



Open Source Sprint Instructions for scikit-learn

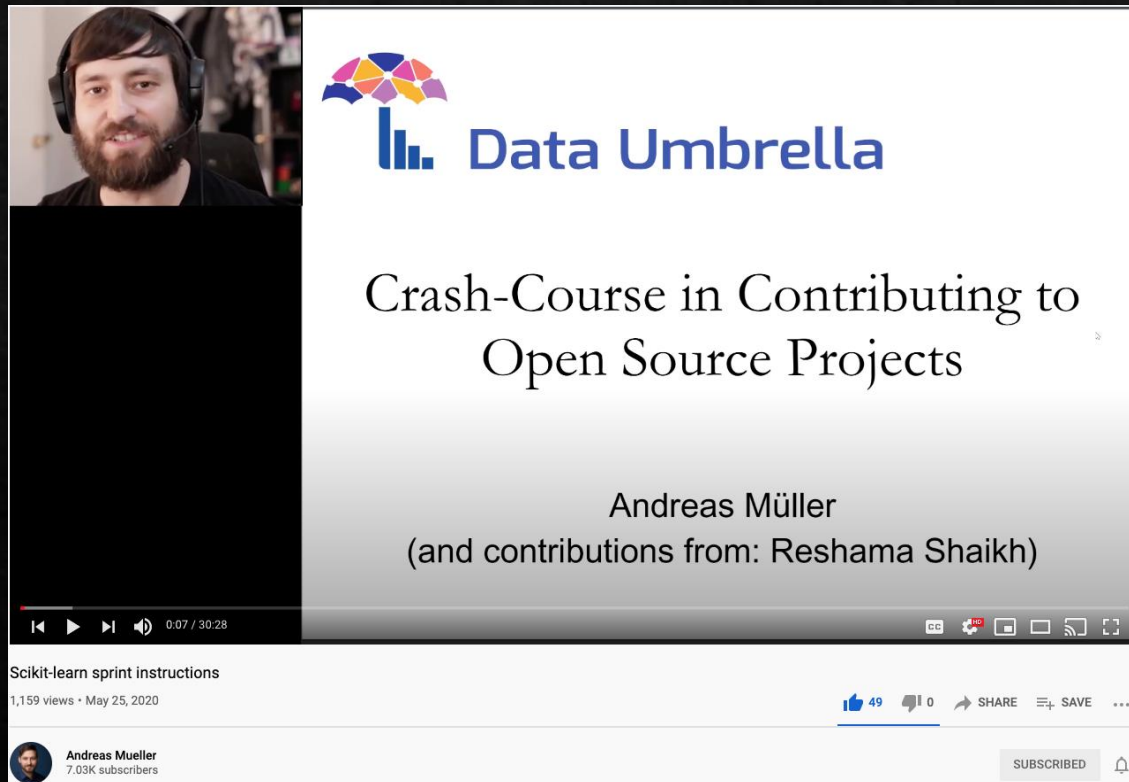
Volume 2

Andreas Mueller
Reshama Shaikh

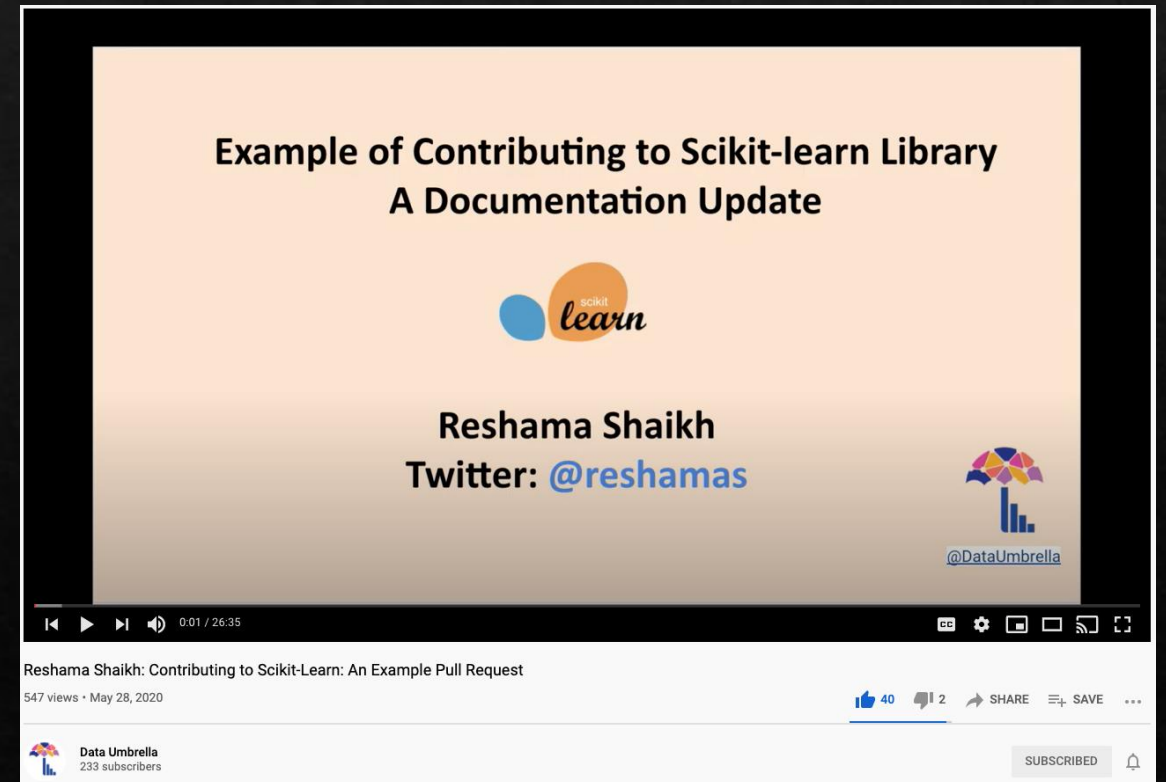
Check Out the Basics: Videos

www.dataumbrella.org/open-source/contributing-to-scikit-learn

Note: Full transcripts are available



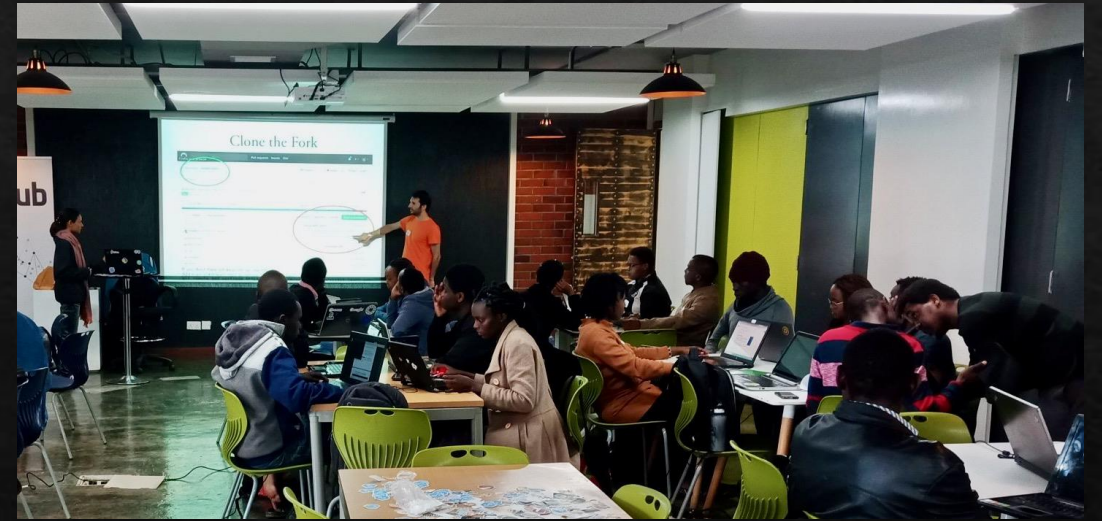
The video player shows a live-action shot of Andreas Müller on the left and a presentation slide on the right. The slide features the Data Umbrella logo (a colorful umbrella) and the text: "Data Umbrella", "Crash-Course in Contributing to Open Source Projects", and "Andreas Müller (and contributions from: Reshama Shaikh)". The video player interface includes a progress bar at 0:07 / 30:28, a video title "Scikit-learn sprint instructions", view count "1,159 views • May 25, 2020", and a channel name "Andreas Mueller" with "7.03K subscribers".



The video player shows a presentation slide with the title "Example of Contributing to Scikit-learn Library A Documentation Update". The slide includes the Scikit-learn logo, the name "Reshama Shaikh", and her Twitter handle "@reshamas". The Data Umbrella logo is in the bottom right corner. The video player interface shows a progress bar at 0:01 / 26:35, a video title "Reshama Shaikh: Contributing to Scikit-Learn: An Example Pull Request", view count "547 views • May 28, 2020", and a channel name "Data Umbrella" with "233 subscribers".

Pair Programming

- ◆ Get more done (or at least better)!
- ◆ Have more fun!
- ◆ Learn more!



Pair Programming: TIPS

- ◆ Be patient with each other
- ◆ Talk through your thoughts and every step
- ◆ Let the less familiar person drive

See https://medium.com/@weblab_tech/pair-programming-guide-a76ca43ff389

Navigating the Code

- ◆ Find the code
- ◆ Find the test
- ◆ Find the documentation
- ◆ (Find the examples)

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

Navigating the Code [just notes]

Basics of Sphinx

sphinx-doc.org

- ◆ Grabs the docstrings from Python files
- ◆ Dedicated documentation (= user guide) written in RST
- ◆ On-demand generation of html

Updating your branch(es) "sync" repo

```
git checkout feature_branch
```

```
git pull upstream master
```

<potentially resolve conflicts>

```
git push origin feature_branch
```


Branches and your second pull request (PR)

- ◆ Create a new branch for every new feature
- ◆ Make sure you switch back to master and updating before making a new branch:

```
git checkout master
```






```
git checkout -b another_feature_branch
```

- ◆ Messed up?

```
git reset --soft upstream/master
```

Data Scientists: Can also contribute to other libraries

<https://www.youtube.com/c/dataumbrella>

- 1  **Matti Picus: Getting Involved with NumPy**
Data Umbrella
57:10
- 2  **Reshama Shaikh: Contributing to Scikit-Learn: An Example Pull Request**
Data Umbrella
26:36
- 3  **Scikit-learn sprint instructions**
Andreas Mueller
30:29
- 4  **Marco Gorelli: Contributing to pandas**
Data Umbrella
32:57
- 5  **Carol Willing: Contributing to Core Python**
Data Umbrella
55:58