Versioning ML Models & Automating ML Pipelines Effectively using DVC & CML

A picture containing graphics, font, logo, graphic design

Description automatically generated

1. To understand why versioning or models and data in important:

A picture containing text, diagram, screenshot

Description automatically generated

Machine Learning/Data Science Pipeline

* Steps:
* Cloud storage
* Download the data from cloud
* Filter/preprocess the data
* Train models
* Restore the model to cloud storage
* If the model is performing well we have to put it into production
* Create a microservice from storage
* Other application services can use this microservice as an API

This is typical machine learning pipeline.

1. The main is experimentation in this process.

A picture containing text, screenshot, font, logo

Description automatically generated

1. We version the model so we don’t lose the model & get the new model as well.
2. To solve this problem, we use DVC (Data Version Control)
3. By Git we can track the code changes, same DVC does in the data & model files.

A picture containing text, screenshot, font, diagram

Description automatically generated

1. To push any code into Git we use Git-add, Git-commit & Git-push, similarly we use in DVC
2. For DVC repository will be any cloud storage
3. To push files we use

* dvc-add
* dvc-commit
* dvc-push
* when we do dvc-add Git will stop tracking the model changes. Because when we do dvc-add dvc will automatically generate this kind of file which is model.pkl.dvc file.
* This model.pkl.dvc file will contain only the path of the data or print of the data which is actually stored, so it will have information about where the actual file will be stored after pushing it to cloud.
* So when we use the dvc the file will be pushed to cloud.

A screenshot of a computer

Description automatically generated with medium confidence

* To update the changes we create a new branch same applies to dvc.
* When we change/create a branch in DVC it creates a data version , we can version the code.

1. Experimentation in DVC:

A picture containing text, screenshot, font, diagram

Description automatically generated

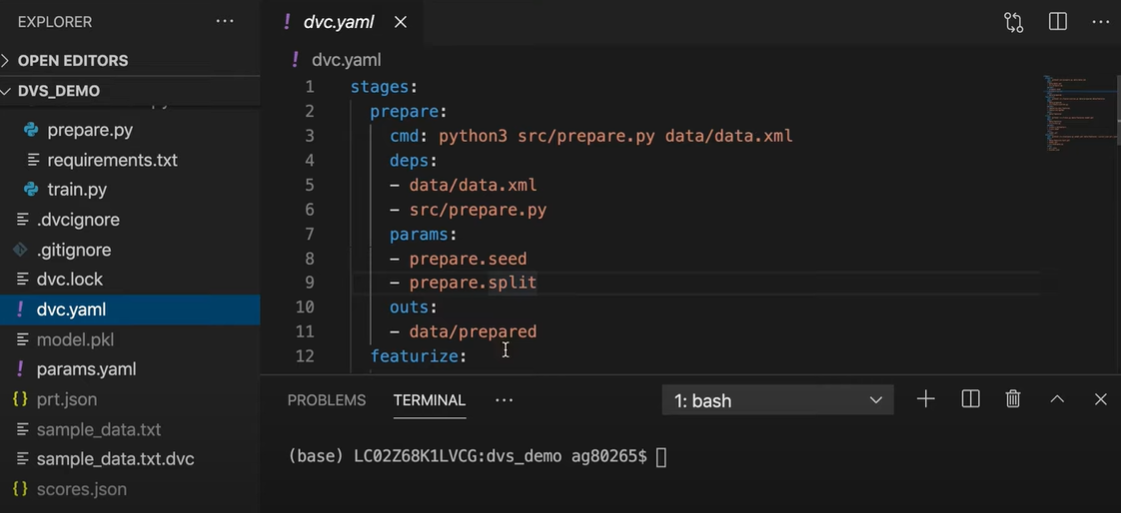
* DVC is Git for data & models
* Git tracks the code changes in our model
* DVC----- data & model files

To understand it better consider an example:

Consider we have:

1. Data
2. Source code
3. Dvc.yaml file

* DVC is not just for versioning but it also helps in automating the pipeline.
* We do that by using DVC.yaml file.
* Yaml file contains stages and information
* Our ML pipeline has stages so yaml file will contain different stages



* Dependencies means to execute the file we need few files

**dvc repro**

* use command dvc reproduce
* by using this command, we get all the stages information in the terminal.
* So as the Git tracks the changes in the code DVC tracks the changes in the data.
* So if data is not changing it wont run gain.

To understand that, suppose we change the train parameter n\_estimators from 50 to 100 in params.yaml file & agin if we run the “dvc repro” command it will run only those stages which are dependent on this n\_estimators parameter.

* So it did not run prepare or factorize stage it only run train & evaluate stage.

1. Versioning:

* Suppose we are master branch & now we want to version our data (sample\_data.txt)
* We have already added dvc on it so it has created sample\_data.txt.dvc file which will be uploaded to github repository & sample\_data.txt will be uploaded to remote storage or cloud storage.
* To do another experiment & change the data we can do that by just changing the git branch
* **git branch** -------- it has list of branches
* we can change the branch to v1 by

**git checkout v1**

* **dvc checkout sample\_data.txt**

the data is changed now

* if we run the command **dvc dag** it will show us all the pipeline stages sequentially.
* This was versioning with simple commands

**How can we version now using CI/CD pipeline?**

Continuous Machine Learning (CML) is CI/CD for Machine Learning projects.

DVC provides a tool CML, which is used to build CI/CD pipelines for machine learning projects.

We can also generate the reports in CML.