Initiation to R Software

Pierre Michel

Master AMSE 1st year, 2019

Problem Set I

Note: to get help on a function named `fun`, use `?fun` or `help(fun)`.

1) Create a directory

a) Create a working directory named **Rproject** containing three folders **code**, **data** and **doc**.
b) In R, create a script *problem_set1.R* and save it in **code**.

2) Generate numerical or character sequences

a) Create 6 objects (`x1`, `x2`, `x3`, `x4`, `x5`, `x6`) of type vector/factor, and of mode numeric/character/logical. Use the function `c()`, simple affectations `=`, and arithmetic/logical operations.
b) Get the list of objects available in your R session.
c) Remove the third and fifth objects.
d) Use `seq()`, `rep()`, `c()`, and `paste()`
e) Use `summary()` to get descriptive statistics of the oobjects you created before.
f) Create the following sequences (avoid the `##`):

```
## 1 2 3 1 2 3 1 2 3
```

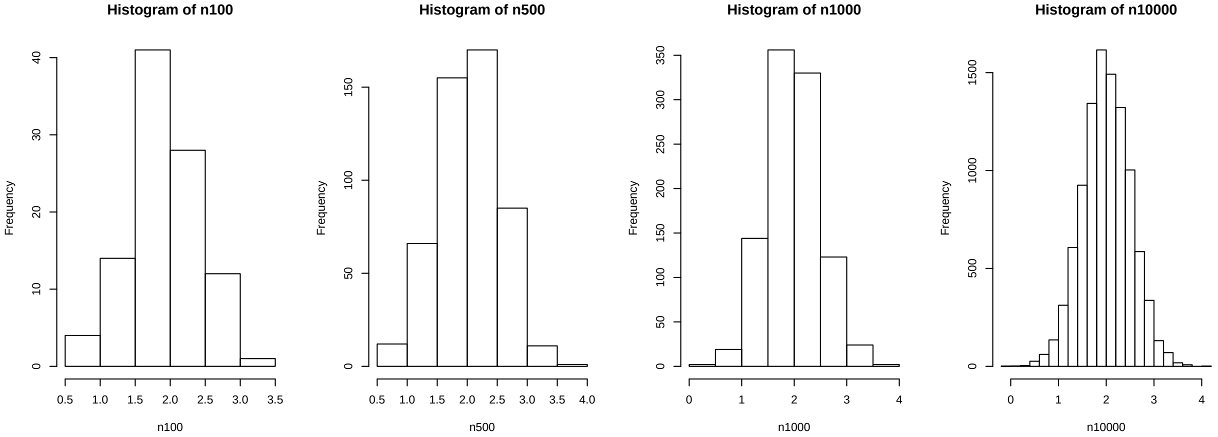
```
## 10
```

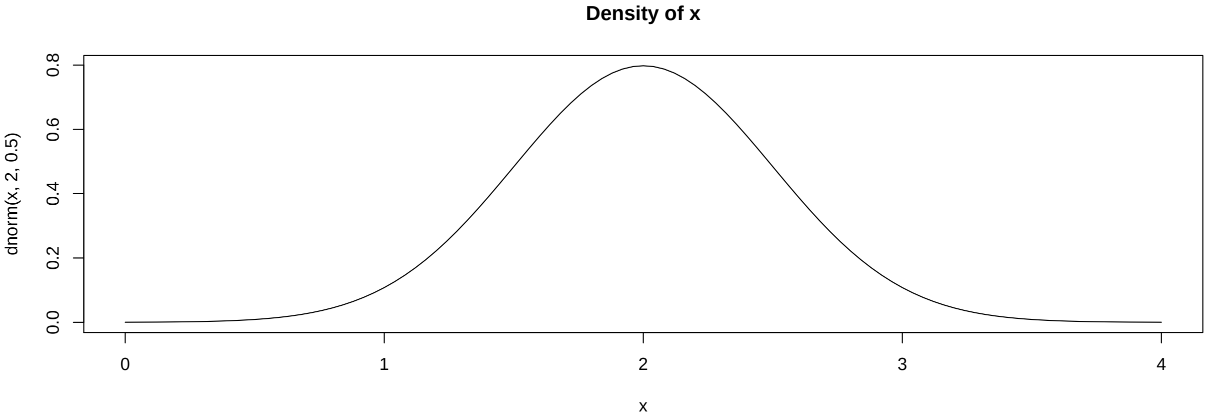
```
## 1 2 3 R 1 2 3 R
```

g) USe the function `scan()` to enter 10 random numbers between 1 and 100. Put the result in a vector `y`.

3) Pseudo-random number generation

a) Draw uniformly 10 random integers between 1 and 1000.
b) Draw 30 random integers between 1 and 5, with replacement.
c) Use the function `table()` on this vector. Explain the result.
d) Draw 100 observations from a normal distribution with parameters 2 (mean) and 0.5 (standard deviation). Use the function `hist()` to plot the histogram of counts of this vector. Repeat this operation for 500, 1000 and 10000 observations, plot all histograms in the same figure and add the corresponding density curve (use the functions `curve()` and `dnorm()`). Below is what you should get…





4) Discretization

a) Use the function `runif()` to create a vector of 1000 real numbers draw uniformly between 0 and 500.
b) Use the function `cut()` to discretize this vector in 4 levels according to the following intervals: $[0, 130]$, $]130, 200]$, $]200, 300]$ and $]300, 500]$.
c) Check that the output of `cut()` is an object of type `factor`. Use `factor()` to modify the printing of this factor.
d) Compute the number of values in each interval with `table()`.
e) Compute the cumulated sum with `cumsum()`.