Introduction

Jiaze Li

February 8, 2026

The University of Hong Kong

Overview

Python

IDE

Python

Python

- In this course, we will use **Python** for textual analysis.

- Why?
    - Python has rich libraries for textual analysis.
    - Python has clear syntax.
    - AI agents are good at Python.

Python

- In this course, we will use **Python** for textual analysis.
- Why?
  - Python has rich libraries for textual analysis.
  - Python has clear syntax.
  - AI agents are good at Python.

Python

- In this course, we will use **Python** for textual analysis.
- Why?
  - Python has rich libraries for textual analysis.
  - Python has clear syntax.
  - AI agents are good at Python.

Python

- In this course, we will use **Python** for textual analysis.

- Why?
    - Python has rich libraries for textual analysis.
    - Python has clear syntax.
    - AI agents are good at Python.

Python

- In this course, we will use **Python** for textual analysis.
- Why?
  - Python has rich libraries for textual analysis.
  - Python has clear syntax.
  - AI agents are good at Python.

Installation

- The best way to install Python is **NOT** to install Python.

- The standard way to install Python is through `conda`, an open-source package and environment manager.
  - I recommend `conda-forge`, a community-led channel. See `https://conda-forge.org/`.

- If you need to use advanced modules like `torch`, consider installing Python through `uv`, a fast package manager. See `https://docs.astral.sh/uv/`.

Installation

- The best way to install Python is **NOT** to install Python.
- The standard way to install Python is through `conda`, an open-source package and environment manager.
  - I recommend `conda-forge`, a community-led channel. See `https://conda-forge.org/`.
- If you need to use advanced modules like `torch`, consider installing Python through `uv`, a fast package manager. See `https://docs.astral.sh/uv/`.

Installation

- The best way to install Python is NOT to install Python.

- The standard way to install Python is through `conda`, an open-source package and environment manager.
  - I recommend `conda-forge`, a community-led channel. See `https://conda-forge.org/`.

- If you need to use advanced modules like `torch`, consider installing Python through `uv`, a fast package manager. See `https://docs.astral.sh/uv/`.

Installation

- The best way to install Python is **NOT** to install Python.

- The standard way to install Python is through `conda`, an open-source package and environment manager.
    - I recommend `conda-forge`, a community-led channel. See `https://conda-forge.org/`.

- If you need to use advanced modules like `torch`, consider installing Python through `uv`, a fast package manager. See `https://docs.astral.sh/uv/`.

Startup

- You can check if you have `conda-forge` installed correctly by running `conda --version` in your Miniforge Prompt (Windows) or Terminal (macOS/Linux).
  - For Windows users, if you want to use `conda` in Terminal, you should run `conda init` in Miniforge Prompt once.
- Your terminal should look like these:
  - `(base) C:\Users\Jiaze Li>`
  - `(base) jiaze@jiaze-legion:~$`

Startup

- You can check if you have `conda-forge` installed correctly by running `conda --version` in your Miniforge Prompt (Windows) or Terminal (macOS/Linux).
    - For Windows users, if you want to use `conda` in Terminal, you should run `conda init` in Miniforge Prompt once.

- Your terminal should look like these:
    - `(base) C:\Users\Jiaze Li>`
    - `(base) jiaze@jiaze-legion:~$`

Environments

- `conda` is a package and environment manager.
  - Packages are collections of code that provide specific functionality.
    - Examples: `pandas`, `matplotlib`, etc.
  - Environments are isolated spaces where you can install packages without affecting other environments.
- To create a new environment, run `conda create -n <env_name> python=<version>` in your Terminal.
  - For example, `conda create -n py312 python=3.12` will create a new environment named `py312` with Python 3.12 installed.
- For many commands (not only `conda`), you may see `Proceed ([y]/n)?`. If you know what you are doing, type `y` and press `Enter`.

Environments

- `conda` is a package and environment manager.
  - Packages are collections of code that provide specific functionality.
    - Examples: `pandas`, `matplotlib`, etc.
  - Environments are isolated spaces where you can install packages without affecting other environments.

- To create a new environment, run `conda create -n <env_name> python=<version>` in your Terminal.
  - For example, `conda create -n py312 python=3.12` will create a new environment named `py312` with Python 3.12 installed.

- For many commands (not only `conda`), you may see `Proceed ([y]/n)?`. If you know what you are doing, type `y` and press `Enter`.

Environments

- `conda` is a package and environment manager.
  - Packages are collections of code that provide specific functionality.
    - Examples: `pandas`, `matplotlib`, etc.
  - Environments are isolated spaces where you can install packages without affecting other environments.

- To create a new environment, run `conda create -n <env_name> python=<version>` in your Terminal.
  - For example, `conda create -n py312 python=3.12` will create a new environment named `py312` with Python 3.12 installed.

- For many commands (not only `conda`), you may see `Proceed ([y]/n)?`. If you know what you are doing, type `y` and press `Enter`.

Environments

- `conda` is a package and environment manager.
    - Packages are collections of code that provide specific functionality.
        - Examples: `pandas`, `matplotlib`, etc.
    - Environments are isolated spaces where you can install packages without affecting other environments.

- To create a new environment, run `conda create -n <env_name> python=<version>` in your Terminal.
    - For example, `conda create -n py312 python=3.12` will create a new environment named `py312` with Python 3.12 installed.

- For many commands (not only `conda`), you may see `Proceed ([y]/n)?`. If you know what you are doing, type `y` and press `Enter`.

Environments

- `conda` is a package and environment manager.
    - Packages are collections of code that provide specific functionality.
        - Examples: `pandas`, `matplotlib`, etc.
    - Environments are isolated spaces where you can install packages without affecting other environments.

- To create a new environment, run `conda create -n <env_name> python=<version>` in your Terminal.
    - For example, `conda create -n py312 python=3.12` will create a new environment named `py312` with Python 3.12 installed.

- For many commands (not only `conda`), you may see `Proceed ([y]/n)?`. If you know what you are doing, type `y` and press `Enter`.

Environments (Cont.)

- To list all your environments, run `conda env list`.
    - `base` is the default environment that comes with `conda`. You should avoid installing packages in the `base` environment to prevent conflicts.

- To activate the environment, run `conda activate <env_name>`.
    - Every time you run commands in the terminal, you should make sure you are in the correct environment.
    - Once you activate the environment, your terminal should look like these:
        - `(py312) C:\Users\Jiaze Li>`
        - `(py312) jiaze@jiaze-legion:~$`

- For more commands, run `conda --help` or check the official documentation at `https://docs.conda.io/`.

Environments (Cont.)

- To list all your environments, run `conda env list`.
    - `base` is the default environment that comes with `conda`. You should avoid installing packages in the `base` environment to prevent conflicts.

- To activate the environment, run `conda activate <env_name>`.
    - Every time you run commands in the terminal, you should make sure you are in the correct environment.
    - Once you activate the environment, your terminal should look like these:
        - `(py312) C:\Users\Jiaze Li>`
        - `(py312) jiaze@jiaze-legion:~$`

- For more commands, run `conda --help` or check the official documentation at `https://docs.conda.io/`.

Environments (Cont.)

- To list all your environments, run `conda env list`.
    - `base` is the default environment that comes with `conda`. You should avoid installing packages in the `base` environment to prevent conflicts.

- To activate the environment, run `conda activate <env_name>`.
    - Every time you run commands in the terminal, you should make sure you are in the correct environment.
    - Once you activate the environment, your terminal should look like these:
        - `(py312) C:\Users\Jiaze Li>`
        - `(py312) jiaze@jiaze-legion:~$`

- For more commands, run `conda --help` or check the official documentation at `https://docs.conda.io/`.

Packages

- To install packages in the activated environment, run `conda install <package_name_1> <package_name_2> ...`.
  - For example, `conda install jupyterlab pandas` will install the `jupyterlab` and `pandas` packages in the currently activated environment.
- Not all packages are available on `conda-forge`. You should check the official documentation of the package for installation instructions.
  - For example, `torch` no longer supports installation through `conda` since version 2.6.0.
- Recommended packages for general use:
  - `jupyterlab`: Add support for .ipynb files, which are interactive notebooks that allow you to run code and see the output in the same document.
  - `pandas`: A powerful library for data manipulation and analysis

Packages

- To install packages in the activated environment, run `conda install <package_name_1> <package_name_2> ...`.
    - For example, `conda install jupyterlab pandas` will install the `jupyterlab` and `pandas` packages in the currently activated environment.
- Not all packages are available on `conda-forge`. You should check the official documentation of the package for installation instructions.
    - For example, `torch` no longer supports installation through `conda` since version 2.6.0.
- Recommended packages for general use:
    - `jupyterlab`: Add support for .ipynb files, which are interactive notebooks that allow you to run code and see the output in the same document.
    - `pandas`: A powerful library for data manipulation and analysis

Packages

- To install packages in the activated environment, run `conda install <package_name_1> <package_name_2> ...`.
    - For example, `conda install jupyterlab pandas` will install the `jupyterlab` and `pandas` packages in the currently activated environment.
- Not all packages are available on `conda-forge`. You should check the official documentation of the package for installation instructions.
    - For example, `torch` no longer supports installation through `conda` since version 2.6.0.
- Recommended packages for general use:
    - `jupyterlab`: Add support for .ipynb files, which are interactive notebooks that allow you to run code and see the output in the same document.
    - `pandas`: A powerful library for data manipulation and analysis

IDE

IDE

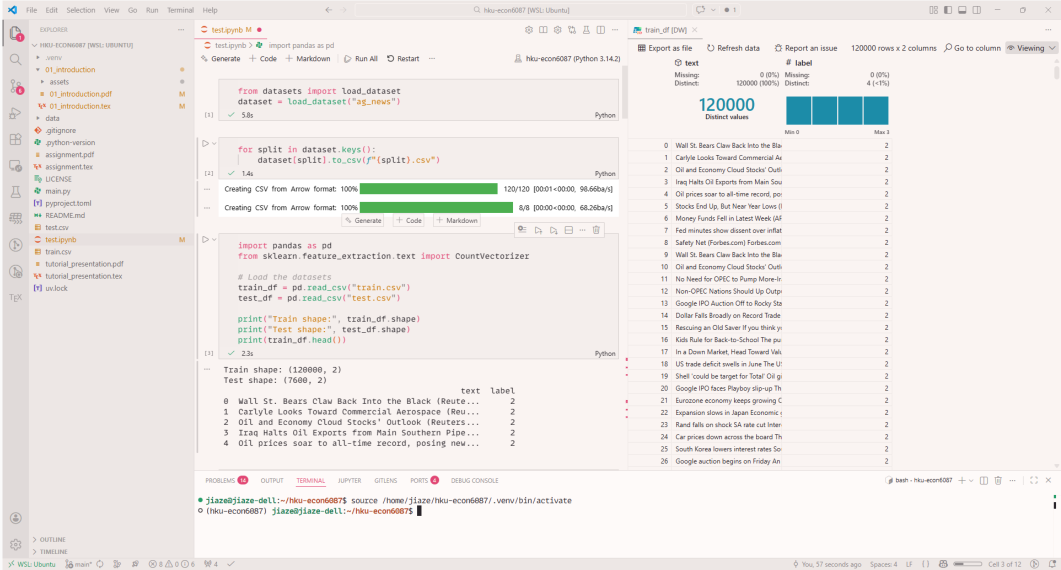
- In principle, you can write any code in any text editor like Notepad, but an Integrated Development Environment (IDE) can provide useful features like syntax highlighting, code completion, debugging, etc.
- I recommend using Visual Studio Code, an open-source code editor developed by Microsoft. See `https://code.visualstudio.com/`.
- You may also consider Cursor, a code editor based on VS Code that integrates AI features. See `https://www.cursor.com/`.
    - Caveat: Although Cursor is forked from VS Code, some extensions like Data Wrangler may not work properly in Cursor.

IDE

- In principle, you can write any code in any text editor like Notepad, but an Integrated Development Environment (IDE) can provide useful features like syntax highlighting, code completion, debugging, etc.
- I recommend using Visual Studio Code, an open-source code editor developed by Microsoft. See `https://code.visualstudio.com/`.
- You may also consider Cursor, a code editor based on VS Code that integrates AI features. See `https://www.cursor.com/`.
    - Caveat: Although Cursor is forked from VS Code, some extensions like Data Wrangler may not work properly in Cursor.

IDE

- In principle, you can write any code in any text editor like Notepad, but an Integrated Development Environment (IDE) can provide useful features like syntax highlighting, code completion, debugging, etc.
- I recommend using Visual Studio Code, an open-source code editor developed by Microsoft. See `https://code.visualstudio.com/`.
- You may also consider Cursor, a code editor based on VS Code that integrates AI features. See `https://www.cursor.com/`.
    - Caveat: Although Cursor is forked from VS Code, some extensions like Data Wrangler may not work properly in Cursor.



```python
from datasets import load_dataset
dataset = load_dataset("ag_news")
```

[1] ✓ 5.8s Python

```python
for split in dataset.keys():
    dataset[split].to_csv(f"{split}.csv")
```

[2] ✓ 1.4s Python

```
Creating CSV from Arrow format: 100% 120/120 [00:01<00:00, 98.66ba/s]
Creating CSV from Arrow format: 100% 8/8 [00:00<00:00, 68.26ba/s]
```

```python
import pandas as pd
from sklearn.feature_extraction.text import CountVectorizer

# Load the datasets
train_df = pd.read_csv("train.csv")
test_df = pd.read_csv("test.csv")

print("Train shape:", train_df.shape)
print("Test shape:", test_df.shape)
print(train_df.head())
```

[3] ✓ 2.3s Python

```
Train shape: (120000, 2)
Test shape: (7600, 2)
                                                text  label
0  Wall St. Bears Claw Back Into the Black (Reute...      2
1  Carlyle Looks Toward Commercial Aerospace (Reu...      2
2  Oil and Economy Cloud Stocks' Outlook (Reuters...      2
3  Iraq Halts Oil Exports from Main Southern Pipe...      2
4  Oil prices soar to all-time record, posing new...      2
```

	text	label
	Missing: 0 (0%) Distinct: 120000 (100%) 120000 Distinct values	Missing: 0 (0%) Distinct: 4 (<1%)
0	Wall St. Bears Claw Back Into the Bla	2
1	Carlyle Looks Toward Commercial Ae	2
2	Oil and Economy Cloud Stocks' Outl	2
3	Iraq Halts Oil Exports from Main Sou	2
4	Oil prices soar to all-time record, po	2
5	Stocks End Up, But Near Year Lows (	2
6	Money Funds Fell in Latest Week (AF	2
7	Fed minutes show dissent over inflat	2
8	Safety Net (Forbes.com) Forbes.com	2
9	Wall St. Bears Claw Back Into the Bla	2
10	Oil and Economy Cloud Stocks' Outl	2
11	No Need for OPEC to Pump More-Ir	2
12	Non-OPEC Nations Should Up Outp	2
13	Google IPO Auction Off to Rocky Sta	2
14	Dollar Falls Broadly on Record Trade	2
15	Rescuing an Old Saver If you think y	2
16	Kids Rule for Back-to-School The pu	2
17	In a Down Market, Head Toward Valu	2
18	US trade deficit swells in June The U	2
19	Shell 'could be target for Total' Oil gi	2
20	Google IPO faces Playboy slip-up Th	2
21	Eurozone economy keeps growing C	2
22	Expansion slows in Japan Economic	2
23	Rand falls on shock SA rate cut Inter	2
24	Car prices down across the board Th	2
25	South Korea lowers interest rates So	2
26	Google auction begins on Friday An	2

```
jiaze@jiaze-dell:~/hku-econ6087$ source /home/jiaze/hku-econ6087/.venv/bin/activate
(hku-econ6087) jiaze@jiaze-dell:~/hku-econ6087$
```

Extensions

- Extensions for Python development:
  - Python: Provides rich support for Python development. See `https://marketplace.visualstudio.com/items?itemName=ms-python.python`.
  - Jupyter: Provides support for Jupyter notebooks. See `https://marketplace.visualstudio.com/items?itemName=ms-toolsai.jupyter`.
- Recommended extensions for Python development:
  - GitHub Copilot Chat: Provides AI-powered code suggestions and explanations. See `https://marketplace.visualstudio.com/items?itemName=GitHub.copilot-chat`.
  - Data Wrangler: Provides a visual interface for data manipulation and analysis. See `https://marketplace.visualstudio.com/items?itemName=ms-toolsai.datawrangler`.

Extensions

- Extensions for Python development:
  - Python: Provides rich support for Python development. See `https://marketplace.visualstudio.com/items?itemName=ms-python.python`.
  - Jupyter: Provides support for Jupyter notebooks. See `https://marketplace.visualstudio.com/items?itemName=ms-toolsai.jupyter`.
- Recommended extensions for Python development:
  - GitHub Copilot Chat: Provides AI-powered code suggestions and explanations. See `https://marketplace.visualstudio.com/items?itemName=GitHub.copilot-chat`.
  - Data Wrangler: Provides a visual interface for data manipulation and analysis. See `https://marketplace.visualstudio.com/items?itemName=ms-toolsai.datawrangler`.