
cancerscope Documentation

Release 1.0

Jasleen Grewal

Mar 19, 2020

Contents

| | | |
|----------|---|-----------|
| 1 | Installation | 1 |
| 2 | Installing previous releases | 3 |
| 3 | Installing bleeding edge release | 5 |
| 4 | Major version changes | 7 |
| 5 | Tutorial with example file | 9 |
| 6 | Overview | 11 |

CHAPTER 1

Installation

Cancerscope is available through pip. It supports all Python versions upwards of Python 2.7. It has been developed and tested with Python 2.7.14, 3.4, 3.5, 3.6, and 3.7.

Installing previous releases

To upgrade to a newer release, simply run the following bash command: .. code:: bash

```
pip install --upgrade cancerscope
```

To install a previous release, for example, v.0.42 (last supported release that used theano and lasagne): .. code:: bash

```
pip install cancerscope=0.42
```


CHAPTER 3

Installing bleeding edge release

To install the latest release instead of the most recent stable release, you can install from source. Use the following bash commands:

```
>>> git clone https://github.com/jasgrewal/cancerscope.git
>>> cd cancerscope
>>> python setup.py install
```


CHAPTER 4

Major version changes

Model setup using theano and lasagne .. versionchanged:: 0.42

Model setup using keras .. versionadded:: 1.0

Plotting option available for python 2.7 .. versionadded:: 0.0

Plotting option removed .. versionchanged:: 0.31

Once you have installed cancerscope, you're set!

Start by importing cancerscope into a python session like so (from command-line): .. code:: bash

python

```
import cancerscope as cs
```

Tutorial with example file

There is a tutorial with sample code, that walks through a small example of obtaining predictions directly from a file containing gene expression profiles. It's available on *GitHub* <<https://github.com/jasgrewal/cancerscope/tree/master/tutorial>> and is useful for understanding how the various data objects need to be structured in order to retrieve predictions from cancerscope.

Predictions from cancerscope can be of two types. You can get the highest predicted category for each sample, with the accompanying confidence in prediction. You can also obtain a more granular decomposition of the prediction by retrieving a detailed view of the predictions. This view will enumerate the predicted cancer type(s) for each sample, along with the confidence for each category. It is useful if the machines in the ensemble are not in agreement about the highest predicted category.

You can obtain predictions from gene expression profiles stored in a text file, in 3 easy steps. Here's the code, assuming each row is a samples and each column is a gene:

```
import cancerscope as cs
scope_obj=cs.scope()
preds_from_file = scope_obj.get_predictions_from_file("path/to/my/file_with_
↳geneexpression.txt")
```

You can also load the data into the sample (row) * gene (column) form in python first, and then obtain predictions as follows:

```
import cancerscope as cs
scope_obj=cs.scope()
preds_df_from_xdat = scope_obj.predict(X = test_x, x_features = test_features, x_
↳features_genecode = test_genecode, x_sample_names=test_samples)
```

Notice that with this approach, you'll need to provide other information like a list of gene names (*test_features*), a string indicating the type of gene name used (*test_genecode*), and optionally, a list of sample names (*test_samples*).

CHAPTER 6

Overview

Cancerscope is a library for cancer diagnosis. You can use it to obtain quantitative diagnosis for cancer sequencing data using RNA sequencing information (gene expression profiles).