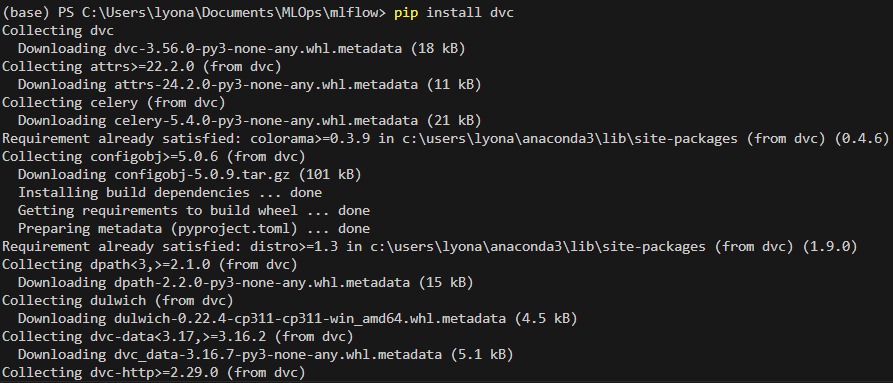
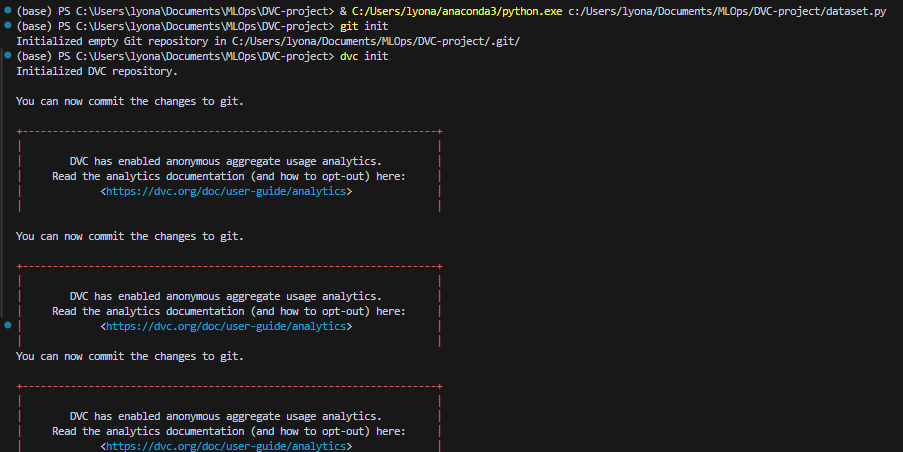
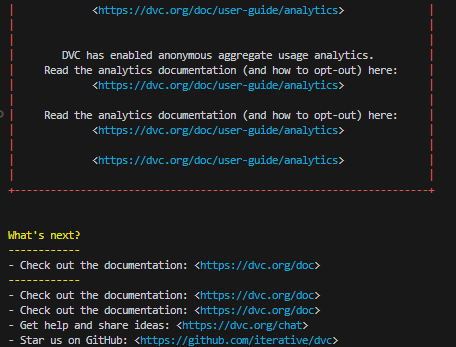
1. **Install DVC**



Untuk menginstall DVC ke dalam environment python, dijalankan perintah *pip install dvc*.

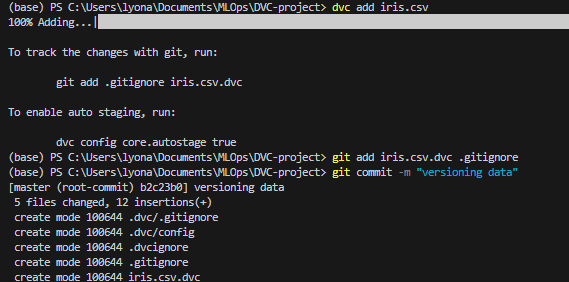
1. **git init dan dvc init**





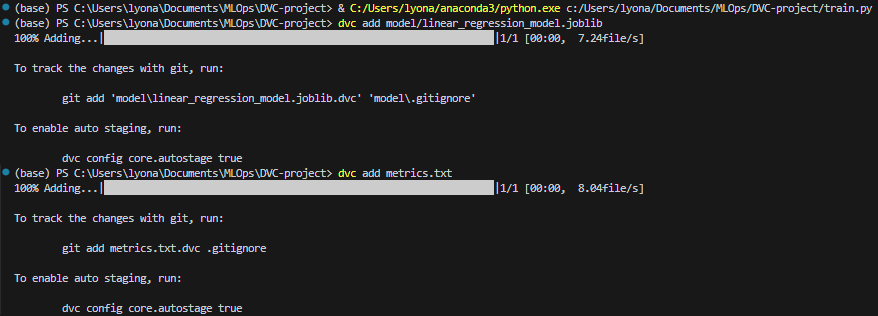
Perintah *git init* dilakukan untuk menginisialisasi repository Git dalam direktori proyek. Perintah *dvc init* berfungsi untuk menginisialisasi DVC dalam proyek. Pada gambar juga dijalankan kode dataset.py untuk menyimpan dataset iris.csv.

1. **dvc add, git add, dan git commit**



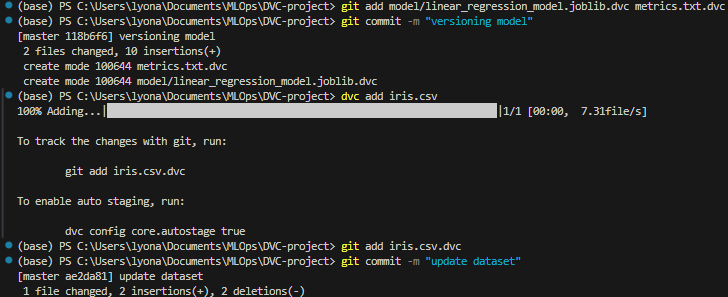
Untuk menambahkan file ke pelacakan DVC dilakukan perintah *dvc add nama\_file*. Dalam kasus ini, digunakan file iris.csv. Setelah menjalankan perintah ini, DVC akan membuat file bernama iris.csv.dvc yang berisi informasi tentang file iris.csv. Setelah itu, dilakukan perintah *git add iris.csv.dvc* yang berfungsi untuk menambahkan file ke staging area di Git, mempersiapkan file tersebut untuk dicommit. Perintah *git commit -m “pesan”* dilakukan untuk membuat commit di Git.

1. **dvc add (2)**



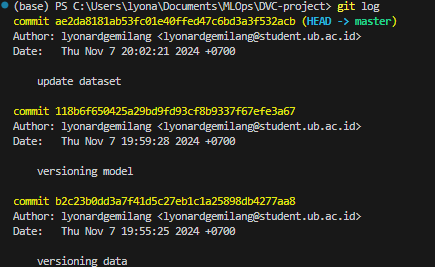
Kalau perintah *dvc add* sebelumnya digunakan untuk menambahkan dataset, kali ini digunakan untuk menambahkan file model ke pelacakan DVC. Pertama-tama dijalankan terlebih dahulu kode train.py yang melatih membuat model linear regression mengenai data iris. Kode tersebut juga menghasilkan metrics sehingga dilakukan lagi *dvc add*.

1. **git add, git commit, perubahan dataset**



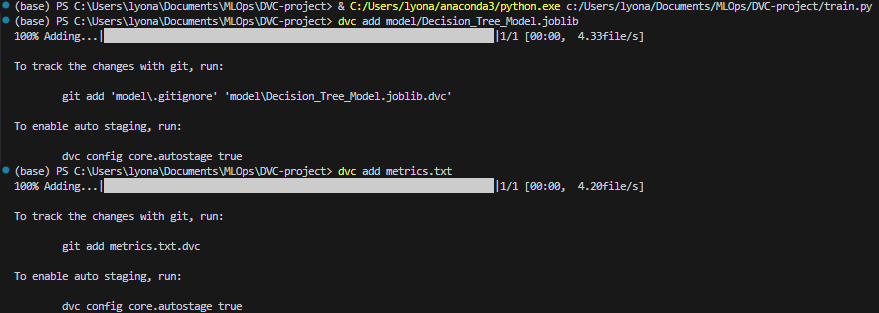
Pada gambar di atas dilakukan kembali *git add* dan *git commit* untuk menambahkan dan membuat commit mengenai model yang sudah dibuatbeserta metrics nya. Setelah itu, dicoba perubahan dataset seperti menambahkan 1 baris data dan dilakukan kembali langkah sebelumnya.

1. **git log**



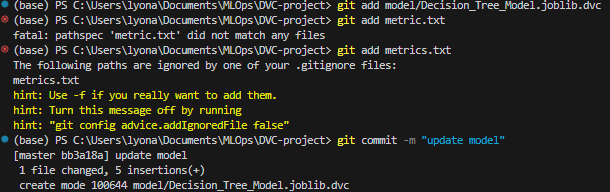
Perintah git log menampilkan history commit di repository git. Dapat dilihat id setiap commit yang di mana id tersebut berguna apabila ingin dikembalikannya repository Git pada commit id tertentu. Hal tersebut dapat dilakukan dengan melakukan perintah *git checkout* *commit\_id*.

1. **Perubahan model**



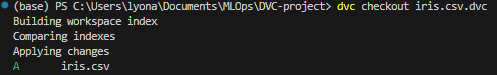
Dilakukan langkah yang sama dalam melakukan perubahan model seperti sebelumnya. Hanya saja yang berbeda kali ini pada train.py menggunakan model decision tree sehingga terbuatlah model Decision\_Tree\_Model.joblib beserta metrics nya.

1. **git add dan git commit**



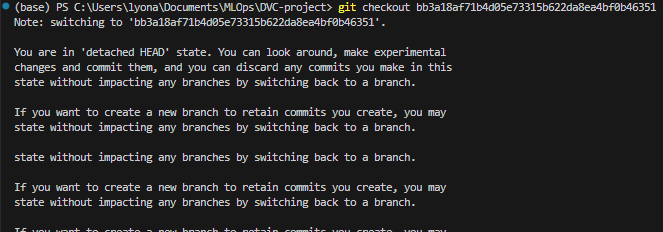
Pada gambar di atas dilakukan kembali *git add* dan *git commit* untuk menambahkan dan membuat commit mengenai model yang sudah dibuatbeserta metrics nya.

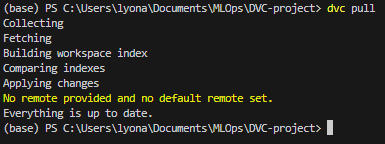
1. **dvc checkout**



Perintah *dvc checkout* berfungsi untuk mengembalikan file yang dilacak DVC ke versi yang sesuai dengan commit Git yang sedang aktif saat ini. Karena tidak dilakukan git checkout untuk pindah ke repo git pada commit id tertentu, tentunya commit Git yang dipakai adalah yang terbaru. Pada kasus ini, dihapus terlebih dahulu dataset iris.csv pada direktori, lalu dijalankan perintah *dvc checkout iris.csv.dvc.* Pada gambar dapat terlihat bahwa iris.csv berhasil dikembalikan.

1. **git checkout dan dvc pull**





Perintah *git checkout* digunakan untuk mengembalikan repository Git ke commit tertentu. Perintah ini dapat digunakan bersama DVC agar dapat memulihkan versi dataset dan model. Perintah *dvc pull* digunakan untuk mengunduh file atau model yang dilacak oleh DVC pada commit yang aktif saat ini.

**Repository Github:**

Berikut merupakan link repository Github:<https://github.com/lyonardgemilang/MLOps/tree/master>