

Exercices on sessions 1 and 2 of Going further with R

Load the package `tidyverse`.

Writing a function

We are going to write a function to standardise a variable to mean 0 and standard deviation 1. *There is already an existing function to do this (the function `scale`), so let's forget that this function exists.* Remember that to standardise a variable in this way, we need to do the following operation for each value x of the variable:

$$\frac{x - mean}{sd}$$

In addition to the variable to standardise, we want this function to take a second argument being the number of decimal we want the standardised values to be rounded to. We will give a default value of 2 to this argument.

- plan the function: identify the arguments and the steps
- create a test vector by randomly drawing 10 values in a normal distribution of mean 16 and standard deviation of 3
- write the code of the function (for keeping thing simple, this function will always remove the NA)
- create the function
- test the function

Conditional execution

- Write a function using conditional execution that take 2 arguments: the temperature (numerical) and the rain (boolean). This function will tell the user to:
 - go to the beach if the temperature is above 25°C and it's not raining
 - take their umbrella if the temperature is above 25°C and it's raining
 - take their jumper if the temperature is below 25°C and it's not raining
 - stay home otherwise
- Test this function
- Add two tests at the beginning of function to check that the arguments are of the correct mode, and stop the execution of the function otherwise

Loop

Write a **for** loop that calculate the cumulative sum of a vector. In other words: the first element of the output is the first element of the input, the second element of the output is the sum of the first two elements of the input, ... the n^{th} element of the output is the sum of the n^{th} first element of the input. *There is a function called `cumsum` that does this, but let's forget about it for now...* To run this loop, make a vector of 10 random draws between 0 and 4.

Advanced tidyverse

- Create the following list and use one of the **map** functions from the package **purrr** to get the maximum value of H for each plot.

```
$plot1  
[1] 8 25 12 16 14
```

```
$plot2  
[1] 31 25 18 19 29 24 16
```

```
$plot3  
[1] 15 12 13 17 19 11
```

- Use the function **across** with the function function you have created in the first section to standardise all the numerical variables of the data set iris
- Do the same thing but only for the variables related to sepals

For next time

- Make sure that git is properly installed on your computer and that git and Github are configured (see the setting up instructions [here](#))
- Install TinyTex by running the following R commands:

```
install.packages('tinytex')  
tinytex::install_tinytex()
```

and checking that it is installed:

```
tinytex::is_tinytex() # should be TRUE
```