
SeqFindr Documentation

Release 0.35.0

Mitchell Stanton-Cook, Nabil Alikhan & Hamza Khan

January 08, 2017

1	User/Quickstart/Tutorial Documentation	3
2	API Documentation	5
3	Developer Documentation	7
4	Indices and tables	9

SeqFindr - easily create informative genomic feature plots.

User/Quickstart/Tutorial Documentation

See on [SeqFindr GitHub site](#).

API Documentation

Explore the available methods of SeqFindr

2.1 SeqFindr

2.1.1 SeqFindr package

Submodules

SeqFindr.blast module

SeqFindr.config module

SeqFindr.imaging module

SeqFindr.seqfindr module

SeqFindr.util module

SeqFindr.vfdb_to_seqfindr module

Module contents

Developer Documentation

Learn how to contribute to the development of SeqFindr.

3.1 SeqFindr Developer HOWTO

In addition to what is described here, [this document by Jeff Forcier](#) and [this talk from Carl Meyer](#) provide wonderful footings for developing on/in open source projects.

3.1.1 Maintaining a consistent development environment

1) Ensure all development is performed within a virtualenv. A good way to bootstrap this is via [virtualenv-burrito](#).

Execute the installation using:

```
$ curl -sL https://raw.githubusercontent.com/brainsik/virtualenv-burrito/master/virtualenv-burrito.sh
```

2) Make a virtualenv called SeqFindr:

```
$ mkvirtualenv SeqFindr
```

3) Install `autoenv`:

```
$ git clone git://github.com/kennethreitz/autoenv.git ~/.autoenv
$ echo 'source ~/.autoenv/activate.sh' >> ~/.bashrc
```

3.1.2 Get the current code from GitHub

Something like this:

```
$ cd $PATH_WHERE_I_KEEP_MY_REPOS
$ git clone https://github.com/mscook/SeqFindr.git
```

3.1.3 Install dependencies

Something like this:

```
$ cd SeqFindr
$ # Assuming you installed autoenv -
$ # You'll want to say 'y' as this will activate the virtualenv each time you enter the code director
$ # Otherwise -
$ # workon SeqFindr
$ pip install -r requirements.txt
$ pip install -r requirements-dev.txt
```

3.1.4 Familiarise yourself with the code

TODO.

3.1.5 Development workflow

Use GitHub. You will have already cloned the SeqFindr repo (if you followed instructions above). To make things easier, please fork (<https://github.com/mscook/SeqFindr/fork>) and update your local copy to point to your fork.

Something like this:

```
$ # Assuming your fork is like this
$ # https://github.com/$YOUR_USERNAME/SeqFindr/
$ vi .git/config
$ # Replace:
$ # url = git@github.com:mscook/SeqFindr.git
$ # with:
$ # url = git@github.com:$YOUR_USERNAME/SeqFindr.git
```

With this setup you will be able to push development changes to your fork and submit Pull Requests to the core SeqFindr repo when you're happy.

Important Note: Upstream changes will not be synced to your fork by default. Please, before submitting a pull request please sync your fork with any upstream changes (specifically handle any merge conflicts). Info on syncing a fork can be found [here](#).

3.1.6 Code style/testing/Continuous Integration

We try to make joining and/or modifying the SeqFindr project simple.

General:

- As close to PEP8 as possible but I ain't no Saint. Just as long as it's clean and readable,
- Using standard lib unittest. There are convenience functions `check_coverage.sh` & `tests/run_tests.sh` respectively. We would prefer SMART test vs 100 % coverage.

In the master GitHub repository we use hooks that call:

- `landscape.io` (code QC)
- Travis CI (continuous integration)
- ReadTheDocs (documentation building)

Indices and tables

- `genindex`
- `modindex`
- `search`