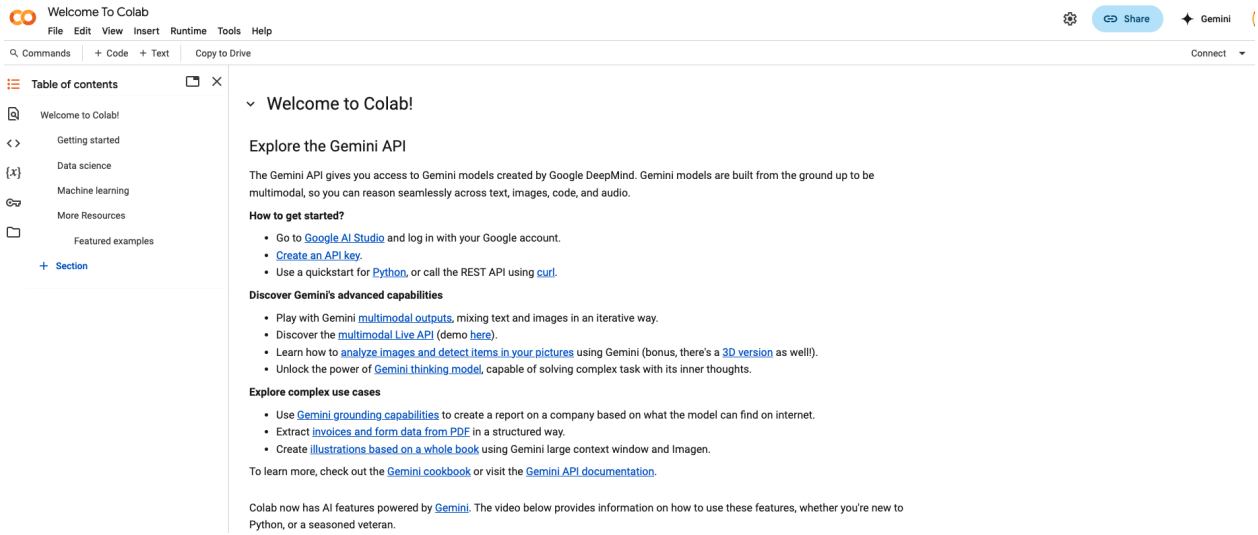
Getting started with Google colab

Google Colab is a coding environment which uses the Jupyter Notebook system integrated with Google drive. It is attached to the Google account and its associated drive. With Colab you can easily create a shared/ collaborative coding notebook. Colab has pre-installed some of the commonly used and population Python and R libraries. Just like Google Docs, you can share the Colab notebook via link and access rights.
Note: You are recommended to initiate the Google Colab environment using your UC Davis Kerberos/ Student ID.

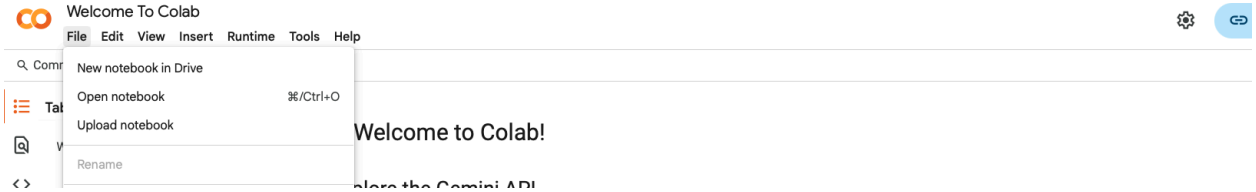
Step 1: Initialize

- Open your web browser of choice to https://colab.research.google.com
- If you are not already signed in, sign in with your UC Davis Kerberos/ Student ID
- By doing this, you should see this page

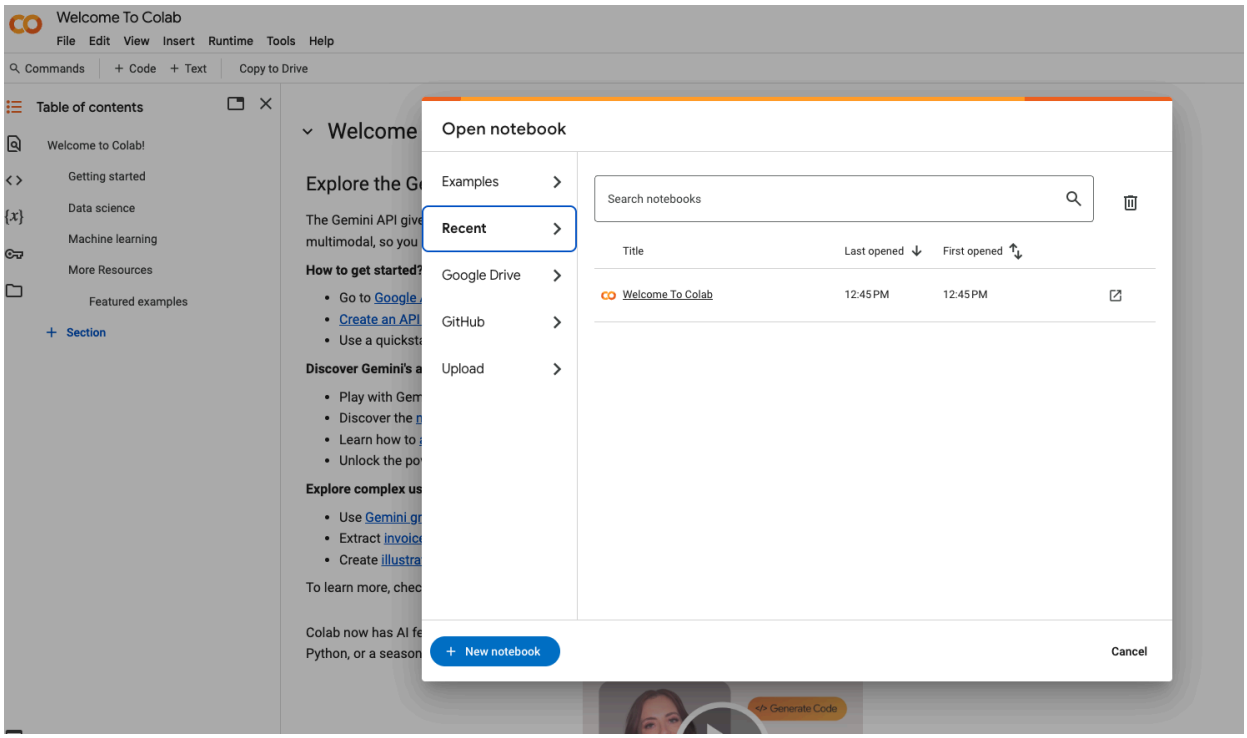


Step 2: Create a first notebook

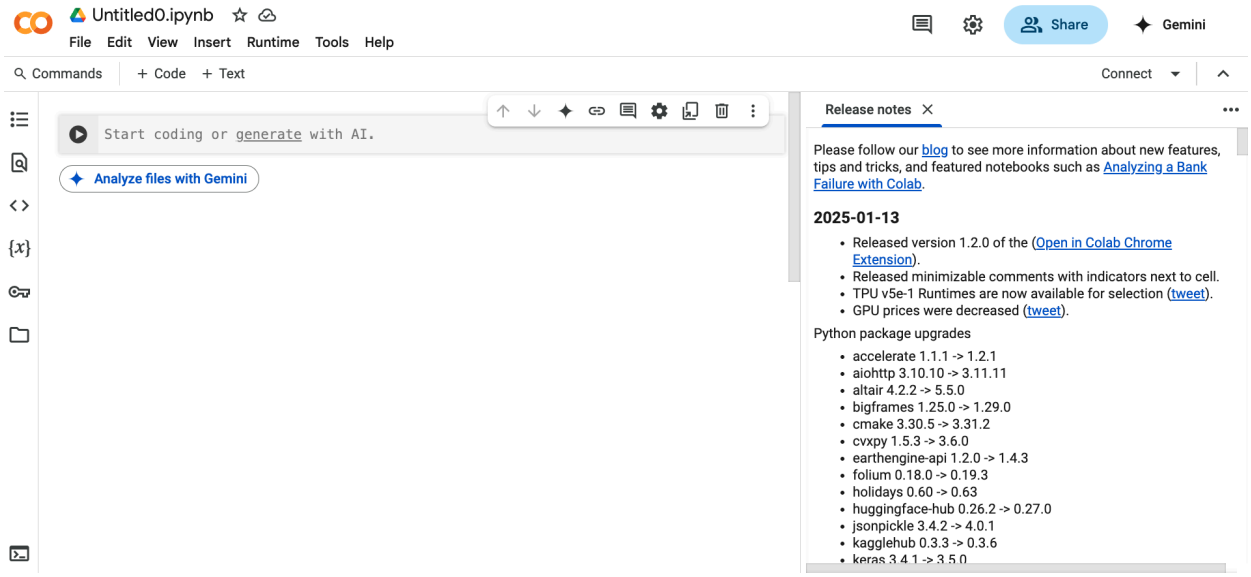
- Create first notebook from File > New notebook in Drive



- Alternatively, when you open Colab homepage, a pop up will appear from which you can create a new notebook by clicking on '+ New Notebook' on the bottom left corner of the pop-up.



- The new notebook will open in a new tab with the name ‘Untitled0.ipynb’. ‘Ipynb’ is the extension used by all Jupyter notebooks. On the right side, you will see the release notes and the latest version of the packages in this environment. Since jupyter notebook is primarily used for coding in Python language, most of the details will be related to releases of python packages.

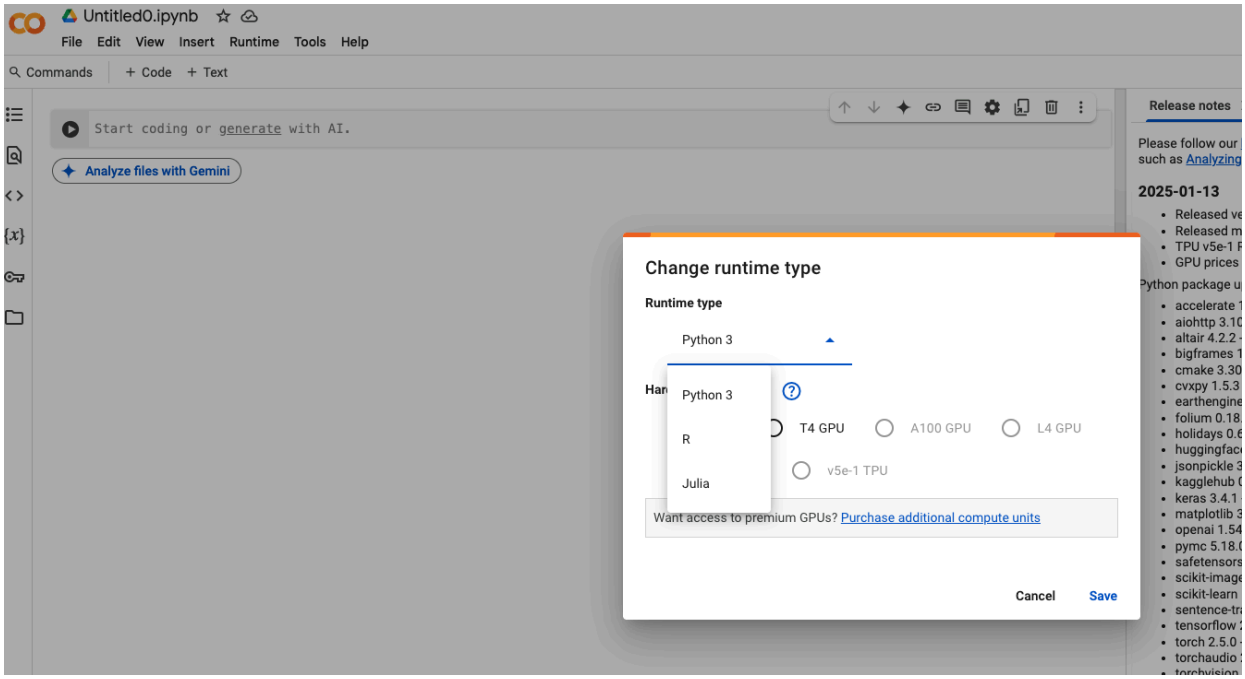


- All the features and commands of Jupyter notebook will work in Colab notebook. To know more, have a look at Jupyter Notebook and Getting Started with Jupyter for python.

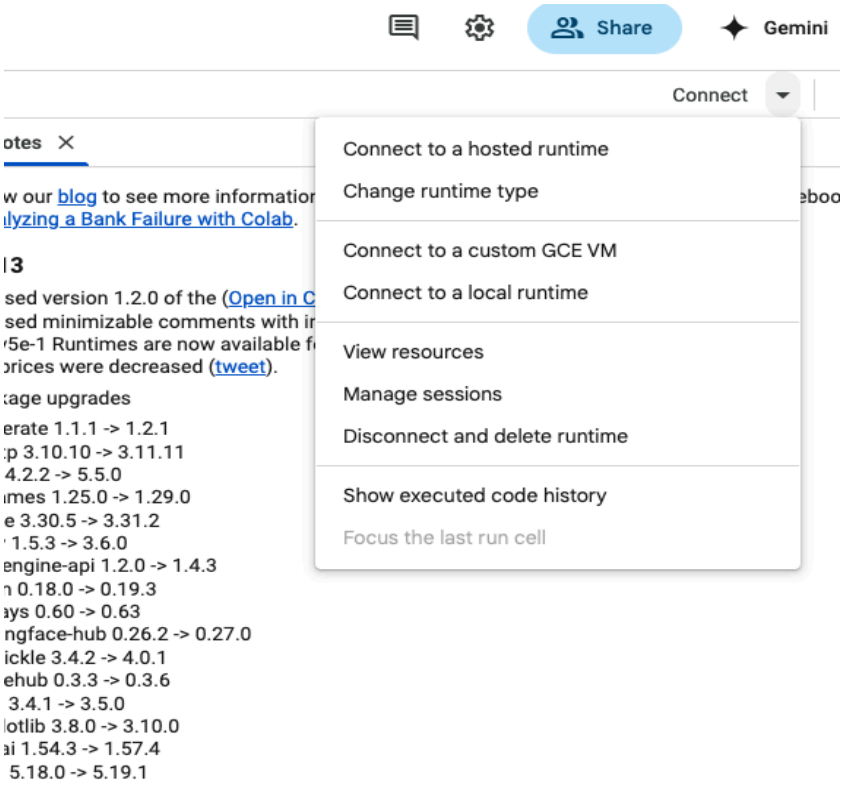
- This notebook is automatically saved in the folder /Colab Notebooks/ in your google drive. All notebooks created in this environment will be saved in this folder.
- It is recommended to change the name from ‘Untitled0.ipynb’ to something meaningful as soon as the notebook is created, since all notebooks will start out with names like ‘Untitled0’, ‘Untitled1’, etc. In our lab, we use the naming convention as follows:
    - Date in YYMMDD
    - Brief description of the notebook (1-2 keywords)
    - Primary author
    - So, for example a notebook about this tutorial will be called ‘250401_colab_tutorial_psk.ipynb’

Step 3: Change runtime environment to R

- To use R in Colab notebook, you must switch from Python core to R core. This is done in the following steps
- Go to menu bar and click on Runtime > Change Runtime Type > R



- Alternatively, you can also click on Connect > Change Runtime Type on the right top corner of your command bar



- **<u>Do not change the GPU or CPU settings since we won't be using these features.</u>** These are features that are used by AI/ML modelers to optimize their computational performance by increasing or decreasing the core capabilities.

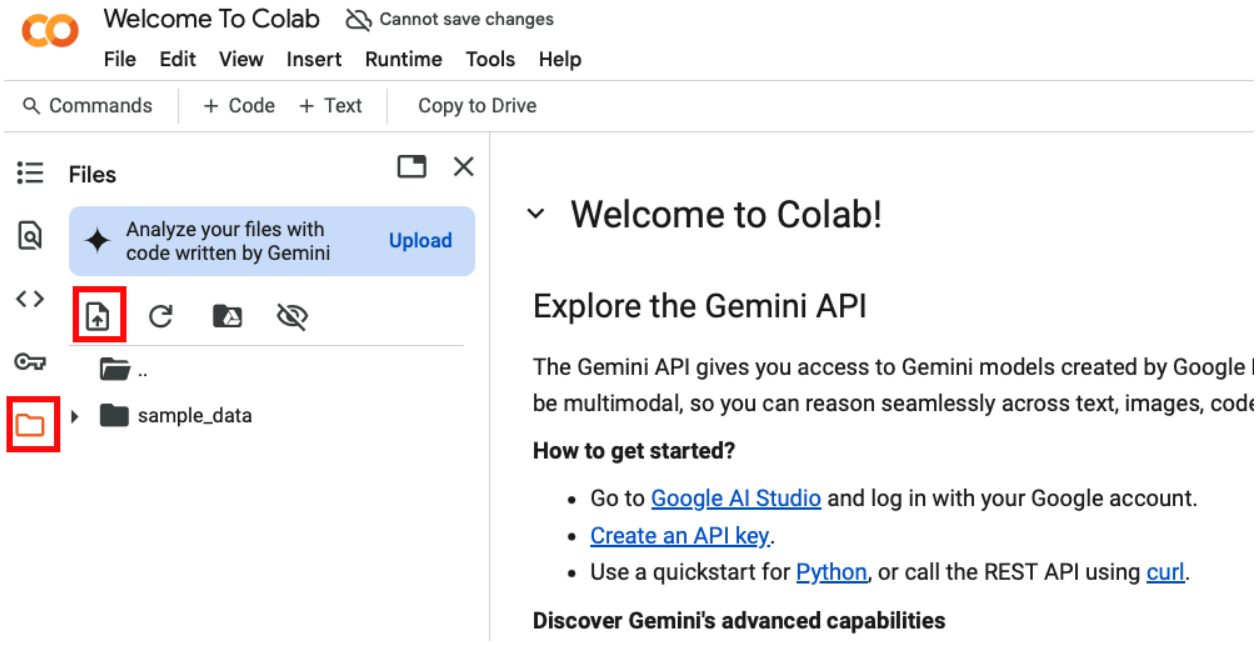
Step 4: Setting up R in Colab Notebook

- R core is primarily deployed in Colab notebook using IRKernel.
- You can see which packages are preloaded in Colab IRKernel by running the following code: `utils::installed.packages()`

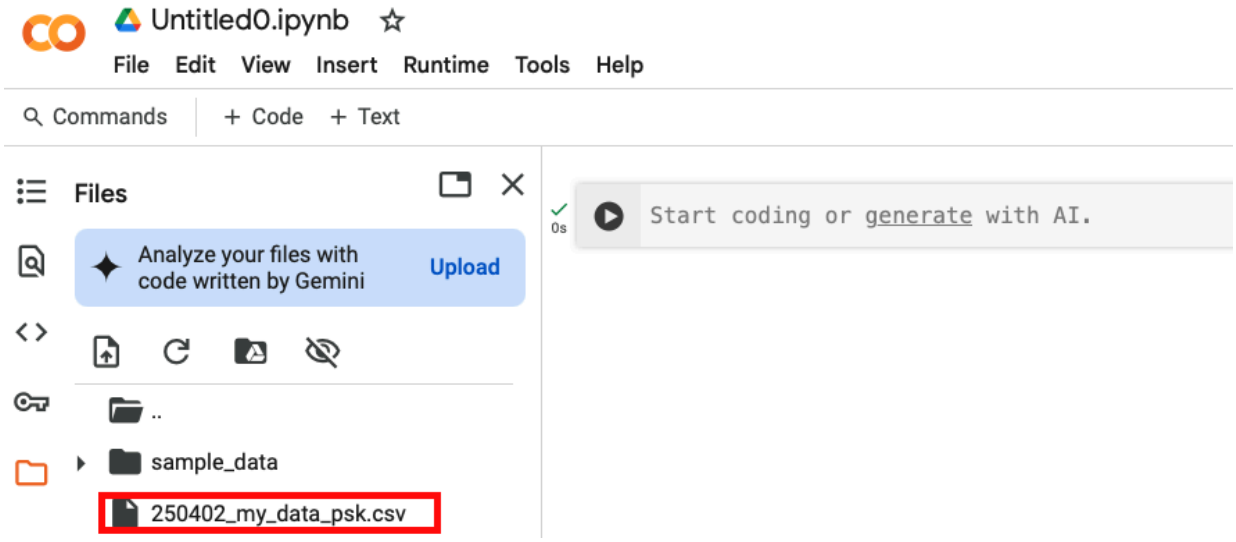
- You can also try running some other commands to get a sense of which packages are loaded and not loaded.
- If packages you want are not installed, just like Rstudio, run the command: `install.packages('name here')`
- **<u>Note: Everytime you close Google colab the workspace is wiped off. So, you will need to run the install command at the start of every session.</u>**

Step 5: Uploading data to Google Drive

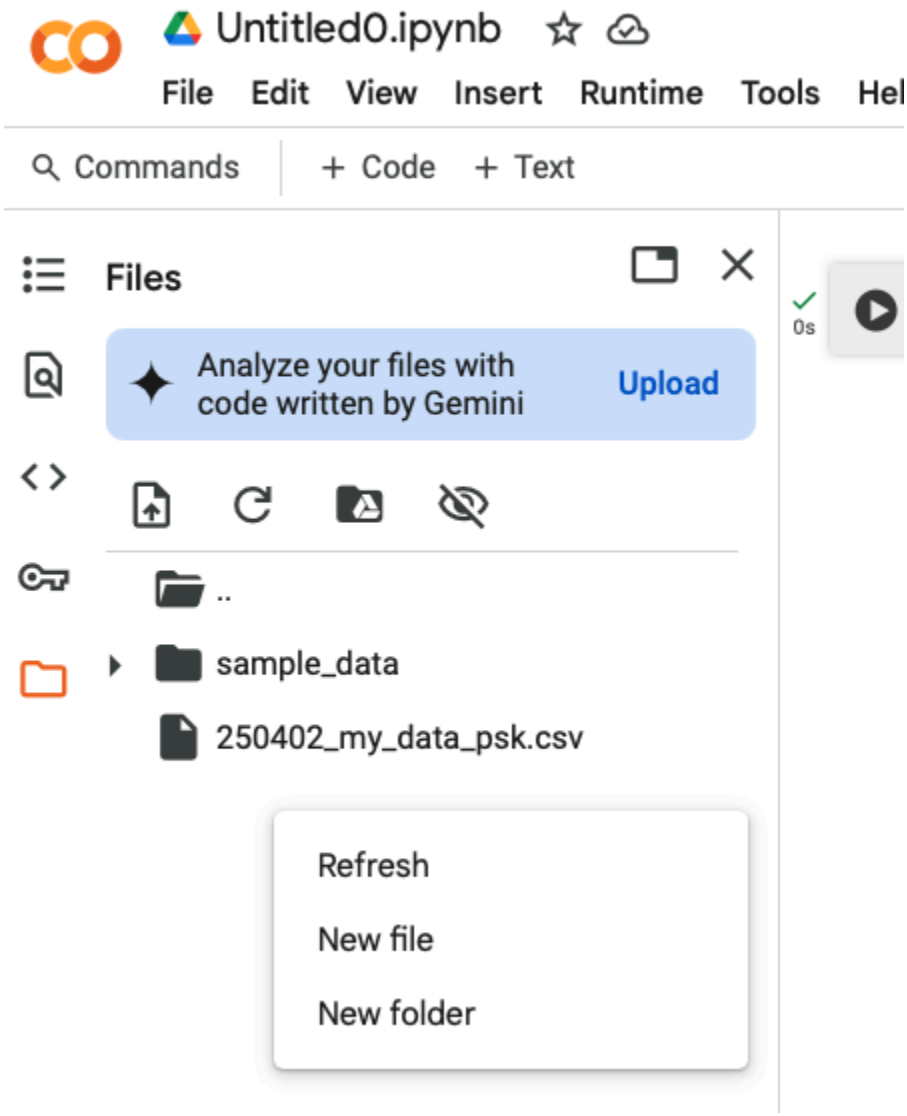
- By default, Google Colab draws a working directory from the root folder of your google drive. So all the data, pre-existing work needs to be in a folder that is available on your google drive. Connecting a relevant folder which contains data from your computer to the Colab notebook requires an upload to google drive.
- To upload a relevant dataset on your computer, click on the folder icon on the extreme right sidebar of your display. Then click on the 'upload' icon which will open the Explorer window (windows) or Finder (macOS) on your computer. Navigate to the relevant folder and upload the data file (.csv, .xlsx, .dat, etc.) to the drive.



- **<u>You will need to repeat this process each time you restart session on Colab notebook since the workspace is wiped off at each exit.</u>**
- To see if your upload was successful, check if the file appears on the left side pane below '/sample_data' sub-folder.



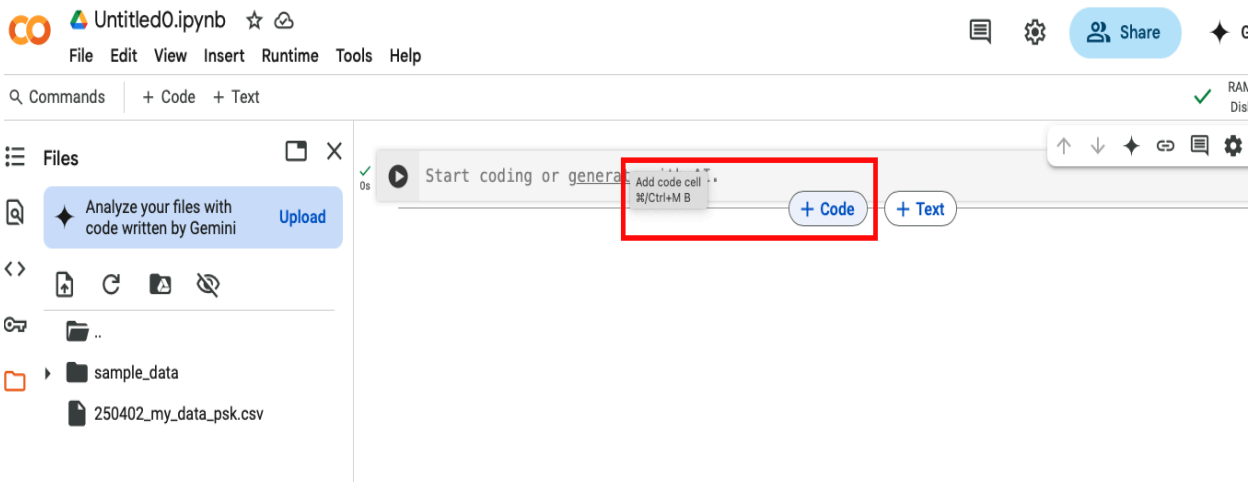
- You can manually create a new folder inside your google colab environment by right clicking near the data folder and clicking on “New Folder”.



- You can change the working directory of your workspace to this new folder by adding the following command to your code;
    - `setwd("./NAME OF YOUR FOLDER")`
    - `getwd() #to check if the change was successful.`
- **Again, repeat this step every time you start a new session!**

Step 6: Adding code to notebook

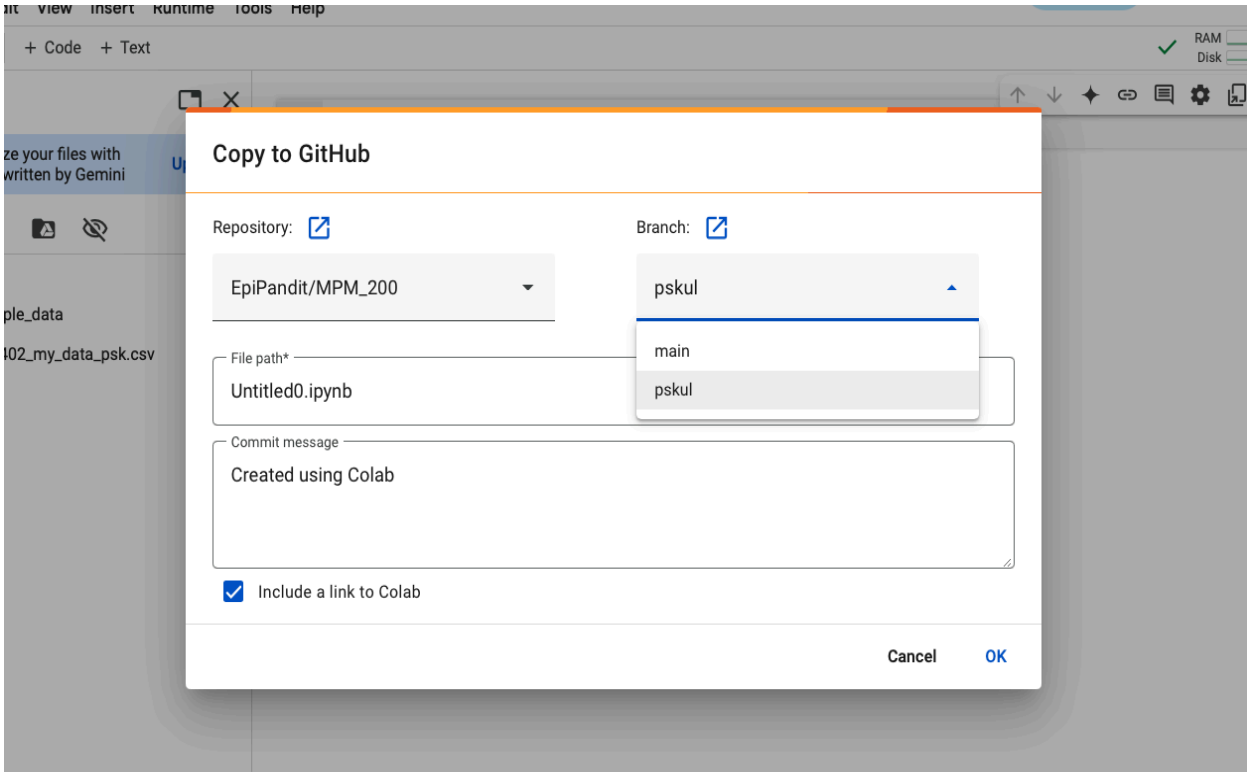
- After your data is mounted and all relevant packages are installed, you can keep adding code to your notebook just like in Rstudio. We strongly recommend that you create a new code chunk for different blocks of code. To make a new chunk, click on ‘+Code’ or the shortcut shown in the image below:



- You can also add text chunks in between your codes that use the same rules as Markdown or RMarkdown.

Step 7: Connecting Colab with Github

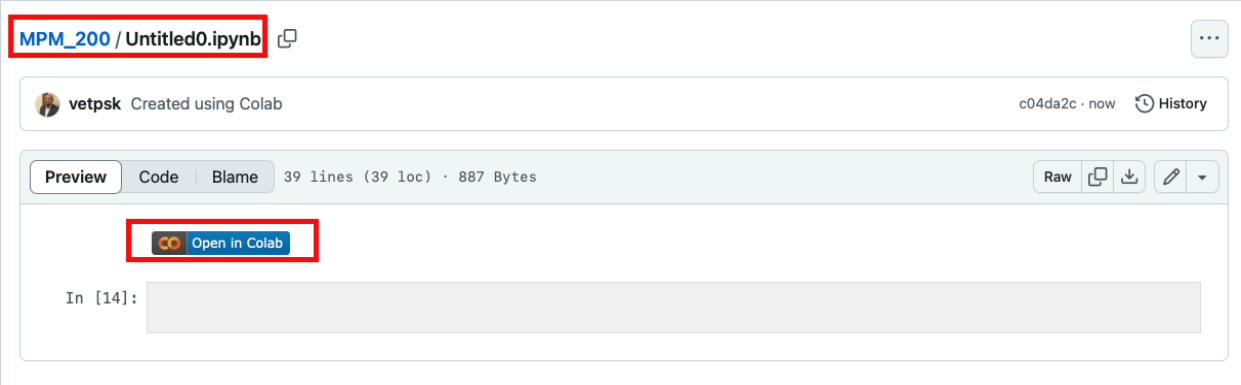
- In order to store the Colab notebook on github repository, navigate to the menu bar and click on File > Save a Copy in Github
- When a pop-up window appears, select the right repository from the dropdown list of repositories and select the right branch for upload. In the image below, there is an example of this which is storing the notebook on Github Repository named 'MPM_200' on the branch 'pskul'' which is a branch of this repository along with the admin branch 'main'..



- **Under no circumstances, upload your notebook on the repository using the 'main' branch. It is intended for the repository administrators. If you do so by mistake, please contact the course teachers or the TA. It will result in loss of all the notebooks on the repository including all assignments.**
- If you don't see a branch with your name within the repository, follow the steps in Github guide to create one for yourself.

Step 8: Making changes to your Colab notebook from the Github repository

- If your Colab notebook is successfully stored in the Github repository, you can open it through Github by navigating to the relevant folder and clicking on the link to Colab. This will prompt your browser to open a separate tab with the Colab environment.



- Make sure you push all changes to the github after editing your notebook. We strongly recommend you to save a copy of your notebook on Colab for future reference or loss of data.

Happy Coding! :-)