Statistical Learning Lab

SHIVAM KUMAR (22IM10037)

Assignment - 2

Logistic Regression Assignment

1. **Load the dataset "diabetes.csv". Display first few rows of the dataset.**

   Ans: Dataset loaded through the following "Environment-> Import Dataset"

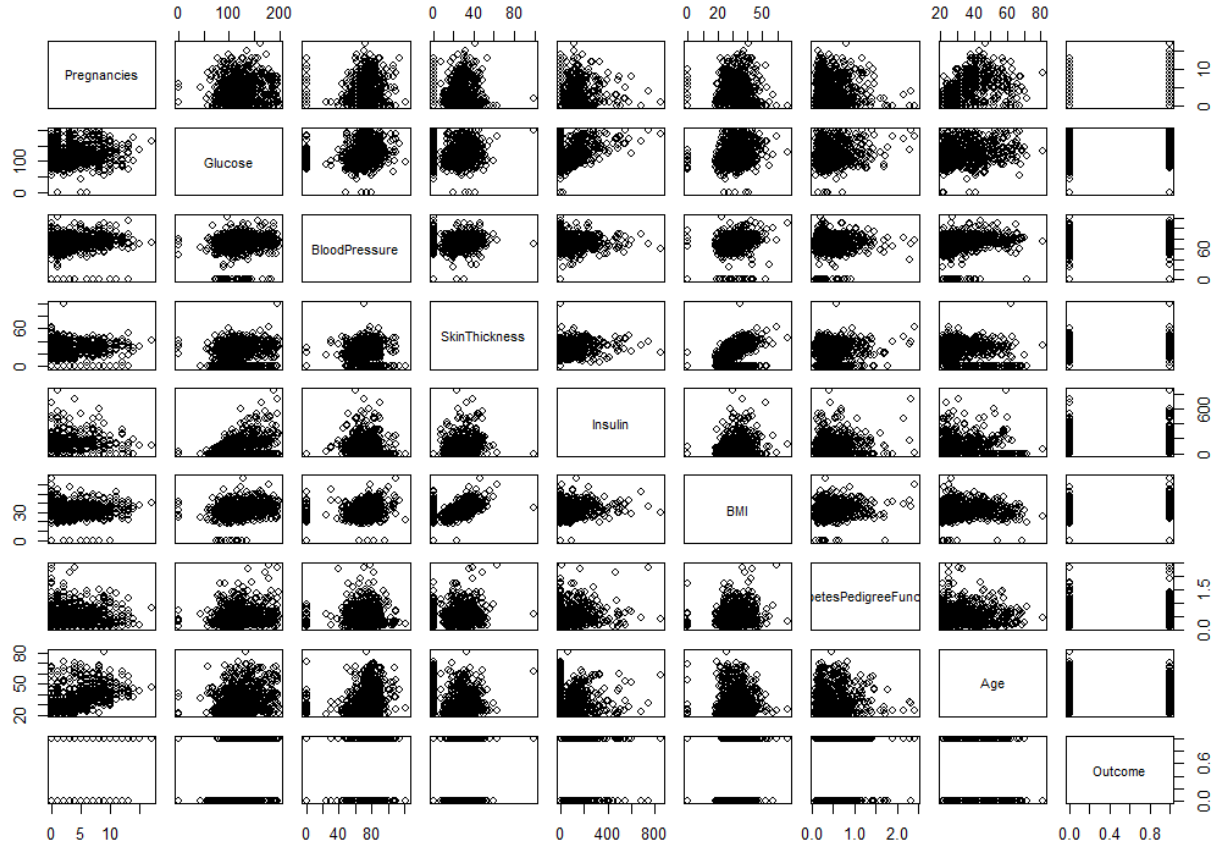
   Code to display the first few rows of the dataset:

```
> head(diabetes)
  Pregnancies Glucose BloodPressure SkinThickness Insulin  BMI DiabetesPedigreeFunction Age Outcome
1           6     148            72            35       0 33.6                    0.627  50       1
2           1      85            66            29       0 26.6                    0.351  31       0
3           8     183            64             0       0 23.3                    0.672  32       1
4           1      89            66            23      94 28.1                    0.167  21       0
5           0     137            40            35     168 43.1                    2.288  33       1
6           5     116            74             0       0 25.6                    0.201  30       0
```

2. **Perform preliminary analysis to show how the variables are related to each other. Use scatter plot, box plot etc. to visualize how different variables impact the "Outcome" variable.**

   Code to show relation between different variables:

   plot(diabetes)

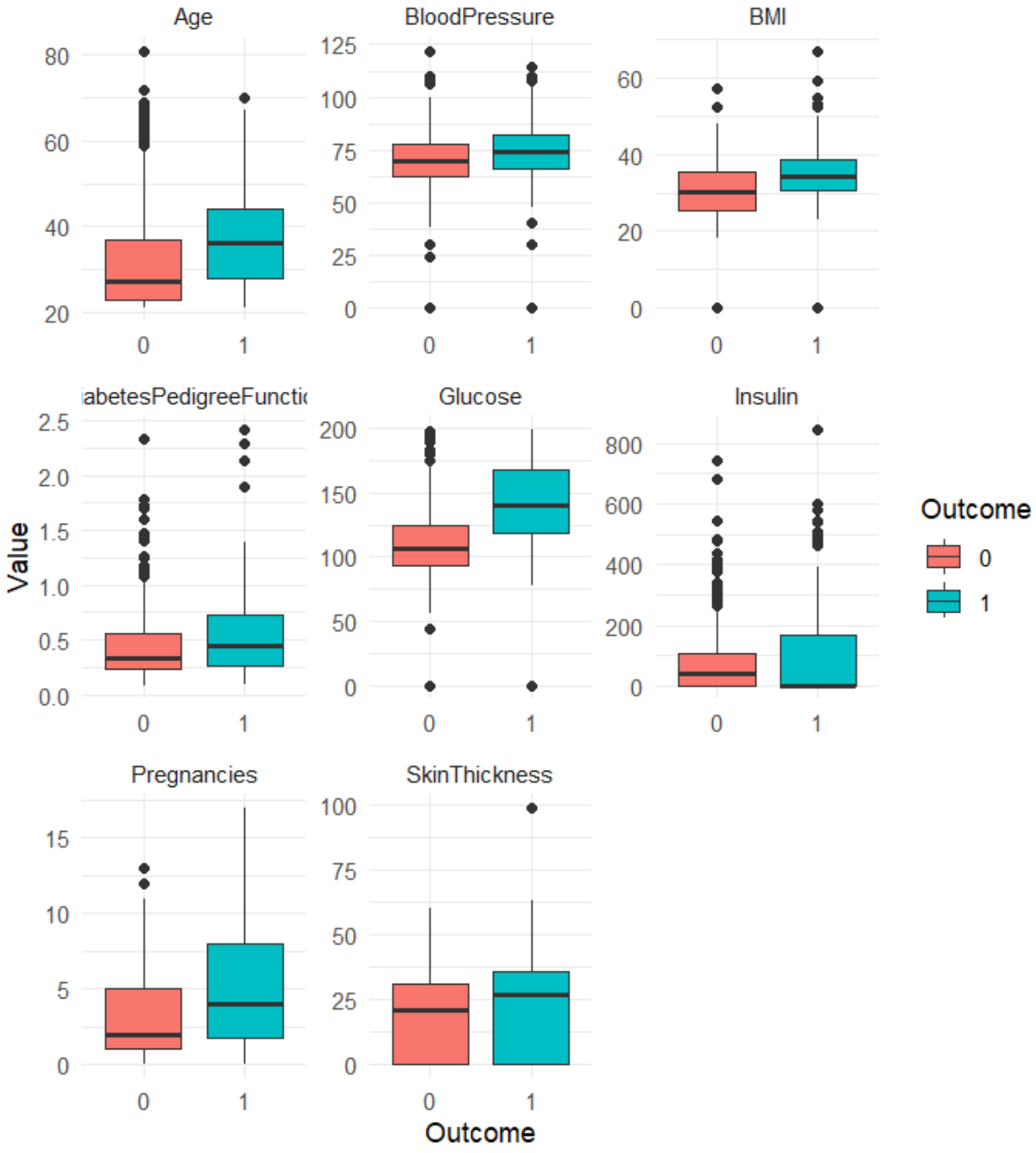


Code for visualize how different variables impact the “Outcome” variable :

```
library(tidyr)
library(dplyr)
library(ggplot2)
library(caret)
boxplot_data <- diabetes %>%
  pivot_longer(cols = -Outcome, names_to = "Variable", values_to = "Value")


ggplot(boxplot_data, aes(x = as.factor(Outcome), y = Value, fill = as.factor(Outcome))) +
  geom_boxplot() +
  facet_wrap(~Variable, scales = "free") +
  theme_minimal() +
  labs(x = "Outcome", y = "Value", fill = "Outcome")
```

Output:



3. Randomly sample 80% of the data as training data and rest as test data. Fit a Logistic Regression model with all the predictors on training data. From the summary which factors seem to be significant? Explain how the predictors impact the log-odds ratio of diagnosed with diabetes (Outcome)

Code :

```
dim(diabetes)
training_size<-0.8*dim(diabetes)
print(training_size)
train <- sample(nrow(diabetes),614) #614 is 80% of 768 i.e. the number of rows in the dataset
train_diabetes<-diabetes[train,]
test_diabetes<-diabetes[-train,]
model_M1 <- glm(Outcome ~ ., data = train_diabetes, family = binomial)
summary(model_M1)
exp(coef(model_M1))
```

Output:

```
Coefficients:
                          Estimate Std. Error z value Pr(>|z|)
(Intercept)             -8.4812440  0.8023267 -10.571  < 2e-16 ***
Pregnancies              0.0934948  0.0356840   2.620  0.00879 **
Glucose                  0.0336925  0.0042295   7.966 1.64e-15 ***
BloodPressure           -0.0135171  0.0056951  -2.373  0.01762 *
SkinThickness            0.0009183  0.0077730   0.118  0.90595
Insulin                 -0.0012372  0.0009743  -1.270  0.20411
BMI                      0.0924932  0.0171769   5.385 7.26e-08 ***
DiabetesPedigreeFunction 0.9908016  0.3372480   2.938  0.00330 **
Age                      0.0236222  0.0103151   2.290  0.02202 *
```

As we can see, Pregnancies, Glucose, BloodPressure, BMI, Age and DiabetesPedigreeFunction seems to be significant (p-value<0.05)

Variable-Specific Interpretation(Log Odds Ratio):

Pregnancies (0.090120, p = 0.01137):

- A one-unit increase in the number of pregnancies increases the log-odds of diabetes diagnosis by 0.090120. This is a statistically significant predictor.

Glucose (0.034376, p < 2e-16)*:

- A one-unit increase in glucose levels significantly increases the log-odds of diabetes diagnosis by 0.034376. This is a highly significant predictor.

BloodPressure (-0.011485, p = 0.05280):

- A one-unit increase in blood pressure slightly decreases the log-odds of diabetes diagnosis by 0.011485. However, this predictor is only marginally significant (. indicates a p-value close to 0.05).

SkinThickness (0.004239, p = 0.58830):

- This predictor has a small positive coefficient but is not statistically significant (p > 0.05), meaning its impact on the Outcome is uncertain.

Insulin (-0.001433, p = 0.16079):

- Insulin has a negligible negative coefficient and is not statistically significant.

BMI (0.094775, p = 1.91e-08)*:

- A one-unit increase in BMI significantly increases the log-odds of diabetes diagnosis by 0.094775. This is a strong and highly significant predictor.

DiabetesPedigreeFunction (0.895416, p = 0.00968):

- A one-unit increase in this metric increases the log-odds of diabetes diagnosis by 0.895416. This variable is statistically significant.

Age (0.024800, p = 0.01955):

- A one-year increase in age increases the log-odds of diabetes diagnosis by 0.024800. Age is statistically significant.

4. From the model fitted in problem 3, derive confusion matrix, accuracy, and F1-score on test data.

Code:

```
test_predictions <- predict(model_M1, newdata = test_diabetes, type = "response")
test_pred_class <- ifelse(test_predictions > 0.5, 1, 0)
summary(test_pred_class)

# Confusion matrix
confusion_matrix <- table(Predicted = test_pred_class, Actual = test_diabetes$Outcome)
confusion_matrix

#Accuracy
accuracy <- sum(diag(confusion_matrix)) / sum(confusion_matrix)
accuracy

# F1-Score
precision <- confusion_matrix[2, 2] / sum(confusion_matrix[2, ])
recall <- confusion_matrix[2, 2] / sum(confusion_matrix[, 2])
f1_score <- 2 * (precision * recall) / (precision + recall)
f1_score
```

Output:

```
> confusion_matrix <- table(Predicted = test_pred_class, Actual = test_diabetes$Outcome)
> confusion_matrix
         Actual
Predicted  0  1
        0 90 21
        1 10 33
>
> #Accuracy
> accuracy <- sum(diag(confusion_matrix)) / sum(confusion_matrix)
> accuracy
[1] 0.7987013
>
> # F1-Score
> precision <- confusion_matrix[2, 2] / sum(confusion_matrix[2, ])
> recall <- confusion_matrix[2, 2] / sum(confusion_matrix[, 2])
> f1_score <- 2 * (precision * recall) / (precision + recall)
> f1_score
[1] 0.6804124
```

5. Let's call the model fitted in problem 3 M1. Now choose predictors "Pregnancies", "Glucose" and "BMI" and fit a model (M2). Compare the deviances among these two models and perform hypothesis test to show whether M1 is significantly more informative than M2.

Code:

```
# Fit model M1 with all predictors and model M2 with selected predictors
model_M1 <- glm(Outcome ~ ., data = train_diabetes, family = binomial)
model_M2 <- glm(Outcome ~ Pregnancies + Glucose + BMI, data = train_diabetes, family = binomial)

# Deviances of the models
deviance_M1 <- model_M1$deviance
deviance_M2 <- model_M2$deviance
deviance_M1-deviance_M2
# Hypothesis testing to compare models
anova(model_M1, model_M2, test = "Chisq")
```

Output:

```
> # Deviances of the models
> deviance_M1 <- model_M1$deviance
> deviance_M2 <- model_M2$deviance
> deviance_M1-deviance_M2
[1] -20.38457
> # Hypothesis testing to compare models
> anova(model_M1, model_M2, test = "Chisq")
Analysis of Deviance Table

Model 1: Outcome ~ Pregnancies + Glucose + BloodPressure + SkinThickness +
    Insulin + BMI + DiabetesPedigreeFunction + Age
Model 2: Outcome ~ Pregnancies + Glucose + BMI
  Resid. Df Resid. Dev Df Deviance Pr(>Chi)
1       605     583.52
2       610     603.91 -5  -20.385 0.001058 **
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Inference:

M1 has a lower deviance, meaning it explains the data better.

The p-value is **0.001058**, which is highly significant (less than 0.01). This shows that the improvement in model fit from M2 to M1 is statistically significant.

Conclusion:

M1, the full model, is **significantly more informative** than M2. The additional predictors (e.g., BloodPressure, SkinThickness, Insulin, DiabetesPedigreeFunction, and Age) collectively improve the model's ability to explain variations in the Outcome variable.

In practical terms, removing these predictors would result in a noticeable drop in predictive performance.