# Contributing

Contributions of all kinds are welcome. In particular pull requests are appreciated.

The authors will endeavour to help walk you through any issues in the pull request

discussion, so please feel free to open a pull request even if you are new to such things.

## Issues

The easiest contribution to make is to [file an issue](https://github.com/lmcinnes/umap/issues/new).

It is beneficial if you check the [FAQ](https://umap-learn.readthedocs.io/en/latest/faq.html),

and do a cursory search of [existing issues](https://github.com/lmcinnes/umap/issues?utf8=%E2%9C%93&q=is%3Aissue).

It is also helpful, but not necessary, if you can provide clear instruction for

how to reproduce a problem. If you have resolved an issue yourself please consider

contributing to the FAQ to add your problem, and its resolution, so others can

benefit from your work.

## Documentation

Contributing to documentation is the easiest way to get started. Providing simple

clear or helpful documentation for new users is critical. Anything that \*you\* as

a new user found hard to understand, or difficult to work out, are excellent places

to begin. Contributions to more detailed and descriptive error messages is

especially appreciated. To contribute to the documentation please

[fork the project](https://github.com/lmcinnes/umap/issues#fork-destination-box)

into your own repository, make changes there, and then submit a pull request.

## Code

Code contributions are always welcome, from simple bug fixes, to new features. To

contribute code please

[fork the project](https://github.com/lmcinnes/umap/issues#fork-destination-box)

into your own repository, make changes there, and then submit a pull request. If

you are fixing a known issue please add the issue number to the PR message. If you

are fixing a new issue feel free to file an issue and then reference it in the PR.

You can [browse open issues](https://github.com/lmcinnes/umap/issues),

or consult the [project roadmap](https://github.com/lmcinnes/umap/issues/15), for potential code

contributions. Fixes for issues tagged with 'help wanted' are especially appreciated.

### Code formatting

If possible, install the [black code formatter](https://github.com/python/black) (e.g.

`pip install black`) and run it before submitting a pull request. This helps maintain consistency

across the code, but also there is a check in the Travis-CI continuous integration system which

will show up as a failure in the pull request if `black` detects that it hasn't been run.

Formatting is as simple as running:

```bash

black .

```

in the root of the project.