02476 Machine Learning Operations

Nicki Skafte Detlefsen

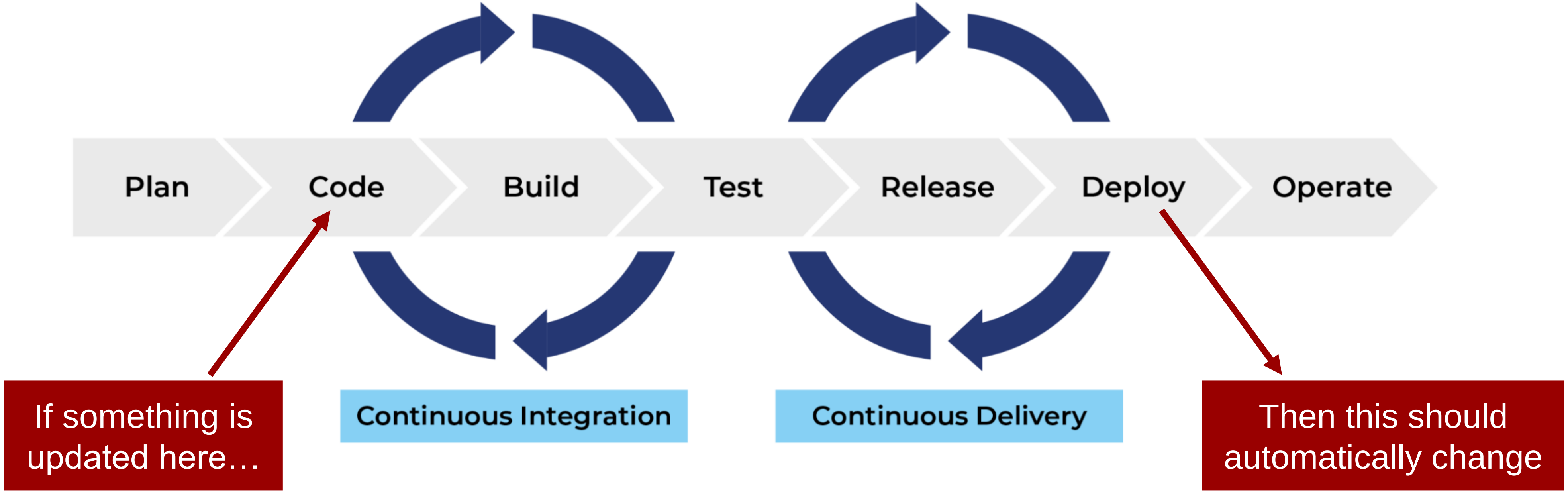
Continuous Integration

Why you should care about today

3 years ago, the day before this lecture, the internet went down for a couple of hours because someone f..ked up their continues integration at Fastly

Continues X

Term refers to a set of software practices for automating tedious tasks and make sure changed in a pipeline are continuously propagated through the pipeline



If something is updated here…

Then this should automatically change

CI

Continues Integration

Core tasks:
💡 How to automatically secure that code does not break during development?

💻App independent concept

CD

Continues Deployment

Core tasks:
💡 How to get your code/application to the user automatically? + monitor life cycle

💻App dependent concept

CML

Continues Machine Learning

Core tasks:
💡 How to automatically retrain machine learning models when data and code changes?

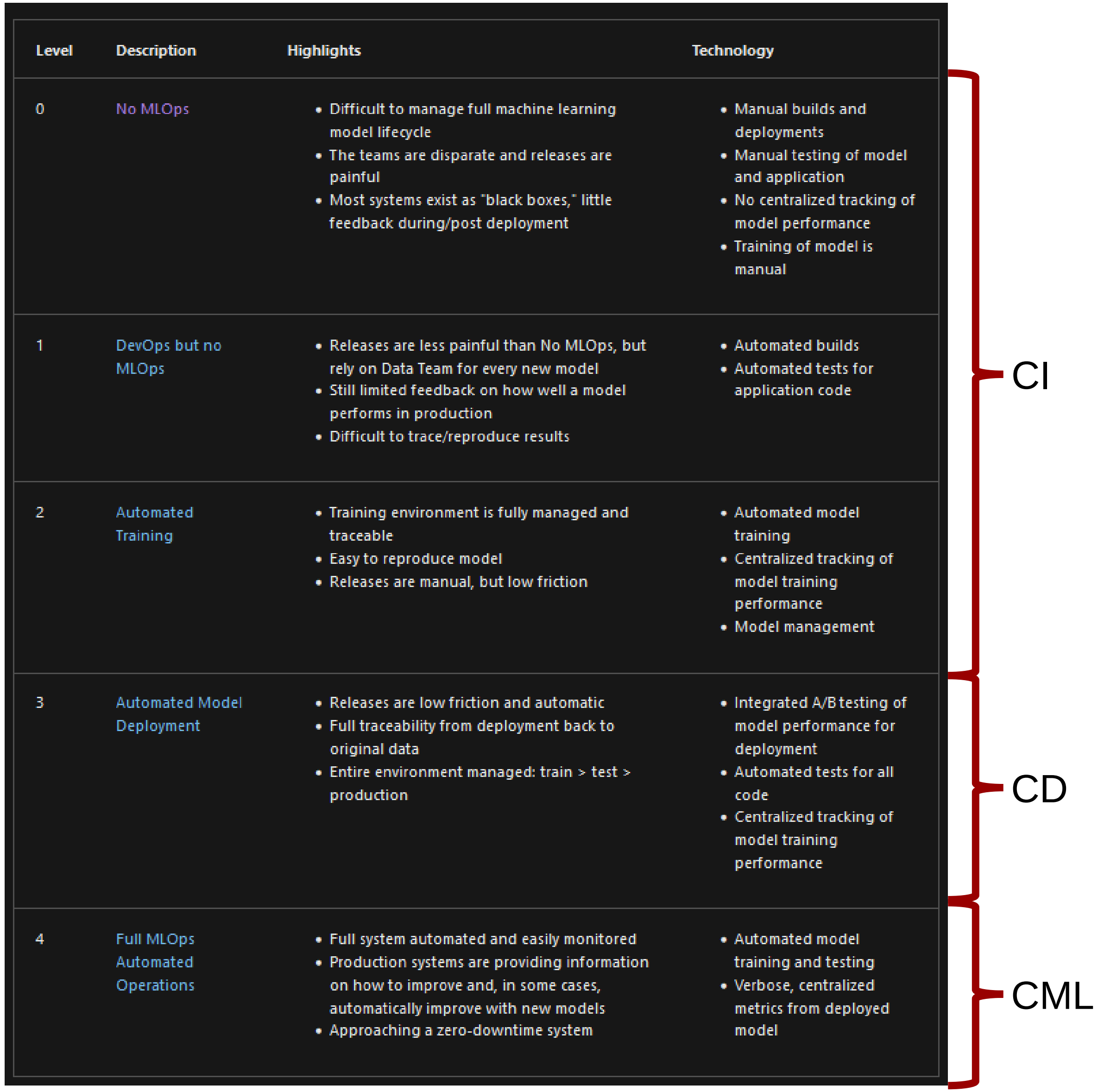
💻Specific to ML applications

MLOps levels

The **Maturity model** overall describes the DevOps practices to run a successful MLOps environment.

Intended to identify gaps in an existing organization's attempt to implement such an environment.

- Estimate the scope of the work for new engagements.
- Establish realistic success criteria.
- Identify deliverables you'll hand over at the conclusion of the engagement.



Level	Description	Highlights	Technology
0	No MLOps	• Difficult to manage full machine learning model lifecycle • The teams are disparate and releases are painful • Most systems exist as "black boxes," little feedback during/post deployment	• Manual builds and deployments • Manual testing of model and application • No centralized tracking of model performance • Training of model is manual
1	DevOps but no MLOps	• Releases are less painful than No MLOps, but rely on Data Team for every new model • Still limited feedback on how well a model performs in production • Difficult to trace/reproduce results	• Automated builds • Automated tests for application code
2	Automated Training	• Training environment is fully managed and traceable • Easy to reproduce model • Releases are manual, but low friction	• Automated model training • Centralized tracking of model training performance • Model management
3	Automated Model Deployment	• Releases are low friction and automatic • Full traceability from deployment back to original data • Entire environment managed: train > test > production	• Integrated A/B testing of model performance for deployment • Automated tests for all code • Centralized tracking of model training performance
4	Full MLOps Automated Operations	• Full system automated and easily monitored • Production systems are providing information on how to improve and, in some cases, automatically improve with new models • Approaching a zero-downtime system	• Automated model training and testing • Verbose, centralized metrics from deployed model

This lecture: continues integration

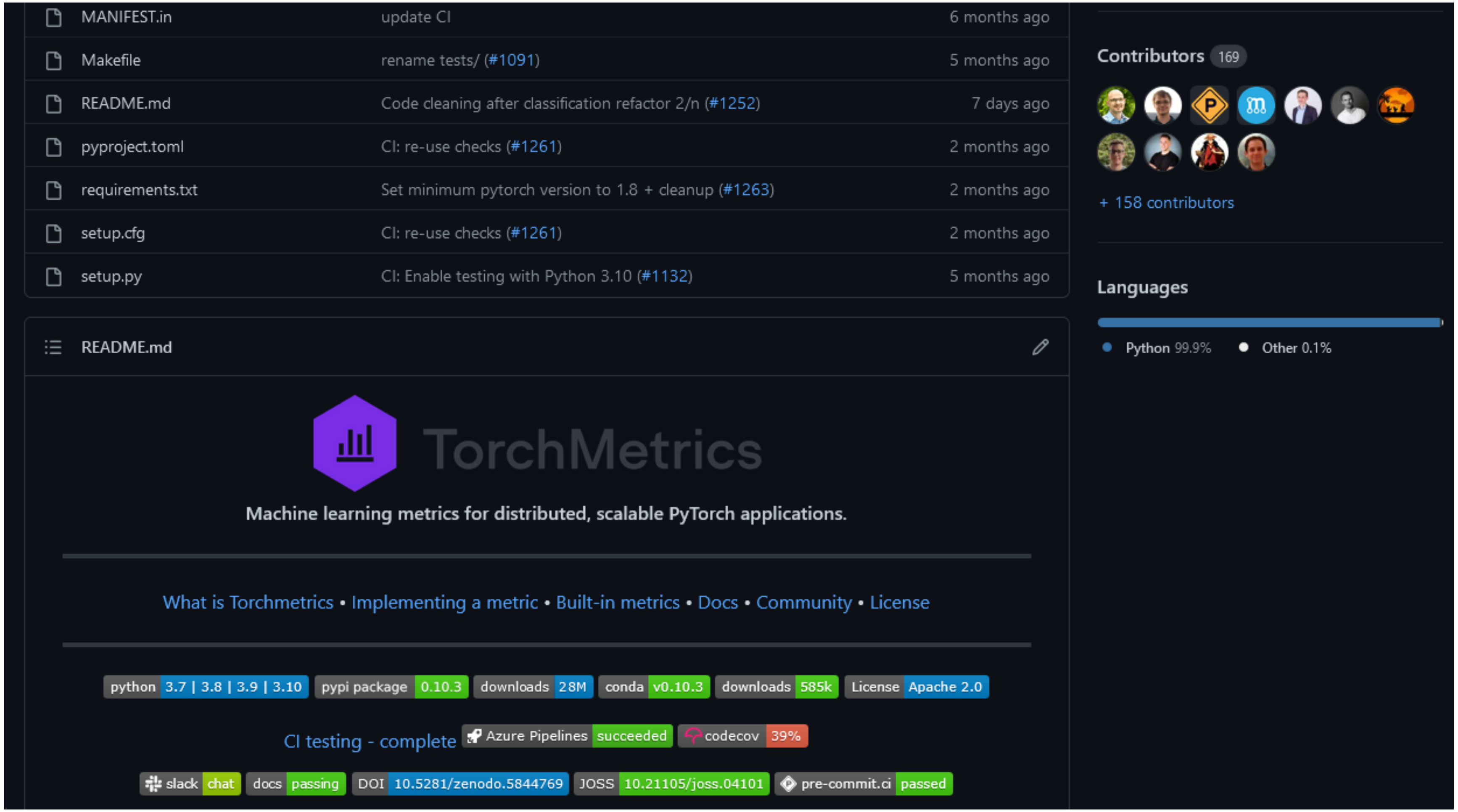
Core task:

🔥 How to automatically secure that code does not break during development? 🔥

3 steps to do this:

- Use version control
  - Frequently committing code to a shared repository
- Write (unit)test for your code
  - Should capture unwanted bugs in your code
- Automate build + testing
  - Automatically run test so code cannot be merged without working

A small case study for continuous integration



MANIFEST.in	update CI	6 months ago
Makefile	rename tests/ (#1091)	5 months ago
README.md	Code cleaning after classification refactor 2/n (#1252)	7 days ago
pyproject.toml	CI: re-use checks (#1261)	2 months ago
requirements.txt	Set minimum pytorch version to 1.8 + cleanup (#1263)	2 months ago
setup.cfg	CI: re-use checks (#1261)	2 months ago
setup.py	CI: Enable testing with Python 3.10 (#1132)	5 months ago

Contributors 169

+ 158 contributors

Languages

● Python 99.9% ● Other 0.1%

README.md

TorchMetrics

Machine learning metrics for distributed, scalable PyTorch applications.

What is Torchmetrics • Implementing a metric • Built-in metrics • Docs • Community • License

python 3.7 | 3.8 | 3.9 | 3.10 pypi package 0.10.3 downloads 28M conda v0.10.3 downloads 585k License Apache 2.0

CI testing - complete Azure Pipelines succeeded codecov 39%

slack chat docs passing DOI 10.5281/zenodo.5844769 JOSS 10.21105/joss.04101 pre-commit.ci passed

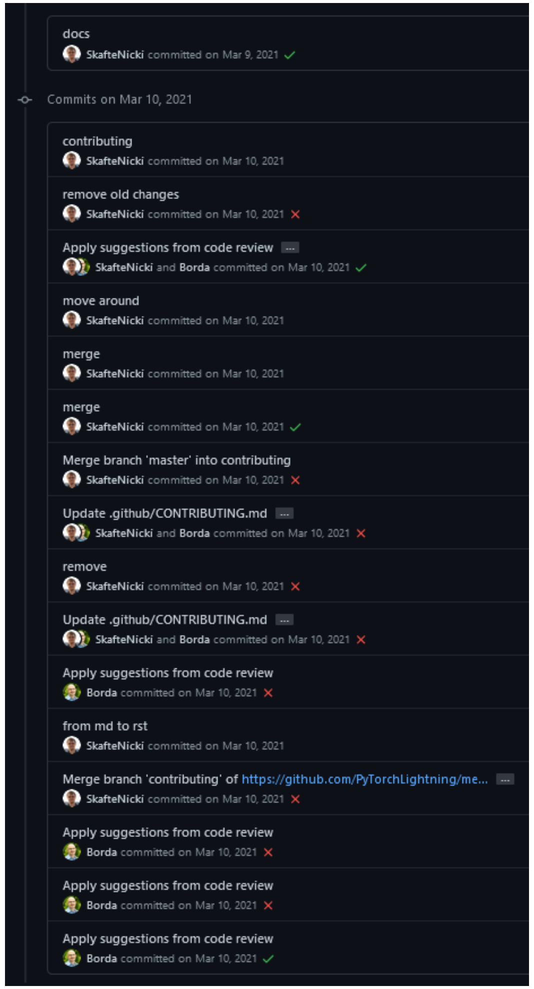
CI step 1: version control

User version control:

- Code changes are tracked
- Branches for parallel work

Commit frequently:

- Catch errors sooner than later
- Revert back easily to when things were working
- Merging can be done automatically



docs
SkafteNicki committed on Mar 9, 2021 ✓

Commits on Mar 10, 2021

contributing
SkafteNicki committed on Mar 10, 2021

remove old changes
SkafteNicki committed on Mar 10, 2021 ✗

Apply suggestions from code review …
SkafteNicki and Borda committed on Mar 10, 2021 ✓

move around
SkafteNicki committed on Mar 10, 2021

merge
SkafteNicki committed on Mar 10, 2021

merge
SkafteNicki committed on Mar 10, 2021 ✓

Merge branch 'master' into contributing
SkafteNicki committed on Mar 10, 2021 ✗

Update .github/CONTRIBUTING.md …
SkafteNicki and Borda committed on Mar 10, 2021 ✗

remove
SkafteNicki committed on Mar 10, 2021 ✗

Update .github/CONTRIBUTING.md …
SkafteNicki and Borda committed on Mar 10, 2021 ✗

Apply suggestions from code review
Borda committed on Mar 10, 2021 ✗

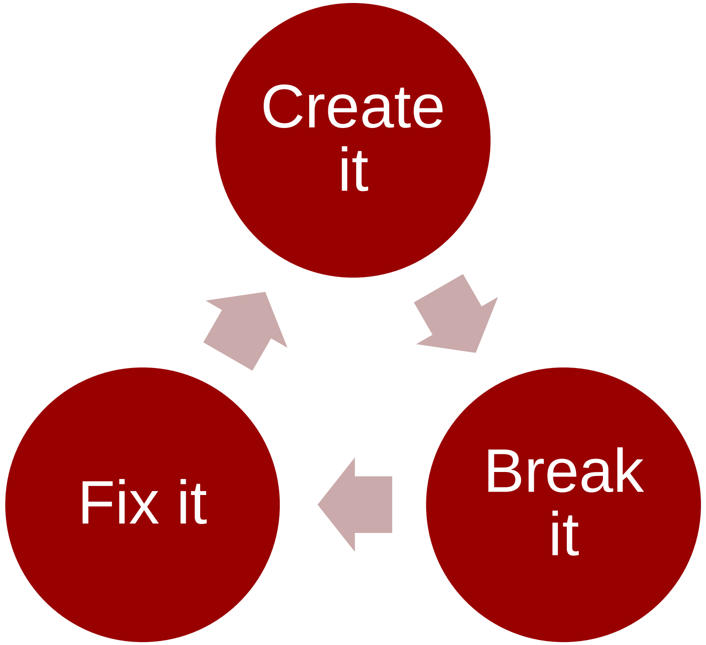
from md to rst
SkafteNicki committed on Mar 10, 2021

Merge branch 'contributing' of https://github.com/PyTorchLightning/me... …
SkafteNicki committed on Mar 10, 2021 ✗

Apply suggestions from code review
Borda committed on Mar 10, 2021 ✗

Apply suggestions from code review
Borda committed on Mar 10, 2021 ✗

Apply suggestions from code review
Borda committed on Mar 10, 2021 ✓



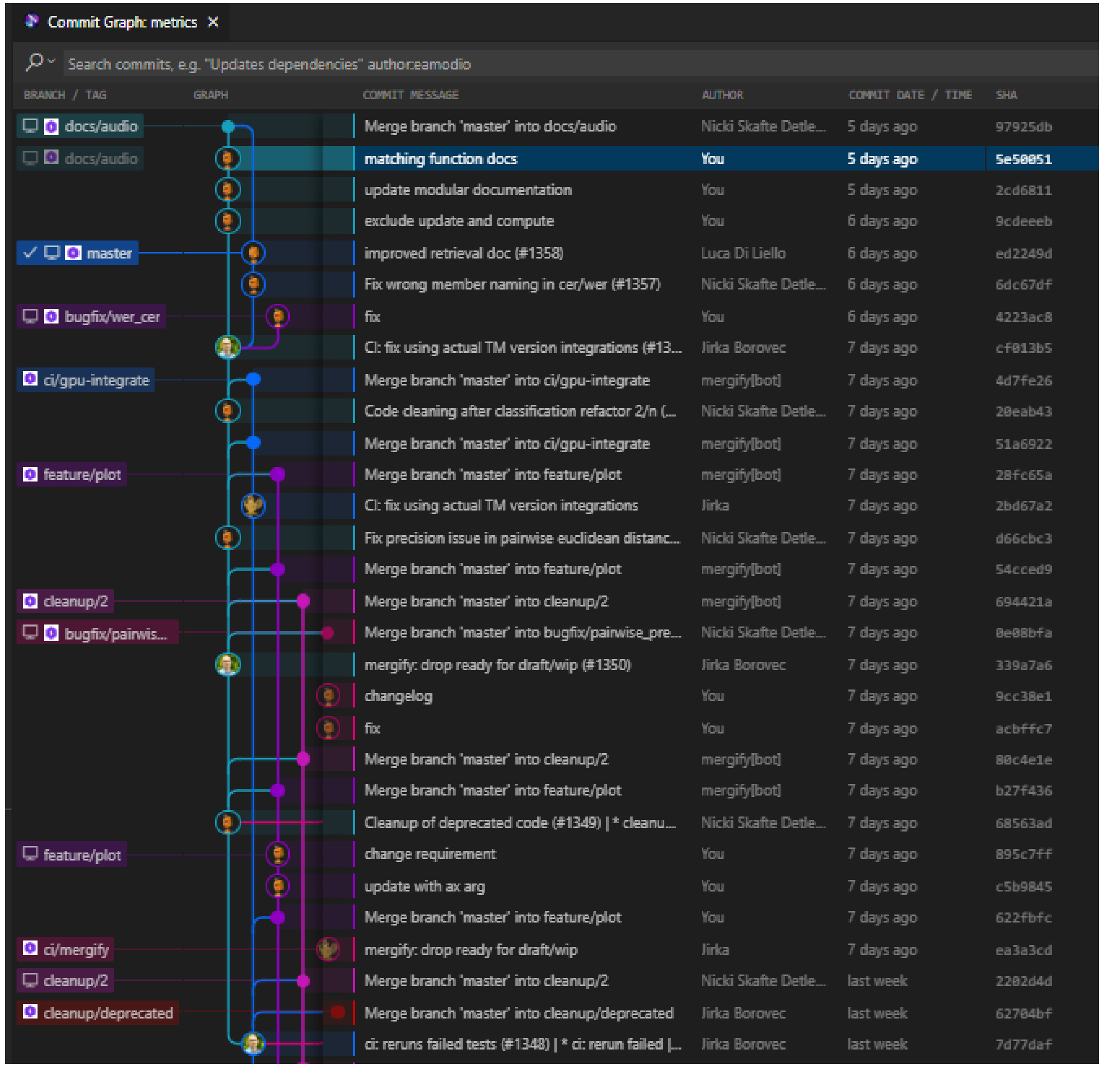
CI step 1: Use branches

Parallel workflow

Experimental features changes are kept away from master/main

Recommend extensions for VS code:

- 💡 Gitlens or GitGraph
- 💡 Github PR and issues



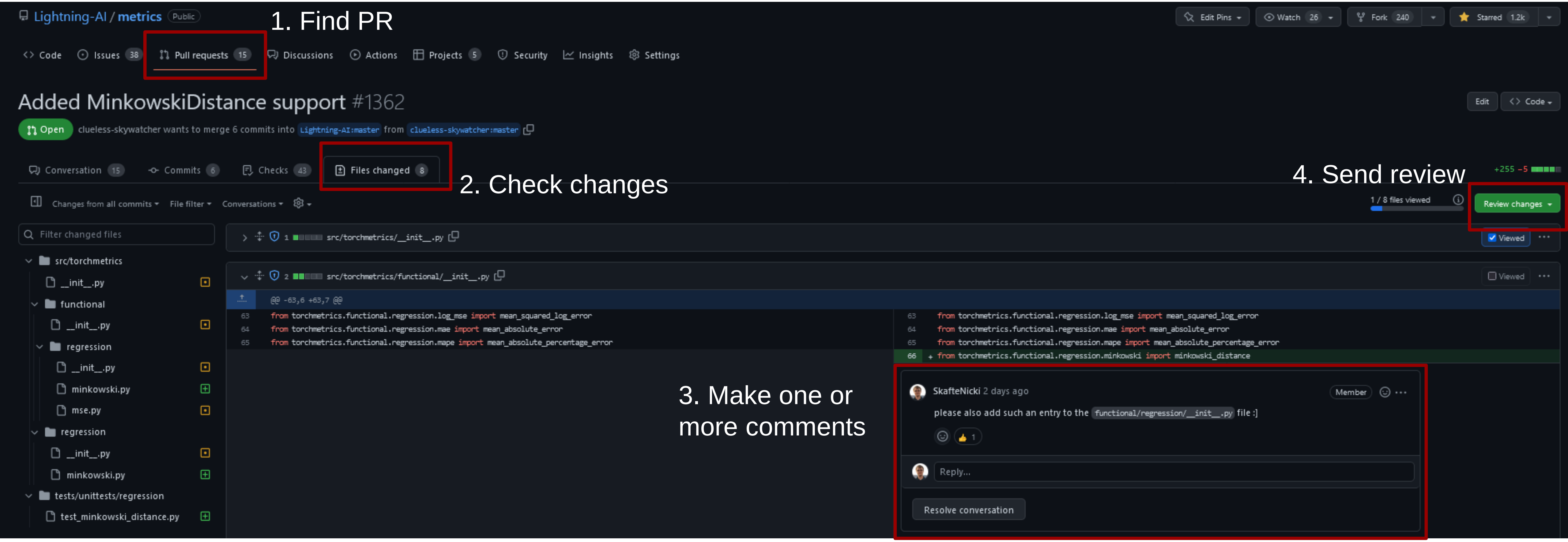
Commit Graph: metrics ✕

Search commits, e.g. "Updates dependencies" author:eamodio

BRANCH / TAG	GRAPH	COMMIT MESSAGE	AUTHOR	COMMIT DATE / TIME	SHA
docs/audio		Merge branch 'master' into docs/audio	Nicki Skafte Detle...	5 days ago	97925db
docs/audio		matching function docs	You	5 days ago	5e50051
		update modular documentation	You	5 days ago	2cd6811
		exclude update and compute	You	6 days ago	9cdeeeb
✓ master		improved retrieval doc (#1358)	Luca Di Liello	6 days ago	ed2249d
		Fix wrong member naming in cer/wer (#1357)	Nicki Skafte Detle...	6 days ago	6dc67df
bugfix/wer_cer		fix	You	6 days ago	4223ac8
		CI: fix using actual TM version integrations (#13...	Jirka Borovec	7 days ago	cf013b5
ci/gpu-integrate		Merge branch 'master' into ci/gpu-integrate	mergify[bot]	7 days ago	4d7fe26
		Code cleaning after classification refactor 2/n (...	Nicki Skafte Detle...	7 days ago	20eab43
		Merge branch 'master' into ci/gpu-integrate	mergify[bot]	7 days ago	51a6922
feature/plot		Merge branch 'master' into feature/plot	mergify[bot]	7 days ago	28fc65a
		CI: fix using actual TM version integrations	Jirka	7 days ago	2bd67a2
		Fix precision issue in pairwise euclidean distanc...	Nicki Skafte Detle...	7 days ago	d66cbc3
		Merge branch 'master' into feature/plot	mergify[bot]	7 days ago	54cced9
cleanup/2		Merge branch 'master' into cleanup/2	mergify[bot]	7 days ago	694421a
bugfix/pairwis...		Merge branch 'master' into bugfix/pairwise_pre...	Nicki Skafte Detle...	7 days ago	0e08bfa
		mergify: drop ready for draft/wip (#1350)	Jirka Borovec	7 days ago	339a7a6
		changelog	You	7 days ago	9cc38e1
		fix	You	7 days ago	acbffc7
		Merge branch 'master' into cleanup/2	mergify[bot]	7 days ago	80c4e1e
		Merge branch 'master' into feature/plot	mergify[bot]	7 days ago	b27f436
		Cleanup of deprecated code (#1349) | * cleanu...	Nicki Skafte Detle...	7 days ago	68563ad
feature/plot		change requirement	You	7 days ago	895c7ff
		update with ax arg	You	7 days ago	c5b9845
		Merge branch 'master' into feature/plot	You	7 days ago	622fbfc
ci/mergify		mergify: drop ready for draft/wip	Jirka	7 days ago	ea3a3cd
cleanup/2		Merge branch 'master' into cleanup/2	Nicki Skafte Detle...	last week	2202d4d
cleanup/deprecated		Merge branch 'master' into cleanup/deprecated	Jirka Borovec	last week	62704bf
		ci: reruns failed tests (#1348) | * ci: rerun failed |...	Jirka Borovec	last week	7d77daf

CI step 1: Use pull requests

⚠ No commit can be pushed to master without being in a pull request



1. Find PR

Added MinkowskiDistance support #1362

clueless-skywatcher wants to merge 6 commits into `Lightning-AI:master` from `clueless-skywatcher:master`

2. Check changes

```
@@ -63,6 +63,7 @@
from torchmetrics.functional.regression.log_mse import mean_squared_log_error
from torchmetrics.functional.regression.mae import mean_absolute_error
from torchmetrics.functional.regression.mape import mean_absolute_percentage_error
```

```
from torchmetrics.functional.regression.log_mse import mean_squared_log_error
from torchmetrics.functional.regression.mae import mean_absolute_error
from torchmetrics.functional.regression.mape import mean_absolute_percentage_error
+ from torchmetrics.functional.regression.minkowski import minkowski_distance
```

SkafteNicki 2 days ago

please also add such an entry to the `functional/regression/__init__.py` file :]

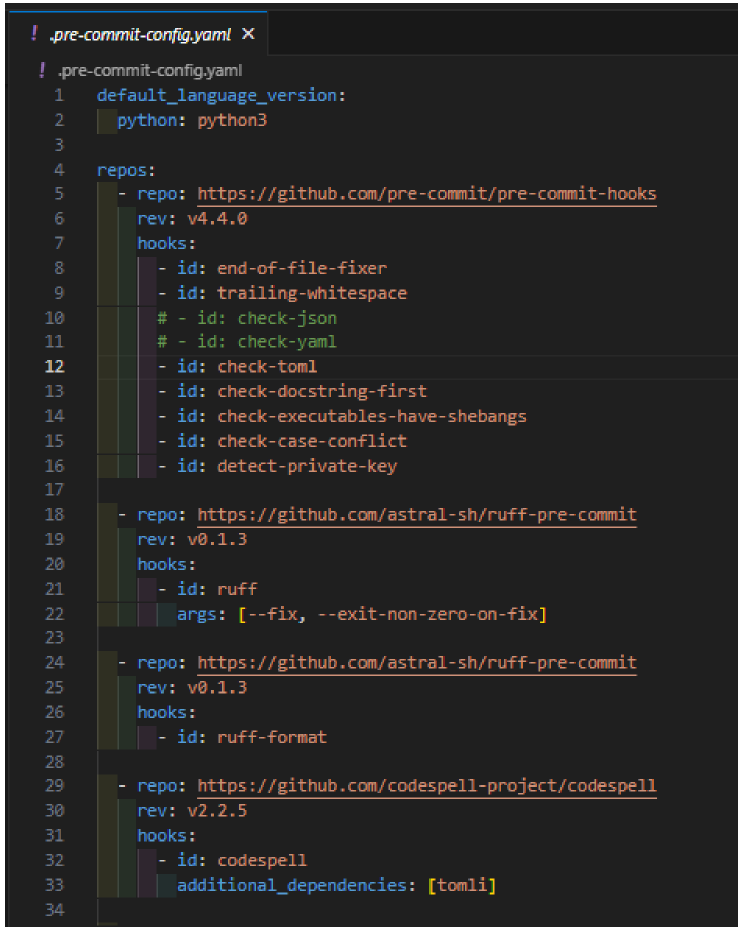
👍 1

3. Make one or more comments

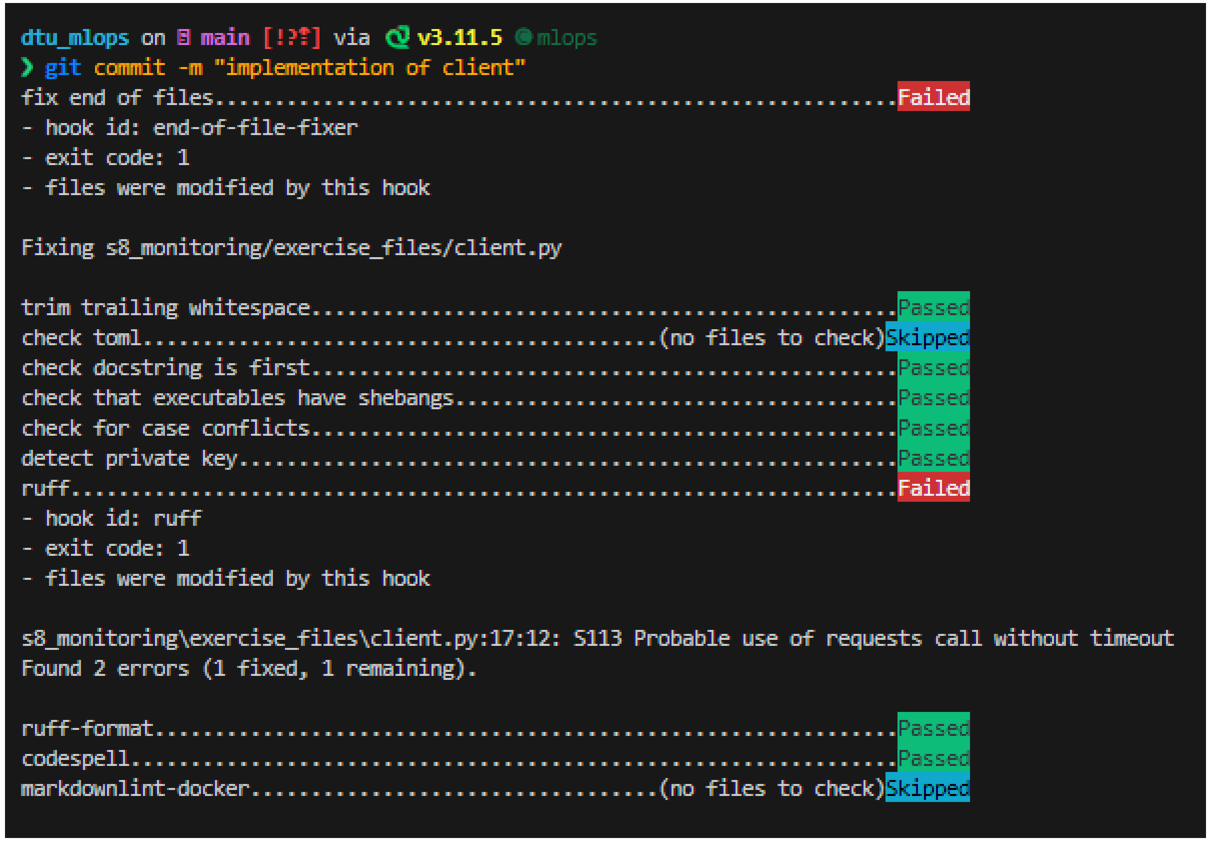
4. Send review

CI step 1: pre-commit

☑ Check that everything is up to standard before commits are created



```yaml
default_language_version:
  python: python3

repos:
  - repo: https://github.com/pre-commit/pre-commit-hooks
    rev: v4.4.0
    hooks:
      - id: end-of-file-fixer
      - id: trailing-whitespace
      # - id: check-json
      # - id: check-yaml
      - id: check-toml
      - id: check-docstring-first
      - id: check-executables-have-shebangs
      - id: check-case-conflict
      - id: detect-private-key

  - repo: https://github.com/astral-sh/ruff-pre-commit
    rev: v0.1.3
    hooks:
      - id: ruff
        args: [--fix, --exit-non-zero-on-fix]

  - repo: https://github.com/astral-sh/ruff-pre-commit
    rev: v0.1.3
    hooks:
      - id: ruff-format

  - repo: https://github.com/codespell-project/codespell
    rev: v2.2.5
    hooks:
      - id: codespell
        additional_dependencies: [tomli]

```

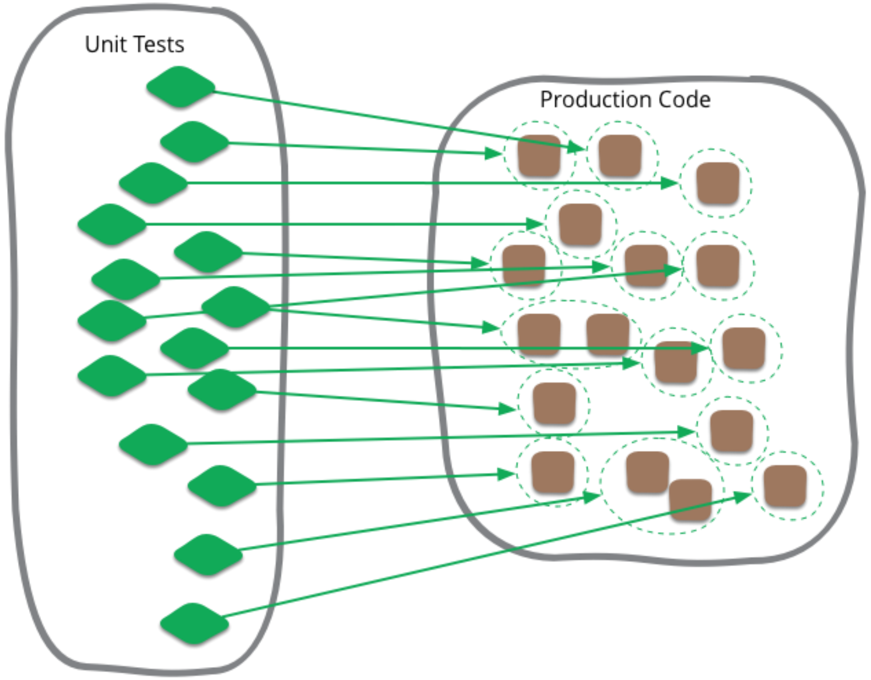


```
dtu_mlops on main [!?⇡] via v3.11.5 mlops
❯ git commit -m "implementation of client"
fix end of files........................................................Failed
- hook id: end-of-file-fixer
- exit code: 1
- files were modified by this hook

Fixing s8_monitoring/exercise_files/client.py

trim trailing whitespace................................................Passed
check toml...........................................(no files to check)Skipped
check docstring is first................................................Passed
check that executables have shebangs....................................Passed
check for case conflicts................................................Passed
detect private key......................................................Passed
ruff....................................................................Failed
- hook id: ruff
- exit code: 1
- files were modified by this hook

s8_monitoring\exercise_files\client.py:17:12: S113 Probable use of requests call without timeout
Found 2 errors (1 fixed, 1 remaining).

ruff-format.............................................................Passed
codespell...............................................................Passed
markdownlint-docker..................................(no files to check)Skipped
```

CI step 2: write tests

Tests are the cornerstones of continuous integration

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



Other test types worth considering:

- Integration tests
- Regression tests
- Performance tests
- Security tests

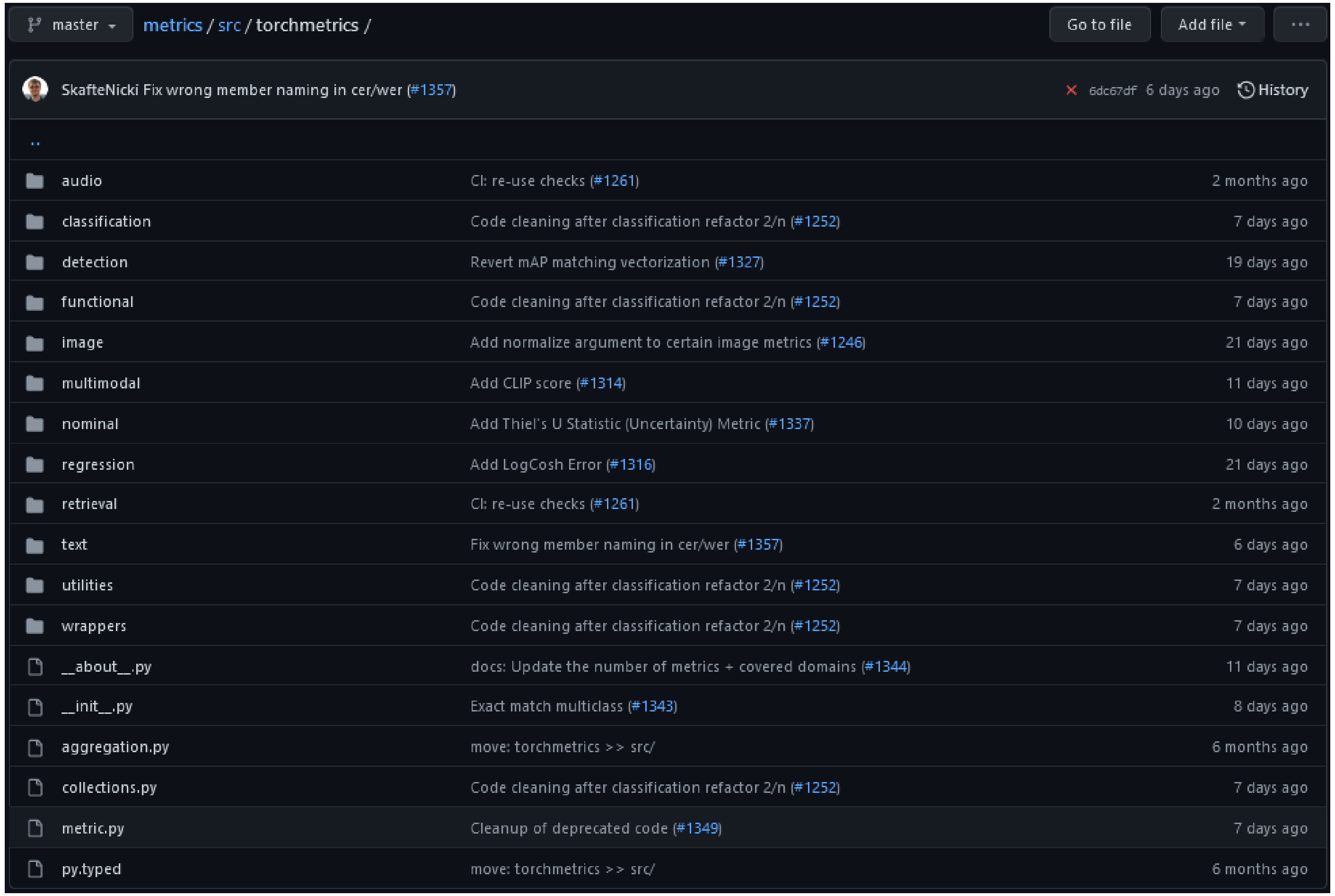
CI step 2: write tests

By Python convention your source code should either be

src/<project_name>
(src-layout)

or

<project_name>
(flat-layout)



SkafteNicki Fix wrong member naming in cer/wer (#1357) ✕ 6dc67df 6 days ago

..

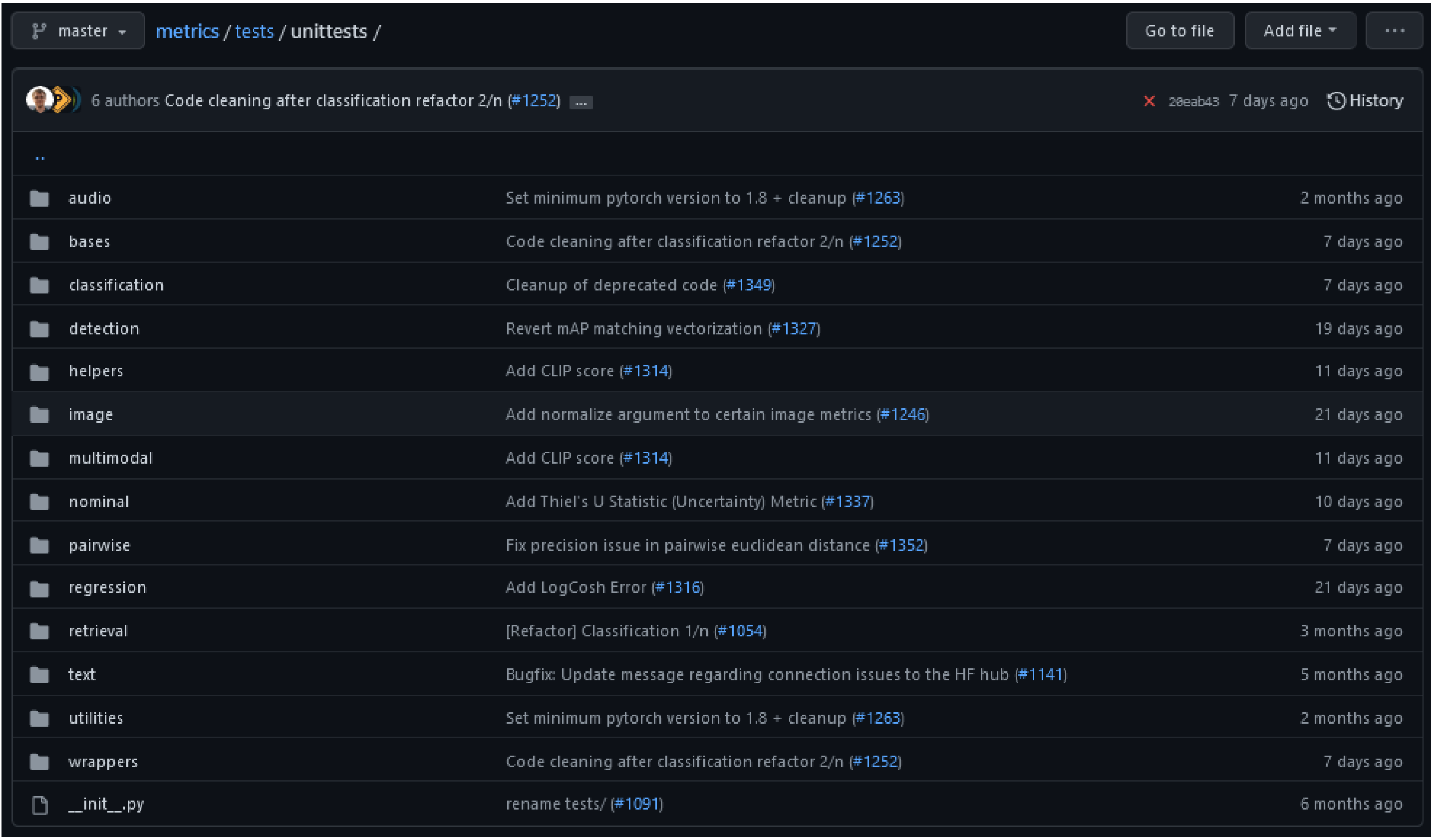
audio	CI: re-use checks (#1261)	2 months ago
classification	Code cleaning after classification refactor 2/n (#1252)	7 days ago
detection	Revert mAP matching vectorization (#1327)	19 days ago
functional	Code cleaning after classification refactor 2/n (#1252)	7 days ago
image	Add normalize argument to certain image metrics (#1246)	21 days ago
multimodal	Add CLIP score (#1314)	11 days ago
nominal	Add Thiel's U Statistic (Uncertainty) Metric (#1337)	10 days ago
regression	Add LogCosh Error (#1316)	21 days ago
retrieval	CI: re-use checks (#1261)	2 months ago
text	Fix wrong member naming in cer/wer (#1357)	6 days ago
utilities	Code cleaning after classification refactor 2/n (#1252)	7 days ago
wrappers	Code cleaning after classification refactor 2/n (#1252)	7 days ago
__about__.py	docs: Update the number of metrics + covered domains (#1344)	11 days ago
__init__.py	Exact match multiclass (#1343)	8 days ago
aggregation.py	move: torchmetrics >> src/	6 months ago
collections.py	Code cleaning after classification refactor 2/n (#1252)	7 days ago
metric.py	Cleanup of deprecated code (#1349)	7 days ago
py.typed	move: torchmetrics >> src/	6 months ago

CI step 2: write tests

For tests, the convention is to either place the tests in a separate tests folder, or put the tests in the same folder as the function/class/submodule they are testing.

```
├── README.md
├── src/
│   ├── __init__.py
│   └── important_functions.py
├── tests/
│   ├── __init__.py
│   └── test_important_functions.py
```

```
├── README.md
├── src/
│   ├── __init__.py
│   ├── submodule/
│   ├────── __init__.py
│   ├────── important_functions.py
│   └────── test_important_functions.py
```



master metricstestsunittests /

Go to file | Add file | ...

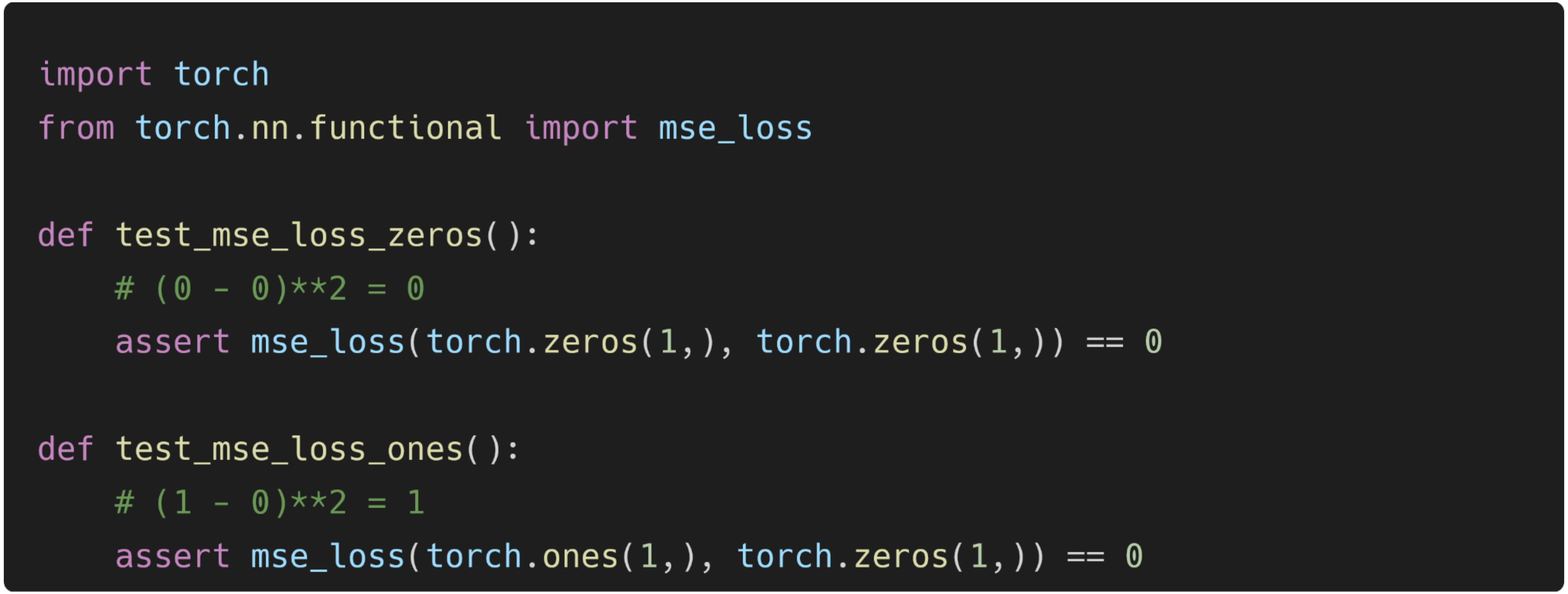
6 authors Code cleaning after classification refactor 2/n (#1252) ... 20eab43 7 days ago History

..

audio	Set minimum pytorch version to 1.8 + cleanup (#1263)	2 months ago
bases	Code cleaning after classification refactor 2/n (#1252)	7 days ago
classification	Cleanup of deprecated code (#1349)	7 days ago
detection	Revert mAP matching vectorization (#1327)	19 days ago
helpers	Add CLIP score (#1314)	11 days ago
image	Add normalize argument to certain image metrics (#1246)	21 days ago
multimodal	Add CLIP score (#1314)	11 days ago
nominal	Add Thiel's U Statistic (Uncertainty) Metric (#1337)	10 days ago
pairwise	Fix precision issue in pairwise euclidean distance (#1352)	7 days ago
regression	Add LogCosh Error (#1316)	21 days ago
retrieval	[Refactor] Classification 1/n (#1054)	3 months ago
text	Bugfix: Update message regarding connection issues to the HF hub (#1141)	5 months ago
utilities	Set minimum pytorch version to 1.8 + cleanup (#1263)	2 months ago
wrappers	Code cleaning after classification refactor 2/n (#1252)	7 days ago
__init__.py	rename tests/ (#1091)	6 months ago

CI step 2: write tests

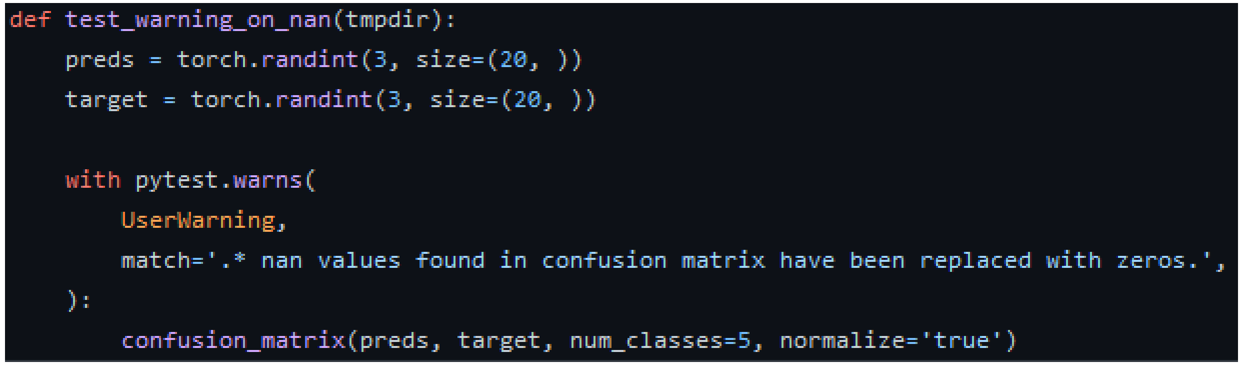
- In python, we recommend using the **pytest** framework.
- Test are simple functions that start with *test_* and uses *assert*



```python
import torch
from torch.nn.functional import mse_loss

def test_mse_loss_zeros():
    # (0 - 0)**2 = 0
    assert mse_loss(torch.zeros(1,), torch.zeros(1,)) == 0

def test_mse_loss_ones():
    # (1 - 0)**2 = 1
    assert mse_loss(torch.ones(1,), torch.zeros(1,)) == 0
```

CI step 2: write tests

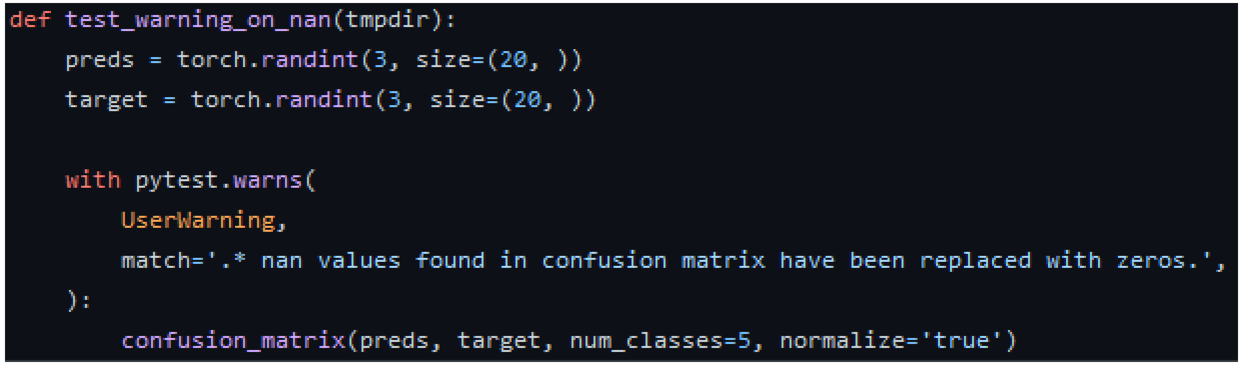
Test can be simple…



```
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')
```

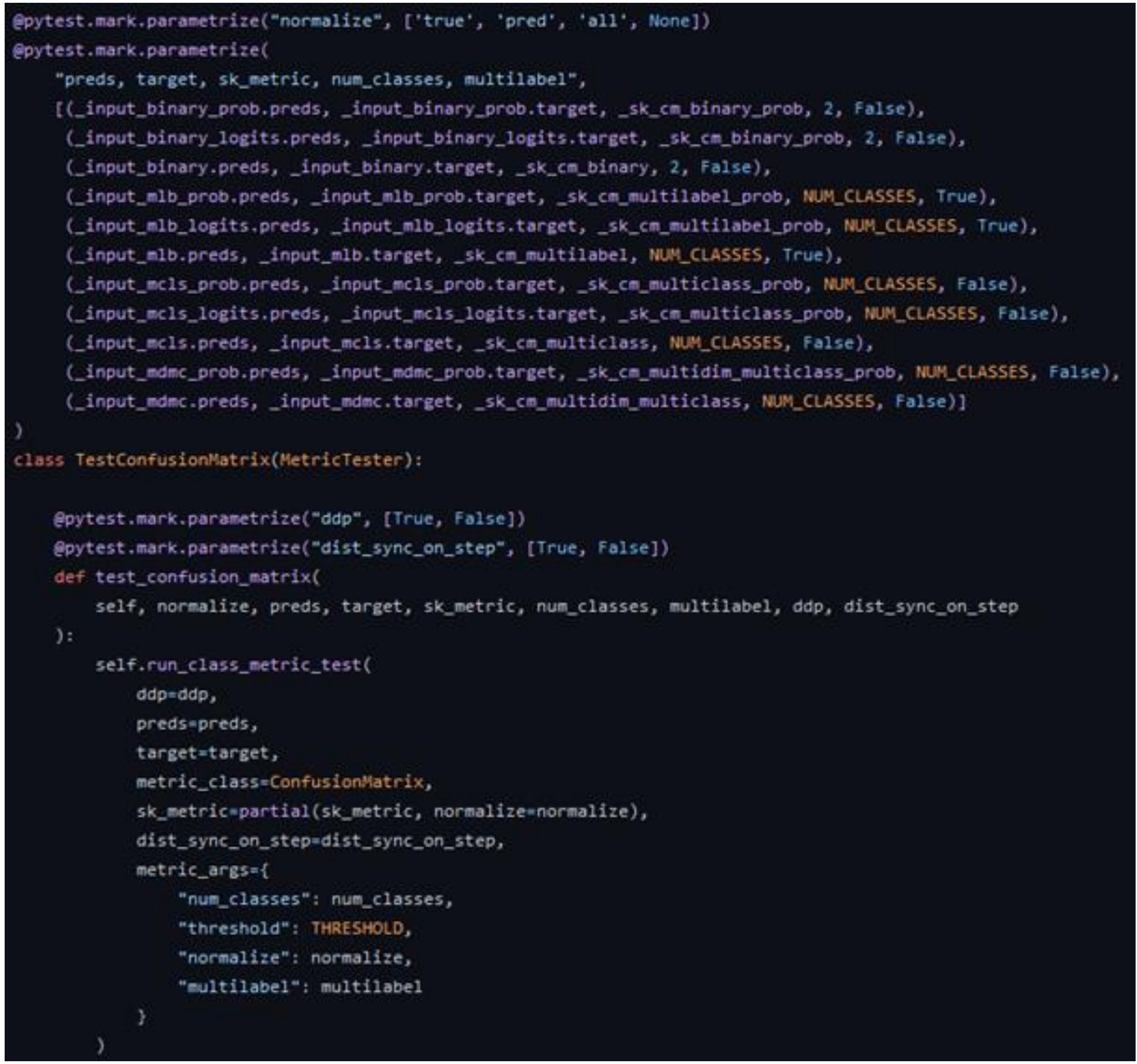
CI step 2: write tests

Test can be simple…



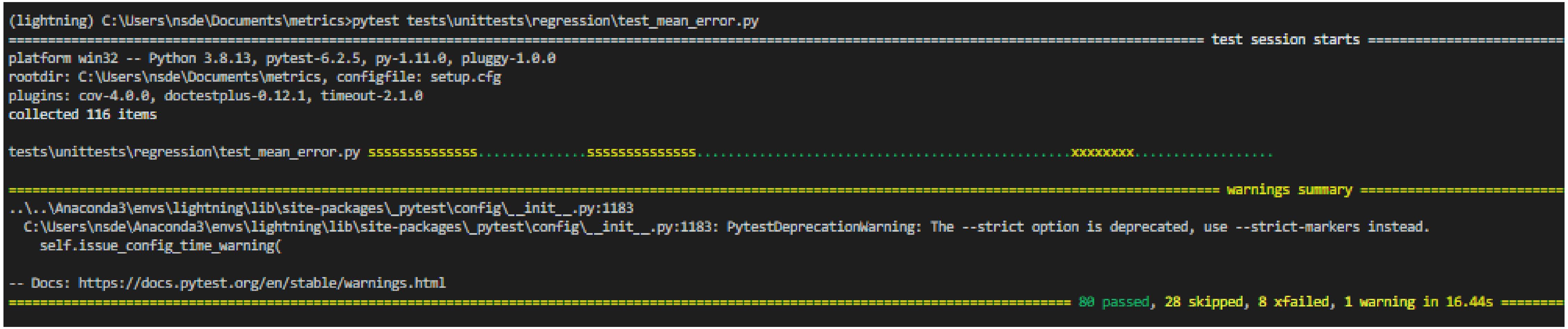
```python
def test_warning_on_nan(tmpdir):
    preds = torch.randint(3, size=(20, ))
    target = torch.randint(3, size=(20, ))

    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
            }
        )
```

CI step 2: execute locally



```
(lightning) C:\Users\nsde\Documents\metrics>pytest tests\unittests\regression\test_mean_error.py
========================================================================================================================================================== test session starts =========================
platform win32 -- Python 3.8.13, pytest-6.2.5, py-1.11.0, pluggy-1.0.0
rootdir: C:\Users\nsde\Documents\metrics, configfile: setup.cfg
plugins: cov-4.0.0, doctestplus-0.12.1, timeout-2.1.0
collected 116 items

tests\unittests\regression\test_mean_error.py ssssssssssssss..............ssssssssssssss................................................xxxxxxxx..................

=========================================================================================================================================================== warnings summary ==========================
..\..\Anaconda3\envs\lightning\lib\site-packages\_pytest\config\__init__.py:1183
  C:\Users\nsde\Anaconda3\envs\lightning\lib\site-packages\_pytest\config\__init__.py:1183: PytestDeprecationWarning: The --strict option is deprecated, use --strict-markers instead.
    self.issue_config_time_warning(

-- Docs: https://docs.pytest.org/en/stable/warnings.html
========================================================================================================================================= 80 passed, 28 skipped, 8 xfailed, 1 warning in 16.44s ========
```

. Test passed

F Test failed

S Test skipped (`pytest.skipif`, `pytest.skip`)

X Test was expected to fail (`pytest.xfail`)

Do you remember to do this before each commit?

Let's automate doing it instead

CI step 3: Automating stuff

What can be automated: EVERYTHING 🤖

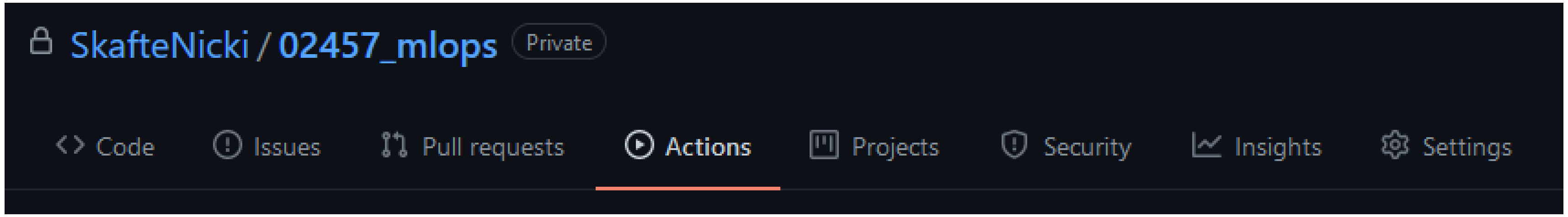
- 💡 Unit testing
- 💡 Integration testing
- 💡 Documentation creation
- 💡 Linters (style formatting)
- 💡 Security checks
- 💡 Code coverage
- 💡 Custom checks…

Only your imagination is the limit…

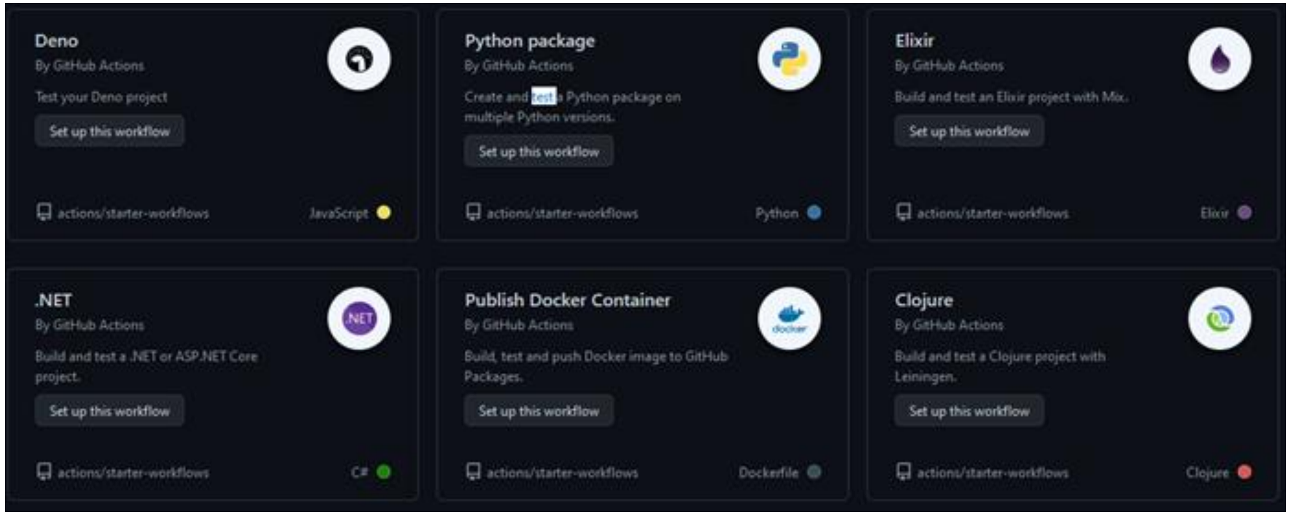
CI step 3: Github actions

Build-in continuous integration in Github.

Free 2000 automation minutes/month (public repository)



Many ready to go workflows



Deno
By GitHub Actions
Test your Deno project
Set up this workflow
actions/starter-workflows — JavaScript

Python package
By GitHub Actions
Create and test a Python package on multiple Python versions.
Set up this workflow
actions/starter-workflows — Python

Elixir
By GitHub Actions
Build and test an Elixir project with Mix.
Set up this workflow
actions/starter-workflows — Elixir

.NET
By GitHub Actions
Build and test a .NET or ASP.NET Core project.
Set up this workflow
actions/starter-workflows — C#

Publish Docker Container
By GitHub Actions
Build, test and push Docker image to GitHub Packages.
Set up this workflow
actions/starter-workflows — Dockerfile

Clojure
By GitHub Actions
Build and test a Clojure project with Leiningen.
Set up this workflow
actions/starter-workflows — Clojure

CI step 3: workflow files

Workflow files are a set of instructions that should be executed on a virtual machine hosted by Github

You can have one or many workflow files (runs in parallel)

When should workflow be triggered

Define OS + python

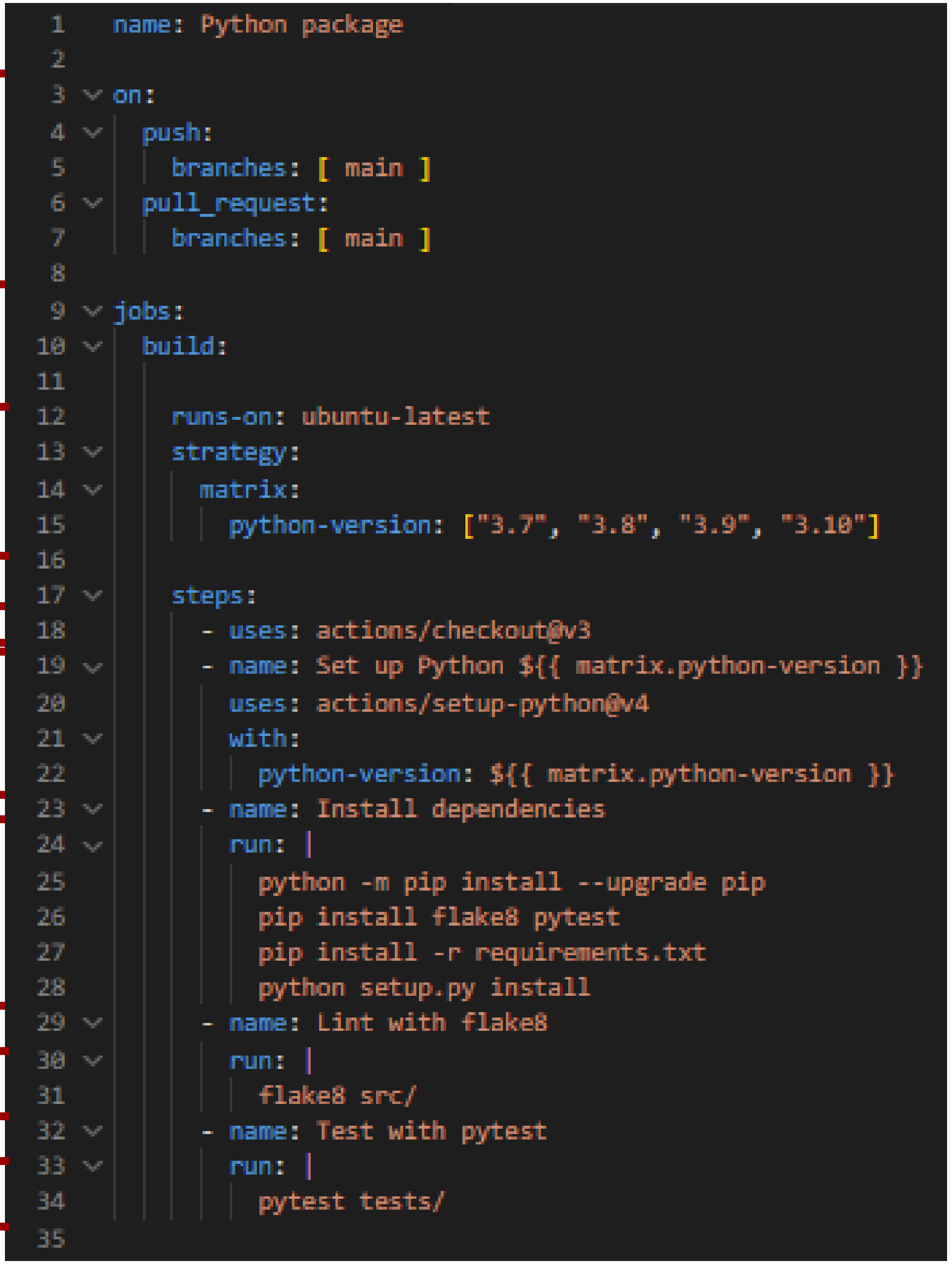
Clones code

Setup Python

Install dependencies

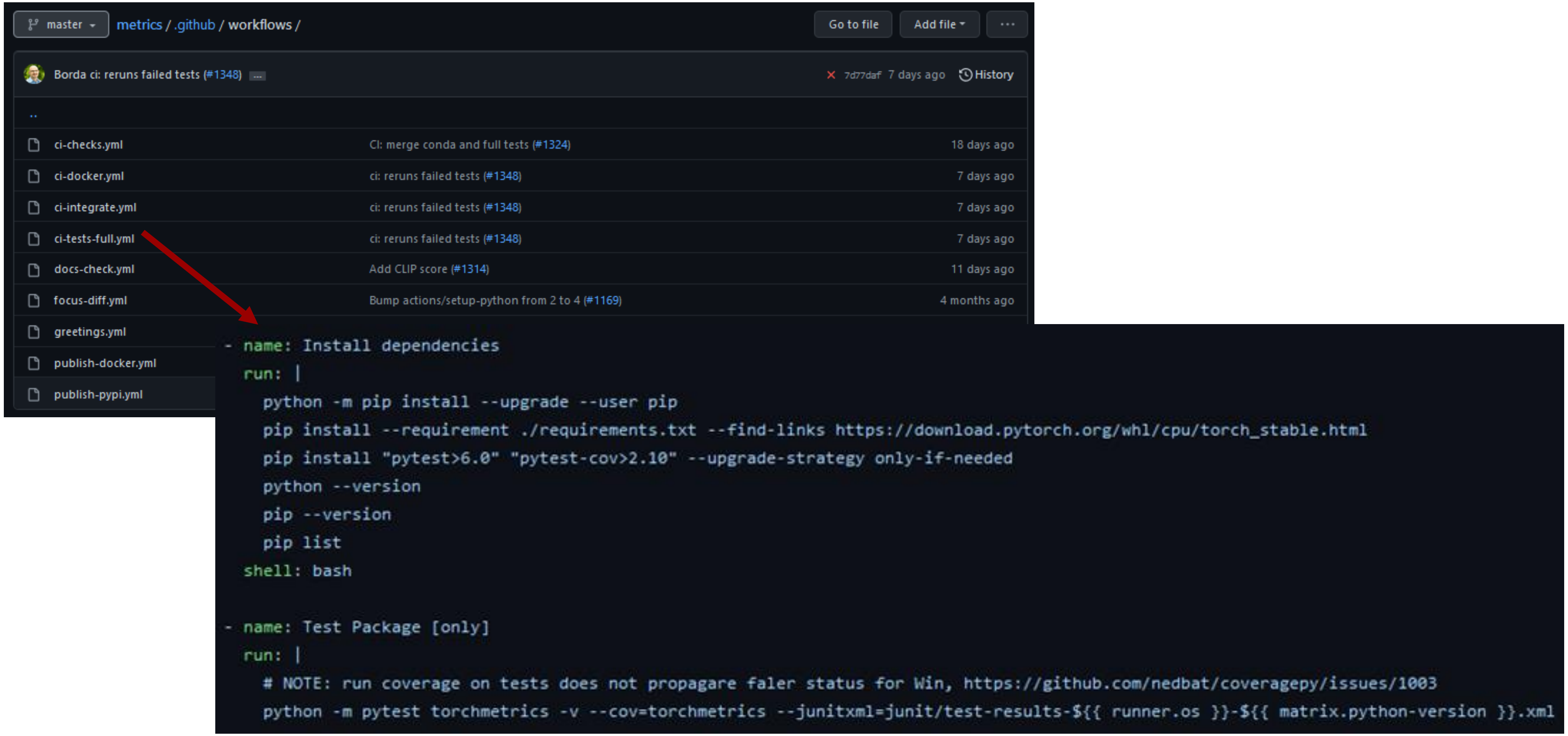
Check formatting

Run tests



```yaml
name: Python package

on:
  push:
    branches: [ main ]
  pull_request:
    branches: [ main ]

jobs:
  build:

    runs-on: ubuntu-latest
    strategy:
      matrix:
        python-version: ["3.7", "3.8", "3.9", "3.10"]

    steps:
      - uses: actions/checkout@v3
      - name: Set up Python ${{ matrix.python-version }}
        uses: actions/setup-python@v4
        with:
          python-version: ${{ matrix.python-version }}
      - name: Install dependencies
        run: |
          python -m pip install --upgrade pip
          pip install flake8 pytest
          pip install -r requirements.txt
          python setup.py install
      - name: Lint with flake8
        run: |
          flake8 src/
      - name: Test with pytest
        run: |
          pytest tests/

```

CI step 3: workflow files



Borda ci: reruns failed tests (#1348) ... ✕ 7d77daf 7 days ago

..		
ci-checks.yml	CI: merge conda and full tests (#1324)	18 days ago
ci-docker.yml	ci: reruns failed tests (#1348)	7 days ago
ci-integrate.yml	ci: reruns failed tests (#1348)	7 days ago
ci-tests-full.yml	ci: reruns failed tests (#1348)	7 days ago
docs-check.yml	Add CLIP score (#1314)	11 days ago
focus-diff.yml	Bump actions/setup-python from 2 to 4 (#1169)	4 months ago
greetings.yml		
publish-docker.yml		
publish-pypi.yml		

```yaml
- name: Install dependencies
  run: |
    python -m pip install --upgrade --user pip
    pip install --requirement ./requirements.txt --find-links https://download.pytorch.org/whl/cpu/torch_stable.html
    pip install "pytest>6.0" "pytest-cov>2.10" --upgrade-strategy only-if-needed
    python --version
    pip --version
    pip list
  shell: bash

- name: Test Package [only]
  run: |
    # NOTE: run coverage on tests does not propagare faler status for Win, https://github.com/nedbat/coveragepy/issues/1003
    python -m pytest torchmetrics -v --cov=torchmetrics --junitxml=junit/test-results-${{ runner.os }}-${{ matrix.python-version }}.xml
```

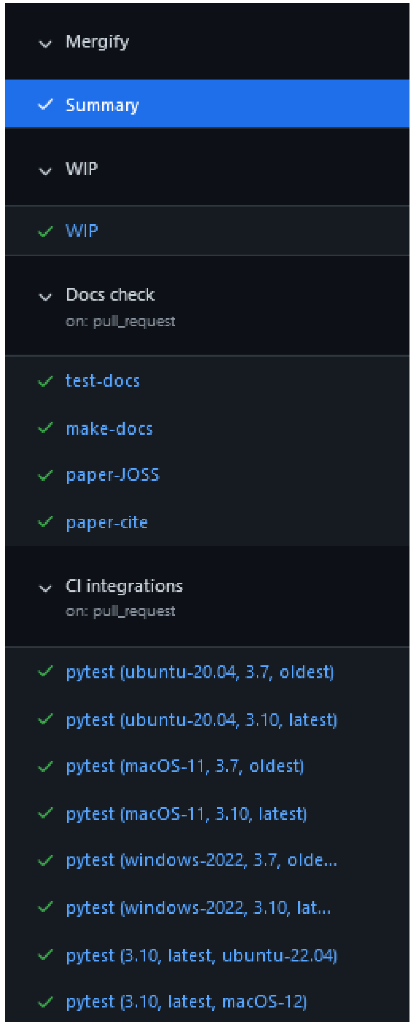
CI step 3: Workflow files

☑ 43 checks in total

Test a combination of

- Hardware setup
- Operating system
- Python version
- Core dependencies

Runs unit tests, build documentation, test coverage, linting of code, package installer etc.



Mergify

✓ Summary

WIP

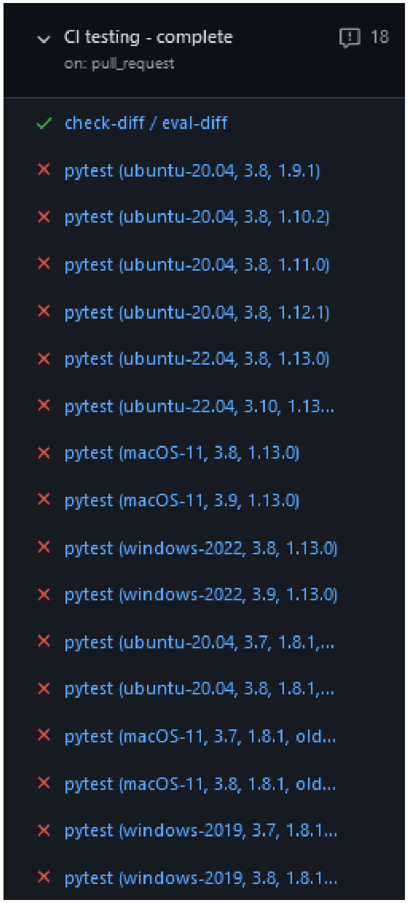
✓ WIP

Docs check
on: pull_request

✓ test-docs
✓ make-docs
✓ paper-JOSS
✓ paper-cite

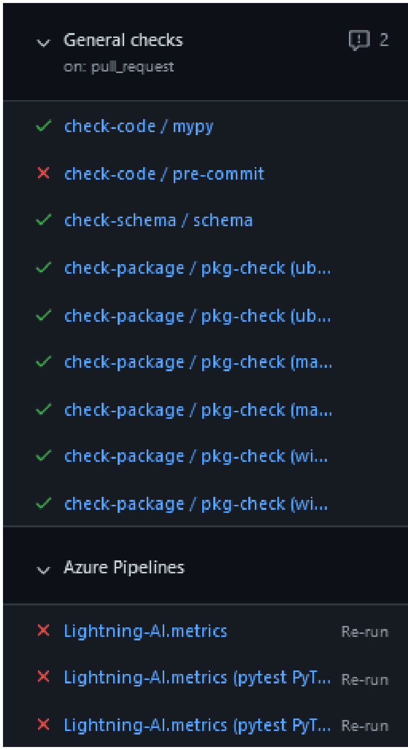
CI integrations
on: pull_request

✓ pytest (ubuntu-20.04, 3.7, oldest)
✓ pytest (ubuntu-20.04, 3.10, latest)
✓ pytest (macOS-11, 3.7, oldest)
✓ pytest (macOS-11, 3.10, latest)
✓ pytest (windows-2022, 3.7, olde...
✓ pytest (windows-2022, 3.10, lat...
✓ pytest (3.10, latest, ubuntu-22.04)
✓ pytest (3.10, latest, macOS-12)



CI testing - complete 18
on: pull_request

✓ check-diff / eval-diff
✗ pytest (ubuntu-20.04, 3.8, 1.9.1)
✗ pytest (ubuntu-20.04, 3.8, 1.10.2)
✗ pytest (ubuntu-20.04, 3.8, 1.11.0)
✗ pytest (ubuntu-20.04, 3.8, 1.12.1)
✗ pytest (ubuntu-22.04, 3.8, 1.13.0)
✗ pytest (ubuntu-22.04, 3.10, 1.13...
✗ pytest (macOS-11, 3.8, 1.13.0)
✗ pytest (macOS-11, 3.9, 1.13.0)
✗ pytest (windows-2022, 3.8, 1.13.0)
✗ pytest (windows-2022, 3.9, 1.13.0)
✗ pytest (ubuntu-20.04, 3.7, 1.8.1,...
✗ pytest (ubuntu-20.04, 3.8, 1.8.1,...
✗ pytest (macOS-11, 3.7, 1.8.1, old...
✗ pytest (macOS-11, 3.8, 1.8.1, old...
✗ pytest (windows-2019, 3.7, 1.8.1...
✗ pytest (windows-2019, 3.8, 1.8.1...



General checks 2
on: pull_request

✓ check-code / mypy
✗ check-code / pre-commit
✓ check-schema / schema
✓ check-package / pkg-check (ub...
✓ check-package / pkg-check (ub...
✓ check-package / pkg-check (ma...
✓ check-package / pkg-check (ma...
✓ check-package / pkg-check (wi...
✓ check-package / pkg-check (wi...

Azure Pipelines

✗ Lightning-AI.metrics Re-run
✗ Lightning-AI.metrics (pytest PyT... Re-run
✗ Lightning-AI.metrics (pytest PyT... Re-run

CI step 3: Code is checked before merging

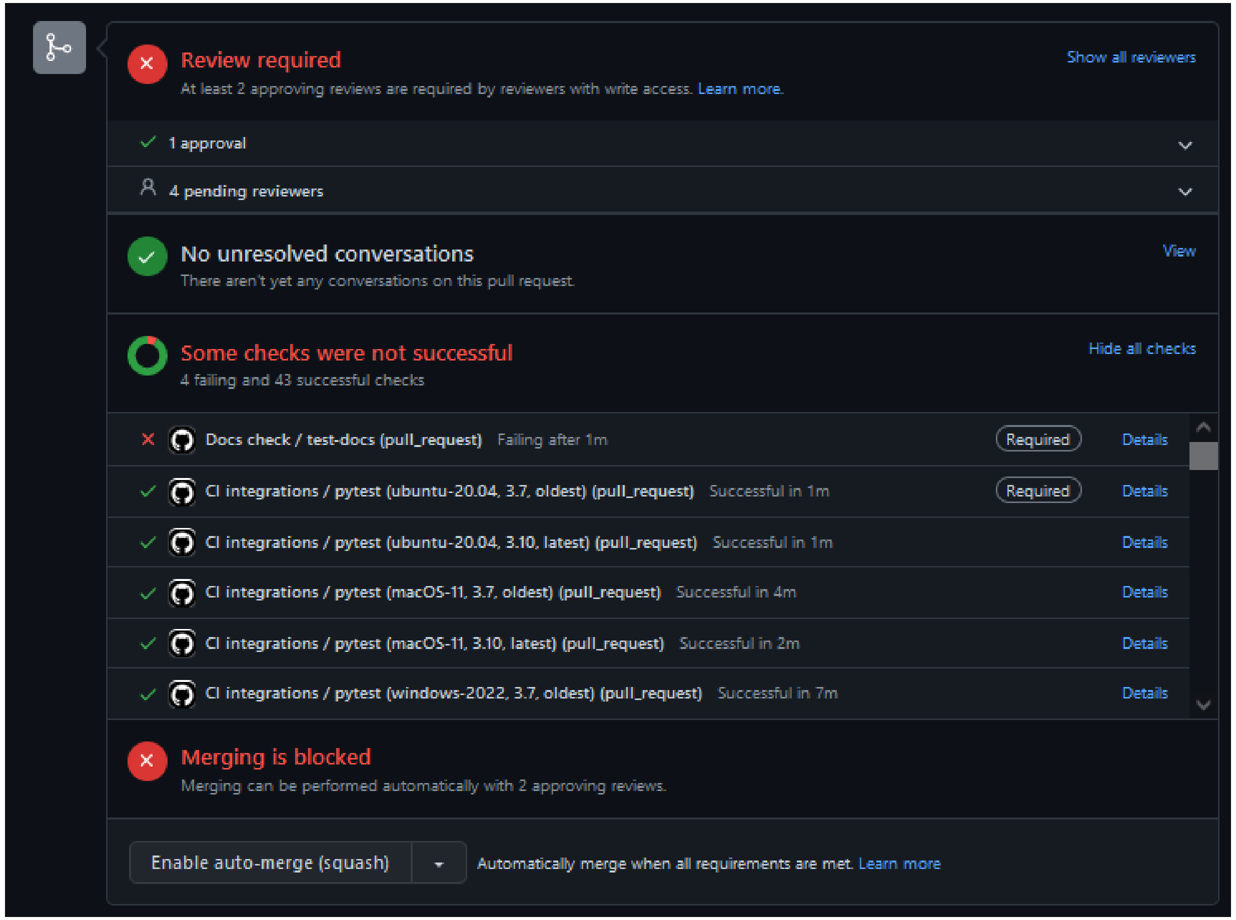
Branch protection rules:

⚠All/some tests should pass

⚠At least x core members need to approve

⚠Comments should be taken care of

View more here



Review required
At least 2 approving reviews are required by reviewers with write access. Learn more.
Show all reviewers

✓ 1 approval

4 pending reviewers

No unresolved conversations
There aren't yet any conversations on this pull request.
View

Some checks were not successful
4 failing and 43 successful checks
Hide all checks

✕ Docs check / test-docs (pull_request) Failing after 1m Required Details

✓ CI integrations / pytest (ubuntu-20.04, 3.7, oldest) (pull_request) Successful in 1m Required Details

✓ CI integrations / pytest (ubuntu-20.04, 3.10, latest) (pull_request) Successful in 1m Details

✓ CI integrations / pytest (macOS-11, 3.7, oldest) (pull_request) Successful in 4m Details

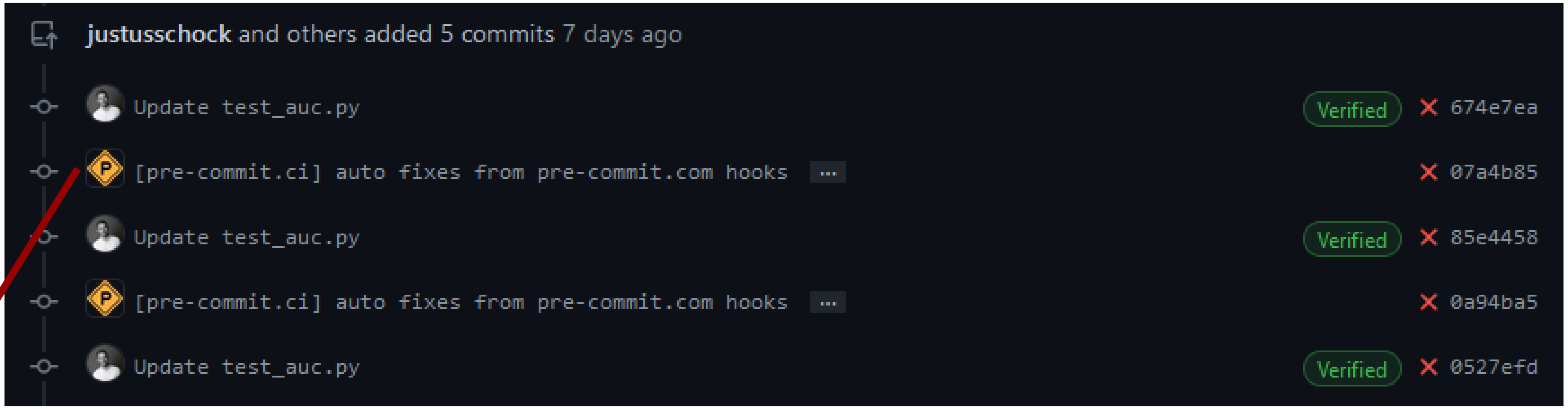
✓ CI integrations / pytest (macOS-11, 3.10, latest) (pull_request) Successful in 2m Details

✓ CI integrations / pytest (windows-2022, 3.7, oldest) (pull_request) Successful in 7m Details

Merging is blocked
Merging can be performed automatically with 2 approving reviews.

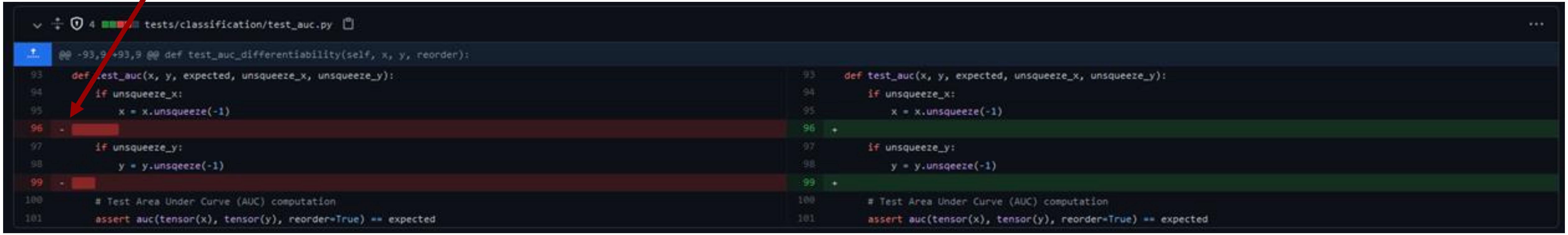
Enable auto-merge (squash) Automatically merge when all requirements are met. Learn more

CI step 3: Automate tedious tasks with bots



justusschock and others added 5 commits 7 days ago

```
Update test_auc.py                                      Verified  ✕ 674e7ea
[pre-commit.ci] auto fixes from pre-commit.com hooks ···          ✕ 07a4b85
Update test_auc.py                                      Verified  ✕ 85e4458
[pre-commit.ci] auto fixes from pre-commit.com hooks ···          ✕ 0a94ba5
Update test_auc.py                                      Verified  ✕ 0527efd
```

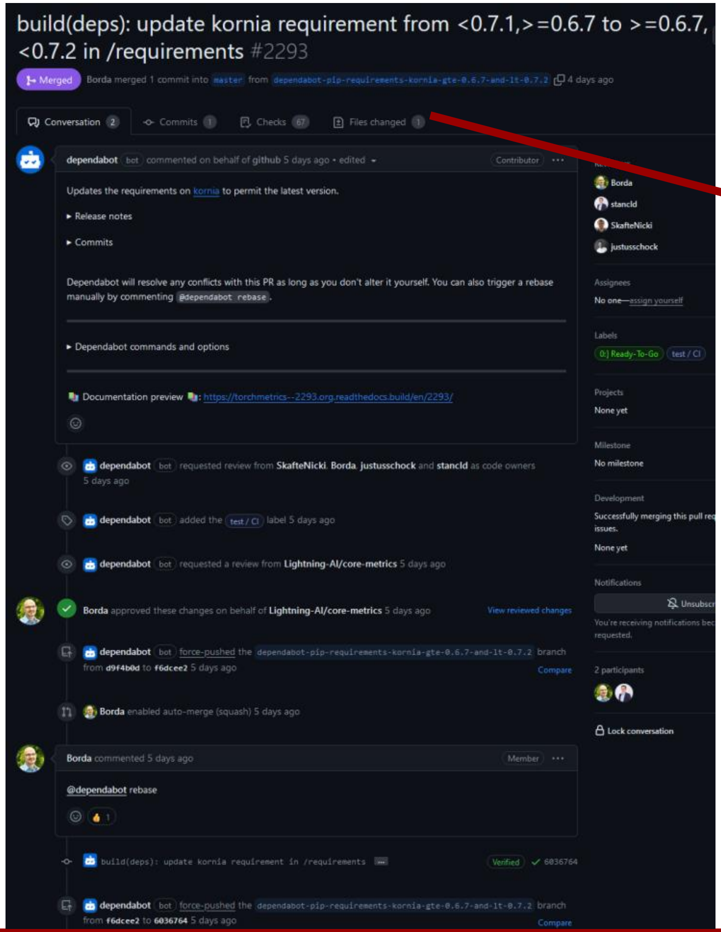


4 tests/classification/test_auc.py

```
@@ -93,9 +93,9 @@ def test_auc_differentiability(self, x, y, reorder):
    def test_auc(x, y, expected, unsqueeze_x, unsqueeze_y):      def test_auc(x, y, expected, unsqueeze_x, unsqueeze_y):
        if unsqueeze_x:                                              if unsqueeze_x:
            x = x.unsqueeze(-1)                                          x = x.unsqueeze(-1)
-                                                             +
        if unsqueeze_y:                                              if unsqueeze_y:
            y = y.unsqeeze(-1)                                           y = y.unsqeeze(-1)
-                                                             +
        # Test Area Under Curve (AUC) computation                    # Test Area Under Curve (AUC) computation
        assert auc(tensor(x), tensor(y), reorder=True) == expected   assert auc(tensor(x), tensor(y), reorder=True) == expected
```

CI step 3: Automate tedious tasks with bots

Dependabot can help auto checking new releases of dependencies in your project



build(deps): update kornia requirement from <0.7.1,>=0.6.7 to >=0.6.7, <0.7.2 in /requirements #2293

Merged Borda merged 1 commit into `master` from `dependabot-pip-requirements-kornia-gte-0.6.7-and-lt-0.7.2` 4 days ago

dependabot bot commented on behalf of github 5 days ago • edited · Contributor

Updates the requirements on kornia to permit the latest version.

▸ Release notes

▸ Commits

Dependabot will resolve any conflicts with this PR as long as you don't alter it yourself. You can also trigger a rebase manually by commenting `@dependabot rebase`.

▸ Dependabot commands and options

Documentation preview: https://torchmetrics--2293.org.readthedocs.build/en/2293/

dependabot bot requested review from SkafteNicki, Borda, justusschock and stancld as code owners 5 days ago

dependabot bot added the test / CI label 5 days ago

dependabot bot requested a review from Lightning-AI/core-metrics 5 days ago

Borda approved these changes on behalf of Lightning-AI/core-metrics 5 days ago · View reviewed changes

dependabot bot force-pushed the `dependabot-pip-requirements-kornia-gte-0.6.7-and-lt-0.7.2` branch from `d9f4b0d` to `f6dcee2` 5 days ago · Compare

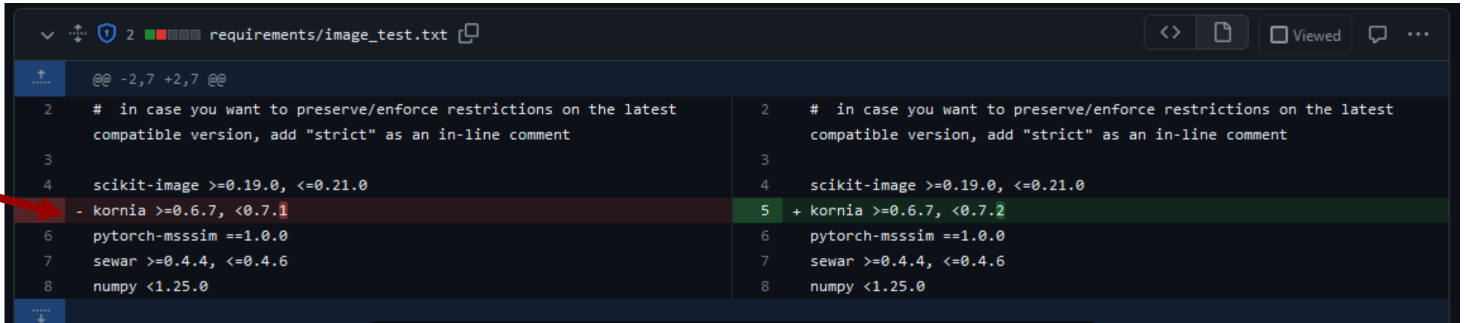
Borda enabled auto-merge (squash) 5 days ago

Borda commented 5 days ago · Member

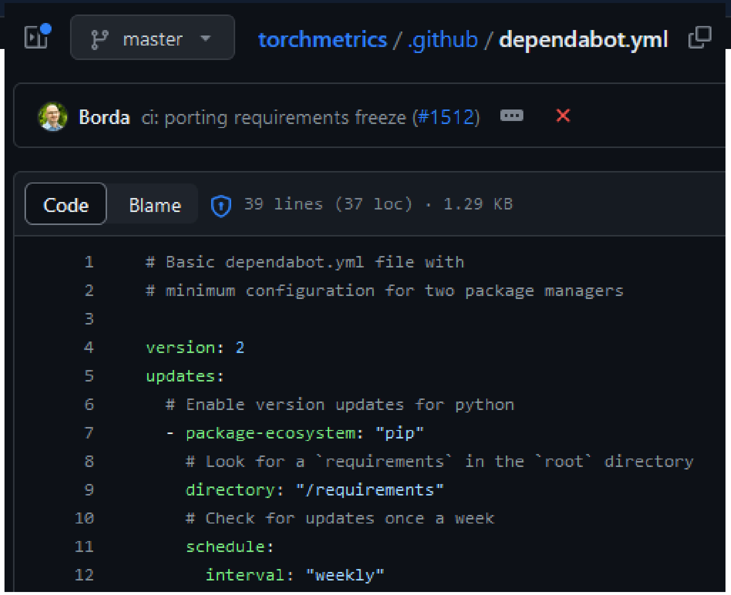
@dependabot rebase

`build(deps): update kornia requirement in /requirements` Verified ✓ 6036764

dependabot bot force-pushed the `dependabot-pip-requirements-kornia-gte-0.6.7-and-lt-0.7.2` branch from `f6dcee2` to `6036764` 5 days ago · Compare



```diff
@@ -2,7 +2,7 @@
 #  in case you want to preserve/enforce restrictions on the latest compatible version, add "strict" as an in-line comment

 scikit-image >=0.19.0, <=0.21.0
- kornia >=0.6.7, <0.7.1
+ kornia >=0.6.7, <0.7.2
 pytorch-msssim ==1.0.0
 sewar >=0.4.4, <=0.4.6
 numpy <1.25.0
```



master
torchmetrics / .github / dependabot.yml
Borda ci: porting requirements freeze (#1512)
Code
Blame
39 lines (37 loc) · 1.29 KB

```yaml
# Basic dependabot.yml file with
# minimum configuration for two package managers

version: 2
updates:
  # Enable version updates for python
  - package-ecosystem: "pip"
    # Look for a `requirements` in the `root` directory
    directory: "/requirements"
    # Check for updates once a week
    schedule:
      interval: "weekly"
```

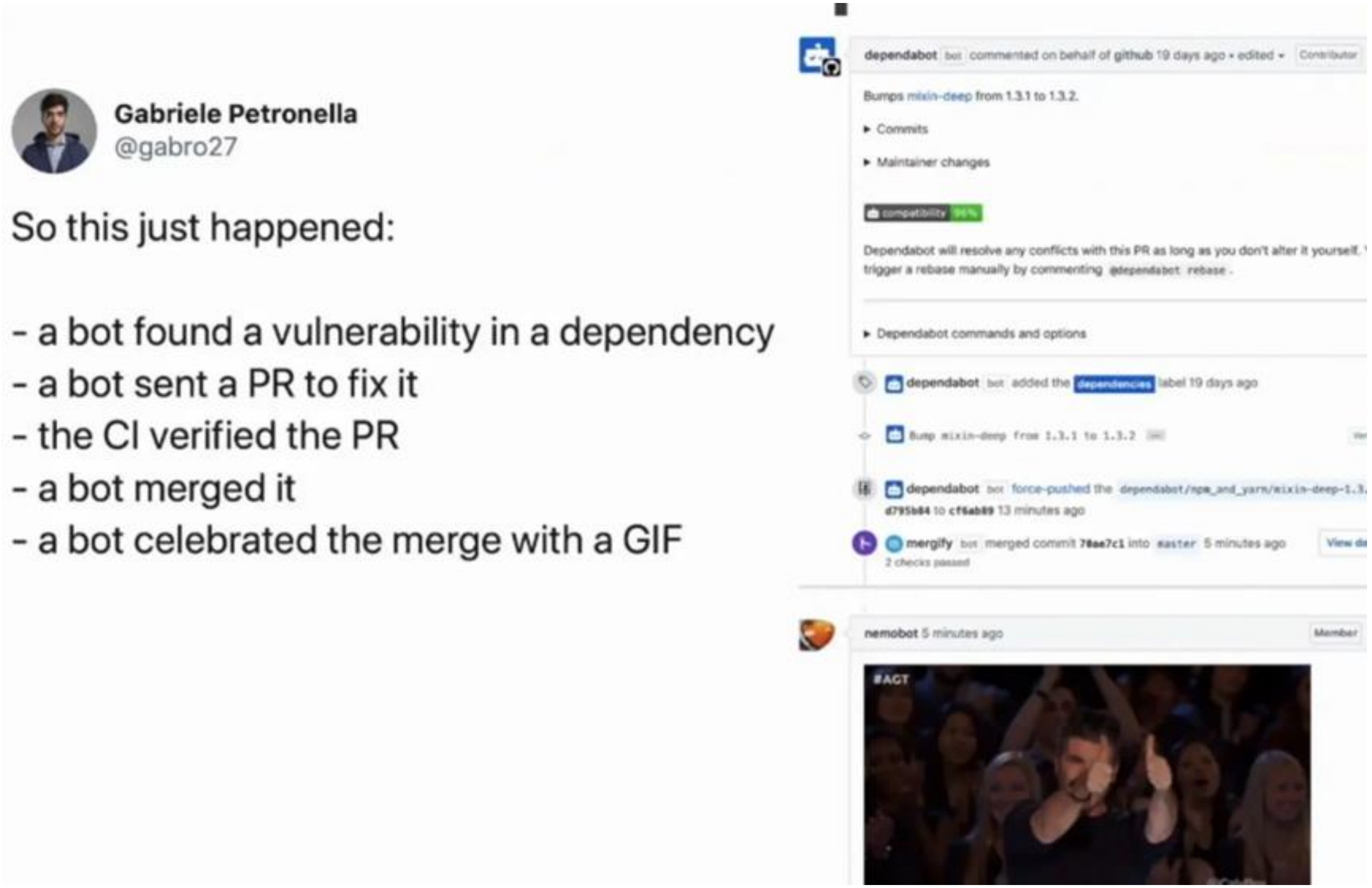
Summary of continues integration

1. Use version control

2. Write (unit-)test for your code

3. Automate build + test

Meme of the day



Gabriele Petronella
@gabro27

So this just happened:

- a bot found a vulnerability in a dependency
- a bot sent a PR to fix it
- the CI verified the PR
- a bot merged it
- a bot celebrated the merge with a GIF

dependabot bot commented on behalf of **github** 19 days ago • edited ▾ Contributor

Bumps mixin-deep from 1.3.1 to 1.3.2.

▸ Commits

▸ Maintainer changes

compatibility 96%

Dependabot will resolve any conflicts with this PR as long as you don't alter it yourself.
trigger a rebase manually by commenting `@dependabot rebase`.

▸ Dependabot commands and options

dependabot bot added the dependencies label 19 days ago

`Bump mixin-deep from 1.3.1 to 1.3.2`

dependabot bot force-pushed the `dependabot/npm_and_yarn/mixin-deep-1.3.`
d795b84 to **cf6ab89** 13 minutes ago

mergify bot merged commit **70ae7c1** into `master` 5 minutes ago
2 checks passed

nemobot 5 minutes ago Member