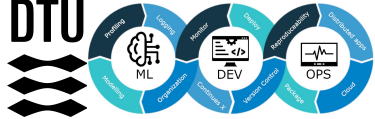


DevOps for Data scientists

Nicki Skafte Detlefsen
nsde@dtu.dk

Who am I

Who am I

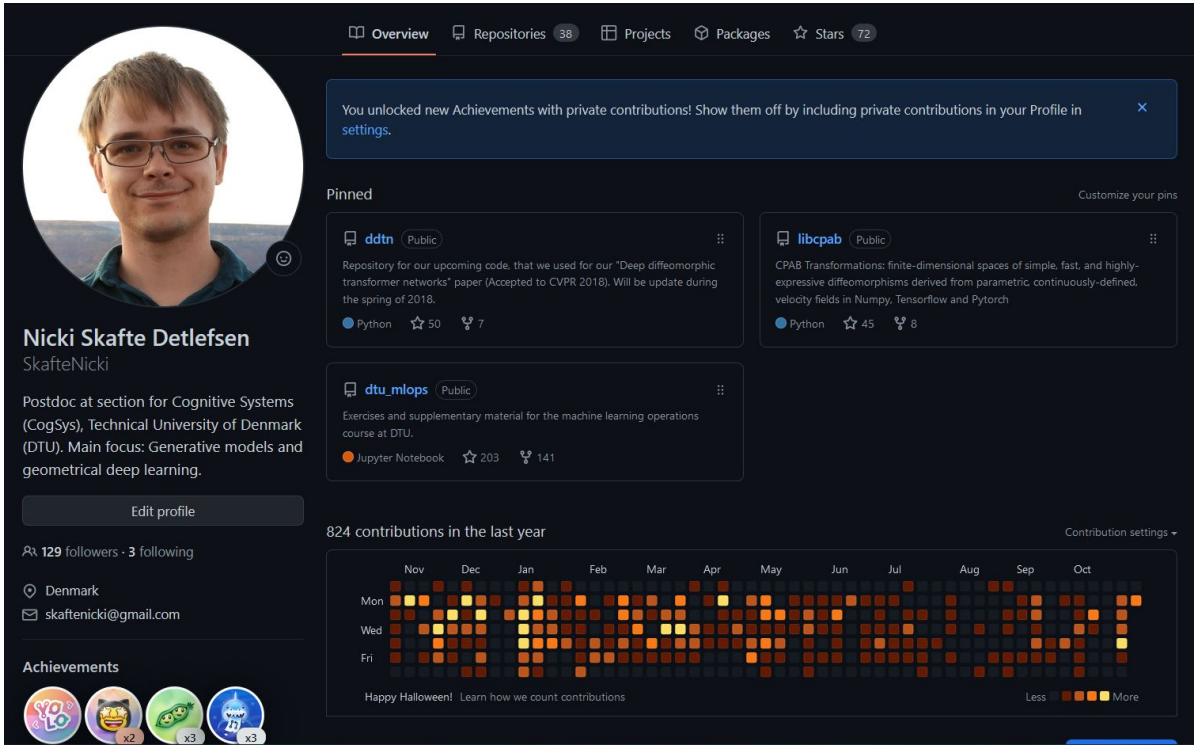
- Bachelor, master, PhD from DTU
- Currently: Postdoc
- Old focus:
  - Inductive biases in deep learning
  - Generative models
  - Geometry aware manifolds
- New focus:
  - MLOps
  - Efficient machine learning

PIONEER CENTRE FOR
ARTIFICIAL INTELLIGENCE

Who am I

- Eager open-source contributor

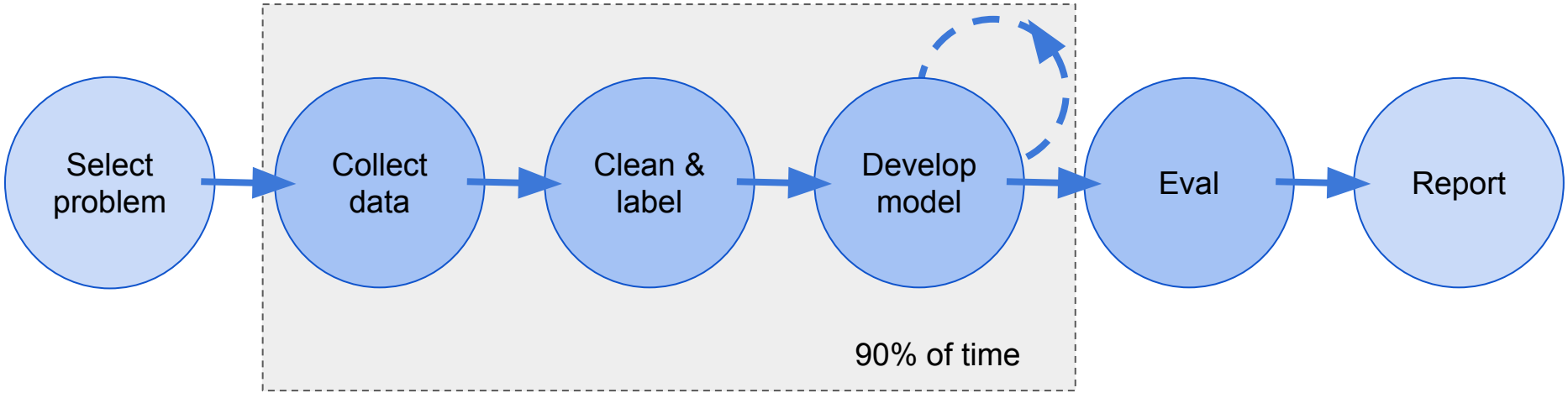
- ML Engineer at https://lightning.ai/



What is DevOps?

Let start where you are now

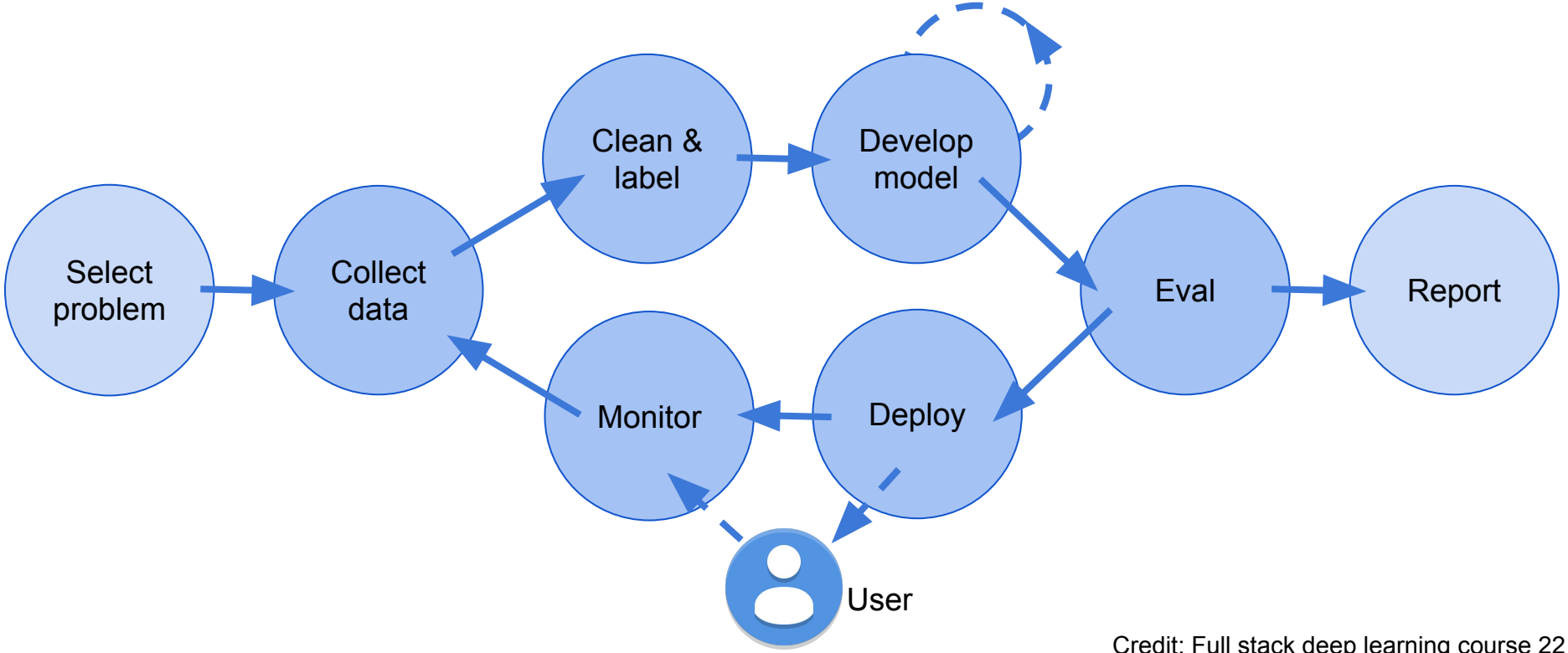
Courses / Projects are linear in nature



Our feedback loop is grades / funding

Credit: Full stack deep learning course 22

Data science in the real world



Credit: Full stack deep learning course 22

Technical debt

Data driven decisions are fantastic

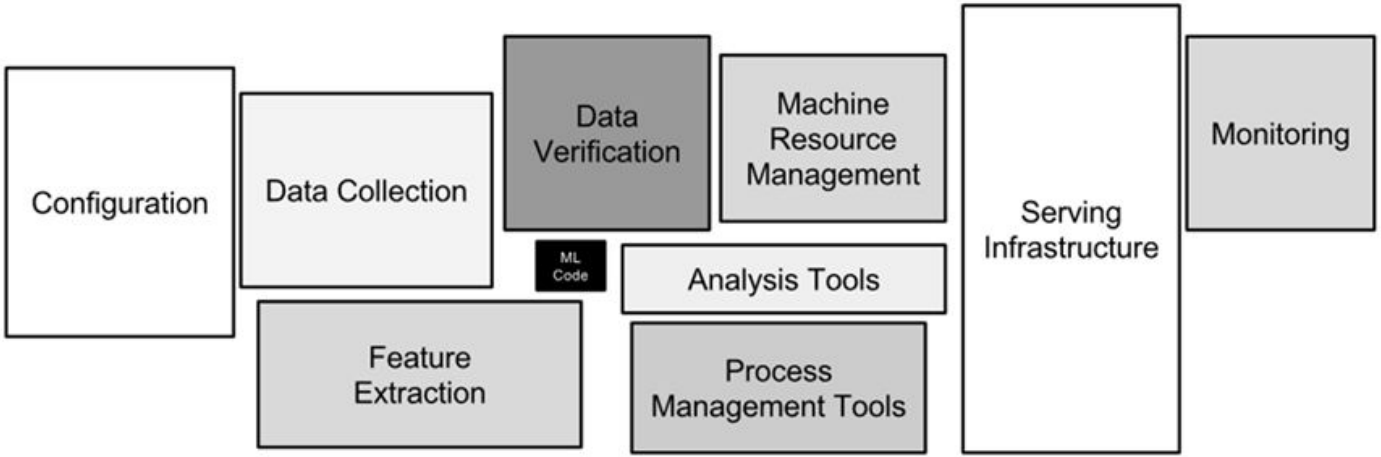
But 🚩

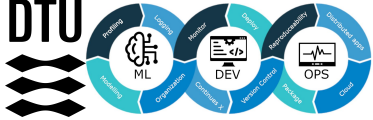
Massive technical debt is incurred if not careful

Hidden Technical Debt in Machine Learning Systems

D. Sculley, Gary Holt, Daniel Golovin, Eugene Davydov, Todd Phillips
{dsculley,gholt,dgg,edavydov,toddphillips}@google.com
Google, Inc.

Dietmar Ebner, Vinay Chaudhary, Michael Young, Jean-François Crespo, Dan Dennison
{ebner,vchaudhary,mwyoung,jfcrespo,dennison}@google.com
Google, Inc.





In a nutshell DevOps is about reducing technical debt

Because we in the real world care about

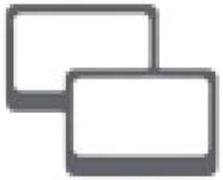
- Maintainability
- Longevity
- Expandability
- Scalability

> **Big ball of Mud**
>
> *A Big Ball of Mud is a haphazardly structured, sprawling, sloppy, duct-tape-and-baling-wire, spaghetti-code jungle. These systems show unmistakable signs of unregulated growth, and repeated, expedient repair. Information is shared promiscuously among distant elements of the system, often to the point where nearly all the important information becomes global or duplicated.*
> *The overall structure of the system may never have been well defined.*
> *If it was, it may have eroded beyond recognition. Programmers with a shred of architectural sensibility shun these quagmires. Only those who are unconcerned about architecture, and, perhaps, are comfortable with the inertia of the day-to-day chore of patching the holes in these failing dikes, are content to work on such systems.*
>
> Brian Foote and Joseph Yoder, Big Ball of Mud. Fourth Conference on Patterns Languages of Programs (PLoP '97/ EuroPLoP '97) Monticello, Illinois, September 1997

We need basic skills/tools to help us secure this

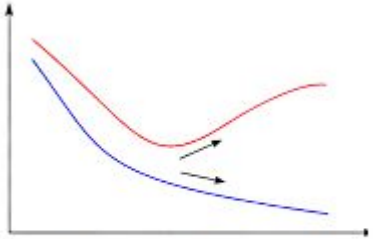
Basic Skills needed for DevOps

Devops in four topics



Virtual Enviroments

Version Control



Experiment Tracking

Code Testing

What is a virtual environment

A virtual environment is tool for keeping **dependencies** for different projects **separated**. It consist of:

1. A python interpreter
2. A folder of dependencies

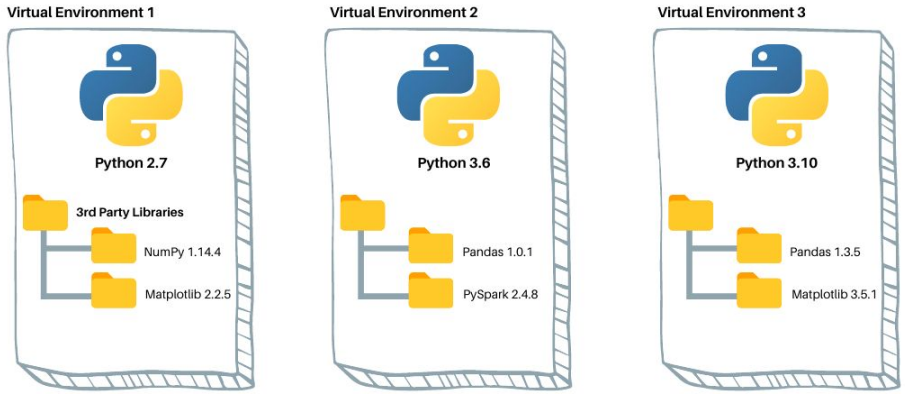


Figure credit: https://www.dataquest.io/blog/a-complete-guide-to-python-virtual-environments/

Virtual Enviroments

Why is virtual environments important?

Do you think the following would run on your computer?

Why?

```
import numpy as np
x = np.array([1,2,3])
y = np.array([4,5,6])
z = np.vstack([x,y], dtype=np.float)
```

Two kind of errors:

- Hard errors: Feature X no longer exist, code fails on use
- Soft errors: Feature X was changed, code runs but produce wrong result

Virtual Enviroments

Figure credit: https://www.dataquest.io/blog/a-complete-guide-to-python-virtual-environments/

Virtual envs in python

Use a package management system

Examples:

- Conda (what I like)
- Pipenv
- venv
- pyenv
- pdb

```
(lightning) C:\Users\nsde\Documents\metrics>conda env list
# conda environments:
#
base                  C:\Users\nsde\Anaconda3
ensemble              C:\Users\nsde\Anaconda3\envs\ensemble
laplace               C:\Users\nsde\Anaconda3\envs\laplace
lightning           * C:\Users\nsde\Anaconda3\envs\lightning
mixerensemble         C:\Users\nsde\Anaconda3\envs\mixerensemble
mlops                 C:\Users\nsde\Anaconda3\envs\mlops
protein               C:\Users\nsde\Anaconda3\envs\protein
pvae                  C:\Users\nsde\Anaconda3\envs\pvae
stochman              C:\Users\nsde\Anaconda3\envs\stochman


(lightning) C:\Users\nsde\Documents\metrics>
```

```
(lightning) C:\Users\nsde\Documents\metrics>conda list
# packages in environment at C:\Users\nsde\Anaconda3\envs\lightning:
#
# Name                    Version                   Build  Channel
absl-py                   1.2.0                     pypi_0    pypi
aiohttp                   3.8.3                     pypi_0    pypi
aiosignal                 1.2.0                     pypi_0    pypi
alabaster                 0.7.12                    pypi_0    pypi
asttokens                 2.0.5              pyhd3eb1b0_0
async-timeout             4.0.2                     pypi_0    pypi
atomicwrites              1.4.1                     pypi_0    pypi
attrs                     22.1.0                    pypi_0    pypi
babel                     2.10.3                    pypi_0    pypi
backcall                  0.2.0              pyhd3eb1b0_0
beautifulsoup4            4.11.1                    pypi_0    pypi
black                     22.8.0                    pypi_0    pypi
blas                      2.116                       mkl    conda-forge
blas-devel                3.9.0                16_win64_mkl    conda-forge
bleach                    5.0.1                     pypi_0    pypi
brotlipy                  0.7.0           py38h294d835_1004    conda-forge
build                     0.8.0                     pypi_0    pypi
ca-certificates           2022.07.19         haa95532_0
cachetools                5.2.0                     pypi_0    pypi
certifi                   2022.9.14          py38haa95532_0
cffi                      1.15.1             py38hd8c33c5_0    conda-forge
cfgv                      3.3.1                     pypi_0    pypi
charset-normalizer        2.1.1              pyhd8ed1ab_0    conda-forge
check-manifest            0.48                      pypi_0    pypi
click                     8.1.3                     pypi_0    pypi
cloudpickle               2.2.0                     pypi_0    pypi
colorama                  0.4.5              py38haa95532_0
commonmark                0.9.1                     pypi_0    pypi
contourpy                 1.0.5                     pypi_0    pypi
coverage                  6.4.4                     pypi_0    pypi
cryptography              37.0.4             py38hb7941b4_0    conda-forge
cudatoolkit               11.6.0               hc0ea762_10    conda-forge
cycler                    0.11.0                    pypi_0    pypi
decorator                 5.1.1              pyhd3eb1b0_0
defusedxml                0.7.1                     pypi_0    pypi
distlib                   0.3.6                     pypi_0    pypi
docutils                  0.17.1                    pypi_0    pypi
dython                    0.7.2                     pypi_0    pypi
```

Virtual Enviroments

How do we inform people of requirements?

When you call

pip install package

What happens is*

- Download package
- Calls *pip install pyproject.toml* or calls *python setup.py install*
  - Both needs a *requirements.txt*

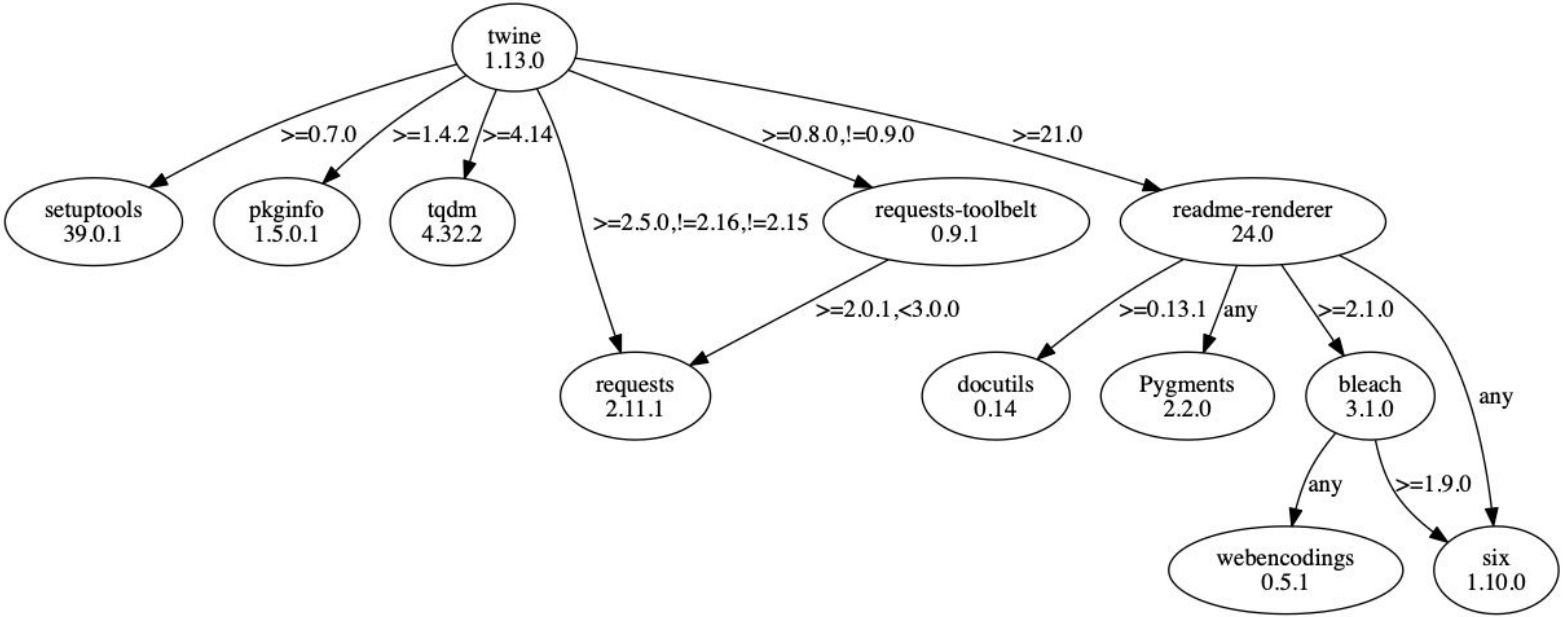
*in most cases

requirements.txt

```
1  cryptography==1.1
2  brotlipy==1.1.1
3  idna==3.0.3
4  pyOpenSSL==2.0.4
5  PySocks==4.2.5
6  glob2==4.2.3
7  py==1.6.7
8  pytest==5.6.7
9  attrs==8.1.1
10 selenium==1.0.2
11 pandas==1.2.4
```

What is the job of pip?

Dependency resolution: Finding versions of all dependencies that works together



Code breakout

Virtual Enviroments

What is version control

For code development in a **team** we need

- A location/method for centrally storing files
- Keeping a record of changes
- Who did what and when in the system

Question:

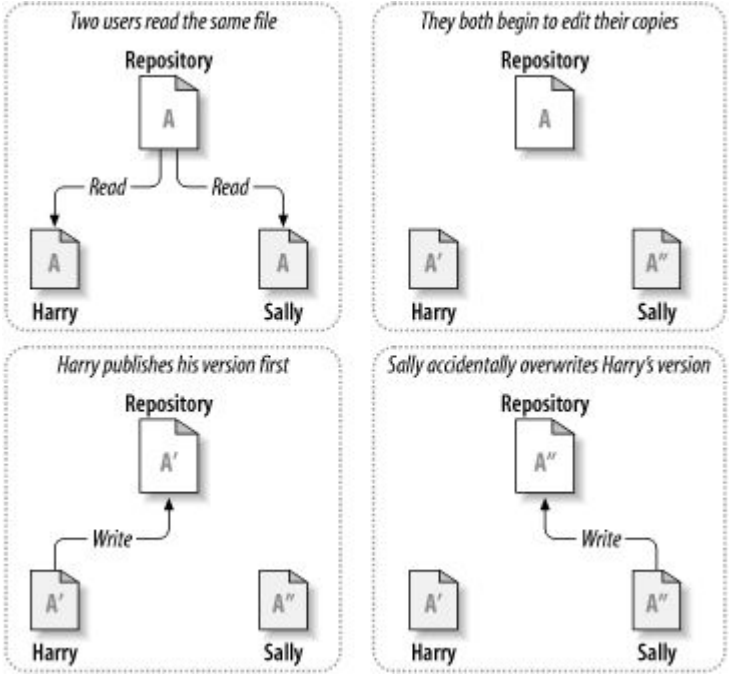
What are you currently using to share coc ?

Would that scale to 10 persons? 100? 1000?

Version Control

What is the problems version control solves?

1. Local + central code communication
2. No accidental overwrites
3. Infinite scaling of team members
4. Possible to roll back



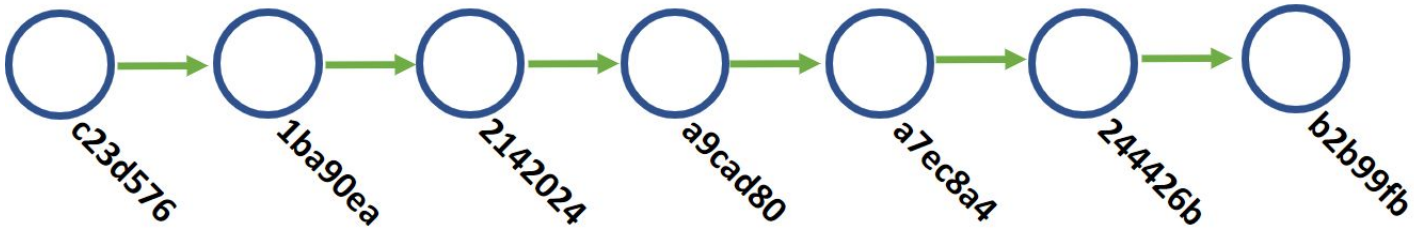
Version Control

In simple terms, VC is a graph of hashes

When you have work you are satisfied with you **commit** it to the graph

Everyone have access to the graph, and can *only add to the end of the graph

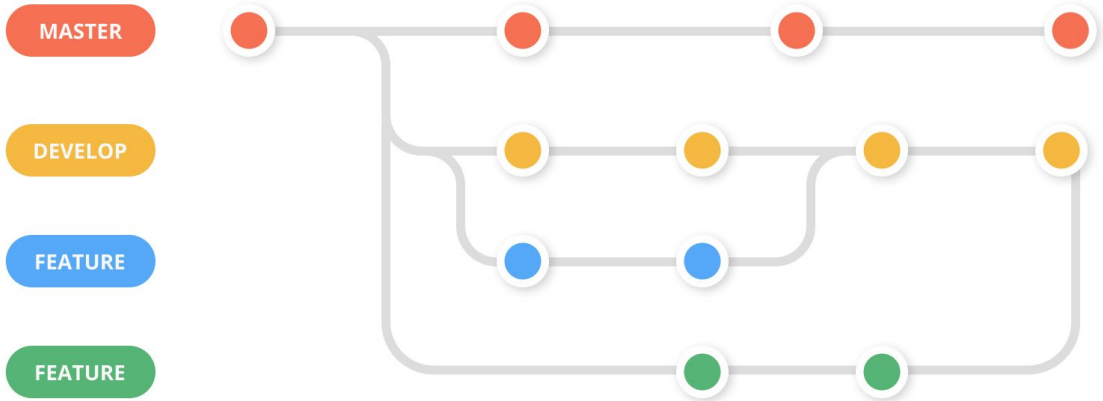
You therefore need to be in sync with the graph to commit.



Version Control

Parallel work using branches

Branching and merging allows multiple users working together



Version Control

Technologies for version control

Biggest version control system is git

Biggest repository system is github

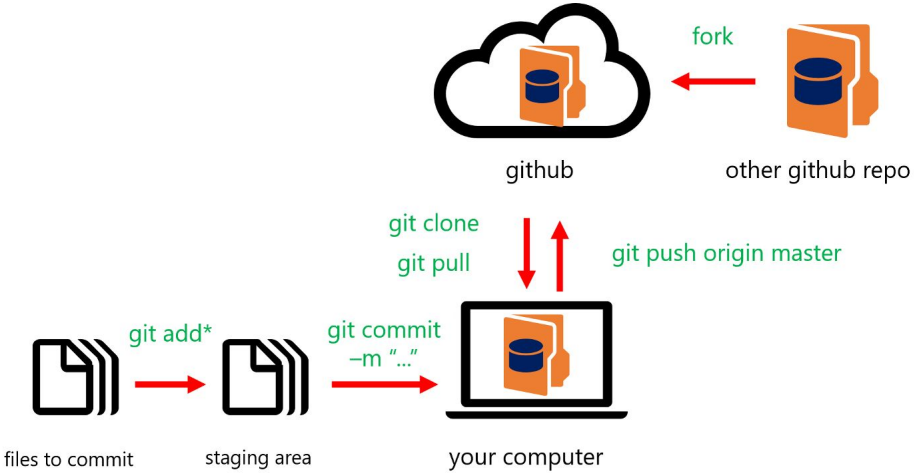


Figure credit:
https://www.analyticsvidhya.com/blog/2021/09/git-and-github-tutorial-for-beginners/
https://medium.com/swlh/an-introduction-to-git-and-github-22ecb4cb1256

Code breakout

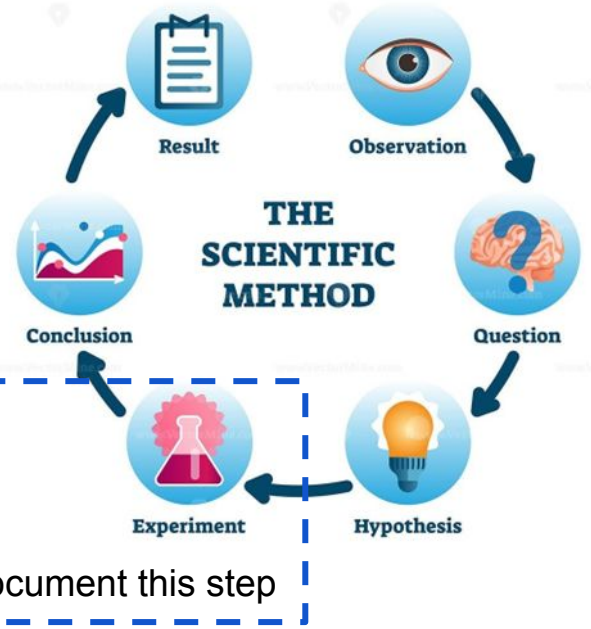
What is experiment tracking

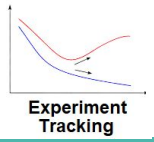
The core part of doing DS/ML is making experiments

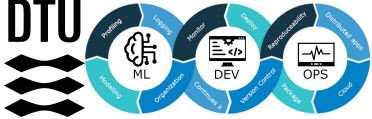
Experiments are the way to test your hypothesis

Question:

What is important in data science to log during an experiment?







What should we log in my option

- Metrics related to the performance of the experiment
- Configuration of experiment (=hyperparameters)
- Virtual environment used
- Compute setup
- Code used (=what commit we ran from)

A reason to be careful about logging

Re-Implementation of 255 paper. Hypothesis testing on what “paper features” have an effect on reproducibility.

A Step Toward Quantifying Independently Reproducible Machine Learning Research

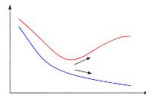
Edward Raff
Booz Allen Hamilton
raff_edward@bah.com
University of Maryland, Baltimore County
raff.edward@umbc.edu

Abstract

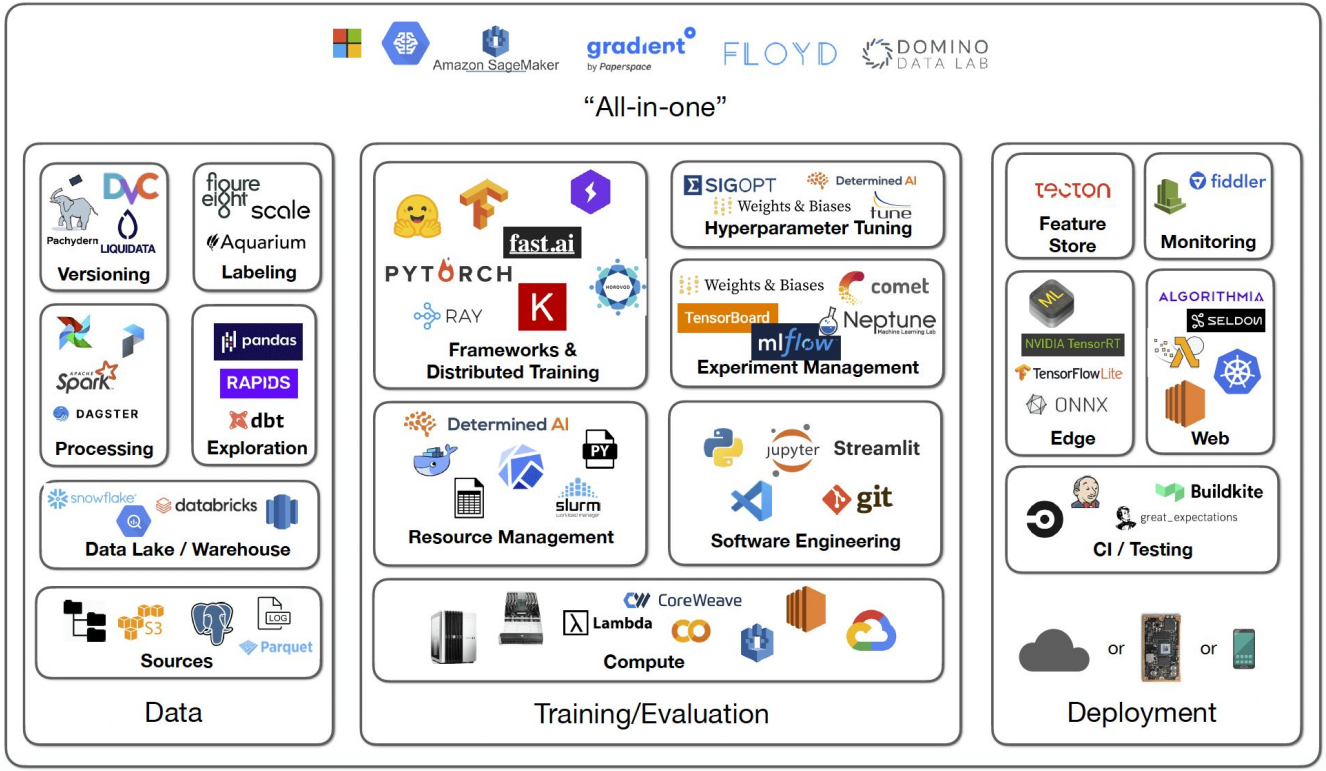
What makes a paper independently reproducible? Debates on reproducibility center around intuition or assumptions but lack empirical results. Our field focuses on releasing code, which is important, but is not sufficient for determining reproducibility. We take the first step toward a quantifiable answer by manually attempting to implement 255 papers published from 1984 until 2017, recording features of each paper, and performing statistical analysis of the results. For each paper, we did not look at the authors code, if released, in order to prevent bias toward discrepancies between code and paper.

Table 1: Significance test of which paper properties impact reproducibility. Results significant at $\alpha \leq 0.05$ marked with“*”.

Feature	p-value
Year Published	0.964
Year First Attempted	0.674
Venue Type	0.631
Rigor vs Empirical*	1.55×10^{-9}
Has Appendix	0.330
Looks Intimidating	0.829
Readability*	9.68×10^{-25}
Algorithm Difficulty*	2.94×10^{-5}
Pseudo Code*	2.31×10^{-4}
Primary Topic*	7.039×10^{-4}
Exemplar Problem	0.720
Compute Specified	0.257
Hyperparameters Specified*	8.45×10^{-6}
Compute Needed*	8.75×10^{-5}
Authors Reply*	6.01×10^{-8}
Code Available	0.213
Pages	0.364
Publication Venue	0.342
Number of References	0.740
Number Equations*	0.004
Number Proofs	0.130
Number Tables*	0.010
Number Graphs/Plots	0.139
Number Other Figures	0.217
Conceptualization Figures	0.365
Number of Authors	0.497



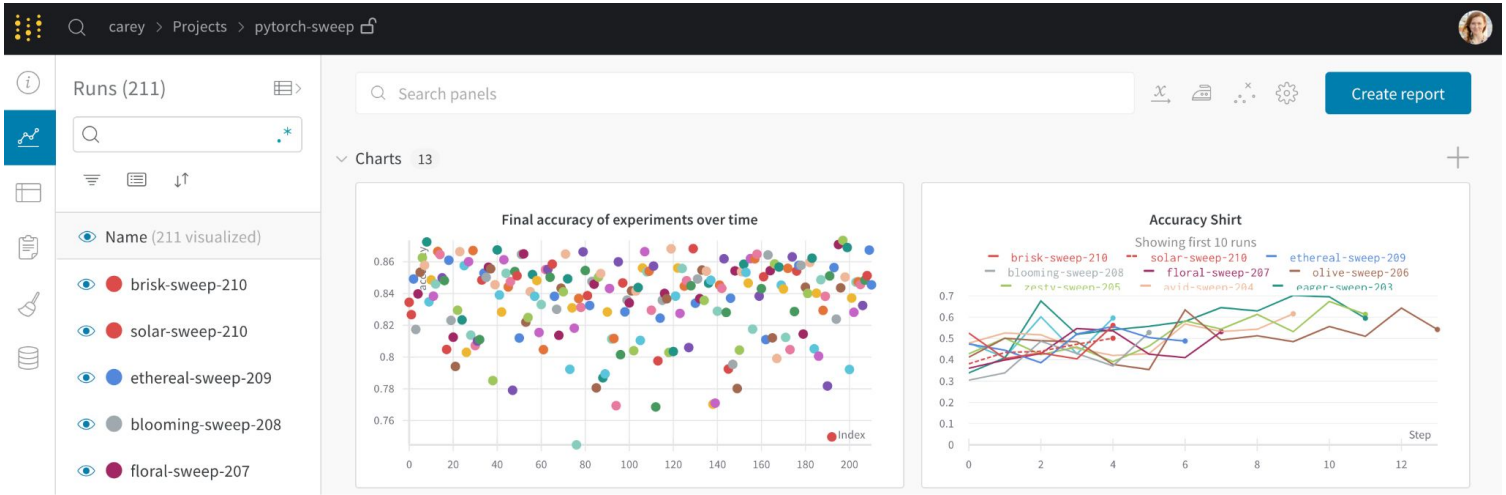
Todays tool landscape is filled with tools for logging

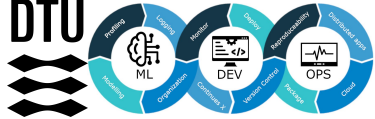


Todays tool landscape is filled with tools for logging

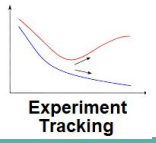
A logging tool should be

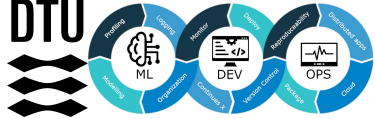
- simple to use
- store various kinds of data





Code breakout





What is code testing

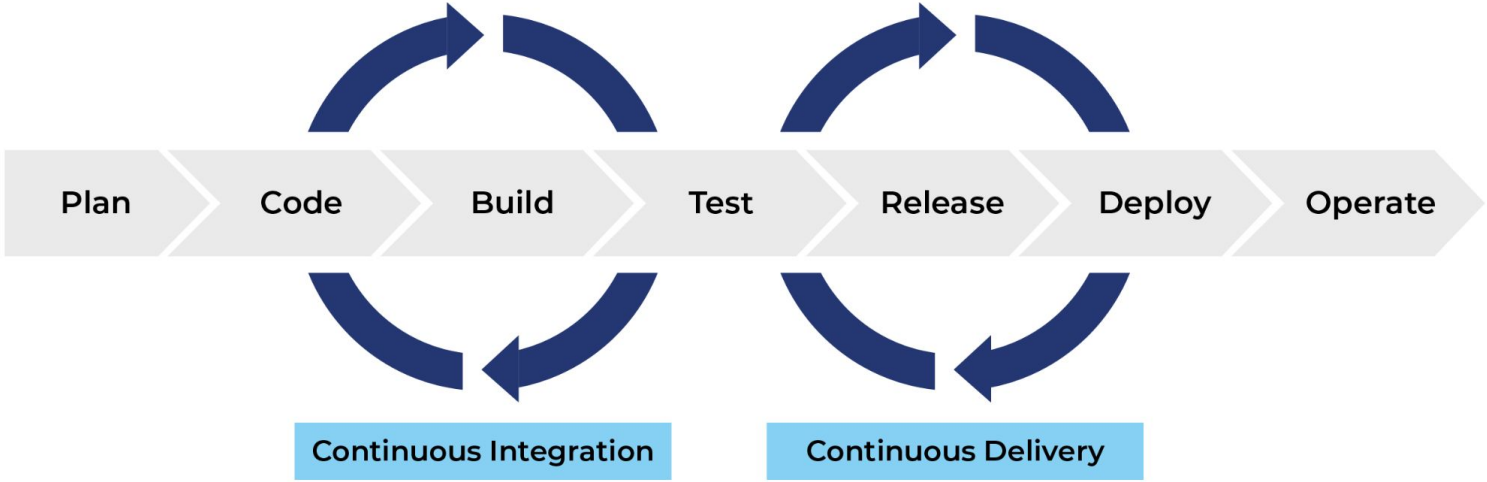
It is the process used to identify the **correctness**, **completeness** and **quality** of develop software

Objectives

- Uncover errors
- Software matches requirements
- Validate quality

Why do we need code testing?

Because fixing bugs in production is expensive!



Code Testing

Figure credit: http://testbone.com/blog/continuous-integration/

Unit tests are the shit

Test are the cornerstones of a good pipeline

- In particular, *unit tests* are important.
- A single unittest, tests a small part of your code
- By testing code in small pieces, bugs are easier to find

Other types of testing: integration, system

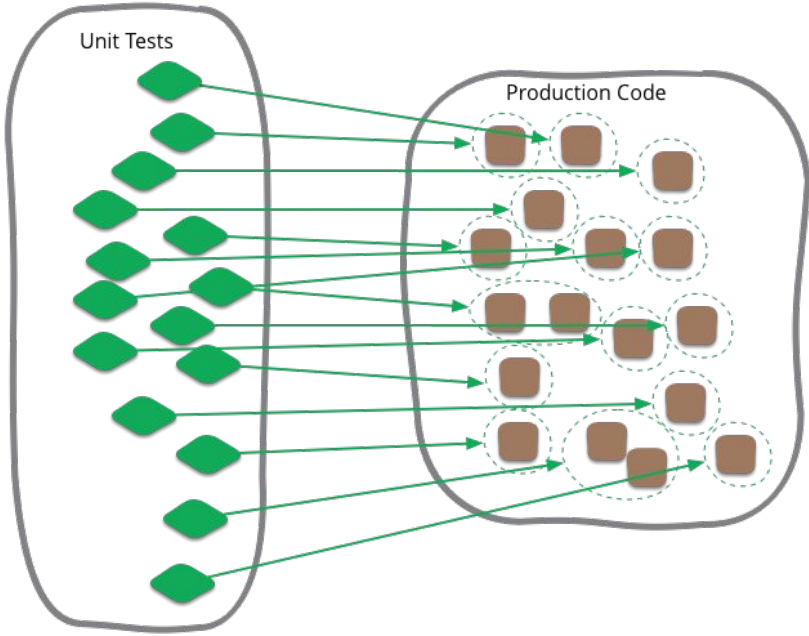


Figure credit: https://martinfowler.com/bliki/UnitTest.html

Code Testing

Writing test in python

In python, we recommend using the **pytest** framework.

Test are simple functions that start with *test_* and uses *assert*

```python
import numpy as np

# functions.py
def mean_squared_error(preds: np.ndarray, target: np.ndarray):
    return np.sum(np.power(np.abs(preds - target), 2.0))

# test_functions.py
def test_mean_squared_error():
    preds = np.zeros(10,)
    target = np.zeros(10,)
    assert mean_squared_error(preds, target) == 0
```

Code Testing

Test can be as simple or complicated as needed

Test can be simple...

```python
def test_warning_on_nan(tmpdir):
    preds = torch.randint(3, size=(20, ))
    target = torch.randint(3, size=(20, ))

    with pytest.warns(
        UserWarning,
        match='.* nan values found in confusion matrix have been replaced with zeros.',
    ):
        confusion_matrix(preds, target, num_classes=5, normalize='true')
```

Code
Testing

Test can be as simple or complicated as needed

Test can be simple...

```python
def test_warning_on_nan(tmpdir):
    preds = torch.randint(3, size=(20, ))
    target = torch.randint(3, size=(20, ))

    with pytest.warns(
        UserWarning,
        match='.* nan values found in confusion matrix have been replaced with zeros.',
    ):
        confusion_matrix(preds, target, num_classes=5, normalize='true')
```

Or complicated

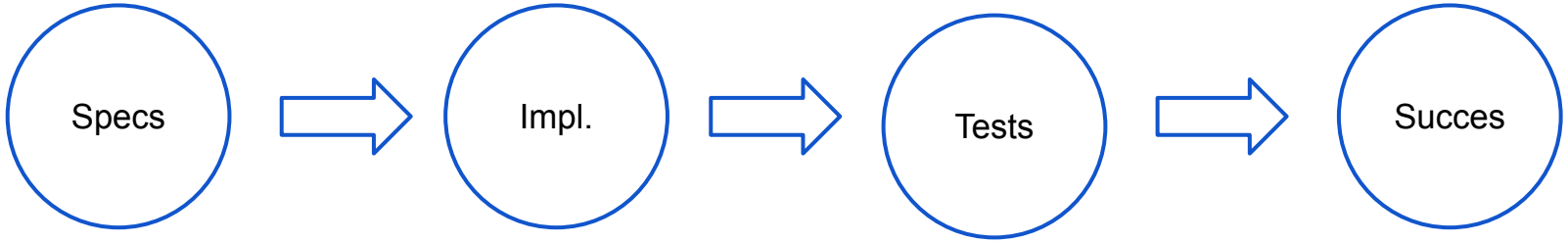
```python
@pytest.mark.parametrize("normalize", ['true', 'pred', 'all', None])
@pytest.mark.parametrize(
    "preds, target, sk_metric, num_classes, multilabel",
    [(_input_binary_prob.preds, _input_binary_prob.target, _sk_cm_binary_prob, 2, False),
     (_input_binary_logits.preds, _input_binary_logits.target, _sk_cm_binary_prob, 2, False),
     (_input_binary.preds, _input_binary.target, _sk_cm_binary, 2, False),
     (_input_mlb_prob.preds, _input_mlb_prob.target, _sk_cm_multilabel_prob, NUM_CLASSES, True),
     (_input_mlb_logits.preds, _input_mlb_logits.target, _sk_cm_multilabel_prob, NUM_CLASSES, True),
     (_input_mlb.preds, _input_mlb.target, _sk_cm_multilabel, NUM_CLASSES, True),
     (_input_mcls_prob.preds, _input_mcls_prob.target, _sk_cm_multiclass_prob, NUM_CLASSES, False),
     (_input_mcls_logits.preds, _input_mcls_logits.target, _sk_cm_multiclass_prob, NUM_CLASSES, False),
     (_input_mcls.preds, _input_mcls.target, _sk_cm_multiclass, NUM_CLASSES, False),
     (_input_mdmc_prob.preds, _input_mdmc_prob.target, _sk_cm_multidim_multiclass_prob, NUM_CLASSES, False),
     (_input_mdmc.preds, _input_mdmc.target, _sk_cm_multidim_multiclass, NUM_CLASSES, False)]
)
class TestConfusionMatrix(MetricTester):

    @pytest.mark.parametrize("ddp", [True, False])
    @pytest.mark.parametrize("dist_sync_on_step", [True, False])
    def test_confusion_matrix(
        self, normalize, preds, target, sk_metric, num_classes, multilabel, ddp, dist_sync_on_step
    ):
        self.run_class_metric_test(
            ddp=ddp,
            preds=preds,
            target=target,
            metric_class=ConfusionMatrix,
            sk_metric=partial(sk_metric, normalize=normalize),
            dist_sync_on_step=dist_sync_on_step,
            metric_args={
                "num_classes": num_classes,
                "threshold": THRESHOLD,
                "normalize": normalize,
                "multilabel": multilabel
            }
        )
```

Parametrize is powerful:
4 x 11 x 2 x 2 = 176 tests!

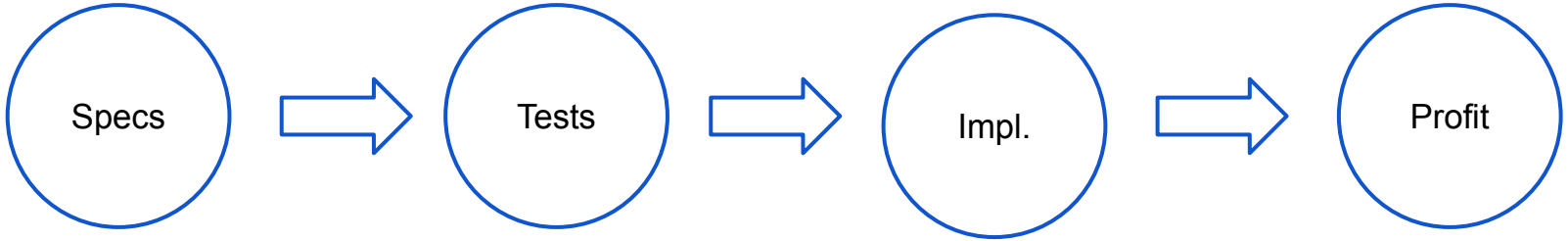
Code
Testing

Test driven development

Change your mindset from this



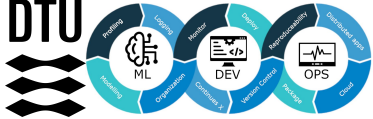
to this



Code Testing

Code breakout

Code
Testing



In summary

DevOps provides methods for

- Virtual environments
- Version control
- Experiment tracking
- Code testing

such that experiments become reproducible and technical dept is reduced.

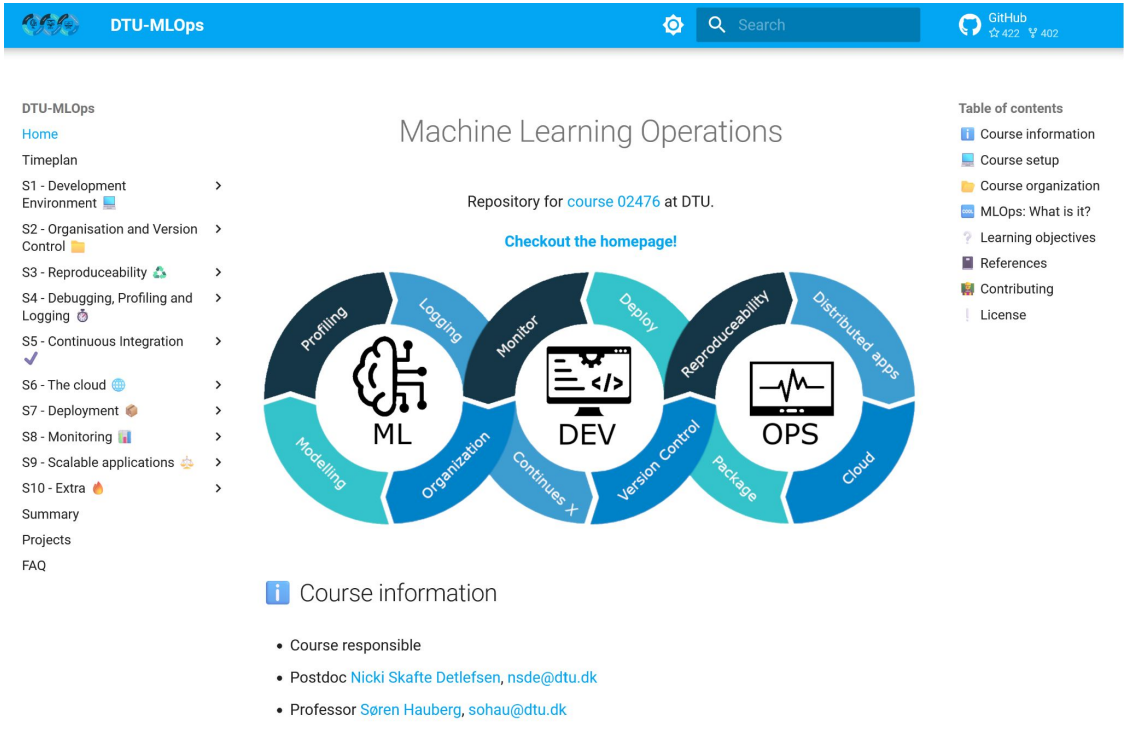
MLOps at DTU

https://kurser.dtu.dk/course/02476

https://github.com/SkafteNicki/dtu_mlops

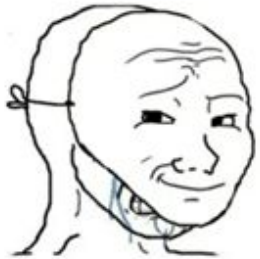
3 weeks in January

5 ECTS

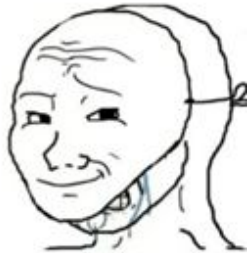


Thanks for your attention

Programmers



This code is unreadable and your dataset is flawed. No one will be able to reproduce your results!



It's not my fault the legacy environment is messed up! We still have 97.3% unit test coverage.

Scientist

This code is unreadable and your dataset is flawed. No one will be able to reproduce your results!

I know :)

Feel free to contact me
nsde@dtu.dk
skaftenicki@gmail.com,
https://www.linkedin.com/in/nicki-skafte-detlefsen/