

<https://gitter.im/scikit-learn/wimlds>

Crash-Course in Contributing to Open Source Projects

(with some specific instructions
to sklearn)

Andreas Müller

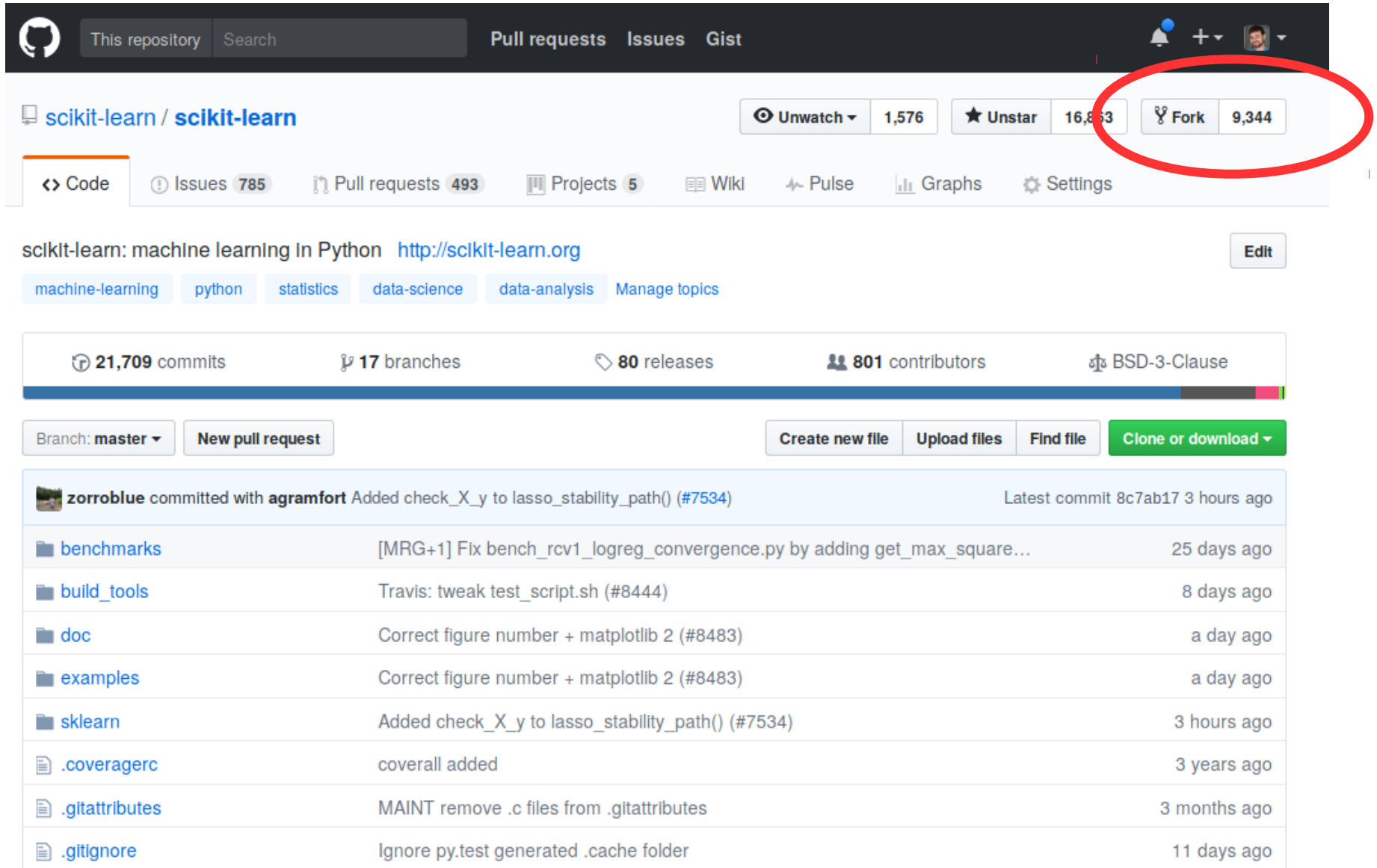
Environment

- 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  
nose sphinx cython  
source activate sklearndev
```

Fork sklearn on github

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



The screenshot shows the GitHub repository page for `scikit-learn / scikit-learn`. The `Fork` button is circled in red. The repository has 1,576 stars, 16,853 forks, and 9,344 forks. The `Code` tab is selected. The repository description is "scikit-learn: machine learning In Python" with a link to <http://scikit-learn.org>. The repository has 21,709 commits, 17 branches, 80 releases, 801 contributors, and is licensed under BSD-3-Clause. The latest commit is by `zorroblue` committed with `agramfort`, adding `check_X_y` to `lasso_stability_path()` (#7534) 3 hours ago. The file list includes `benchmarks`, `build_tools`, `doc`, `examples`, `sklearn`, `.coveragerc`, `.gitattributes`, and `.gitignore`.

GitHub repository page for `scikit-learn / scikit-learn`. The `Fork` button is circled in red.

Repository details:

- Stars: 1,576
- Forks: 16,853
- Forks count: 9,344

Repository description: scikit-learn: machine learning In Python <http://scikit-learn.org>

Repository statistics:

- 21,709 commits
- 17 branches
- 80 releases
- 801 contributors
- BSD-3-Clause license

Repository tabs: `Code`, `Issues 785`, `Pull requests 493`, `Projects 5`, `Wiki`, `Pulse`, `Graphs`, `Settings`

Repository actions: `Create new file`, `Upload files`, `Find file`, `Clone or download`

Repository history:

Commit	Message	Time
<code>zorroblue</code> committed with <code>agramfort</code>	Added <code>check_X_y</code> to <code>lasso_stability_path()</code> (#7534)	Latest commit 8c7ab17 3 hours ago
<code>[MRG+1]</code>	Fix <code>bench_rcv1_logreg_convergence.py</code> by adding <code>get_max_square...</code>	25 days ago
Travis	tweak <code>test_script.sh</code> (#8444)	8 days ago
	Correct figure number + <code>matplotlib 2</code> (#8483)	a day ago
	Correct figure number + <code>matplotlib 2</code> (#8483)	a day ago
	Added <code>check_X_y</code> to <code>lasso_stability_path()</code> (#7534)	3 hours ago
	<code>coverall</code> added	3 years ago
	MAINT remove <code>.c</code> files from <code>.gitattributes</code>	3 months ago
	Ignore <code>py.test</code> generated <code>.cache</code> folder	11 days ago

Clone the Fork

The screenshot shows the GitHub interface for a forked repository. At the top, the repository name 'amuel / scikit-learn' is circled in green. Below it, the 'Code' button is also circled in green. On the right, there are buttons for 'Unwatch', 'Unstar', and 'Fork'. The 'Fork' button shows 9,344 forks. Below the repository name, there are buttons for 'Code', 'Pull requests', 'Projects', 'Wiki', 'Pulse', 'Graphs', and 'Settings'. The main content area shows the repository's statistics: 21,555 commits, 275 branches, 43 releases, and 752 contributors. A red circle highlights the 'Clone or download' button, which has a dropdown menu open. The dropdown menu shows 'Clone with SSH' (selected) and 'Use HTTPS'. The SSH URL is 'git@github.com:amuel/scikit-learn.git'. Below the dropdown, there is a 'Download ZIP' button. The repository's description and commit history are visible in the background.

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

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

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

Build and run tests!

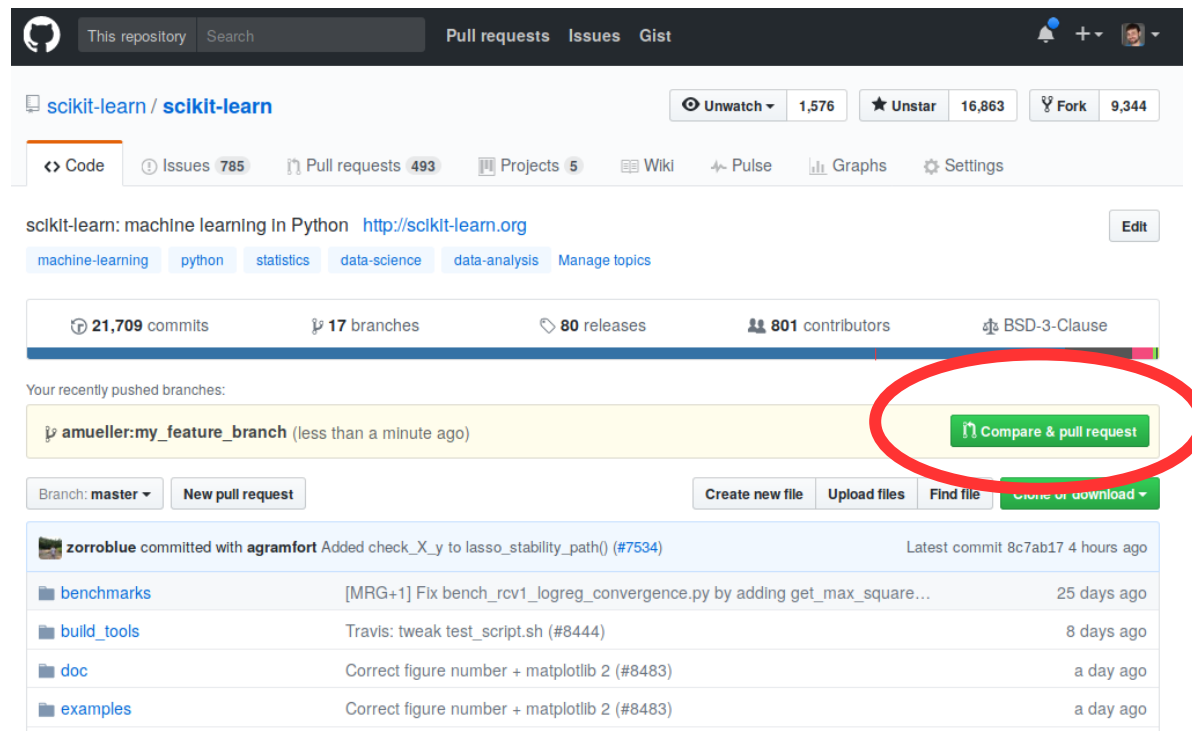
- Easiest way:
`cd scikit-learn`
`pip install -e .`
- Will overwrite existing installations!
- Or:
`make`
(won't "install", you have to add folder to your
PYTHONPATH)

Starting on Issues

- Go to
<https://github.com/scikit-learn/scikit-learn/issues?q=is%3Aissue+is%3Aopen+label%3ASprint>
- Comment on issue saying “I’m working in this”.
- Update local master branch from main sklearn repo (`git pull upstream master`).
- Create feature branch
`git checkout -b <branchname>`
- Commit changes to branch, run tests
`nosetests -sv 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 into the base branch.

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



[MRG] minor pep8

Write

Preview

AA ▼ B i “ < > 🔗 ☰ ☷ ☹

<!--
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

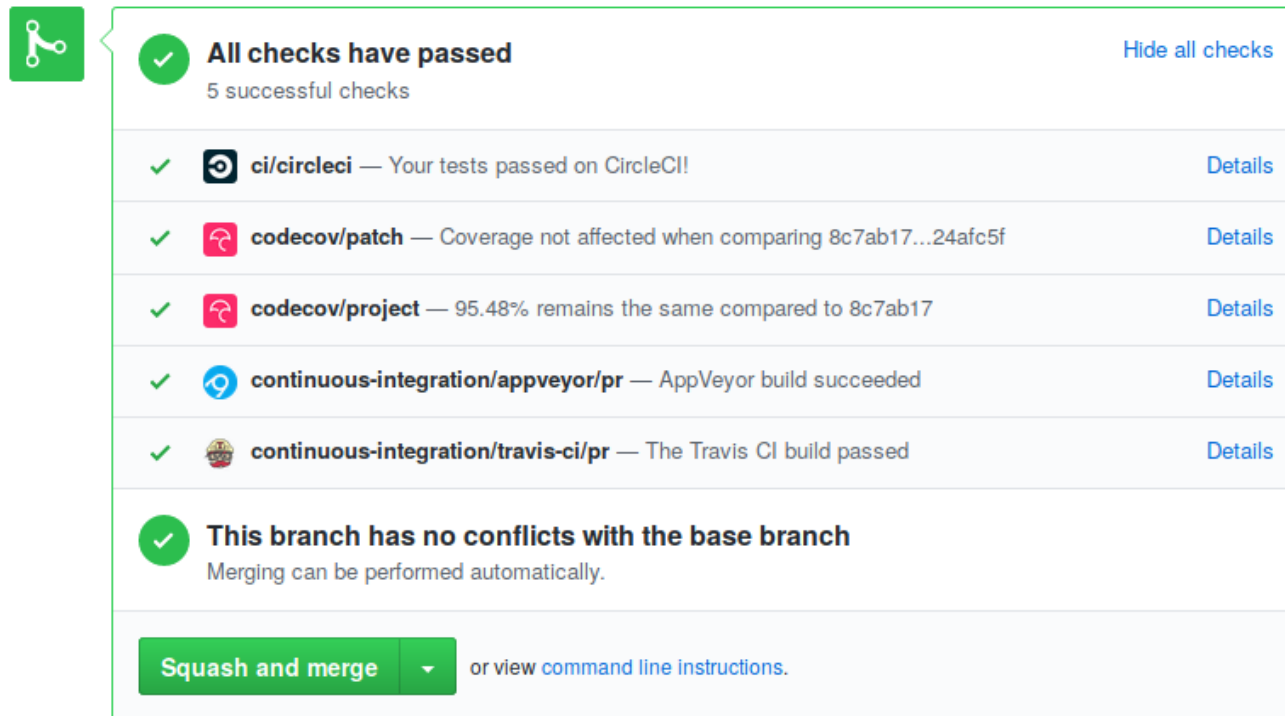
Labels ⚙️
None yet

Milestone ⚙️
No milestone



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











Regression tests


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



The image shows a GitHub 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 link to "Hide all checks". Below this, there are five individual check items, each with a green checkmark, a service icon, 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 "Squash and merge" button and a link to "or view command line instructions."

  **All checks have passed** [Hide all checks](#)
5 successful checks

	 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

 **This branch has no conflicts with the base branch**
Merging can be performed automatically.

[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!

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!

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!