GENIE3: documentation

Author: Vân Anh Huynh-Thu, vahuynh@ulg.ac.be

This is the documentation for the python implementation of GENIE3. This implementation is a research prototype and is provided “as is”. No warranties or guarantees of any kind are given. Do not distribute the GENIE3 python code or use it other than for your own research without the permission of the author.

The GENIE3 method is described in the following paper:
Huynh-Thu V. A., Irrthum A., Wehenkel L., and Geurts P. (2010) Inferring regulatory networks from expression data using tree-based methods. *PLoS ONE*, 5(9):e12776.

1 Installation

To be able to run GENIE3, you have to install the **scikit-learn** module (≥ version 0.16). Instructions for installing sciki-learn are provided here:
http://scikit-learn.org/dev/install.html

2 Load the GENIE3 package

```
from GENIE3 import *
```

3 Run GENIE3

GENIE3 is meant to be run on gene expression data. A file ’data.txt’, containing an example of gene expression dataset, is provided for this tutorial. To load this data:

```
data = loadtxt('data.txt',skiprows=1)
f = open('data.txt')
gene_names = f.readline()
f.close()
gene_names = gene_names.rstrip('\n').split('\t')
```

- *data* is an array containing the expression data of 10 genes in 136 conditions.
- *genes_ names* is a list containing the names of the genes.

Run GENIE3 with its default parameters

The only mandatory input argument to the function $GENIE3()$ is the expression matrix:

```
(VIM, prediction_score, treeEstimators) = GENIE3(data)
```

$GENIE3()$ returns 3 outputs:

- an array VIM containing the scores of the putative regulatory links. *VIM(i,j)* is the weight of the link directed from the i-th gene to j-th gene.
- *prediction_ score*, which is an empty list by default.
- *treeEstimators*, which is an empty list by default.

Restrict the candidate regulators to a subset of genes

```
# Genes that are used as candidate regulators
regulators = ['CD19', 'CDH17','RAD51','OSR2','TBX3']
(VIM2, prediction_score, treeEstimators) = ...
  GENIE3(data,gene_names=gene_names,regulators=regulators)
```

In $VIM2$, the links that are directed from genes that are not candidate regulators have a score equal to 0.

Change the tree-based method and its settings

```
# Use Extra-Trees method
tree_method='ET'

# Number of randomly chosen candidate regulators at each node of a tree
K = 7

# Number of trees per ensemble
ntrees = 50

# Run the method with these settings
(VIM3, prediction_score, treeEstimators) = ...
  GENIE3(data,tree_method=tree_method,K=K,ntrees=ntrees)
```

Compute the prediction score on out-of-bag samples

The prediction score is a measure of the predictive performance of the learned tree models. In practice, we estimate the prediction score by computing the Pearson correlation between the predicted and true expression values, over the out-of-bag samples (i.e. the samples left out when learning a tree).
To compute the prediction score, the tree method must be Random Forests and the input argument *compute_prediction_score* must be set to *True*.

```
# The prediction score can only be computed when using Random Forests
tree_method = 'RF'

(VIM4, prediction_score, treeEstimators) = ...
  GENIE3(data,tree_method=tree_method,compute_prediction_score=True)
```

prediction_score is the estimated prediction score, averaged over all the genes and all the trees.

Save the tree models

The tree models can be saved, e.g. to be later used for the prediction of gene expression profiles. To save the models, the input argument *save_models* must be set to *True*.

```
(VIM5, prediction_score, treeEstimators) = GENIE3(data,save_models=True)
```

treeEstimators is a list containing the different tree models (one for each gene).

Obtain more information

```
help(GENIE3)
```

4 Write the predictions

Get the predicted ranking of all the regulatory links

```
get_link_list(VIM)
```

The output will look like this:

```
## G1 G5 0.527481
## G5 G1 0.517994
## G6 G8 0.384952
## G8 G6 0.343328
## G9 G10 0.320162
## G2 G8 0.257154
## G9 G7 0.240072
## ...
```

Each line corresponds to a regulatory link. The first column shows the regulator, the second column shows the target gene, and the last column indicates the score of the link.
If the gene names are not provided, the i-th gene is named "Gi".
Note that the ranking that is obtained will be slightly different from one run to another. This is due to the intrinsic randomness of the Random Forest and Extra-Trees methods. The variance of the ranking can be decreased by increasing the number of trees per ensemble.

Important note on the interpretation of the scores: The weights of the links returned by *GENIE*3() **do not have any statistical meaning** and only provide a way to rank the regulatory links. There is therefore no standard threshold value, and caution must be taken when choosing one.

Show the names of the genes

```
get_link_list(VIM,gene_names=gene_names)
```

```
## TBX3 XRCC2 0.527481
## XRCC2 TBX3 0.517994
## CD93 CREB5 0.384952
## CREB5 CD93 0.343328
## CD19 RAD51 0.320162
## GATA5 CREB5 0.257154
## ...
```

Show only the links that are directed from the candidate regulators

```
get_link_list(VIM,gene_names=gene_names,regulators=regulators)
```

Show the first 5 links only

```
get_link_list(VIM,gene_names=gene_names,regulators=regulators,maxcount=5)
```

Write the predicted links in a file

```
get_link_list(VIM,gene_names=gene_names, ...
  regulators=regulators,file_name='ranking.txt')
```

Obtain more information

```
help(get_link_list)
```