Logistic_regression

Eira Fontanals Muñoz and Ricard Garcia Isern

```
library(readxl)
binary_data<-read_excel("binary_v2_missing.xlsx")
```

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

```
sample_size <- nrow(binary_data)
cat("Sample size:", sample_size)
```

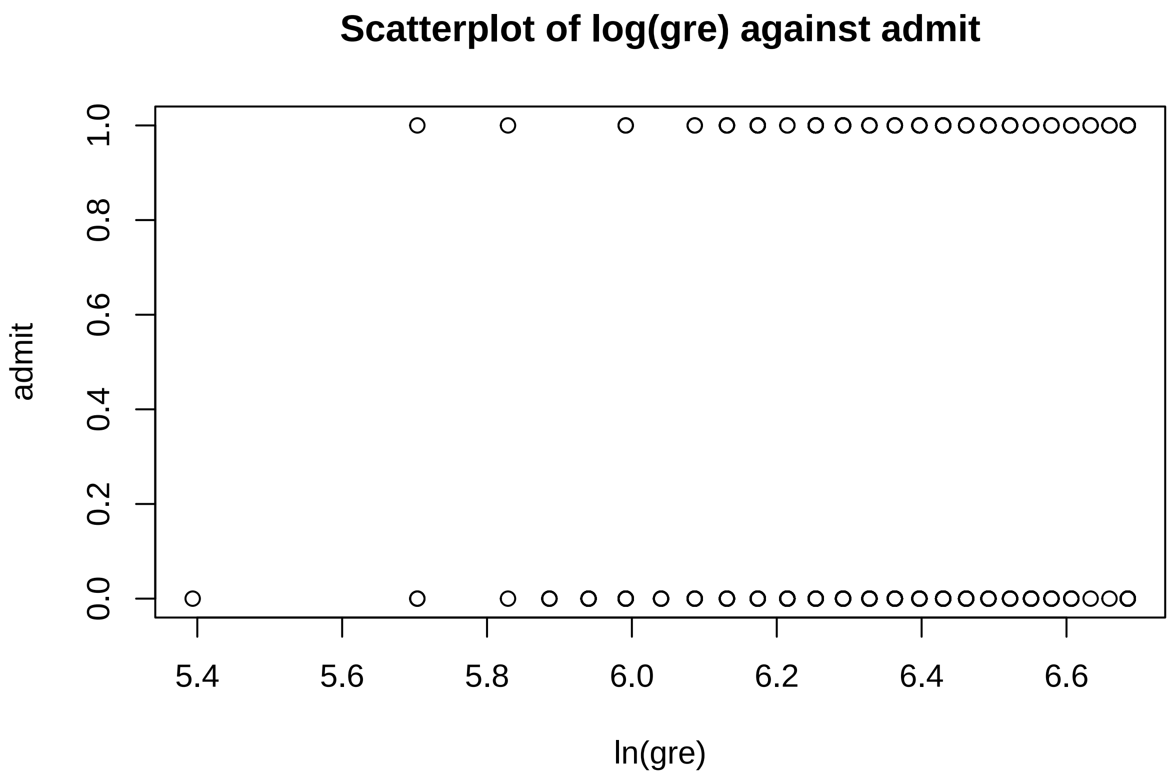
2. Make a scatter plot of gre against admit. Do you think there is any relationship between the two variables? What is the sample size? Can you think of alternative ways to better visualize the relationship between the variables?

```
## Sample size: 400
```

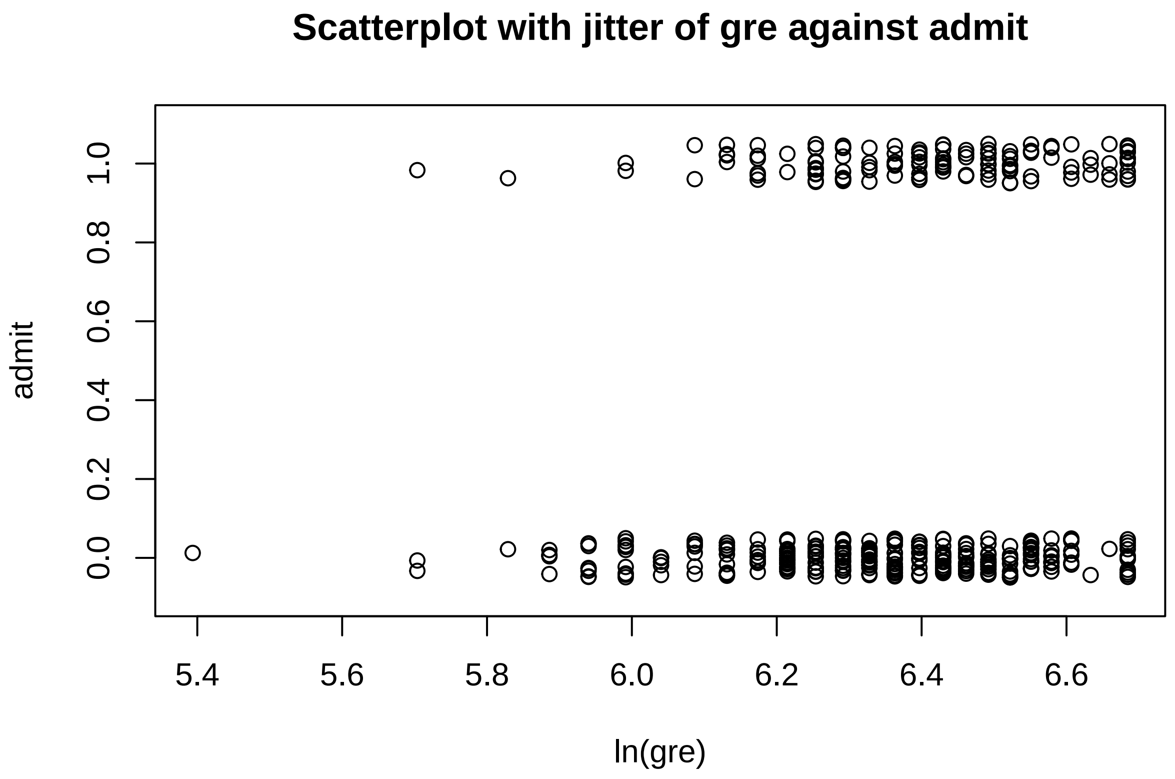
```
#Delete the rows that have 'NA' in our response variable which changes the sample size
rowwithmissing <- function(x) {
  any(is.na(x))
}
ind <- apply(binary_data,1,rowwithmissing)
binary_data <- binary_data[!ind,]
sample_size_no_NA <- nrow(binary_data)
cat("Sample size without NA:", sample_size_no_NA)
```

```
## Sample size without NA: 390
```

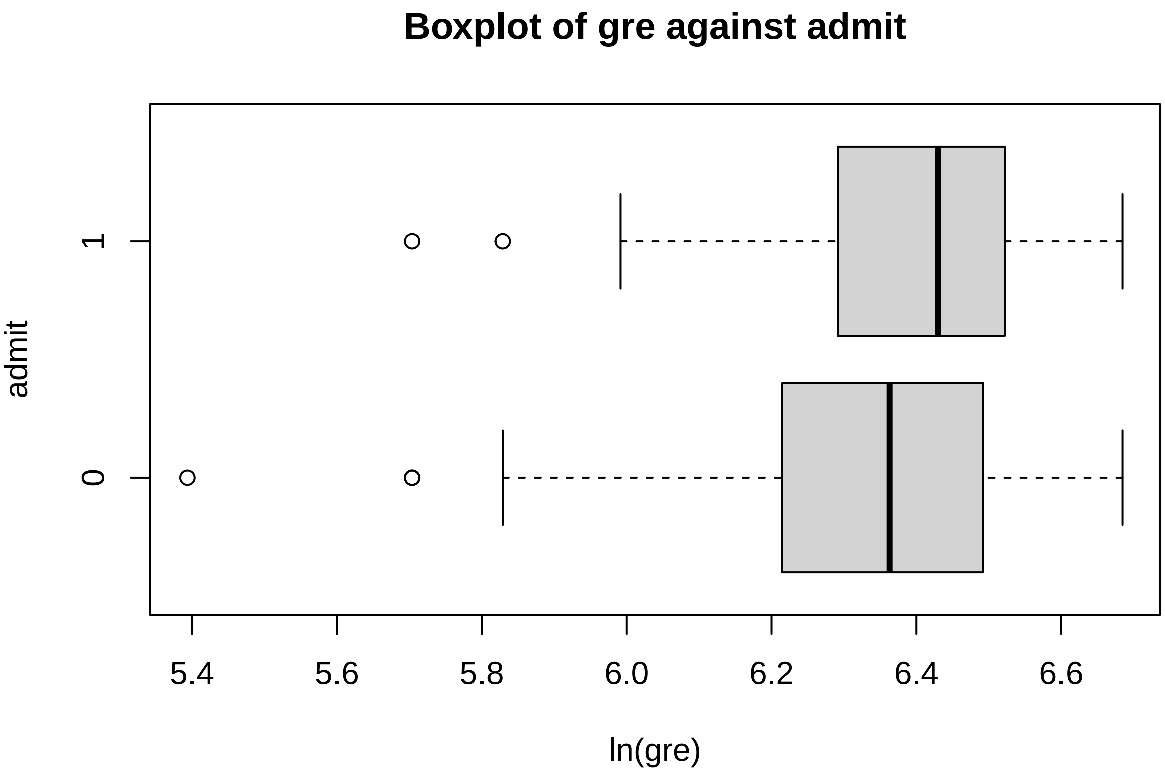
```
#Convert the variable gre into log(gre) in order to reduce the number of outliers, we have to take into
binary_data$gre <- log(binary_data$gre)


#Scatterplot
plot(binary_data$gre, binary_data$admit, xlab="ln(gre)", ylab="admit",
     main = "Scatterplot of log(gre) against admit", )
```



```
#Scatterplot with jitter, to better visualize the data.
plot(binary_data$gre,jitter(binary_data$admit,amount=0.05),xlab="ln(gre)",
     ylab="admit", ylim=c(-0.1,1.1), main="Scatterplot with jitter of gre against admit")
```



```
#Boxplot
boxplot(binary_data$gre ~ binary_data$admit, xlab="ln(gre)", ylab="admit",
        horizontal = TRUE, main = "Boxplot of gre against admit")
```



After visualizing those 3 plots, we can not say that there exist a relationship between GRE and Admit variables. In order to better see the relationship between these variables, we can do a range for the predictor variable, in this case GRE.

```
model1 <- glm(admit~gre, family = binomial(link = "logit"), trace=TRUE, data=binary_data)
```

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

```
## Deviance = 475.5433 Iterations - 1
## Deviance = 474.7836 Iterations - 2
## Deviance = 474.7834 Iterations - 3
## Deviance = 474.7834 Iterations - 4
```

```
summary(model1)
```

```
##
## Call:
## glm(formula = admit ~ gre, family = binomial(link = "logit"),
##     data = binary_data, trace = TRUE)
##
## Deviance Residuals:
```

```
##     Min       1Q   Median       3Q      Max
## -1.0990  -0.9092  -0.7657   1.3632   2.0812
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.8404     3.7259  -3.446 0.000568 ***
## gre           1.8929     0.5829   3.247 0.001165 **
## ---
## 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: 474.78  on 388  degrees of freedom
## AIC: 478.78
##
## Number of Fisher Scoring iterations: 4
```

```r
coefficients <- coef(model1)
cat("Estimated logit=", coefficients[2])
```

```
## Estimated logit= 1.892861
```

```r


#Odds ratio
odds_ratio <- exp(coefficients[2])
cat("Odds ratio=", odds_ratio)
```

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

```
## Odds ratio= 6.638331
```

The log-odds of being admitted increase by approximately 1.89 for every one-unit increase in the ‘gre’ score. We have already computed the odds ratio and its relationship with the variables; an odds ratio of 6.64 implies that for every one-unit increase in the natural logarithm of ‘gre,’ the odds of being admitted are about 6.64 times higher. A high odds ratio suggests a significant positive effect of ‘gre’ on the likelihood of being admitted. In this case, an increase in ‘gre’ (even after the logarithmic transformation) is associated with a substantial increase in the odds of admission.

5. Is gre a significant predictor (use = 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 = 0.1? First of all we have to state the hypothesis:

Ho: The coefficient for “gre” (1) is equal to 0.

Ha: The coefficient for “gre” (1) is not equal to 0.

```r
summary(model1)
```

```
##
## Call:
## glm(formula = admit ~ gre, family = binomial(link = "logit"),
##     data = binary_data, trace = TRUE)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0990  -0.9092  -0.7657   1.3632   2.0812
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.8404     3.7259  -3.446 0.000568 ***
## gre           1.8929     0.5829   3.247 0.001165 **
## ---
## 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: 474.78  on 388  degrees of freedom
## AIC: 478.78
##
## Number of Fisher Scoring iterations: 4
```

```
#We can compute the test statistic as the ratio of the estimated coefficient for "gre" to its standard
z <- (coefficients[2]/0.5829)
cat("Z=",z)
```

```
## Z= 3.247316
```

```
#We compute the p-value:
p_value <- 2 * (1 - pnorm(abs(z)))
cat("p-value=",p_value)
```

```
## p-value= 0.001164989
```

```
#We observe that the p-value is the one that we observe in the previous table of model 1 and also the z
```

Since the p-value < (0.05) we can reject the null hypothesis and therefore, gre is a good predictor at =0.05.

Since the p-value < (0.1) we can reject the null hypothesis and therefore, gre is a good predictor at =0.1.

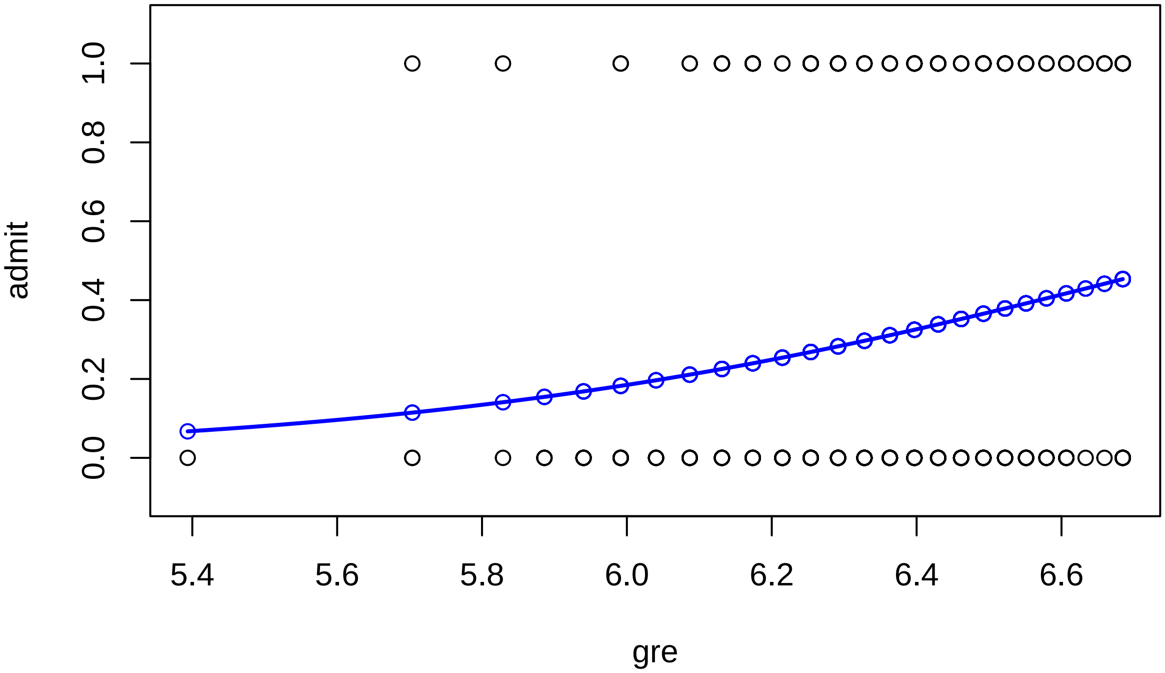
```


library(Correlplot)
```

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

```
## Loading required package: calibrate
```

```
## Loading required package: MASS
```

```
plot(binary_data$gre,binary_data$admit,xlab="gre",ylab="admit",
     ylim=c(-0.1,1.1))
curve(predict(model1,data.frame(gre=x),type="resp"),
      add=TRUE,col="blue",lwd=2)
fitted_model <- fitted(model1)
points(binary_data$gre,fitted_model,pch=1,col="blue")
```



```
estimated_logit <- coefficients[2]
standard_error_gre <- 0.5829
confidence_level <- 0.95
z_critical <- qnorm((1 + confidence_level) / 2)
margin_of_error <- z_critical * standard_error_gre

lower_limit_logit <- estimated_logit - margin_of_error
upper_limit_logit <- estimated_logit + margin_of_error
cat("95% Confidence Interval for log(gre): [", lower_limit_logit, ", ", upper_limit_logit,"]" )
```

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.

```
## 95% Confidence Interval for log(gre): [ 0.7503976 ,  3.035324 ]
```

```
lower_limit_odds <- exp(lower_limit_logit)
upper_limit_odds <- exp(upper_limit_logit)
cat("95% Confidence Interval for gre: [", lower_limit_odds, ", ", upper_limit_odds,"]" )
```

```
## 95% Confidence Interval for gre: [ 2.117842 ,  20.80771 ]
```

We can state that for a confidence interval of 95%, if we exponentiate the limits to untransform the previous logarithm modifications, we can interpret that for a one-unit increase in "gre," the odds of admission are estimated to be between approximately 2.12 times and 20.81 times higher, with 95% confidence.

```
#Read the data again with all NA and convert gre into logarithmic
data <- read_excel("binary_v2_missing.xlsx")
data$gre <- log(data$gre)

#Create a subset of the dataset for the ten individuals with "NA" in the 'admit' column
subset_data <- subset(data, is.na(admit), select = c("admit", "gre"))

#Predict function to obtain logit predictions
logit_predictions <- predict(model1, newdata = subset_data, type = "link")

#Predict function to obtain probability predictions
probability_predictions <- predict(model1, newdata = subset_data, type = "response")

#Combine the predictions with the subset_data
new_data_with_predictions <- data.frame(subset_data, logit = logit_predictions, probability = probabili


threshold <- 0.5
new_data_with_predictions$admit <- ifelse(probability_predictions >= threshold, 1, 0)
print(new_data_with_predictions)
```

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?

```
##    admit      gre      logit probability
## 1      0 6.633318 -0.2844231   0.4293697
## 2      0 6.173786 -1.1542537   0.2397130
## 3      0 5.828946 -1.8069887   0.1410025
## 4      0 6.040255 -1.4070100   0.1967061
## 5      0 6.492240 -0.5514652   0.3655245
## 6      0 5.828946 -1.8069887   0.1410025
## 7      0 6.040255 -1.4070100   0.1967061
## 8      0 6.429719 -0.6698075   0.3385399
## 9      0 6.327937 -0.8624680   0.2968240
## 10     0 6.429719 -0.6698075   0.3385399
```

The predict function gives us the logit and the probability for each individual which has "NA" as admit using our model from the whole data set. So we state a threshold=0.5 saying that if this probability is bigger than than 0.5, the admit column must get a 1 saying that the individual has been admitted, otherwise, it is not admitted having 0 as consequence. Seeing the results, we can conlcude that none of our selected individuals had been admitted.

```r
logLik_fitted <- logLik(model1)
#Fit a null model with only the intercept
null_model <- glm(admit ~ 1, data = binary_data, family = binomial(link = "logit"))
logLik_null <- logLik(null_model)

#Calculate McFadden's Pseudo R-squared
McFadden_R2 <- 1 - (logLik_fitted / logLik_null)

cat("McFadden's Pseudo R-squared:", McFadden_R2)
```

9. Calculate McFadden's pseudo R2 for the logistic regression of admit on gre. What does a zero value for this pseudo R2 mean?

```
## McFadden's Pseudo R-squared: 0.02349393
```

A 0 value in McFadden's Pseudo R-squared means that the model does not provide any improvement in explaining the variance in the response variable. As closer to 1, the model better explains the variance in the response variable.

In our case, we have a relatively low value indicating that the model explains only a small portion of the variance in the response variable "admit.

```r
#Calculate the median of 'gpa'
median_gpa <- median(binary_data$gpa)

#Create the 'medgpa' variable
binary_data$medgpa <- ifelse(binary_data$gpa <= median_gpa, "Below Median", "Above Median")

#Count the individuals in each category
below_median_count <- sum(binary_data$medgpa == "Below Median")
above_median_count <- sum(binary_data$medgpa == "Above Median")

cat("Number of individuals with GPA Below or Equal to the Median:", below_median_count, "\n")
```

10. (1p) 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.

```
## Number of individuals with GPA Below or Equal to the Median: 201
```

```r
cat("Number of individuals with GPA Above the Median:", above_median_count, "\n")
```

```
## Number of individuals with GPA Above the Median: 189
```

```r
#Fit the logistic regression model with 'admit' as the response and 'medgpa' as the predictor
model2 <- glm(admit ~ medgpa, data = binary_data, family = binomial)
summary(model2)
```

11. (2p) Perform the logistic regression of admit on medgpa. Report the estimated logit. What is the interpretation of the exponentiated slope of this regression?

```
##
## Call:
## glm(formula = admit ~ medgpa, family = binomial, data = binary_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0492  -1.0492  -0.6938   1.3113   1.7562
##
## Coefficients:
##                    Estimate Std. Error z value Pr(>|z|)
## (Intercept)         -0.3093     0.1472  -2.101   0.0356 *
## medgpaBelow Median  -0.9921     0.2264  -4.382 1.18e-05 ***
## ---
## 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: 466.23  on 388  degrees of freedom
## AIC: 470.23
##
## Number of Fisher Scoring iterations: 4
```

```
#estimated logit
estimated_logit <- coef(model2)["medgpaBelow Median"]
cat("Estimated logit for 'medgpa' (Below Median)=", estimated_logit, "\n")
```

```
## Estimated logit for 'medgpa' (Below Median)= -0.9920737
```

```
#Exponentiate the estimated logit to get the odds ratio
odds_ratio <- exp(estimated_logit)
cat("Exponentiated slope (Odds Ratio) for 'medgpa' (Below Median)=", odds_ratio, "\n")
```

```
## Exponentiated slope (Odds Ratio) for 'medgpa' (Below Median)= 0.370807
```

First of all, we can see that we have obtained a significant p-value and therefore we can draw reliable conclusions from the relationship of the admit variable with the medgpa.

The interpretation would be that an odds ratio of 0.371 means that individuals with a GPA below or equal to the median have lower odds of admission compared to individuals with a GPA above the median. Specifically, individuals with a GPA below or equal to the median have odds of admission that are approximately 0.371 times (or about 37.1% of) the odds of admission for individuals with a GPA above the median. So it makes sense as the gpa is the grade point average and if this average is higher there are more chances of being admitted.

```


#Fit the logistic regression model with 'admit' as the response and 'gpa' as the predictor
model3 <- glm(admit ~ gpa, data = binary_data, family = binomial)
summary(model3)
```

12. (2p) 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.

```
##
## Call:
## glm(formula = admit ~ gpa, family = binomial, data = binary_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1077  -0.8850  -0.7584   1.3213   1.9900
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -4.3821     1.0519  -4.166  3.1e-05 ***
## gpa           1.0540     0.3032   3.476 0.000509 ***
## ---
## 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: 473.48  on 388  degrees of freedom
## AIC: 477.48
##
## Number of Fisher Scoring iterations: 4
```

```
#estimated logit for 'gpa'
estimated_logit_gpa <- coef(model3)["gpa"]
cat("Estimated logit for 'gpa'=", estimated_logit_gpa, "\n")
```

```
## Estimated logit for 'gpa'= 1.053991
```

The positive estimated logit of 1.054 suggests that, on the log-odds scale, as gpa increases by one unit, the log-odds of being admitted also increase by approximately 1.054 units. In other words, higher gpa's are associated with a higher likelihood of admission according to the logistic regression model. This means that the results are consistent with the previous ones. Results from both models conclude that higher gpa or gpa above median are related to higher chances of being admitted. Maybe the analysis with gpa is better because it is a continious variable and it may be more specific, however, in our opinion, if we want to be more precise and have better results it is a good idea to analyse both variables (as we have done) and check if the results are the same. This way we would get more reliable results.

```
#Fit the logistic regression model with all available predictors (excluding 'medgpa')
model4 <- glm(admit ~ gre + gpa + rank + gender + month, data = binary_data, family = binomial)

#Perform a likelihood ratio test to determine global significance
#We compare the full model with all predictors to a null model (intercept-only model)

# Fit a null model
null_model <- glm(admit ~ 1, data = binary_data, family = binomial)
```

```
# Perform the likelihood ratio test
likelihood_ratio_test <- anova(null_model, model4, test = "Chisq")

# Report the test statistic, its reference distribution, and the p-value
cat("Likelihood Ratio Test Statistic=", likelihood_ratio_test$Deviance[2], "\n")
```

13. (2p) 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.

```
## Likelihood Ratio Test Statistic= 39.06025
```

```
cat("Reference Distribution: Chi-squared with degrees of freedom =", likelihood_ratio_test$Df[2], "\n")
```

```
## Reference Distribution: Chi-squared with degrees of freedom = 5
```

```
cat("P-Value=", likelihood_ratio_test$Pr[2], "\n")
```

```
## P-Value= 2.309438e-07
```

A p-value (2.309438e-07)< 0.05 indicates that the logistic regression model with all available predictors (excluding the medgpa indicator) is globally significant. In conclusion, there is strong evidence that at least one of the predictors in the model has a significant effect on admit.

```


#First we need to see which variables are significant (p-value < 0.05)
summary(model4)
```

14. (1p) Try to simplify the model by eliminating non-significant predictors (one by one, use = 0.05). What is your final model?

```
##
## Call:
## glm(formula = admit ~ gre + gpa + rank + gender + month, family = binomial,
##     data = binary_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5490  -0.8838  -0.6318   1.1423   2.1620
##
## Coefficients:
##               Estimate Std. Error z value Pr(>|z|)
## (Intercept) -8.9660052  3.9311845  -2.281   0.0226 *
## gre          1.0735334  0.6450701   1.664   0.0961 .
## gpa          0.8084407  0.3344101   2.418   0.0156 *
## rank        -0.5778924  0.1299137  -4.448 8.66e-06 ***
## gender      -0.0900727  0.2324512  -0.387   0.6984
## month       -0.0002307  0.0514282  -0.004   0.9964
## ---
## 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: 447.15  on 384  degrees of freedom
## AIC: 459.15
##
## Number of Fisher Scoring iterations: 4
```

```
#We found that gender and month are not significant at all, rank and gpa are clearly significant and gr
model_without_month <- glm(admit ~ gre + gpa + rank + gender, data = binary_data, family = binomial)
model__without_gender <- glm(admit ~ gre + gpa + rank + month, data = binary_data, family = binomial)
#We see the same results so we decide to exclude them from the model

model5 <- glm(admit ~ gre + gpa + rank, data = binary_data, family = binomial)
summary(model5)
```

```
##
## Call:
## glm(formula = admit ~ gre + gpa + rank, family = binomial, data = binary_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.5313  -0.8814  -0.6345   1.1543   2.1771
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -9.0136     3.8999  -2.311   0.0208 *
## gre           1.0699     0.6443   1.661   0.0968 .
## gpa           0.8160     0.3332   2.449   0.0143 *
## rank         -0.5761     0.1297  -4.443 8.88e-06 ***
## ---
## 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: 447.30  on 386  degrees of freedom
## AIC: 455.3
##
## Number of Fisher Scoring iterations: 4
```

```
model6 <- glm(admit ~ gpa + rank, data = binary_data, family = binomial)
summary(model6)
```

```
##
## Call:
## glm(formula = admit ~ gpa + rank, family = binomial, data = binary_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.4596  -0.8903  -0.6602   1.1583   2.1945
```

```
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept)  -2.8432     1.1128  -2.555  0.01061 *
## gpa           1.0209     0.3113   3.279  0.00104 **
## rank         -0.5975     0.1291  -4.630 3.66e-06 ***
## ---
## 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: 450.13  on 387  degrees of freedom
## AIC: 456.13
##
## Number of Fisher Scoring iterations: 3
```

```
model_gre <- glm(admit ~ gre, data = binary_data, family = binomial)
summary(model_gre)
```

```
##
## Call:
## glm(formula = admit ~ gre, family = binomial, data = binary_data)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.0990  -0.9092  -0.7657   1.3632   2.0812
##
## Coefficients:
##             Estimate Std. Error z value Pr(>|z|)
## (Intercept) -12.8404     3.7259  -3.446 0.000568 ***
## gre           1.8929     0.5829   3.247 0.001165 **
## ---
## 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: 474.78  on 388  degrees of freedom
## AIC: 478.78
##
## Number of Fisher Scoring iterations: 4
```

Once we have excluded the clearly non-significant predictors we have two different models, model5 which have gre, gpa and rank predictors and model6 which is the same without gre. We need to choose between these two models. On the one hand, we believe that gre should be included due to its importance (we observed that gre was significant when considered alone). On the other hand, gre is not significant in model6, which includes only the consistently significant variables gpa and rank. Knowing this, we chose model6 (which includes gpa and rank) because it is a simpler and more interpretable model that maintains the significance of all predictors included.

```r
#Our full model is model4 and our final model is model6

#Perform the likelihood ratio test to compare the full model to the reduced model
lrt <- anova(model6, model4, test = "Chisq")
p_value_joint_nullity <- lrt$Pr[2]
cat("P-Value=", p_value_joint_nullity, "\n")
```

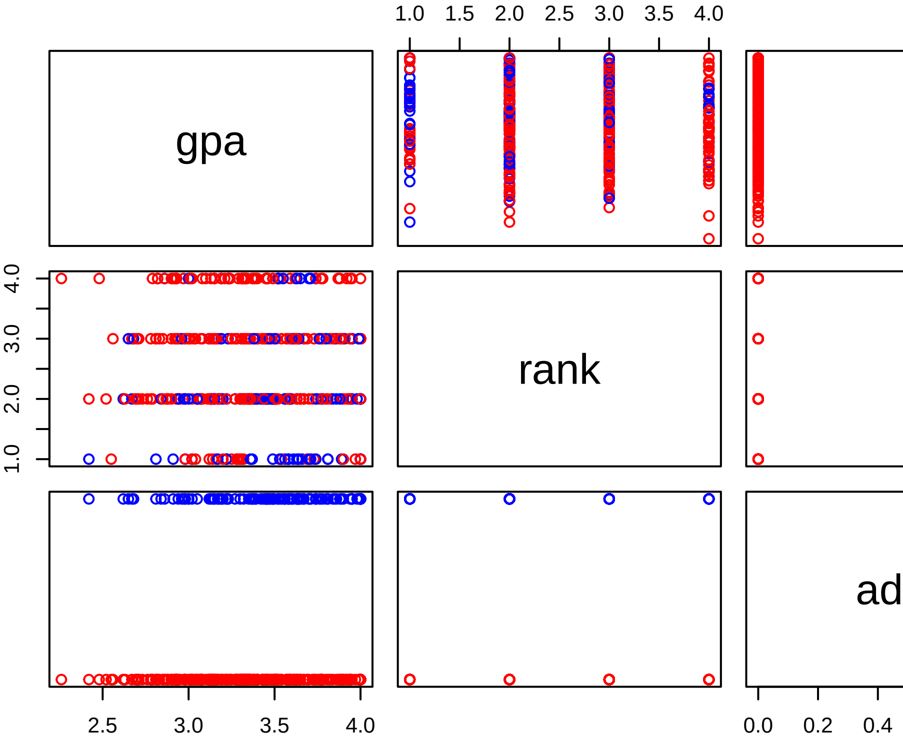
15. (1p) Test by a likelihood ratio test if we can consider the joint nullity of the coefficients of all variables that you have eliminated. Report the p-value of this test.

```
## P-Value= 0.3936222
```

A p-value bigger than 0.05 indicates that the removal of the non-significant predictors (the variables that were eliminated) did not significantly affect the fit of the model. This suggests that the final model, which includes only the significant predictors (gre, gpa, and rank), is a reasonable model that explains the relationship between the predictors and the response variable (admission).

```r
#New data frame with the variables in the final model
final_data <- data.frame(gpa = binary_data$gpa, rank = binary_data$rank, admit = binary_data$admit)

#different colors/symbols for admitted (1) and non-admitted (0)
colors <- ifelse(final_data$admit == 1, "blue", "red")

#scatter plot matrix with colors and symbols
pairs(final_data, col = colors)
```

16. (1p) 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?

We can clearly see how admitted and non-admitted are distributed in all the variables but between gpa and rank there's no visual and clear separation of the two groups. That can be because they may not be strongly related to each other when considered together. However, they can still be individually significant predictors for the admit variable in your logistic regression model.

```r
#Predict admission probabilities
predicted_probabilities <- predict(model6, type = "response")

threshold <- 0.5

#Create a binary vector of observed admission based on the threshold
observed_admission <- ifelse(binary_data$admit == 1, 1, 0)

#Create a binary vector of predicted admission based on the threshold
predicted_admission <- ifelse(predicted_probabilities >= threshold, 1, 0)

#Create a cross table of predicted admission vs. observed admission
cross_table <- table(Predicted = predicted_admission, Observed = observed_admission)
print(cross_table)
```

17. (1p) Predict the admission of the individuals in the database with your final model, and make a cross table of predicted admit against observed admit.

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

```
#classification rate (accuracy) from the cross table
classification_rate <- sum(diag(cross_table)) / sum(cross_table)

#Print the classification rate as a percentage
cat("Classification Rate (Accuracy):", round(classification_rate * 100, 2), "%\n")
```

18. (1p) 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.

```
## Classification Rate (Accuracy): 72.82 %
```

```
#Predict admission using the full model
predicted_admit_full <- predict(model4, type = "response")

#Convert predicted probabilities to binary outcomes (1: Admitted, 0: Not Admitted) using a threshold (e
predicted_admit_binary_full <- ifelse(predicted_admit_full > 0.5, 1, 0)

#Create a cross table for the full model
cross_table_full <- table(predicted_admit_binary_full, binary_data$admit)
print(cross_table_full)
```

19. (1p) Also calculate the classification rate for predictions made with the full model, which contains all predictors. Does this work better? Justify your answer

```
##
## predicted_admit_binary_full   0   1
##                           0 248  95
##                           1  19  28
```

```
#Calculate the classification rate (accuracy) for the full model
classification_rate_full <- sum(diag(cross_table_full)) / sum(cross_table_full)
cat("Classification Rate (Accuracy) for Full Model:", round(classification_rate_full * 100, 2), "%\n")
```

```
## Classification Rate (Accuracy) for Full Model: 70.77 %
```

This one, the model with all predictors does not work better. We can state this because the percentage of accuracy is lower in this case compared with our correct model, the one that contains all predictors that are significant.