

Working with SpidermiR package_pdf

Claudia Cava, Antonio Colaprico, Alex Graudenzi, Gloria Bertoli, Tiago C. Silva, Catharina Olsen, Houtan Noushmehr, Gianluca Bontempi, Giancarlo Mauri, Isabella Castiglioni

2016-05-26

Introduction

Biological systems are composed of multiple layers of dynamic interaction networks. These networks can be decomposed, for example, into: co-expression, physical, co-localization, genetic, pathway, and shared protein domains.

GeneMania provides us with an enormous collection of data sets for interaction network studies (Warde-Farley D, Donaldson S, Comes O, Zuberi K, Badrawi R, and others 2010). The data can be accessed and downloaded from different database, using a web portal. But currently, there is not a R-package to query and download these data.

An important regulatory mechanism of these network data involves microRNAs (miRNAs). miRNAs are involved in various cellular functions, such as differentiation, proliferation, and tumourigenesis. However, our understanding of the processes regulated by miRNAs is currently limited and the integration of miRNA data in these networks provides a comprehensive genome-scale analysis of miRNA regulatory networks. Actually, GeneMania doesn't integrate the information of miRNAs and their interactions in the network.

SpidermiR allows the user to query, prepare, download network data (e.g. from GeneMania), and to integrate this information with miRNA data with the possibility to analyze these downloaded data directly in one single R package. This technical report gives a short overview of the essential **SpidermiR** methods and their application.

Installation

To install use the code below.

```
source("https://bioconductor.org/biocLite.R")
biocLite("SpidermiR")
```

SpidermiRquery: Searching network

You can easily search GeneMania data using the **SpidermiRquery** function.

SpidermiRquery_species: Searching by species

The user can query the species supported by GeneMania, using the function **SpidermiRquery_species**:

```
require(SpidermiR)
```

```
## Loading required package: SpidermiR
```

```
## Loading required package: miRnAtap
```

```

## Loading required package: AnnotationDbi

## Loading required package: stats4

## Loading required package: BiocGenerics

## Loading required package: parallel

##
## Attaching package: 'BiocGenerics'

## The following objects are masked from 'package:parallel':
##
##   clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##   clusterExport, clusterMap, parApply, parCapply, parLapply,
##   parLapplyLB, parRapply, parSapply, parSapplyLB

## The following objects are masked from 'package:stats':
##
##   IQR, mad, xtabs

## The following objects are masked from 'package:base':
##
##   anyDuplicated, append, as.data.frame, as.vector, cbind,
##   colnames, do.call, duplicated, eval, evalq, Filter, Find, get,
##   grep, grepl, intersect, is.unsorted, lapply, lengths, Map,
##   mapply, match, mget, order, paste, pmax, pmax.int, pmin,
##   pmin.int, Position, rank, rbind, Reduce, rownames, sapply,
##   setdiff, sort, table, tapply, union, unique, unlist, unsplit

## Loading required package: Biobase

## Welcome to Bioconductor
##
##   Vignettes contain introductory material; view with
##   'browseVignettes()'. To cite Bioconductor, see
##   'citation("Biobase")', and for packages 'citation("pkgname)".

## Loading required package: IRanges

## Loading required package: S4Vectors

## Loading required package: DBI

##

org<-SpidermiRquery\_species(species)

```

The list of species is shown below:

```
knitr::kable(org, digits = 2,
              caption = "List of species", row.names = TRUE)
```

Table 1: List of species

	tabOrgd
1	Arabidopsis_thaliana
2	Caenorhabditis_elegans
3	Danio_rerio
4	Drosophila_melanogaster
5	Escherichia_coli
6	Homo_sapiens
7	Mus_musculus
8	Rattus_norvegicus
9	Saccharomyces_cerevisiae

SpidermiRquery_networks_type: Searching by network categories

The user can query the network types supported by GeneMania for a specific specie, using the function `SpidermiRquery_networks_type`. The user can select a specific specie using an index obtained by the function `SpidermiRquery_species` (e.g. `organismID=org[6,]` is the input for `Homo_sapiens`)

```
net_type<-SpidermiRquery_networks_type(organismID=org[9,])
```

The list of network categories in *Saccharomyces cerevisiae* is shown below:

```
net_type
```

```
## [1] "Co-localization"      "Other"
## [3] "Shared protein domains" "Predicted"
## [5] "Co-expression"        "Physical interactions"
## [7] "Genetic interactions"
```

SpidermiRquery_spec_networks: Searching by species, and network categories

You can filter the search by species using organism ID (above reported), and the network category. The network category can be filtered using the following parameters:

- **COexp** Co-expression
- **PHint** Physical_interactions
- **COloc** Co-localization
- **GENint** Genetic_interactions
- **PATH** Pathway
- **SHpd** Shared_protein_domains

```
net_shar_prot<-SpidermiRquery_spec_networks(organismID = org[9,],
                                             network = "SHpd")
```

The databases, which data are collected, are the output of this step. An example is shown below (for Shared protein domains in *Saccharomyces_cerevisiae* data are collected in INTERPRO, and PFAM):

```
net_shar_prot
```

```
## [1] "http://genemania.org/data/current/Saccharomyces_cerevisiae/Shared_protein_domains.INTERPRO.txt"
## [2] "http://genemania.org/data/current/Saccharomyces_cerevisiae/Shared_protein_domains.PFAM.txt"
```

SpidermiRquery_disease: Searching by miRNA-disease

The user can visualize the disease supported by SpidermiR, in order to focus only on miRNAs have already studied in a particular disease (retrieving data from miR2Disease (Jiang Q, Wang Y, Hao Y, Juan L, Teng M, Zhang X, Li M, Wang G, Liu Y. 2009)).

```
disease<-SpidermiRquery_disease(diseaseID)
```

SpidermiRdownload: Downloading network data

The user in this step can download the data, as previously queried.

SpidermiRdownload_net: Download network

The user can download the data (previously queried) with `SpidermiRdownload_net`.

```
out_net<-SpidermiRdownload_net(net_shar_prot)
```

SpidermiRdownload_miRNAprediction: Downloading miRNA predicted data target

The user can download the miRNA predicted data target from 4 database:DIANA, Miranda, PicTar and TargetScan

```
mirna<-c('hsa-miR-567','hsa-miR-566')
SpidermiRdownload_miRNAprediction(mirna_list=mirna)
```

SpidermiRdownload_miRNAvalidate: Downloading miRNA validated data target

The user can download the miRNA validated data target from: miRTAR and miRwalk.

```
list<-SpidermiRdownload_miRNAvalidate(validated)
```

SpidermiRdownload_pharmacomir: Download Pharmacomir Verified Sets from Pharmacomir database

The user can download Pharmacomir Verified Sets from Pharmacomir database (Rukov J, Wilentzik R, Jaffe I, Vinther J, Shomron N. 2013).

```
mir_pharmaco<-SpidermiRdownload_pharmacomir(pharmacomir=pharmacomir)
```

SpidermiRprepare: Preparing the data

SpidermiRprepare_NET: Prepare matrix of gene network with Ensembl Gene ID, and gene symbols

The user in this step obtained a gene network matrix with the integration of gene symbols ID.

```
geneSymb_net<-SpidermiRprepare_NET(organismID = org[9,],  
                                   data = out_net)
```

SpidermiRanalyze: : Analyze data from network data

SpidermiRanalyze_mirna_network: Integration of microRNA-target interactions.

The user in this step obtained a network matrix with miRNA-target interactions starting from a specific network. The user can focus on miRNAs have already linked to a particular disease or take all miRNAs. miRNA-gene interactions include data from validated studies (currently, mirTAR, miR2disease, and miRNAwalk (Jiang Q, Wang Y, Hao Y, Juan L, Teng M, Zhang X, Li M, Wang G, Liu Y. 2009) (Dweep H, Sticht C, Pandey P, Gretz N. 2011)). You can filter the search by disease. The miRNA network can be filtered by disease using the name of the disease, as obtained from SpidermiRquery_disease.

```
miRNA_NET<-SpidermiRanalyze_mirna_network(data=geneSymb_net,disease="prostate cancer",  
                                           miR_trg="val")
```

SpidermiRanalyze_mirna_gene_complnet: Integration of microRNA-target complete interactions

The user in this step obtained a gene network matrix with interaction miRNA and target, and gene-gene interaction. The user can focus on miRNA have already linked to a particular disease or take all miRNAs. miRNA-gene interactions include data from validated studies (currently, mirTAR, miR2disease, and miRNAwalk (Jiang Q, Wang Y, Hao Y, Juan L, Teng M, Zhang X, Li M, Wang G, Liu Y. 2009) (Dweep H, Sticht C, Pandey P, Gretz N. 2011)). The miRNA network can be filtered by disease using the name of the disease, as obtained from SpidermiRquery_disease.

```
miRNA_complNET<-SpidermiRanalyze_mirna_gene_complnet(data=geneSymb_net,disease="prostate cancer",  
                                                       miR_trg="val")
```

SpidermiRanalyze_mirnanet_pharm: Integration of pharmacomiR in the network

The user in this step can integrate the pharmacomiR database in order to link miRNA and drug effect in a specific network.

```
mir_pharmnet<-SpidermiRanalyze_mirnanet_pharm(mir_ph=mir_pharmaco,net=miRNA_NET)
```

SpidermiRanalyze_mirna_extra_cir: Integration of Extracellular/Circulating miRNA

The user can select the extracellular/circulating miRNAs found in the network obtained from SpidermiRanalyze_mirna_network or SpidermiRanalyze_mirna_gene_complnet. Extracellular/circulating miRNAs include data from mirandola database (Russo F, Di Bella S, Nigita G, Macca V, Lagana A, Giugno R, Pulvirenti A, Ferro A. 2012). The user using the following parameters can specify the network type:

- **mT** to obtain a microRNA-target interactions
- **mCT** to obtain a miRNA and target, and gene-gene interaction.

```
miRNA_NET_ext_circmT<-SpidermiRanalyze_mirna_extra_cir(data=miRNA_complNET,"mT")
```

```
miRNA_NET_ext_circmCT<-SpidermiRanalyze_mirna_extra_cir(data=miRNA_complNET,"mCT")
```

SpidermiRanalyze_direct_net: Searching by biomarkers of interest with direct interaction

This function finds other genes that are related to a set of biomarkers of interest (the input of user) with direct interactions.

```
biomark_of_interest<-c("hsa-miR-214","PTEN","FOXO1","hsa-miR-27a")
```

```
GIdirect_net<-SpidermiRanalyze_direct_net(data=miRNA_NET,BI=biomark_of_interest)
```

The data frame of SpidermiRanalyze_direct_net, GIdirect_net, is shown below:

SpidermiRanalyze_direct_subnetwork: Network composed by only the nodes in a set of biomarkers of interest

This function create a sub-network composed by only the nodes in genes of interest and the edges between them.

```
subnet<-SpidermiRanalyze_direct_subnetwork(data=miRNA_NET,BI=biomark_of_interest)
```

SpidermiRanalyze_subnetwork_neigh: Network composed by the nodes in biomarker of interest and all the edges among this bunch of nodes.

This function create a sub-network composed by the nodes in BI and, if some of them are connected to other nodes (even if not in BI), take also them (include all the edges among this bunch of nodes).

```
GIdirect_net_neigh<-SpidermiRanalyze_subnetwork_neigh(data=miRNA_NET,BI=biomark_of_interest)
```

SpidermiRanalyze_degree centrality: Ranking degree centrality genes

This function provides degree centrality, defined as the total number of direct neighbors for each biomarkers. Degree centrality is ranked, and parameter cut is able to cut off other genes.

```
top10_cent_gene<-SpidermiRanalyze_degree_centrality(miRNA_NET,cut=10)
```

SpidermiRanalyze_Community_detection: Find community detection

This function try to find dense subgraphs in directed or undirected graphs, by optimizing some criteria, using igraph library (Csardi G, Nepusz T. 2006).

The user can choose the algorithm to calculate the community structure:

- **EB** edge.betweenness.community
- **FC** fastgreedy.community
- **WC** walktrap.community
- **SC** spinglass.community
- **LE** leading.eigenvector.community
- **LP** label.propagation.community

```
comm<- SpidermiRanalyze_Community_detection(data=miRNA_NET,type="FC")
```

SpidermiRanalyze_Community_detection_net: Find the network of community detection

This function with input the community detection obtained by SPIDERanalyze_Community_detection find the direct network of the biomarker inside the indicated community.

```
cd_net<-SpidermiRanalyze_Community_detection_net(data=miRNA_NET,comm_det=comm,size=1)
```

SpidermiRanalyze_Community_detection_bi: Community detection from a set of biomarkers of interest

This function, starting from communities obtained from SpidermiRanalyze_Community_detection, is able to indicate in which communities are presented a set of biomarkers of interest.

```
gi=c("CF","ROCK1","KIT","CCND2")  
mol<-SpidermiRanalyze_Community_detection_bi(data=comm,BI=gi)
```

SpidermiRvisualize: Visualize network miRNA-target and gene-gene interaction

SpidermiRvisualize_mirnanet: Visualize the network.

The user can easily visualize network data.

```
library(networkD3)  
SpidermiRvisualize_mirnanet(data=mir_pharmnet[sample(nrow(mir_pharmnet), 150), ] )
```

SpidermiRvisualize_BI: Visualize the network from a set of biomarkers of interest.

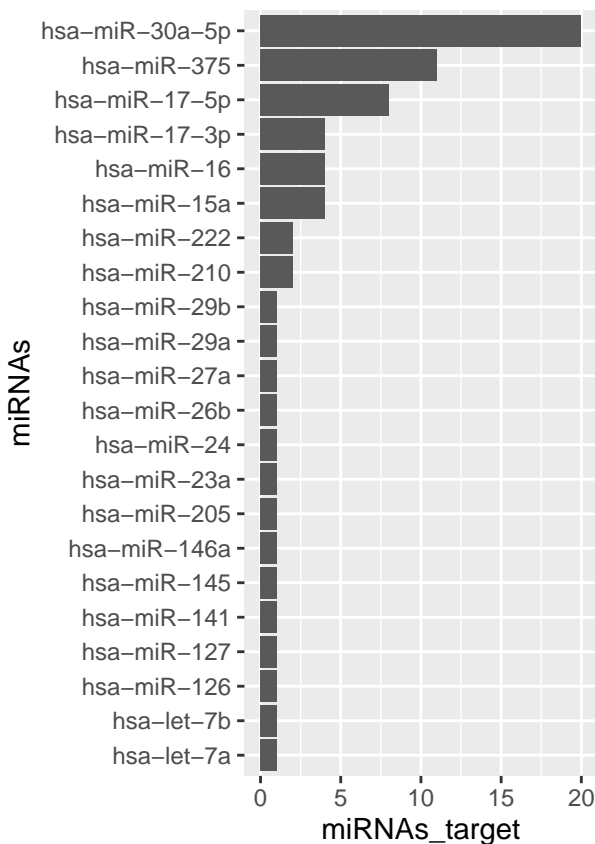
The user can visualize results obtained by SpidermiR analysis starting from a set of biomarkers of interest

```
biomark_of_interest<-c("hsa-let-7b","MUC1","PEX7","hsa-miR-222")
SpidermiRvisualize_BI(data=miRNA_NET,BI=biomark_of_interest)
```

SpidermiRvisualize_plot_target: Visualize the plot with miRNAs and the number of their targets in the network.

The user can Visualize results obtained by SpidermiRanalyze_mirna_network showing a plot with miRNAs and the number of their targets in the network.

```
SpidermiRvisualize_plot_target(data=miRNA_NET)
```

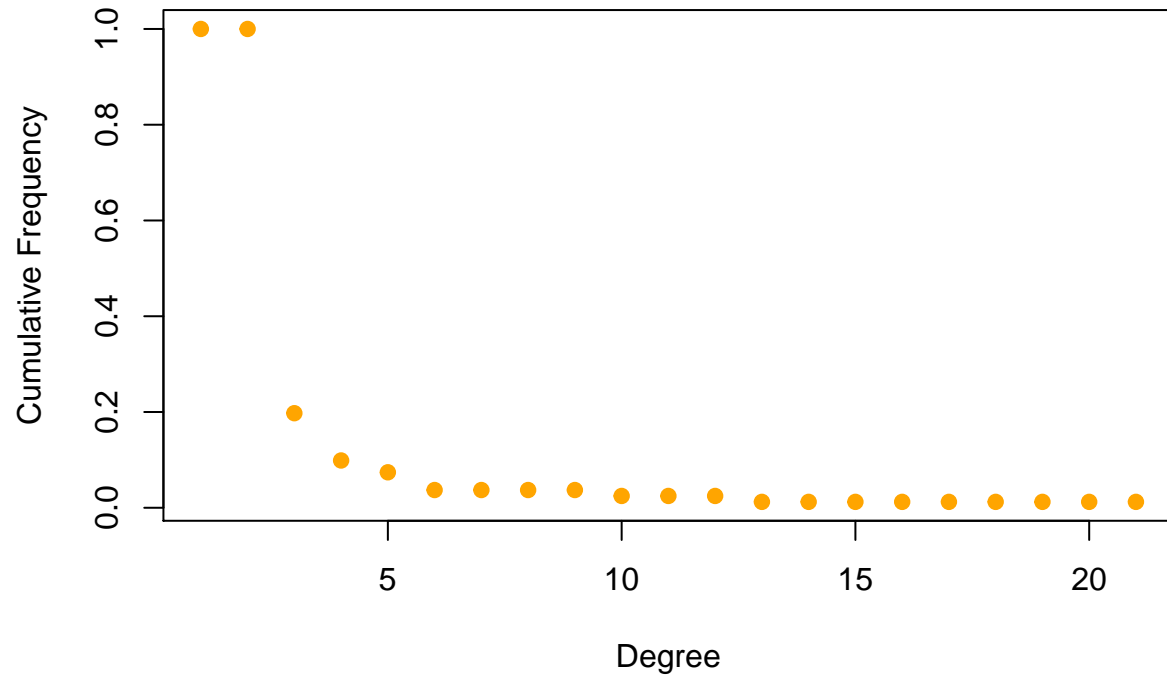


NULL

SpidermiRvisualize_degree_dist: plots the degree distribution of the network

The user can Visualize the degree distribution of the network

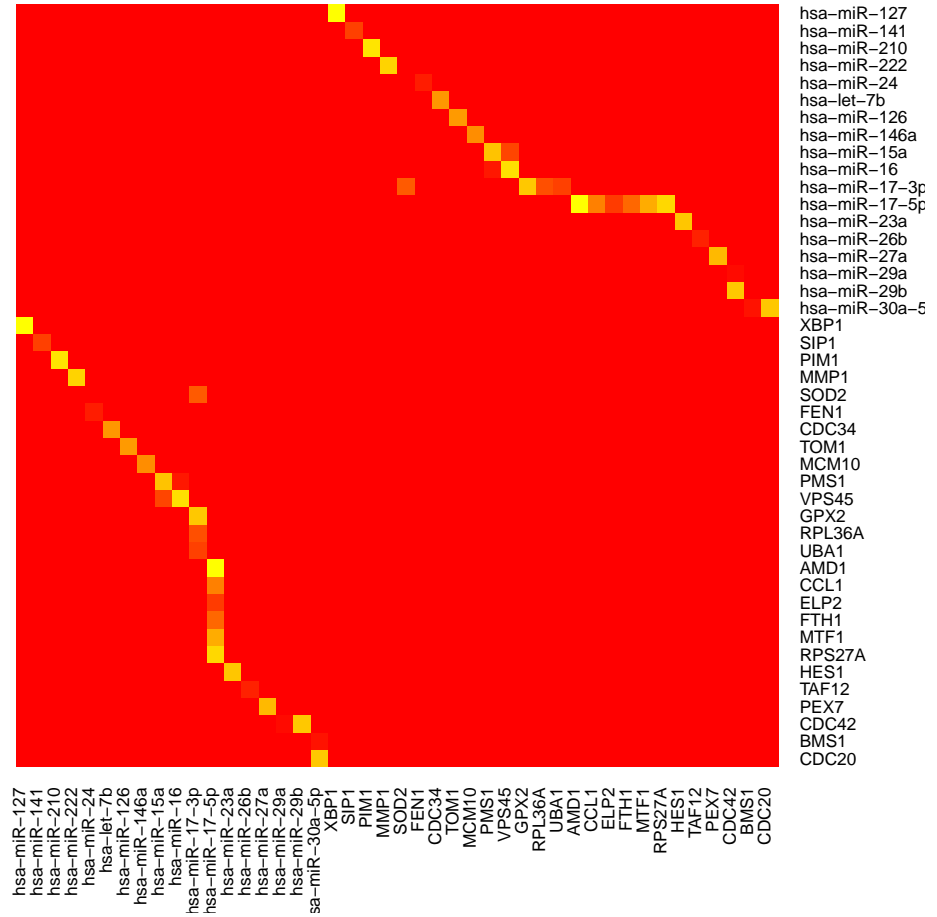
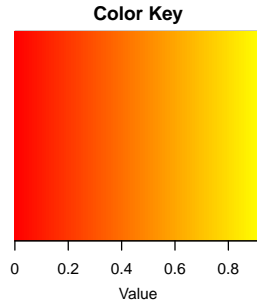

```
SpidermiRvisualize_degree_dist(data=miRNA_NET)
```



SpidermiRvisualize_adj_matrix: plots the adjacency matrix of the network

The user can Visualize the adjacency matrix of the network

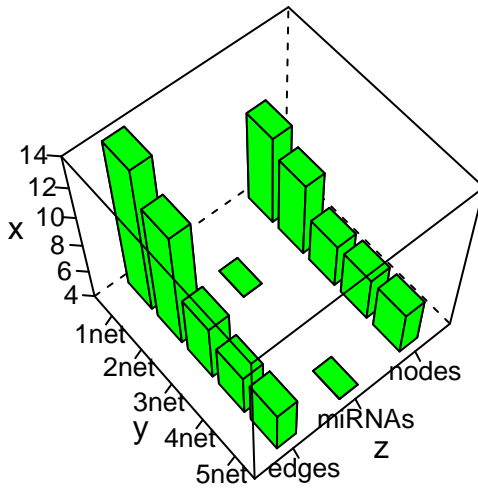
```
SpidermiRvisualize_adj_matrix(data=miRNA_NET[1:30,])
```



SpidermiRvisualize_3Dbarplot: 3D barplot

The user can visualize 3D barplot with a summary representation of the network (number of edges, nodes, miRNAs):

```
SpidermiRvisualize_3Dbarplot(Edges_1net=1041003,Edges_2net=100016,Edges_3net=3008,Edges_4net=1493,Edges_5net=100016,Edges_6net=100016,Edges_7net=100016,Edges_8net=100016,Edges_9net=100016,Edges_10net=100016)
```



SpidermiR Downstream Analysis: Case Study

Case Study n.1: Role of miRNAs in shared protein domains network in Prostate Cancer

In this case study, we downloaded shared protein domains network in Homo Sapiens, using SpidermiRquery, SpidermiRprepare, and SpidermiRdownload.

Then, we focused on role of miRNAs in this network. We integrated miRNA information using SpidermiRanalyze. We obtained a big network, and in order to understand the underlying biological process of a set of biomarker of interest (e.g. from lab) we performed an analysis to identify their neighbor biomarkers in the shared protein domains network.

SpidermiRvisualize was used to see the results.

```
org<-SpidermiRquery_species(species)
net_shar_prot<-SpidermiRquery_spec_networks(organismID = org[6,],network = "SHpd")
out_net<-SpidermiRdownload_net(net_shar_prot)
geneSymb_net<-SpidermiRprepare_NET(organismID = org[6,],data = out_net)
miRNA_complNET<-SpidermiRanalyze_mirna_gene_complnet(data=geneSymb_net,
                                                       disease="prostate cancer")

biomark_of_interest<-read.delim("C:/Users/UserInLab05/Google Drive/MIRNA AND GENEMANIA/
                                1 case study/deg_prostate.txt",header=FALSE)
subnet<-SpidermiRanalyze_direct_subnetwork(data=miRNA_complNET,BI=biomark_of_interest$V1)
comm2<- SpidermiRanalyze_Community_detection(data=subnet,type="FC")
cd_net<-SpidermiRanalyze_Community_detection_net(data=subnet,comm_det=comm2,size=2)
SpidermiRvisualize_mirnanet(data=cd_net)
```

```
miRNA_NET<-SpidermiRanalyze_mirna_network(data=geneSymb_net,disease="prostate cancer")

cd_net_miRNA<-SpidermiRanalyze_Community_detection_net(data=miRNA_NET,comm_det=comm2,size=2)

SpidermiRvisualize_mirnanet(data=cd_net_miRNA)
```

Case Study n.2: miRNAs regulating degree centrality genes in physical interactions network in breast cancer

```
org<-SpidermiRquery_species(species)
net_PHint<-SpidermiRquery_spec_networks(organismID = org[6,],network = "PHint")
out_net<-SpidermiRdownload_net(net_PHint)
geneSymb_net<-SpidermiRprepare_NET(organismID = org[6,],data = out_net)
ds<-do.call("rbind", geneSymb_net)
data1<-as.data.frame(ds[!duplicated(ds), ])

sdas<-cbind(data1$gene_symbolA,data1$gene_symbolB)

miRNA_NET<-SpidermiRanalyze_mirna_network(data=geneSymb_net,disease="breast cancer")
topwhol<-SpidermiRanalyze_degree_centrality(sdas)
top10_cent_gene<-SpidermiRanalyze_degree_centrality(miRNA_NET)
miRNA_degree<-top10_cent_gene[grepl("hsa",top10_cent_gene$dfer),]
```

References

- Csardi G, Nepusz T. 2006. "The Igraph Software Package for Complex Network Research."
- Dweep H, Sticht C, Pandey P, Gretz N. 2011. "MiRWalk - Database Prediction of Possible MiRNA Binding Sites by 'Walking' the Genes of 3 Genomes."
- Jiang Q, Wang Y, Hao Y, Juan L, Teng M, Zhang X, Li M, Wang G, Liu Y. 2009. "MiR2Disease a Manually Curated Database for MicroRNA Deregulation in Human Disease."
- Rukov J, Wilentzik R, Jaffe I, Vinther J, Shomron N. 2013. "Pharmaco MiR Linking MicroRNAs and Drug Effects."
- Russo F, Di Bella S, Nigita G, Macca V, Lagana A, Giugno R, Pulvirenti A, Ferro A. 2012. "MiRandola Extracellular Circulating MicroRNAs Database."
- Warde-Farley D, Donaldson S, Comes O, Zuberi K, Badrawi R, and others. 2010. "The Gene Mania Prediction Server Biological Network Integration for Gene Prioritization and Predicting Gene Function."