

MODÈLES STATISTIQUES EN SANTÉ
- M2 SNS -
LET'S START

Before to (really) start : Creation of a github account To get started with GitHub, you'll need to create a free personal account and verify your email address.

1. Navigate to <https://github.com/>.
2. Click Sign up.
3. Alternatively, click on Continue with Google to sign up using social login.
4. Follow the prompts to create your personal account.

Exercise 1 - Github beginners

Step 0 : First we need to install Git. Visit <https://git-scm.com/downloads>, download the package for your system and install.

After you install Git, the 'git' command should be available to you in the terminal :

```
$ git
usage: git [--version] [--help] [-C <path>] [-c <name>=<value>]
        [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
        [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
        [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
        <command> [<args>]
...
```

Step 1 : Let's say you have started with a hobby mini project and wrote a simple script in a file greeter.py. You can use Spyder or vscode as a Python editor.

```
def main():
    print("Ehlo World!")
```

```
main()
```

Now you would like to save it in a Git repository.

Step 2 : Move to the folder where greeter.py is located and create an empty Git repository using a command git init.

```
$ git init
Initialized empty Git repository in /Users/vmonbet/BIOSTATS/TP/TP0
```

```
$ ls -a
.
..
.git
```

To check the status of your repo, issue

```
$ git status
On branch main
```

```
No commits yet
```

```
Untracked files:
```

```
(use "git add <file>..." to include in what will be committed)
greeter.py
```

Step 3 : Let's tell Git to track our greeter.py file and add it to the list for the next commit.

```
$ git add greeter.py
$ git status
On branch main
```

```
No commits yet
```

```
Changes to be committed:
```

```
(use "git rm --cached <file>..." to unstage)
new file:   greeter.py
```

Now the current version of the greeter.py is staged for commit and we can add it to our repository.

```
$ git commit -m "Initial version of greeter function."
```

```
$ git status
[main (root-commit) 86474e9] Initial version of greeter function.
```

Step 4 : In the greeter script we have a typo, so we change the text "Ehlo World!" to "Hello BioStatistics!". We must save the file in our text editor so the changes are written to disk, before we proceed !

```
$ git status
On branch main
```

```
Changes not staged for commit:
```

```
(use "git add <file>..." to update what will be committed)
(use "git restore <file>..." to discard changes in working directory)
modified:   greeter.py
```

```
$ git commit -a -m "Hello text update."
[main dca6083] Hello text update.
1 file changed, 1 insertion(+), 1 deletion(-)
```

```
$ git status
On branch master
nothing to commit, working tree clean
```

Step 5 : We would now like to provide a more personal script, so we change the greeter.py as follows :

```
def main():
    student = input("Please enter your name: ")

    print(f"Dear {student}, we are happy to have you in the BioStatistics track!")

main()
```

Again, add the file to stage and commit it to the repo.

```
$ git status
On branch master
Changes not staged for commit:
  (use "git add <file>..." to update what will be committed)
  (use "git restore <file>..." to discard changes in working directory)
    modified:   greeter.py

no changes added to commit (use "git add" and/or "git commit -a")

$ git commit -a -m "Personal update."
[master 29710cc] Personal update.
 1 file changed, 3 insertions(+), 1 deletion(-)
```

```
$ git status
On branch master
nothing to commit, working tree clean
```

Step 6 : Now our project has some history,

```
$ git log
commit 6f1e32e19aebab631a1728f984527c81fbea2d3c (HEAD -> main)
Author: Valerie Monbet <valerie.monbet@univ-rennes.fr>
Date:   Sun Aug 10 12:46:51 2025 +0200
```

Personal update.

```
commit dca608385f2d75958afa36a7b20630447a656ed2
Author: Valerie Monbet <valerie.monbet@univ-rennes.fr>
Date:   Sun Aug 10 12:42:41 2025 +0200
```

Hello text update.

```
commit 86474e907f4a5d4cb652bb1a5871c635cbbdbe02
Author: Valerie Monbet <vmonbet@RT7F232XKJ.local>
Date: Sun Aug 10 12:27:47 2025 +0200
```

Initial version of greeter function.

Exercise 2 - Start your first project

In this project, we will study liver steatosis (see <https://en.wikipedia.org/wiki/Steatosis>). For that, we use a mice model as described in [Martin et al., 2025].

1. What is *metabolic hepatic steatosis*?
2. How many mice are used? What treatments do they receive? How many groups are there?
3. Create a directory for the project. The directory will contain at least one paper, one dataset and some codes. You can create subdirectories to make your research more reproducible.
4. Download the dataset [SteatosisPhenotype.csv](#) and the companion paper.
5. Start writing a R (or python) code to read the data (with python use `pandas.read_csv`). One of the best IDE for R is Rstudio (for python vscode or spyder).

```
library(readr)
meta_data <- read_delim("Steatosis_phenotype.csv",
  delim = ";", escape_double = FALSE, trim_ws = TRUE)
View(meta_data)
str(meta_data)
```

6. Remove lines with missing data (with python use `pandas.dropna`)

```
lines_with_na = which(is.na(apply(meta_data[, -c(2,4), 1, sum))))
```

7. Find the number of samples, the number of treatment groups, the number of samples per treatment groups. Does it correspond to the informations available in the paper?
8. Compute descriptive statistics for the Cholesterol

```
library(dplyr)
meta_data %>%
  group_by(Group) %>%
  summarise(mean = mean(Cholesterol, na.rm = TRUE),
    std = sd(Cholesterol, na.rm = TRUE),
    n = n())
```

Plot boxplots to highlight the Cholesterol distribution in each group. You can use ggplot2 library.

Interpret the results.

9. Then, you can also compute the statistics for all the numeric variables as in Table in [\[Martin et al., 2025\]](#).

```
df.explore = meta_data[,-c(1,4)]
resume_stats <- df.explore %>%
  pivot_longer(cols = -Group,
    names_to = "Variable",
    values_to = "Valeur") %>%
  group_by(Group, Variable) %>%
  summarise(
    Mean = mean(Valeur, na.rm = TRUE),
    SD = sd(Valeur, na.rm = TRUE),
    .groups = "drop"
  )
```

10. For reproducible research and, for multiple users, is is convenient to work with Quarto or Markdown (or notebooks).
Move your code to a Quarto or R markdown file. It allows to combine text and code and generate html outputs.
11. Before to leave, let's tell Git to track the files (paper, dataset and code(s)). Don't forget to commit and push.

Références

[Martin et al., 2025] Martin, D., Monbet, V., Leroyer, P., Oliviero, N., Turlin, B., Salim, Z., Fautrel, A., Ropert, M., Sire, O., and Loréal, O. (2025). Sequential integration of multi-modal data from serum improves the predictive performance of hepatic lipid accumulation in mice.