210701142

Installation guide for R and RStudio

Step 1 – Install R

1. Download the R installer from https://cran.r---project.org/

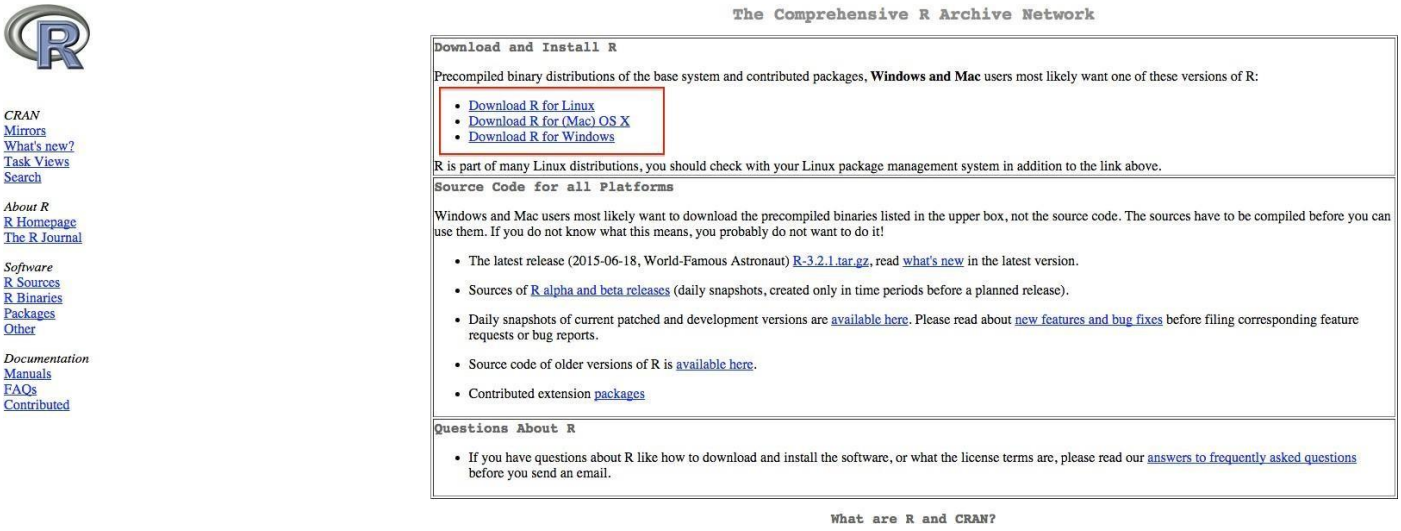


Figure 1. Screenshot of http://cran.csiro.au/

2. Run the installer. Default settings are fine. If you do not have admin rights on your laptop, thenask you local IT support. In that case, it is important that you also ask them to give you full permissions to the R directories. Without this, you will not be able to install additional packages later

Step 2 – Install RStudio

1. Download RStudio: https://www.rstudio.com/products/rstudio/download/

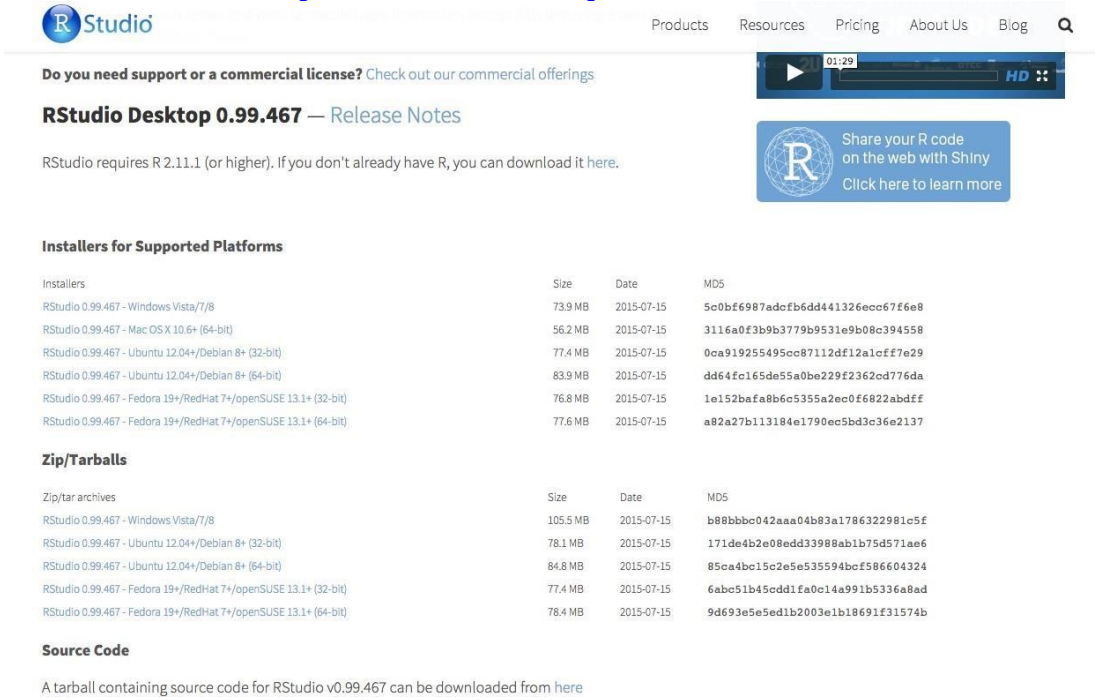


Figure 2. Download RStudio on https://www.rstudio.com/products/rstudio/download/

2. Once the installation of R has completed successfully (and not before), run the RStudio installer.

3. If you do not have administrative rights on your laptop, step 2 may fail. Ask your IT Support or download a pre---built zip archive of RStudio which doesn't need installing. The link for this is towards the bottom of the download page, highlighted in Image 2.

    a. Download the appropriate archive for your system (Windows/Linux only – the Mac version can be installed into your personal "Applications" folder without admin rights).

    b. Double clicking on the zip archive should automatically unpack it on most Windows machines.

Step 3 – Check that R and RStudio are working

1. Open RStudio. It should open a window that looks similar to image 3 below.

2. In the left hand window, by the '>'sign, type '4+5'(without the quotes) and hit enter. An outputline reading '[1] 9' should appear. This means that R and RStudio are working.

3. If this is not successful, contact us or your local IT support for further advice

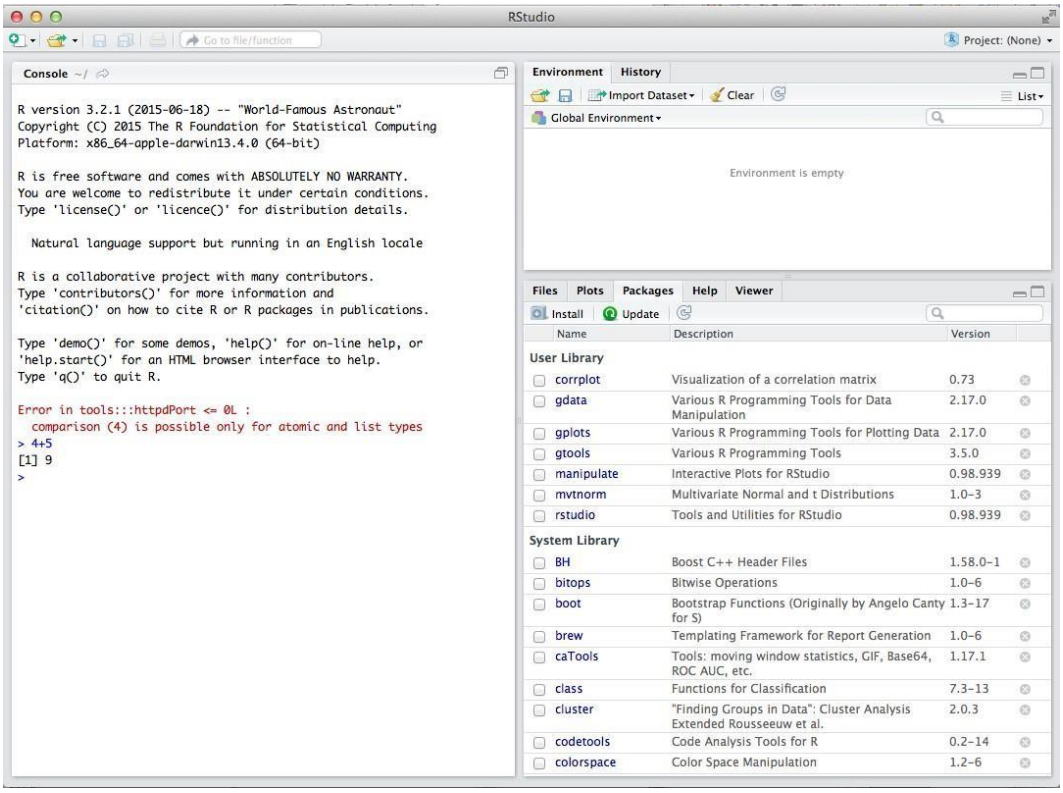


Figure 3. Running R with RStudio

Step 4 – Install R packages required for the workshop

1. Click on the tab ' Packages' then 'Install' as shown in Image 4. Or Tools ---> Install packages.
2. Install the following packages: mixOmics version 6.1.0, mvtnorm, RColorBrewer, corrplot, igraph (see Image 4). For apple mac users, if youare unable to install the mixOmics imported library rgl, you will need to install the XQuartz software first https://www.xquartz.org/

3. Check that the packages are installed by typing ‘library(mixOmics)’ (without thequotes) in the prompt and press enter (see Image 5).
4. Then type ‘sessionInfo()’ and check that mixOmics version 6.1.0 has been installed(image 6).

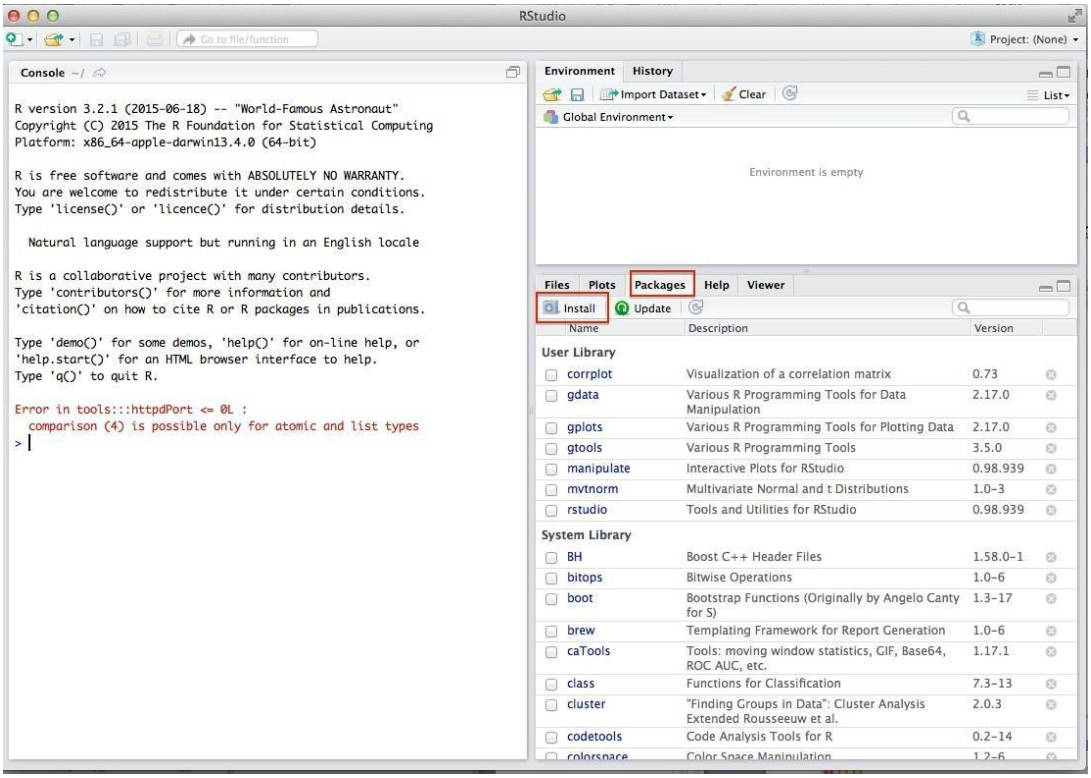


Figure 4. Click on Install to install R packages.

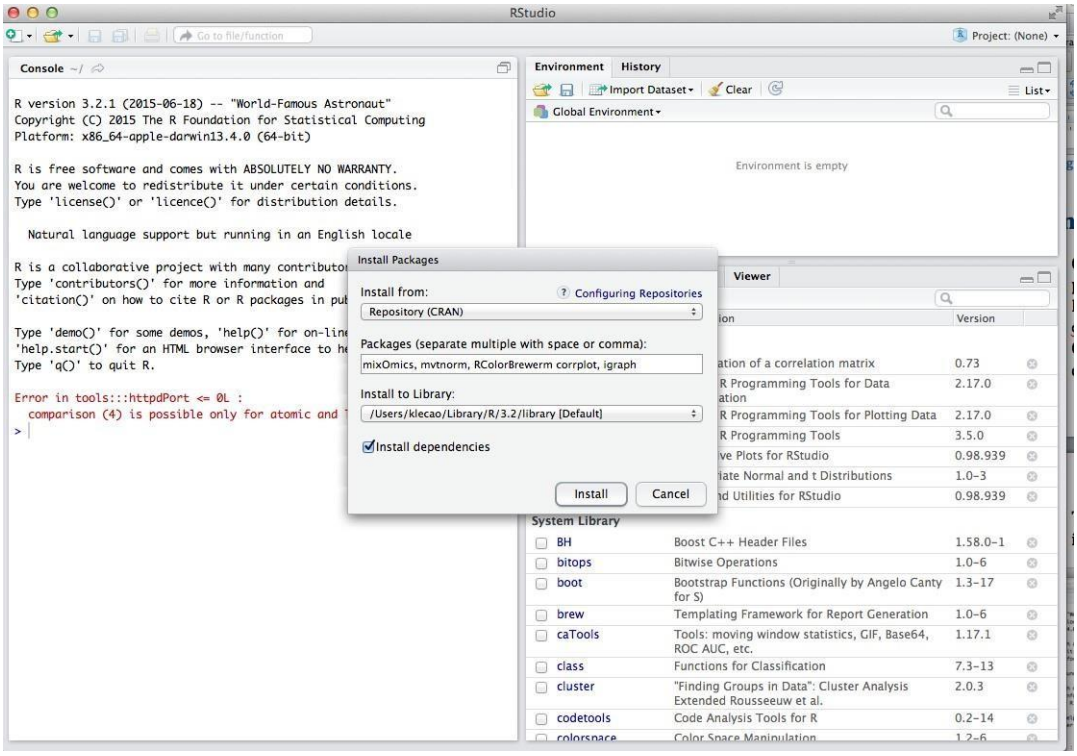


Figure 5. Specify the list of packages to be installed

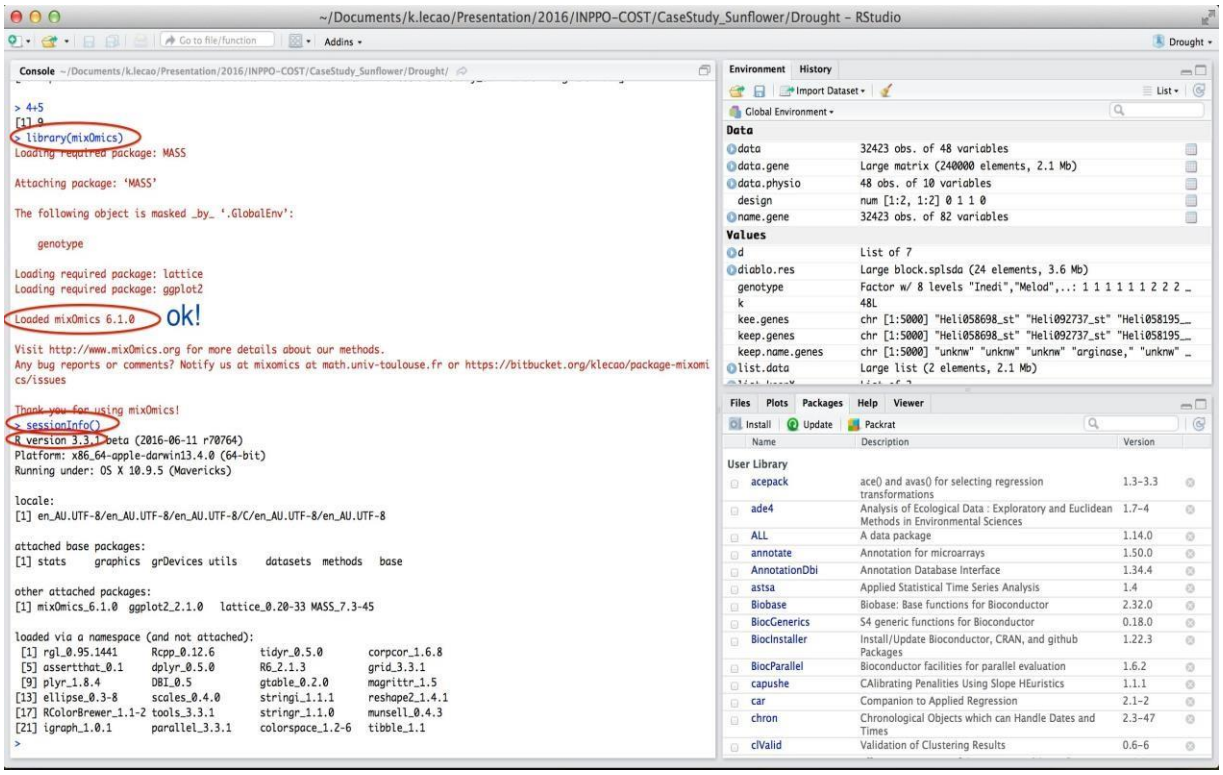


Figure 6. Check that the package mixOmics is installed and has the version 6.1.0.

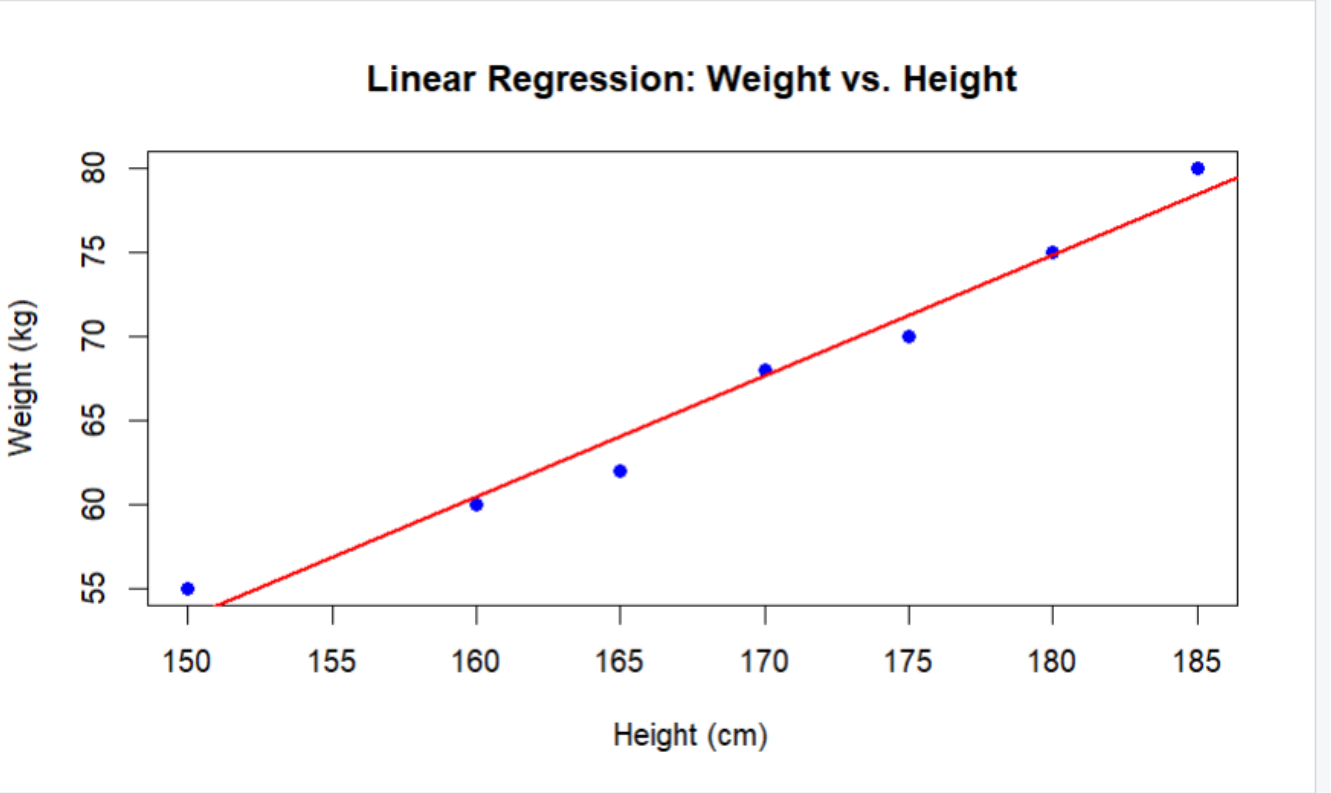
Exp:7

210701142

Implement Linear and Logistic Regressiona

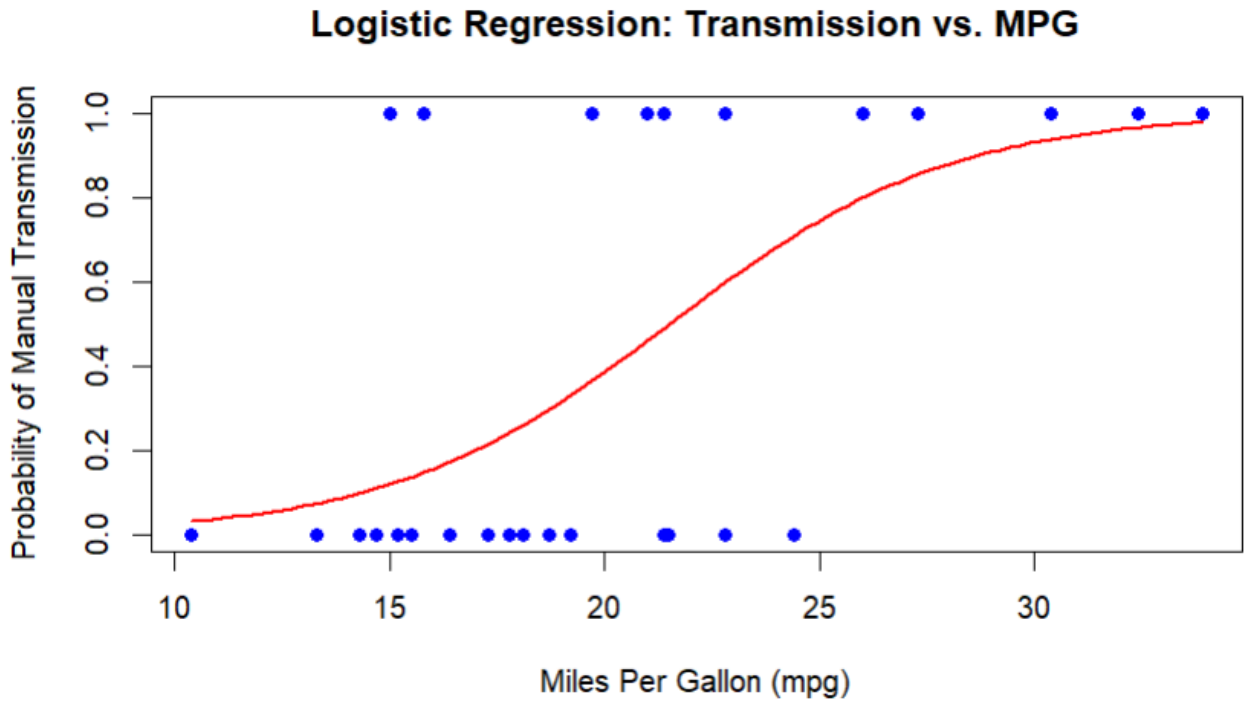
a) Linear regression

```
# 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)
```



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)
```


Logistic Regression: Transmission vs. MPG
Probability of Manual Transmission
0.0
0.2
0.4
0.6
0.8
1.0
10
15
20
25
30
Miles Per Gallon (mpg)