Lab 1 - Basics in R

Systems modelling and data analysis
2016/2017

1 Data types, subsets

1. Create a directory on the disk (the first letter of the name + surname). It will be your Working Directory.

   ```
   "F:/isurname"
   ```

2. Run RStudio

3. Display the informations about setwd() function

   ```
   ?setwd
   ```

4. Set your Working Directory using the command setwd()

   ```
   setwd("F:/isurname")
   ```

5. Make sure that the Working Directory is set correctly (use the command getwd()).

6. Enter an expression that assigns a value of "36.6"temp.

   ```
   temp <- "36.6"
   ```

7. Show temp variable by typing its name.

8. Try to add 0.1 to created variable. Is it possible? Why?

9. Check the type of variable temp

   ```
   class(temp)
   ```

10. Convert the variable “temp” to the numeric type and then check its value.

    ```
    temp <- as.numeric(temp)
    temp
    ```

11. Convert the variable “temp” to the integer and then check its value.

    ```
    temp <- as.integer(temp)
    temp
    ```

12. Check the types of the following values:

    ```
    class("36")
    class(36)
    class(36L)
    class(36+0i)
    class(TRUE)
    ```

13. Create a vector using the following sequence, then print a value of this vector.

    ```
    v1 <- 3:0
    v1
    ```

14. Check how to display the selected elements of the vector by typing the following expressions:

    ```
    v1[1]
    v1[length(v1)]
    v1[c(T,T,T,F)]
    v1[3:1]
    v1[-2]
    ```

15. Create the following vectors using the function c(), and then check their type and values.

    ```
    v2 <- c(0/0, 1/0, 1/Inf, TRUE, as.numeric("abc"))
    v3 <- as.logical(c("T", "False", "abc"))
    ```

16. Pay particular attention to the NA and NaN values. What is.na() function return if it accept the argument NaN? What is.nan() function return if it accept the argument NA?

    ```
    is.na(NaN)
    is.nan(NA)
    ```

17. Create an empty vector using the function vector(). Assign to it values: 1,2,3,4, ńull", 5,6,7,8,9. What type of data contain this vector after assigning? Convert the vector to a vector of numeric values, and then delete the NA of the vector.

    ```
    v4 <- vector("numeric", length=9)
    v4[1:4] <- 1:4
    v4[5] <- "null"
    v4[6:9] <- 6:9
    class(v4)
    v4 <- as.numeric(v4)
    bad <- is.na(v4)
    v4 <- v4[!bad]
    ```

18. Create vector v5 and v6. Then check the results of the following expressions:

```
v5 <- 1:5
v6 <- 6:10
v5 + v6
v5 - v6
v5 * v6
v5 / v6
v5 == v6
v5 >= 3
```

19. Create a matrix using the functions cbind and rbind.

```
m1 <- cbind(v5,v6)
m2 <- rbind(v5,v6)
```

20. Display m1 matrix, then display the item on the 1,2 position in the matrix. Is the first value determines the row or column? How the parameter drop = FALSE affects the outcome?

```
m1
m1[1,2]
m1[1,2, drop=FALSE]
```

21. Display rows 2 and 3 of the matrix.

```
m1[2:3,]
```

22. Create a matrix m3 with 2 rows and 5 columns. Complete m3 with values from 1 to 10. Note how the values have been entered into the matrix. Compare the result with the matrix m1.

```
m3 <- matrix(1:10, nrow=2, ncol=5)
```

23. Create a 10-element vector, and then use it to create a matrix m4 of 5 rows and 2 columns. Note how the values have been entered into the matrix. Check the attributes of the matrix m4. Compare the result with the matrix m2.

```
m4 <- 1:10
dim(m4) <- c(5,2)
attributes(m4)
```

24. Create two matrices of size 2x2, then multiply them to each other.

```
m5 <- matrix(rep(1,4),nrow=2,ncol=2)
m6 <- matrix(rep(2,4),nrow=2,ncol=2)
m5 * m6
m5 %*% m6
```

25. Create a list using the list function. Check the types of the variable and its individual elements. Notice that list may have elements of different types.

```
l1 <- list(id=1L, name="Kowalski", temp=36.6) class(l1)
class(l1[[1]])
class(l1[[2]])
```

26. Check the different ways to access the items in the list. Are all these expressions give you access to the data?

```
l1$name
l1["temp"]
arg <- "id"
l1$arg
l1[arg]
l1["arg"]
l1$i
l1[["i"]]
l1[["i", exact=FALSE]]
```

27. Create a dataframe, then check its attributes and the number of rows and columns.

```
df <- data.frame(id=1:5, temp=c(36.6, NA, 37.2, 37.1, 36.8))
attributes(df)
nrow(df)
ncol(df)
```

28. Delete all rows in which there is NA:

```
good <- complete.cases(df)
df <- df[good,]
```

29. Convert dataframe into a matrix:

```
m <- data.matrix(df)
```

30. Create a factor. Print the tabular summary and convert factor into the vector.

```
f <- factor(c("male","female","female","male"), level=c("male","female"))
table(f)
v <- unclass(f)
```

31. For selected vectors modify the name attribute.

```
v <- c("Kowalski", "Jan")
names(v) <- c("nazwisko", "imie")
l <- list(nazwisko="Kowalski", imie="Jan")
m <- matrix(nrow=2, ncol=2)
m[1,1] <- "Kowalski"
m[1,2] <- "Jan"
m[2,1] <- "Nowak"
m[2,2] <- "Adam"
dimnames(m) <- list(c("1", "2"), c("nazwisko", "imie"))
```