
GDSCtools

Release 0.9.1

Thomas Cokelaer

May 26, 2016

1	Overview	3
2	Installation	5
3	User Guide	7
4	Developers Guide	9
5	Issues	11
6	Contributions	13

Contents

- *GDSCtools*
 - *Overview*
 - *Installation*
 - * *You know Python and/or **pip** utility ?*
 - * *You don't know Python or have compilation issues ?*
 - * *Testing your installation*
 - *User Guide*
 - *Developers Guide*
 - *Issues*
 - *Contributions*

Overview

GDSCTools contains utilities used to find significant associations between drug response and genomic features.

Installation

2.1 You know Python and/or pip utility ?

Releases of **GDSCTools** are available on [Pypi](#). Therefore **GDSCTools** can be installed in a shell using **pip**:

```
pip install gdsctools
```

Dependencies (e.g., Pandas, matplotlib) should be taken care of automatically.

2.2 You don't know Python or have compilation issues ?

If you are not familiar with Python, or have issues with compilation, or do not have root access, we would recommend to use the [Anaconda](#) solution. It will install binary packages required to install **GDSCTools** itself; since it does not require root access, it should not interfere with your system.

Please, visit the link above and install Anaconda following the instructions.

Once you have installed Anaconda, open a new shell.

As a **developer**, you can get the latest source code from github and install GDSCTools from source as follows (in a shell)

```
# go in a working directory and type:
git clone https://github.com/CancerRxGene/gdsctools
cd gdsctools
python setup.py install
```

As an **end-user**, open a shell and type. This will install the latest official release of **GDSCTools**:

```
pip install gdsctools
```

2.3 Testing your installation

You should now be ready to use **GDSCTools**. A good test is to check that the following executable is available. In a shell, type:

```
gdsctools_anova --test
```

or

```
gdsc tools_anova --help
```

or for developers, starts an IPython shell, and type e.g.:

```
from gdsc tools import *  
an = ANOVA(ic50_test)
```

Please, see the quickstart session for the usage.

3.1 Quick Start

3.1.1 Data format

You will need several type of file to perform the analysis.

- **IC50**
- **Genomic Features**
- **DRUG_DECODER**
- **Concentrations**

The first one is compulsory, the second one may be changed. The third and fourth ones are not required for the analysis but used in the HTML output.

3.1.2 Example

You can analyse a given drug for a given genomic feature

```
from gdsctools import readers, anova
r = readers.IC50('ANOVA_input.txt')
an = anova.GDSC_ANOVA(r.ic50, r.features)
an.anova_one_drug_one_feature('Drug_1_IC50', 'TP53_mut',
                              show_boxplot=True)
```

Or analyse a given drug across all features:

```
from gdsctools import readers, anova
r = readers.IC50('ANOVA_input.txt')
an = anova.GDSC_ANOVA(r.ic50, r.features)
df = an.anova_one_drug('Drug_1_IC50') # no plots are generated here.
```

Or analyse a all drugs across all features. This takes a long depending on the number of drugs and features (30 minutes for 250 drugs and 1000 features):

3.2 Output and visualisation

3.3 HTML report

3.4 Standalone applications

3.5 Notebooks

Developers Guide

4.1 References

Contents

- *References*
 - *ANOVA related*
 - *Statistical Tools*
 - *Readers*
 - *Visualisation*
 - * *Volcano plot*
 - * *Boxplot and beeswarm*
 - *Pipelines and reports*
 - *Others*

4.1.1 ANOVA related

4.1.2 Statistical Tools

4.1.3 Readers

4.1.4 Visualisation

Volcano plot

Boxplot and beeswarm

4.1.5 Pipelines and reports

4.1.6 Others

4.2 For developers

4.2.1 The GDSCTools test suite

- The entire GDSCTools packages is tested. Tests are in the `./test/gdsctools` directory of the source code.
- As of version 0.3, about 80% of the code is covered.
- If you add a new module, please add a corresponding test file in `./test/gdsctools`

4.2.2 Documentation

The documentation is based on Sphinx. This means that all code is documented using the REST syntax. Docstring are added in classes and functions as much as possible with code examples.

See for example the file in `gdsctools/anova.py` and in particular the class `ANOVA`.

In addition, in the `./doc` directory there are a set of files using REST syntax. If you go to that directory and type:

```
make html
```

The entire documentation including Tutorial and Developer guide will be parsed and interpreted. The resulting HTML documentation can then be found in `doc/build/html`.

Note: the command above will work only if `gdsctools` source code is available and installed in the shell where the command is executed.

The documentation can then be uploaded on pypi. Go to the directory where is the file `setup.py` and type:

```
python setup.py upload_docs
```

4.3 ChangeLog

4.3.1 Version 0.3 27 Oct 2015

Cancer specific now included and tested on BRCA and BLCA cases.

4.3.2 Version 0.2 23 Oct 2015

First working version with HTML output.

4.3.3 Version 0.1 14 Oct 2015

First working version of `gdsctools` with test and documentation. Tested against version 17. A standalone app is also provided as a command line argument (named `gdsctools_anova`).

Issues

Please fill bug report in <https://github.com/CancerRxGene/gdsctools/issues>

Contributions

Please join <https://github.com/CancerRxGene/gdsctools>

6.1 Glossary

EC50 The EC50 is the half maximal Effective Concentration and refers to the concentration of a drug, antibody or toxicant which induces a response halfway between the baseline and maximum after a specified amount of time. It is used as a measure of drug's potency

IC50 The IC50 is the half maximal Inhibitory Concentration that is a measure of effectiveness of a drug or substance in inhibiting a specific biological or biochemical function.

Sometimes, IC50 values are converted to the pIC50 scale that is $-\log_{10}(IC_{50})$.

pIC50 is usually given in terms of molar concentration (mol/L or M). Therefore to obtain pIC50, an IC50 should be specified in units of M. When IC50 is expressed in microM or nanoM, it will need to be converted to M before conversion to pIC50.

MSI, Microsatellite Instability Markers indicating presence or absence of a MSI shift, allele homozygosity/heterozygosity, and loss of heterozygosity (LOH) observed in the tumor sample for each participant

E

EC50, [13](#)

ec50, [13](#)

I

IC50, [13](#)

ic50, [13](#)

M

MSI, Microsatellite Instability, [13](#)