CSCI 347 Cheat Sheet: Graph Data Analysis

Adjacency matrix of a graph G:

$$A_{ij} = \begin{cases} 1 & \text{if there is an edge from vertex } x_i \text{ to vertex } x_j \\ 0 & \text{otherwise} \end{cases}$$

Creating a graph from a data matrix:
A common way to create edges between data instances is to use the similarity function: $sim(x_i, x_j) = e^{\frac{-||x_i - x_j||^2}{2\sigma^2}}$ and then to create an edge between x_i and x_j if the similarity $sim(x_i, x_j)$ is greater than a specified threshold.

Degree of a node x_i :

$$d(x_i) = \sum_{j=1}^{n} A_{ij}$$ where A_{ij} is entry in row i and column j in the adjacency matrix A

Eccentricity of a node x_i :

$e(x_i) = \max_j \{d(x_i, x_j)\}$ where $d(x_i, x_j)$ is the shortest path length between x_i and x_j

Betweenness centrality of a node :

$$bc(x_i) = \sum_{j \neq i} \sum_{k \neq i} \frac{\eta_{jk}(x_i)}{\eta_{jk}}$$

where η_{jk} is the number of shortest paths between x_i and x_j.

Eigenvector centrality of a node x_i is the value of the *i*th entry in the dominant eigenvector *v* of the transposed adjacency matrix of the graph. Using power iteration to solve the equation $p = A^T p$ will converge to *v*.

Pagerank of a node x_i is the value of the *i*th entry of the vector *p* that solves the equation: $p = (1 - \alpha)N^T p + \alpha N_r p$, where N is the normalized adjacency matrix and N_r is the matrix that assigns equal probability to all nodes in the graph. We can think of pagerank as the probability that a user lands on a node when taking a random walk on the graph, where each edge is traveled with probability proportional to the node degree, and where there is a probability α that the walk jumps to a random node in the network.

Clustering coefficient of a node x_i is:

$$\frac{m_i}{\binom{n_i}{2}} = \frac{\text{number of edges among neighbors of } x_i}{\text{number of possible edges among neighbors of } x_i}$$

Clustering coefficient of a graph G is:

$$\frac{1}{n} \sum_{ii=1}^{n} \text{clustering coefficient of node } x_i$$

Small-world property: A graph G exhibits small-world behavior if the average path length μ_L scales logarithmically with the number of nodes in the graph, that is, if: $\mu_L \propto \log(n)$

Scale-free property: A graph G exhibits the scale-free property if the empirical degree distribution $f(k)$ has a power-law relationship with k, that is, if: $f(k) \propto k^{-\gamma}$

Clustering effect: Letting $C(k)$ be the average clustering coefficient of nodes with degree k, a graph exhibits a clustering effect if: $C(k) \propto k^{-\gamma}$

Erdös-Rényi Random Graph: Given an input number of nodes and edges, generates a random graph by placing edges uniformly at random between vertices, giving each edge equal probability of existing. Exhibits small-world property, but does not exhibit scale-free property or the clustering effect.

Barabási-Albert Scale-Free Graph: Given an input number of initial nodes and edges, as well as a parameter q for new edges to add at each iteration, generates a random graph by iteratively adding a new vertex and q edges in each iteration, where each new vertex connects an edge endpoint to each existing vertex with probability dependent on the degree of the existing vertex. Exhibits the ultra-small-world property, scale-free behavior, but no clustering effect.