## Clustering

### Sample Data

Clustering is a technique for classifying data. It is most easily considered using the traditional Cartesian graph. To quickly demonstrate my point I would like to propose a hypothetical scenario. A factory produces an unknown number of different circular objects. Measurements are available for two aspects of these objects. The total diameter of the object, and the diameter of this central hole. Below is a table of the contrived data;

|  |  |  |
| --- | --- | --- |
| Total diameter | Central hole diameter | Number of objects |
| 5 inches | 1 inches | 12 |
| 10 inches | 0 inches | 10 |
| 3 inches | 0 inches | 20 |

Table 1; a table of hypothetical values, solely for demonstration purposes.

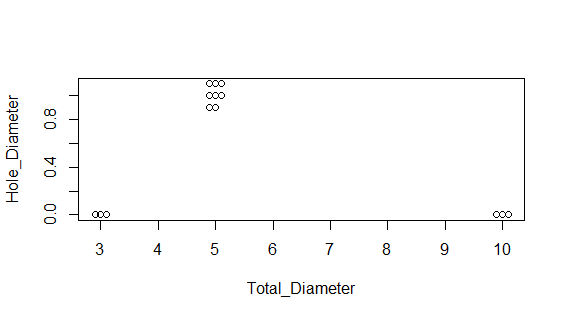


Figure 1; For the graph a margin of error of 0.1 was introduced in the data. This was primarily to avoid all the points being stacked on each other, while still keeping the clustering quite tight.

When viewing the data of this two variables on a Cartesian plot we see groups of datum. Those groups are clusters of the data. From this I spectate that there is 3 distinct clusters. Each of these is actually a food. The foods are donuts, pizza, and scones, respective to order in the table. An argument could also be made for the classification of the datum into two distinct clusters, those who have and do not have a hole. However I would argue that the resulting cluster of pizza and scones is a poor cluster, as it entire overlaps the donut cluster in one axis.

Using these measurements, it is now possible to classify new data and sort them according to the attributes they possess. So if an item, which was made incorrectly, and had a total diameter of 2 inches was found, it could be either a scone or a donut, by examining the diameter of the central it should become clear into which category to place it. Another example would be the variety in pizza sizes. The size would likely differ enough from the other objects in order for the pizza of a new size to be grouped with the current pizza cluster. With enough pizza differing in size, new cluster may be formed, allowing the sub-classification of pizzas into small, medium, and large for instance.

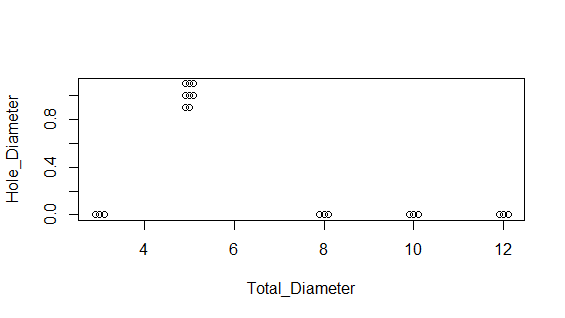


Figure 2; New values were added, this were the small pizza, centred around 8 inches, and the large pizza centred around 12 inches.

### Clustering the data

Hierarchical clustering methods are able to make use a distance matrix to categorise which data is closest to which. The most commonly used measure for this is the Euclidean distance, or the square Euclidean distance, which as the name suggests is the same as the prior, merely squared. This distance allows the expression of points which span multiple dimensions in a single dimension. Where each point has a distance to all other points. The set of these distances is the called the distance matrix. The hierarchical clustering builds its clusters based on the relation of the points to each other using this as its reference.

To save on the number of computations needed the demonstration of the distance matrix creation, only the values from the original table, and not the ones with the margin of error included. The method of calculating the distance between two points is the Euclidean distance.

Equation 1;

As there is only two variables in the example being used here the equation will only expand to n=2 in the calculations shown. The distance between a point and itself will be zero. On the distance matrix table this lies on the diagonal. To begin with this distance is filled in first.

The distances that need to be calculated are Donut-Scones Donut-Pizza, and Scones-Pizza

Donut-Scones =

Calculation 1;

= 2.236  
Donut-Pizza =

Calculation 2;

= 5.099  
Scone-Pizza =

Calculation 3;

= 7

Using these values the rest of the distance matrix is made

|  |  |  |  |
| --- | --- | --- | --- |
|  | Donut | Scones | Pizza |
| Donut | 0 |  |  |
| Scones | 2.236 | 0 |  |
| Pizza | 5.099 | 7 | 0 |

Table 2; the completed distance matrix. Note the zero values along the diagonal.

If this were to be clustered the using the UPGMA method the average of the distances would be taken for 2 objects to be paired. The ones which are selected are those that are already the closest. In the above table it would be Pizza-Scones. The distance they are to the other objects in the table are removed, and replaced by a profile. This is the average of the distance to the other object, in this case Donuts. The example distance matrix is shown below.

|  |  |  |
| --- | --- | --- |
|  | Donut | Scones/Pizza |
| Donut | 0 |  |
| Scones/Pizza | 3.6675 | 0 |

Table 3; a distance matrix after a profile had been created.

This is of course just for demonstration stake. As stated earlier the 3 clusters for this data appears to be the best fit. As such there is no need to include this amalgamation. Using a function in R, which follows the same procedure as outline here, was applied to the expanded dataset. Including the small and large pizzas. The method of producing the profiles was the “average” or UPGMA (Unweighted Pair Group Method with Arithmetic Mean), as it is also known by. The resulting tree is shown below. As expected the small and large pizzas grouped with the original pizza sample. While two other distinct groups are the scone and donut samples.

#### Average linkage

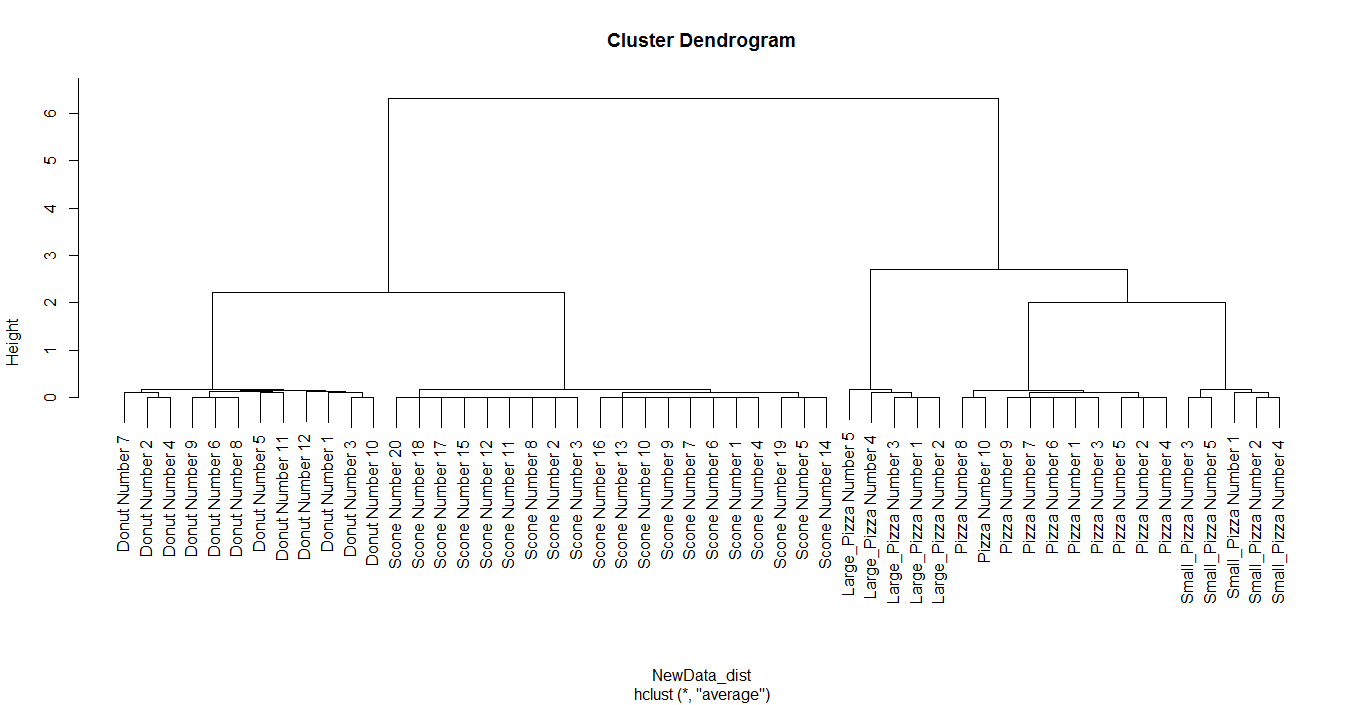


Figure 3; A tree of the hierarchical clustering of the hypothetical data, using average linkage.

In hierarchical clustering to access the clusters, the tree is cut and the components of the resulting clades are the clusters. In the given example the most obvious selection to me would be 4 clades, or clusters. However this does not give the clustering that would be expected when looking at the groups. The obvious clusters are the pizza (small medium and large all clustered together), scones and donuts. This raises the question of why the clustering is not like this. The average linking method used creates a grouping between the medium pizza and either the large or small pizza, with these being the smallest distance. The resulting profile is however is stretched away from the medium. Causing the next smallest distance to be the scones donuts. Despite the fact it is obvious to the human eye to expand the pizza cluster rather than create a donut/scone cluster.

#### Single Linkage

To avoid this an alternative method to UPGMA should be used. Given the explanation of why the clusters divided the way they did a more appropriate method of building the tree would be single linkage. Where UPGMA would use the average of the clusters, and pair this with the nearest neighbour based on these averages, the single linkage method pairs clusters based on the shortest distance between two components of the clusters. This means that where the average of the medium pizza and the small large pizza is was pulled away from the medium, which would have been the ideal medium of the larger all-inclusive pizza cluster, it will no longer affect the large pizza grouping with this cluster. As the short distance between the medium pizza and the large pizza is no longer ignored in favour of the average of the small and medium pizza profile. The single linkage method also does not create a distinction in the relation of the small and large pizza to the medium pizza as the UPGMA method had.

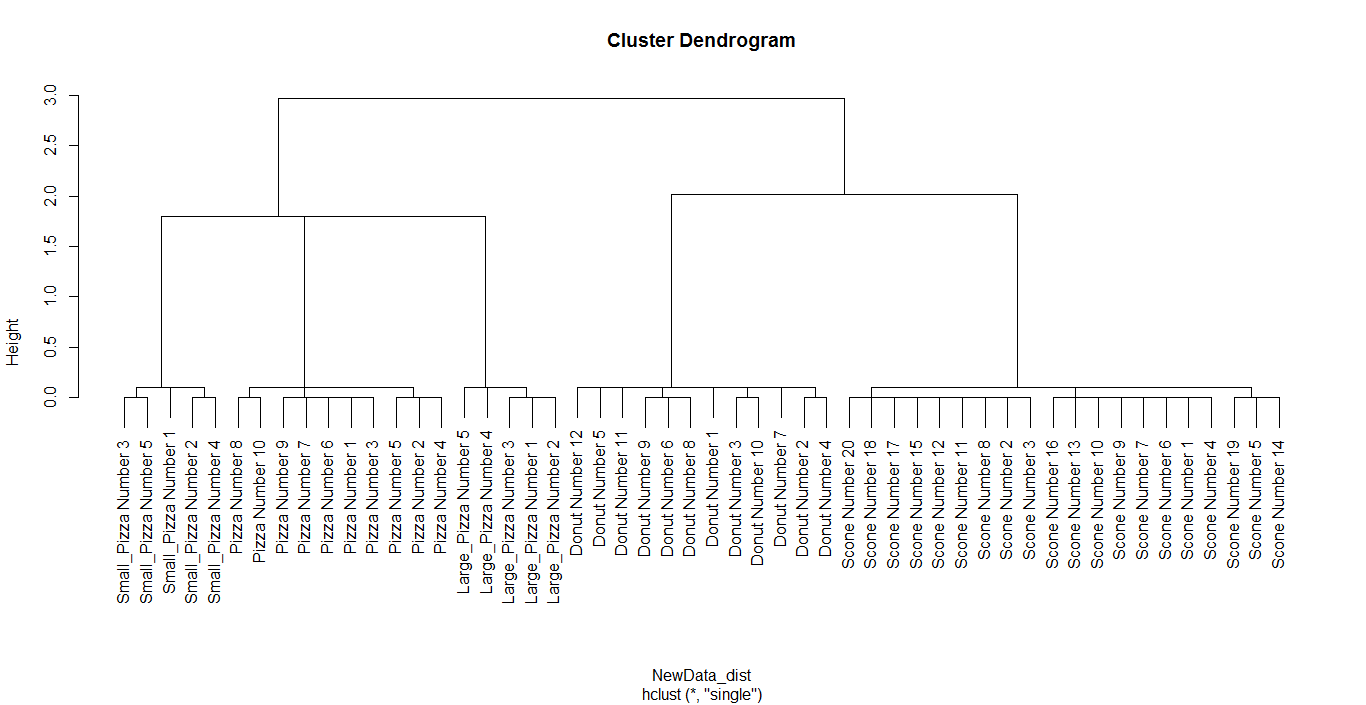


Figure 4; The same data as the earlier previous tree but not clustered using single linkage methodology.

#### Wards Linkage

Another method to determine how the tree would be formed is via wards methods. Or wards linkage. This method, similar to the other mentioned methods, takes the clusters of minimum distance and fuses them, to create a new cluster. It is of course in the definition of this distance. Wards method defines the distance as the sum of variances created by joining the clusters. As such the selection of the shortest distance makes wards method one which minimises the variance in the tree. The clustering from this method appears more appropriate as seen the increased difference in the heights of the pizza cluster and non pizza clusters clade.

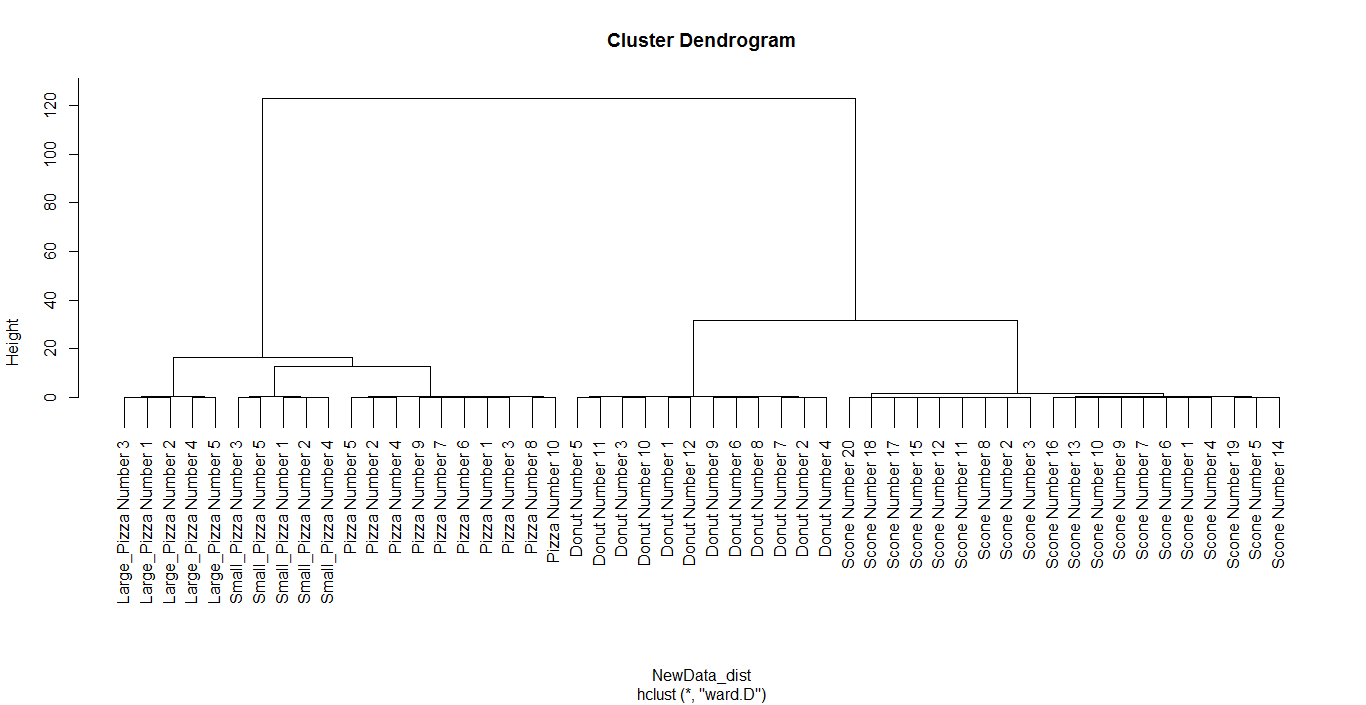


Figure 5; The simulated data clustered with wards method. Showing a more appropriate clustering pattern.

### Clustering scores

#### Silhouette

Various methods exist to evaluate goodness of a cluster. These methods are often to estimate the ideal number of clusters as well, by estimate the validity of each cluster at various values of k and returning the average of the clusters scores. By using this average score, a glimpse is given into the overall validity of the clusters as a whole, and thus gives estimates on the appropriate value of k to be used.

The silhouette method was first described by Rousseuw in 1986 [12]. The silhouette method measures the similarity of the clusters data points to each other, and the dissimilarity of the points to the centre of the nearest cluster to that point.

Equation 2; . Where a(i) is the average dissimilarity of i, with its own cluster, and b(i) is the dissimilarity of i and the closest point to i which is not part of its cluster.

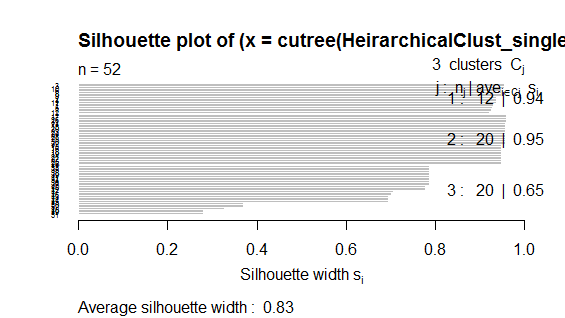


Figure 6; an example of graphical output from the silhouette calculation, at k =3.

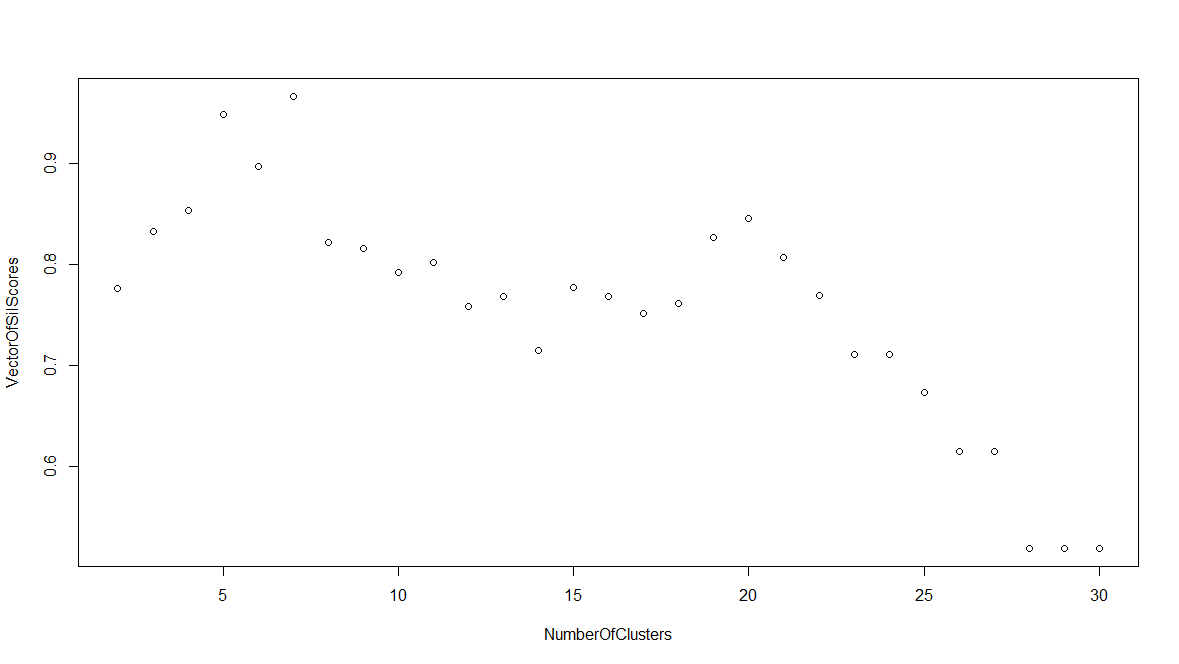


Figure 7; A figure of the average silhouette scores for the number of cluster included in the model.

Examining the figure of scores vs the number of clusters it is apparent that the method suggest that the best model uses a cluster number of 7, with 5 being a close second. While 6 has a noticeable drop off in comparison but would still be a decent classification for the data. The high score for 5 is expected as there is five values around which each set of data is centred. The value of 3 clusters is still seen to be quite high as it clustering of the small and large pizza with the medium pizza, does not cause the too much dissimilarity in the cluster, but obviously more is created than the when it is divided into its 5 clusters. High values are again seen at k = 7. This is dividing the scones in into multiple clusters. It is only via human intuition that this can be spotted and accounted for.

#### Gap Statistic

The gap statistic was proposed by Tibshirani *et al*, as formalised method for the “elbow method” heuristic of choosing the optimal cluster size [13]. The graph is highlights the most optimal point with the broken vertical blue line. The criteria for selecting a point is twofold. First the data is compared to a reference distribution to see if any clustering is appropriated. If not the “optimal” cluster is decided to be zero. For data where the clustering is appropriate the value of k chosen is the lowest value of k which differs from k+1 by at least one standard error. In figure 8 this is difficult to see clearly. However a careful examinations shows that the points are indeed one standard error apart.

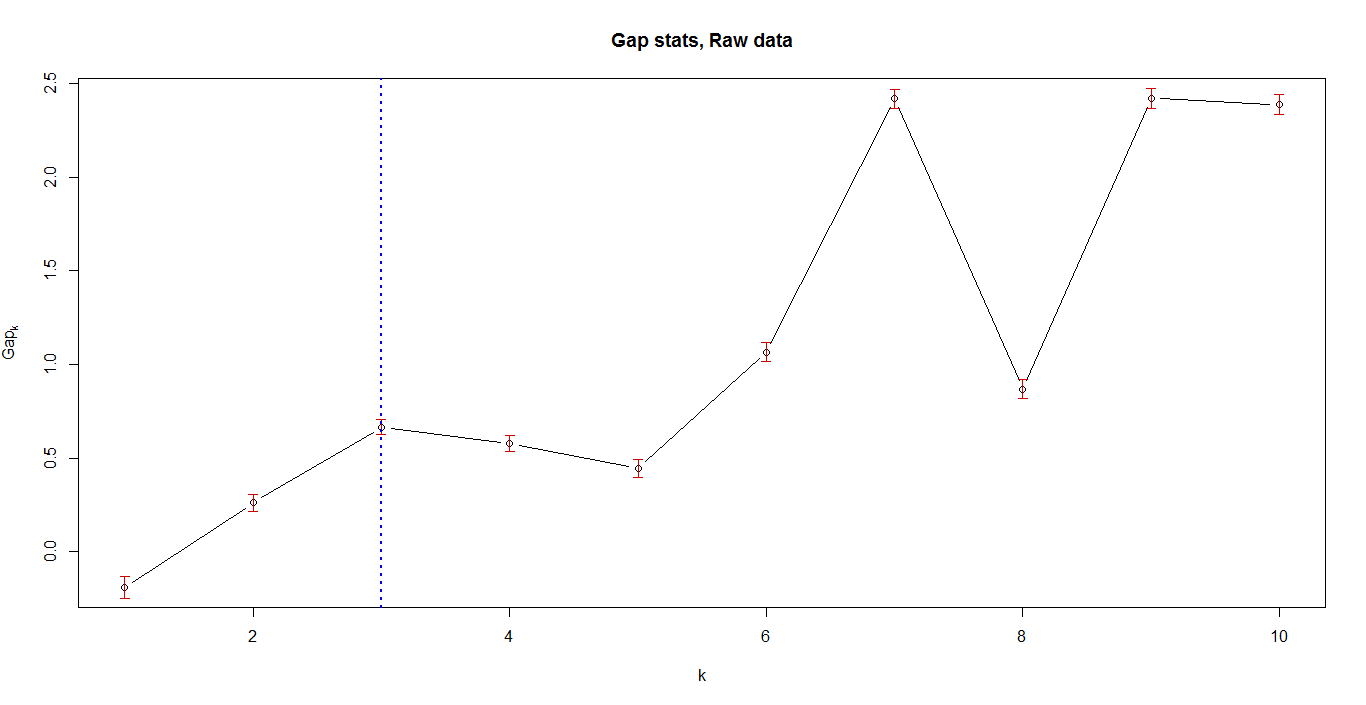


Figure 8; A graphical output of the gap statistic applied to the data. The broken blue line shows the value of k which the gap statistic considers he best. Here it is 3.