

 100 XP

# Identify where you can help

5 minutes

In this unit, we'll discuss how you can get started with open-source contributions. We'll also use issues and tags to find tasks to contribute to.

**Open-source** software can be freely used, modified, and shared by anyone. Using open-source software, anyone can view, modify, and distribute a project for any purpose. The idea behind open-source software is that sharing code leads to better, more reliable software.

There are many ways to contribute to open-source projects. Making your first contribution can often be a scary experience, but it shouldn't be. Open source is a place for everyone, and contributions happen at all levels.

## Find an open-source project that needs contributions

You can get started by thinking about the projects you already use, or want to use. Contributing is easier when you're familiar with the project and its community.

Perhaps while reading a project's README file, you find a broken link or some typos. Maybe you noticed something isn't working as expected, or the documentation is out of date. These are all great opportunities to help and contribute to the project.

### Tip

One important tip: *All* kinds of contributions are valuable. Your level of experience or knowledge of the project doesn't matter here. We all have something we can contribute. Be confident in yourself. The most important thing here is the will to help.

## Use GitHub search

You can also use GitHub search to explore topics and related projects. Head to [GitHub search](#), and enter your topic word.

Let's say you're interested in machine learning.

Q Search more than 295M repositories

machine-learning

Search

ProTip! For an advanced search, use some of our prefixes.

You can then narrow your search by selecting **Topics** in the left sidebar.

Repositories386K

Code10M

Commits310K+

Issues254K

Discussions1K

Packages76

Marketplace15

Topics494

Wikis36K

Users62K

Languages

Jupyter Notebook	161,053
Python	80,020
HTML	25,388
MATLAB	18,530
R	8,589
Java	5,113
JavaScript	5,046
C++	3,009
C#	1,937
TeX	1,616

Machine learning

Machine learning is a way of modeling and interpreting data that allows a piece of software to respond intelligently.

See topic

386,222 repository results

Sort: Best match

username/ awesome-machine-learning

A curated list of awesome Machine Learning frameworks, libraries and software.

☆ 53.7k Python Updated 4 days ago

user/ MachineLearning

Basic Machine Learning and Deep Learning

☆ 4.3k Python Updated on 4 Oct 2021

user-2/ machine-learning

Public archive

Content for Udacity's Machine Learning curriculum

☆ 3.7k Jupyter Notebook Updated 26 days ago

user-3456/ Machine-Learning

机器学习实战 (Python3) : kNN, 决策树, 贝叶斯, 逻辑回归, SVM, 线性回归, 树回归

machine-learning python svm regression logistic python3 adaboost smo knn decision-tree

navie-bayes-algorithm adaboost-algorithm

☆ 5.9k Python Updated on 7 Jul 2021

user-7/ MachineLearning\_Python

机器学习算法python实现

☆ 4.6k Python MIT license Updated on 16 Dec 2020

Sponsor

Advanced search

Cheat sheet

From there, you can find repositories relevant to your search keyword and repositories curated by community members.

## Familiarize yourself with an open-source project

Something worth mentioning here is that every open-source community is different. After you've found a project, you'll need to familiarize yourself with the project and its participation guidelines.

Most projects will have these documents at the top level of the repository:

- **LICENSE:** The project must contain an [open-source license](#). If the project doesn't have a license, it's not open source.
- **README:** The README file usually serves as the welcome page for the project. It generally provides information on how to get started using the project. It's also common for it to add information on how to engage with the community.
- **CONTRIBUTING:** As its name suggests, this document provides guidance on how to contribute to the project. It usually describes how the contribution process works, and includes details on how to set up your development environment.
- **CODE\_OF\_CONDUCT:** The code of conduct sets ground rules for community members. By doing so, it helps make the community a safe and welcoming environment for all.

Although not all projects have CONTRIBUTING or CODE\_OF\_CONDUCT documents, having these documents is a good indication of how friendly and welcoming a project is.

Open-source contributors and maintainers come from all over the world. Projects usually have multiple communication channels to organize discussions and ask for help. A good way to familiarize yourself with the community is by reading through some of these communication channels:

- **Issue tracker:** Where folks discuss issues and tasks related to the project. To find the issues in GitHub, you can go to the main page of the repository on GitHub and add `/issues` to the end of the URL, for example: <https://github.com/jupyter/notebook/issues>.
- **Pull request:** Where folks discuss and review changes to the project. You can find it in GitHub by adding `pulls` to the project's URL, for example, <https://github.com/jupyter/notebook/pulls>.
- **Chat channels and forums:** Some projects use chat channels, such as Slack, Gitter, and IRC, or forums like Discourse for conversations and discussions.

## Identify tasks to work on

You've found a project, you've read the contribution guidelines, and now you're ready to contribute.

Perhaps you've already identified something to work on, such as fixing broken links or updating the docs. A good way to find beginner-friendly issues to help with is by visiting the project's `/contribute` URL, for example: <https://github.com/jupyter/notebook/contribute>.

jupyter / notebook Public

<> Code Issues 2k Pull requests 49 Actions Projects 8 Security 3 Insights

## Contribute to jupyter/notebook

Make your first contribution to this repository by tackling one of the issues listed below.

Each issue displayed here is a "good first issue," selected for its relative approachability for first-time contributors.

[Read the contributing guidelines](#)  
notebook / CONTRIBUTING.md

### Good first issues

<p>🕒 Add more Prometheus metrics <span>help wanted</span> <span>good first issue</span></p> <p>#3682 opened on 13 Jun 2018 by &lt;GitHub-name&gt;</p>	27
<p>🕒 Convert JS tests to Selenium <span>enhancement</span> <span>help wanted</span> <span>good first issue</span></p> <p>#3335 opened on 13 Feb 2018 by &lt;GitHub-name&gt;</p>	64
<p>🕒 Wishlist: Usability of new "Find and Replace" dialog for finding strings <span>enhancement</span> <span>good first issue</span></p> <p>#1238 opened on 21 Mar 2016 by &lt;GitHub-name&gt;</p>	15
<p>🕒 Highlight TODO comments <span>enhancement</span> <span>tag:URAP</span> <span>good first issue</span></p> <p>#1092 opened on 15 Feb 2016 by &lt;GitHub-name&gt;</p>	19
<p>🕒 Closing many notebooks at once <span>enhancement</span> <span>good first issue</span></p> <p>#970 opened on 18 Jan 2016 by &lt;GitHub-name&gt;</p>	26
<p>🕒 Review docs on troubleshooting and correcting a stale JS build <span>documentation</span> <span>status:Work in Progress</span> <span>good first issue</span></p> <p>#872 opened on 15 Dec 2015 by &lt;GitHub-name&gt;</p>	12
<p>🕒 Log on disk path of files served when using --debug <span>good first issue</span></p> <p>#857 opened on 13 Dec 2015 by &lt;GitHub-name&gt;</p>	10
<p>🕒 List of notebook files should include more file information <span>component:File Browser</span> <span>good first issue</span></p> <p>#396 opened on 5 Sep 2015 by &lt;GitHub-name&gt;</p>	15

You'll notice that most of the issues displayed in the `contribute` URL will have labels such as `good-first-issue`, `help wanted`, `beginner-friendly`, and so on. Labels are often used to provide top-level information of the issue and the type of help needed.

You can head to the labels page, for example: <https://github.com/jupyter/notebook/labels>.

Then, select issues that have labels like `help wanted`, `discussion`, or other labels relevant to the type of contribution in which you're interested.

As you explore issues, you might also notice that some have other issues or pull requests linked.

## [Bug]: Confusing deprecation warning when empty data passed to axis with category units #22640

New issue

Open

&lt;name&gt; opened this issue 20 days ago · 2 comments

May be fixed by #22719



&lt;name&gt; commented 20 days ago • edited



### Bug summary

I'm seeing a `MatplotlibDeprecationWarning` when using calling axes methods on empty data structures for axes that are using string unit converters. I think this is either a false alarm or a non-actionable warning.

### Code for reproduction

```
import matplotlib.pyplot as plt
f, ax = plt.subplots()
ax.xaxis.update_units(["a", "b"])
ax.plot([], [])
```

### Actual outcome

`MatplotlibDeprecationWarning`: Support for passing numbers through unit converters is deprecated since 3.5 and support will be removed two minor releases later; use `Axis.convert_units` instead.

```
ax.plot([], [])
```

#### Assignees

No one assigned

#### Labels

None yet

#### Projects

None yet

#### Milestone

No milestone

#### Development

Successfully merging a pull request may close this issue.

**Fix incorrect deprecation warning**  
<name> /matplotlib

## Sponsor a project

There are many ways to contribute to open source. You can financially support the folks who build and maintain the open-source ecosystem through code, leadership, mentorship, design, and beyond.

Open source heavily relies on volunteer work. GitHub Sponsors allow you to fund projects and individuals to help them keep doing their open-source work, while giving them the recognition they deserve.

If a project is eligible for sponsorship through GitHub Sponsors, you'll find a **Sponsor** button on the project's main page.

matplotlib / matplotlib Public

Sponsor Watch 576 Fork 6.3k Star 15.3k

Code Issues 1.5k Pull requests 321 Actions Projects 7 Wiki Security Insights

main 19 branches 108 tags Go to file Add file Code

Merge pull request #22752 from <other-name> /docs/fix-in... 162e306 11 hours ago 42,500 commits

.circleci	Disallow setuputils 60.6.0 on CI	2 months ago
.github	CI: Add nightly upload of wheels to Anaconda Cloud	yesterday
LICENSE	Improve the Type-1 font parsing	7 months ago
ci	Move setup.cfg to mplsetup.cfg.	7 months ago
doc	DOC: Correct nightly wheels pip install command	14 hours ago
examples	Small style fixes.	3 days ago
extern	Add protection against out-of-bounds read in ttconv	9 months ago
lib	Merge pull request #22742 from QuLogic/fix-deprecation	18 hours ago
plot_types	DOC: remove sample_plots from tutorials	5 months ago
requirements	Require sphinx>=3 & numpydoc>=1.0 for building docs.	last month
src	Fix new leak in ft2font introduced in #22604	18 days ago
tools	Properly capitalize "Unicode".	2 months ago
tutorials	FIX: ax	24 days ago
.appveyor.yml	Drop support for Python 3.7	2 months ago
.coveragerc	Add default exclusion to coverage exclude_lines.	2 years ago
.flake8	Linewrap setupext to 79 character lines.	last month
.git_archival.txt	Switch to setuputils_scm.	12 months ago
.gitattributes	Update gitattributes so that objc diffs are correctly contextualized.	4 months ago

About matplotlib: plotting with Python

matplotlib.org/stable

python gtk data-science qt data-visualization tk matplotlib plotting hacktoberfest wx

Readme Code of conduct Cite this repository 15.3k stars 576 watching 6.3k forks

Releases 62

REL: v3.5.1 Latest on 11 Dec 2021

+ 61 releases

Sponsor this project

matplotlib Matplotlib Develop... Sponsor

<https://numfocus.org/donate-to-matpl...>

[Learn more about GitHub Sponsors](#)

You can select the sponsorship tier and if you want your contribution to be public.

### Select a tier

\$10 a month

Select

We really appreciate your help! Your donation will pay for some of our infrastructure costs like cloud credits and website maintenance.

You will receive a Sponsor badge 🏆 on your profile.

\$25 a month

Select

Thank you for your donation which will help fund critical documentation updates and library maintenance.

You will receive a Sponsor badge 🏆 on your profile.

## Unit recap

In this unit, you learned how to get started with open-source contributions. You now know how to choose a project to work on and use GitHub issues and labels to identify tasks to work on.

Here's a handy checklist to use when you interact with a project for the first time:

- Does it have a license?
- Are issues and pull request discussions used actively by maintainers and contributors?
- Does the project use labels like `help wanted` or `good first issue` for newcomers?
- Does the project have a code of conduct?
- Does the project have clear Contributing Guidelines?

Finally, remember that all contributions are welcome, and the open-source ecosystem greatly benefits from your ideas and participation. There are many ways to contribute to open source, from submitting code or engaging in project discussions to sponsoring projects through GitHub Sponsors.

---

## Next unit: Contribute to an open-source repository

Continue >

---