Practical Data Science – How to Track your Development Process with DVC

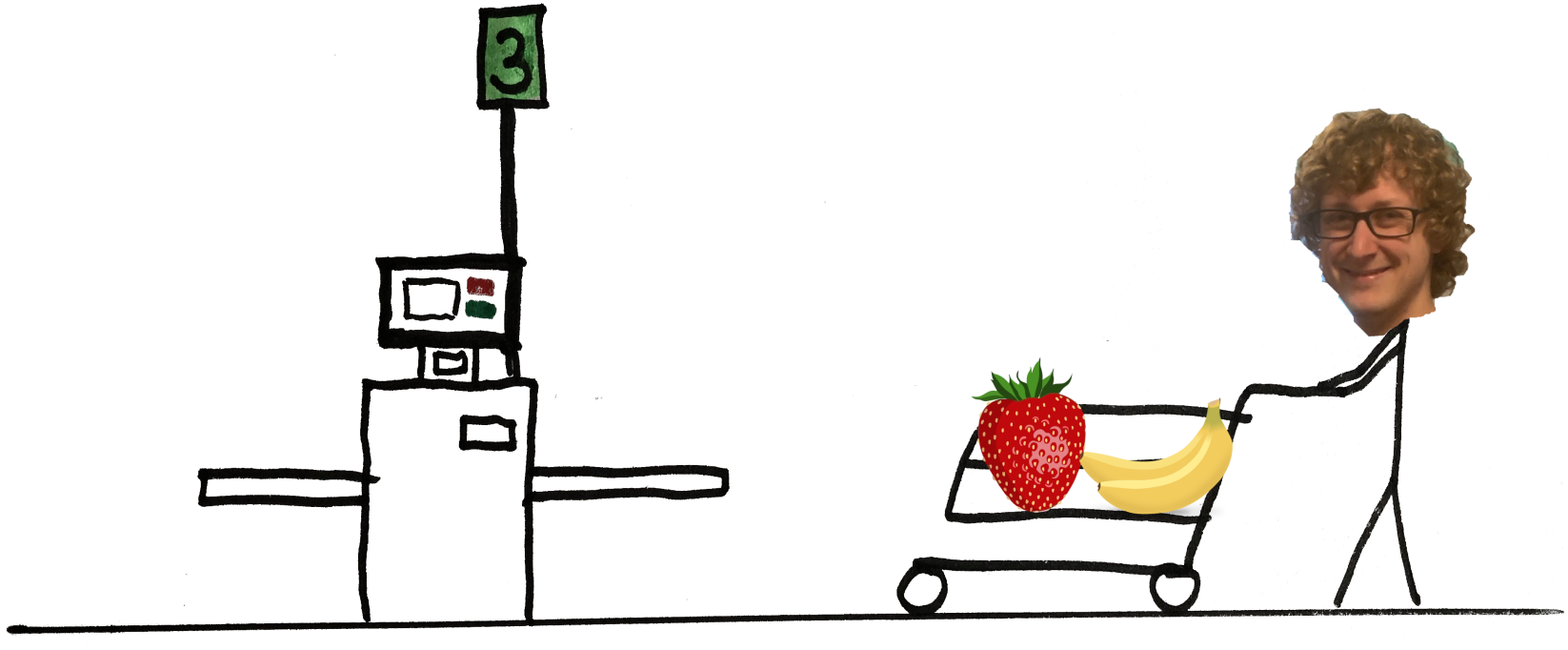
About Us

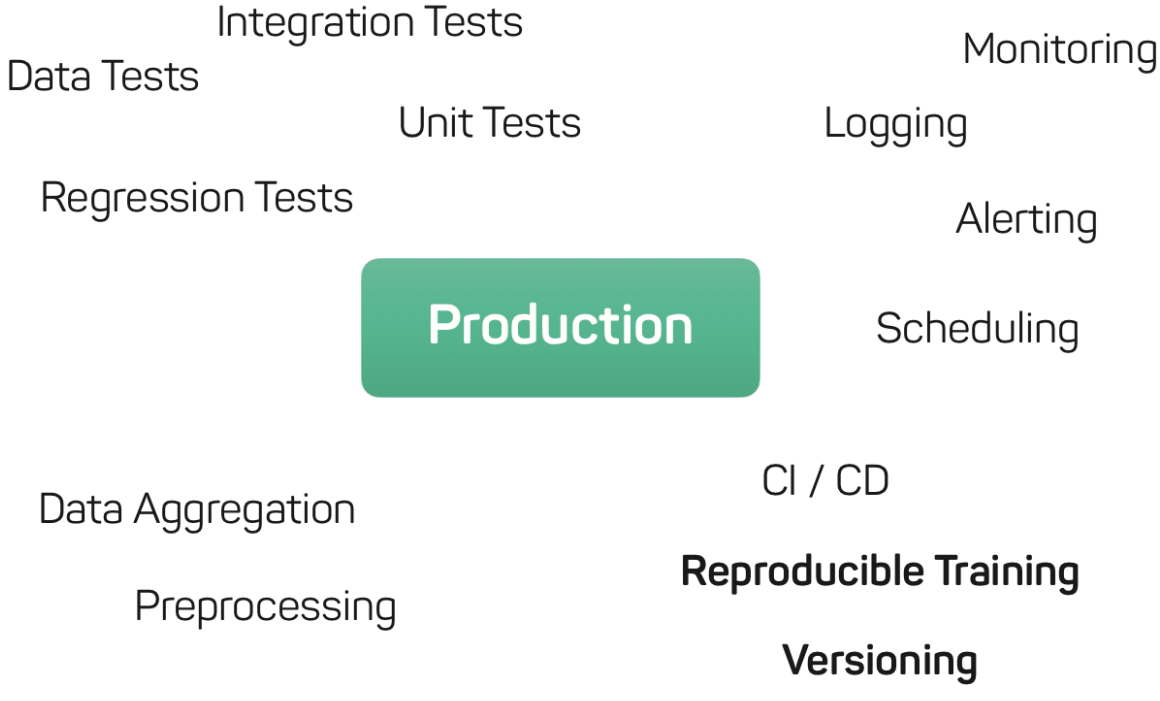
Mark Keinhörster
Data Architect

Tim Sabsch
Data Scientist

codecentric

Our Project

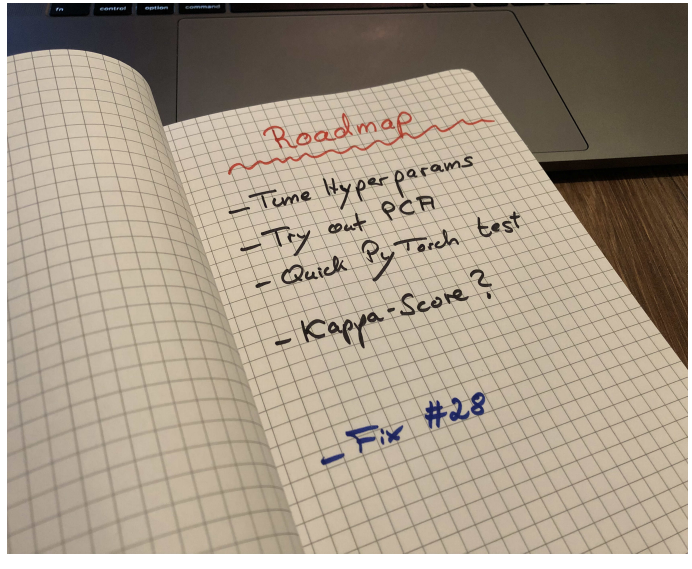


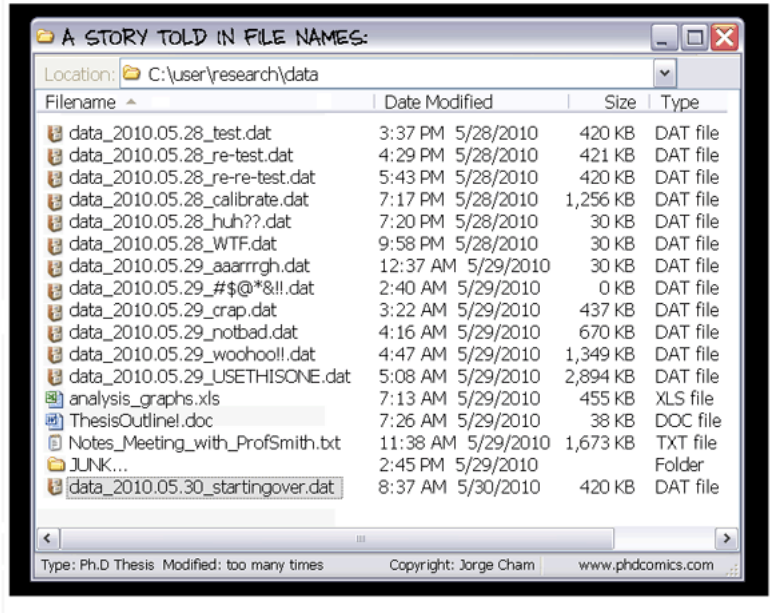

Integration Tests
Data Tests
Monitoring
Unit Tests
Logging
Regression Tests
Alerting
Production
Scheduling
Data Aggregation
CI / CD
Reproducible Training
Preprocessing
Versioning

Data and Model Versioning with DVC

Why should I use this? Reproducibility!

Data Science is experiment-driven …



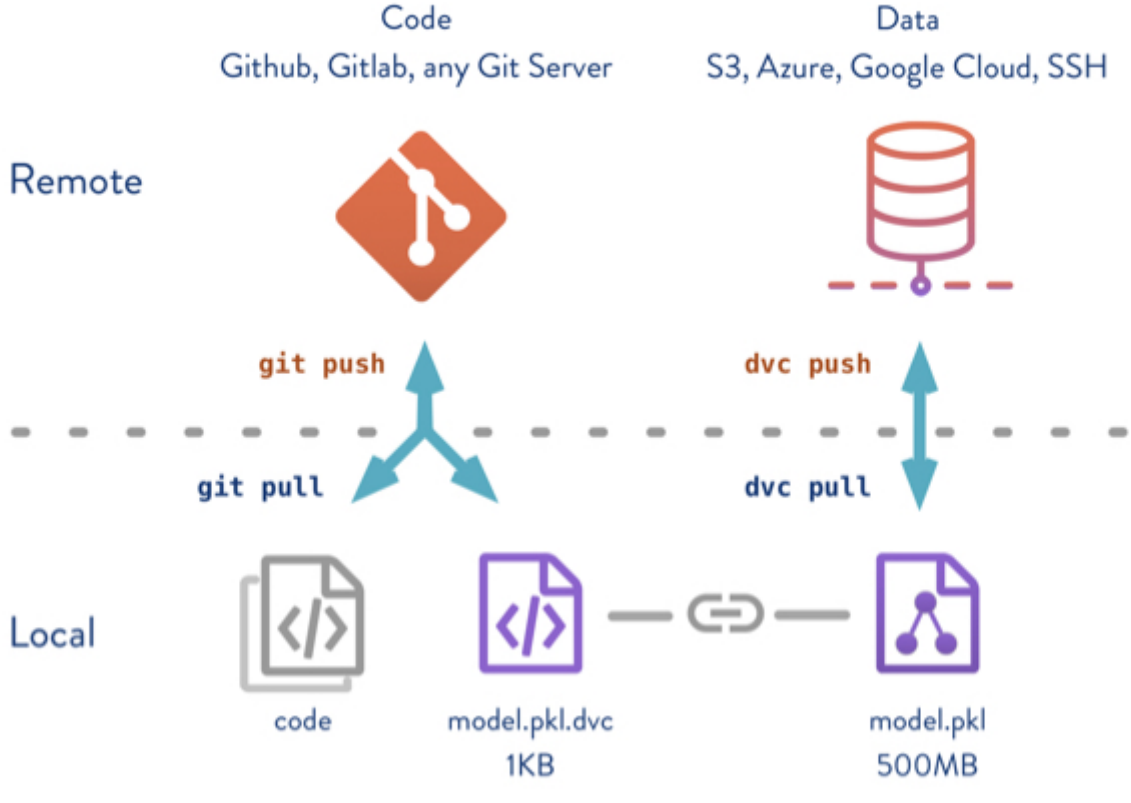


… and collaborative!

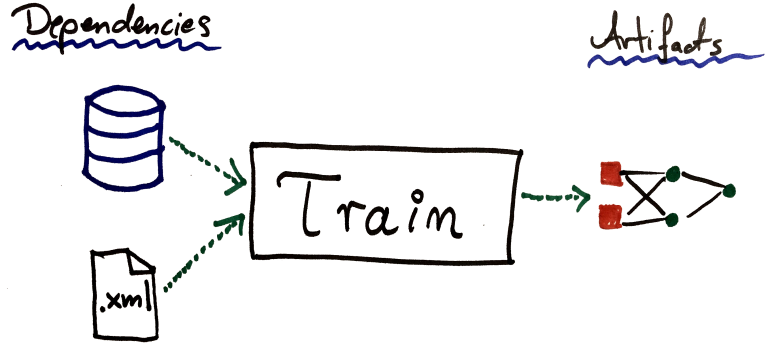
```
Tims-MacBook-Pro:code tsabsch$ git log --graph --decorate --oneline
*   fc88d36 (HEAD -> master) Merge branch 'tim_fix_zerodivisionerror'
|\
| * 2cc23f0 (tim_fix_zerodivisionerror) Fix ZeroDivisionError
* |   e486d76 Merge branch 'mark_batchnorm'
|\ \
| |/
|/|
| * b988f6e (mark_batchnorm) Decrease BN threshold
| * c83736d Add batch normalisation
* | f80b108 Add preprocessing
|/
* a87aa63 Add simple tensorflow model
* 7c0d21e Initial commit
Tims-MacBook-Pro:code tsabsch$
```

DVC to the Rescue!

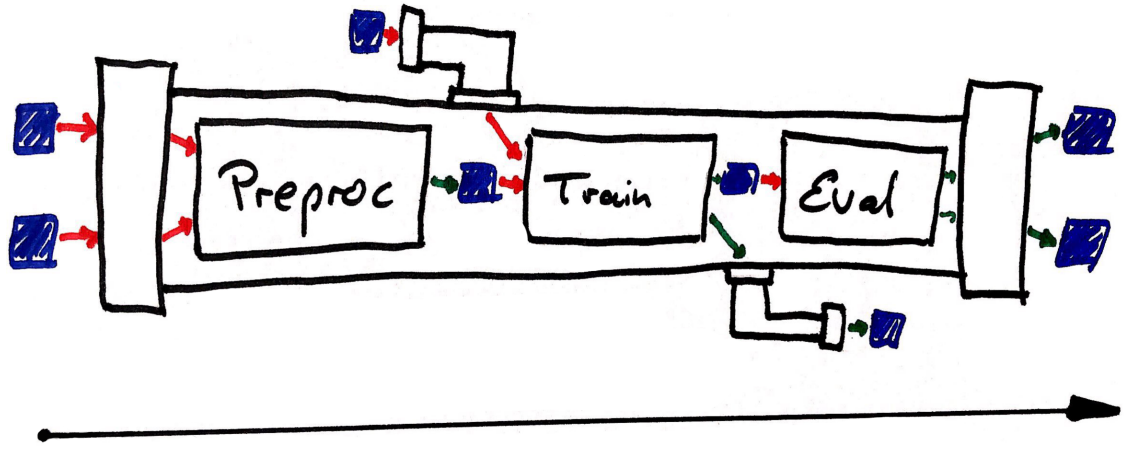
- Large File Storage (similar to git-lfs)
  - Metadata in git (name, hashsum etc.)
  - Contents in cache + remote storage



- Pipelines
    - Optimise reproduction by splitting your process into stages
    - Stages store dependencies, processing and output artifacts

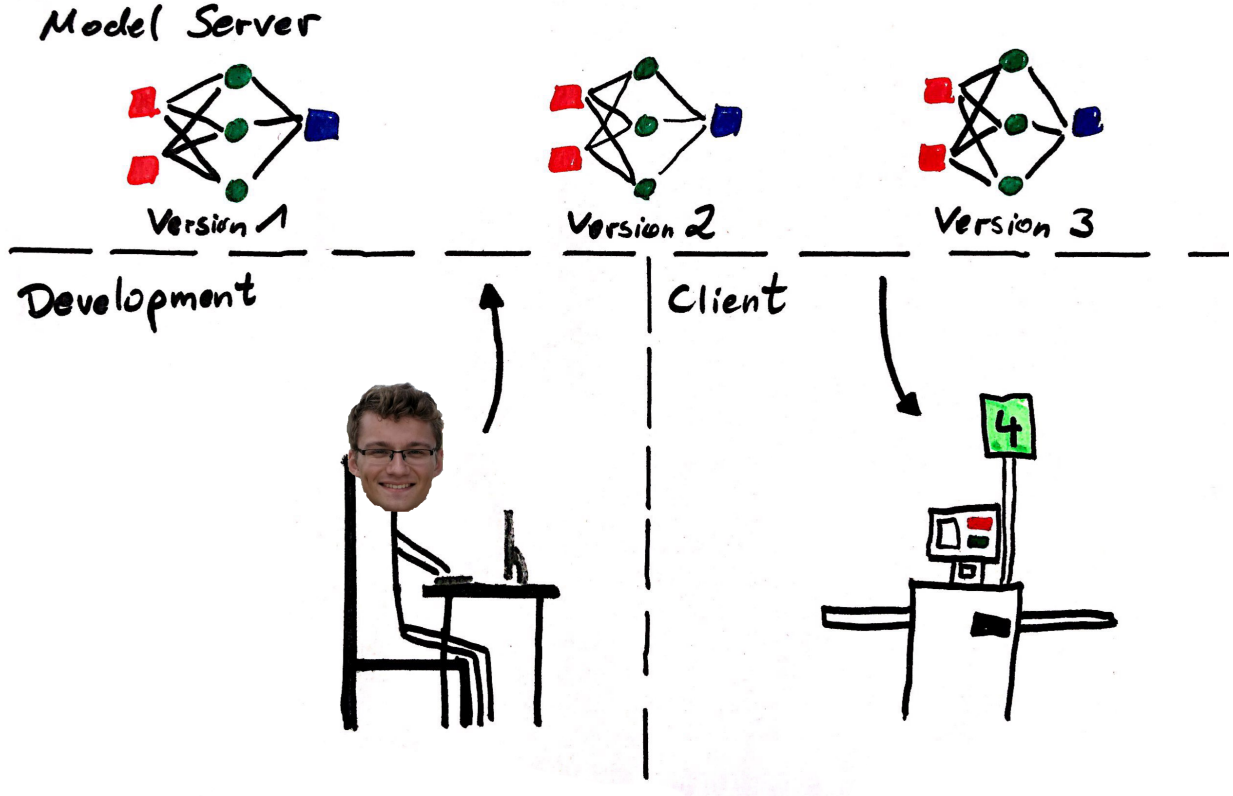


- Pipelines
    - Connect stages to pipeline via dependencies



Continuous Delivery with TensorFlow Serving

TensorFlow Serving



Developing a fruit classifier ”the proper way”

Tools + Software Stack

- Data: Fruits360 (https://www.kaggle.com/moltean/fruits) data set (Kaggle, Github)
- Classification with Tensorflow + Keras
- Versioning with DVC
- Continuous delivery with Tensorflow Serving
- Live prediction on our webcam

Prepare our version control

In [1]:
```
%cd /dvc/code
!git init
!dvc init
```

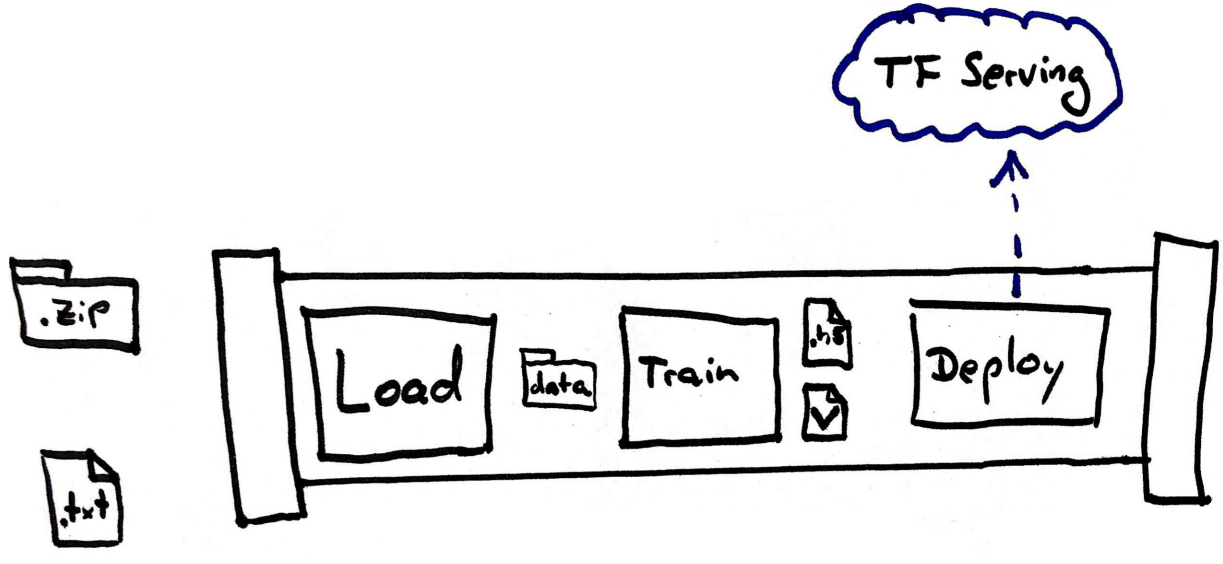
```
/dvc/code
Initialized empty Git repository in /dvc/code/.git/
Adding '.dvc/lock' to '.dvc/.gitignore'.
Adding '.dvc/config.local' to '.dvc/.gitignore'.
Adding '.dvc/updater' to '.dvc/.gitignore'.
Adding '.dvc/updater.lock' to '.dvc/.gitignore'.
Adding '.dvc/state-journal' to '.dvc/.gitignore'.
Adding '.dvc/state-wal' to '.dvc/.gitignore'.
Adding '.dvc/state' to '.dvc/.gitignore'.
Adding '.dvc/cache' to '.dvc/.gitignore'.

You can now commit the changes to git.
```

Set remote storage location for large files

In [2]:
```
!git init --bare /tmp/git-storage/.git
!git remote add local /tmp/git-storage/.git
!dvc remote add -d -f  local_storage /tmp/dvc-storage
```

```
Initialized empty Git repository in /tmp/git-storage/.git/
Setting 'local_storage' as a default remote.
```

Create Pipeline: Load data → Train → Deploy



In [3]:

```
# our fruits to be classified
!echo Banana,Strawberry > fruits.txt

# Start with the first pipeline step
!dvc run -f load_data.dvc \
         -d load_data.sh -d Fruit-Images-Dataset-master.zip -d fruits.txt \
         -o data \
         ./load_data.sh
```

```
Running command:
        ./load_data.sh
Adding 'data' to '.gitignore'.
Computing md5 for a large number of files. This is only done once.
Multi-Threaded: ...Saving information to 'load_data.dvc'.

To track the changes with git, run:

        git add .gitignore load_data.dvc
```

In [4]:
```
!git add load_data.sh load_data.dvc
!git commit -m "Add data loading"
!git push -u local master
!dvc push
```

```
[master (root-commit) 2136165] Add data loading
 4 files changed, 40 insertions(+)
 create mode 100644 .dvc/.gitignore
 create mode 100644 .dvc/config
 create mode 100644 load_data.dvc
 create mode 100755 load_data.sh
Counting objects: 7, done.
Delta compression using up to 6 threads.
Compressing objects: 100% (6/6), done.
Writing objects: 100% (7/7), 957 bytes | 0 bytes/s, done.
Total 7 (delta 0), reused 0 (delta 0)
To /tmp/git-storage/.git
 * [new branch]      master -> master
Branch master set up to track remote branch master from local.
Multi-Threaded: ...
```

In [5]:

```
!dvc run -f train.dvc -d fruit_detector.py -d data -o /tmp/model.h5 -o /tmp/VERSIO
N \
         python fruit_detector.py

!git add fruit_detector.py train.dvc
!git commit -m "Add training" && git push
!dvc push
```

```
Running command:
        python fruit_detector.py
Using TensorFlow backend.
Found 982 images belonging to 2 classes.
Found 330 images belonging to 2 classes.
2019-09-19 17:37:25.286049: I tensorflow/core/platform/cpu_feature_guard.cc:14
1] Your CPU supports instructions that this TensorFlow binary was not compiled
to use: AVX2 FMA
2019-09-19 17:37:25.289657: I tensorflow/core/platform/profile_utils/cpu_util
s.cc:94] CPU Frequency: 2592000000 Hz
2019-09-19 17:37:25.290359: I tensorflow/compiler/xla/service/service.cc:150]
XLA service 0x5653b7aea510 executing computations on platform Host. Devices:
2019-09-19 17:37:25.290427: I tensorflow/compiler/xla/service/service.cc:158]
StreamExecutor device (0): <undefined>, <undefined>
Epoch 1/5
Epoch 2/5
Epoch 3/5
Epoch 4/5
Epoch 5/5
[5.486854553222656, 0.5306462645530701]
Saving '../../tmp/model.h5' to '.dvc/cache/cd/4425c716813656b54f29f078607308'.
Saving '../../tmp/VERSION' to '.dvc/cache/c4/ca4238a0b923820dcc509a6f75849b'.
Saving information to 'train.dvc'.

To track the changes with git, run: ...
```

In [6]:

```
!dvc run -f deploy.dvc -d deploy.py -d /tmp/model.h5 -d /tmp/VERSION \
         python deploy.py

!git add deploy.py deploy.dvc
!git commit -m "Add deployment" && git push
!dvc push
```

```
Running command:
        python deploy.py
Using TensorFlow backend.
2019-09-19 17:38:25.286595: I tensorflow/core/platform/cpu_feature_guard.cc:14
1] Your CPU supports instructions that this TensorFlow binary was not compiled
to use: AVX2 FMA
2019-09-19 17:38:25.290254: I tensorflow/core/platform/profile_utils/cpu_util
s.cc:94] CPU Frequency: 2592000000 Hz
2019-09-19 17:38:25.291294: I tensorflow/compiler/xla/service/service.cc:150]
XLA service 0x55afe89fb870 executing computations on platform Host. Devices:
2019-09-19 17:38:25.291337: I tensorflow/compiler/xla/service/service.cc:158]
StreamExecutor device (0): <undefined>, <undefined>
Saving information to 'deploy.dvc'.

To track the changes with git, run: ...
```

Our pipeline is ready! Let's check it out!

In [7]:
```
%%bash
dvc pipeline show deploy.dvc --ascii
```

```
+---------------+
| load_data.dvc |
+---------------+
        *
        *
        *
  +-----------+
  | train.dvc |
  +-----------+
        *
        *
        *
  +------------+
  | deploy.dvc |
  +------------+
```

What about our live prediction?

In [9]:
```
!echo Banana,Strawberry,Walnut > fruits.txt
!dvc repro deploy.dvc
```

```
WARNING: Dependency 'fruits.txt' of 'load_data.dvc' changed because it is 'mod
ified'.
WARNING: Stage 'load_data.dvc' changed.
Running command:
        ./load_data.sh
Computing md5 for a large number of files. This is only done once.
Multi-Threaded: ...
Saving information to 'load_data.dvc'.
WARNING: Dependency 'data' of 'train.dvc' changed because it is 'modified'.
WARNING: Stage 'train.dvc' changed.
Running command:
        python fruit_detector.py
Using TensorFlow backend.
Found 1717 images belonging to 3 classes.
Found 579 images belonging to 3 classes.
2019-09-19 17:40:13.730103: I tensorflow/core/platform/cpu_feature_guard.cc:14
1] Your CPU supports instructions that this TensorFlow binary was not compiled
to use: AVX2 FMA
2019-09-19 17:40:13.733645: I tensorflow/core/platform/profile_utils/cpu_util
s.cc:94] CPU Frequency: 2592000000 Hz
2019-09-19 17:40:13.734250: I tensorflow/compiler/xla/service/service.cc:150]
 XLA service 0x562416e6a880 executing computations on platform Host. Devices:
2019-09-19 17:40:13.734284: I tensorflow/compiler/xla/service/service.cc:158]
StreamExecutor device (0): <undefined>, <undefined>
Epoch 1/5
Epoch 2/5
Epoch 3/5
Epoch 4/5
Epoch 5/5
[0.001984937349334359, 1.0]
Saving '../../tmp/model.h5' to '.dvc/cache/f2/67cf039a7e3a4172b713216bffb74b'.
Saving '../../tmp/VERSION' to '.dvc/cache/c8/1e728d9d4c2f636f067f89cc14862c'.
Saving information to 'train.dvc'.
WARNING: Dependency '../../tmp/model.h5' of 'deploy.dvc' changed because it is
```

```
WARNING: Dependency '../../tmp/model.h5' of 'deploy.dvc' changed because it is
'modified'.
WARNING: Stage 'deploy.dvc' changed.
Running command:
        python deploy.py
Using TensorFlow backend.
2019-09-19 17:40:33.783824: I tensorflow/core/platform/cpu_feature_guard.cc:14
1] Your CPU supports instructions that this TensorFlow binary was not compiled
to use: AVX2 FMA
2019-09-19 17:40:33.788088: I tensorflow/core/platform/profile_utils/cpu_util
s.cc:94] CPU Frequency: 2592000000 Hz
2019-09-19 17:40:33.788858: I tensorflow/compiler/xla/service/service.cc:150]
 XLA service 0x55e5c34a78b0 executing computations on platform Host. Devices:
2019-09-19 17:40:33.788920: I tensorflow/compiler/xla/service/service.cc:158]
StreamExecutor device (0): <undefined>, <undefined>
Saving information to 'deploy.dvc'.

To track the changes with git, run:

        git add train.dvc load_data.dvc deploy.dvc
```

But wait, there's more!

- Track metrics
- Import versioned artefacts from other projects
- Support for S3, GCS, HDFS, ...
- Remote caching
- ...

But does it also cover my use case %s?

- It depends ¯\(ツ)/¯
- Plenty of alternatives: Sacred, MLFlow, Pachyderm, Polyaxon
- See also https://github.com/EthicalML/awesome-production-machine-learning (https://github.com/EthicalML/awesome-production-machine-learning)

Thanks! :-)