**What is a List?**

A **list** is a great tool to store many kinds of object in the order expected. We can include matrices, vectors data frames or lists. We can imagine a list as a bag in which we want to put many different items. When we need to use an item, we open the bag and use it. A list is similar; we can store a collection of objects and use them when we need them.

We can use list() function to create a list.

list(element\_1, ...)

arguments:

-element\_1: store any type of R object

-...: pass as many objects as specifying. each object needs to be separated by a comma

In the example below, we create three different objects, a vector, a matrix and a data frame.

**Step 1)** Create a Vector

# Vector with numeric from 1 up to 5

vect <- 1:5

**Step 2)** Create a Matrices

# A 2x 5 matrix

mat <- matrix(1:9, ncol = 5)

dim(mat)

**Output:**

## [1] 2 5

**Step 3)** Create Data Frame

# select the 10th row of the built-in R data set EuStockMarkets

df <- EuStockMarkets[1:10,]

**Step 4)** Create a List

Now, we can put the three object into a list.

# Construct list with these vec, mat, and df:

my\_list <- list(vect, mat, df)

my\_list

**Output:**

## [[1]]

## [1] 1 2 3 4 5

## [[2]]

## [,1] [,2] [,3] [,4] [,5]

## [1,] 1 3 5 7 9

## [2,] 2 4 6 8 1

## [[3]]

## DAX SMI CAC FTSE

## [1,] 1628.75 1678.1 1772.8 2443.6

## [2,] 1613.63 1688.5 1750.5 2460.2

## [3,] 1606.51 1678.6 1718.0 2448.2

## [4,] 1621.04 1684.1 1708.1 2470.4

## [5,] 1618.16 1686.6 1723.1 2484.7

## [6,] 1610.61 1671.6 1714.3 2466.8

## [7,] 1630.75 1682.9 1734.5 2487.9

## [8,] 1640.17 1703.6 1757.4 2508.4

## [9,] 1635.47 1697.5 1754.0 2510.5

## [10,] 1645.89 1716.3 1754.3 2497.4

**Select elements from list**

After we built our list, we can access it quite easily. We need to use the [[index]] to select an element in a list. The value inside the double square bracket represents the position of the item in a list we want to extract. For instance, we pass 2 inside the parenthesis, R returns the second element listed.

Let's try to select the second items of the list named my\_list, we use my\_list[[2]]

# Print second element of the list

my\_list[[2]]

**Output:**

## [,1] [,2] [,3] [,4] [,5]

## [1,] 1 3 5 7 9

## [2,] 2 4 6 8 1

**Built-in data frame**

Before to create our own data frame, we can have a look at the R data set available online. The prison dataset is a 714x5 dimension. We can get a quick look at the bottom of the data frame with tail() function. By analogy, head() displays the top of the data frame. You can specify the number of rows shown with head (df, 5). We will learn more about the function read.csv() in future tutorial.

# Print the head of the data

PATH<-'https://raw.githubusercontent.com/vincentarelbundock/Rdatasets/master/csv/wooldridge/prison.csv'

df <- read.csv(PATH)[1:5]

head(df, 5)

**Output:**

## X state year govelec black

## 1 1 1 80 0 0.2560

## 2 2 1 81 0 0.2557

## 3 3 1 82 1 0.2554

## 4 4 1 83 0 0.2551

## 5 5 1 84 0 0.2548

We can check the structure of the data frame with str:

# Structure of the data

str(df)

**Output:**

## 'data.frame': 714 obs. of 5 variables:

## $ X : int 1 2 3 4 5 6 7 8 9 10 ...

## $ state : int 1 1 1 1 1 1 1 1 1 1 ...

## $ year : int 80 81 82 83 84 85 86 87 88 89 ...

## $ govelec: int 0 0 1 0 0 0 1 0 0 0 ...

## $ black : num 0.256 0.256 0.255 0.255 0.255 ...

All variables are stored in the **numerical** format.