

# Package ‘SpidermiR’

November 20, 2023

**Type** Package

**Title** SpidermiR: An R/Bioconductor package for integrative network analysis with miRNA data

**Version** 1.32.0

**Date** 10-10-2022

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

**Maintainer** Claudia Cava <claudia.cava@ibfm.cnr.it>

**Depends** R (>= 3.0.0)

**Imports** httr, igraph, utils, stats, miRNAatap, miRNAatap.db,  
AnnotationDbi, org.Hs.eg.db, gdata

**Description** The aims of SpidermiR are : i) facilitate the network open-access data retrieval from GeneMania data, ii) prepare the data using the appropriate gene nomenclature, iii) integration of miRNA data in a specific network, iv) provide different standard analyses and v) allow the user to visualize the results. In more detail, the package provides multiple methods for query, prepare and download network data (GeneMania), and the integration with validated and predicted miRNA data (mir-Walk, miRTarBase, miRandola, Miranda, PicTar and TargetScan). Furthermore, we also present a statistical test to identify pharmaco-mir relationships using the gene-drug interactions derived by DGIdb and MATADOR database.

**License** GPL (>= 3)

**biocViews** GeneRegulation, miRNA, Network

**Suggests** BiocStyle, knitr, rmarkdown, testthat, devtools, roxygen2

**VignetteBuilder** knitr

**URL** <https://github.com/claudiacava/SpidermiR>

**BugReports** <https://github.com/claudiacava/SpidermiR/issues>

**RoxygenNote** 7.2.1

**NeedsCompilation** no

**git\_url** <https://git.bioconductor.org/packages/SpidermiR>

**git\_branch** RELEASE\_3\_18

**git\_last\_commit** 02f4a52

**git\_last\_commit\_date** 2023-10-24

**Repository** Bioconductor 3.18

**Date/Publication** 2023-11-19

## R topics documented:

SpidermiRanalyze_degree_centrality . . . . .	2
SpidermiRanalyze_direct_net . . . . .	3
SpidermiRanalyze_direct_subnetwork . . . . .	4
SpidermiRanalyze_subnetwork_neigh . . . . .	4
SpidermiRdownload_drug_gene . . . . .	5
SpidermiRdownload_miRNAprediction . . . . .	6
SpidermiRdownload_net . . . . .	6
SpidermiRprepare_NETWORK . . . . .	7
SpidermiRquery_disease . . . . .	8
SpidermiRquery_networks_type . . . . .	8
SpidermiRquery_species . . . . .	9
SpidermiRquery_spec_networks . . . . .	9

<b>Index</b>	<b>11</b>
--------------	-----------

---

SpidermiRanalyze\_degree\_centrality  
*Ranking degree centrality genes*

---

### Description

SpidermiRanalyze\_degree\_centrality provides degree centrality, defined as the total number of direct neighbors for each gene.

### Usage

```
SpidermiRanalyze_degree_centrality(data, cut = NULL)
```

### Arguments

data	SpidermiRanalyze_mirna_network output or SpidermiRanalyze_mirna_gene_complnet
cut	parameter cut is able to cut off other genes

**Value**

dataframe with the ranked number of direct neighbors for each gene of the network

**Examples**

```
miRNA_cN <-data.frame(gA=c('hsa-let-7a', 'hsa-miR-300'), gB=c('FOXO1', 'KPNA4'), stringsAsFactors=FALSE)
biomark_of_interest<-c("hsa-let-7a", "CDK", "FOXO1", "hsa-miR-27a")
top10_cent<-SpidermiRanalyze_degree_centrality(miRNA_cN)
```

---

SpidermiRanalyze\_direct\_net

*Searching by biomarkers of interest with direct interaction*

---

**Description**

SpidermiRanalyze\_direct\_net finds other biomarkers that are related to a set of biomarkers of interest (the input of user) with direct interactions.

**Usage**

```
SpidermiRanalyze_direct_net(data, BI)
```

**Arguments**

data	SpidermiRanalyze_mirna_network output or SpidermiRanalyze_mirna_gene_complnet
BI	a set of biomarkers of interest

**Value**

dataframe with direct interaction of biomarkers of interest

**Examples**

```
miRNA_cN <-data.frame(gA=c('hsa-let-7a', 'FOXO1'), gB=c('FOXO1', 'KPNA4'), stringsAsFactors=FALSE)
biomark_of_interest<-c("hsa-let-7a", "CDK", "FOXO1", "hsa-miR-27a")
GI_direct_net<-SpidermiRanalyze_direct_net(data=miRNA_cN, BI=biomark_of_interest)
```

---

SpidermiRanalyze\_direct\_subnetwork

*Searching by biomarkers of interest with direct interaction by ONLY the nodes in BI*

---

### Description

SpidermiRanalyze\_direct\_subnetwork creates a sub network composed by ONLY the nodes in genes of interest and the edges between them

### Usage

```
SpidermiRanalyze_direct_subnetwork(data, BI)
```

### Arguments

data	SpidermiRanalyze_mirna_network output or SpidermiRanalyze_mirna_gene_complnet
BI	a set of biomarkers of interest

### Value

dataframe with direct interaction of biomarkers of interest

### Examples

```
miRNA_cN <- data.frame(gA=c('hsa-let-7a', 'FOXO1'), gB=c('FOXO1', 'KPNA4'), stringsAsFactors=FALSE)
biomark_of_interest <- c("hsa-let-7a", "CDK", "FOXO1", "hsa-miR-27a")
subnet <- SpidermiRanalyze_direct_subnetwork(data=miRNA_cN, BI=biomark_of_interest)
```

---

SpidermiRanalyze\_subnetwork\_neigh

*Searching by biomarkers of interest and all the edges among this bunch of nodes*

---

### Description

SpidermiRanalyze\_subnetwork\_neigh 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).

### Usage

```
SpidermiRanalyze_subnetwork_neigh(data, BI)
```

**Arguments**

data SpidermiRanalyze\_mirna\_network output or SpidermiRanalyze\_mirna\_gene\_complnet  
BI a set of biomarkers of interest

**Value**

dataframe with interactions

**Examples**

```
miRNA_cN <-data.frame(gA=c('hsa-let-7a','hsa-miR-300'),gB=c('FOXO1','KPNA4'),stringsAsFactors=FALSE)
biomark_of_interest<-c("hsa-let-7a","CDK","FOXO1","hsa-miR-27a")
GIdirect_net_neigh<-SpidermiRanalyze_subnetwork_neigh(data=miRNA_cN,BI=biomark_of_interest)
```

---

SpidermiRdownload\_drug\_gene

*Download drug-gene interactions in DGldb*

---

**Description**

SpidermiRdownload\_drug\_gene will download drug gene interactions

**Usage**

```
SpidermiRdownload_drug_gene(drug_gene)
```

**Arguments**

drug\_gene parameter

**Value**

a dataframe with miRNA target validated interactions

**Examples**

```
drug_genetarget<-SpidermiRdownload_drug_gene(drug_gene)
```

SpidermiRdownload\_miRNAPrediction

*Download human miRNA predicted database*

---

**Description**

SpidermiRdownload\_miRNAPrediction will download miRNA predicted target

**Usage**

```
SpidermiRdownload_miRNAPrediction(mirna_list)
```

**Arguments**

mirna\_list      miRNA list of interest

**Value**

a dataframe with miRNA target validated interactions

**Examples**

```
mirna<-c('hsa-miR-567')  
list<-SpidermiRdownload_miRNAPrediction(mirna_list=mirna)
```

---

SpidermiRdownload\_net *Download the network from GeneMania.*

---

**Description**

SpidermiRdownload\_net function will download the data

**Usage**

```
SpidermiRdownload_net(data)
```

**Arguments**

data              The SpidermiRquery\_spec\_networks output

**Value**

Download GeneMania network

**Examples**

```
org<-SpidermiRquery_species(species)
net_shar_prot<-SpidermiRquery_spec_networks(organismID = org[9,],
network = "SHpd")
out_net<-SpidermiRdownload_net(data=net_shar_prot)
```

---

SpidermiRprepare\_NET *Prepare matrix of gene network from Genamania with Ensembl Gene ID, and gene symbols*

---

**Description**

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

**Usage**

```
SpidermiRprepare_NET(organismID, data)
```

**Arguments**

organismID	is the index of SpidermiRquery_spec_networks output
data	is the output of function SpidermiRdownload_net

**Value**

A list of tables.

**Examples**

```
org<-SpidermiRquery_species(species)
net_shar_prot<-SpidermiRquery_spec_networks(organismID = org[9,],
network = "SHpd")
out_net<-SpidermiRdownload_net(data=net_shar_prot)
geneSymb_net<-SpidermiRprepare_NET(organismID = org[9,],
data = out_net)
```

SpidermiRquery\_disease

*Visualize disease categories*

---

**Description**

The user can visualize the disease supported by SpidermiR

**Usage**

```
SpidermiRquery_disease(diseaseID)
```

**Arguments**

diseaseID      variable name

**Value**

a list of disease.

**Examples**

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

---

SpidermiRquery\_networks\_type

*Network categories*

---

**Description**

The user can visualize the network types supported by GeneMania for a specific specie using SpidermiRquery\_networks\_type

**Usage**

```
SpidermiRquery_networks_type(organismID)
```

**Arguments**

organismID      describes index of a specific specie obtained by SpidermiRquery\_species output

**Value**

a list of network categories in a specie indicated.



### Examples

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

---

SpidermiRquery\_species

*Searching by network species*

---

### Description

The user can visualize the species supported by GeneMania, using the function SpidermiRquery\_species

### Usage

```
SpidermiRquery_species(species)
```

### Arguments

species            a variable parameter

### Value

table of species

### Examples

```
org<-SpidermiRquery_species(species)
```

---

SpidermiRquery\_spec\_networks

*Searching by network categories*

---

### Description

The user can visualize the database or reference where the information came from

### Usage

```
SpidermiRquery_spec_networks(organismID, network)
```



# Index

SpidermiRanalyze\_degree centrality, 2  
SpidermiRanalyze\_direct\_net, 3  
SpidermiRanalyze\_direct\_subnetwork, 4  
SpidermiRanalyze\_subnetwork\_neigh, 4  
SpidermiRdownload\_drug\_gene, 5  
SpidermiRdownload\_miRNAprediction, 6  
SpidermiRdownload\_net, 6  
SpidermiRprepare\_NET, 7  
SpidermiRquery\_disease, 8  
SpidermiRquery\_networks\_type, 8  
SpidermiRquery\_spec\_networks, 9  
SpidermiRquery\_species, 9