

# Package: kinograte (via r-universe)

November 20, 2024

**Type** Package

**Title** Kinograte: Network-based multi-omics Integration

**Version** 0.0.2.9000

**Description** Network-based multi-omics integration using a prize-collecting Steiner forest (PCSF) algorithm.

**License** MIT + file LICENSE

**Encoding** UTF-8

**Roxygen** list(markdown = TRUE)

**RoxygenNote** 7.1.2

**Imports** dplyr, echarts4r, ggplot2, igraph, magrittr, PCSF, reactable, visNetwork

**VignetteBuilder** knitr

**Suggests** knitr, rmarkdown

**Depends** R (>= 2.10)

**LazyData** true

**Config/pak/sysreqs** libglpk-dev make libicu-dev libpng-dev libxml2-dev libssl-dev zlib1g-dev

**Repository** <https://cogdisreslab.r-universe.dev>

**RemoteUrl** <https://github.com/CogDisResLab/kinograte>

**RemoteRef** HEAD

**RemoteSha** 28859ba6f5384ea7c99d7d7f42a3cf87070424c8

## Contents

combine_scores . . . . .	2
kinograte . . . . .	2
kinomics_exmample . . . . .	3
network_enrichment . . . . .	4
percentile_rank . . . . .	4
ppi_network_example . . . . .	5

proteomics_exmaple . . . . .	5
rnaseq_example . . . . .	6
score_plot . . . . .	6
top_hits . . . . .	7
visualize_network . . . . .	7

<b>Index</b>	<b>9</b>
--------------	----------

---

combine_scores	<i>Combines standardized scores</i>
----------------	-------------------------------------

---

### Description

This function ranks combines standardized scores from each omic ranked dataset

### Usage

```
combine_scores(df_rna = NULL, df_prot = NULL, df_kin = NULL, tf_kin = NULL)
```

### Arguments

df_rna	dataframe with percentile ranking of RNA features
df_prot	dataframe with percentile ranking of protein features
df_kin	dataframe with percentile ranking of kinase features
tf_kin	dataframe with percentile ranking of transcription factor features

### Value

dataframe with combined scores

### Examples

```
TRUE
```

---

kinorate	<i>Integrate omic datasets using PCSF</i>
----------	---

---

### Description

This function integrates omic datasets using prize-collecting Steiner forest (PCSF) algorithm

**Usage**

```
kinorate(  
  df,  
  ppi_network,  
  n = 10,  
  w = 10,  
  r = 0.1,  
  b = 2,  
  mu = 0.005,  
  cluster = TRUE,  
  seed = NULL  
)
```

**Arguments**

df	combined ranked omic datasets
ppi_network	dataframe of protein-protein interactions
n	An integer value to determine the number of runs with random noise added edge costs. A default value is 10.
w	A numeric value for tuning the number of trees in the output. A default value is 2.
r	A numeric value to determine additional random noise to edge costs. A random noise upto r percent of the edge cost is added to each edge. A default value is 0.1
b	A numeric value for tuning the node prizes. A default value is 1.
mu	A numeric value for a hub penalization. A default value is 0.0005.
cluster	set TRUE to cluster the network
seed	(optional) set seed number

**Value**

list(network, nodes, edges)

---

kinomics\_exmaple      *an example of Kinome data*

---

**Description**

an example of Kinome data

**Usage**

```
kinomics_exmaple
```

**Format**

A data frame with 234 rows and 2 variables:

**hgnc\_symbol** Gene Symbols of kinases

**score** normalized score

---

network_enrichment	<i>Network pathway analysis</i>
--------------------	---------------------------------

---

**Description**

This function performs pathway enrichment analysis on the integrated network

**Usage**

```
network_enrichment(network, ...)
```

**Arguments**

network	integrated network
...	arguments passed to PCSF::enrichment_analysis()

**Value**

df enriched terms

---

percentile_rank	<i>Ranks genes/proteins using a percentile ranking for a selected variable</i>
-----------------	--

---

**Description**

This function ranks genes or proteins using a percentile ranking for a selected variable (for example, fold change or pvalue)

**Usage**

```
percentile_rank(df, symbol, metric, desc = FALSE)
```

**Arguments**

df	dataframe that contains genes/proteins to rank (tidy format)
symbol	column name that contains the gene/protein symbols
metric	column name to be used as the metric to rank
desc	boolean, ranking in a descending or ascending order. Default = FALSE

**Value**

dataframe with percentile ranking

**Examples**

TRUE

---

`ppi_network_example`    *an example of ppi network*

---

**Description**

an example of ppi network

**Usage**

`ppi_network_example`

**Format**

A data frame with 175205 rows and 3 variables:

**head** Protein 1

**tail** Protein 2

**cost** inverse degree of confidence

---

`proteomics_exmaple`    *an example of proteomics data*

---

**Description**

an example of proteomics data

**Usage**

`proteomics_exmaple`

**Format**

A data frame with 141 rows and 2 variables:

**Gene.Symbol** Gene Symbols

**LFC** log2 fold change

---

rnaseq_example	<i>an example of RNA differential gene expression data</i>
----------------	--

---

**Description**

an example of RNA differential gene expression data

**Usage**

```
rnaseq_example
```

**Format**

A data frame with 3207 rows and 3 variables:

**Gene name** Gene Symbols

**LFC** log2 fold change

**P\_Value** pvalue

---

score_plot	<i>Plot the normalized score</i>
------------	----------------------------------

---

**Description**

This function plots the normalized score. Two options available: static or interactive plot

**Usage**

```
score_plot(
  df,
  prec_cutoff = 0.8,
  title = "Score Plot",
  subtitle = "",
  interactive = T
)
```

**Arguments**

df	dataframe that contains ranked genes/proteins
prec_cutoff	the percentile cutoff
title	plot title
subtitle	plot subtitle
interactive	boolean, option for an interactive plot. Default = TRUE

**Value**

dataframe of the top hits

---

top_hits	<i>Extracts top hits (genes/proteins) based on the normalized score</i>
----------	---

---

**Description**

This function extracts the top hits (genes/proteins) based on the normalized score which is the percentile rank using an adjustable cutoff

**Usage**

```
top_hits(df, prec_cutoff, omic_type)
```

**Arguments**

df	dataframe that contains ranked genes/proteins
prec_cutoff	the percentile cutoff
omic_type	name of omic dataset (eg. RNA, Protein, Kinase, ... etc)

**Value**

dataframe of the top hits

**Examples**

```
TRUE
```

---

visualize_network	<i>Visualize integrated network</i>
-------------------	-------------------------------------

---

**Description**

This function visualizes the integrated results with an interactive network

**Usage**

```
visualize_network(  
  nodes,  
  edges,  
  cluster_df = NULL,  
  layout = "layout_with_fr",  
  seed = 123,  
  options_by = "group"  
)
```

**Arguments**

nodes	network nodes
edges	network edges
cluster_df	(optional) network clusters dataframe generated by the kinorate function
layout	layout option from igraph. default = "layout_with_fr". <a href="#">See full list</a>
seed	(optional) set seed
options_by	(optional) set dropdown menu, "group" or "cluster". Set NULL to remove dropdown menu

**Value**

visNetwork object



# Index

## \* datasets

- kinomics\_exmaple, 3
- ppi\_network\_example, 5
- proteomics\_exmaple, 5
- rnaseq\_example, 6

combine\_scores, 2

kinograte, 2

kinomics\_exmaple, 3

network\_enrichment, 4

percentile\_rank, 4

ppi\_network\_example, 5

proteomics\_exmaple, 5

rnaseq\_example, 6

score\_plot, 6

top\_hits, 7

visualize\_network, 7