install.packages("pROC")

install.packages("aod")

install.packages("ResourceSelection")

# Load required libraries

library(tibble)

library(dplyr)

library(aod)

library(ResourceSelection)

library(pROC)

# Set random seed for reproducibility

set.seed(100)

# Define the number of rows in the dataset

n <- 100

# Generate synthetic data for each variable

gender <- sample(c("Male", "Female"), n, replace = TRUE)

age <- sample(18:65, n, replace = TRUE)

income <- as.integer(rnorm(n, mean = 50000, sd = 15000))

brand\_attitude <- sample(1:10, n, replace = TRUE)

purchase\_intention <- sample(c("Low", "Moderate", "High"), n, replace = TRUE)

actual\_purchase <- sample(c("Yes", "No"), n, replace = TRUE)

message\_type <- sample(c("Personalized","Non-Personalized"), n, replace = TRUE)

product\_category <- sample(c("Electronics", "Apparel", "Beauty"), n, replace = TRUE)

prior\_brand\_loyalty <- sample(c("Low", "Moderate", "High"), n, replace = TRUE)

# Create a tibble (DataFrame) with the generated data

data <- data.frame(

Gender = gender,

Age = age,

Income = income,

Brand\_Attitude = brand\_attitude,

Purchase\_Intention = purchase\_intention,

Actual\_Purchase = actual\_purchase,

Message\_Type = message\_type,

Product\_Category = product\_category,

Prior\_Brand\_Loyalty = prior\_brand\_loyalty

)

# Display the first few rows of the generated dataset

head(data)

summary(data)

#Converting Categorical Variables into factors

data$Gender <- as.factor(data$Gender)

data$Purchase\_Intention <- as.factor(data$Purchase\_Intention)

data$Actual\_Purchase <- as.factor(data$Actual\_Purchase)

data$Message\_Type <- as.factor(data$Message\_Type)

data$Prior\_Brand\_Loyalty <- as.factor(data$Prior\_Brand\_Loyalty)

# Split data into training and testing sets

set.seed(100)

train\_indices <- sample(1:nrow(data), 0.7 \* nrow(data))

train\_data <- data[train\_indices, ]

test\_data <- data[-train\_indices, ]

# Fit logistic regression model

model1 <- glm(Message\_Type ~ Gender + Age + Income + Brand\_Attitude + Purchase\_Intention + Actual\_Purchase + Product\_Category + Prior\_Brand\_Loyalty,

data = train\_data, family = "binomial")

summary(model1)

# Predict on test data

predictions <- predict(model1, newdata = test\_data, type = "response")

# Convert predicted probabilities to binary responses

predicted\_responses <- ifelse(predictions > 0.5, "Yes", "No")

# Create confusion matrix

conf\_matrix <- table(predicted\_responses, test\_data$Message\_Type)

print(conf\_matrix)

# Calculate accuracy

accuracy <- sum(diag(conf\_matrix)) / sum(conf\_matrix)

print(paste("Accuracy:",accuracy))

#Wald's Test

wald.test(vcov(model1),model1$coefficients,Terms = 2)

wald.test(vcov(model1),model1$coefficients,Terms = 3)

wald.test(vcov(model1),model1$coefficients,Terms = 4)

wald.test(vcov(model1),model1$coefficients,Terms = 5)

wald.test(vcov(model1),model1$coefficients,Terms = 6)

wald.test(vcov(model1),model1$coefficients,Terms = 8)

wald.test(vcov(model1),model1$coefficients,Terms = 9)

wald.test(vcov(model1),model1$coefficients,Terms = 11)

#H-L Test

hltest <- hoslem.test(model1$y,fitted(model1),g=10)

hltest

#Calculate Precision

precision = conf\_matrix[2,2]/sum(conf\_matrix[2,])

precision

#Calculate Sensitivity

sensitivity = conf\_matrix[2,2]/sum(conf\_matrix[2,])

sensitivity

#Calculate Specificity

specificity = conf\_matrix[1,1]/sum(conf\_matrix[1,])

specificity

# Calculate AUC

roc\_Curve <- roc(test\_data$Message\_Type,predictions)

plot(roc\_Curve, main = "ROC Curve", col = "blue")

roc\_Curve