Source Code

```
# ---------- Data Set Normalization ----------
# Import relevant libraries
library(tidyverse)
library(dslabs)

library(caret)

# Read "exams.csv" from Rstudio directory
exams <- read_csv(file = "exams.csv")
str(exams)

# Remove columns 2 to 4 from the data set
exams <-  exams[, -2:-4]
str(exams)

# Convert the class variable to a factor (male/female)
exams$gender <- as.factor(exams$gender)
str(exams)

# Convert the 'Test preparation course' values to numerical
exams$`test preparation course` <- as.numeric(as.factor(exams$`test preparation course`))
str(exams)

# Removes rows with NA values and then checks for any null values
exams <- na.omit(exams)
any(is.na(exams))

# View final processed data set
View(exams)
```

```
#split to training and testing
set.seed(1) # For reproducibility

# train_indices <- sample(1:nrow(exams), 0.7 * nrow(exams))

# Splits data distribution (gender) evenly for training and testing
train_indices <- exams$gender %>%
  createDataPartition(p = 0.7, list = FALSE)
train_data <- exams[train_indices, ]
test_data <- exams[-train_indices, ]

print(train_data)
print(test_data)



# ---------- Logistic Regression ----------
# Imports for graph display
library(ggplot2)
library(dplyr)
library(patchwork)

theme_set(theme_bw())

# ---------- Prediction Using Test Data ----------
model <- glm( gender ~ `test preparation course` + `math score` + `reading score` + `writing score`,
        data = train_data, family = binomial)
summary(model)$coef

probabilities <- model %>% predict(test_data, type = "response")
head(probabilities)
```

```
contrasts(test_data$gender)


predicted.classes <- ifelse(probabilities > 0.5, "male", "female")
head(predicted.classes)


mean(predicted.classes == test_data$gender)


# ---------- Prediction Using Training Data ----------
model <- glm( gender ~ `test preparation course` + `math score` + `reading score` + `writing score`,
        data = train_data, family = binomial)
summary(model)$coef


probabilities <- model %>% predict(train_data, type = "response")
head(probabilities)


contrasts(train_data$gender)


predicted.classes <- ifelse(probabilities > 0.5, "male", "female")
head(predicted.classes)


mean(predicted.classes == train_data$gender)


# ---------- Training Data Graphs ----------
# Plotting Probability Graph with Math Score
plot1 <- train_data %>%
 mutate(prob = ifelse(gender == "male", 1, 0)) %>%
 ggplot(aes(`math score`, prob)) +
 geom_point(alpha = 0.2) +
 geom_smooth(method = "glm", method.args = list(family = "binomial")) +
```

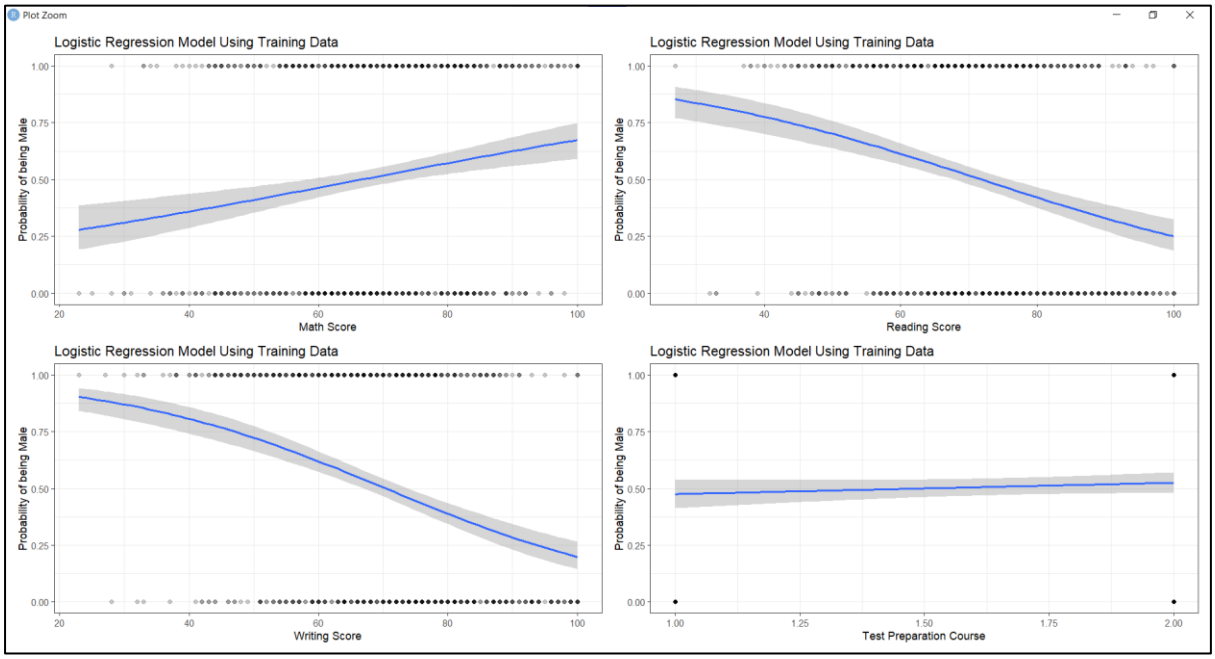
```
 labs(
  title = "Logistic Regression Model Using Training Data",
  x = "Math Score",
  y = "Probability of being Male"
 )

# Plotting Probability Graph with Reading Score
plot2 <- train_data %>%
 mutate(prob = ifelse(gender == "male", 1, 0)) %>%
 ggplot(aes(`reading score`, prob)) +
 geom_point(alpha = 0.2) +
 geom_smooth(method = "glm", method.args = list(family = "binomial")) +
 labs(
  title = "Logistic Regression Model Using Training Data",
  x = "Reading Score",
  y = "Probability of being Male"
 )

# Plotting Probability Graph with Writing Score
plot3 <- train_data %>%
 mutate(prob = ifelse(gender == "male", 1, 0)) %>%
 ggplot(aes(`writing score`, prob)) +
 geom_point(alpha = 0.2) +
 geom_smooth(method = "glm", method.args = list(family = "binomial")) +
 labs(
  title = "Logistic Regression Model Using Training Data",
  x = "Writing Score",
  y = "Probability of being Male"
 )

# Plotting Probability Graph with Test Preparation Course
```

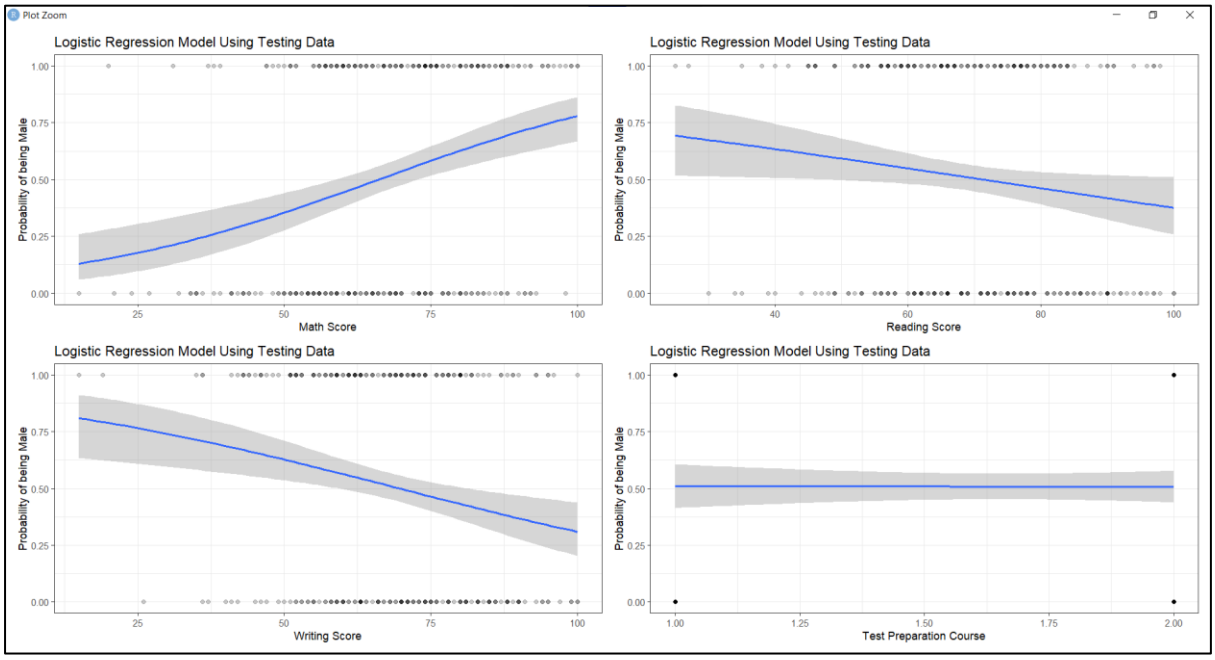
```
plot4 <- train_data %>%
 mutate(prob = ifelse(gender == "male", 1, 0)) %>%
 ggplot(aes(`test preparation course`, prob)) +
 geom_point(alpha = 0.2) +
 geom_smooth(method = "glm", method.args = list(family = "binomial")) +
 labs(
  title = "Logistic Regression Model Using Training Data",
  x = "Test Preparation Course",
  y = "Probability of being Male"
 )

# Displaying all 4 probability graphs at once
plot1 + plot2 + plot3 + plot4

# ---------- Testing Data Graphs ----------
# Plotting Probability Graph with Math Score
plot1 <- test_data %>%
 mutate(prob = ifelse(gender == "male", 1, 0)) %>%
 ggplot(aes(`math score`, prob)) +
 geom_point(alpha = 0.2) +
 geom_smooth(method = "glm", method.args = list(family = "binomial")) +
 labs(
  title = "Logistic Regression Model Using Testing Data",
  x = "Math Score",
  y = "Probability of being Male"
 )

# Plotting Probability Graph with Reading Score
plot2 <- test_data %>%
 mutate(prob = ifelse(gender == "male", 1, 0)) %>%
 ggplot(aes(`reading score`, prob)) +
```

```
 geom_point(alpha = 0.2) +
 geom_smooth(method = "glm", method.args = list(family = "binomial")) +
 labs(
  title = "Logistic Regression Model Using Testing Data",
  x = "Reading Score",
  y = "Probability of being Male"
 )

# Plotting Probability Grap with Writing Score
plot3 <- test_data %>%
 mutate(prob = ifelse(gender == "male", 1, 0)) %>%
 ggplot(aes(`writing score`, prob)) +
 geom_point(alpha = 0.2) +
 geom_smooth(method = "glm", method.args = list(family = "binomial")) +
 labs(
  title = "Logistic Regression Model Using Testing Data",
  x = "Writing Score",
  y = "Probability of being Male"
 )

# Plotting Probability Graph with Test Preparation Course
plot4 <- test_data %>%
 mutate(prob = ifelse(gender == "male", 1, 0)) %>%
 ggplot(aes(`test preparation course`, prob)) +
 geom_point(alpha = 0.2) +
 geom_smooth(method = "glm", method.args = list(family = "binomial")) +
 labs(
  title = "Logistic Regression Model Using Testing Data",
  x = "Test Preparation Course",
  y = "Probability of being Male"
 )
```

```
# Displaying all 4 graphs at once
plot1 + plot2 + plot3 + plot4
```

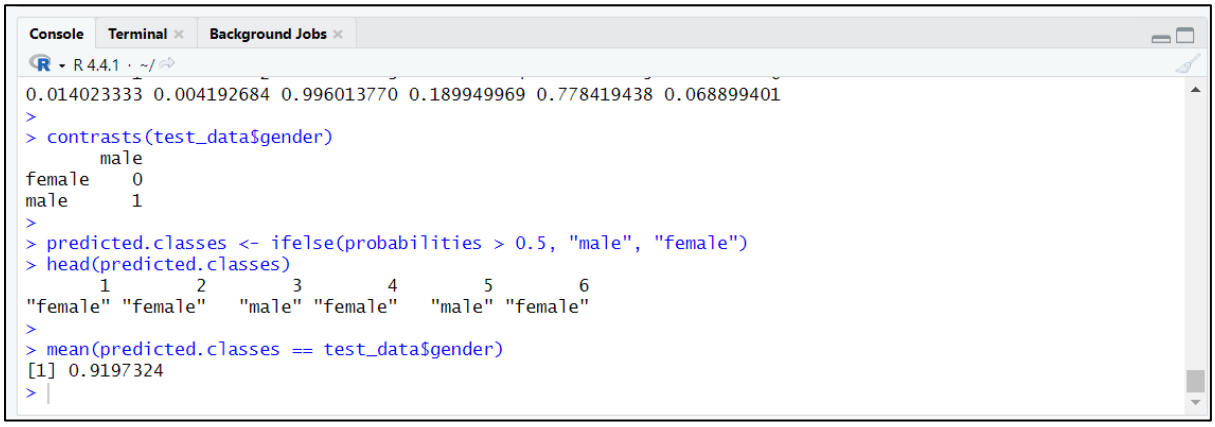
Graph Screenshots





Logistic Regression Accuracy Screenshot

Against Testing Data:



Against Training Data: