Clustering Project Report

Alexander Russo

UMRF Ventures Inc. IT Command Center

Abstract

The UMRF Ventures Inc. IT Command Center has recently been tasked with performing root cause analysis on incidents and problems reported in FedEx’s Process Driven Service Management (PDSM) software. To accomplish this objective, we have begun working on a clustering project. The main objective of the clustering project is to determining clusters in problems/incidents to automate the process of determining the root cause. Automating the process of root cause analysis involves determining which clustering algorithms to use, matching those algorithms with human-determined categories to determine classification, and then integrating said algorithms with PDSM. The purpose of this document is to report on the research that I have been doing on clustering algorithms and to recommend to the UMRF Ventures Inc. IT Command Center Technical Lead which clustering algorithms should be used.

Report

Types of Clustering

There are multiple types of data clustering, but the most common types of clustering are hard clustering, soft clustering, hierarchal clustering, flat clustering, and model-based clustering. In hard clustering, each data point either belongs to a cluster completely or not. More specifically, a given point in n-dimensional space only belongs to one cluster. This is also known as exclusive clustering. In soft clustering, instead of putting each data point into a separate cluster, a probability or likelihood of that data point to be in those clusters is assigned. Basically, a given data point can belong to more than one cluster in soft clustering. This is also known as overlapping clustering. In hierarchal clustering, a hierarchy of clusters is built using the top-down (divisive) or bottom-up (agglomerative) approach. So, basically, objects that belong to a child cluster also belong to a parent cluster. A flat clustering is a simple clustering with no hierarchy present. In model-based clustering, data is modeled using a standard statistical model to work with different distributions. The idea is to find a model that best fits the data. Considering that there are overlapping or similar values present in the data that we have been given, I believe that either soft or hierarchal clustering should be used on the data.

Clustering Algorithms

There are four main types of clustering algorithms: connectivity models, centroid models, distribution models, and density models. In connectivity models, clustering is done in one of two ways. In the first approach, start by classifying all data points into separate clusters and then aggregating them as the distance decreases. In the second approach, all data points are classified as a single cluster and then partitioned as the distance increases. Also, the choice of the distance function is subjective. So, basically, the iterative process of this algorithm is to continually incorporate a data point or group of data points with other data points and/or groups until all points are engulfed into one big cluster. The main drawback of this model is that it lacks scalability for large data sets; the critical input for this type of algorithm is knowing where to stop the grouping from getting bigger. A great physical example of this type of model is a family tree.

Centroid clustering is one of the most common clustering models used in cluster analysis. The user chooses the number and type of clusters that they want to classify. The algorithm will then start by randomly selecting centroids (aka cluster centers) to group the data into pre-defined clusters. A line is then drawn separating the data points into the clusters based on the proximity to the centroids. The algorithm will then reposition the centroid relative to all the points within each cluster. The centroids and points in a cluster will adjust through all iterations, resulting in optimized clusters. The result of this analysis is the segmentation of your data into the pre-defined clusters. The downside of this clustering model is that you have to define your clusters ahead of time, and the number and types of clusters for our data may not necessarily be static.

Distribution models are based on the notion of how probable it is that all data points in the cluster belong to the same distribution. Basically, distribution clustering identifies the probability that a point belongs to a cluster. Around each possible centroid the algorithm defines the density distributions for each cluster, quantifying the probability of belonging based on those distributions. The algorithm then optimizes the characteristics of the distribution to best represent the data. One drawback of these models is that they can suffer from overfitting; the algorithm skews the data to fit the distributions. A good physical example fo this would be an archery range.

Density clustering models group data points by how densely populated they are. More specifically, the model searches the data space for areas of varied density of data points in the space. It then isolates various different density regions and assigns the data points within these regions in the same cluster. This process is iterated until the best clusters can be identified. The downside of this is that most density algorithms require a predetermined distance between data points be selected to benchmark how closely the points need to be to one another in order to be considered related. This distance may not be static among our data sets. Some popular examples of these are DBSCAN and OPTICS.

Synopsis

Based on what I have read, I have determined that the best clustering algorithm for our root cause analysis project would be either a connectivity algorithm or a density algorithm. Hierarchal clustering is the most common type of connectivity algorithm. As previously stated, there are two major types of hierarchal clustering models, agglomerate and divisive. In agglomerate clustering, each data point starts out as a single cluster and then merge pairs of clusters until all clusters have been merged into a single cluster that contains all data points. In divisive clustering, the data is first represented in the same cluster, then the largest cluster is split into separate clusters. This process continues until each object is represented by its own cluster. Given that the goal of this project is root cause analysis, if we decide to use connectivity clustering, we should use an agglomerate clustering algorithm.

If we instead use density-based clustering, we will need some pre-existing knowledge of the data domain in order to set parameters. There are three common methods for density-based clustering: density-based spatial clustering of applications with noise (DBSCAN), ordering points to identify the clustering structure (OPTICS), and mean-shift.

Given a set of points in some space, the DBSCAN algorithm groups together points that are closely packed together, marking points that lie alone in low-density regions as outlier points. An advantage of DBSCAN is that it does not require you to specify the number of clusters in the data a priori. However, some of the drawbacks are that DBSCAN cannot cluster sets well with large differences in density and if the data and scale are not well understood, choosing a meaningful distance threshold can be difficult.

OPTICS is similar to DBSCAN, with one main difference; OPTICS can detect meaningful clusters of data of varying density, which is something that DBSCAN cannot do. To accomplish this, the points of a database are linearly ordered such that the spatially closest points become neighbors in the ordering. An example of this would be a dendrogram.

The mean-shift clustering algorithm is a nonparametric clustering technique which does not require prior knowledge of the number of clusters and does not constrain the shape of the clusters. A good example of a mean-shift algorithm would be to consider a set of points in two-dimensional space. Assume a circular window centered at C and having radius r as the kernel. Mean shift is a hill climbing algorithm which involves shifting this kernel iteratively to a higher density region until convergence. Every shift is defined by a mean shift vector. The mean shift vector always points toward the direction of the maximum increase in the density. At every iteration the kernel is shifted to the centroid or the mean of the points within it. The method of calculating this mean depends on the choice of the kernel. In this case if a Gaussian kernel is chosen instead of a flat kernel, then every point will first be assigned a weight which will decay exponentially as the distance from the kernel's center increases. At convergence, there will be no direction at which a shift can accommodate more points inside the kernel. One of the drawbacks of using mean-shift is that the selection of the window size is not trivial and often requires the use of an adaptive window size.

If we decide on using a density-algorithm, I strongly recommend that we should use either OPTICS or mean-shift. There are several online tutorials for building OPTICS algorithms using Python and R, while mean-shift does not require us to know the number of clusters before we build it. If we decide to use a connectivity (hierarchal) algorithm, I would strongly recommend using an agglomerate model, as our goal for the project is to identify root cause analysis, and agglomerate models would allow us to build clusters based on patterns within the data, forming clusters and, eventually, identifying root causes for problems/incidents.

References

Gervelis, G. (2018, July 8). 4 Types of Cluster Analysis Techniques used in Data Science. [www.decisivedata.net](http://www.decisivedata.net), retrieved: February 28, 2019: from <https://www.decisivedata.net/blog/4-types-of-cluster-analysis-techniques-used-in-data-science>

Kaushik, S. (2016, November 3). An Introduction to Clustering and different methods of clustering. [www.analyticsvidhya.com](http://www.analyticsvidhya.com), retrieved: February 28, 2019: from <https://www.analyticsvidhya.com/blog/2016/11/an-introduction-to-clustering-and-different-methods-of-clustering/>

Seif, G. (2018, February 5). The Five Clustering Algorithms Data Scientists Need to Know. towardsdatascience.com, retrieved: February 28, 2019: from <https://towardsdatascience.com/the-5-clustering-algorithms-data-scientists-need-to-know-a36d136ef68>

Theodoridis, Sergios; Koutroumbas, Konstantinos. (2009). *Pattern Recognition (4th Edition).* AcademicPress: <https://www.sciencedirect.com/science/article/pii/B9781597492720500165?via%3Dihub>

Wikipedia contributors. (2019, March 4). Cluster analysis. In *Wikipedia, The Free Encyclopedia*. Retrieved 19:13, March 5, 2019, from <https://en.wikipedia.org/w/index.php?title=Cluster_analysis&oldid=886222784>