fastatools Documentation

Release 1.2.0

Steve Davis

Feb 13, 2019

Contents

Contents:

fastatools

Tools for working with fasta files.

The fastatools package was developed by the United States Food and Drug Administration, Center for Food Safety and Applied Nutrition.

- Free software
- Documentation: https://fastatools.readthedocs.io
- Source Code: https://github.com/CFSAN-Biostatistics/fastatools
- PyPI Distribution: https://pypi.python.org/pypi/fastatools

1.1 Features

- Print the lengths of sequences.
- Determine if two fasta files are equivalent.
- Fix inconsistent line lengths and inconsistent lowercase.
- Generate a reverse complement of a fasta file.
- Extract a slice from a fasta file delimited by primers.
- Extract a range of positions from a fasta file.

1.2 Citing fastatools

To cite fastatools, please reference the fastatools GitHub repository:

https://github.com/CFSAN-Biostatistics/fastatools

1.3 License

See the LICENSE file included in the fastatools distribution.

CHAPTER 2

Installation

At the command line:

```
$ pip install --user fastatools
```

Update your .bashrc file with the path to user-installed python packages:

```
export PATH=~/.local/bin:$PATH
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv fastatools
$ pip install fastatools
```

2.1 Upgrading fastatools

If you previously installed with pip, you can upgrade to the newest version from the command line:

```
$ pip install --user --upgrade fastatools
```

2.2 Uninstalling fastatools

If you installed with pip, you can uninstall from the command line:

```
$ pip uninstall fastatools
```

Usage

To print the lengths of sequences:

```
fastatools length FILE [FILE ...]
```

To determine if two fasta files are equivalent, ignoring the line lengths, uppercase / lowercase characters, and sequence order:

```
fastatools equiv [--order] [--ignore_defline] FILE1 FILE2
```

To fix inconsistent line lengths and inconsistent lowercase:

```
fastatools rewrite --upper FILE
```

To generate a reverse complement of a fasta file:

```
fastatools reverse FILE
```

To extract a slice from a fasta file delimited by primers:

```
fastatools between --no_rev_comp FWD REV FILE
```

To extract a range of positions from a fasta file:

```
fastatools range id start end FILE
```

Contributing

Contributions are welcome, and they are greatly appreciated! Every little bit helps, and credit will always be given.

You can contribute in many ways:

4.1 Types of Contributions

4.1.1 Report Bugs

Report bugs at https://github.com/CFSAN-Biostatistics/fastatools/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.

4.1.2 Fix Bugs

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

4.1.3 Implement Features

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

4.1.4 Write Documentation

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

4.1.5 Submit Feedback

The best way to send feedback is to file an issue at https://github.com/CFSAN-Biostatistics/fastatools/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 :)

4.2 Get Started!

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

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

```
$ git clone git@github.com:your_name_here/fastatools.git
```

3. Install your local copy into a virtualenv. Assuming you have virtualenvwrapper installed, this is how you set up your fork for local development:

```
$ mkvirtualenv fastatools
$ cd fastatools/
$ pip install sphinx_rtd_theme    # the documentation uses the ReadTheDocs theme
$ pip install pytest
$ python setup.py develop
```

4. Create a branch for local development:

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

Now you can make your changes locally.

5. When you're done making changes, check that your changes pass flake8 and the tests, including testing other Python versions with tox:

```
$ flake8 fastatools tests
$ pytest -v
$ tox
```

To get flake8 and tox, just pip install them into your virtualenv.

6. Update the documentation and review the changes locally with sphinx:

```
$ cd docs
$ sphinx-build -b html . ./_build
$ xdg-open _build/index.html
```

7. 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
```

8. Submit a pull request through the GitHub website.

4.3 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 2.7, 3.4, 3.5, 3.6, and for PyPy.

4.4 Tips

To run a subset of tests:

```
$ pytest -v tests/test_fastatools.py
```

Credits

5.1 Development Lead

- Steve Davis

5.2 CFSAN Bioinformatics Team

- Steve Davis

5.3 External Contributors

None yet. Why not be the first?

CHAPTER 6

History

1.2.0 (2018-12-10)

- First public release with docs and automated tests.

CHAPTER 8

1.1.0 (2018-10-25)

- Add range command to extract a range of positions from a fasta file.
- Add equiv command to check for equivalent fasta files.
- Add rewrite command to fix line length and uppercase/lowercase inconsistencies.

CHAPTER 9

1.0.0 (2018-09-21)

- Add length command to print the length of all the sequences in fasta files.

CHAPTER 10

0.1.0 (2017-11-30)

- Project started.

CHAPTER 11

Indices and tables

- genindex
- modindex
- search