refchooser Documentation

Release 0.2.1

Steve Davis

Aug 20, 2019

Contents

Contents:

CHAPTER 1

refchooser

Tools to help choose a reference from a list of assemblies.

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

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

1.1 Features

- Print a table of metrics to help choose an assembly from a collection.
- N50
- N90
- Number of contigs
- Assembly length
- Mean mash distance to all other assemblies

1.2 Citing refchooser

To cite refchooser, please reference the refchooser GitHub repository:

> https://github.com/CFSAN-Biostatistics/refchooser

1.3 License

See the LICENSE file included in the refchooser distribution.

CHAPTER 2

Installation

2.1 Executable Software Dependencies

You should install `Mash` before installing refchooser.

https://github.com/marbl/Mash/releases

2.2 Installing refchooser

At the command line:

```
$ pip install --user refchooser
```

Or, if you have virtualenvwrapper installed:

```
$ mkvirtualenv refchooser
$ pip install refchooser
```

2.3 Upgrading refchooser

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

```
$ pip install --user --upgrade refchooser
```

2.4 Uninstalling refchooser

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

```
$ pip uninstall refchooser
```

CHAPTER 3

Usage

You can use refchooser to select a good reference from a list of assemblies. You will need either a directory of assemblies or a file containing the paths to the assemblies. Refchooser prints a table of metrics for each assembly.

The captured metrics are:

- N50
- N90
- Number of contigs
- Assembly length
- Mean mash distance to all other assemblies

The results can be sorted by any metric you choose. By default, the assemblies are sorted by a simple score which is the N50/Distance ratio.

To print the table of assemblies sorted by N50:

```
# Choose the top 10 from a collection of 900 assemblies
refchooser metrics --sort N50 --top 10 assembly_paths.txt sketch_directory

Assembly   N50    N90    Contigs Length  Mean_Distance Path                    Score
SRR5868281 791291 119061 30      4845891 7.045173e-04  fasta/SRR5868281.fasta  1.
↪123168e+09
SRR7439260 775386 146629 24      4815254 4.927176e-04  fasta/SRR7439260.fasta  1.
↪573693e+09
SRR7906469 775033 432700 18      4779049 6.352714e-04  fasta/SRR7906469.fasta  1.
↪220003e+09
SRR6949545 774499 146519 21      4814882 5.308503e-04  fasta/SRR6949545.fasta  1.
↪458978e+09
SRR6949610 774132 105140 33      4888983 8.929775e-04  fasta/SRR6949610.fasta  8.
↪669110e+08
SRR7426190 774120 146629 30      4820457 5.317999e-04  fasta/SRR7426190.fasta  1.
↪455660e+09
SRR7426155 774120 146449 29      4775352 6.618484e-04  fasta/SRR7426155.fasta  1.
↪169633e+09
```

(continues on next page)

(continued from previous page)

```
SRR7441818 774120 146519 25      4797608 5.506614e-04  fasta/SRR7441818.fasta  1.
↪405800e+09
SRR7439259 774120 146629 25      4815750 4.911346e-04  fasta/SRR7439259.fasta  1.
↪576187e+09
SRR7439242 774120 146519 32      4803747 5.681594e-04  fasta/SRR7439242.fasta  1.
↪362505e+09
```

To print the table of assemblies sorted by mean mash distance:

```
# Choose the top 10 from a collection of 900 assemblies
refchooser metrics --sort Mean_Distance --top 10 assembly_paths.txt sketch_directory

Assembly   N50    N90    Contigs Length  Mean_Distance Path                    Score
SRR1645597 226490 55522  55      4803421 4.611227e-04  fasta/SRR1645597.fasta  4.
↪911708e+08
SRR1965968 237440 55508  59      4804728 4.614244e-04  fasta/SRR1965968.fasta  5.
↪145805e+08
SRR1963305 166064 47353  61      4804588 4.618624e-04  fasta/SRR1963305.fasta  3.
↪595530e+08
SRR1646405 226711 56774  58      4800826 4.629222e-04  fasta/SRR1646405.fasta  4.
↪897389e+08
SRR1967694 287598 63327  54      4800251 4.637451e-04  fasta/SRR1967694.fasta  6.
↪201639e+08
SRR7458586 216691 68846  48      4802064 4.642351e-04  fasta/SRR7458586.fasta  4.
↪667700e+08
SRR7439539 333102 76943  45      4796679 4.646953e-04  fasta/SRR7439539.fasta  7.
↪168180e+08
SRR5584738 216691 54594  62      4797573 4.649960e-04  fasta/SRR5584738.fasta  4.
↪660061e+08
SRR7439240 774109 146629 34      4814374 4.658887e-04  fasta/SRR7439240.fasta  1.
↪661575e+09
SRR8691682 216324 75764  47      4795530 4.659022e-04  fasta/SRR8691682.fasta  4.
↪643121e+08
```

To print the table of assemblies sorted by the N50/Mean_Distance ratio score:

```
# Choose the top 10 from a collection of 900 assemblies
refchooser metrics --top 10 assembly_paths.txt sketch_directory

Assembly   N50    N90    Contigs Length  Mean_Distance Path                    Score
SRR7439240 774109 146629 34      4814374 4.658887e-04  fasta/SRR7439240.fasta  1.
↪661575e+09
SRR7439252 774092 146519 26      4810069 4.722259e-04  fasta/SRR7439252.fasta  1.
↪639241e+09
SRR5237981 749843 146968 30      4811975 4.681156e-04  fasta/SRR5237981.fasta  1.
↪601833e+09
SRR7439259 774120 146629 25      4815750 4.911346e-04  fasta/SRR7439259.fasta  1.
↪576187e+09
SRR7439260 775386 146629 24      4815254 4.927176e-04  fasta/SRR7439260.fasta  1.
↪573693e+09
SRR7140222 773996 105140 27      4810707 4.939575e-04  fasta/SRR7140222.fasta  1.
↪566928e+09
SRR7426191 774120 146629 28      4813941 5.066112e-04  fasta/SRR7426191.fasta  1.
↪528036e+09
SRR7347002 774109 146519 34      4822419 5.137192e-04  fasta/SRR7347002.fasta  1.
↪506872e+09
SRR1793292 774006 119046 56      4833765 5.247949e-04  fasta/SRR1793292.fasta  1.
↪474873e+09
```

(continues on next page)

(continued from previous page)

```
SRR6945020 774120 146519 21      4813673 5.303144e-04  fasta/SRR6945020.fasta  1.
↪459738e+09
```

CHAPTER 4

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/refchooser/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

The refchooser project could always use more documentation, whether as part of the official refchooser 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/refchooser/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 *refchooser* for local development.

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

```
$ git clone git@github.com:your_name_here/refchooser.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 refchooser
$ cd refchooser/
$ 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 refchooser 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:

```
$ make docs
```

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 3.7.

4.4 Tips

To run a subset of tests:

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

CHAPTER 5

Credits

5.1 Development Lead

- Steve Davis <steven.davis@fda.hhs.gov>

5.2 CFSAN Bioinformatics Team

- Steve Davis <steven.davis@fda.hhs.gov>

5.3 External Contributors

None yet. Why not be the first?

5.4 Acknowledgments

The refchooser project was inspired by and makes use of the following prior works:

Genomic diversity affects the accuracy of bacterial SNP calling pipelines Stephen J. Bush, Dona Foster, David W. Eyre, Emily L. Clark, Nicola De Maio, Liam P. Shaw, Nicole Stoesser, Tim E. A. Peto, Derrick W. Crook, A. Sarah Walker bioRxiv 653774; doi: https://doi.org/10.1101/653774

Ondov, Brian D., et al. "Mash: fast genome and metagenome distance estimation using MinHash." Genome biology 17.1 (2016): 132.

CHAPTER 6

History

6.1 0.2.1 (2019-08-20)

- Detect and exclude empty assemblies.

6.2 0.2.0 (2019-07-12)

- Add N50 and N90 metrics.
- Calculate a simple “Score” for each assembly, the N50/Distance ratio.
- Combine all metrics in a single table sortable by any column.

6.3 0.1.0 (2019-06-21)

- Initial version.

CHAPTER 7

Indices and tables

- genindex
- search