Package 'linkprediction'

October 13, 2022

Title Link Prediction Methods

Version 1.0-0

Description Implementations of most of the existing proximity-based methods of
link prediction in graphs. Among the 20 implemented methods are e.g.:
Adamic L. and Adar E. (2003) <doi:10.1016/S0378-8733(03)00009-1>,
Leicht E., Holme P., Newman M. (2006) <doi:10.1103/PhysRevE.73.026120>,
Zhou T. and Zhang Y (2009) <doi:10.1140/epjb/e2009-00335-8>, and
Fouss F., Pirotte A., Renders J., and Saerens M. (2007) <doi:10.1109/TKDE.2007.46>.

Depends R (>= 3.1.1),

License MIT + file LICENSE

LazyData true

Imports igraph, intergraph

Suggests knitr, testthat

RoxygenNote 6.1.0

VignetteBuilder knitr

BugReports https://github.com/recon-icm/linkprediction/issues

URL https://github.com/recon-icm/linkprediction

NeedsCompilation no

Author Michal Bojanowski [aut, cre],
Bartosz Chrol [aut],
National Science Centre [fnd] (grant 2012/07/D/HS6/01971)

Maintainer Michal Bojanowski <michal2992@gmail.com>

Repository CRAN

Date/Publication 2018-10-19 13:40:03 UTC

R topics documented:

---

linkprediction   *Link Prediction Methods*

Description

Implements most of existing methods proximity-based methods of link prediction in graphs. See proxfun.

Note

Authors thank (Polish) National Science Centre for support through SONATA grant 2012/07/D/HS6/01971 for the project *Dynamics of Competition and Collaboration in Science: Individual Strategies, Collaboration Networks, and Organizational Hierarchies* (recon.icm.edu.pl).

---

proxfun   *Vertex proximity indexes*

Description

General function for calculating several types of vertex proximities in a graph.

Usage

```
proxfun(graph, ...)

## S3 method for class 'igraph'
proxfun(graph, method, v1 = NULL, v2 = v1,
  value = c("matrix", "edgelist", "graph"), ...)

## S3 method for class 'network'
proxfun(graph, method, v1 = NULL, v2 = v1,
  value = c("matrix", "edgelist", "graph"), ...)
```

Arguments

graph	an object of class igraph or network
...	additional arguments specific for a selected measure
method	single character, the method to be used, see Details
v1, v2	vectors of vertices between which similarity will be calculated. Character vector is interpreted as vertex names. Numeric vector as vertex ids.
value	a character string giving a type of the object that should be returned. This must be one of "matrix", "graph" or "edgelist", with default "matrix".

Details

This function calculates vertex proximities in graph `graph` with the selected `method`. The `graph` has to be undirected and connected. Some of the methods support computation only for selected vertices, which should be more efficient when needed. Supplying vertex IDs or names (if present in the `graph`) to `v1` and `v2` will calculate proximities of $v1 x v2$.

The following `methods` are available (see `vignette("proxfun",package="linkprediction")` for more details and formal definitions):

`aa` Adamic-Adar index (Adamic and Adar 2001). Additional arguments are passed to igraph::similarity.

`act` Average Commute Time (Fouss, Pirotte, Renders, and Saerens 2007)

`act_n` Normalized Average Commute Time (Fouss et al. 2007)

`cn` Common Neighbours

`cos` Cosine similarity (Salton and McGill 1986)

`cos_l` cosine similarity on L+ (Fouss et al. 2007)

`dist` graph distance

`hdi` Hub Depressed Index (Ravasz, Somera, Mongru, Oltvai, and Barabasi 2002)

`hpi` Hub Promoted Index (Ravasz et al. 2002)

`jaccard` Jaccard coefficient (Jaccard 1912)

`katz` Katz index (Katz 1953)

`l` L+ directly (Fouss et al. 2007)

`lhn_local` Leicht-Holme-Newman Index (Leicht, Holme, and Newman 2006)

`lhn_global` Leicht-Holme-Newman Index global version (Leicht et al. 2006)

`lp` Local Path Index (Zhou, Lu, and Zhang 2009)

`mf` Matrix Forest Index (Chebotarev P. Yu. 1997)

`pa` preferential attachment (Barabasi and Albert 1999)

`ra` resource allocation (Zhou et al. 2009)

`rwr` random walk with restart (Brin and Page 1998). Additional argument `alpha` (default value 0.3) is the probability that the walk will restart after a step.

`sor` sorensen index/dice coefficient (Sorensen 1948)

Value

If `value = "matrix"` a matrix with `length(v1)` rows and `length(v2)` with `rownames` and `colnames` equal to `v1` and `v2` respectively. If `value = "edgelist"` a `data.frame` with three columns:

from ID of a start node of an edge

to ID of an end node of an edge

value similarity score for that edge

Edges with similarity score 0 are omitted. If `value = "graph"` an object of class `igraph` or `network`, depending on the class of input graph. Returned graph has the same structure (graph and node attributes, etc.) as the input graph, except for edges - original edges are skipped, and new edges with positive similarity score are added. Edged attribute "weight" indicates similarity score.

References

Adamic L and Adar E (2003). "Friends and Neighbors on the Web." *Social Networks*, *25*, pp. 211-230 doi: 10.1016/S03788733(03)000091.

Barabasi A and Albert R (1999). "Emergence of Scaling in Random Networks." *Science*, *286*(5439), pp. 509-512.

Brin S and Page L (1998). "The anatomy of a large-scale hypertextual Web search engine ." _Computer Networks and ISDN Systems _, *30*(1-7), pp. 107 - 117. Proceedings of the Seventh International World Wide Web Conference .

Chebotarev P. Yu. SEV (1997). "The matrix-forest theorem and measuring relations in small social groups ." _Automation and Remote Control _, *58*(9), pp. 1505-1514.

Fouss F, Pirotte A, Renders J and Saerens M (2007). "Random-Walk Computation of Similarities Between Nodes of a Graph with Application to Collaborative Recommendation." *IEEE Transactions on Knowledge and Data Engineering*, *19*(3), pp. 355-369 doi: 10.1109/TKDE.2007.46.

Jaccard P (1912). "The Distribution of the Flora in the Alpine Zone 1" *New Phytologist*, *11*(2), pp. 37-50.

Katz L (1953). "A new status index derived from sociometric analysis." *Psychometrika*, *18*(1), pp. 39-43.

Leicht EA, Holme P and Newman MEJ (2006). "Vertex similarity in networks." *Phys. Rev. E*, *73*(2), pp. 026120 doi: 10.1103/PhysRevE.73.026120.

Ravasz E, Somera AL, Mongru DA, Oltvai ZN and Barabasi A (2002). "Hierarchical Organization of Modularity in Metabolic Networks." *Science*, *297*(5586), pp. 1551-1555.

Salton G and McGill MJ (1986). *Introduction to Modern Information Retrieval*. McGraw-Hill, Inc., New York, NY, USA.

Sorensen T (1948). "A Method of Establishing Groups of Equal Amplitude in Plant Sociology Based on Similarity of Species Content and Its Application to Analyses of the Vegetation on Danish Commons." *Biologiske Skrifter*, *5*, pp. 1-34.

Zhou T, Lu L and Zhang Y (2009). "Predicting missing links via local information." *The European Physical Journal B*, *71*(4), pp. 623-630 doi: 10.1140/epjb/e2009003358.

Examples

```
if(requireNamespace("igraph")) {
  g <- igraph::make_graph(~ A -- C:D:E -- B -- F -- G:H -- I)

# Adamic-Adar
proxfun(g, method="aa", value="edgelist")

# Random Walk with Restart
proxfun(g, method="rwr", value="edgelist")
}
```

Index