

# Package ‘grndata’

July 11, 2024

**Type** Package

**Title** Synthetic Expression Data for Gene Regulatory Network Inference

**Version** 1.37.0

**Date** 2014-12-01

**Author** Pau Bellot, Catharina Olsen, Patrick E Meyer

**Maintainer** Pau Bellot <pau.bellot@upc.edu>

**Description** Simulated expression data for five large Gene Regulatory Networks from different simulators

**LazyLoad** yes

**LazyData** yes

**License** GPL-3

**Suggests** RUnit, BiocGenerics, knitr

**VignetteBuilder** knitr

**biocViews** ExperimentData, NetworkInference, GeneExpression, Microarray, GeneRegulation, Network

**Depends** R (>= 2.10)

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

**git\_branch** devel

**git\_last\_commit** 62f3d39

**git\_last\_commit\_date** 2024-04-30

**Repository** Bioconductor 3.20

**Date/Publication** 2024-07-11

## Contents

Availabledata . . . . .	2
getData . . . . .	3
gnw1565.data . . . . .	4
gnw1565.net . . . . .	5

gnw2000.data . . . . .	6
gnw2000.net . . . . .	7
rogers1000.data . . . . .	8
rogers1000.net . . . . .	9
syntren1000.data . . . . .	10
syntren1000.net . . . . .	11
syntren300.data . . . . .	12
syntren300.net . . . . .	13
toy.data . . . . .	14
toy.net . . . . .	15
<b>Index</b>	<b>16</b>

---

Availabledata

*Available datasources*

---

## Description

Availabledata contains a character vector with the names of the datasources that the package contains.

## Usage

```
data("Availabledata")
```

## Format

A character vector with the names of the datasources.

## Value

character vector with the names of the datasources.

## Examples

```
data(Availabledata)
cat(Availabledata)
```

---

getData	<i>Datasource and underlying network loading</i>
---------	--

---

### Description

Function to load a gene expression datasource and the network that have generate it.

### Usage

```
getData(datasource.name=NULL, getNet=TRUE)
```

### Arguments

datasource.name	A character containing the name of network datasources to (default: NULL).
getNet	Logical specifying if the true underlying network should be returned (default: TRUE).

### Details

If `getNet` is `TRUE` the function will return a list with two components:

1. A data.frame with the specified datasoruce.
2. A matrix with the true underlying network.

### Value

The function `getData` loads the specified datasource by `datasource.name` and its true underlying network if it is specified by `getNet`.

### Examples

```
# Get only datasource
data <- getData(datasource.name="syntren300", getNet=FALSE)
# Get the datasource and network in a list
mydata <- getData(datasource.name="syntren1000")
data <- mydata[[1]]
net <- mydata[[2]]
```

---

`gnw1565.data`*Simulated Gene Expression Data*

---

**Description**

Dataset containing 1565 samples and 1565 genes generated by the publicly available GNW generator using an ecoli source network with no added noise.

**Usage**

```
data(gnw1565.data)
```

**Format**

`gnw1565.data` is a data frame containing 1565 rows and 1565 columns. Each row contains a simulated microarray experiment and each column contains a gene. The dataset was generated with GNW generator which relies on ODEs for modeling gene regulation.

**Value**

data.frame containing the data.

**Source**

GNW v3.1 with source network : Ecoli

**References**

Schaffter, Thomas, Daniel Marbach, and Dario Floreano. "GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods." *Bioinformatics* 27.16 (2011): 2263-2270.

**See Also**

[gnw1565.net](#), [gnw2000.data](#), [rogers1000.data](#), [syntren300.data](#), [syntren1000.data](#)

**Examples**

```
data(gnw1565.data)
# Print size
print(dim(gnw1565.data))
```

---

`gnw1565.net`*Simulated Gene Expression Data*

---

**Description**

True underlying network used to generate the dataset at [gnw1565.data](#).

**Usage**

```
data(gnw1565.net)
```

**Format**

`true.net` is the true underlying network used to generate the dataset loaded at [gnw1565.data](#).

**Value**

matrix containing underlying network.

**Source**

GNW v3.1 with source network : Ecoli

**References**

Schaffter, Thomas, Daniel Marbach, and Dario Floreano. "GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods." *Bioinformatics* 27.16 (2011): 2263-2270.

**See Also**

[gnw1565.data](#), [gnw2000.data](#), [rogers1000.data](#), [syntren300.data](#), [syntren1000.data](#)

**Examples**

```
data(gnw1565.net)
# Number of directed edges
nEdges<- sum(gnw1565.net)
```

---

`gnw2000.data`*Simulated Gene Expression Data*

---

**Description**

Dataset containing 2000 samples and 2000 genes generated by the publicly available GNW generator using a yeast source network with no added noise.

**Usage**

```
data(gnw2000.data)
```

**Format**

`gnw2000.data` is a data frame containing 2000 rows and 2000 columns. Each row contains a simulated microarray experiment and each column contains a gene. The dataset was generated with GNW generator which relies on ODEs for modeling gene regulation.

**Value**

data.frame containing the data.

**Source**

GNW v3.1 with source network : Yeast

**References**

Schaffter, Thomas, Daniel Marbach, and Dario Floreano. "GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods." *Bioinformatics* 27.16 (2011): 2263-2270.

**See Also**

[gnw2000.net](#), [gnw1565.data](#), [rogers1000.data](#), [syntren300.data](#), [syntren1000.data](#)

**Examples**

```
data(gnw2000.data)
# Print size
print(dim(gnw2000.data))
```

---

`gnw2000.net`*Simulated Gene Expression Data*

---

**Description**

True underlying network used to generate the dataset at [gnw2000.data](#).

**Usage**

```
data(gnw2000.net)
```

**Format**

`true.net` is the true underlying network used to generate the dataset loaded at [gnw2000.data](#).

**Value**

matrix containing underlying network.

**Source**

GNW v3.1 with source network : Yeast

**References**

Schaffter, Thomas, Daniel Marbach, and Dario Floreano. "GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods." *Bioinformatics* 27.16 (2011): 2263-2270.

**See Also**

[gnw2000.data](#), [gnw1565.data](#), [rogers1000.data](#), [syntren300.data](#), [syntren1000.data](#)

**Examples**

```
data(gnw2000.net)
# Number of directed edges
nEdges<- sum(gnw2000.net)
```

---

rogers1000.data      *Simulated Gene Expression Data*

---

### Description

Dataset containing 1000 samples and 1000 genes generated by the publicly available generator using a power-law tail topology network with no added noise.

### Usage

```
data(rogers1000.data)
```

### Format

rogers1000.data is a data frame containing 1000 rows and 1000 columns. Each row contains a simulated microarray experiment and each column contains a gene.

### Value

data.frame containing the data.

### Source

sRogers with Power-law tail topology

### References

Rogers, Simon, and Mark Girolami. "A Bayesian regression approach to the inference of regulatory networks from gene expression data." *Bioinformatics* 21.14 (2005): 3131-3137.

### See Also

[rogers1000.net](#), [gnw1565.data](#), [gnw2000.data](#), [syntren300.data](#), [syntren1000.data](#)

### Examples

```
data(rogers1000.data)
# Print size
print(dim(rogers1000.data))
```



---

rogers1000.net	<i>Simulated Gene Expression Data</i>
----------------	---------------------------------------

---

**Description**

True underlying network used to generate the dataset at [rogers1000.data](#).

**Usage**

```
data(rogers1000.net)
```

**Format**

true.net is the true underlying network used to generate the dataset loaded at [rogers1000.data](#).

**Value**

matrix containg underlying network.

**Source**

sRogers with Power-law tail topology

**References**

Rogers, Simon, and Mark Girolami. "A Bayesian regression approach to the inference of regulatory networks from gene expression data." *Bioinformatics* 21.14 (2005): 3131-3137.

**See Also**

[rogers1000.data](#), [gnw1565.data](#), [gnw2000.data](#), [syntren300.data](#), [syntren1000.data](#)

**Examples**

```
data(rogers1000.net)
# Number of directed edges
nEdges<- sum(rogers1000.net)
```

---

syntren1000.data      *Simulated Gene Expression Data*

---

### Description

Dataset containing 1000 samples and 1000 genes generated by the publicly available SynTReN generator using an ecoli source network.

### Usage

```
data(syntren1000.data)
```

### Format

syntren1000.data is a data frame containing 1000 rows and 1000 columns. Each row contains a simulated microarray experiment and each column contains a gene.

### Value

data.frame containing the data.

### Source

SynTReN 1.1.3 with source network : ecoli\\_nn.sif

### References

Van den Bulcke, Tim, et al. "SynTReN: a generator of synthetic gene expression data for design and analysis of structure learning algorithms." *BMC Bioinformatics* 7.1 (2006): 43.

### See Also

[gnw1565.data](#), [gnw2000.data](#), [rogers1000.data](#), [syntren300.data](#)

### Examples

```
data(syntren1000.data)
# Print size
print(dim(syntren1000.data))
```

---

syntren1000.net	<i>Simulated Gene Expression Data</i>
-----------------	---------------------------------------

---

**Description**

True underlying network used to generate the dataset at [syntren1000.data](#).

**Usage**

```
data(syntren1000.net)
```

**Format**

syntren1000.net is the true underlying network used to generate the dataset loaded at [syntren1000.data](#).

**Value**

matrix containg underlying network.

**Source**

SynTReN 1.1.3 with source network : ecoli\\_nn.sif

**References**

Van den Bulcke, Tim, et al. "SynTReN: a generator of synthetic gene expression data for design and analysis of structure learning algorithms." *BMC Bioinformatics* 7.1 (2006): 43.

**See Also**

[syntren1000.data](#), [gnw1565.data](#), [gnw2000.data](#), [rogers1000.data](#), [syntren300.data](#)

**Examples**

```
data(syntren1000.net)
# Number of directed edges
nEdges<- sum(syntren1000.net)
```

---

syntren300.data      *Simulated Gene Expression Data*

---

### Description

Dataset containing 800 samples and 300 genes generated by the publicly available SynTReN generator using an ecoli source network.

### Usage

```
data(syntren300.data)
```

### Format

syntren300.data is a data frame containing 800 rows and 300 columns. Each row contains a simulated microarray experiment and each column contains a gene.

### Value

data.frame containing the data.

### Source

SynTReN 1.1.3 with source network : ecoli\_nn.sif

### References

Van den Bulcke, Tim, et al. "SynTReN: a generator of synthetic gene expression data for design and analysis of structure learning algorithms." *BMC Bioinformatics* 7.1 (2006): 43.

### See Also

[syntren300.net](#), [syntren1000.data](#), [rogers1000.data](#), [gnw1565.data](#), [gnw2000.data](#)

### Examples

```
data(syntren300.data)
# Print size
print(dim(syntren300.data))
```

---

syntren300.net	<i>Simulated Gene Expression Data</i>
----------------	---------------------------------------

---

**Description**

True underlying network used to generate the dataset at [syntren300.data](#).

**Usage**

```
data(syntren300.net)
```

**Format**

true.net is the true underlying network used to generate the dataset loaded at [syntren300.data](#).

**Value**

matrix containg underlying network.

**Source**

SynTReN 1.1.3 with source network : ecoli\\_nn.sif

**References**

Van den Bulcke, Tim, et al. "SynTReN: a generator of synthetic gene expression data for design and analysis of structure learning algorithms." BMC bioinformatics 7.1 (2006): 43.

**See Also**

[syntren300.data](#), [syntren1000.data](#), [rogers1000.data](#), [gnw1565.data](#), [gnw2000.data](#)

**Examples**

```
data(syntren300.net)
# Number of directed edges
nEdges<-sum(syntren300.net)
```

---

`toy.data`*Simulated Toy example of Gene Expression Data*

---

**Description**

Dataset containing 64 samples and 64 genes generated by the publicly available GNW generator using an ecoli source network.

**Usage**

```
data(toy.data)
```

**Format**

`toy.data` is a data frame containing 64 rows and 64 columns. Each row contains a simulated microarray experiment and each column contains a gene.

**Value**

data.frame containing the data.

**Source**

GNW v3.1 with source network : Example (Ecoli)

**References**

Schaffter, Thomas, Daniel Marbach, and Dario Floreano. "GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods." *Bioinformatics* 27.16 (2011): 2263-2270.

**See Also**

[toy.net](#)

**Examples**

```
data(toy.data)
# Print size
print(dim(toy.data))
```

---

`toy.net`*Simulated Toy example of Gene Expression Data*

---

**Description**

True underlying network used to generate the dataset at [toy.data](#).

**Usage**

```
data(toy.net)
```

**Format**

`toy.net` is the true underlying network used to generate the dataset loaded at [toy.data](#).

**Value**

matrix containg underlying network.

**Source**

GNW v3.1 with source network : Example (Ecoli)

**References**

Schaffter, Thomas, Daniel Marbach, and Dario Floreano. "GeneNetWeaver: in silico benchmark generation and performance profiling of network inference methods." *Bioinformatics* 27.16 (2011): 2263-2270.

**See Also**

[toy.data](#)

**Examples**

```
data(toy.net)
# Print size
nEdges<- sum(toy.net)
```

# Index

## \* datasets

Availabledata, 2  
gnw1565.data, 4  
gnw1565.net, 5  
gnw2000.data, 6  
gnw2000.net, 7  
rogers1000.data, 8  
rogers1000.net, 9  
syntren1000.data, 10  
syntren1000.net, 11  
syntren300.data, 12  
syntren300.net, 13  
toy.data, 14  
toy.net, 15

## \* misc

getData, 3

Availabledata, 2

getData, 3

gnw1565.data, 4, 5–13

gnw1565.net, 4, 5

gnw2000.data, 4, 5, 6, 7–13

gnw2000.net, 6, 7

rogers1000.data, 4–7, 8, 9–13

rogers1000.net, 8, 9

syntren1000.data, 4–9, 10, 11–13

syntren1000.net, 11

syntren300.data, 4–11, 12, 13

syntren300.net, 12, 13

toy.data, 14, 15

toy.net, 14, 15