

Virtual Class 1

Victor Adrian

2025-03-14

Intro

We will use RStudio.

Concepts

Click to expand (works only in HTML)

- **Data Mining and Big Data:** extract key data from large and scattered datasources using different methods (statistics, etc.).
- **Data Architecture:** constructs the basis for everything to work as intended or expected.
- **Machine Learning:** differs from econometrics 'cuz does not intend to explain the reasons of a result. Econometrics is more “academic”. ML is a black, Pandora box.
 - Supervised: learns from labeled data.
 - Unsupervised: finds patterns in non-labeled data.

Data analysis steps

1. Define problem.
2. Collect data.
3. Clean and prepare data.
4. Explore and analyze data.
5. Model data.
6. Evaluate and interpret data.
7. Implement and communicate.

Class standards

- **R** for most of the course, and a little bit of Python and SQL.
- **GitHub** repo for course materials.
- Try not to use GPT, instead opt for **web searches** (stackoverflow, blogs, etc.). Then go to AI, then the forum, and lastly, the teacher. **Prioritize old school methods.**

When you say you are forking a repository (repo) you are basically creating a copy of the repo under your GitHub ID. The main point to note here is that any changes made to the original repo will be reflected back to your forked repositories if you fetch and rebase them. However, if you make any changes to your forked repo you will have to explicitly create a pull request to the original repo in case you want to commit your changes on it. If your pull request is approved by the administrator (project maintainer) of the original repo, then your changes will be committed/merged with the existing original code-base. Until then, your changes will be reflected only in the copy you forked.

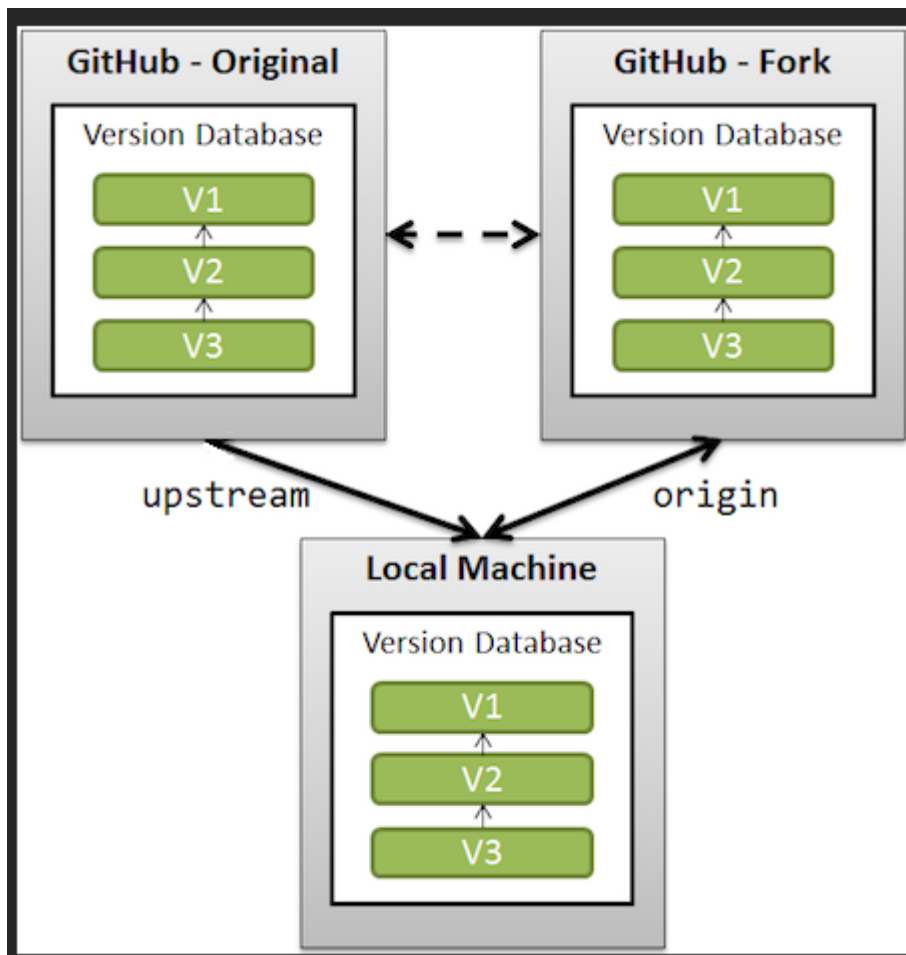
Note that after forking you can clone your repository (the one under your name) locally on your machine. Make changes in it and push it to your forked repository. However, to reflect your changes in the original repository your pull request must be approved.

Fork is a clone. It emerged because, you cannot push to others' copies without their permission. They make a copy of it for you (fork), where you will have write permission as well.

Related info [here](#).

66 @TestSubject528491 no, with a fork, that means you are cloning the upstream repo as *your own* repo on the GitHub server side. Then you can locally clone that new "fork" repo on your computer and freely push back on it, since you are the creator and owner of that fork – VonC Aug 3, 2013 at 19:00

14 To me, the key point is that you can't submit a PR from your local copy *unless you're declared to be a contributor*. I'm so used to submitting PRs from my local repo, but that's because I'm always marked as a contributor. If you think about it, to submit a PR you have to push a branch to the remote repo and then create the PR. I guess it makes sense if you don't want random people creating branches on your repo. And that you'd prefer them to fork it and submit PRs that way instead. – Adam Zerner Feb 22, 2016 at 1:18



Other stuff

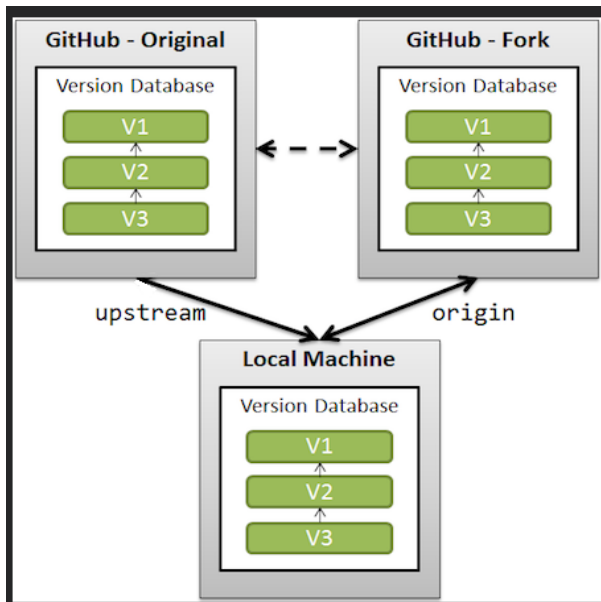
```
# Embed images alternative with Python
# Why? When exported as a html file, image links may broke
from IPython.display import Image
screens = "C:/Users/victo/OneDrive/1_Brocolito/8_Multi/Imágenes/Screenshots/"
pic_one = Image(filename = screens + "Screenshot 2025-03-15 202335.png")
pic_two = Image(filename = screens + "Screenshot 2025-03-15 205749.png", width=300, height=300)
```

```
pic_one
```

66 @TestSubject528491 no, with a fork, that means you are cloning the upstream repo as *your own* repo on the GitHub server side. Then you can locally clone that new "fork" repo on your computer and freely push back on it, since you are the creator and owner of that fork. – VonC Aug 3, 2013 at 19:00

14 To me, the key point is that you can't submit a PR from your local copy *unless you're declared to be a contributor*. I'm so used to submitting PRs from my local repo, but that's because I'm always marked as a contributor. If you think about it, to submit a PR you have to push a branch to the remote repo and then create the PR. I guess it makes sense if you don't want random people creating branches on your repo. And that you'd prefer them to fork it and submit PRs that way instead. – Adam Zerner Feb 22, 2016 at 1:18

pic_two



```
import os
os.environ['PATH'].split(";")
```

```
['C:\\Users\\victo\\miniforge3\\envs\\uni',
 'C:\\Users\\victo\\miniforge3\\envs\\uni\\Library\\mingw-w64\\bin',
 'C:\\Users\\victo\\miniforge3\\envs\\uni\\Library\\usr\\bin',
 'C:\\Users\\victo\\miniforge3\\envs\\uni\\Library\\bin',
 'C:\\Users\\victo\\miniforge3\\envs\\uni\\Scripts',
 'C:\\Users\\victo\\miniforge3\\envs\\uni\\bin',
 'C:\\Users\\victo\\miniforge3\\condabin',
 'C:\\Program Files\\Microsoft MPI\\Bin',
 'C:\\WINDOWS\\system32',
 'C:\\WINDOWS',
```

```
'C:\\WINDOWS\\System32\\Wbem',  
'C:\\WINDOWS\\System32\\WindowsPowerShell\\v1.0',  
'C:\\WINDOWS\\System32\\OpenSSH',  
'C:\\Program Files (x86)\\Microsoft SQL Server\\160\\DTS\\Binn',  
'C:\\Program Files (x86)\\Microsoft SQL Server\\160\\Tools\\Binn',  
'C:\\Program Files\\Microsoft SQL Server\\160\\Tools\\Binn',  
'C:\\Program Files\\Microsoft SQL Server\\Client SDK\\ODBC\\170\\Tools\\Binn',  
'C:\\Program Files\\Microsoft SQL Server\\160\\DTS\\Binn',  
'C:\\Program Files\\Git\\cmd',  
'C:\\Program Files\\GitHub CLI',  
'C:\\Users\\victo\\AppData\\Local\\Microsoft\\WindowsApps',  
'C:\\Users\\victo\\vscode\\bin',  
'C:\\Users\\victo\\AppData\\Local\\Pandoc',  
'C:\\Users\\victo\\AppData\\Local\\Programs\\MiKTeX\\miktex\\bin\\x64']
```