KAYODE OKUNOLA

(002845768)

LINEAR STATISTICS

LOGISTIC REGRESSION

PREDICTING THE ODDS OF STUDENT CHANCE OF ADMISSION INTO GRADUATE SCHOOL

Introduction

The pursuit of postgraduate degrees in higher education frequently entails a critical decision-making process for prospective students. The ambiguity over admission standards and the competitiveness of academic programs might induce anxiety in candidates. This research aims to create a reliable prediction model to estimate the probability of a student being admitted to a graduate school. The principal predictors in our model are the Cumulative Grade Point Average (CGPA), Graduate Record Examination (GRE) scores and higher institution ranked, three critical factors commonly evaluated by universities.

This study aims to provide a significant resource for prospective students and educational institutions, elucidating the complex dynamics of the admission process. We examine the importance of CGPA and GRE scores as determinants of admission probabilities through statistical analysis and logistic regression modeling. The response variable, designated as "Chance of Admit," is the key component of our predictive model.

The prediction model is developed using a logistic regression framework, with careful consideration of fitness evaluation, interaction terms, and assumption test. We analyze the model accuracy, an indicator of the model's explanatory capacity. Additionally, we examine the requirement of interaction terms via statistical analysis, guaranteeing the incorporation of significant elements in our model.

Methods

Since our dependent variable is a binary variable (0, 1), the dependent variable follow a logit binomial family, give as;

$$Log\ (odds) = logit(Admit) = \ln\left(\frac{Admit}{1-Admit}\right) \tag{1}$$

Putting in a regression form, we have

$$\ln\left(\frac{Admit}{1-Admit}\right) = \beta_0 + \beta_i x_i \tag{2}$$

By simplifying equation (2), we have

$$\left(\frac{Admit}{1-Admit}\right) = e^{\beta_0 + \beta_i x_i} \tag{3}$$

$$Admit = \frac{e^{\beta_0 + \beta_i x_i}}{1 + e^{\beta_0 + \beta_i x_i}} \tag{4}$$

Where;

x_i are the independent variables

$\beta_0\ is\ the\ intercept\ of\ the\ model$

$\beta_i\ are\ the\ coefficient\ of\ the\ independent\ variables$

Equation (4) follow a binomial distribution and using MLE to estimates the parameters, we have

$$\max_{\beta_0,\beta_1,\beta_2,\beta_3\ldots\beta_i} \prod_{i=1}^{n} \left(\frac{e^{\beta_0+\beta_i x_i}}{1+e^{\beta_0+\beta_i x_i}}\right)^{y_i} \left(\frac{1}{1+e^{\beta_0+\beta_i x_i}}\right)^{1-y_i} \quad (5)$$

To solve equation (5), we will use the "GLM" function in R studio.

Note: in the coding, gpa represent cummulative grade point average, gre represent graduate record examination and rank represent institution rating.

Binary Logistic Model

$Ho: \beta = 0$

$H1: \beta \neq 0$

Model Selection

```
Coefficients:
             Estimate Std. Error z value Pr(>|z|)    
(Intercept) -52.37273   10.07931  -5.196 2.04e-07 ***
gre           0.06825    0.03229   2.114   0.0345 *  
gpa           4.27863    0.82194   5.206 1.93e-07 ***
rank         -0.46555    0.32709  -1.423   0.1546    
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Model Interpretation

Holding other variables constant;

- Having a good gre score as a student seeking admission into graduate school, the log odds of admision increase by 0.06825.
- Having a good gpa score as a student seeking admission into graduate school, the log odds of admision increase by 4.27863.
- Having graduated from a ranked school and seeking admission into graduate school, the log odds of admision decrease by 0.46555

Transformed estimates into odd ratio

```
 (Intercept)          gre          gpa         rank 
1.798087e-23 1.070636e+00 7.214150e+01 6.277907e-01 
```

Considering these estimates, we can say (while holding the other variables constant):

- Having a good gre score, the odds of being admitted to graduate school increase by 1.0706.
- Having a good gpa, the odds of being admitted to graduate school increase by 72.145.
- Having graduated from a ranked school, the odds of being admitted to graduate school decrease by 0.6278

95% confidence intervals for the odds ratios are as follows:

	Estimate <dbl>	Odds_Ratio <dbl>	X2.5..CI <dbl>	X97.5..CI <dbl>
(Intercept)	-52.37273389	1.798087e-23	8.671073e-33	1.673562e-15
gre	0.06825284	1.070636e+00	1.006863e+00	1.143779e+00
gpa	4.27862941	7.214150e+01	1.572926e+01	4.065667e+02
rank	-0.46554846	6.277907e-01	3.261565e-01	1.183213e+00

Stepwise Procedure

By adding the interaction term “gre*gpa”, “gre*rank” and “gpa*rank” to the model, the results of the Stepwise procedure after removing the insignificant interaction produce a higher AIC value and the final stepwise result suggest the model with to interaction and rearrange the predator variable inorder of importance. **However, I choose model 1 above which is with no interaction over the stepwise procedure because it has a lesser AIC value. Also, I do not have many predator variables, hence, to aviod overfitting the data, bias estimate and inflated type 1 error (Harrell, 2015) I choose model 1 over the stepwise model.**

```
Coefficients:
               Estimate Std. Error z value Pr(>|z|)
(Intercept) -201.40415  141.11996  -1.427    0.154
rank          -0.47127    0.33501  -1.407    0.160
gpa           23.06116   17.67656   1.305    0.192
gre            0.55067    0.45422   1.212    0.225
gpa:gre       -0.06076    0.05677  -1.070    0.285
```

```
Call:  glm(formula = admit ~ gpa + gre + rank, family = binomial, data = data)

Coefficients:
(Intercept)          gpa          gre         rank  
  -52.37273      4.27863      0.06825     -0.46555  

Degrees of Freedom: 399 Total (i.e. Null);  396 Residual
Null Deviance:      237.4 
Residual Deviance: 129.8        AIC: 137.8
```

Goodness of Fit

It is however important that, the model should fit the data adequately. Using Hosmer-Lemeshow Test

```
        Hosmer and Lemeshow goodness of fit (GOF) test

data:  data$admit, fitted(model)
X-squared = 3.6487, df = 8, p-value = 0.8873
```

The $p > 0.05$, which is statistically not significant, reject the null hypothesis and conclude that our model Indicates a good fit.

Model Assumption

Assumption were carried out to be sure no assumption is violated and to validate our model selection

- The dependent variable admit (1=admitted 0=not admitted) is a binary variable and the
- The observations are independent of each other.

Multicollinearity: It is however, important that the predators variables should not be perfectly correlated. In this aid, variance inflation factor (VIF) was used to check the multicollinearity of the model.

Result reveals that the Vif values for the predators variable are less than 5, which indicate no presence of multicollinearity.

```
     gre      gpa     rank
1.243898 1.540565 1.521677
```

Linearity assumption: Box-Tidwell Test was use for this assumption, add interaction terms for log-transformed predictors. If interaction terms are significant, the linearity assumption may be violated. Result of the Box-Tidwell reveals the interaction effect pvalue >0.05 which is statistically not significant, therefore it was concluded that the assumption of linearity is not violated.

```
Coefficients:
                 Estimate Std. Error z value Pr(>|z|)
(Intercept)    -73639.520 181458.759  -0.406    0.685
gre              -489.334   1221.948  -0.400    0.689
log_gre         19675.626  48651.232   0.404    0.686
gpa              -840.951   3862.787  -0.218    0.828
log_gpa          1556.481   7450.886   0.209    0.835
rank                9.043     27.272   0.332    0.740
log_rank           -8.682     20.098  -0.432    0.666
gre:log_gre        63.223    158.090   0.400    0.689
gpa:log_gpa       211.328    951.822   0.222    0.824
rank:log_rank      -2.982      9.824  -0.304    0.761
```

Power

To assess the predictive power of the model, we use the McFadden R^2.

```
        llh      llhNull          G2    McFadden        r2ML        r2CU
-64.8982065 -118.6861026 107.5757923   0.4531946   0.2358105   0.5268671
```

A McFadden R2 score ranging from 0.2 to 0.4 is deemed satisfactory. Consequently, given that our McFadden R2 of 0.45, we may assert that the chosen model is effective good for predicting chance of admission.

Cross Validation

Using Cross Validation techniques on the model, we obtain the following results:

To evaluate the model's validity, I initially divide my data into 80% for training and 20% for testing and construct the model using the training data. The train model was employed to predict the testing outcome. The confusion matrix displays the count of student admitted and those who were not. The model's accuracy was determined to be 90.95%.

```
Confusion Matrix and Statistics

          Reference
Prediction  0  1
         0  7  1
         1  3 69

               Accuracy : 0.95
                 95% CI : (0.8769, 0.9862)
    No Information Rate : 0.875
    P-Value [Acc > NIR] : 0.02237

                  Kappa : 0.75

 Mcnemar's Test P-Value : 0.61708

            Sensitivity : 0.7000
            Specificity : 0.9857
         Pos Pred Value : 0.8750
         Neg Pred Value : 0.9583
             Prevalence : 0.1250
         Detection Rate : 0.0875
   Detection Prevalence : 0.1000
      Balanced Accuracy : 0.8429

       'Positive' Class : 0
```

```
[1] "Accuracy: 0.95"
```

The model's total accuracy in predicting the admission rate is 0.95. This suggests that our approach is more effective at accurately predicting the likelihood of students being admitted.

Variable Important

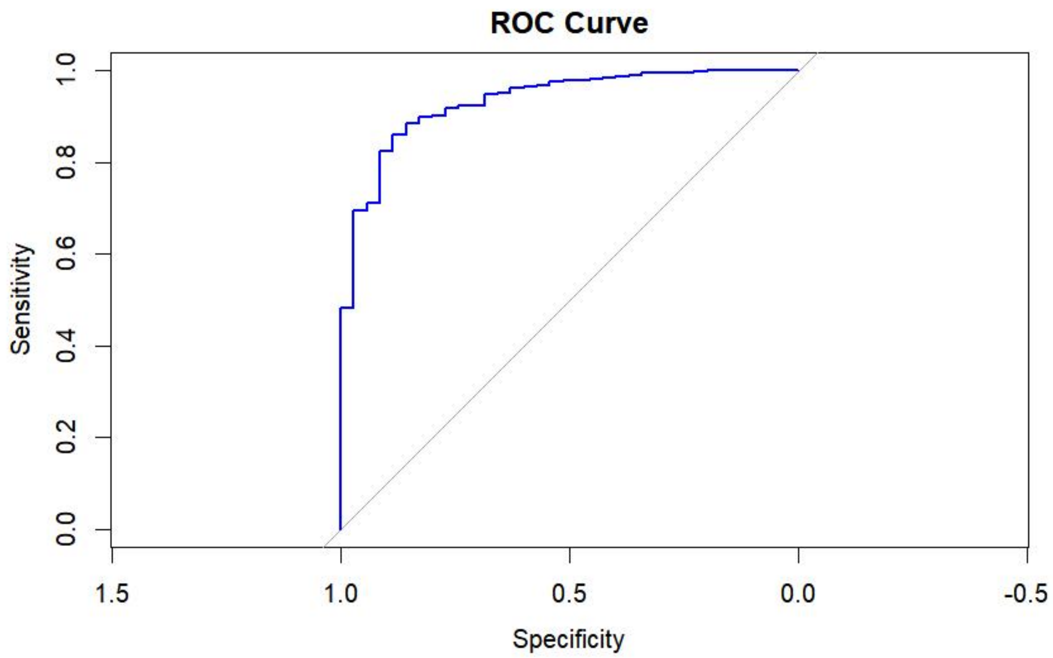
	Overall <dbl>
gre	2.113974
gpa	5.205537
rank	1.423319

Result reveal that gpa (cummulative grade point average) variable has the biggest impact on chance of admission follow by gre (graduate records examination) score and rank (institution rating) respectively.

Area Uder Curve (AUC)

```
Area under the curve: 0.9332
```

The AUC is the total area under the ROC curve. It summarizes the overall performance of the model. At all threshold levels, the model performs significantly better than random guessing.



The curve rises steeply toward the top-left corner, The area beneath this ROC curve is 0.9332. This suggests that the model possesses a good degree of accuracy.

Conclusion

I choose and interpreted the model with no interaction because it has the better AIC value and best model accuracy. Thus, two predator variable (gpa and gre) are statistically significance for the pvalue less than 0.05 while rank (institution rating) having a negative coefficient and not significant,which means institution rank has no effect or impact on students chace of admission into graduate school while graduate record examination score and undergraduate cummulative grade point average score have a positive impact on students chance of admission into graduate school. It is therefore recommeded to high school students to put in effort in their studies to have a good gpa and gre score as these have a significance effect on their chances of securing admission into graduate school.

Reference

[1] Harrell, F.(2015). Regression modeling strategies: with application to linear models, logistic and ordinal regression and survival analysis (2nd ed.). New York, NY:Springer.

[2] Link to dataset: *https://www.kaggle.com/datasets/mohansacharya/graduate-admissions?resource=download*

Rcode

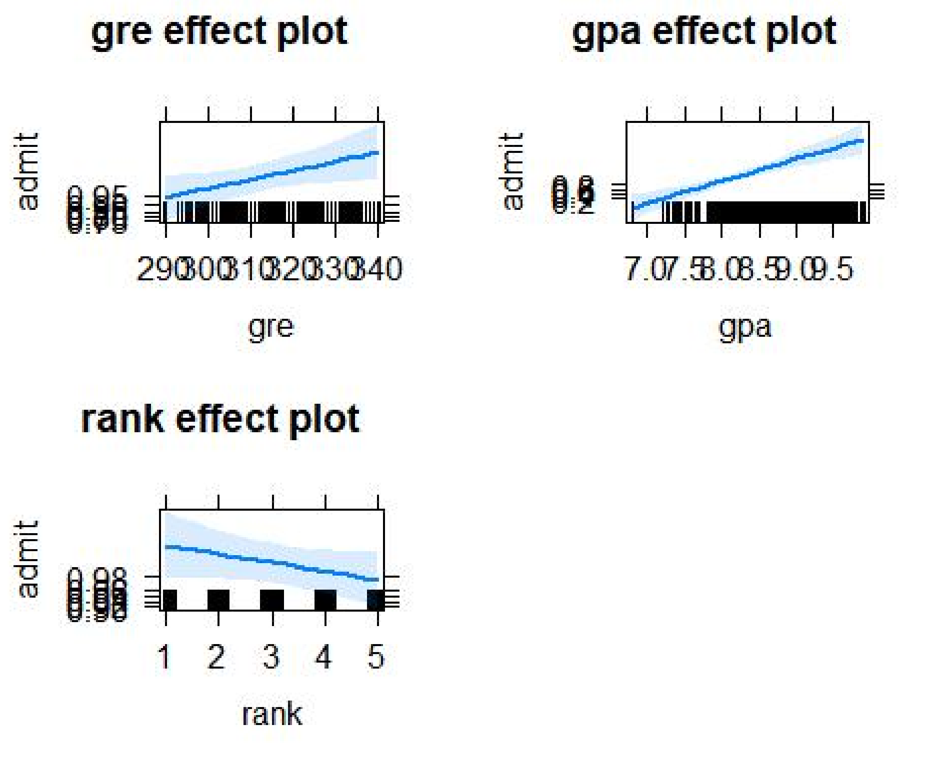
```
> library(ggplot2)
> require(GGally)
> require(reshape2)
> require(lme4)
> library(effects)
> library(tidyverse)
> library(caret)
> library(car)
> library(ResourceSelection)
> library(pROC)
> library(pscl)
> library(survey)

> #import dataset

> data <- read.csv(file.choose())
> data$gre=data$GRE.Score
> data$gpa=data$CGPA
> data$rank=data$University.Rating

> #Verifying my dependent variable
> table(data$admit)
  0   1
 35 365

>  #Hence, the dependent variable is a binary
>  # Use Variance Inflation Factor (VIF) to detect multicollinearity
> model <- glm(admit ~ gre + gpa + rank, data = data, family = binom
ial)
> summary(model)

Call:
glm(formula = admit ~ gre + gpa + rank, family = binomial, data = da
ta)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.1871   0.0338   0.1002   0.2800   1.7134

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept) -52.37273   10.07931  -5.196 2.04e-07 ***
gre           0.06825    0.03229   2.114   0.0345 *
gpa           4.27863    0.82194   5.206 1.93e-07 ***
rank         -0.46555    0.32709  -1.423   0.1546
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)
```

```
    Null deviance: 237.37  on 399  degrees of freedom
Residual deviance: 129.80  on 396  degrees of freedom
AIC: 137.8

Number of Fisher Scoring iterations: 7
>
 # Extract coefficients> coef_summary <- summary(model)$coefficients

> # Calculate Odds Ratios (OR) and 95% Confidence Intervals (CI)

> odds_ratios <- exp(coef_summary[, "Estimate"])
> conf_int <- exp(confint(model))  # 95% confidence intervals
> p_values <- coef_summary[, "Pr(>|z|)"]

>  # Combine results into a data frame

> results <- data.frame(
   Estimate = coef_summary[, "Estimate"],
   Odds_Ratio = odds_ratios,
   `2.5% CI` = conf_int[, 1],
   `97.5% CI` = conf_int[, 2],
   P_Value = p_values
 )

>  # Print the results> print(results)

                Estimate   Odds_Ratio     X2.5..CI    X97.5..CI
 P_Value
(Intercept) -52.37273389 1.798087e-23 8.671073e-33 1.673562e-15 2.03
5540e-07
gre           0.06825284 1.070636e+00 1.006863e+00 1.143779e+00 3.45
1751e-02
gpa           4.27862941 7.214150e+01 1.572926e+01 4.065667e+02 1.93
4367e-07
rank         -0.46554846 6.277907e-01 3.261565e-01 1.183213e+00 1.54
6436e-01

> model_stepwise <- glm(admit ~ gre+gpa+rank+gre*gpa+gre*rank+gpa*ra
nk, data = data, family = binomial)
> null=glm(admit ~ 1, data = data, family = binomial)
> step(null,scope=list(lower=null,upper=model_stepwise),direction="b
oth")Start:  AIC=239.37
admit ~ 1

       Df Deviance    AIC
+ gpa   1   135.59 139.59
+ gre   1   166.69 170.69
+ rank  1   201.52 205.52
<none>      237.37 239.37

Step:  AIC=139.59
admit ~ gpa

       Df Deviance    AIC
+ gre   1   131.86 137.86
<none>      135.59 139.59
+ rank  1   134.56 140.56
- gpa   1   237.37 239.37

Step:  AIC=137.86
admit ~ gpa + gre

          Df Deviance    AIC
+ rank     1   129.80 137.80
<none>         131.86 137.86
+ gre:gpa  1   130.85 138.85
- gre      1   135.59 139.59
- gpa      1   166.69 170.69
```

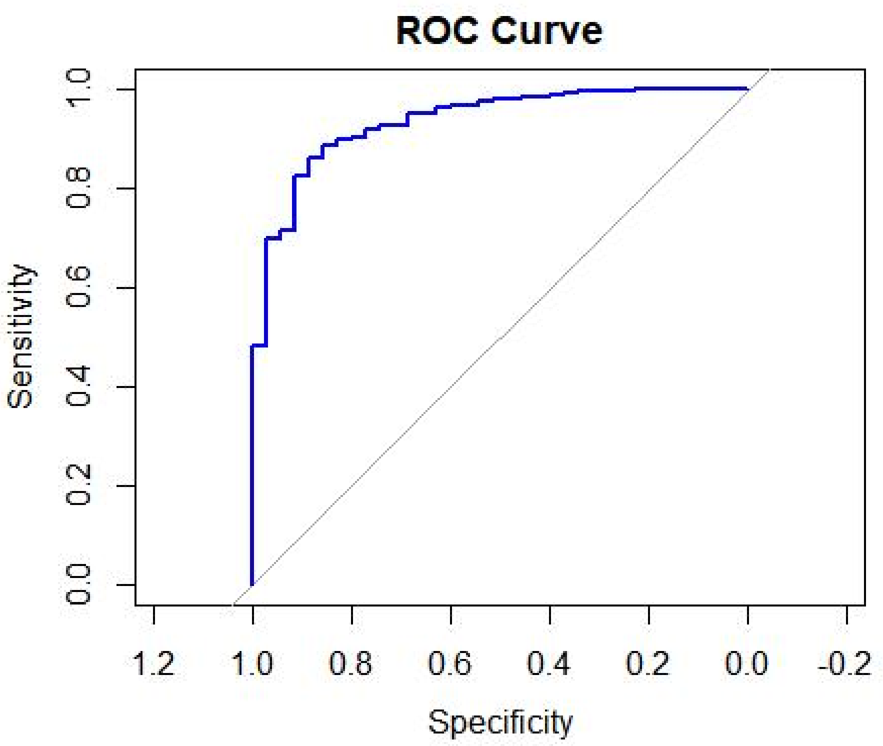
```
Step:  AIC=137.8
admit ~ gpa + gre + rank

           Df Deviance    AIC
<none>          129.80 137.80
- rank      1   131.86 137.86
+ gpa:rank  1   128.21 138.21
+ gre:gpa   1   128.84 138.84
+ gre:rank  1   129.20 139.20
- gre       1   134.56 140.56
- gpa       1   164.36 170.36

Call:  glm(formula = admit ~ gpa + gre + rank, family = binomial, da
ta = data)

Coefficients:
(Intercept)          gpa          gre         rank
  -52.37273      4.27863      0.06825     -0.46555

Degrees of Freedom: 399 Total (i.e. Null);  396 Residual
Null Deviance:      237.4
Residual Deviance: 129.8      AIC: 137.8>

> model_2 <- glm(admit ~  rank + gpa + gre + gpa:gre, data = data, f
amily = binomial)
> summary(model_2)
Call:
glm(formula = admit ~ rank + gpa + gre + gpa:gre, family = binomial,

    data = data)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-3.03478   0.06566   0.12224   0.26801   1.80646

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -201.40415  141.11996  -1.427    0.154
rank          -0.47127    0.33501  -1.407    0.160
gpa           23.06116   17.67656   1.305    0.192
gre            0.55067    0.45422   1.212    0.225
gpa:gre       -0.06076    0.05677  -1.070    0.285

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 237.37  on 399  degrees of freedom
Residual deviance: 128.84  on 395  degrees of freedom
AIC: 138.84

Number of Fisher Scoring iterations: 8


  #Goodness of Fit

>  cooks.distance<-cooks.distance(model)
> which(cooks.distance>1)

named integer(0)

> #The model should fit the data adequately. Using Hosmer-Lemeshow T
est

># Hosmer-Lemeshow test
> hoslem.test(data$admit, fitted(model),g=10)


        Hosmer and Lemeshow goodness of fit (GOF) test
```

```
data:  data$admit, fitted(model)
X-squared = 3.6487, df = 8, p-value = 0.8873

> #Wald Test to determine if predictors are significant
> regTermTest(model,"gpa")Wald test for gpa
 in glm(formula = admit ~ gre + gpa + rank, family = binomial, data
= data)
F =  27.09761  on  1  and  396  df: p= 3.1138e-07

> regTermTest(model,"gre")Wald test for gre
 in glm(formula = admit ~ gre + gpa + rank, family = binomial, data
= data)
F =  4.468885  on  1  and  396  df: p= 0.035142

> regTermTest(model,"rank")Wald test for rank
 in glm(formula = admit ~ gre + gpa + rank, family = binomial, data
= data)
F =  2.025838  on  1  and  396  df: p= 0.15543

> # Check correlation for numeric variables
> Iv=data[c("gre", "gpa", "rank")]

> cor(Iv)
           gre       gpa      rank
gre  1.0000000 0.8330605 0.6689759
gpa  0.8330605 1.0000000 0.7464787
rank 0.6689759 0.7464787 1.0000000

> vif(model)

     gre      gpa     rank
1.243898 1.540565 1.521677

> # Add interaction terms for log-transformed predictors
> data$log_gre <- log(data$gre)
> data$log_gpa <- log(data$gpa)
> data$log_rank <- log(data$rank)
> model_lin <- glm(admit ~ gre*log_gre + gpa*log_gpa + rank*log_rank,
 data = data, family = binomial)
> summary(model_lin)

Call:
glm(formula = admit ~ gre * log_gre + gpa * log_gpa + rank *
    log_rank, family = binomial, data = data)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-3.2819   0.0098   0.0718   0.2804   1.6116

Coefficients:
                Estimate Std. Error z value Pr(>|z|)
(Intercept)   -73639.520 181458.759  -0.406    0.685
gre             -489.334   1221.948  -0.400    0.689
log_gre        19675.626  48651.232   0.404    0.686
gpa             -840.951   3862.787  -0.218    0.828
log_gpa         1556.481   7450.886   0.209    0.835
rank               9.043     27.272   0.332    0.740
log_rank          -8.682     20.098  -0.432    0.666
gre:log_gre       63.223    158.090   0.400    0.689
gpa:log_gpa      211.328    951.822   0.222    0.824
rank:log_rank     -2.982      9.824  -0.304    0.761

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 237.37  on 399  degrees of freedom
Residual deviance: 127.63  on 390  degrees of freedom
AIC: 147.63

Number of Fisher Scoring iterations: 11
```

```
fitting null model for pseudo-r2
> pseudo_r2 <- pR2(model)

#fitting null model for pseudo-r2
> print(pseudo_r2)


         llh      llhNull          G2     McFadden         r2ML
     r2CU
 -64.8982065 -118.6861026  107.5757923    0.4531946    0.2358105
0.5268671

> plot(allEffects(model))
```




```
> #Area Under the Curve (AUC)
> #Sensitivity and specificity
> # ROC Curve and AUC

> roc_obj <- roc(data$admit, fitted(model))

Setting levels: control = 0, case = 1
Setting direction: controls < cases

> auc(roc_obj)
Area under the curve: 0.9332

> plot(roc_obj, main = "ROC Curve", col = "blue", lwd = 2)
```




```
> #Training and Testing Model
> # Make the dependent variable binary (factor)
> data$admit <- as.factor(data$admit)

># Split into training (80%) and testing (20%) sets

> set.seed(123)  # For reproducibility
> train_index <- sample(seq_len(nrow(data)), size = 0.8 * nrow(data))
> train_data <- data[train_index, ]
> test_data <- data[-train_index, ]

> #Fit the model using the training data.

> # Train the logistic regression model

> log_model <- glm(admit ~ gre + gpa + rank, data = train_data, fami
ly = binomial)

> # Summary of the model

> #summary(log_model)

> # Predict probabilities on the test set
> test_data$pred_prob
<- predict(log_model, newdata = test_data, type = "response")

># Convert probabilities to binary predictions (threshold = 0.5)
> test_data$pred_class <- ifelse(test_data$pred_prob > 0.5, 1, 0)
> # Create confusion matrix
> conf_matrix <- confusionMatrix(as.factor(test_data$pred_class), te
st_data$admit)
> # Print the confusion matrix

> print(conf_matrix)Confusion Matrix and Statistics

          Reference
Prediction  0  1
```

```
         0  7  1
         1  3 69

               Accuracy : 0.95
                 95% CI : (0.8769, 0.9862)
    No Information Rate : 0.875
    P-Value [Acc > NIR] : 0.02237

                  Kappa : 0.75

 Mcnemar's Test P-Value : 0.61708

            Sensitivity : 0.7000
            Specificity : 0.9857
         Pos Pred Value : 0.8750
         Neg Pred Value : 0.9583
             Prevalence : 0.1250
         Detection Rate : 0.0875
   Detection Prevalence : 0.1000
      Balanced Accuracy : 0.8429

       'Positive' Class : 0

> # Calculate accuracy manually

> accuracy <- mean(test_data$pred_class == test_data$admit)

> print(paste("Accuracy:", round(accuracy, 2)))

[1] "Accuracy: 0.95"

>varImp(model)


      Overall
gre  2.113974
gpa  5.205537
rank 1.423319
```