New features for graph study

Cuong Nguyen
Grinnell College

me again,
LaTeX Academy

February 22, 2014

Old features

1. 39 binary columns of survival in the previous period
2. 1 weight in G40
3. 1 percent of 1
4. 1 number of switch
5. 1 count to last 0

New features

1. The number of degrees in the previous graph(s) Let the degree = 0 if the edges does not exist
2. Sum of the degree of the second level of neighbor nodes
3. Sum of the degree of the third level of neighbor nodes
4. Google Pagerank of a vertices
5. The neighborhood that this edge/node belong to

New functions to learn

1. get.adjlist: Create adjacency lists from a graph, either for adjacent edges or for neighboring vertices.
2. get.edgeids:
3. components: Finds all vertices reachable from a given vertex, or the opposite: all vertices from which a given vertex is reachable via a directed path.
4. neighborhood: These functions find the vertices not farther than a given limit from another fixed vertex, these are called the neighborhood of the vertex.

5. cohesive.blocks
6. communities: igraph community detection functions return their results as an object from the communities class. This manual page describes the operations of this class.
7. community.to.membership: community.to.membership takes a merge matrix, a typical result of community structure detection algorithms and creates a membership vector by performing a given number of merges in the merge matrix.
8. clusters: Calculate the maximal (weakly or strongly) connected components of a graph
9. edge.betweenness.community: Community structure detection based on edge betweenness
10. evcent: Eigen vector centrality
11. fastgreedy.community: Community structure via greedy optimization of modularity
12. graph.coreness: The k-core of graph is a maximal subgraph in which each vertex has at least degree k. The coreness of a vertex is k if it belongs to the k-core but not to the (k+1)-core.
13. vertex.connectivity: The vertex connectivity of a graph or two vertices, this is recently also called group cohesion.

How am I implement all of this?

Don't know...