Introduction to Open & Reproducible Data Science (IORDS)

Michael Dayan, Data Scientist Manager

Methods & Data facility
Human Neuroscience Platform
Foundation Campus Biotech Geneva

Virtual machine info

To get an IP, please fill the form at:
`https://tinyurl.com/IORDS2021-IP-python3`

Connecting your:	WIFI SSID	WIFI Password
Laptop (no phones)	NIDS_course	reproduciblescience
Phone	CAMPUS_VISITORS	welcomecampus

START VS CODE AND JUPYTER ON YOUR BROWSER:

- *Start an internet browser on your own machine*
- *VS Code:* `<your_IP>:8080`
- *Jupyter:* `<your_IP>:8888`

PASSWORD: braincode!

On site support (including coding):

PLEASE CONNECT TO THE VM
→ Login: brainhacker
→ Password: brainhack!

Maël

Nathan

Connect to Slack and download the exercise slides

Remote support (including coding):

Serafeim

ANY PROBLEM? Please raise your hand or ask questions on Slack: channel **`#python`**

HNP METHODS & DATA

LECTURE OBJECTIVES

Python lectures objectives (you should be able to…):

- Know what is a Jupyter notebook useful for and know how to use it
- Know basic Markdown formatting (e.g. "*" for bullet lists, etc.)
- Understand basic Python types, and distinguish mutable and immutable entities
- Know how to implement main control flows in Python (for loop, if-else statements)
- Understand the concepts of branches and branch merging

Python Part 1

- Know how to define a Python function, with documentation and type hints
- Understand what a function returns, and what is the None type
- Describe why using a integrated development environment (IDE) is important
- Know how to import python modules and what is the role of `sys.path`
- Know what are exceptions and assertions, and how to implement them in Python

Python Part 2

LECTURE OBJECTIVES

Python lectures objectives (you should be able to…):

- Understand the main aspects of Object Oriented Programming (OOP)
- Distinguish between (data) attributes and methods (attributes)
- Understand what Conda is useful for and how to use it
- Know how to do basic numerical analysis with numpy
- Know how to plot data with matplotlib

Python Part 3

CONDA

Official description:

> Package, dependency and environment management for any language — **Python**, R, Ruby, Lua, Scala, Java, JavaScript, C/ C++, FORTRAN Conda:
>
> - runs on Windows, macOS and Linux.
> - installs, runs and updates packages and their dependencies
> - creates, saves, loads and switches between environments on your local computer

Environment definition

> A conda environment is a directory that contains a specific collection of conda packages that you have installed (and their dependencies). If you change one environment, your other environments are not affected.

Use cases:

- Use/develop a python package **which works** with a known set of python libraries (can simply export environment)
- Develop a python package and test it in different environments (i.e. different version of libraries)
- Create lightweight environments per project (only install required dependencies)
- Python package working only with a specific version of python (Python 2 and not Python 3)
- Etc.

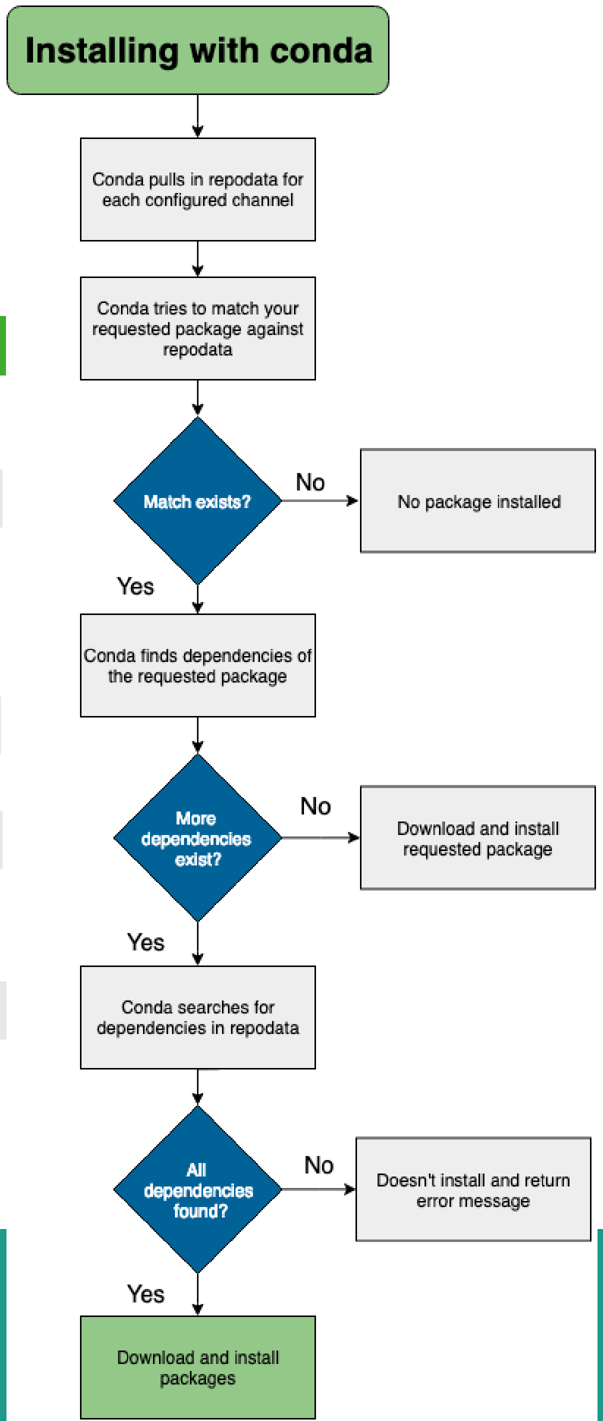
Distribution

- Anaconda is a full distribution of the central software in the PyData ecosystem, and includes Python itself along with binaries for several hundred third-party open-source projects.
- Miniconda is essentially an installer for an empty conda environment, containing only Conda and its dependencies, so that you can install what you need from scratch.

CONDA

Working with Environments	
Create a new environment named ENVNAME with specific version of Python and packages installed.	`conda create --name ENVNAME python=3.6 "PKG1>7.6" PKG2`
Activate a named Conda environment	`conda activate ENVNAME`
Activate a Conda environment at a particular location on disk	`conda activate /path/to/environment-dir`
Deactivate current environment	`conda deactivate`
List all packages and versions in the active environment	`conda list`
List all packages and versions in a named environment	`conda list --name ENVNAME`
List all revisions made within the active environment	`conda list --revisions`
List all revisions made in a specified environment	`conda list --name ENVNAME --revisions`
Restore an environment to a previous revision	`conda install --name ENVNAME --revision REV_NUMBER`
Delete an entire environment	`conda remove --name ENVNAME --all`

TIP: Anaconda Navigator is a desktop graphical user interface to manage packages and environments with Conda. With Navigator you do not need to use a terminal to run Conda commands, Jupyter Notebooks, JupyterLab, Spyder, and other tools. Navigator is installed with Anaconda, and may be added with Miniconda.

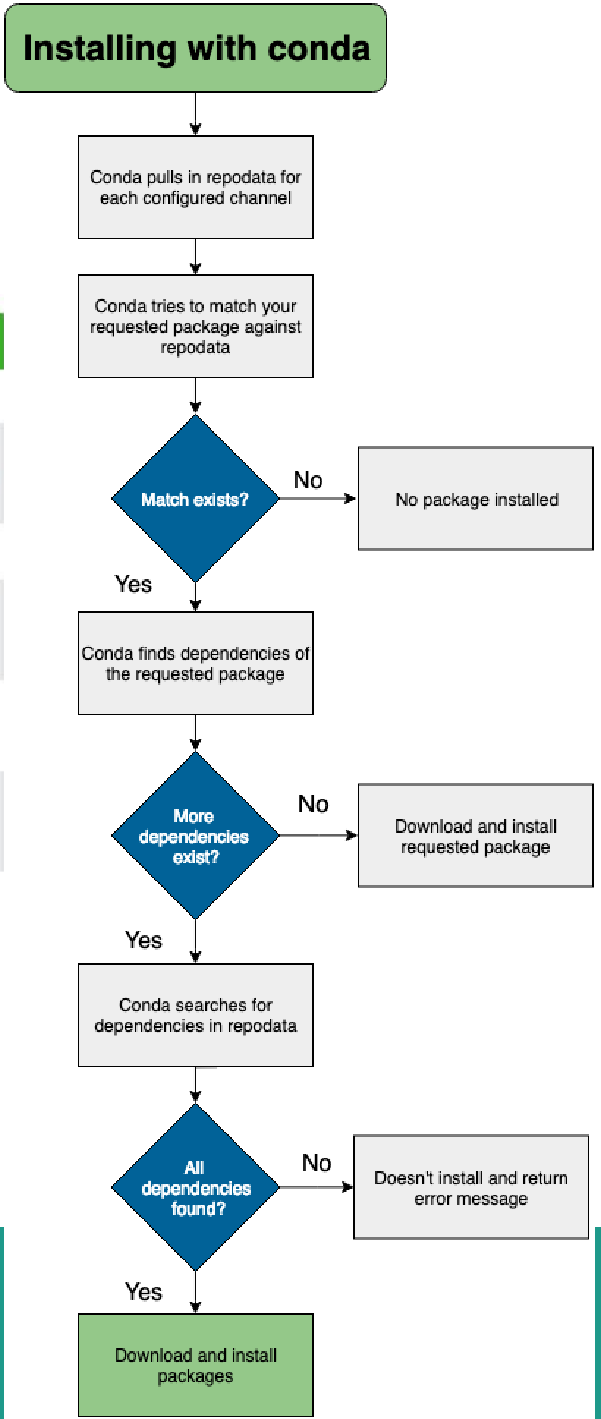

Installing with conda
Conda pulls in repodata for each configured channel
Conda tries to match your requested package against repodata
Match exists?
No
No package installed
Yes
Conda finds dependencies of the requested package
More dependencies exist?
No
Download and install requested package
Yes
Conda searches for dependencies in repodata
All dependencies found?
No
Doesn't install and return error message
Yes
Download and install packages

HNP METHODS & DATA

campus biotech Fondation Campus Biotech Geneva

CONDA

Sharing Environments	
Make an exact copy of an environment	`conda create --clone ENVNAME --name NEWENV`
Export an environment to a YAML file that can be read on Windows, macOS, and Linux	`conda env export --name ENVNAME > envname.yml`
Create an environment from YAML file	`conda env create --file envname.yml`
Create an environment from the file named environment.yml in the current directory	`conda env create`
Export an environment with exact package versions for one OS	`conda list --explicit > pkgs.txt`
Create an environment based on exact package versions	`conda create --name NEWENV --file pkgs.txt`

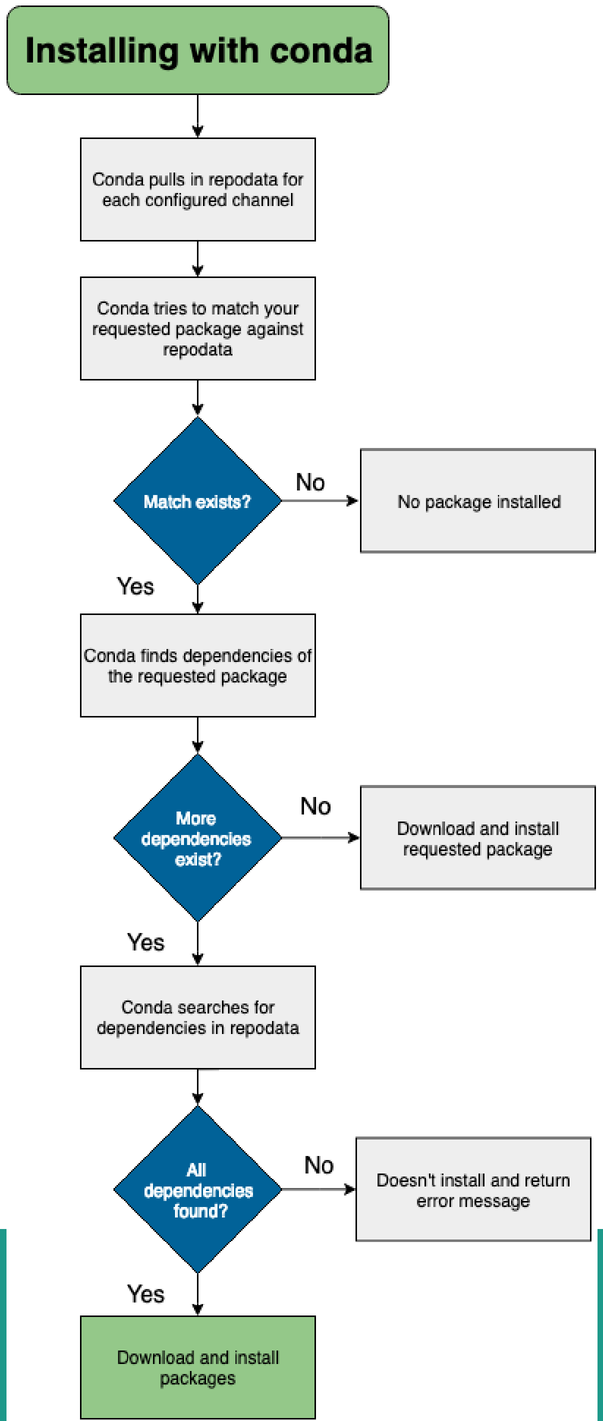


Installing with conda
Conda pulls in repodata for each configured channel
Conda tries to match your requested package against repodata
Match exists?
No
No package installed
Yes
Conda finds dependencies of the requested package
More dependencies exist?
No
Download and install requested package
Yes
Conda searches for dependencies in repodata
All dependencies found?
No
Doesn't install and return error message
Yes
Download and install packages

CONDA

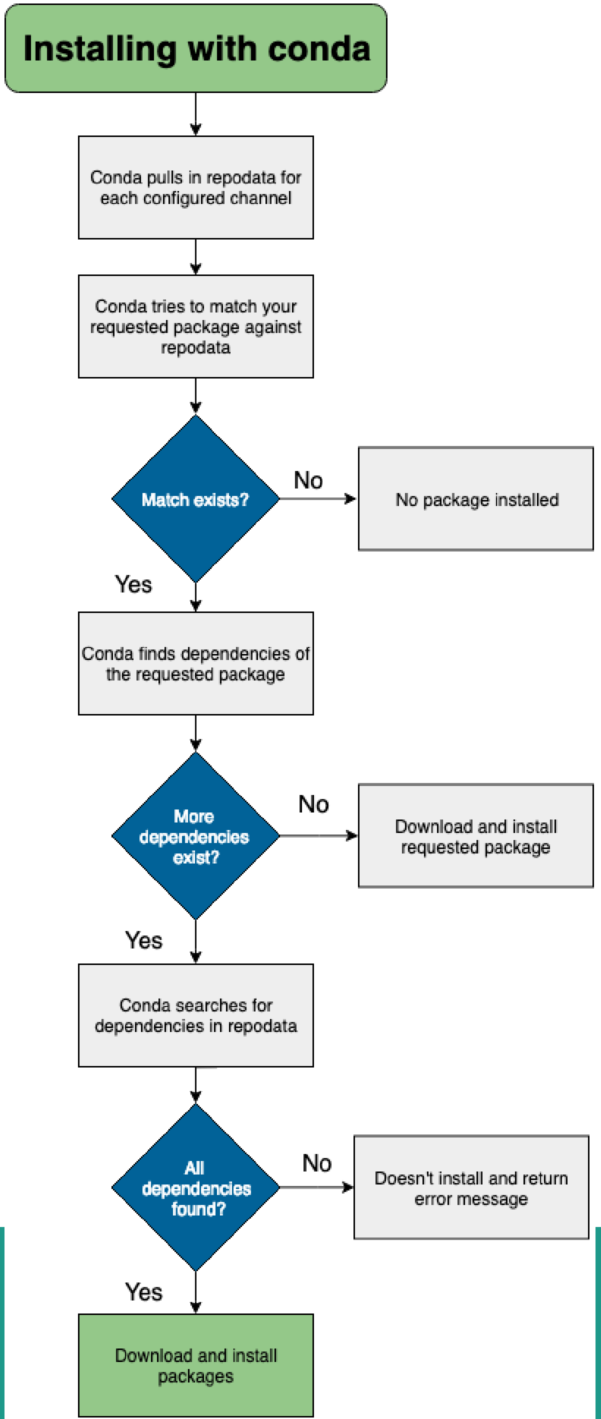
Using Packages and Channels

Search for a package in currently configured channels with version range >=3.1.0, <3.2"	`conda search PKGNAME=3.1 "PKGNAME [version='>=3.1.0,<3.2']"`
Find a package on all channels using the Anaconda Client	`anaconda search FUZZYNAME`
Install package from a specific channel	`conda install conda-forge::PKGNAME`
Install a package by exact version number (3.1.4)	`conda install PKGNAME==3.1.4`
Install one of the listed versions (OR)	`conda install "PKGNAME[version='3.1.2|3.1.4']"`
Install following several constraints (AND)	`conda install "PKGNAME>2.5,<3.2"`
Add a channel to your Conda configuration	`conda config --add channels CHANNELNAME`


Installing with conda
Conda pulls in repodata for each configured channel
Conda tries to match your requested package against repodata
Match exists?
No
No package installed
Yes
Conda finds dependencies of the requested package
More dependencies exist?
No
Download and install requested package
Yes
Conda searches for dependencies in repodata
All dependencies found?
No
Doesn't install and return error message
Yes
Download and install packages

CONDA

Additional Useful Hints	
Detailed information about package versions	`conda search PKGNAME --info`
Remove unused cached files including unused packages	`conda clean --all`
Remove a package from an environment	`conda uninstall PKGNAME --name ENVNAME`
Update all packages within an environment	`conda update --all --name ENVNAME`
Run most commands without requiring a user prompt. Useful for scripts.	`conda install --yes PKG1 PKG2`
Examine Conda configuration and configuration services	`conda config --show` `conda config --show-sources`


Installing with conda
Conda pulls in repodata for each configured channel
Conda tries to match your requested package against repodata
Match exists?
No
No package installed
Yes
Conda finds dependencies of the requested package
More dependencies exist?
No
Download and install requested package
Yes
Conda searches for dependencies in repodata
All dependencies found?
No
Doesn't install and return error message
Yes
Download and install packages

HOW TO SET YOUR DEV ENVIRONMENT AT HOME

- Install Visual Studio Code (instructions pinned in `#linux` channel)
  - For Windows, there is an extra step for WSL 2 (WSL 2 instructions pinned in `#linux` too)
- In a terminal, install conda
  - All the instructions will be posted and pinned in `#python` channel
- To use Jupyter with multiple conda environment
  - Install `jupyter` in the main `base` default environment
  - Install `ipykernel` in each conda environment you want to appear in Jupyter
- To use Jupyter
  - Activate conda base environment (`conda activate base`)
  - Start Jupyter from the directory (or parent directory) containing your notebooks

    ```
    jupyter notebook
    ```
  - Open a tab on your browser, go to the link displayed on the terminal (typically: `localhost:8888`) and create a notebook with the wanted kernel
- Use `git` to track your Python code, and import it in your Jupyter notebooks where necessary

COURSE SUPPORT

SLACK (iords2021.slack.com)

- Course main channel: `#general`
- Topic channels: `#linux, #linux-capstone, #git, #git-capstone, #python, #full-example, #machine-learning`

→ Check regularly for course info (esp. pinned items)

→ Do not hesitate to ask questions (please reply "in thread")

Channels are where teams share ideas and make decisions.

Zoe Maxwell 9:47 AM
How are those year-end numbers coming along?

Lee Hao 10:18 AM
Year-on-year Growth
78 kB PDF

Sara Parker 10:19 AM
Wow, so much great progress!

1-to-1 OFFICE HOURS for course questions:

- 20-min slots every Friday morning between 9AM and 11AM

→ Book a time slot here: https://tinyurl.com/IORDS-office-hours

→ Do not hesitate to ask any kind of question, this is a beginner course !

EMAIL: methods@fcbg.ch

Please whitelist!

Thank You!

Michael Dayan: methods@fcbg.ch