

COM6018 Data Science with Python

Getting Set Up

Varvara Papazoglou

Overview

- Tools
- Python Versions
- Python Release Cycle

Tools

There are a number of tools that we will use in this module. These are:

- `Python` – the programming language
- `Jupyter Notebooks` – a way of writing and running Python code
- `Git` – a version control system
- `GitHub` – a website for hosting Git repositories
- `Visual Studio Code` – a text editor (recommended)
- `uv` – an environment, package and project management tool for Python
- `bash` – a command line shell

These tools are all free and open source and all widely used in industry. They will also all run on `Windows`, `Mac` and `Linux`.

Python Versions

There are two versions of Python in common use: Python 2 and Python 3. We will be using Python 3 in this module. Python 2 is no longer supported and should not be used for new projects.

Python Release Cycle

Python 3 is being continually developed and new subversions are released regularly.

We will be using Python 3.12 .

[Python Release Cycle](#)

Getting the Materials

The materials for this module are hosted on GitHub.

Visit <https://github.com/UOS-COM6018-2627/COM6018>

You can `clone` the repository using `git`. Open a terminal and then type:

```
git clone https://github.com/UOS-COM6018-2627/COM6018.git
```

This will have made a directory called `COM6018` in your current directory. You can `cd` into this directory and then `ls` to see the contents.

```
cd COM6018  
ls
```

Contents

The materials are organised as follows:

```
├── materials
│   ├── labs      # This will contain the lab materials
│   ├── lectures  # Lecturer notes (slides)
│   ├── solutions # Solutions to the lab classes
│   └── tutorials # Tutorials covering the lecture content
```

Lectures and tutorials will generally be released on Friday.

Labs will be released on Monday.

Solutions will be released after the lab class.

Setting up a Python Virtual Environment

We will use `uv` to manage the Python environment. This ensures that the correct version of Python and all required packages are installed.

The project includes a file called `pyproject.toml`, which `uv` reads to determine the dependencies and their versions.

The environment.yml file

```
[project]
name = "com6018"
version = "2026.0.0"
description = "Materials for the 2026/2027 COM6018 module"
readme = "README.md"
requires-python = ">=3.12.5"
dependencies = [
    "cartopy>=0.25.0",
    "ipykernel>=6.30.1",
    "jupyterlab>=4.4.7",
    "matplotlib>=3.10.6",
    "notebook>=7.4.5",
    "numpy>=2.1.1",
    "pandas>=2.3.2",
    "scikit-learn>=1.7.2",
    "seaborn>=0.13.2",
    "xlrd>=2.0.2",
]
```

Creating the Python Virtual Environment

To set up the environment, run:

```
cd COM6018  
uv sync
```

If you haven't used `uv` before, you'll need to install it first. It works on Windows, Linux, and macOS.

[uv installation instructions](#)

Installing Visual Studio Code

I highly recommend using Visual Studio Code as your text editor. It is free and open source and has excellent support for Python.

<https://code.visualstudio.com/download>

It also provides support for running the `Jupyter Notebooks` that we use in the lab classes.

Installing Visual Studio Code (continued)

However, I strongly recommend that you disable AI features.

Settings > Chat > Miscellaneous

Jupyter Notebooks

Jupyter Notebooks are a way of writing and running Python code. They are very popular in data science and machine learning.

They allow you to integrate notes, Python code and the output of the code in a single document.

They are ideal for exploratory data analysis and for sharing your work with others. They are also great for lab classes.

```
cd COM6018 # replace COM6018 with the path to your COM6018 directory
source .venv/bin/activate
cd materials/tutorials
code 010_Introducing_Python.ipynb
```

How to open VS Code from command line

What Next?

Read the tutorials `010_Introducing_Python.ipynb` and `015_Further_Python.ipynb` to get started with Python.

Come to the Lab on Tuesday (tomorrow) and complete the `010_python_intro.ipynb` notebook.