University of Southern Denmark
IMADA
Ricardo Campello
(adapted and extended from Arthur Zimek's Originals for DM870)

DM583: Data Mining

Exercise 1: Introduction to R

Exercise 1-1 Get Started with R

(a) Download and install R-Studio: `https://www.rstudio.com/products/rstudio/download/`. You can try the following exercise suggestions yourself (and explore more of R as much as you want).

(b) Start a new R script containing your solutions. Save the script for later reference.

(c) Some commands that can come handy when learning R are: `help()`, `class()`, `typeof()`, `mode()`. You can use these commands on variables, functions, objects, and datasets to obtain information on them.

(d) If you need any packages that are not yet installed, you can install them from the 'packages' panel/tab in RStudio, or using the command `install.packages()` in the console (you can try, for instance, `help("install.packages")` if you need help).

Exercise 1-2 Dimensionless Arrays (so-called Vectors) in R

(a) Create a vector of length 5 containing both positive and negative numbers, using the concatenate (`c()`) command.

(b) Find the `mean()`, `max()`, `min()` of the vector. Then compute the mean of the absolute values.

(c) Taking a subset of the vector can be done using the following notation:
`vector[1:2]` will take the first two elements of the vector (*R* starts indexing with 1).
Insert 42 into the third position of the vector you created earlier.

(d) Create a new vector and build the sum of the two vectors.

(e) Create a random vector of length 30 using the `rnorm()` function with `n = 30` argument.

- Calculate the mean — what do you observe?
- Take the last 5 elements of the vector using the indexing described above.

Exercise 1-3 Matrices (Dimensional Arrays) in R

(a) Create a 2×2 matrix A by row binding vectors using the `rbind()` command.

(b) Nullify matrix A by adding another matrix that you define.

(c) Double all the values in the original matrix A by multiplication with another matrix that you define. **Note**: In R, the operator * will produce element-wise multiplication, whereas %*% will produce matrix multiplication.

Exercise 1-4 Data Frames and Exploration of Datasets in R

(a) `R` comes with many historical datasets. One of them is the `CO2` dataset. Use the `help()` function to read about the dataset.

(b) The `CO2` dataset is stored as a Data Frame, which is a very popular R object (essentially, a *list of lists* potentially of different types) used to manipulate tabular data in R. You can confirm that this is indeed a Data Frame with the command `is.data.frame()`. Obtain information about the dataset variables and types using the `str()` command.

(c) Visually inspect the first few rows (data observations) of the dataset using the command `head()`.

(d) Obtain summary statistics about the dataset using the `summary()` command.

(e) The results from items (a) and (b) above suggest that variables `Plant` and `Type` (to be referred to as `CO2$Plant` and `CO2$Type` in your code in R) have been represented as categorical ordinal (*ordered factor*) and categorical nominal (*non-ordered factor*) variables, respectively. Confirm that this is indeed the case and check the possible values (*factor levels*) taken by these categorical variables using the commands `is.factor()`, `is.ordered()`, and `levels()`.