TECHNIQUE FOR ROBUST DATA-DRIVEN CLUSTERING

Developed by Clusters

Clusters

CHALLENGE SOLUTION RESULT

INTRODUCTION

This paper describes an algorithm designed to reliably cluster real data of high dimension into self-similar groups. The paper was written by the authors of the algorithm and covers at a high level the processes and reasoning that gave rise to the approach taken. The intention is to describe some of the common problems encountered in this style of analysis and the characteristics of the solutions invented or adopted to cope with them. The algorithm is embodied in a software program written in Java. Specifics of the implementation of the algorithm are not covered in this paper.

The authors have developed the algorithm and associated software for exclusive use by Clusters, the segmentation specialists.

HOW DOES IT WORK?

In the current implementation of the algorithm there are four steps:

1. Reading the data
2. Pre-processing the data
3. Clustering the data and
4. Deciding what is the best result

The steps are performed in sequence and the results are produced during step 3. The majority of the time spent in the algorithm is looking for things that are “close” to one another. The clever part is how we define “close”.

1 READING THE DATA

Data is presented to the algorithm in a tabular textual form (comma separated values (.csv) such as is commonly created by Microsoft Excel®. Metadata is added to the second row of the table of data, describing the variable represented by the column and providing a context for its usage by the algorithm. Further discussion of the detail of this portion of the process is left for a detailed description of the implementation.

2 PRE-PROCESSING

WHY PRE-PROCESS?

Data pre-processing ensures that data is treated in an unbiased fashion and that the results are not skewed by arbitrary factors such as the scale or range of the values presented in the data.

Our unique pre-processing techniques involve standardising all data, the accurate treatment of textual data and the smoothing of the subsequent analytical space. We believe ours to be the only technique that combines all of these steps prior to clustering and gives us the most robust and meaningful results.

We have developed our pre-processing techniques based on observation of a large amount of real world data over years of practicing numerical analysis. We deal with the issues that can cause other clustering and segmentation techniques to provide sub-optimal results. Our experience leads us to believe that the characteristics we have observed which cause this are prevalent in almost all data collected for the purposes of segmentation.

DATA STANDARDISATION

Data standardisation is the process of adjusting all data so that it falls within a prescribed range and has common basic statistical characteristics. There are a variety of techniques available for use in the algorithm but we feel that the spherical geometry of the solution we provide means that normalisation of the data so that the average value is 0, the standard deviation is 1, provides the best method.

The process for standard numerical variables is to inspect the data in each column and compute its basic statistics (mean, standard deviation, sum, sum of squares, maximum, minimum etc.).

Every row in that column is then recalculated so that when they are taken as a whole they fall in the $(-,\infty + \infty)$ interval, their mean is 0 and their standard deviation is 1.

The result of standardisation is a data space which has had removed the bias which appears as a result of scale and consequently all input variables are treated equally. It also realigns the data into a common scale for analysis and gives it a degree of “spherical” symmetry. The spherical symmetry is important because of how we go on treat regions of high information density.

SPHERICAL DATA SMOOTHING

In the case of normally distributed data (bell curve distributions), even after standardisation, lots of the data are clumped together very closely about the mean (for non-normally distributed data there is generally a region of high data density, it is just not necessarily at the mean).

If clustering is performed with the data in this condition, differences in the data that may contain very useful information are invisible. Commonly it results in clusters that are very obvious and therefore contain a low degree of actionable value. This can greatly reduce the usefulness of the resulting solution.

Our smoothing technique treats the data in a way that causes important differences that are normally hidden to be detectable and unimportant similarities, which are obvious but uninformative, to be treated with less significance.

The result of the technique is a space that has been compressed where the information density is low, and exploded where it is high. This allows the subsequent clustering to distinguish subtle, but meaningful patterns in data that are normally hidden either by the data all being on top of each other, or scattered very far apart. We are careful to do this in a way that preserves the symmetry of the data, so we do not introduce any artificial bias, even in high dimension and when textual data is present.

We believe this to be a key problem when performing clustering analysis on real world data that, if not addressed, can significantly reduce the quality of the results.

HAUSDORFF DIMENSION ESTIMATION

When applying a technique like spherical smoothing, it is very important that consideration is given to the amount of explosion that occurs. It is quite possible to over-explode the data, pushing it all to the edge of the analysis space. This can cause many of the problems we are trying to avoid. For this reason it is necessary to inspect the data carefully and assess the real dimensionality of the space.

In the real world the dimensionality of any set of data is the number of variables you would need to identify a coordinate in the space. For a plane you need an x and a y coordinate, so it is considered 2-D, x, y and z for a 3-D volume and so on. It is tempting to treat our observed data in the same way and have an n-D space where n is the number of columns in the data.

However, we deduce that arbitrary spherical explosion of the data using this measure of the dimensionality has its own attendant problems. If columns in the data are correlated the explosion is too extreme. For this reason we developed a technique for estimating the “real” dimensionality of the data.

We called this Hausdorff Estimation after the German mathematician who specialised in topology and formulated techniques for estimating the dimensionality of fractals. We compute our estimate in an empirical way by inspection of the regions of the data space that are occupied by observations, but it bears close resemblance to the problem Hausdorff was addressing and that Minkowski-Bouligand addressed in fractal geometry.

The result is an estimate of the dimensionality that tends towards 1 for highly correlated data and n for uncorrelated data, where n is the number of columns or dimensions in the data space. The estimate gives just the right amount of explosion of the data for the density of the information contained within it.

CATEGORICAL DATA

Most clustering (indeed many mathematical) techniques deal with categorical data very badly. Either they insist on giving the textual categories a number (a numeric ordinal value) and thereafter treat it as a continuous numeric variable, or more commonly they cannot deal with the categories at all. The problem is that, by their definition, categories are discrete, discontinuous and orthogonal.

Treatment of categorical variables as continuous numbers gives rise to results that fall between the ordinal values and ambiguous or erroneous interpretation. If you make Yellow = 2 and Blue = 3 what does 2.478 mean? It is tempting to say Yellowy-Blue, but if the categories are Male and Female, or Yes and No, this method of answering the question is useless for all real-world applications of the results.

The practical reality is that many applications give rise to distinct categories that are neither ordinal nor continuous and should not be treated as such. Our technique considers them distinct throughout the analysis and does not blur them together into statistically accurate but meaningless hybrids.

We are also able to standardise them in a manner similar to how we treat other numerical variables so they do not skew the clustering analysis. We believe this is of significant benefit in clustering applications and that we are unique in our treatment of categorical data.

3 CLUSTERING

NEAREST NEIGHBOUR

Having pre-processed the data in the manner described we then cluster using a fairly standard nearest-neighbour technique. The name says it all. We start by measuring the Euclidean (straight line) distance between all pairs of points and then stick the closest ones together to form potential cluster centres.

The cluster centre (centroid) is adjusted to be at the Euclidean mean of the vectors that form the cluster. This is a complicated way of saying “in the middle of the dots”.

As successive points are added the centre shifts slightly. Imagine it starts by adding two points together that are connected by a line. The cluster centre is at the mid-point of that line. When you add a third point the data forms a triangle and the cluster centre moves to be at the centre of the triangle, and so it goes on as points get added.

Points belonging to clusters are taken out of play and represented by their cluster centre. The distance between the new cluster centre and all other data points, and clusters, is then computed and the next nearest neighbour found and the process repeats.

Once the clusters begin to form a moment is reached where the closest things in the space are clusters, so they are combined to form new clusters, the new cluster adopting all the data points of the old ones.

THROWING BACK

Observation of iterative clustering techniques such as nearest neighbour shows they have a propensity to make elongated, wrapped, or odd dumb-bell shaped clusters. Often data points reside in a cluster for historical reasons achieved by successive merging, rather than because they are legitimately closest to the cluster centre.

To avoid this we introduced a sub-process which analyses the merged cluster and returns a proportion of the data points back into play based on their illegitimacy in the cluster. This is done in such a way that we can reliably state that the algorithm converges, and that we will find robust and meaningful clusters where all members are assigned to their best possible cluster.

4 WHEN TO STOP - THE MEASURE OF GOODNESS OF SOLUTION

Eventually, left to its own devices the algorithm lumps all data into one enormous cluster. This is the null result and simply identifies the middle of the data – which we could have easily derived in one step by inspection of the incoming variables. However, it is an important statement about the convergence of the algorithm. The fact that the logical progression leads to the null result is a desirable and important proof of the stability of the algorithm.

Of course the null result may be heart-warming for the algorithm developer, but is useless for all practical purposes. The useful solutions all lie in the domain where there are several clusters at play and data is distributed meaningfully between them. During the clustering process there are many such solutions and the trick is to pick the best one.

We have developed a technique of measuring the “goodness” of the clustered solution. This is more than a subjective heuristic of what we consider to be a good solution, but inspects the clusters and their membership from a statistical standpoint. The resulting solutions identified by this measure have the dual characteristics of being statistically reliable and making sense.

The best solutions do not always arise when all data points are allocated to clusters. Often the best clustering identifies solid reliable centroids for the majority of data and some data that are ambiguously placed between them. In segmentation studies it is obviously important to be able to identify every data point as belonging to one or other cluster, but this practical necessity should not dictate the underlying “best” solution. The measure of goodness takes this into account.

The measure is computed at each step of the process as the clustering proceeds and allows us to identify the optimum solution and gives us the opportunity to compare different solutions to one another. The tolerance of the measure to ambiguously positioned data means that our results are not skewed by outliers or noise. Using this characteristic we go on to identify the optimum solution having allocated all the data to clusters. As a result our solutions are shown to take into account all data.

DUST

Ambiguously positioned data can arise for several reasons. Most commonly, if the data is genuinely continuous and normally distributed and comes from any real-world system, it will contain abnormalities and noise.

Our metaphor for this is the universe itself. Most matter is clumped into planets, stars, galaxies and the like, but there is a significant proportion of inter-stellar dust that floats around in between the clusters of matter. We liken the ambiguous noisy data points to inter-stellar dust.

It is important that this data is considered but not allowed to skew the results of the analysis. Treatment of dust in an even-handed way based on its significance to the results is an important characteristic of our approach and ensures that we extract all the information we can from the data presented.

In order to produce results which have genuine usability, rather than theoretical value, we ensure that all dust is allocated to its “nearest” cluster as part of the process. This means the every observation provided to the analysis will be found its best home in the solution. When computing near-ness during this part of the process we take into account all the standardisation and smoothing techniques discussed above.

Clusters

www.clusters.uk.com

Address: 30 Park Street, London, SE 19EQ, UK

Email: info@clusters.uk.com **Tel:** +44 (0)20 7842 6830

Follow us: @ClustersLtd ClustersLtd Clusters Limited