Working With Data in R

In this exercise, you will use R commands to examine the data structures, query rows, columns, and subsets of a number of data sets using alternative R commands.

**1: Querying from data sets**

1. Retrieve data from the first row and second column in the mtcars data frame:

**mtcars[1, 2]**

3. Examine the structure of the iris data frame, using the str() function:

**str(iris)**

4. What values can the Species factor in iris take (i.e., what are its levels)?

**levels(iris$Species)**

5. Examine the attributes in iris for their particular data type. Use one of:

is.numeric()

is.character()

is.vector()

is.matrix()

is.data.frame()

1. Use row and column names instead of numeric coordinates to find out how many miles per gallon (mpg) a Merc280C 2480 gets (query the mtcars data set).

**mtcars["Merc 280C", "mpg"]**

7. Find out how many rows are in the mtcars data frame:

**nrow(mtcars)** **# number of data rows**

* 1. **32**

1. Find out how many columns are in the mtcars data frame:

**ncol(mtcars)** **# number of columns**

* 1. **11**

1. Preview the first few rows of the mtcars data frame:

**head(mtcars)**

10. Use the c() function to select the mpg and gear attributes from mtcars:

myvars <- c("mpg", "gear")

newdata <- mtcars[myvars]

11. Select the first and fifth through tenth variables of mtcars:

**newdata <- mtcars[c(1,5:10)]**

1. Exclude the variables mpg, cyl, and disp from the mtcars data set using a logical vector:

**myvars <- names(mtcars) %in% c("mpg", "cyl", "disp")**

**newdata <- mtcars[!myvars]**

13. Exclude the third and fifth variables:

**newdata <- mtcars[c(-3,-5)]**

14. Delete the variables qsec and vs from a copy of mtcars:

**mtcarsCopy<-mtcars**

**mtcarsCopy$qsec <- mtcarsCopy$vs <- NULL**

1. Retrieve the ninth column vector of mtcars using the double square bracket

([[]]) operator:

**mtcars[[9]]**

* 1. **1 1 1 0 0 0 0 0 0 0 0 ...**

1. Retrieve the same column vector by its name:

**mtcars[["am"]]**

* 1. **1 1 1 0 0 0 0 0 0 0 0 ...**

1. Use the $ operator instead of the double square bracket operator, to retrieve am:

**mtcars$am**

* 1. **1 1 1 0 0 0 0 0 0 0 0 ...**

1. Use a comma character with the [] operator to indicate that all rows are to be retrieved from the am column vector:

**mtcars[,"am"]**

* 1. **1 1 1 0 0 0 0 0 0 0 0 ...**

1. Retrieve the first five complete rows of the data set:

**newdata <- mtcars[1:5,]**

20. Load the weather data set from the rattle package:

**library(rattle)**

**data(weather,package="rattle")**

21. Retrieve rows where the Rainfall in Canberra is greater than 16:

**newdata <- weather[ which(weather $Location=='Canberra'**

* + **weather$Rainfall > 16), ]**

1. Use the attach() function to make objects within the data frame accessible with fewer keystrokes and rerun the previous query with shorter syntax:

**attach(weather)**

**newdata <- weather[ which(Location=='Canberra' & Rainfall > 16),]**

1. Select the Location, Date, and Rainfall columns for all rows where the

Rainfall is greater than or equal to 15:

**newdata <- subset(weather, Rainfall >= 15, select=c(Location, Date, Rainfall))**

1. Select all column values between MinTemp and Sunshine (inclusive) where

WindGustDir is NW and it is raining on the following day:

**newdata <- subset(weather, WindGustDir == "NW" & RainTomorrow == "Yes", select=MinTemp:Sunshine)**