Manual of NetSAM

Jing Wang, Bing Zhang

February 18, 2014

1 Introduction

The last decade of systems biology research has demonstrated that networks rather than individual genes govern the onset and progression of complex diseases. Meanwhile, real-world complex networks usually exhibit hierarchical organization, in which nodes can be combined into groups that can be further combined into larger groups, and so on over multiple scales. Thus, identifying the hierarchical organization of a network becomes indispensable in complex disease studies. A traditional and useful method for revealing hierarchical architecture of network is hierarchical clustering, which groups data over a variety of scales by creating a hierarchical tree. However, hierarchical clustering has three major limitations. First, there are many different leaf node orderings consistent with the structure of a hierarchical tree, and hierarchical clustering does not optimize the ordering. Secondly, hierarchical clustering does not assess the statistical significance of the modular organization of a network. Finally, it does not specify relevant hierarchical levels and modules at different scales. To address these limitations, we developed the NetSAM (Network Seriation and Modularization) package which will identify the hierarchical modules from a network (network modularization) and find a suitable linear order for all leaves of the identified hierarchical organization (network seriation). NetSAM takes an edge-list representation of a network as an input and generates as files that can be used as an input for the one-dimensional network visualization tool NetGestalt (http://www.netgestalt.org) or other network analysis. NetSAM uses random walk distance-based hierarchical clustering to identify the hierarchical modules of the network and then uses the optimal leaf ordering (OLO) method to optimize the one-dimensional ordering of the genes in each module by minimizing the sum of the pair-wise random walk distance of adjacent genes in the ordering. The detailed description of the NetSAM method can be found in our recently published Nature Methods paper ”NetGestalt: integrating multidimensional omics data over biological networks” (http://www.nature.com/nmeth/journal/v10/n7/full/nmeth.2517.html).

2 Environment

NetSAM requires R version 2.15.1 or later, which can be downloaded from the website http://www.r-project.org/. Because the seriation step requires pair-wise distance between all nodes, NetSAM is memory consuming. We recommend to use the 64 bit version of R to run the NetSAM. For networks with less than 10,000 nodes, we recommend to use a computer with 8GB memory. Using our computer with 2.7 GHz Intel Core i5 processor and 8 GB 1333 MHz DDR3 memory, NetSAM took 402 seconds to analyze the HPRD network (http://www.hprd.org) with 9198 nodes. For networks with more than 10,000 nodes, a computer with at least 16GB memory is recommended. NetSAM requires the following packages: igraph (>=0.6-1), seriation (>=1.0-6) and graph (>=1.34.0), which can be installed as follows.

```
>install.packages(”igraph”)

>install.packages(”seriation”)

>source(”http://bioconductor.org/biocLite.R”)

>biocLite(”graph”)
```

3 Network Seriation and Modularization

After building up the basic environment mentioned above, the users can install the NetSAM package and use it to analyze networks.

```
> library("NetSAM")
```

```
*****************************************
*                                       *
*         Welcome to use NetSAM !       *
*                                       *
*****************************************
```

```
> cat("The input network can be a file.\n")
```

```
The input network can be a file.
```

```
> inputNetwork <- system.file("extdata","exampleNetwork.txt",package="NetSAM")
> cat("The input network can be also a data object, such as graphNEL object.\n")
```

```
The input network can be also a data object, such as graphNEL object.
```

```
> data(inputNetwork)
> outputFileName <- paste(getwd(),"/NetSAM",sep="")
> result <- NetSAM(inputNetwork, outputFileName, minModule=(-1), maxStep=4,
+ method="Modularity Cutoff", ModularityThr=0.2, ZRandomNum=10, permuteNum=100, pThr=0.05)
```

```
Network has 320 nodes and 769 edges

Identifying the hierarchical modules of the network...

Start to analysis subnetwork  1 !
Evaluate Leve 1 network...
Evaluate Level 2 networks...
Evaluate Level 3 networks...
Evaluate Level 4 networks...


Reorder the genes in the one dimentional layout...
Processing completed!
```

3.1 Input

This section describes the arguments of the NetSAM function:

1. *inputNetwork* is the network under analysis, which can be the name of the input network file in the edge-list format (each row represents an edge with two node names separated by a tab or space) or be a data object in R (data object must be graphNEL class or data.frame (or matrix) class with two columns).

2. *outputFileName* is the name of the output file. If no file path is provided, the output file will be saved to the current path.

3. *minModule* is the minimum number of nodes for a module (or minimum module size). If the size of a module identified by the function is less than the specified number, the module will not be further partitioned into sub-modules. The default is -1 which means NetSAM will set *minModule* as 5 or 0.3 percent of the number of nodes in the input network, whichever is larger.

4. Because NetSAM uses random walk distance-based hierarchical clustering to reveal the hierarchical organization of an input network, it requires a specified length of the random walks. To get the optimal length, the function will test a range of lengths ranging from 2 to *maxStep*. The default is 4.

5. To test whether a network under consideration has a non-random internal modular organization, the function provides three options: ”Modularity Cutoff”, ”ZScore” and ”Permutation”. ”Modularity Cutoff” means if the modularity score of the network is above a specified cutoff value, the network will be considered to have internal organization and will be further partitioned. For ”ZScore” and ”Permutation”, the function first uses the edge switching method to generate a given number of random networks with the same number of nodes and an identical degree sequence and calculates the modularity scores for these random networks. Then, ”ZScore” method will transform the real modularity score to a z score based on the random modularity scores and then transform the z score to a p value assuming a standard normal distribution. The ”Permutation” method will compare the real modularity score with the random ones to calculate a p value. Finally, under a specified significance level, the function determines whether the network can be further partitioned. The default is ”Modularity Cutoff”.

6. *ModularityThr* is the threshold of modularity score for the ”Modularity Cutoff” method. The default is 0.2.

7. *ZRandomNum* is the number of random networks that will be generated for the ”ZScore” calculation. The default is 10.

8. *permuteNum* is the number of random networks that will be generated for the ”Permutation” p value calculation. The default is 100.

9. *pThr* is the significance level for determining whether a network has non-random internal modular organization for the ”ZScore” or ”Permutation” methods.

3.2 Output

The NetSAM function outputs a ”nsm” file that can be used as an input for the one-dimensional network visualization tool NetGestalt (http://www.netgestalt.org). Meanwhile, the NetSAM function also outputs a list object in R which contains all information in ”nsm” file.