Cluster generators for large high-dimensional data sets with large numbers of clusters

Julia Handl
University of Manchester
J.Handl@postgrad.manchester.ac.uk

Joshua Knowles
University of Manchester
J.Knowles@manchester.ac.uk

1 Data generators

In order to obtain test data of sufficient complexity, two new cluster generators were developed. Both generators and the specific test data sets used in this paper are made available at www.dbk.ch.umist.ac.uk/handl/generators/.

1.1 Gaussian cluster generator

The first generator is based on a standard cluster model using multivariate normal distributions. The covariance matrices need to be symmetric and positive definite, which is ensured by constructing them to be diagonally dominant, with positive diagonal elements only.

Briefly, a single multivariate cluster is defined as follows:

1. the mean, uniformly in the range $[-10, 10]$
2. the off-diagonal entries of the covariance matrix, generated as a random number in the range $[-1, 1]$ with a distribution following $\pm y, y = x^2$ where x is a uniformly random deviate in $[0, 1]$ and the sign of y is determined randomly by a pseudorandom 'coin toss'
3. the diagonal entries of the covariance matrix, generated as the sum of all off-diagonal entries plus a random number in the range $[0, 20 \cdot \sqrt{D}]$ with a distribution following $\pm y, y = x^2$, where x is a uniformly random deviate in $[0, 1]$, the sign of y is determined randomly by a 'coin toss', and D is the dimensionality of the data set

Here, the use of a distribution $y = x^2$ serves to encourage the production of elongated clusters.

Data generation is based on a simple trial-and-error scheme. Clusters are iteratively constructed, and a simple heuristic is used to detect overlap between them. Overlapping clusters are rejected and regenerated, until a valid set of clusters has been found. Using this method, 80 different data sets were generated, as described in Table 1.

In low dimensions, the clusters generated are frequently elongated and of arbitrary orientation (see Figure 1). However, in higher dimensions (that is, more than 10), the shape of a cluster becomes more (hyper)spherical and more axis-aligned: the former because a high variance in one dimension hardly affects the Euclidean distance of points when there are very many dimensions; the latter because with higher dimensions the non-diagonal entries in the covariance matrix are forced to be relatively smaller compared with those on the diagonal. Because of the lack of generality of spherical clusters, we have developed a second alternative cluster generator that produces more elongated cluster shapes in arbitrarily high dimensions.

Table 1: Gaussian data sets of low dimension. For each of the 8 combinations of cluster number and dimension, 10 different instances were generated, giving 80 data sets in all.

Parameter	*range*
Number of clusters	4, 10, 20, 40
Dimension	2, 20
Size of each cluster	uniformly in $[50, 500]$ for 4 and 10 cluster instances, and $[10, 100]$ for 20 and 40 cluster instances

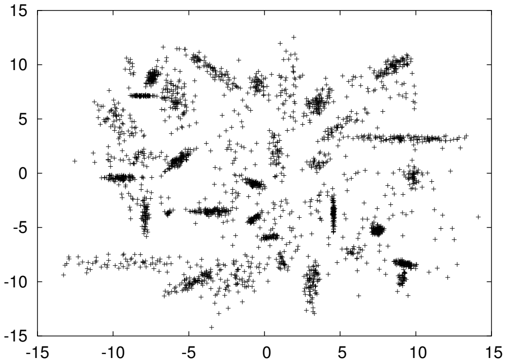

Figure 1: Examples of a data set generated with the Gaussian cluster generator. This data is two-dimensional and contains 40 clusters.

1.2 Ellipsoidal cluster generator

This generator creates ellipsoidal clusters with the major axis at an arbitrary orientation. The boundary of a cluster is defined by four parameters:

1. the origin (which is also the first focus)
2. the interfocal distance, uniformly in the range $[1.0, 3.0]$
3. the orientation of the major axis, uniformly from amongst all orientations
4. the maximum sum of Euclidean distances to the two foci, uniformly in the range $[1.05, 1.15]$ — equivalent to an eccentricity ranging from $[0.870, 0.952]$

For each cluster, data points are generated at a Gaussian-distributed distance from a uniformly random point on the

Table 2: Ellipsoidal data sets of high dimension. For each of the 8 combinations of cluster number and dimension, 10 different instances were generated, giving 80 data sets in all.

Parameter	*range*
Number of clusters	4, 10, 20, 40
Dimension	50, 100
Size of each cluster	uniformly in $[50, 500]$ for 4 and 10 cluster instances, and $[10, 100]$ for 20 and 40 cluster instances

major axis, in a uniformly random direction, and are rejected if they lie outside the boundary.

After the data points of all clusters are generated (with the origin initially at $0, \ldots, 0$), a genetic algorithm is used to translate the location of the origin of each cluster so that a cost consisting of the overall deviation of the entire data set, plus a penalty term for any overlapping clusters, is minimized. This has the effect of 'arranging' the clusters in a compact configuration (see Figure 2).

Using this method, 80 different data sets were generated, as described in Table 2.

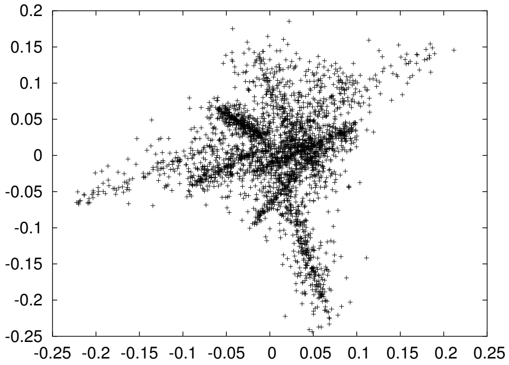

Figure 2: Examples of a data set generated with the Gaussian cluster generator. This data is 100-dimensional and contains 10 clusters (projection to two dimensions).