Assigning Confidence to Clustering Algorithms

Aamal Hussain

March 18, 2020

- Clustering algorithms in general, with a focus on route clustering
- Adding confidence bounds to clustering algorithms

0.1 Clustering of time series data: A survey

In this case, the data features changes over time. These are divided into five categories

1. Partitioning: Constructs $k \leq n$ partitions of the n data tuples. Each partition represents a cluster containing at least one object. This allows for objects to be in clusters with different degrees Allows for the use of fuzzy methods to calculate routes off fuzzy sets.
    - Fuzzy c-means
    - Fuzzy c-medioids
2. Hierarchical: Groups data into trees of clusters.
3. Density Based: Grow a cluster as long as the density in the neighbourhood exceeds some threshold
4. Grid based: Quantise the space into cells and perform clustering on these cells
5. Model Based: Assumes a model for the data Would allow for Bayesian clustering, which would allow for the 'no class' property to be considered.

Clustering for time series data the following methods are mentioned

1. Relocation clustering: Uses a particular criterion function and works by comparing the resulting function with members attached in different clusters. This would also allow for the 'no class' option to be considered. This method works only when the time series data are of equal length.
2. Agglomerative Hierarchical: Places each object into its own cluster and then merges these into larger clusters by trying to minimise the sum of squares variance. This method falters once clusters are chosen since it is unable to adjust
3. Fuzzy k-means considers using a fuzzy membership scheme for partitions. By warping it is possible to make these more appropriate for time series data of unequal length, but then the distance metric needs to account for this.

The major problem here concerns the metric used to judge the distance between two paths. This is particularly a problem in route clustering as we have to take into account issues of varying traffic conditions etc.

The modalities for distance/similarity measures which are proposed are

- Euclidean distance (Obviously if time is not taken into account this is shite)
- Cross correlation coefficients (such as Pearson's rank etc.). Can also be a function of time (Can also use these as the metric which considers the confidence of attaching a route to a cluster.)
- Short time series distance (STS): In which each time series is considered as a piecewise linear function
- Dynamic time warping distance. First align the time series and then use Euclidean distance
- KL divergence (I quite like this one if the data sets are somewhat aligned first)
- J divergence/Chernoff divergence

The paper then suggests methods for clustering time series data. I focus on the ones which use the raw time series data since that's what we have. (The others are feature based approaches and model based approaches).

0.2 Adding Confidence to Gene Expression Clustering

In this study, the authors consider how to cluster gene expression data which contains large amounts of random variation. I anticipate that this would be the same case in the GPS data which will be subject to noise. Importantly, they are able to assign a statistical significance metric when clustering to improve the reliability of the cluster. Previous work relies on doing this by counting how many times a gene is assigned to the same cluster (seems long). Other recent work which has used GWN to perturb data assigns a perturbed cluster to the original iff it contains a majority of elements in common with the original cluster. (If there is a way to extract features from the GPS data, we could consider this as a method of evaluation, e.g. if routes passed under the same 10 bridges).

Their dissimilarity metric, known as the 'Munneke metric' is able to take into account dissimilarity based on the magnitude and the pattern of gene expression (Perhaps such a combination of metrics would also be useful in route clustering). Importantly, however, this metric provides a strong confidence measure of the clustering algorithm as it provides a non-negative real value to measure the dissimilarity between the data and a particular cluster.

0.3 Correlation based dynamic time warping of multivariate time series

In this paper they also develop a similarity metric called correlation based dynamic time warping. This is particularly handy for multivariate time series data which presents some correlation between the variables. Then there is the standard 'our algorithm outperforms others'.

0.4 Hierarchical Clustering Given Confidence Intervals of Metric Distances

They consider the case where the exact dissimilarity between two sets of data lie within some interval. The types of clustering algorithms which are currently available are

- Exclusive clustering: Only belongs in one cluster (e.g. k-means)
- Overlapping clustering: Belongs to multiple clusters with degree of membership (e.g. fuzzy means)
- Hierarchical clustering: Considers a family of partitions which are indexed by a connectivity parameter
- Probabilistic clustering which takes a probabilistic approach (Mixture of Gaussians)

Axiomatic hierarchical clustering is a sub-group in which algorithms must satisfy reasonable axioms. E.g.

- Value: Nodes form a single cluster determine by their distance
- Transformation: A space which is uniformly dominated by another space should have clusters which are uniformly dominated

The advantage of this method is that it takes into account the uncertainty in the metric itself and the interval effectively becomes the confidence of the clustering algorithm.

They apply this method on moving point data (which shows its applicability to the problem in question) and show that it is able to successfully determine the underlying clusters, although (as would be expected), the performance deteriorates with increasing distance intervals.

0.5 Scoring clustering solutions by their biological relevance

This study differs from the others in that it considers evaluating the performance of a clustering result after the algorithm is run. This essentially gives a method of assigning the confidence in the clustering algorithm once all the clusters are generated. The confidence is assigned based on prior knowledge and so some model is required for this to work. This model is given in the form of an attribute matrix whose rows correspond to data. The clusters $C_{1:n}$ are assigned to the data so that $a_{ij} = (a_{ij}^1 ... a_{ij}^p$ is the attribute vector of data point j in cluster i. Then the clustering of the data points is evaluated against A. It should be pointed out that this requires some prior understanding of the attributes of the data point and how they correspond to the cluster. In other words, a model is required for this to work. In order to carry out the evaluation, an Analysis of Variance method is used This metric is given by

$$F_H = \frac{SSH/(l-1)}{SSE/(n-l)} \tag{1}$$

where

$$SSH = \sum_{i=1}^{l} s_i(\bar{a_i} - \bar{a})^2 \tag{2a}$$

$$SSL = \sum_{i=1}^{l} \sum_{j=1}^{s_i} (a_{ij} - \bar{a_i})^2 \tag{2b}$$

Here, l is the number of population distibutions, $s_i = |C_i|$, $\bar{a_i}$ is the mean of the data points in cluster i and $\bar{a}$ is the total mean of all of the n elements.

The evaluation is done as follows

1. Compute a linear combination of the attributes: Each data point is assigned a real-valued number by computing a weighted sum of each of its attributes. There is often not an a priori method for calculating these weights and so these will need to be determined at user preference.
2. Determine $-logp$ where p is the probability that all values in the projection have been taken from the same population
3. Determine the sensitivity of the above metric to small modifications of the clustering solution. For a given solution, generate a group of perturbed clustering solutions by exchangin random pairs of data points across clusters. Determine the standard deviation of the above CQS metric. This will act as the confidence value.

0.6 clValid, an R package for cluster validation

This paper provides an introduction to a module which can be used for assessing the validity of clustering solutions. The method of validation used is user defined but falls into one of three categories:

- internal validation: takes only the dataset and clustering solution as input.
- Stability validation: considers the consistency of the clustering solution after small perturbations
- Biological validation: considers the ability to produce biologically relevant results. This would need to be replaced with considering information relevant to route clustering. Therefore, I do not focus on this one in the review

0.6.1 Internal Measures

The available measuers are:

- Connectivity: considers the extent to which observations are placed in the same cluster as their nearest neighbours.
- Silhouette Width: Considers the normalised average distance between the observation and others in the same cluster, as well as the distance between the observation and the nearest cluster. This metric results in a value between -1 and 1, where 1 denotes a perfect cluster.
- Dunn Index: Considers the ratio of the smallest distance between observations which are not in the same cluster against the largest distance across clusters.

0.6.2 Stability measures

- Average proportion of non-overlap (APN): First clusters based on the full data and then with a single column removed. The APN value considers the ration of numbers of data points which are still placed in the same cluster.
- Average distance: Measures the distance between observations placed in the same cluster when a single column is removed.
- Average Distance between Means: Distance between cluster centres when a column is removed

Handl et al. give a good review of internal measures

Chapter 1

Formal Write Up

1.1 Clustering of time series data

To be specific, the problem that the data science team face regards clustering routes which is considered to be a form of time series data. As such, it is important that the method of clustering used is suited to this sort of problem. Methods which involve clustering time series data can largely be categorised into the following categories (as defined by **something et al.**) (GIVE EXAMPLES OF THESE TO SQUEEZE IN REFERENCES):

1. Partitioning: Divides the n data tuples into $k < n$ partitions, each of which represent a cluster containing at least one tuple.
2. Hierarchical: Groups data into trees of clusters and iteratively decides whether a new branch is required for each data tuple.
3. Density Based: Clusters will continue to grow as long as the density of data tuples in the neighbourhood (in metric space) exceeds some threshold.
4. Grid Based: Quantise the space into discrete cells and perform clustering on these cells independently.
5. Model Based: Assumes a model for the data and assigns data tuples to these clusters based on the resulting agreement with the model.

For the sake of brevity, I focus on the first two of these methods as they appear to be most relevant to the problem at hand.

A common question which is expected to arise when new data arrives is whether the given tuple should be clustered within an existing cluster or whether it should be considered to be unique. To this end, Agglomerative Hierarchical methods (GIVE A REFERENCE) show particular promise. In this, each object is first placed into its own cluster. These clusters are then iteratively merged such that the sum of squares variance is minimised. Once this minimum is achieved the resulting clusters are chosen as the final solution. Note that it is possible to design hierarchical algorithms in such a manner that a maximum number of clusters can never be exceeded, i.e. the program terminates once a certain number of clusters is reached. This method gives each data tuple the opportunity to be classed as its own cluster and, in fact, if a route is completely unique, it should not be merged with another cluster. Whilst at first glance this seems to solve the aforementioned problem, it should be noted that when clusters are merged through the Agglomerative Hierarchical technique, they remain immutable and cannot be broken. Therefore, if a new data tuple arrives which requires that existing data tuples are merged into its cluster, then it may be necessary to run the algorithm from scratch. However, it is unlikely that this will be required often.

A similar problem is that, often, routes will partially overlap with one another and therefore, they could be placed in clusters with different degrees rather than placed entirely in one. As mentioned in **Section 1.4**, Fuzzy clustering allows for this to be accounted for. However, with time series data, it is of particular importance that the metric be chosen to accurately compare routes with potential clusters. This is especially

true when time series data is of inhomogeneous length. This will likely be the case with route clustering since the speed of all vehicles will not be the same at all times, due to varying traffic conditions. **This should probably be placed with Henrik's bit**

In light of the issues surrounding the choice of distance measures, I provide a short review of potential metrics that can be used for time series data below. These are considered in further depth in **REF**.

1. Euclidean distance
2. Cross correlation coefficients
3. Short time series distance
4. Dynamic time warping distance
5. KL divergence
6. J/Chernoff divergence

I'll go into more depth on these in the final write up

1.2 Evaluation of clustering solutions

Alongside the clustering solution itself, the confidence in the solution must be considered. This gives the data science team an indication of the extent to which the solution can be trusted. In the literature, this problem has, for the most part, been explored in the context of gene expression clustering. Here, it is important for biological studies, that the clustering solutions provide some useful biological relavance. However, the ideas can be transferred towards route clustering which too contains large amounts of uncertainty and must hold geographical relevance.