**Implement Linear and Logistic Regression**

**AIM:**

To implement Linear and Logistic Regression using R programming in R Studio. **a)Linear regression**

# Sample data

heights <- c(150, 160, 165, 170, 175, 180, 185) weights <- c(55, 60, 62, 68, 70, 75, 80)

# Create a data frame data <- data.frame(heights, weights)

# Fit a linear regression model linear\_model <- lm(weights ~ heights,

data = data)

# Print the summary of the model

print(summary(linear\_model))

# Plotting the data and regression line plot(data$heights, data$weights, main = "Linear Regression: Weight vs. Height", xlab = "Height (cm)", ylab

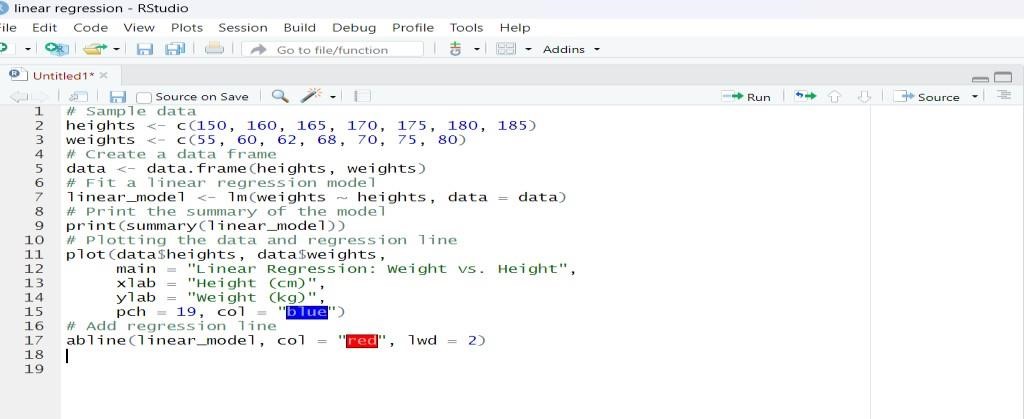
= "Weight (kg)", pch

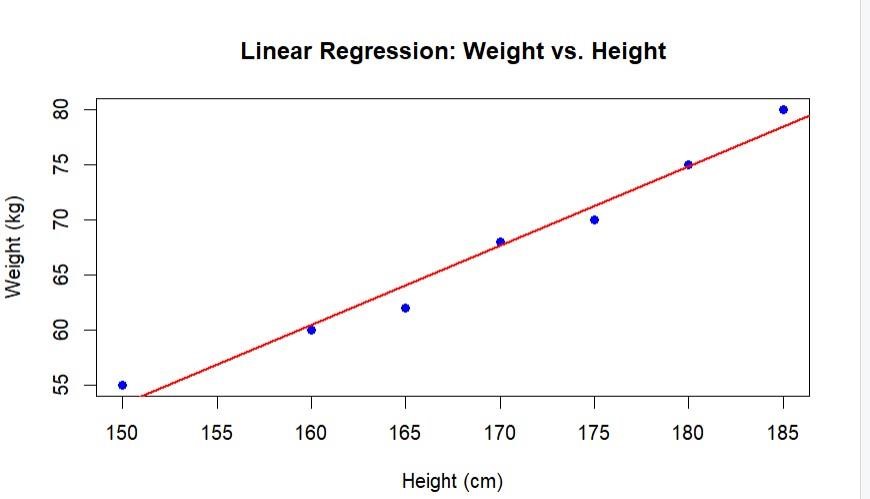
= 19, col = "blue")

# Add regression line

abline(linear\_model, col = "red", lwd = 2)

**OUTPUT:**





**b) Logistic regression**

# Load the dataset

data(mtcars)

# Convert 'am' to a factor (categorical variable) mtcars$am <- factor(mtcars$am, levels = c(0, 1), labels = c("Automatic", "Manual"))

# Fit a logistic regression model logistic\_model <- glm(am

~ mpg, data = mtcars, family = binomial)

# Print the summary of the model

print(summary(logistic\_model))

# Predict probabilities for the logistic model predicted\_probs <- predict(logistic\_model, type =

"response")

# Display the predicted probabilities print(predicted\_probs)

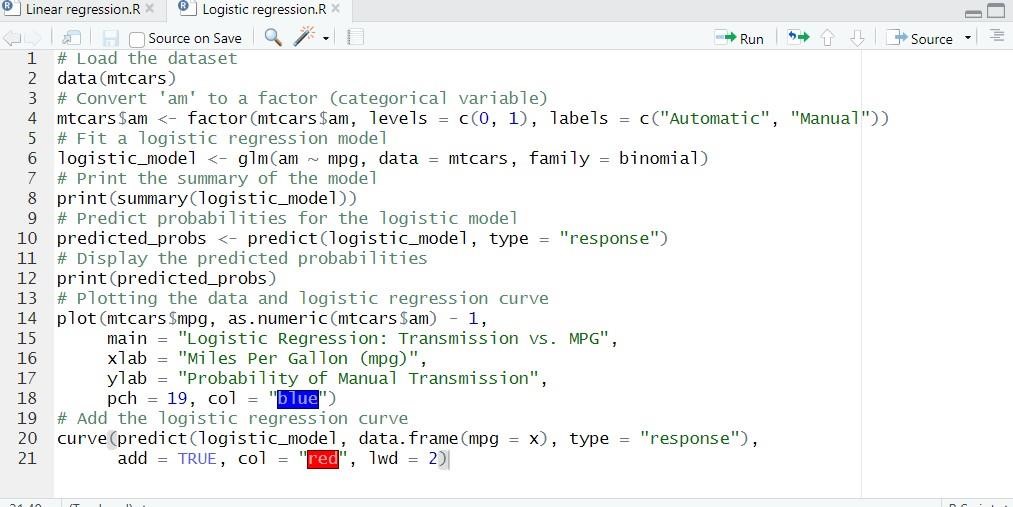
# Plotting the data and logistic regression curve plot(mtcars$mpg, as.numeric(mtcars$am) - 1, main = "Logistic Regression: Transmission vs. MPG", xlab = "Miles Per Gallon (mpg)", ylab

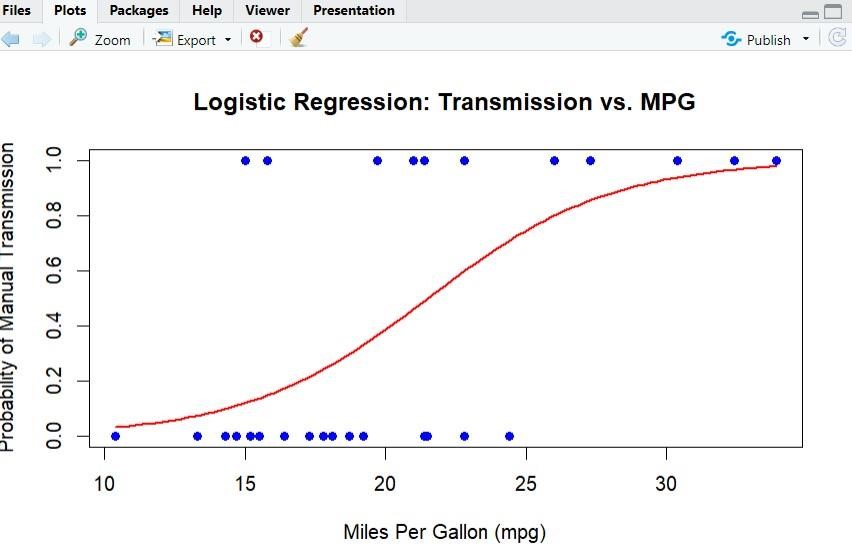
= "Probability of Manual Transmission", pch =

19, col = "blue")

# Add the logistic regression curve curve(predict(logistic\_model, data.frame(mpg = x), type = "response"), add = TRUE, col = "red", lwd = 2)

**OUTPUT:**





**RESULT:**

Thus the implementationLinear and Logistic Regression using R programming in R Studio have been successfully executed.