Clustering Directed Graphs

Summer 2021 Research Project Report - Jeremy Nohel

A graph is a set of vertices together with some connections between them known as edges. A directed graph, or digraph, has the property that the edges are given direction, i.e., if one were to move along edges of the graphs between the vertices, it is only possible to move in the direction specified (a decent analogy is that of one-way highways connecting different cities). Clustering involves statistical techniques to discern larger patterns in the graph. To use the highway analogy, the goal of clustering aims to group the cities into clusters and discern larger patterns of movement between them. Clustering techniques in graphs have wide-ranging applications, from analysis of the internet to financial transactions to migration patterns. All of these have the common thread of using a graph to represent the transfer of *something* between different websites, people, or places, respectively. Algorithms for clustering graphs have come to the forefront of current research, particularly given the rise of big data and artificial intelligence, for which they also have many applications. These were the main focus of my research.

Having been introduced to some basic clustering techniques through our Prelims Statistics course and the requisite linear algebra involved in the Linear Algebra course, I began by reading one paper of many co-authored by Dr. Mihai Cucuringu, my supervisor, on the subject. I had just learnt about spectral clustering techniques for undirected graphs in a summer course, based on a matrix representation of the graph, and this gave me the intuition to understand this paper. Undirected graphs give rise to a symmetric matrix representation, which gives it many nice properties which spectral clustering exploits. The directed graph problem is much trickier due to its inherent asymmetry (an edge from a to b does not guarantee an edge from b to a), and Dr. Cucuringu's paper introduced a complex number representation (Hermitian clustering) that preserved the same properties of the symmetric matrix, allowing for a very similar clustering technique. He also introduced the Directed Stochastic Block Model (DSBM), a way of generating synthetic data sets with pre-known patterns which can be used to test how well the algorithm performs.

The first interpretation that Dr. Cucuringu suggested I investigate was that of electrical resistance, considering each edge to be a resistor in an electrical circuit. A very influential paper had been written in 1993 on the use of resistance in undirected graphs to determine a distance function, citing the fact that electrical resistance considers every possible path between points, while classical distance only considers the shortest path between points. This has great relevance to the clustering problem, as if two vertices have more connections between them it would make sense to put them into different clusters, showing a larger scale flow between those distinct groups. Since resistance as a physical property does not depend on direction (there is no such thing as a "directed wire" to my knowledge), the challenge was to generalise this to the context of directed graphs. A paper I found from 2013 describes in great detail a development of a notion of resistance in directed graphs, which corresponds to that of undirected graphs for suitable directed graphs, yet it fundamentally differed in interpretation from an electrical network - a key point in its application to the clustering problem. I then developed my own idea of resistance in directed graphs.

This idea of resistance was a probabilistic one - I reasoned that as long as two nodes were connected, but the directions of the connections were unknown, one would have a 50% chance of measuring the current in one direction and a 50% chance of measuring the current in the other, depending on the orientation of a battery between the nodes. My definition was to be a superposition of these two, each individually treated as undirected graphs. I then wrote some MATLAB code to calculate the resistances that way. Using the resistances as a measure of "similarity", I implemented a hierarchical clustering method to assemble the points into clusters. Testing it against other clustering methods on very small graphs generated by the DSBM, it did as well as the Hermitian clustering method and better than some other methods I had come across. Unfortunately, my idea had a number of drawbacks -

for one, it was limited to graphs without double edges (double edges allow for both directions between vertices without considering net flow), which was one of the things it was supposed to work better for. Another was that it was extremely computationally expensive and only testable on small graphs. I also proved that my method required computing resistances between every pair of points whenever the graph contained a directed cycle (where, for instance, in a set of vertices {1,2,3,4} the set of edges contains the set {(1,2), (2,3), (3,4), (4,1)}, where (a,b) denotes an edge from a to b). The other methods for clustering (including the Hermitian method) could be executed in $\sim n^2$ operations while mine was on the order of n^4, where n is the number of vertices in the graph. I did take comfort, however, in the fact that I had come up with a clever way to reduce it to polynomial time, as my original program was exponential in n.

I followed this up by comparing it all to the notion of directed resistance I had come across in other literature, and finding that, though much quicker, the algorithm tended to output extremely unbalanced clusters, while the synthetic data was based on clusters of the same size. I reasoned that this was due to the departure from interpretation based on electrical circuits, and that somehow that definition of resistance failed to distinguish the nodes of the graph enough, based on considerations of direction.

Having no more ideas in that interpretation, I then investigated defining distance between nodes a and b as a weighted average of the path lengths between a and b. This amounted to computing all paths between a and b, the probabilities of ending up on such a path, and the lengths of each path separately - a supremely difficult computational problem. And again, cycles were a major difficulty, since there is always a probability of continuing around the cycle ad infinitum.

It eventually became clear to me that investigating clustering techniques with no particular goal of what the clusters should look like was turning into a dead end. In every paper I read, the particular clustering algorithm described involved optimising a “goal function”, based on the particular kind of data at hand. I tried to investigate different kinds of goal functions to see if there was a common thread, ultimately to no avail. It exposed to me the very specific nature of applied mathematics and the importance of these optimisation problems in every field.

The skills I began to develop in considering the many interpretations and getting to the heart of a given problem will surely have a dramatic impact in the future. Though I was unable to establish any comprehensive result, the investigation was thought-provoking and thoroughly enjoyable. I would like to thank Dr. Cucuringu and Merton College for their support throughout.