

---

# **DataStore Documentation**

*Release 0.2.2*

**Todd Young**

**Feb 18, 2020**



---

# Contents

---

<b>1</b>	<b>Installation</b>	<b>1</b>
1.1	Stable release . . . . .	1
1.2	From sources . . . . .	1
<b>2</b>	<b>Usage</b>	<b>3</b>
<b>3</b>	<b>Contributing</b>	<b>5</b>
3.1	Types of Contributions . . . . .	5
3.2	Get Started! . . . . .	6
3.3	Pull Request Guidelines . . . . .	6
3.4	Tips . . . . .	7
<b>4</b>	<b>Credits</b>	<b>9</b>
4.1	Development Lead . . . . .	9
4.2	Contributors . . . . .	9
<b>5</b>	<b>History</b>	<b>11</b>
5.1	0.1.0 (2019-22-07) . . . . .	11
5.2	0.2.0 (2019-25-07) . . . . .	11
<b>6</b>	<b>Feedback</b>	<b>13</b>



### 1.1 Stable release

To install datastore, run this command in your terminal:

```
pip install datastore
```

This is the preferred method to install datastore, as it will always install the most recent stable release.

If you don't have `pip` installed, this [Python installation guide](#) can guide you through the process.

### 1.2 From sources

The sources for datastore can be downloaded from the [Github repo](#).

You can either clone the public repository:

```
git clone git://github.com/yngtodd/datastore
```

Or download the [tarball](#):

```
curl -OL https://github.com/yngtodd/datastore/tarball/master
```

Once you have a copy of the source, you can install it with:

```
python setup.py install
```



## CHAPTER 2

---

### Usage

---

To use DataStore in a project:

```
import datastore
```





Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given. You can contribute in many ways:

### 3.1 Types of Contributions

#### 3.1.1 Report Bugs

Report bugs at <https://github.com/yngtodd/datastore/issues>.

If you are reporting a bug, please include:

- Your operating system name and version.
- Any details about your local setup that might be helpful in troubleshooting.
- Detailed steps to reproduce the bug.

#### 3.1.2 Fix Bugs

Look through the GitHub issues for bugs. Anything tagged with “bug” is open to whoever wants to implement it.

#### 3.1.3 Implement Features

Look through the GitHub issues for features. Anything tagged with “feature” is open to whoever wants to implement it.

### 3.1.4 Write Documentation

DataStore could always use more documentation, whether as part of the official DataStore docs, in docstrings, or even on the web in blog posts, articles, and such.

### 3.1.5 Submit Feedback

The best way to send feedback is to file an issue at <https://github.com/yngtodd/datastore/issues>.

If you are proposing a feature:

- Explain in detail how it would work.
- Keep the scope as narrow as possible, to make it easier to implement.
- Remember that this is a volunteer-driven project, and that contributions are welcome :)

## 3.2 Get Started!

Ready to contribute? Here's how to set up *datastore* for local development.

1. Fork the *datastore* repo on GitHub.
2. Clone your fork locally:

```
git clone git@github.com:your_name_here/datastore.git
```

3. Create a branch for local development:

```
git checkout -b name-of-your-bugfix-or-feature
```

Now you can make your changes locally.

4. When you're done making changes, check that your changes pass style and unit tests, including testing other Python versions with tox:

```
tox
```

To get tox, just pip install it.

5. Commit your changes and push your branch to GitHub:

```
git add .
git commit -m "Your detailed description of your changes."
git push origin name-of-your-bugfix-or-feature
```

6. Submit a pull request through the GitHub website.

## 3.3 Pull Request Guidelines

Before you submit a pull request, check that it meets these guidelines:

1. The pull request should include tests.
2. If the pull request adds functionality, the docs should be updated. Put your new functionality into a function with a docstring, and add the feature to the list in README.rst.

3. The pull request should work for Python  $\geq 3.6$  and for PyPy. Check <https://travis-ci.org/yngtodd/datastore> under pull requests for active pull requests or run the `tox` command and make sure that the tests pass for all supported Python versions.

## 3.4 Tips

To run a subset of tests:

```
py.test test/test_datastore.py
```



### 4.1 Development Lead

- Todd Young GitHub: [yngtodd](#)

### 4.2 Contributors

None yet. Why not be the first?



### 5.1 0.1.0 (2019-22-07)

- First release on PyPI.

### 5.2 0.2.0 (2019-25-07)

#### **Support for multitask datasets**

1. Adds meta and abstract classes for multitask data
2. Adds stratified cross-validation method for multitask data





## CHAPTER 6

---

### Feedback

---

If you have any suggestions or questions about **DataStore** feel free to email me at [youngmt1@ornl.gov](mailto:youngmt1@ornl.gov).

If you encounter any errors or problems with **DataStore**, please let me know! Open an Issue at the GitHub <http://github.com/yngtodd/datastore> main repository.