



Crash-Course in Contributing to Open Source Projects

Andreas Müller
(and contributions from: Reshama Shaikh)

Gitter: <https://gitter.im/scikit-learn/sprint>

Crash-Course in Contributing to Open Source Projects

(with some specific instructions
to sklearn)

(Great!) Reasons to Contribute to Open Source

- Giving back. Users of open source like to give back and participate in the community.
- Contributing to open source is an excellent learning opportunity which advances one's data science and Python coding skills.
- Fix bugs, add features & solve a problem that (we) developers faces in work.
- OS is movement, projects flourish with various contributors (FB: Pytorch, Microsoft: VSC)

(Great!) Reasons to Contribute to Open Source

- In DS more appealing to work with OS tools; the familiarity, allows job transition, increases value of tech skills because they are transferable, synergy of knowledge
- Interact with core developers, who are experts
- Job opportunities
- Fun

Environment

- If you don't have a working python installation, install Anaconda.
- If you don't want to install the scikit-learn development version in your main environment, create a separate environment (with conda or virtualenv):

```
conda create -n sklearndev numpy scipy matplotlib pytest  
sphinx cython ipykernel
```

```
source activate sklearndev
```

```
conda install -c conda-forge sphinx-gallery
```

Fork sklearn on github

<https://github.com/scikit-learn/scikit-learn>

scikit-learn / scikit-learn

Used by 109k

Watch 2.2k

Unstar 40.8k

Fork 19.7k

<> Code

Issues 1,541

Pull requests 760

Actions

Projects 17

Wiki

Security 0

Insights

scikit-learn: machine learning in Python <https://scikit-learn.org>

machine-learningpythonstatisticsdata-scienceanalysis

25,540 commits

21 branches

0 packages

111 releases

1,661 contributors

View license

Branch: masterNew pull request

Create new fileUpload filesFind fileClone or download

amuel ENH use colors from style in tree plotting (#17187) Latest commit 2f26540 21 hours ago

.binder	MNT Update nightly build URL and release staging config (#16435)	3 months ago
.circleci	MNT CI dropping python-3.5 (#15106)	3 months ago
.github	MNT Reduces github API calls greatly in autolabeler (#16617)	3 months ago
benchmarks	MNT New options for higgs-boson benchmark (#16779)	2 months ago
build_tools	CI Check default doctest directives in linter (#17269)	2 days ago
doc	ENH use colors from style in tree plotting (#17187)	21 hours ago
examples	DOC Amend plot_digits_classification.py to use notebook style (#17016)	yesterday
maint_tools	DOC cleaning up to 0.23/whats new (#17015)	24 days ago
sklearn	ENH use colors from style in tree plotting (#17187)	21 hours ago
.codecov.yml	MAINT don't track coverage on setups (#15175)	7 months ago

Clone the Fork

Firefox Web browser

amuellet / scikit-learn
forked from scikit-learn/scikit-learn

Unwatch 3 Unstar 9 Fork 9,344

Code Pull requests 0 Projects 0 Wiki Pulse Graphs Settings

scikit-learn main repo <http://scikit-learn.sourceforge.net>

New Add topics

21,555 commits 275 branches 43 releases 752 contributors BSD-3-Clause

Branch: master New pull request

Create new file Upload files Find file Clone or download

This branch is 154 commits behind scikit-learn:master.

TomDLT committed with ogriamel [MRG+] Add multiplicative-update solver in NMF, with all beta-diverg...

benchmarks MAING make benchmarks scripts Python 3 compatible

build_tools CI report which doc files were likely affected (#8132)

doc [MRG+] Add multiplicative-update solver in NMF, with all beta-diverg... 3 months ago

examples [MRG+] Add multiplicative-update solver in NMF, with all beta-diverg... 3 months ago

sklearn [MRG+] Add multiplicative-update solver in NMF, with all beta-diverg... 3 months ago

Clone with SSH ? Use HTTPS

Use an SSH key and passphrase from account.

git@github.com:amuellet:scikit-learn.git

use https

Download ZIP

If you don't have ssh keys set up, use https!

Add main scikit-learn repo as remote called "upstream":

```
git remote add upstream https://github.com/scikit-learn/scikit-learn.git
```

Build and run tests!

- **Easiest way:**

```
cd scikit-learn  
pip install -e .
```

- **Will overwrite existing installations!**

Starting on Issues

- Go to:

<https://github.com/data-umbrella/2020-sklearn-sprint/projects/1>

- Comment on issue saying: “I'm working on this.”
- Update local master branch from main sklearn repo

```
git pull upstream master
```

- Fetching someone else's PR:

```
git fetch
```

```
https://github.com/theirusername/reponame.git  
theirbranch:ourbranch
```

Starting on Issues

- Create feature branch

```
git checkout -b <branchname>
```

- Commit changes to branch, run tests

```
pytest sklearn
```

(or individual test files)

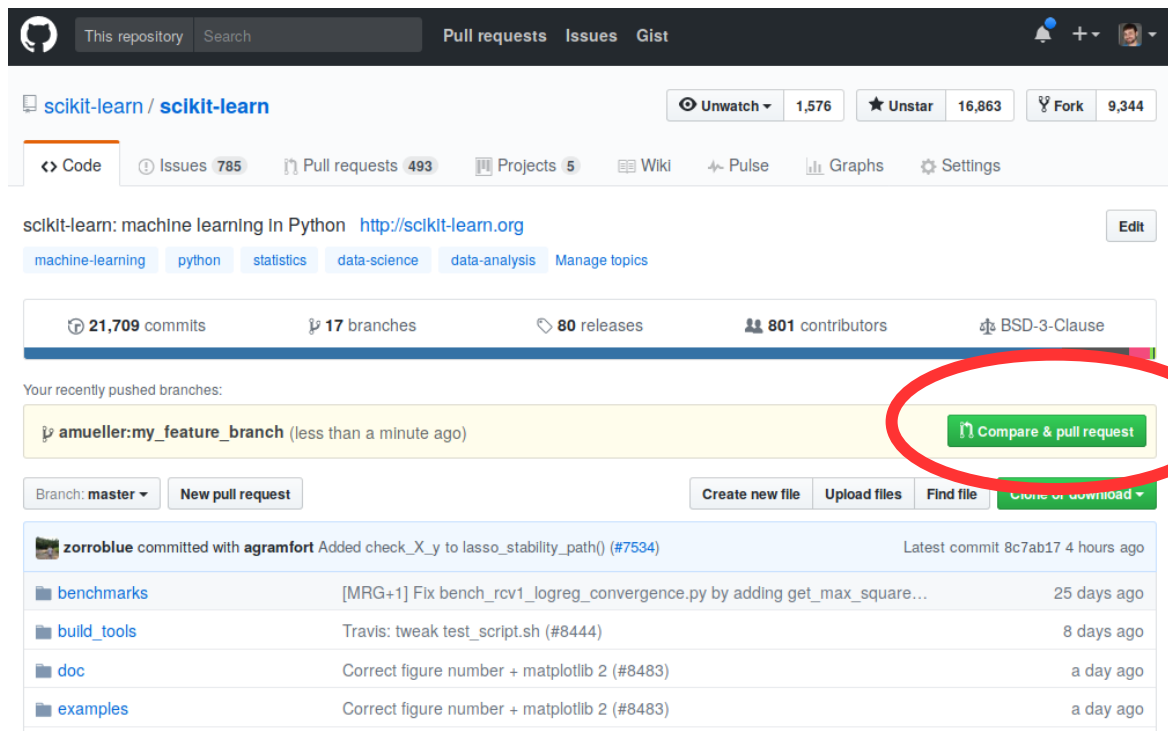
- Run `flake8` on changed files

Creating a PR

- Push changes to your fork

```
git push origin <branchname>
```


- Create PR using GitHub UI:




Describing PR

Open a pull request

Create a new pull request by comparing changes across two branches. If you need to, you can also [compare across forks](#).

 base fork: **scikit-learn/scikit-learn** ▼ base: **master** ▼ ... head fork: **amueller/scikit-learn** ▼ compare: **my_feature_branch** ▼ ✓ **Able to merge.** These branches can be merged.

Please review the [guidelines for contributing](#) to this repository.



[MRG] minor pep8

Write


Preview

<!--
Thanks for contributing a pull request! Please ensure you have taken a look at
the contribution guidelines: <https://github.com/scikit-learn/scikit-learn/blob/master/CONTRIBUTING.md#Contributing-Pull-Requests>
-->
Reference Issue
<!-- Example: Fixes #1234 -->


Attach files by dragging & dropping, [selecting them](#), or pasting from the clipboard.

☒ **Allow edits from maintainers.** [Learn more](#)


Create pull request

Reviewers 


No reviews—request one

Assignees 

No one—assign yourself

Labels 

None yet

Milestone 

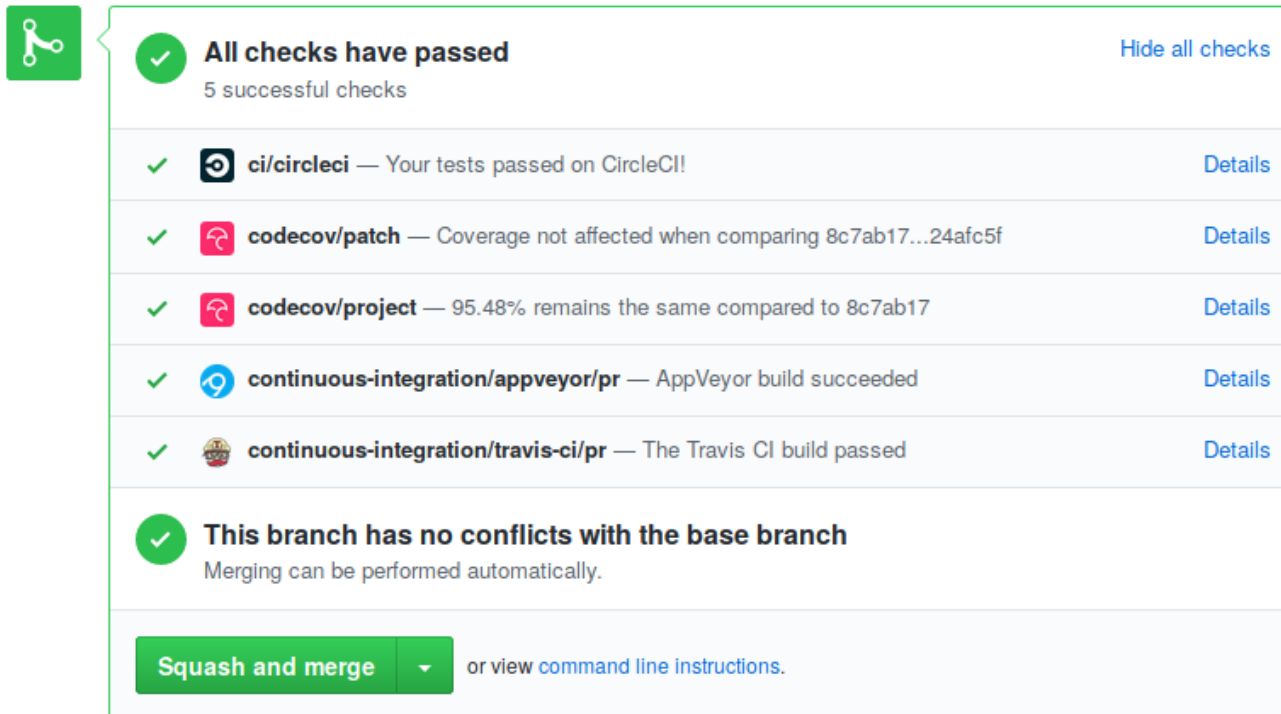
No milestone

[WIP] = work in progress
[MRG] = ready to merge

12

Regression tests

- Are mandatory! For everything (except documentation changes)!
- Make sure continuous integration passes:



The screenshot shows the GitHub Actions status bar for a pull request. It features a green checkmark icon on the left. The main status is "All checks have passed" with a green checkmark and "5 successful checks". A link "Hide all checks" is on the right. Below this, a list of checks is shown, each with a green checkmark, the check name, a description, and a "Details" link. The checks are: "ci/circleci" (Your tests passed on CircleCI), "codecov/patch" (Coverage not affected when comparing 8c7ab17...24afc5f), "codecov/project" (95.48% remains the same compared to 8c7ab17), "continuous-integration/appveyor/pr" (AppVeyor build succeeded), and "continuous-integration/travis-ci/pr" (The Travis CI build passed). At the bottom, there is a green button "Squash and merge" with a dropdown arrow, followed by the text "or view [command line instructions](#)."

Check Name	Description	Details
ci/circleci	Your tests passed on CircleCI!	Details
codecov/patch	Coverage not affected when comparing 8c7ab17...24afc5f	Details
codecov/project	95.48% remains the same compared to 8c7ab17	Details
continuous-integration/appveyor/pr	AppVeyor build succeeded	Details
continuous-integration/travis-ci/pr	The Travis CI build passed	Details

[Squash and merge](#) or view [command line instructions](#).

What's next?

- Wait for reviews (be patient).
- Address review comments in the same branch.
- Pushing to your fork will update the PR.
- Reviewers will “approve” PR or change title to [MRG + 1]
- You need two approvals for a merge.

Finding Issues

- Check “need contributor”, “easy” and “sprint” issues.
- Something unclear in the docs? Fix it!
- Can’t fix something that’s unclear? Open an issue!
- Problem that you keep running into: Open an issue!
- Find stalled PRs (the author didn’t address reviews for $\sim > 1$ month) and continue them!

<https://github.com/data-umbrella/2020-sklearn-sprint/projects/1>

Reviewing

- You can review PRs and issues!
- Some bugs are not confirmed. See if you can confirm them and under what conditions?
- You can review documentation PRs for language and whether they are clear to you.
- You can review code changes on whether they address the issue (might be a bit tricky).
- Don't be afraid to ask clarifying questions!

Workflow Reminders

- Why we do pair programming
- Include @mention of other pair programmer in PR submission
- Following up on work; expect a lot of back / forth, it's part of the process. It is worth the experience to see a PR through to the merge state!
- Watch your GitHub notifications (especially for folks who use an alternative email address that is not checked often, or it gets filtered into a folder)
- Policy of "abandoned / stalled PRs." Etiquette is to communicate status.

Final words

- Pick something TRIVIALY SIMPLE as the first contribution.
- You can do the cool stuff afterwards!
- There might be interesting issues that are not appropriately tagged.

Thank you for your help!
Enjoy!