

Tracking changes and switching between versions

Now that we are versioning and sharing our data and models using DVC, we can explore how to track changes and how to switch between different versions.

We can use dvc status to check the status of the files tracked by DVC. A benefit to using DVC when working with data is that it adds a cache layer. This makes it easy to switch between different versions of data without having to replace the entire dataset (e.g. by downloading). Whenever we update an image dataset by adding more images, for example, we will only retrieve the added images while keeping the older ones as they are.

- + Automatic Zoom +

When we run dvc status, DVC will tell us what has changed in our data:

```
# run command (no changes to data)
dvc status

# output
Data and pipelines are up to date.

# run command (data has changed)
dvc status
```

We can switch between versions by using dvc checkout, just like we can switch between versions of code using git checkout. We use these commands in unison: first checkout the required code version in Git, and then checkout its associated data in DVC.

git checkout v1.0 dvc checkout

output

datadir.dvc:

changed outs:

modified: datadir

1 of 2

