

Getting started

Machine Learning for Behavioral Data (CS-421)

February 18, 2025

Quiz



SpeakUp

Quiz



<https://www.python.org/>



<https://github.com/>



<https://www.anaconda.com/products/individual>



<https://jupyter.org/>



<https://noto.epfl.ch/>



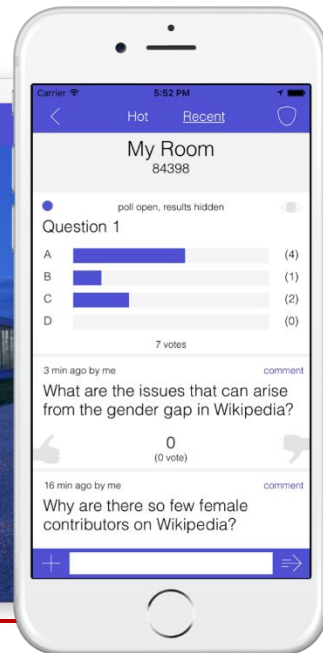
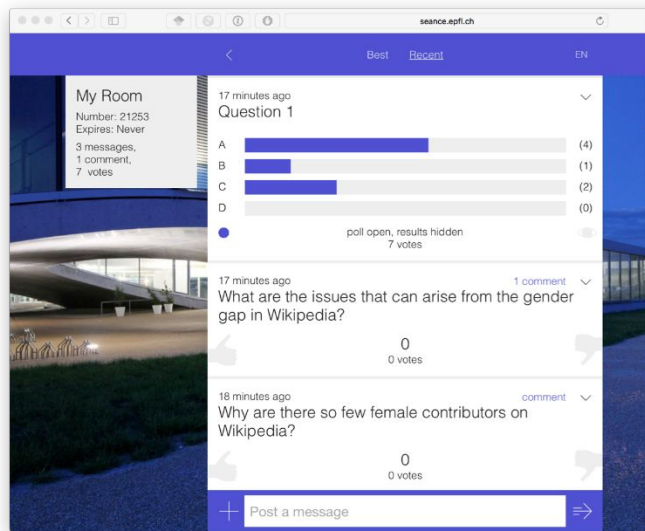
<https://colab.research.google.com/>

SpeakUp

<http://speakup.info/>

SpeakUp

<https://go.epfl.ch/speakup-mlbd>



Python



SpeakUp: How much do you know about Python?

A: It's a family of nonvenomous snakes with 10 genera and 42 species.

B: I have heard about the programming language Python.

C: I have used Python a few times (e.g. for courses).

D: I use Python on a regular basis.



Jupyter



SpeakUp: How much do you know about Jupyter?

A: It's the largest planet of our solar system.

B: I have heard about Jupyter notebooks.

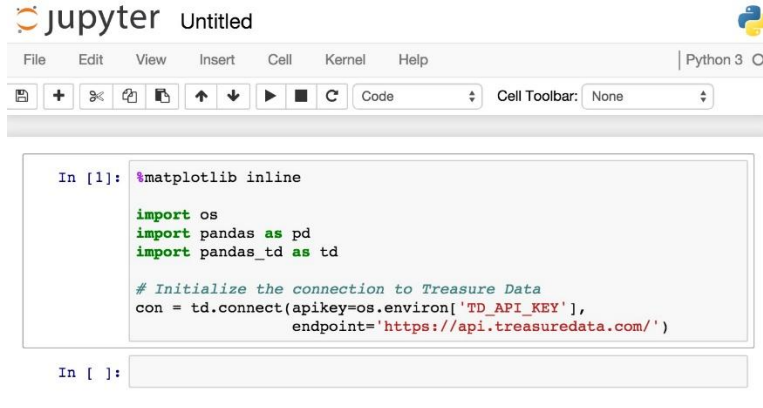
C: I have used Jupyter notebooks a few times (e.g. for courses).

D: I use Jupyter notebooks on a regular basis.



Jupyter

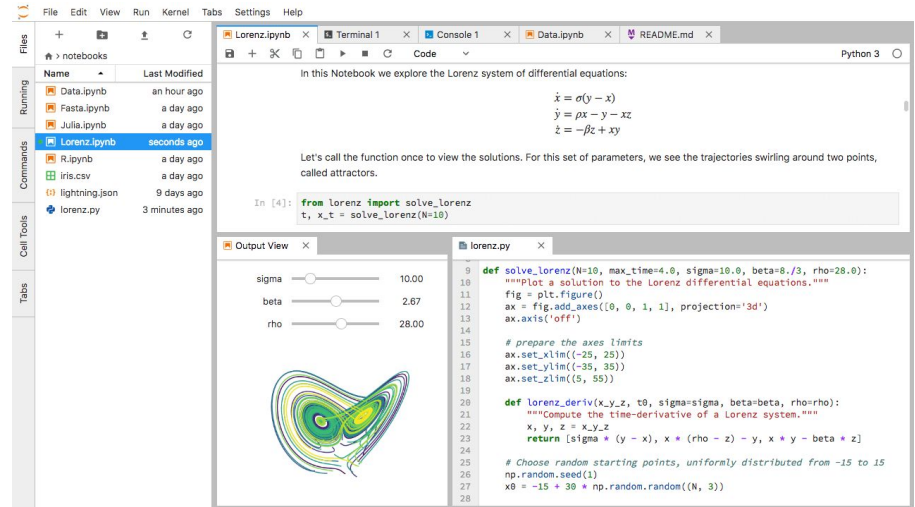
Jupyter notebook



```
In [1]: %matplotlib inline\nimport os\nimport pandas as pd\nimport pandas_tdata as td\n\n# Initialize the connection to Treasure Data\ncon = td.connect(apikey=os.environ['TD_API_KEY'],\n                endpoint='https://api.treasuredata.com/')
```

Tutorial: <https://www.dataquest.io/blog/jupyter-notebook-tutorial/>

JupyterLab



Why JupyterLab:

<https://towardsdatascience.com/jupyterlab-a-next-gen-python-data-science-ide-562d216b023d>

Anaconda (local env)

SpeakUp: How much do you know about Anaconda?



A: It's the heaviest and one of the longest known snake species.

B: I have heard about Anaconda.

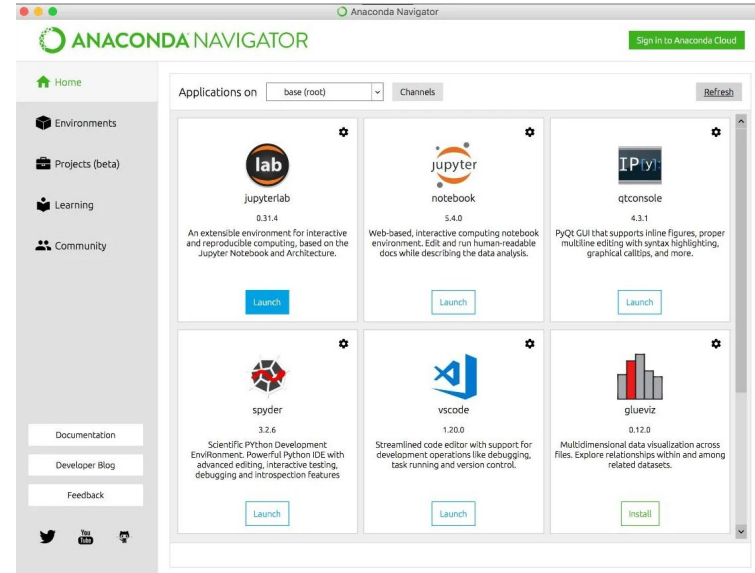
C: I have used Anaconda a few times.

D: I use Anaconda on a regular basis.



Anaconda (local env)

- You have the full control
- Works offline
- <https://www.anaconda.com/products/individual>



- **Tutorial:** <https://www.edureka.co/blog/python-anaconda-tutorial/>

Google Colab (online env)

SpeakUp: How much do you know about Colab?



A: It's an abbreviation for an artist group from New York.

B: I have heard about Colab.

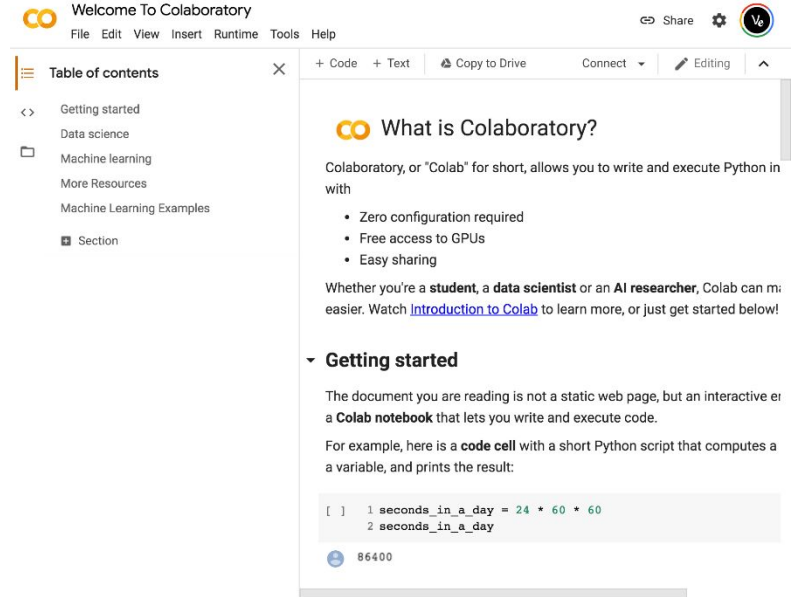
C: I have used Colab a few times.

D: I use Colab on a regular basis.



Google Colab (online env)

- Ready environment
- Uses Google's infrastructure
- Collaborative functionality
- Requires Google account
- <https://colab.research.google.com/>
- **Video:** <https://www.youtube.com/watch?v=inN8seMm7UI>



EPFL Noto (online env)

SpeakUp: How much do you know about Noto?

Noto

A: It's a city in Sicily declared a UNESCO world heritage in 2002.

B: I have heard about Noto.

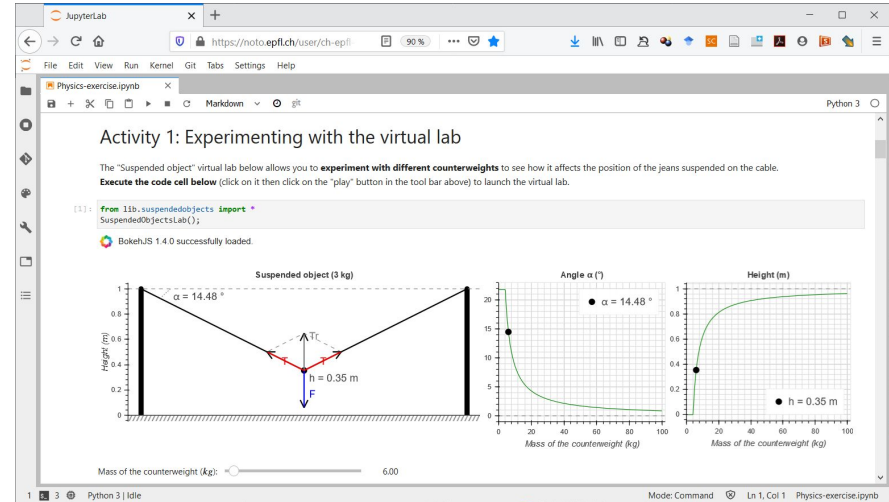
C: I have used Noto a few times.

D: I use Noto on a regular basis.



EPFL Noto (online env)

- Ready environment
- Login with your Gaspar
- <https://noto.epfl.ch/>



Noto

- Using Noto:
 - Go to <https://noto.epfl.ch/>
 - Login with your GASPAR
 - Go to Git → Clone
 - Clone the course repository: <https://github.com/epfl-ml4ed/mlbd-2025>

Git

SpeakUp: How much do you know about Git?

A: Git.....what?

B: I have heard about Git.

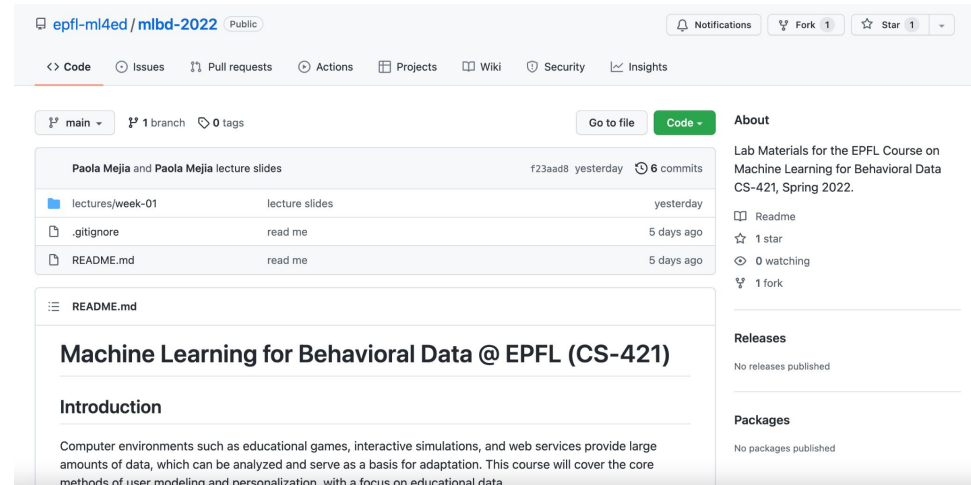
C: I have used Git a few times.

D: I use Git on a regular basis.



GitHub

- Share files and code
- Version control (git)
- **Tutorial:**
<https://www.edureka.co/blog/how-to-use-github/>



(Demo)

Setting up the environment

- Set up an environment on which you can
 - Run Jupyter notebooks in Python
 - Connect to course repository:
<https://github.com/epfl-ml4ed/mlbd-2025>
- We will use <https://noto.epfl.ch/>
 - But you are free to use whatever you want (e.g. Anaconda, Colab etc.)
 - It's your responsibility to have a working environment
- **Task:** Pull course's GitHub repository

Anaconda

- Virtual environment:
 - <https://janakiev.com/blog/jupyter-virtual-envs/>
 - Create virtual environment: `python -m venv myenv`
 - Activate virtual environment: `source myenv/bin/activate`
 - add to Jupyter: `python -m ipykernel install --user --name=myenv`

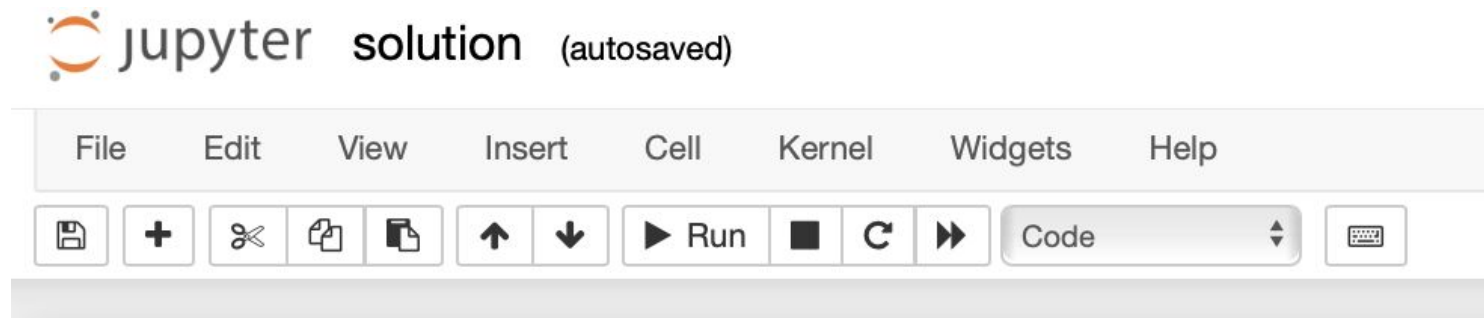


Notebook



Basic functions

[Colab](#) intro.



Git Intro

1. Basic git tutorial (add, commit, status).
 2. Github introduction.
 3. Branches (team work).
-

Git | Hello World

- New directory for Git repository
 - ◆ `mkdir gitdemo`
 - ◆ `cd gitdemo`
 - Now we're inside our new folder. Time to make it a proper Git repo:
 - ◆ `git init`
 - Now we're inside our new folder. Time to make it a proper Git repo:
 - ◆ `git init`
 - You'll see Initialized empty Git repository in `/path/to/your/repo/.git/`. What's that `.git`? If you list all files in your directory (`ls -a`), you'll see a new hidden `.git/` directory. That's where Git stores the information about this new repository.
- Time to add some files.
 - ◆ `touch new.txt`
 - ◆ `echo "Hello, World!" > new.txt`
 - You'll have a new file, `new.txt`
 - But this isn't just any old folder; it's Git repository! Git has tracked that we have a new file. Enter the following command:
 - ◆ `git status`

Why can't you see the file?

Git | Hello World

- `git add new.txt`
- `git status`
- Git knows about our file now. Time to commit our changes to Git's history.
 - ◆ `git commit -m "Add new.txt"`

The `-m` flag provides a commit message. Such a message is required for all commits.

- let's make some changes.
 - ◆ `echo "Foobar!" >> new.txt`
- This adds a new line (again, no text editor needed) to our `new.txt`.

How can you see the changes?

Git | Hello World

→ `git add new.txt`

→ `git status`

→ Git knows about our file now. Time to commit our changes to Git's history.

◆ `git commit -m "Add new.txt"`

→ `git status`

→ `git dif new.txt`

How can **add** the changes?

The `-m` flag provides a commit message. Such a message is required for all commits.

→ let's make some changes.

◆ `echo "Foobar!" >> new.txt`

→ This adds a new line (again, no text editor needed) to our `new.txt`.

How can you see the changes?

Git | Hello World

- `git add new.txt`
- `git status`
- Git knows about our file now. Time to commit our changes to Git's history.
 - ◆ `git commit -m "Add new.txt"`

The `-m` flag provides a commit message. Such a message is required for all commits.

- let's make some changes.
 - ◆ `echo "Foobar!" >> new.txt`
- This adds a new line (again, no text editor needed) to our `new.txt`.

How can you see the changes?

- `git status`
- `git dif new.txt`

How can **add** the changes?

- `git add new.txt`
- `git commit -m "adds changes"`

How can you **push** to github?

Github | Hello World

Create a new repository

A repository contains all project files, including the revision history. Already have a project repository elsewhere? [Import a repository](#).

Repository template
Start your repository with a template repository's contents.

No template ▾

Owner * **Repository name ***

paola-md ▾ /

Great repository names are short and memorable. Need inspiration? How about [special-engine](#)?

Description (optional)

☒ **Public**
Anyone on the Internet can see this repository. You choose who can commit.

☐ **Private**
You choose who can see and commit to this repository.

Initialize this repository with:
Skip this step if you're importing an existing repository.

☐ **Add a README file**
This is where you can write a long description for your project. [Learn more](#).

Add .gitignore
Choose which files not to track from a list of templates. [Learn more](#).

.gitignore template: None ▾

- `git branch -M main`
- `git remote add origin`
`https://github.com/paola-md/test.git`
- `git push -u origin main`

Github | Challenge

Try solving the tasks on your own and raise your hand if you need help.

Instructions:

1. Create a team of three and decide who is person A, B and C.
2. Person A: **Fork** the course's repo (<https://github.com/epfl-ml4ed/mlbd-2025>) and add B and C as collaborators.
3. B and C: **Clone** the forked repo.
4. A, B and C: **Create a branch** <person>-challenge-<number>. For example: a-challenge-1.
5. A, B and C: In your branch solve the corresponding task in <https://github.com/epfl-ml4ed/mlbd-2025/blob/main/lectures/week-01/git-challenge.py>

Github | Challenge

6. A, B and C: Create a **pull request** with your changes.
7. B: **Merge pull requests**.
8. C: **Pull** changes and run challenge.py locally.

Project

- Teams of 3 people
- We will provide data sets
- We will provide example research questions
- You will suggest an additional analysis/extension to the selected research question
- We will give feedback during the semester (see milestones)
- We will do project office hours (during lab sessions)
- You will do a presentation in the last week of the semester
- Final project (Code + Report)

Milestone M1

<https://go.epfl.ch/mlbd-m1-2025>

Fill out with team and start-up preference

Deadline: Tuesday, Feb 25th, 23:59

Feedback

We are actively looking for feedback to improve

<https://go.epfl.ch/mlbd-feedback>

Questions?