ALY 6015 71629 – Intermediate Analytics – Week 3 – R Assignment

GLM and Logistic Regression

Name: Hari Priya Ramamoorthy

NUID: 002324226

Date: November 20, 2024

Introduction

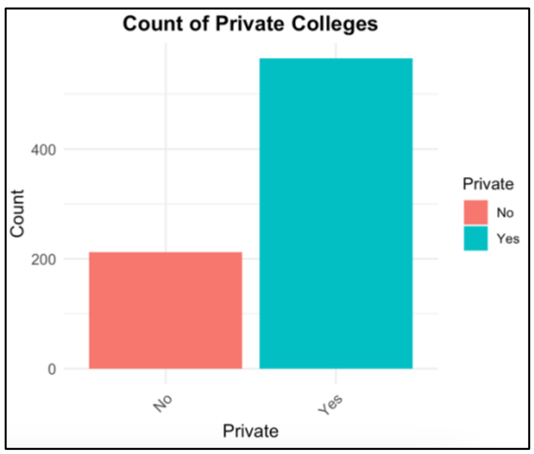
This study analyzes a dataset to classify colleges as either private or non-private using a logistic regression model. The dataset, sourced from the ISLR R package, contains 777 observations and 18 variables that capture various characteristics of colleges such as enrollment, graduation rate, and financial metrics. The goal is to identify key predictors of college type through exploratory data analysis (EDA) and regression techniques.

Data Analysis

Step-1 : Data Cleaning and EDA

The dataset was pre-processed to ensure data quality, with no missing values detected. Column names were standardized for clarity. An initial analysis revealed a higher number of private colleges compared to non-private colleges (Figure 1), emphasizing the importance of representative train-test splitting due to this class imbalance.

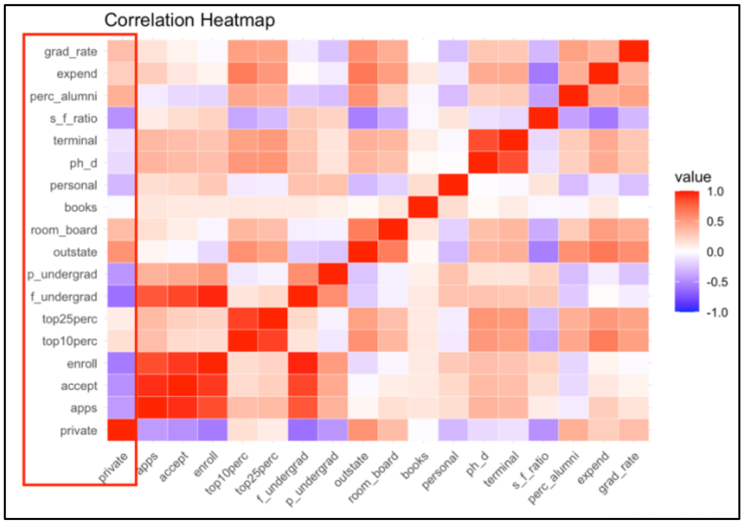
Figure 1: More Private Colleges in Dataset than Non-Private



Step-2: Correlation Analysis With Corrplot() on Numerical Variables

Correlation analysis (Figure 2) identified significant relationships between the target variable (private status) and predictors. Variables such as full-time undergraduates (f_undergrad), out-of-state tuition (outstate), faculty with PhDs (ph_d), and graduation rate (grad_rate) showed strong correlations with private status. High interdependencies among enrollment-related variables, such as enroll, accept, and apps, were also observed.

Figure 2 : Correlation Analysis to Understand the Relationship between the College Characteristics Vs Private



Step-3: Step-wise Regression

Stepwise regression identified the most significant predictors of private status by iteratively minimizing the **Akaike Information Criterion (AIC)**. Key predictors included f_undergrad, outstate, ph_d, grad_rate, and expend (Figure 3). This approach ensured the inclusion of variables with the strongest contribution to the model's predictive power.

Figure 3 : Step-wise Regression to Identify the Important Predictors of Private Colleges

Table: Stepwise Regression: Steps, AIC, and Variables Added/Removed

Step	AIC	Variable Added/Removed
1	912.7486	
2	573.9036	+ f_undergrad
3	307.0524	+ outstate
4	282.6118	+ ph_d
5	273.2785	+ perc_alumni
6	268.4700	+ expend
7	266.5684	+ apps
8	265.5721	+ top10perc

Step-4: Logistic Regression Model

A logistic regression model (Figure 4) was developed based on the significant predictors identified in the stepwise regression. The model achieved an impressive accuracy of 92.45% and less misclassifications(Figure 5) on the training set. The Confusion matrix performance metrics(Figures 6) for the training set are as follows:

- **Accuracy:** 92.45% (proportion of correctly classified instances)
- **Precision:** 81% (ratio of true positives to predicted positives)
- **Recall (Sensitivity):** 99% (ability to correctly identify private colleges)
- **Specificity:** 91.91% (ability to correctly identify non-private colleges)

Figure 4 : Logistic Model Equation based on Step-wise parameters

```
> summary(logistic_model)

Call:
glm(formula = private ~ f_undergrad + outstate + grad_rate +
    ph_d + expend, family = binomial, data = train_set)

Coefficients:
              Estimate Std. Error z value Pr(>|z|)
(Intercept) -5.436e-01  1.153e+00  -0.472 0.637216
f_undergrad -5.793e-04  8.153e-05  -7.105 1.21e-12 ***
outstate     7.521e-04  1.130e-04   6.656 2.82e-11 ***
grad_rate    1.561e-02  1.219e-02   1.281 0.200159
ph_d        -6.733e-02  1.832e-02  -3.676 0.000237 ***
expend       1.275e-04  1.092e-04   1.167 0.243140
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 641.97  on 542  degrees of freedom
Residual deviance: 180.23  on 537  degrees of freedom
AIC: 192.23

Number of Fisher Scoring iterations: 7
```

Figure 5 : Confusion Matrix of Training Set

```
> print(conf_matrix_train)
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 145  35
         1   6 357
```

Figure 6 : Model Performance Metrics of Training Set

```
Table: Confusion Matrix Metrics (Train Set)

|            |Metric               |  Value|
|:-----------|:--------------------|------:|
|Accuracy    |Accuracy             | 0.9245|
|Precision   |Precision            | 0.8056|
|Recall      |Recall (Sensitivity) | 0.9603|
|Specificity |Specificity          | 0.9107|
>
```

Step-5: Logistic Regression Model – Test Set

The confusion matrix for the test set indicated that there were only 1 False Positive (FP) and 14 False Negatives (FN). This demonstrates that the model successfully classified most colleges with minimal misclassification. When applied to the test set, the model showed a classification accuracy of **93.59%**, with the following performance metrics (Figure 8):

- **Accuracy**: 93.59%
- **Precision**: 85.71%
- **Recall (Sensitivity)**: 98%
- **Specificity**: 91.91%

Figure 7 : Confusion Matrix of Training Set

```
Table: Confusion Matrix Metrics (Test Set)

|            |Metric               |  Value|
|:-----------|:--------------------|------:|
|Accuracy    |Accuracy             | 0.9359|
|Precision   |Precision            | 0.8108|
|Recall      |Recall (Sensitivity) | 0.9836|
|Specificity |Specificity          | 0.9191|
>
```

Figure 8 : Model Performance Metrics of Training Set

```
> print(conf_matrix_test)
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0  60  14
         1   1 159
```

Step-6: Interpretation of the ROC Curve:

The ROC curve (Figure 9) and AUC (Area Under the Curve) value were plotted to assess the model's discriminative power. The ROC curve shows a clear distinction between the true positive rate and the false positive rate, confirming the model's ability to balance sensitivity and specificity effectively. The AUC value was found to be **0.9865**, indicating excellent model performance. A high AUC indicates that the model is very effective at distinguishing between private and non-private colleges.

Figure 9 : ROC Curve of Model

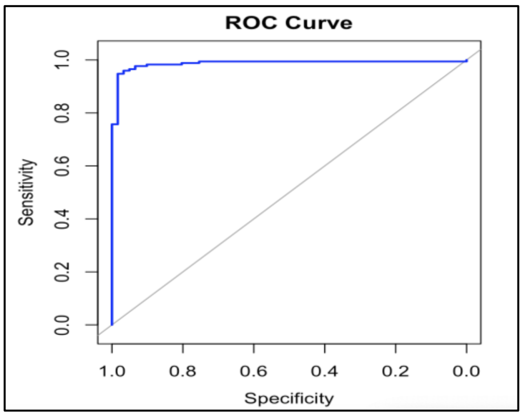
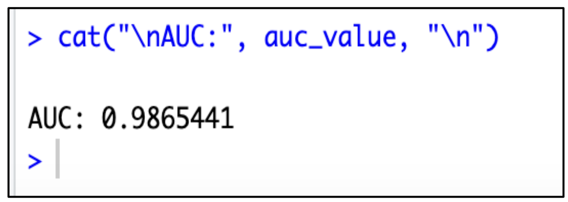


Figure 10 : AUC Value



Key Insights and Recommendations

1. **Significant Predictors:** Variables such as f_undergrad, outstate, ph_d, grad_rate, and expend are critical in predicting whether a college is private.
2. **Model Performance:** The logistic regression model explains over 92% of the variance in the data, providing reliable classifications with an accuracy of 93.59% on the test set.
3. **Misclassification Analysis:**

   - **False Positives (FP):** The model only made 14 FP, which misclassified a non-private college as private. The **Recall (Sensitivity)** of **98%** suggests a strong emphasis on minimizing **False Negatives**, making sure that private colleges are not misclassified as non-private.This minor error could lead to miscommunications in marketing or funding allocation.

   - **False Negatives (FN):** The model made 1 FNs, misclassifying private colleges as non-private. These errors may affect enrollment predictions and marketing strategies, particularly for private institutions.

The ideal scenario is minimizing both **False Positives** and **False Negatives**. However, the decision on which is more critical depends on the broader business context and the potential consequences of each type of misclassification.

Conclusion

The logistic regression model effectively classifies colleges as private or non-private based on key characteristics. With a high AUC of **0.9865** and an accuracy of **93.59%** on the test set, the model demonstrates strong predictive performance. While the model performs well overall, addressing **False Negatives (FN)**, where private colleges are misclassified as non-private, could improve its real-world applicability, particularly for decision-making in areas like marketing and enrollment strategies.

APPENDIX

R Code

```
# load libraries and data
#install.packages('ISLR','lubridate','janitor','tidyr','pROC','caret','reshape2', 'kableExtra')
library(ISLR)
library(dplyr)
library(tidyr)
library(ggplot2)
library(lubridate)
library(janitor)
library(reshape2)
library(ggplot2)
library(pROC)
library(caret)
library(kableExtra)

# Function to create a glimpse table
create_glimpse_table <- function(df) {
  tibble(
    Column_Name = names(df),
    Data_Type = sapply(df, class),
    Example_Value = sapply(df, function(x) if (length(x) > 0) x[1] else NA)
```

```
  )
}

#Glimpse of Data
raw_data_glimpse<-create_glimpse_table(College)
# Summary of Numeric columns
summary(College)
names(College)

# Check for missing values
sum(is.na(College)) # no missing values
College<-janitor::clean_names(College)
names(College)

############################################## EDA
##############################################
# Count of Private colleges (Yes/No)
College %>%
  count(private) %>%
  ggplot(aes(x = private, y = n, fill = private)) +
  geom_bar(stat = "identity") +
  labs(title = "Count of Private Colleges", x = "Private", y = "Count") +
  theme_minimal() +
  theme(
    plot.title = element_text(hjust = 0.5, size = 14, face = "bold"),
    axis.title.x = element_text(size = 12),
    axis.title.y = element_text(size = 12),
    axis.text.x = element_text(size = 10, angle = 45, hjust = 1),
    axis.text.y = element_text(size = 10)
  )


# Compute correlations for numeric variables
numeric_vars <- select(College, where(is.numeric))
cor_matrix <- cor(numeric_vars, use = "complete.obs")

# Visualize correlations
cor_df <- melt(cor_matrix)
ggplot(cor_df, aes(Var1, Var2, fill = value)) +
  geom_tile(color = "white") +
```

```
  scale_fill_gradient2(low = "blue", high = "red", mid = "white", midpoint = 0, limit = c(-1, 1)) +
  labs(title = "Correlation Heatmap", x = "", y = "") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))

##############################################  EDA
##############################################

#################################### Select Predictors : Step-Wise Regression
########################################
# Convert 'Private' to a binary variable (1 for 'Yes', 0 for 'No')
College$private <- ifelse(College$private == "Yes", 1, 0)

# Fit a logistic regression model with stepwise selection
model_initial <- glm(private ~ 1, data = College, family = "binomial")  # Null model
model_full <- glm(private ~ ., data = College, family = "binomial")    # Full model

# Perform stepwise selection
model_step <- step(model_initial, scope = formula(model_full), direction = "both")

# Summary of the final model
summary(model_step)
# Extract the anova table from the stepwise model, which contains the step information.
step_info <- model_step$anova

# Create a data frame to store step information for plotting
stepwise_df <- data.frame(
  Step = 1:nrow(step_info),
  AIC = step_info$AIC,
  Variable = step_info$Step
)

# Use kable to display the table in a report-friendly format
kable(stepwise_df,
      caption = "Stepwise Regression: Steps, AIC, and Variables Added/Removed",
      col.names = c("Step", "AIC", "Variable Added/Removed"),
      format = "markdown")

# Print the table for reporting
print(stepwise_df)
```

```
#################################### Step-Wise Regression
#######################################

#################################### Fit Logistic Regression Model - Train Set
####################################
# Split the data into training and testing sets
set.seed(42)  # For reproducibility
train_indices <- createDataPartition(1:nrow(College), size = 0.7 * nrow(College))
train_set <- College[train_indices, ]
test_set <- College[-train_indices, ]

# Fit logistic regression model using glm()
logistic_model <- glm(private ~ f_undergrad + outstate + grad_rate + ph_d + expend ,
                      data = train_set,
                      family = binomial)

# Summary of the model
summary(logistic_model)


# Predict on the train set
train_pred_prob <- predict(logistic_model, train_set, type = "response")
train_pred <- ifelse(train_pred_prob > 0.8, 1, 0)

# Confusion Matrix
conf_matrix_train <- confusionMatrix(as.factor(train_pred), as.factor(train_set$private))

# Print the confusion matrix
print(conf_matrix_train)


# Extract metrics
metrics_train <- data.frame(
  Metric = c("Accuracy", "Precision", "Recall (Sensitivity)", "Specificity"),
  Value = c(
    conf_matrix_train$overall["Accuracy"],
    conf_matrix_train$byClass["Precision"], # Precision
    conf_matrix_train$byClass["Recall"],   # Recall
    conf_matrix_train$byClass["Specificity"]    # Specificity
  )
```

```
)

# Print as a kable
kable(metrics_train, col.names = c("Metric", "Value"), digits = 4, caption = "Confusion Matrix
Metrics (Train Set)")
##################################### Fit Logistic Regression Model - Train
Set#####################################

##################################### Model Confusion Matrix & ROC Curve - Test Set
#####################################
# Predict on the test set
test_set$predicted_prob <- predict(logistic_model, newdata = test_set, type = "response")
test_set$predicted_class <- ifelse(test_set$predicted_prob > 0.8, 1, 0)

# Confusion Matrix for the Test Set
conf_matrix_test <- confusionMatrix(as.factor(test_set$predicted_class),
as.factor(test_set$private))

# Print the confusion matrix for the test set
print(conf_matrix_test)

# Extract metrics
metrics_test <- data.frame(
  Metric = c("Accuracy", "Precision", "Recall (Sensitivity)", "Specificity"),
  Value = c(
    conf_matrix_test$overall["Accuracy"],
    conf_matrix_test$byClass["Precision"], # Precision
    conf_matrix_test$byClass["Recall"],   # Recall
    conf_matrix_test$byClass["Specificity"]    # Specificity
  )
)

# Print as a kable
kable(metrics_test, col.names = c("Metric", "Value"), digits = 4, caption = "Confusion Matrix
Metrics (Test Set)")


# ROC Curve
roc_curve <- roc(test_set$private, test_set$predicted_prob)
```

```
# Plot the ROC curve
plot(roc_curve, col = "blue", main = "ROC Curve")
#abline(a = 0, b = 1, lty = 2, col = "gray")

# Calculate AUC
auc_value <- auc(roc_curve)
cat("\nAUC:", auc_value, "\n")

###################################### Model Confusion Matrix & ROC Curve - Test Set
######################################
```

Reference

Ises, S., & Tukey, J. W. (2020). *College dataset*. In *ISLR: Introduction to statistical learning with applications in R* (R package). Retrieved from https://rdrr.io/cran/ISLR/man/College.html

DigitalOcean. (2020, February 25). *Confusion matrix in R*. DigitalOcean. Retrieved November 20, 2024, from https://www.digitalocean.com/community/tutorials/confusion-matrix-in-r