

# LES VENVs PYTHON ET LA RECHERCHE



# Le cauchemar du chercheur

- Le dilemme classique du chercheur: tantôt bleeding edge, tantôt retrocomputing.
- Projets multiples = conflits de versions.
- Une mise à jour de paquet peut casser un autre projet.
- Partager son code avec un collègue, un doctorant ou pour une publication devient pénible.
- Résultat : perte de temps, stress avant une deadline, et environnement non fiable.

**Projet A**

```
numpy 2.0  
pandas 2.3  
scipy 1.16.1
```



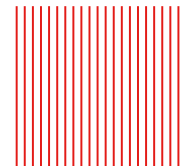
**Projet B**

```
numpy 2.4.3  
pandas 3.0.1  
jax 0.9.2
```

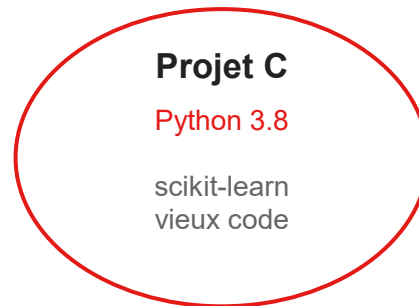
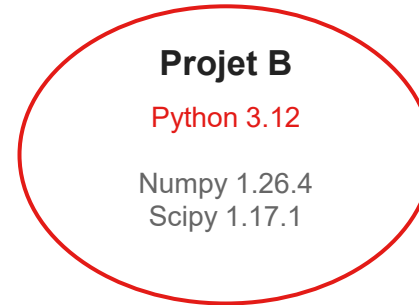
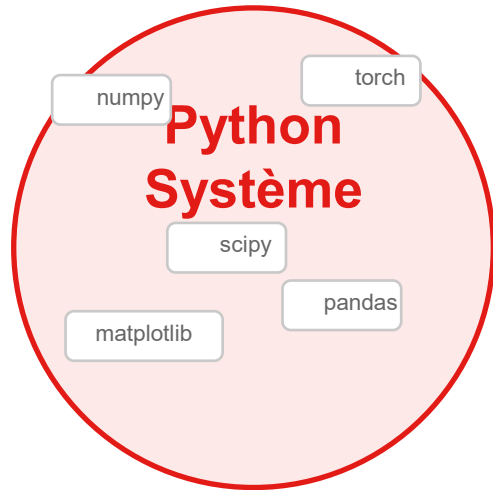
**Projet C**

```
torch  
CUDA ?  
driver ?
```

**Même machine, besoins incompatibles.**



# Les environnements virtuels sont les amis du chercheur



**Définition**

Un environnement virtuel est un répertoire contenant une installation autonome de Python, incluant des bibliothèques spécifiques au projet.

**Avantage**

Isolation totale : ce qui se passe dans un projet reste dans ce projet.





# Pourquoi le système se défend-il ? (PEP 668)

## Message de mise en garde sous Ubuntu

```
pip install numpy
error: externally-managed-environment

× This environment is externally managed
↳ To install Python packages system-wide, try apt install
python3-xyz, where xyz is the package you are trying to
install.

If you wish to install a non-Debian-packaged Python package,
create a virtual environment using python3 -m venv path/to/venv.
Then use path/to/venv/bin/python and path/to/venv/bin/pip. Make
sure you have python3-full installed.
```

- Depuis Ubuntu 22.04, Python système est protégé contre les installations system-wide.
- But : éviter de corrompre l'OS et les outils internes qui dépendent de Python.
- Le message d'erreur est une sécurité, pas un bug.
- La bonne réponse n'est pas de forcer : c'est de créer un environnement virtuel.

**Règle simple : ne jamais faire de  
sudo pip install.**



# L'outil standard : venv

---

- Inclus par défaut dans Python 3.
- Léger, natif et parfaitement adapté à la majorité des projets de recherche.
- À l'INSA, Python 3.10 et 3.12 sont installés par défaut sur les machines Linux d'enseignement et sur les postes Ubuntu 22.04 du personnel.

## Cas d'usage par défaut

C'est l'outil à utiliser dans environ 80 % des cas.

Démo 1 : création et activation

```
python3 -m venv mon_env
source mon_env/bin/activate
pip list
```

Variante Windows

```
# sous Windows
C:\Python312\python.exe -m venv mon_env
```





# L'alternative Anaconda à l'INSA

---

- Distribution orientée data science et calcul scientifique.
- À l'INSA : Anaconda 2024.02 est disponible via APT sous Linux et via le self-service WAPT sous Windows.
- Son intérêt principal ici : créer un environnement avec une autre version de Python que celle du système.

**Exemple:**  
Relancer un vieux code de 2020 qui exige Python 3.8.

```
Créer un environnement avec une version spécifique

conda create --name vieux_code python=3.8
conda init
conda activate vieux_code
```

Point clé : venv suit le Python système. Quand la version de Python elle-même doit changer, conda devient utile.



# La règle d'or INSA pour Anaconda

## OUI

Utiliser conda pour créer l'environnement et choisir la version de Python.

## NON

Éviter conda install pour les librairies : résolution lente et pénible.

## SOLUTION

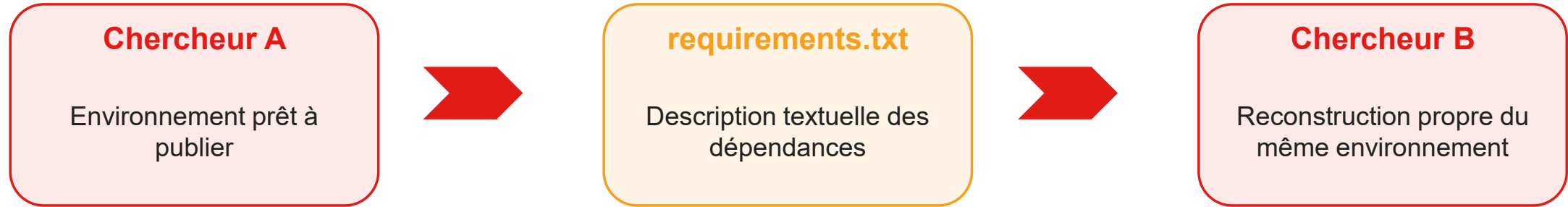
Une fois l'environnement créé, installer les paquets avec pip.

```
Démo 2 : conda pour l'enveloppe, pip pour les paquets
conda create --name mon_tp_en_py38 python=3.8
conda init
conda activate mon_tp_en_py38
pip install numpy pandas jupyter
```

```
Désactivation: conda deactivate (temporaire) ou conda init -reverse
Pour ne pas activer conda automatiquement dans le terminal: conda config --set auto_activate_base false
```



# Partager son travail



**!!! Ne jamais copier-coller le dossier de l'environnement ou le partager dans un repo git !!!**

```
Figer  
  
pip freeze > requirements.txt
```

```
Reconstruire  
  
pip install -r requirements.txt
```



# Conclusion & questions

---

## À retenir

Toujours isoler ses projets.

venv par défaut : léger et natif.

conda + pip si une version Python spécifique est requise.

requirements.txt pour partager et reproduire.

**Place aux questions**

