

Gaining information about graph structure

<code>vcount(g)</code>	count the number of vertices
<code>ecount(g)</code>	count the number of edges
<code>neighbors(g, v)</code>	list the neighbors of a vertex
<code>incident(g, v)</code>	list the incident edges of a vertex
<code>is.directed(g)</code>	determine whether the graph is directed
<code>are.connected(-g, v1, v2)</code>	determine whether there is an edge between two vertices
<code>get.edge(g, id)</code>	get endpoints of an edge
<code>get.edges(g, es)</code>	get endpoints of a list of edges

Graph, vertex and edge attributes

In the following functions, ? can be graph, vertex or edge.

<code>get.?.attribute(g, n)</code>
<code>set.?.attribute(g, n, v)</code>
<code>list.?.attributes(g)</code>
<code>?.attributes(g)</code>
<code>remove.?.attribute(g, n)</code>

Various methods for creating graphs

<code>graph.empty()</code>
<code>graph.star(n)</code>
<code>graph.lattice(c(n, m))</code>
<code>graph.ring(n)</code>
<code>graph.tree(n)</code>
<code>graph.full(n)</code>
<code>graph.full.citation(n)</code>
<code>graph.atlas(0-1252)</code>
<code>graph(c(1, 2, 2, 3, 3, 4, ...))</code>
<code>graph.edgelist(edge.matrix)</code>
<code>graph.formula(1-2, 3, 4-+5)</code>

Vertex and edge sequences and iterators

<code>V(g)</code>	list vertices in a <code>igraph.vs</code> object
<code>V(g)\$number</code>	get or set vertices properties
<code>V(g)[number<50]</code>	get a subset of vertices
<code>E(g)</code>	list edges in a <code>igraph.es</code> object

Method for structural manipulation of graphs

<code>g[]</code>	get and set adjacency matrix
<code>g[][]</code>	get adjacency list

Degree and degree distribution of the vertices

<code>degree(g)</code>
<code>degree.distribution(g)</code>

Graph Algebra

<code>graph.union(g1, g2)</code>
<code>graph.difference(g1, g2)</code>

Print Graph

<code>summary(g)</code>	summary
<code>str(g)</code>	summary with edge list
<code>print(g)</code>	customizable printing function

Neighborhood of graph vertices

<code>neighborhood.size(g, o)</code>	gives a list of neighborhood size for each vertex
<code>neighborhood(g, o)</code>	gives a list of neighborhood vertices for each vertex
<code>graph.neighborhood(g, o)</code>	gives the neighborhood graphs
<code>connect.neighborhood(g, o)</code>	creates a new graph by connect each vertex with its neighbor vertices



By **zhuxun2** (trvoldemort)

Published 2nd October, 2014.

Last updated 13th May, 2016.

Page 1 of 1.

Sponsored by **Readable.com**

Measure your website readability!

<https://readable.com>