NCIgraph: networks from the NCI pathway integrated database as graphNEL objects.

Laurent Jacob

March 30, 2012

Abstract

The NCI pathway integrated database is a large database of biological networks, including nature curated pathways as well as other pathways imported from biocarta and reactome. *NCIgraph* is a data package that imports the NCI PID networks as R graphNEL objects. It also gives several options to parse and transform the networks.

1 Introduction

An increasing number of approaches in statistics and machine learning for the analysis of molecular data use known biological networks. The objective is generally to obtain results that make sense at the biological system level and/or to improve the performances of the method [Ideker et al., 2002, **?**, Rapaport et al., 2007, Jacob et al., 2009, Vaske et al., 2010, Vandin et al., 2010, Jacob et al., 2010].

All these methods require the knowledge of some biological networks. Several online databases regroup lists of such biological networks. One of them, KEGG [1], has already been interfaced with R through the bioconductor package *KEGGgraph* which provides tools to read KEGG networks from xml files and manipulate the resulting R objects. Another major public database is the pathway interaction database (PID) maintained by Nature and the National Cancer Institute (NCI) [2]. *NCIgraph* provides systematic importation of these networks in R.

The way *NCIgraph* proceeds is the following :

1. Read the networks in BioPAX format [3] in Cytoscape [4]. This is done using the BioPAX Cytoscape plugin [5].

[1] `http://www.genome.jp/kegg/`
[2] `http://pid.nci.nih.gov/`
[3] `http://www.biopax.org/`
[4] `http://www.cytoscape.org/`
[5] `http://cbio.mskcc.org/cytoscape/plugins/biopax/`

2. Read graphNEL objects (defined in the *graph* R package) from Cytoscape. This is done using the CytoscapeRPC Cytoscape plugin [6] in combination with the *RCytoscape* bioconductor package.

3. Optionally, parse the obtained raw network to get a graph which can be more easily used in statistics methods.

The output of the first two steps is available for download through *NCIgraph* for convenience, but users who want to load different versions of the networks, or networks stored in other BioPAX files can also perform the first two steps themselves and use *NCIgraph* to read the networks from Cytoscape.

The idea of the parsing in the third step is to build a network whose nodes are genes and whose edges represent direct or indirect interactions at the *expression* level. Some nodes in the raw networks stored in the BioPAX files represent proteins, protein complexes or concepts like transport, biochemical reactions etc. If a protein A is known to activate a protein B which is a transcription factor for gene C, a relevant network in terms of expression correlation should be A and B pointing to C, whereas the network will most likely be represented as A pointing to B pointing to C. Since many statistical methods essentially use biological networks as a prior on the covariance structure of the gene expression, it is important to be able to perform such a transformation.

2 Software features

NCIgraph offers the following functionalities:

Reading of all networks opened in Cytoscape This is essentially a call to functions of *RCytoscape* to systematically load all the networks currently opened in Cytoscape as graphNEL objects.

Providing data files of the loaded networks In case you don't want to load the networks through Cytoscape, *NCIgraph* allows you to download the resulting R objects.

Parsing networks *NCIgraph* provides a set of functions to transform the networks that have been read from Cytoscape, *e.g.* to only keep the nodes corresponding to genes or propagate regulation relationships.

NCIgraph objects *NCIgraph* defines a *NCIgraph* class. *NCIgraph* extends *graphNEL*, and is assigned special methods for `getSubtype` and `subGraph`.

[6] `https://wiki.nbic.nl/index.php/CytoscapeRPC`

3 Data

Since loading networks from Cytoscape is very lengthy and requires Cytoscape along with two of its plugins, we provide pre-loaded raw networks in the *NCIgraphData* bioconductor data package. *NCIgraphData* provides `NCI.cyList` and `reactome.cyList`. The former contains 460 of the *NCI-Nature curated* and *BioCarta imported* pathways of the NCI PID. The latter contains 487 if the *Reactome imported* pathways of the NCI PID. Some NCI-PID networks are not in the list for one of the following reasons:

- The corresponding BioPAX file could not be read by the BioPAX Cytoscape plugin (some of them even crash Cytoscape).
- RCytoscape couldn't read the network from Cytoscape.

For the latter problem, the `getNCIPathways` function catches the errors, and returns a list of the networks that were loaded in Cytoscape but could not be read into R.

An important remark is that none of the networks imported from Reactome contains nodes associated with entrez ids, so parsing them as presented in the following case study will yield empty graphs.

4 Case studies

We now show on a simple example how *NCIgraph* can be used. In this vignette, we simply load some raw networks, parse them and visualize the results. For a more complete example where the *NCIgraph* objects are used to identify differentially expressed pathways for some gene expression data, the reader is referred to the `Loi2008` demo of the *DEGraph* bioconductor package.

4.1 Loading the library and the data

We load the *NCIgraph* package by typing or pasting the following codes in R command line:

```
> library(NCIgraph)
```

In this example, the raw networks have been pre-stored in an `.RData` file to avoid lengthy downloading and formatting. For examples on how to build these variables, see the `NCIgraphDemo` demo in the package.

```
> data("NCIgraphVignette", package="NCIgraph")
```

4.2 Parsing the networks

The loaded NCI pathways can now be parsed :

```
> grList <- getNCIPathways(cyList=NCI.demo.cyList, parseNetworks=TRUE, entrezOnly=TRUE, ver

Loading network acetylation and deacetylation of rela in nucleus
Loading network visceral fat deposits and the metabolic syndrome
Loading network role of egf receptor transactivation by gpcrs in cardiac hypertrophy
Loading network role of parkin in ubiquitin-proteasomal pathway
Loading network fmlp induced chemokine gene expression in hmc-1 cells
Loading network west nile virus
Loading network chromatin remodeling by hswi/snf atp-dependent complexes
Loading network alk in cardiac myocytes
Loading network BARD1 signaling events
Loading network cystic fibrosis transmembrane conductance regulator (cftr) and beta 2 adren
```

Note that we specify that we want the networks to be parsed (parseNetworks=TRUE), and that the parsing should only keep nodes for which an entrez id is available in the raw networks (entrezOnly=TRUE). Other parsing options are directly passed to the `parseNCINetwork` function. In particular, it is possible to ask that regulation edges are not "propagated" (propagateReg=FALSE), *i.e.*, referring to the example in Introduction, that the resulting network has A pointing to B, not to C.

4.3 Visualizing the result

We now plot the first element of the list before and after parsing. We first need another library to plot *graphNEL* objects.

```
> library('Rgraphviz')
```

Then we can plot the graphs :

```
> graph <- NCI.demo.cyList[[1]]
> gNames <- unlist(lapply(graph@nodeData@data,FUN=function(e) e$biopax.name))
> names(gNames) <- nodes(graph)
> graph <- layoutGraph(graph)
> nodeRenderInfo(graph) <- list(label=gNames, cex=0.75)
> renderGraph(graph)
```

```
> graph <- grList[[1]]
> gNames <- unlist(lapply(graph@nodeData@data,FUN=function(e) unique(e$biopax.xref.ENTREZGE
> names(gNames) <- nodes(graph)
```

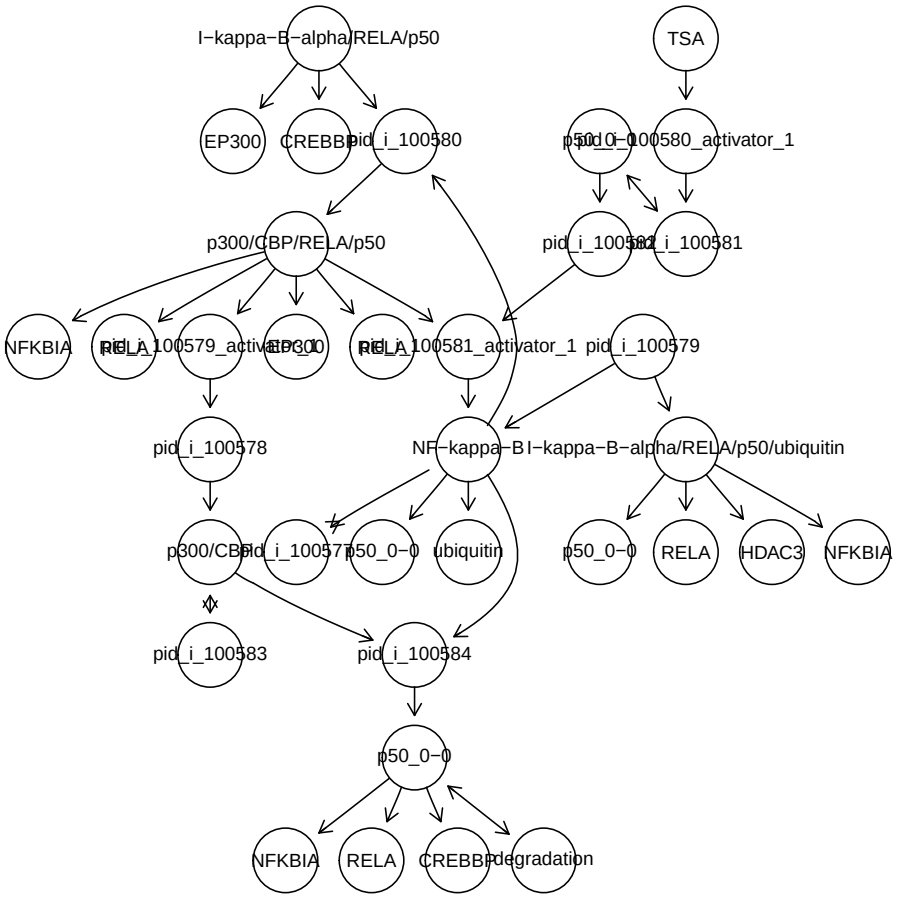

Figure 1: Raw network.

```
> graph <- layoutGraph(graph)
> nodeRenderInfo(graph) <- list(label=gNames, cex=0.75)
> renderGraph(graph)
```

Acknowledgements

We are very grateful to Paul Shannon for his very helpful advices on loading BioPAX files into R through Cytoscape and Sandrine Dudoit for her help on R package writing. We also thank the editorial team of the NCI Pathway Interaction Database who kindly provided the BioPAX files of all their networks and for helping us understand the organization of

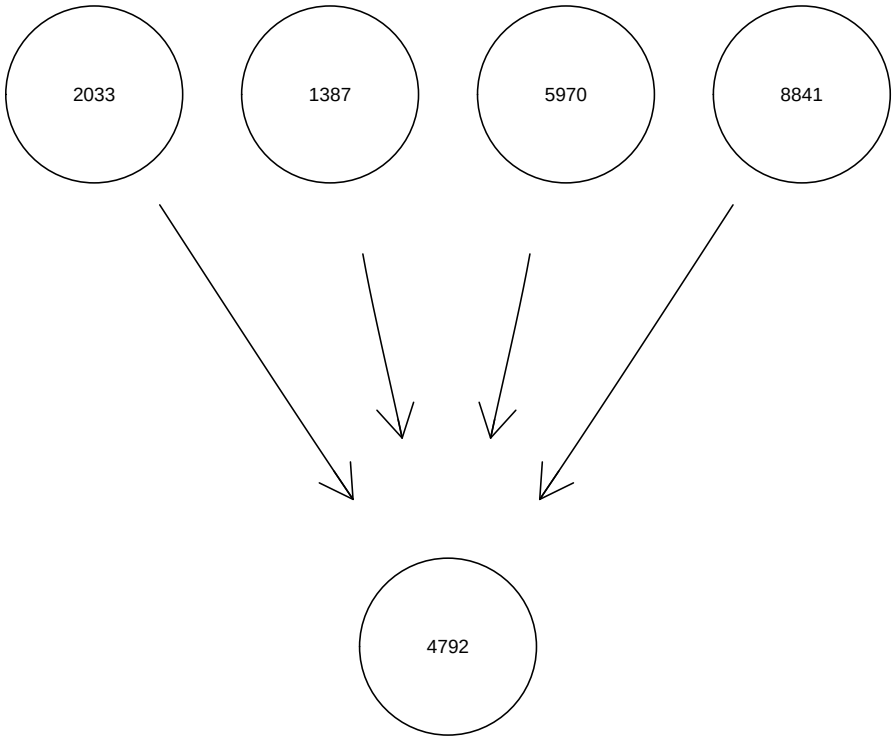

Figure 2: Parsed network.

these files. Finally, we thank Nishant Gopalakrishnan who reviewed this package for his helpful corrections.

References

Trey Ideker, Owen Ozier, Benno Schwikowski, and Andrew F. Siegel. Discovering regulatory and signalling circuits in molecular interaction networks. In *ISMB*, pages 233–240, 2002.

L. Jacob, P. Neuvial, and S. Dudoit. Gains in power from structured two-sample tests of means on graphs. Technical Report arXiv:q-bio/1009.5173v1, arXiv, 2010.

Laurent Jacob, Guillaume Obozinski, and Jean-Philippe Vert. Group lasso with overlap and graph lasso. In *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*, pages 433–440, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-516-1. doi: http://doi.acm.org/10.1145/1553374.1553431.

F. Rapaport, A. Zynoviev, M. Dutreix, E. Barillot, and J.-P. Vert. Classification of microarray data using gene networks. *BMC Bioinformatics*, 8:35, 2007.

Fabio Vandin, Eli Upfal, and Benjamin J. Raphael. Algorithms for detecting significantly mutated pathways in cancer. In *RECOMB*, pages 506–521, 2010.

Charles Vaske, Stephen Benz, Zachary Sanborn, Dent Earl, Christopher Szeto, Jingchun Zhu, David Haussler, and Joshua Stuart. Inference of patient-specific pathway activities from multi-dimensional cancer genomics data using PARADIGM. In *ISMB*, 2010.