

# Comment installer/configurer python sur ma machine?

18 septembre 2023

## Table des matières

<b>1</b>	<b>Installation de python</b>	<b>2</b>
1.1	J'utilise Linux (sous Ubuntu, Mint ou debian) . . . . .	2
1.1.1	Avec conda . . . . .	2
1.1.2	Avec pip . . . . .	3
1.2	J'utilise Windows . . . . .	3
1.3	J'utilise Mac . . . . .	4
1.4	Aucun des choix précédents ne fonctionne . . . . .	4
<b>2</b>	<b>Autres outils</b>	<b>4</b>
2.1	Utilisation de jupyter . . . . .	4
2.1.1	Sur Linux ou Mac . . . . .	4
2.1.2	Sur Windows . . . . .	5
2.2	Utilisation d'un éditeur . . . . .	5
2.3	Utilisation de conda . . . . .	5
2.3.1	Liste de tous ses environnements conda . . . . .	5
2.3.2	Liste des paquets d'un environnement conda donné . . . . .	5
2.3.3	Installer un paquet dans un environnement . . . . .	6
2.3.4	Désinstaller un paquet dans un environnement . . . . .	6
2.3.5	Supprimer un environnement conda . . . . .	6
2.3.6	Supprimer miniconda et Anaconda . . . . .	6

On utilisera exclusivement *python3* dans tous les TPs car *python2* est abandonné depuis début 2020.

Dans ce qui suit, ne lisez que les parties qui vous concernent (sauf si vous êtes intéressé).

# 1 Installation de python

## 1.1 J'utilise Linux (sous Ubuntu, Mint ou debian)

Si vous utilisez une distribution Linux différente des trois considérées ici, vous devez adapter les commandes à exécuter mais la démarche à suivre reste très semblable.

### 1.1.1 Avec conda

[La liste de commandes suivantes est à exécuter dans un terminal.]  
*conda* permet la création d'environnements de travail dans lesquels vous pouvez installer certains modules python selon le besoin. On peut facilement désactiver un environnement et changer d'environnement. On peut avoir des modules python différents selon chaque environnement, on peut aussi choisir des versions spécifiques de modules dans un environnement.

*conda* contient *python* et permet l'installation de modules python.

- Si votre machine personnelle est sous Ubuntu, Mint ou debian, il faut d'abord télécharger *Anaconda*, en exécutant dans le terminal ce qui suit (si *wget* n'est pas installé, faire *sudo apt install wget*)

```
wget https://repo.anaconda.com/archive/Anaconda3-2023.07-2-Linux-x86_64.sh
```

- On procède ensuite à l'installation d'*Anaconda*, en exécutant

```
bash Anaconda3-2023.07-2-Linux-x86_64.sh
```

Accepter ensuite les conditions d'utilisation et les réglages par défaut.

*conda* sera installé dans le dossier *\$HOME/anaconda3* (c'est ce qu'on appelle le *conda path* (=chemin pour *conda*)).

- Exécuter ensuite la commande suivante pour activer *conda*, si votre shell est *bash* (=résultat de la commande *echo \$SHELL* dans le terminal). Si vous utilisez un shell différent de *bash*, remplacez *bash* par la valeur de votre shell dans la commande

```
eval "$($HOME/anaconda3/bin/conda shell.bash hook)"
```

- Pour que la commande *conda* soit utilisable de façon permanente, il faut exécuter

```
conda init
```

Fermez le terminal et ouvrez-en un nouveau ou déconnectez-vous et reconnectez-vous.

- On peut vérifier que l'installation de *conda* a bien fonctionné, en tapant

```
conda --version
```

(ce qui doit renvoyer 23.7.2 ou une version proche)

- Si *conda* est correctement initialisé, vous pouvez créer un environnement *conda* nommé *LOO*, par exemple, qui contient les modules python nécessaires au cours, en exécutant ce qui suit dans le terminal (tout est sur une ligne dans la commande suivante)

```
conda create --name LOO -c conda-forge numpy scipy
matplotlib ipython sympy jupyter
```

- Pour pouvoir utiliser l'environnement, il faut l'activer ; la commande suivante doit être exécutée à chaque fois que vous aurez besoin de l'environnement *conda* créé

```
conda activate LOO
```

- Si vous n'avez plus besoin de l'environnement, il faut le désactiver (c'est automatique si vous vous déconnectez ou si vous éteignez votre machine).

```
conda deactivate
```

### 1.1.2 Avec pip

[La liste de commandes suivantes est à exécuter dans un terminal.]  
Si votre machine personnelle est sous Ubuntu, Mint ou debian, python3 s'installe comme suit :

```
sudo apt install python3
```

pip est un bon moyen de gérer les modules python. Il faut installer alors

```
sudo apt install python3-pip
```

Pour l'installation de matplotlib, le paquet Linux suivant est nécessaire

```
sudo apt install python3-tk
```

Il faut ensuite installer les modules python habituels via pip

```
pip3 install --user numpy scipy matplotlib ipython
```

Pour installer jupyter, faire

```
pip3 install --user jupyter
```

A la différence de *conda*, *pip* ne requiert pas l'utilisation d'environnements.

## 1.2 J'utilise Windows

Une possibilité est d'installer *Miniconda* qui contient *conda*.

- Si votre machine est 32 bits, il faut télécharger  
<https://repo.anaconda.com/miniconda/Miniconda3-latest-Windows-x86.exe> (32 bits)
- Si votre machine est 64 bits, il faut télécharger

[https://repo.anaconda.com/miniconda/Miniconda3-latest-Windows-x86\\_64.exe](https://repo.anaconda.com/miniconda/Miniconda3-latest-Windows-x86_64.exe) (64 bits)

- Installer ensuite ce fichier téléchargé
- numpy, scipy, matplotlib et jupyter devraient être installés d'office. Si ce n'est pas le cas, faire

```
conda install numpy scipy matplotlib jupyter
```

- Pour utiliser python, il suffit de lancer jupyter.

### 1.3 J'utilise Mac

Si votre machine est récente (puce M1 ou M2), un moyen d'installer python est de télécharger et installer *Miniconda* qui contient *conda* :

<https://repo.anaconda.com/miniconda/Miniconda3-latest-MacOSX-arm64.pkg>

Si votre machine est 64 bits, un moyen d'installer python est de télécharger et installer *Miniconda* qui contient *conda* :

[https://repo.anaconda.com/miniconda/Miniconda3-latest-MacOSX-x86\\_64.pkg](https://repo.anaconda.com/miniconda/Miniconda3-latest-MacOSX-x86_64.pkg)

Vous pouvez accepter le processus d'installation par défaut, *conda* sera installé dans votre HOME, en général sous *opt/miniconda3* (c'est ce qu'on appelle le *conda path* (=chemin pour *conda*)). Vous devez exécuter ensuite ce qui suit dans un terminal, pour obtenir les modules python souhaités pour le cours,

```
conda install -c conda-forge numpy scipy matplotlib  
jupyter spyder
```

### 1.4 Aucun des choix précédents ne fonctionne

Vous devez le signaler aux enseignants.

## 2 Autres outils

### 2.1 Utilisation de jupyter

#### 2.1.1 Sur Linux ou Mac

Si *jupyter* n'apparaît pas dans le menu des applications, vous pouvez utiliser le terminal pour lancer jupyter.

- Si l'installation est standard, le chemin de jupyter doit être pris en compte par défaut. Exécuter alors dans le terminal :

```
jupyter notebook &
```

- Si le chemin de l'installation de python est spécifique et est noté dans ce qui suit <path\_python>, faire alors

```
<path-python>/bin/jupyter notebook &
```

- Vous pouvez ensuite écrire et exécuter votre code. N'oubliez pas de sauvegarder le notebook!

### 2.1.2 Sur Windows

Un raccourci pour jupyter a dû être installé et jupyter doit être accessible depuis le menu des applications.

Vous pouvez écrire et exécuter votre code dans jupyter. N'oubliez pas de sauvegarder le notebook!

Il vous est fortement conseillé de sauvegarder votre code au format python (extension .py)

## 2.2 Utilisation d'un éditeur

Vous pouvez utiliser l'éditeur de votre choix pour écrire du code, le tout est d'obtenir un résultat correct.

- Si vous n'avez pas de préférence, sous Linux, utilisez *gedit*, simple d'utilisation et installé partout.
- Vous pouvez aussi utiliser *spyder*, le débogueur pour python, en tant qu'éditeur.
- Vous pouvez utiliser *Visual Studio Code* disponible sur les différents OS. Voir le lien <https://code.visualstudio.com>
- Sous Linux ou Mac, *emacs* ou *vim* sont d'autres éditeurs couramment utilisés mais d'utilisation plus compliquée.

## 2.3 Utilisation de conda

### 2.3.1 Liste de tous ses environnements conda

```
conda env list
```

### 2.3.2 Liste des paquets d'un environnement conda donné

Si l'environnement a déjà été activé, faire

```
conda list
```

Pour obtenir la liste des paquets d'un environnement conda non activé, faire

```
conda list -n <env>
```

Pour obtenir la liste des paquets d'un environnement conda non activé (on considère par exemple l'environnement conda *LOO*) et l'exporter dans le fichier *LOO.txt*, faire

```
conda list -n L00 > L00.txt
```

### 2.3.3 Installer un paquet dans un environnement

*conda-forge* est le channel conda le plus couramment utilisé.

```
conda install -c conda-forge <paquet>
```

### 2.3.4 Désinstaller un paquet dans un environnement

```
conda uninstall <paquet>
```

### 2.3.5 Supprimer un environnement conda

```
conda remove -n <env> --all
```

### 2.3.6 Supprimer miniconda et Anaconda

Il faut supprimer le dossier correspondant au *conda path* (dans ce qui suit, remplacer <conda\_path> par sa bonne valeur).

Attention, avec ce qui suit, tout est supprimé, vous perdez également les environnements installés

```
rm -fr <conda_path>
```