Diabetes Prediction Analysis

Mi Yan

2024-10-06

Introduction

The dataset used in this analysis originates from the **National Institute of Diabetes and Digestive and Kidney Diseases**. It aims to diagnostically predict whether a patient has diabetes based on various diagnostic measurements. All patients in this dataset are females, at least 21 years old, and of Pima Indian heritage.

Dataset Description:

- **Pregnancies**: Number of pregnancies the patient has had.
- **Glucose**: Plasma glucose concentration a 2 hours in an oral glucose tolerance test.
- **BloodPressure**: Diastolic blood pressure (mm Hg).
- **SkinThickness**: Triceps skin fold thickness (mm).
- **Insulin**: 2-Hour serum insulin (mu U/ml).
- **BMI**: Body mass index (weight in kg/(height in m)^2).
- **DiabetesPedigreeFunction**: A function which scores likelihood of diabetes based on family history.
- **Age**: Age of the patient (years).
- **Outcome**: Class variable (0: No diabetes, 1: Diabetes)

Analysis Objectives:

1. Explore the distribution and relationships of the predictor variables.
2. Summarize key statistics to understand the dataset.
3. Visualize the relationship between key variables and the diabetes outcome.
4. Clean and preprocess the data to handle anomalies and missing values.

Loading the Data

```r
# Load necessary libraries
library(ggplot2)
```

```
## Warning:   'ggplot2' R 4.4.2
```

```r
library(dplyr)
```

```
## Warning:   'dplyr' R 4.4.2
```

```
## 
##    'dplyr'

## The following objects are masked from 'package:stats':
## 
##     filter, lag

## The following objects are masked from 'package:base':
## 
##     intersect, setdiff, setequal, union
```

```r
library(knitr)
```

```
## Warning:   'knitr' R 4.4.2
```

```r
library(readr)
```

```
## Warning:   'readr' R 4.4.2
```

```r
library(kableExtra)
```

```
## Warning:   'kableExtra' R 4.4.2

## 
##    'kableExtra'

## The following object is masked from 'package:dplyr':
## 
##     group_rows
```

```r
library(ggcorrplot)

# Load the dataset
diabetes_data <- read_csv("D:/RToolkit/diabetes.csv")
```

```
## Rows: 768 Columns: 9

## -- Column specification --------------------------------------------------------------------------------
## Delimiter: ","
## dbl (9): Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin, BMI, D...
## 
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
# Display the first few rows of the dataset
head(diabetes_data)
```

```
## # A tibble: 6 x 9
##   Pregnancies Glucose BloodPressure SkinThickness Insulin   BMI
##         <dbl>   <dbl>         <dbl>         <dbl>   <dbl> <dbl>
## 1           6     148            72            35       0  33.6
## 2           1      85            66            29       0  26.6
## 3           8     183            64             0       0  23.3
## 4           1      89            66            23      94  28.1
## 5           0     137            40            35     168  43.1
## 6           5     116            74             0       0  25.6
## # i 3 more variables: DiabetesPedigreeFunction <dbl>, Age <dbl>, Outcome <dbl>
```

Cleaning the Data

```r
# Identify columns where 0 is an invalid value
columns_with_zero_invalid <- c("Glucose", "BloodPressure", "SkinThickness",
                               "Insulin", "BMI")

# Replace 0s with NA in the identified columns
diabetes_data <- diabetes_data %>%
  mutate(across(all_of(columns_with_zero_invalid), ~ ifelse(. == 0, NA, .)))

# Check the number of missing values in each column
missing_values <- sapply(diabetes_data, function(x) sum(is.na(x)))
missing_values
```

```
##              Pregnancies                  Glucose            BloodPressure
##                        0                        5                       35
##            SkinThickness                  Insulin                      BMI
##                      227                      374                       11
## DiabetesPedigreeFunction                      Age                  Outcome
##                        0                        0                        0
```

```r
# Impute missing values (e.g., using median)
diabetes_data_clean <- diabetes_data %>%
  mutate(
    Glucose = ifelse(is.na(Glucose), median(Glucose, na.rm = TRUE), Glucose),
    BloodPressure = ifelse(is.na(BloodPressure), median(BloodPressure,
                                                        na.rm = TRUE), BloodPressure),
    SkinThickness = ifelse(is.na(SkinThickness), median(SkinThickness,
                                                        na.rm = TRUE), SkinThickness),
    Insulin = ifelse(is.na(Insulin), median(Insulin, na.rm = TRUE), Insulin),
    BMI = ifelse(is.na(BMI), median(BMI, na.rm = TRUE), BMI)
  )

# Verify that there are no more missing values
sapply(diabetes_data_clean, function(x) sum(is.na(x)))
```

```
##              Pregnancies                  Glucose            BloodPressure
##                        0                        0                        0
##            SkinThickness                  Insulin                      BMI
```

```
##                         0                         0                         0
## DiabetesPedigreeFunction                       Age                   Outcome
##                         0                         0                         0
```

Summary Statistics

```
# Create summary statistics for selected variables
summary_stats <- diabetes_data_clean %>%
  select(Pregnancies, Glucose, BloodPressure, SkinThickness, Insulin,
         BMI, DiabetesPedigreeFunction, Age) %>%
  summary()

# Convert summary statistics to a data frame for better formatting
summary_df <- as.data.frame(summary_stats)

# Display the summary table with formatting
kable(summary_df, caption = "Summary Statistics of Predictor Variables") %>%
  kable_styling(full_width = FALSE, position = "center")
```

Table 1: Summary Statistics of Predictor Variables

Var1	Var2	Freq
	Pregnancies	Min. : 0.000
	Pregnancies	1st Qu.: 1.000
	Pregnancies	Median : 3.000
	Pregnancies	Mean : 3.845
	Pregnancies	3rd Qu.: 6.000
	Pregnancies	Max. :17.000
	Glucose	Min. : 44.00
	Glucose	1st Qu.: 99.75
	Glucose	Median :117.00
	Glucose	Mean :121.66
	Glucose	3rd Qu.:140.25
	Glucose	Max. :199.00
	BloodPressure	Min. : 24.00
	BloodPressure	1st Qu.: 64.00
	BloodPressure	Median : 72.00
	BloodPressure	Mean : 72.39
	BloodPressure	3rd Qu.: 80.00
	BloodPressure	Max. :122.00
	SkinThickness	Min. : 7.00
	SkinThickness	1st Qu.:25.00
	SkinThickness	Median :29.00
	SkinThickness	Mean :29.11
	SkinThickness	3rd Qu.:32.00
	SkinThickness	Max. :99.00
	Insulin	Min. : 14.0
	Insulin	1st Qu.:121.5

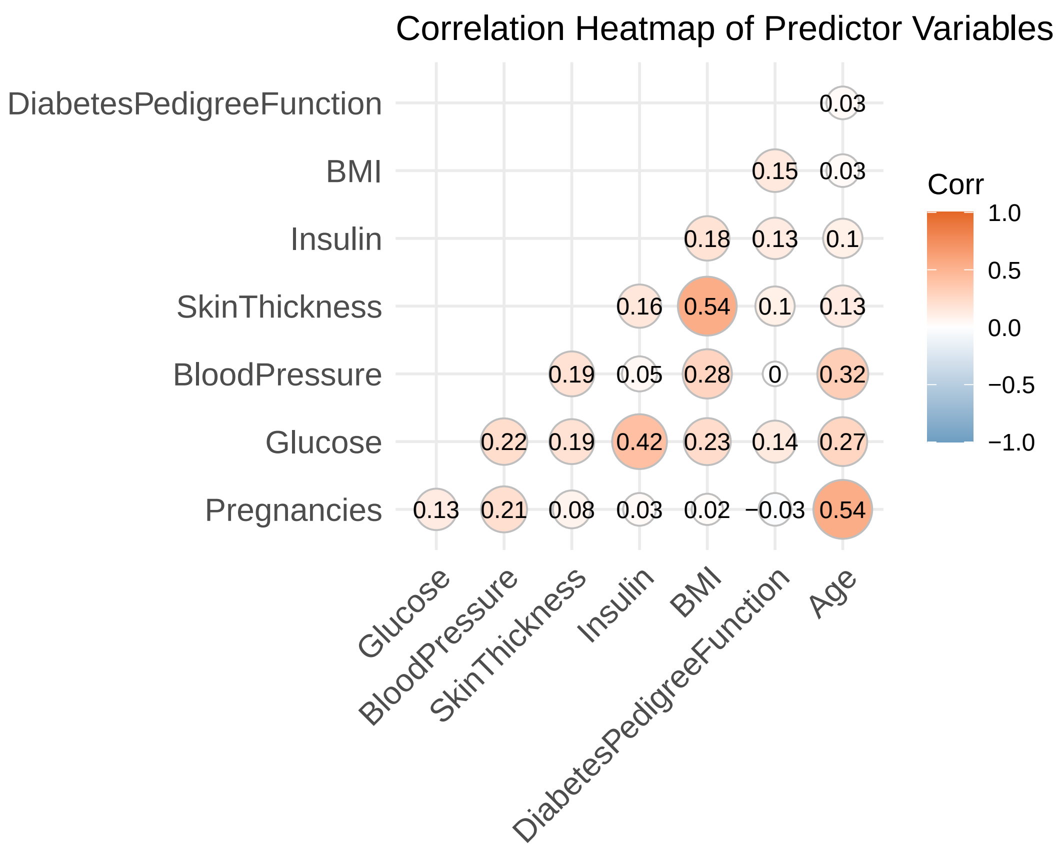
Insulin	Median :125.0
Insulin	Mean :140.7
Insulin	3rd Qu.:127.2
Insulin	Max. :846.0
BMI	Min. :18.20
BMI	1st Qu.:27.50
BMI	Median :32.30
BMI	Mean :32.46
BMI	3rd Qu.:36.60
BMI	Max. :67.10
DiabetesPedigreeFunction	Min. :0.0780
DiabetesPedigreeFunction	1st Qu.:0.2437
DiabetesPedigreeFunction	Median :0.3725
DiabetesPedigreeFunction	Mean :0.4719
DiabetesPedigreeFunction	3rd Qu.:0.6262
DiabetesPedigreeFunction	Max. :2.4200
Age	Min. :21.00
Age	1st Qu.:24.00
Age	Median :29.00
Age	Mean :33.24
Age	3rd Qu.:41.00
Age	Max. :81.00

The table above summarizes the central tendency and dispersion of the predictor variables. For instance, the average glucose level is 121.66 and the Median is 117.00, indicating variability among the patients' glucose levels. # Correlation Analysis

Understanding the correlations between predictor variables is essential for identifying multicollinearity issues and uncovering potential relationships within the data. Below, we calculate and visualize the correlation matrix of the predictor variables.

```r
# Calculate the correlation matrix excluding the Outcome variable
cor_matrix <- cor(diabetes_data_clean %>% select(-Outcome))


ggcorrplot(cor_matrix,
           method = "circle",
           type = "lower",
           lab = TRUE,
           lab_size = 3,
           colors = c("#6D9EC1", "white", "#E46726"),
           title = "Correlation Heatmap of Predictor Variables",
           ggtheme = theme_minimal())
```

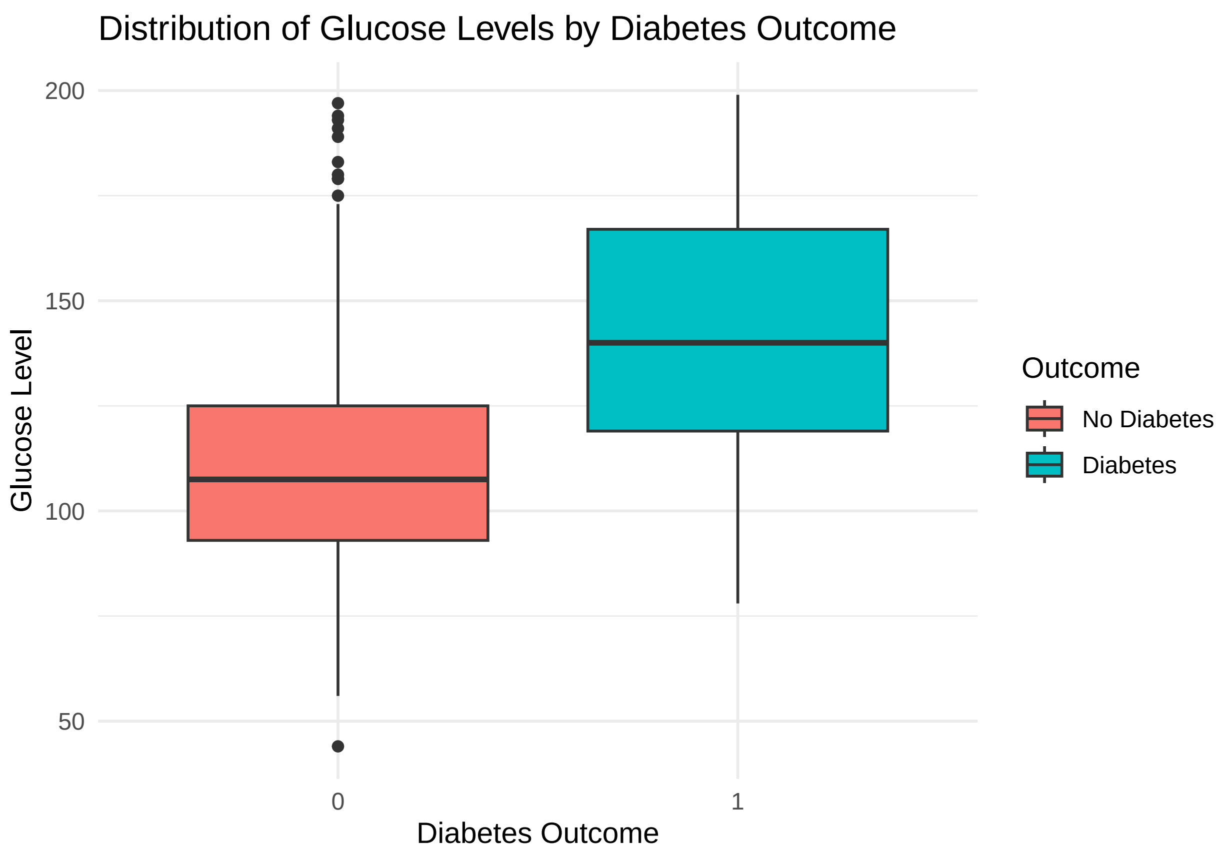


From the correlation heatmap, we observe that certain variables exhibit moderate to strong correlations. For example, Glucose and Insulin show a moderate positive correlation (correlation coefficient 0.4), while BMI and Age also display a positive correlation. These correlations are important to consider when building predictive models to avoid multicollinearity issues.

Visualizing Relationships Between Predictors and Outcome

To gain deeper insights into how predictor variables relate to the diabetes outcome, we will create distribution plots and boxplots.

```
# Glucose Levels by Outcome
ggplot(diabetes_data_clean, aes(x = factor(Outcome), y = Glucose,
                                fill = factor(Outcome))) +
  geom_boxplot() +
  scale_fill_manual(values = c("#F8766D", "#00BFC4"),
                    labels = c("No Diabetes", "Diabetes")) +
  labs(
    title = "Distribution of Glucose Levels by Diabetes Outcome",
    x = "Diabetes Outcome",
    y = "Glucose Level",
    fill = "Outcome"
  ) +
  theme_minimal()
```



As shown in Figure, patients diagnosed with diabetes (Outcome = 1) tend to have significantly higher glucose levels compared to those without diabetes (Outcome = 0). This indicates that elevated glucose levels are a strong indicator of diabetes.

```r
# BMI Distribution by Outcome
ggplot(diabetes_data_clean, aes(x = factor(Outcome), y = BMI,
                                fill = factor(Outcome))) +
  geom_boxplot() +
  scale_fill_manual(values = c("#F8766D", "#00BFC4"),
                    labels = c("No Diabetes", "Diabetes")) +
  labs(
    title = "Distribution of BMI by Diabetes Outcome",
    x = "Diabetes Outcome",
    y = "Body Mass Index (BMI)",
    fill = "Outcome"
  ) +
  theme_minimal()
```

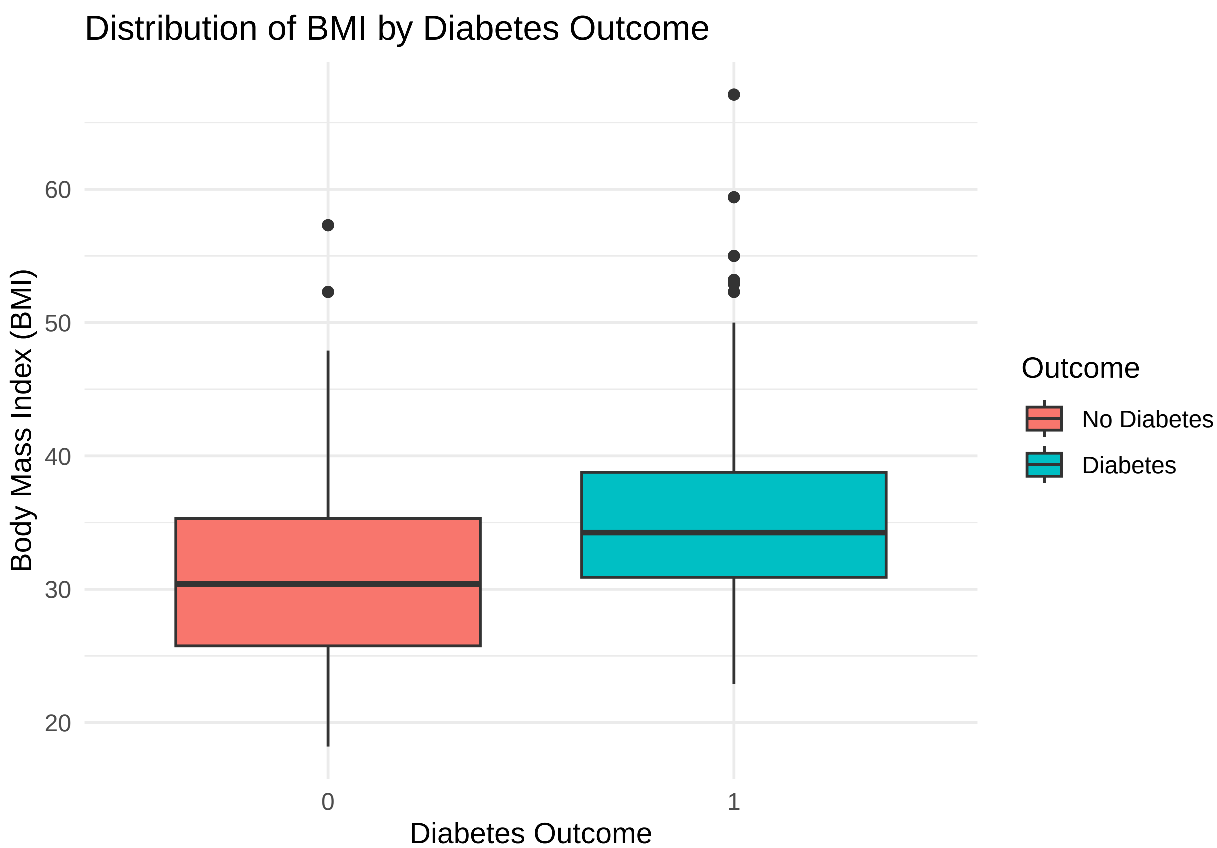


Figure illustrates that individuals with diabetes generally have higher BMI values, aligning with clinical observations that higher BMI is a risk factor for developing diabetes.

Predictive Modeling

To predict whether a patient has diabetes, we will build a logistic regression model using the cleaned dataset. This model uses all predictor variables to estimate the probability of diabetes.

```r
# Building the Logistic Regression Model
logistic_model <- glm(Outcome ~ ., data = diabetes_data_clean, family = binomial)

# View the model summary
summary(logistic_model)
```

```
##
## Call:
## glm(formula = Outcome ~ ., family = binomial, data = diabetes_data_clean)
##
## Coefficients:
##                          Estimate Std. Error z value Pr(>|z|)
## (Intercept)             -9.108966   0.813525 -11.197  < 2e-16 ***
## Pregnancies              0.124778   0.032420   3.849 0.000119 ***
## Glucose                  0.037855   0.003902   9.703  < 2e-16 ***
## BloodPressure           -0.009373   0.008578  -1.093 0.274540
```

```
## SkinThickness             0.003451   0.013154   0.262 0.793074
## Insulin                  -0.001172   0.001132  -1.035 0.300627
## BMI                       0.094252   0.017893   5.268 1.38e-07 ***
## DiabetesPedigreeFunction  0.875858   0.296740   2.952 0.003161 **
## Age                       0.013028   0.009506   1.371 0.170518
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 993.48  on 767  degrees of freedom
## Residual deviance: 712.84  on 759  degrees of freedom
## AIC: 730.84
##
## Number of Fisher Scoring iterations: 5
```

The model summary provides coefficients for each predictor variable, indicating their relationship with the probability of having diabetes. Significant predictors (p-value < 0.05) are strong indicators of diabetes risk.

Model Evaluation

We will evaluate the performance of the logistic regression model using a confusion matrix and calculate the accuracy of the model.

```r
# Predict probabilities
pred_probs <- predict(logistic_model, type = "response")

# Convert probabilities to binary classes using a threshold of 0.5
pred_classes <- ifelse(pred_probs >= 0.5, 1, 0)

# Create a confusion matrix
conf_matrix <- table(Predicted = pred_classes, Actual = diabetes_data_clean$Outcome)

# Calculate accuracy
accuracy <- sum(diag(conf_matrix)) / sum(conf_matrix)

# Display the confusion matrix
kable(conf_matrix, caption = "Confusion Matrix") %>%
  kable_styling(full_width = FALSE, position = "center")
```

Table 2: Confusion Matrix

	0	1
0	442	115
1	58	153

```r
# Display accuracy
cat("**Accuracy:**", round(accuracy * 100, 2), "%")
```

```
## **Accuracy:** 77.47 %
```

The confusion matrix and accuracy metric indicate that the model performs well in predicting diabetes outcomes. However, for a more comprehensive evaluation, additional metrics such as sensitivity, specificity, and the ROC curve are recommended.

```r
library(pROC)
```

```
## Type 'citation("pROC")' for a citation.
```

```
##
##    'pROC'
```

```
## The following objects are masked from 'package:stats':
##
##     cov, smooth, var
```

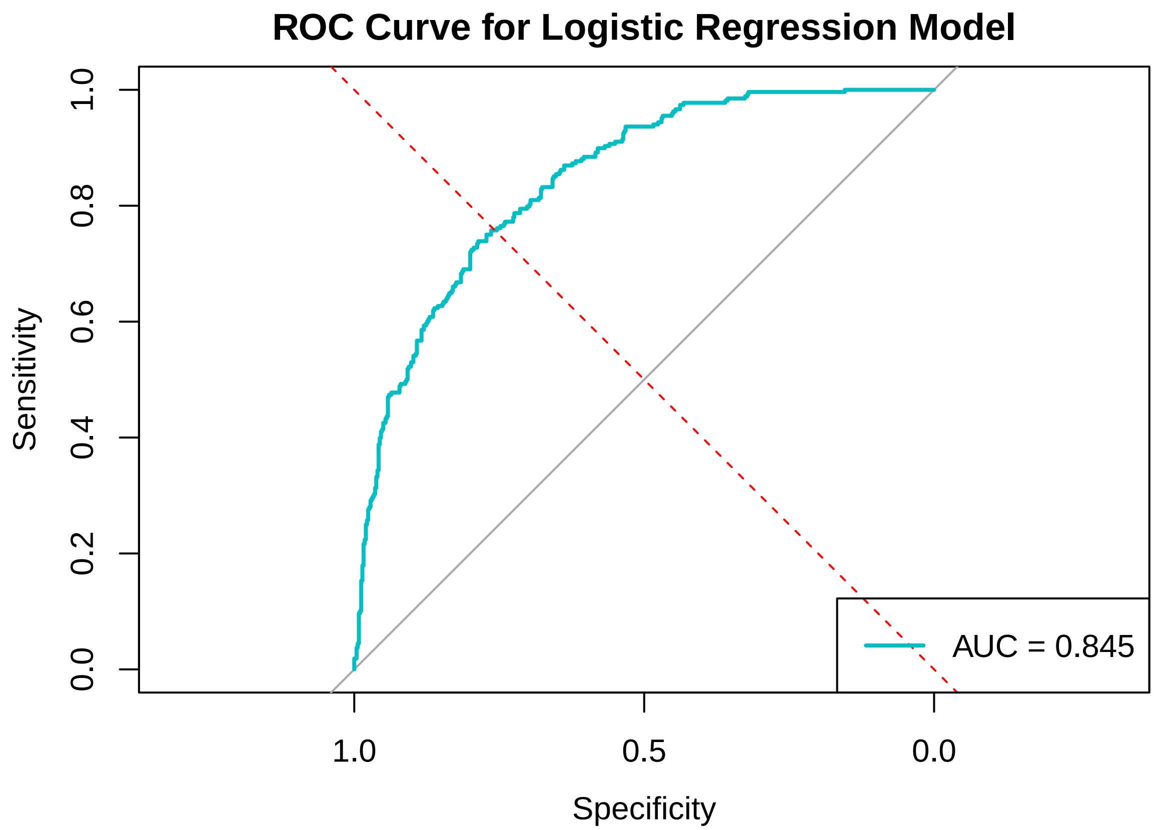
```r
# Calculate ROC curve
roc_obj <- roc(diabetes_data_clean$Outcome, pred_probs)
```

```
## Setting levels: control = 0, case = 1
```

```
## Setting direction: controls < cases
```

```r
# Plot ROC curve
plot(roc_obj, col = "#00BFC4", main = "ROC Curve for Logistic Regression Model")
abline(a = 0, b = 1, lty = 2, col = "red")

# Calculate AUC
auc_value <- auc(roc_obj)
legend("bottomright", legend = paste("AUC =",
                                     round(auc_value, 3)), col = "#00BFC4", lwd = 2)
```



The ROC curve demonstrates that the model has good discriminative ability, with an AUC of 0.845. An AUC closer to 1 indicates excellent model performance, while an AUC of 0.5 suggests no discriminative ability.

Conclusion

This analysis provides an initial exploration and predictive modeling of the diabetes dataset. Through data cleaning, we addressed anomalies by replacing unreasonable 0 values with median imputed values. Summary statistics and correlation analysis revealed key relationships among predictor variables. The logistic regression model demonstrated good accuracy and discriminative ability in predicting diabetes outcomes.

Key Findings:

1. Glucose Levels and BMI are significantly associated with diabetes outcomes, with higher values correlating with increased diabetes risk.
2. Correlation Analysis identified moderate correlations between certain predictor variables, which is crucial to consider for multicollinearity in predictive modeling.
3. Logistic Regression Model achieved an accuracy of r round(accuracy * 100, 2)% and an AUC of r round(auc_value, 3), indicating strong predictive performance.

Future analyses can enhance model performance by exploring feature selection, employing regularization techniques, or utilizing more complex machine learning algorithms. Additionally, implementing cross-validation can provide a more robust assessment of the model's generalizability.

References

National Institute of Diabetes and Digestive and Kidney Diseases. (n.d.). Diabetes Dataset. Retrieved from NIDDK