Computational Statistics and Data Analysis 68 (2013) 170–189

Contents lists available at ScienceDirect

Computational Statistics and Data Analysis

journal homepage: www.elsevier.com/locate/csda

On the statistical detection of clusters in undirected networks

Marcus B. Perry [a,*], Gregory V. Michaelson [b], M. Alan Ballard [a]

[a] *Department of Information Systems, Statistics & Management Science, The University of Alabama, 300 Alston Hall, Box 870226, Tuscaloosa, AL 35487, United States*

[b] *Travelers Insurance, Claim Analytics, One tower square, Hartford, CT 06183, United States*

ARTICLE INFO

Article history:
Received 21 November 2012
Received in revised form 8 April 2013
Accepted 21 June 2013
Available online 4 July 2013

Keywords:
Undirected networks
Network clustering
Community structure detection
Log-likelihood
Log-likelihood ratio
Hypothesis testing

ABSTRACT

The goal of network clustering algorithms is to assign each node in a network to one of several mutually exclusive groups based upon the observed edge set. Of the network clustering algorithms widely available, most make the effort to maximize the modularity metric. Although modularity is an intuitive and effective means to cluster networks, it provides no direct basis for quantifying the statistical significance of the detected clusters. In this paper, we consider undirected networks and propose a new objective function to maximize over the space of possible group membership assignments. This new objective function lends naturally to the use of information criterion (e.g., Akaike or Bayesian) for determining the "best" number of groups, as well as to the development of a likelihood ratio test for determining if the clusters detected provide significant new information. The proposed method is demonstrated using two real-world networks. Additionally, using Monte Carlo simulation, we compare the performances of the proposed clustering framework relative to that achieved by maximizing the modularity objective when applied to LFR benchmark graphs.

1. Introduction

Clustering has a wide array of applications, from pattern recognition and spatial data analysis to data mining and military intelligence. Regardless of the application, clustering methodologies are often used to explore a data set where the goal is to partition the sample into distinct groups, or to provide new understanding about the underlying structure of the data.

Different approaches have been developed to address the problem of clustering. A popular approach, known as hierarchical clustering, seeks to identify *nested* clusters in a data set (see Gordon (1987)). Either agglomerative or divisive, hierarchical clustering algorithms either combine or separate observational units in order to produce the clusters. The output of such an algorithm is the dendrogram, where the user is left to determine the appropriate number of clusters for the particular data set.

Another approach is k-means clustering. Using this approach, the user specifies the number of groups *a priori* and then randomly assigns each observational unit to one of those groups. The centroid of each of the groups is calculated, and each observational unit is reassigned to the nearest cluster. The centroids of these new groups are then recalculated and the observational units are again reassigned to the closest group. The process continues until group membership stabilizes. A good review of k-means clustering is given by Steinley (2006).

Unlike the previous methods, spectral clustering does not require the user to specify the number of groups *a priori*. This approach requires the calculation of a matrix to describe the similarity between each pair of observational units, i.e., the similarity matrix. The eigenvectors and eigenvalues of this matrix (i.e., the spectrum) are then calculated and used to identify

[*] Corresponding author. Tel.: +1 205 348 9864.
E-mail addresses: mperry@cba.ua.edu (M.B. Perry), gmichael@travelers.com (G.V. Michaelson), talktoalan@hotmail.com (M.A. Ballard).

0167-9473/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.csda.2013.06.019

group membership, e.g., one might bipartition the sample based on the sign of the elements of the eigenvector associated with the largest eigenvalue. The interested reader is directed to von Luxburg (2007) for a straightforward review of the technique.

Although clustering algorithms are often applied to conventional data sets, they can also be applied to network data (e.g., social networks, biological networks, computer networks, etc.). In such a case, the goal is typically to assign each node in the network to one of several mutually exclusive groups based upon information contained in the edge set. In general, a network can be defined as a graph $\mathcal{G} = (V, E)$ with vertex set V and edge set E, where each edge $e_{ii'} \in E$ denotes a connection (or relationship) between node $v_i \in V$ and node $v_{i'} \in V$ (for $i = 1, \ldots, n, i' = 1, \ldots, n$, where $i \neq i'$). Networks can be characterized based on the types of edges that exist between nodes. An *undirected* network is a network in which $e_{ii'} = e_{i'i}$, i.e., a relationship from node i to node i' implies an equal relationship from node i' to node i. In contrast, a *directed* network does not have this restriction. The values taken by the elements of E further characterize a network. In a *binary* network, the edges may take only the binary values 0 and 1, indicating the absence or presence of a link, respectively. Conversely, *weighted* networks allow the edges to take continuous values, although these values are often restricted to be non-negative.

The most popular approach to network clustering is to maximize the modularity metric, originally proposed by Newman (2004). Network clustering via modularity is available in a number of network analysis software packages and thus is widely available to network analysts. To define modularity, consider a network of size n, and let $\omega_{ij} = 1$ if node i belongs to group j, and 0 otherwise ($i = 1, \ldots, n$ and $j = 1, \ldots, k$). Further, let $A_{ii'}$ denote the (ii')th element of the adjacency matrix $\mathbf{A}$, m denote the total number of edges in the network, and d_i the degree of node i. For a given $n \times k$ group membership matrix $\boldsymbol{\omega}$ and $n \times n$ adjacency matrix $\mathbf{A}$, the modularity is defined as

$$Q(\boldsymbol{\omega}|\mathbf{A}) = \frac{1}{2m} tr(\boldsymbol{\omega}^T \mathbf{B} \boldsymbol{\omega}) \tag{1}$$

where $tr(\mathbf{G})$ denotes the trace of the matrix $\mathbf{G}$ and

$$B_{ii'} = A_{ii'} - \frac{d_i d_{i'}}{2m} \tag{2}$$

denotes the elements of the so called *modularity matrix*.

Modularity is a useful, intuitive, and effective statistic for measuring the extent to which a given partition of a network is modular. Specifically, it measures the fraction of edges that fall within the given groups minus the expected such fraction if edges were distributed at random. Larger values of modularity suggest the presence of densely intra-connected and sparsely inter-connected nodes. The clustering problem involves finding $\boldsymbol{\omega}^*$, i.e., the group membership matrix in the set of all group membership matrices $\boldsymbol{\Omega}_k$ that yields the maximum modularity value, or

$$\boldsymbol{\omega}^* = \arg\max_{\boldsymbol{\omega} \in \boldsymbol{\Omega}_k} \left[Q(\boldsymbol{\omega}|\mathbf{A}) \right]. \tag{3}$$

In general, the optimization problem given above is not an easy one. In particular, for even moderately-sized networks, the number of possible ways to partition the vertex set is quite vast, rendering an exhaustive search infeasible. As such, heuristic search algorithms are often employed by which the number of possible network partitions evaluated is greatly reduced. In what follows we discuss some of these methods. Although maximizing modularity is by far the most popular objective, the methods discussed below are not exclusive to modularity. For a more comprehensive review of the large number of algorithms available for performing community detection in graphs, the interested reader is referred to Fortunato (2010).

The fastest method in common use was developed by Clauset et al. (2004). Their work is a modification of Newman (2004), where although they do not alter the general approach of Newman's algorithm, they do optimize its memory usage, data storage, and computational methods for use with sparse networks; i.e., a vast majority of networks of interest. Newman's algorithm is a greedy, agglomerative, hierarchical clustering algorithm that seeks to maximize modularity at each step. The algorithm begins by assuming that each node represents an individual module, then merges the modules that lead to the greatest increase in modularity.

Others have employed stochastic search methods such as simulated annealing or genetic algorithms (Guimerà et al., 2004; Kü cükpetek et al., 2005). These methods are generally found to be slower but more accurate than other deterministic methods. In fact, Danon et al. (2005) found that simulated annealing produced the most accurate results of any of the algorithms that were tested.

Another approach involves examining the spectral properties of various matrices. Newman (2006), for example, denotes the assignment of nodes into two groups in terms of an $n \times 1$ vector $\mathbf{s}$ in which node i's membership in subgroup 1 (2) is denoted by $s_i = 1\,(-1)$. By choosing the assignment of group membership in such a way as to maximize the inner product of $\mathbf{s}$ and the eigenvector associated with the largest eigenvalue of a function of the adjacency matrix, an approximately optimal partition can be determined. Each of these two subgroups can then be divided using a similar procedure.

Still another approach uses extremal optimization (Duch and Arenas, 2005), which focuses on correcting those nodes with the worst fit. Kernigan and Lin (1970) proposed a similar but more simplistic approach in which the graph is divided into equal parts. Also, there exists a class of search procedures that work by cutting the links between particular nodes or by otherwise physically dividing the global network into smaller pieces, Girvan and Newman (2002) and Newman and Girvan