plot(gre_val, type="l", main="Admission probability for GRE", xlab="GRE", ylab="Probability")
```




```
## We can deduct that higher GRE scores lead to a higher likelihood of admission
```

5 **Is gre a significant predictor (use a = 0.05)? Perform the corresponding hypothesis test and report the test-statistic, its reference distribution and the p-value. Is gre a significant predictor if we use a = 0.1?** .

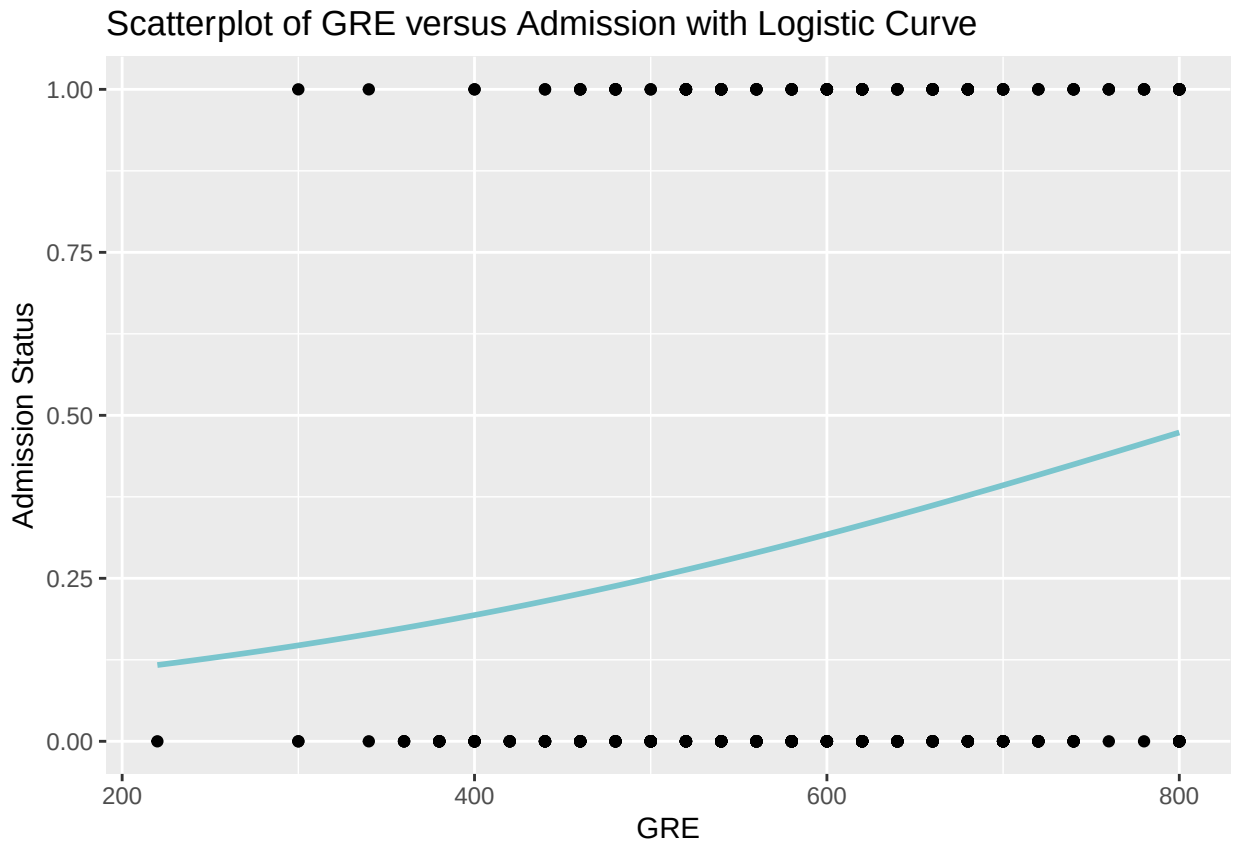
```
coef_gre <- coef(logistic_model)["gre"]
se_gre <- summary(logistic_model)$coefficients["gre", "Std. Error"]
wald_stat <- coef_gre / se_gre
p_val <- 2 * (1 - pnorm(abs(wald_stat)))
```

```
## Since the p-value is 0.0009668648 gre is a significant predictor in both alpha 0.05 and 0.1
```

6 **Make a scatter plot of gre versus admit and plot the logistic curve in the scatter plot.** .

```
# Scatterplot with logistic curve
ggplot(data, aes(x = gre, y = admit)) +
  geom_point() +
  geom_smooth(method = "glm", method.args = list(family = "binomial"), se = FALSE, color
  = "cadetblue3") +
  labs(title = "Scatterplot of GRE versus Admission with Logistic Curve",
       x = "GRE",
       y = "Admission Status")
```

```
## `geom_smooth()` using formula 'y ~ x'

## Warning: Removed 10 rows containing non-finite values (stat_smooth).

## Warning: Removed 10 rows containing missing values (geom_point).
```



7 Give a 95% confidence interval for gre using the estimated logit. Exponentiate the limits of this interval and give your interpretation of the result. .

```
conf_lvl<-0.05
a<-0.05
z <- qnorm(1 - a / 2)
#the estimated logit that we are going to use is coef_gre
lim_det<-z*se_gre
lower_bound <- exp(coef_gre - lim_det)
upper_bound <- exp(coef_gre + lim_det)
```

```
## Confidence interval of 95% for gre using the exponentiated limits of the previously estimated logit
##  [ 1.001342 , 1.005279 ]
```

***8* The admission (variable admit) of ten individuals is unknown. Use the predict function to predict the logit for these ten individuals, using the model with gre as the sole predictor. Also predict the probability of being admitted or not for each individual. How would you classify these ten individuals?** .

```r
missing_admission_data <- data[is.na(data$admit), ]

# Predict logit for the 10 individuals
logit_predictions <- predict(logistic_model, newdata = missing_admission_data, type = "link")

# probability of admission for the 10 individuals
probability_predictions <- predict(logistic_model, newdata = missing_admission_data, type = "response")
result <- data.frame(GRE = missing_admission_data$gre, Logit = logit_predictions, Probability =
probability_predictions)
threshold <- 0.5

# Classification of individuals based on the threshold
classification <- ifelse(probability_predictions >= threshold, 1, 0)
result$Classification <- classification
print(result)
```

```
##    GRE      Logit Probability Classification
## 1  760 -0.2375440   0.4408917              0
## 2  480 -1.1624032   0.2382309              0
## 3  340 -1.6248329   0.1645394              0
## 4  420 -1.3605874   0.2041449              0
## 5  660 -0.5678509   0.3617329              0
## 6  340 -1.6248329   0.1645394              0
## 7  420 -1.3605874   0.2041449              0
## 8  620 -0.6999736   0.3318181              0
## 9  560 -0.8981577   0.2894292              0
## 10 620 -0.6999736   0.3318181              0
```

```
## As we can see in the result, the classification is that none of the ten individuals would be
## admitted
```

9 **Calculate McFadden's pseudo R^2 for the logistic regression of admit on gre. What does a zero value for this pseudo R^2 mean?** .

```r
log_likelihood_model <- logLik(logistic_model)
log_likelihood_null <- logLik(glm(admit ~ 1, data = data, family = binomial(link= "logit")))

# McFadden's pseudo R^2
mcfadden_r2 <- 1 - (log_likelihood_model / log_likelihood_null)
cat("McFadden's Pseudo R^2:", mcfadden_r2, "\n")
```

```
## McFadden's Pseudo R^2: 0.02342857
```

```
## McFadden's Pseudo R^2: 0.02342857
```

```
## McFadden's pseudo R^2 ranges from 0 to 1. A value close to 1 represents a good fit of the model
## while a value close to 0 indicates that the model does not improve much upon a null model
```

10 **Calculate the median of the gpa and create an indicator variable medgpa that registers if the gpa is less than or equal to, or above the median. Report how many individuals you have in each category.** .

```r
m<-median(data$gpa)
data$medgpa<-"-"
c=1
for(i in data$gpa){
  if(i>m){data$medgpa[c]=1}
  else {data$medgpa[c]=0}
  c=c+1
};c=1
```

```
## There are 200 individuals with a gpa avobe the median
```

```
## There are 200 individuals with a gpa below the median
```

```
## There are 0 individuals with a gpa equal to the median
```

11 **Perform the logistic regression of admit on medgpa. Report the estimated logit. What is the interpretation of the exponentiated slope of this regression?** .

```r
logistic_model_mgpa<- glm(admit ~ medgpa, data = data, family = binomial)
co_mgpa<-coef(logistic_model_mgpa)
```

```
## The estimated logit of admit on medgpa is
```

```
## (Intercept)     medgpa1
##  -1.2560797   0.8845162
```

12 **Perform the logistic regression of admit on gpa. Report the estimated logit. Are the results consistent with the previous analysis? Would you prefer the analysis with medgpa over the analysis with gpa? Justify your answer.** .

```r
logistic_model_gpa<- glm(admit ~ gpa, data = data, family = binomial)
co_gpa<-coef(logistic_model_gpa)
```

```
## The estimated logit of admit on gpa is 1.053991
```

13 **Make a logistic regression model with all available predictors, except the indicator medgpa you created. Determine, by doing a hypothesis test, whether the model is globally significant. Report the test statistic, its reference distribution and the p-value.** .

```r
logistic_model_all<-glm(admit ~ gre+gpa+rank+gender+month+order, data=data, family="binomial")
null_logistic_model_all<-glm(admit ~ 1, data = data, family = "binomial")

all_anova_test<-anova(null_logistic_model_all,logistic_model_all, test="Chisq")
all_test_statistics<-all_anova_test$Deviance[2]

all_pvalues<-all_anova_test$`Pr(>Chi)`[2]
```

```
## ANOVA results
```

```
## Analysis of Deviance Table
##
## Model 1: admit ~ 1
## Model 2: admit ~ gre + gpa + rank + gender + month + order
##   Resid. Df Resid. Dev Df Deviance  Pr(>Chi)
## 1       389     486.21
## 2       383     447.02  6   39.191 6.567e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
## The test statistic result is 39.1908

## We used Chisquare distribution as a refernce statistic

## The p-value of this model is 6.566871e-07

## Which makes this distribution significant, as 0.05> 6.566871e-07
```

14 **Try to simplify the model be eliminating non-significant predictors (one by one, use a = 0.05). What is your final model?** .

```
res <- summary(logistic_model_all)$coef[,"Pr(>|z|)"]
significance <- summary(logistic_model_all)$coef[,"Pr(>|z|)"] <= 0.05
```

```
## The only significant variables are rank and gpa, as they are the only ones with pvalue below
## the critical value of the model

## (Intercept)         gre         gpa        rank      gender       month
##        TRUE       FALSE        TRUE        TRUE       FALSE       FALSE
##       order
##       FALSE
```

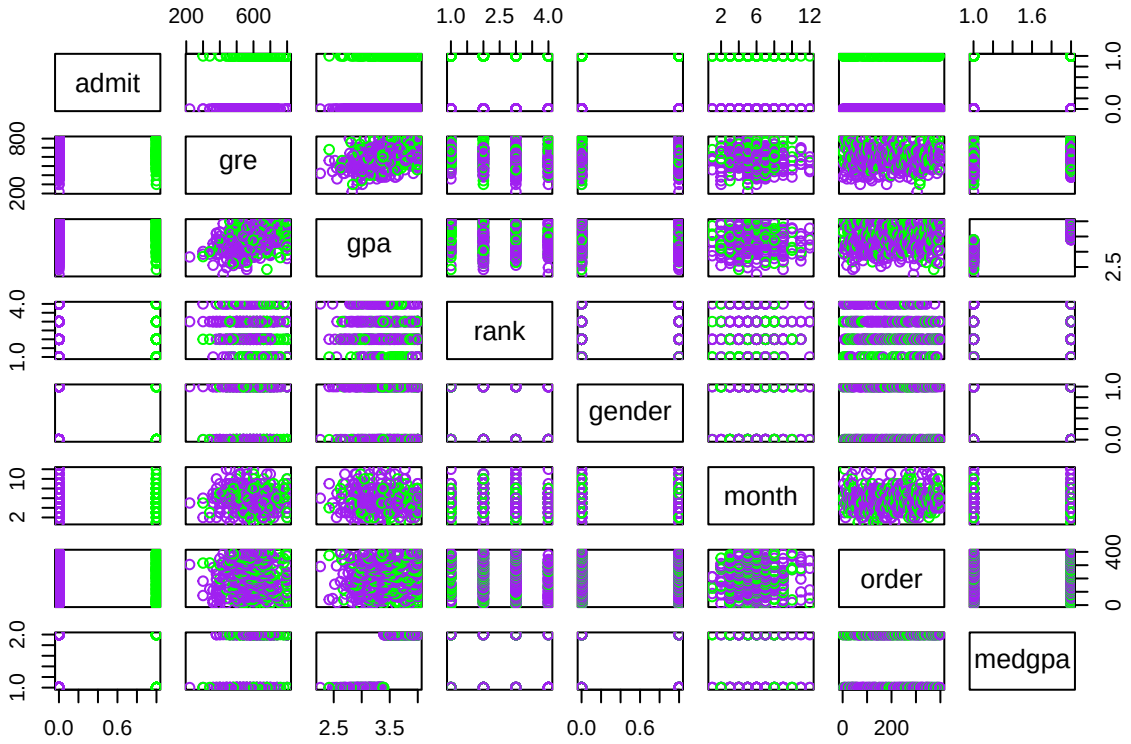
15 **Test by a likelihood ratio test if we can consider the joint nullity of the coefficients of all variables that you have eliminated. Report the p-value of this test.** .

```
reduced_model <- glm(admit ~ gpa + rank, data = data, family = binomial)

lr_test <- lrtest(logistic_model_all, reduced_model)

print(lr_test)
```

```
## Likelihood ratio test
##
## Model 1: admit ~ gre + gpa + rank + gender + month + order
## Model 2: admit ~ gpa + rank
##   #Df  LogLik Df  Chisq Pr(>Chisq)
## 1   7 -223.51
## 2   3 -225.07 -4 3.1176     0.5383

## As the p-value, 0.5383371 , is lower than 0.05 we fail to reject the null hypothesis of joint
## nullity for the coefficients of the eliminated variables (month, gender, and gre)
```

16 **Make a scatter plot matrix with the pairs instruction of using the variables in your final model, distinguishing the two varieties with a different symbol or color. Do you see a good graphical separation of the two groups?** .

```
plot(data, col=ifelse(data$admit==1, "green", "purple"))
```




```
## There is a good separation between the 2 groups in most plots of the matrix
```

17 **Predict the admission of the individuals in the database with your final model, and make a cross table of predicted admit against observed admit.** .

```
final_model_variables <- c("gpa", "rank")

data_for_prediction <- data[c("admit", final_model_variables)]
data_for_prediction$predicted_admit <- predict(reduced_model, newdata = data_for_prediction, type =
"response")

data_for_prediction$predicted_admit_binary <- ifelse(data_for_prediction$predicted_admit > 0.5, 1, 0)

cross_table <- table(data_for_prediction$predicted_admit_binary, data_for_prediction$admit)
print(cross_table)
```

```
##
##       0   1
```

```
##   0 255  94
##   1  12  29
```

```
## 255 is the the number of individuals that won't be admitted and we have predicted that they
## won't (correct prediction)
## 94 is the number of individuals that will be admitted that we haven't accounted for in our
## prediction (incorrect prediction)
## 12 is the number of individuals that we won't be admitted but we have wrongly predicted that will be
## admitted (incorrect prediction)
## 29 is the number of indiciduals that will be admitted and will be accounted by our prediction
## (correct prediction)
```

***18* How often does the model correctly predict the observed admit? Calculate the classification rate, defined as the number of correct classifications divided by the total of classifications made.**

.

```
accuracy <- (255 + 29) / sum(c(255, 94, 12, 29))
print(paste("Classification Rate (Accuracy):", accuracy))
```

```
## [1] "Classification Rate (Accuracy): 0.728205128205128"
```

***19* Also calculate the classification rate for predictions made with the full model, which contains all predictors. Does this work better? Justify your answer.** .

```
full_model_variables <- c("gre", "gpa", "rank", "gender", "month")
data_full_model <- data[c("admit", full_model_variables)]
data_full_model$predicted_admit <- predict(logistic_model, newdata = data_full_model, type =
"response")
data_full_model$predicted_admit_binary <- ifelse(data_full_model$predicted_admit > 0.5, 1, 0)

cross_table_full_model <- table(data_full_model$predicted_admit_binary, data_full_model$admit)

accuracy_full_model <- sum(diag(cross_table_full_model)) / sum(cross_table_full_model)
```

```
## [1] "Classification Rate (Accuracy) for Full Model: 0.684615384615385"
```

```
## This does not work better, as the accuracy for the reduced model is bigger than the one for the
## full model 0.7282051 > 0.6846154
```