FastMap in dimensionality reduction: ensemble clustering of high dimensional data

Imran Khan*
Shenzhen Key Laboratory of High Performance Data Mining, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China
Email: imran.khan@siat.ac.cn
*Corresponding author

Joshua Z. Huang
Shenzhen Key Laboratory of High Performance Data Mining, Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China

and

College of Computer Science and Software Engineering, Shenzhen University, Shenzhen 518060, China
Email: zx.huang@siat.ac.cn

Abstract: In this paper we propose an ensemble clustering method for high dimensional data which uses FastMap projection (FP) to generate component datasets. In comparison with subspace component data generation methods such as random sampling (RS), random projection (RP) and principal component analysis (PCA), FP can better preserve the clustering structure of the original data in the component datasets so that the performance of ensemble clustering can be improved significantly. We present experiment results on six real world high dimensional datasets to demonstrate the better preservation of the clustering structure of the original data in the component datasets generated with FastMap, in comparison with the component datasets generated with RS, RP and PCA. The experiment results of 12 ensemble clustering methods from combinations of four subspace component data generation methods and three consensus functions also demonstrated that the ensemble clustering methods with FastMap outperformed other ensemble clustering methods with RS, RP and PCA. Ensemble clustering with FastMap also performed better than the k-means clustering algorithm.

Keywords: ensemble clustering; FastMap; RS; random sampling; RP; random projection; PCA; principal component analysis.


Copyright © 2017 Inderscience Enterprises Ltd.
1 Introduction

The emergence of new application domains results in very high dimensional big data such as text data, microarray data and smart phone user behaviour data. Such high dimensional data with thousands of features presents a big challenge to current data mining techniques. Among others, the curse of dimensionality and sparsity are two main problems that handicap many clustering algorithms to find strongly cohesive clusters from very high dimensional big data. The curse of dimensionality is a big problem faced in high dimensional data analysis, because as the number of dimensions in a dataset increases, the evaluating distance across all attributes becomes increasingly meaningless. Furthermore, clusters often exist in different subspaces, comprised of different types of dimensions. Ensemble clustering for full space clustering algorithms is not useful to cluster such data. The innovation of subspace ensemble clustering techniques is promised to resolve this problem.

Ensemble clustering is a new approach for clustering that integrates results of different clusterings generated from the same original data by different clustering algorithms or from different component datasets sampled from the original data (Strehl and Ghosh, 2002; Kuncheva and Hadjitodorov, 2004). For high dimensional data, the accuracy of ensemble clustering is always better than those of individual clusterings. A large number of research results have accelerated this field (Chen and Yang, 2009). Different ensemble clustering methods have been proposed to ensemble different types of clusterings like ensemble of clusterings from the same data using one algorithm with different parameters.
FastMap in dimensionality reduction

initialisations (Kuncheva and Vetrov, 2006), ensemble of clusterings from the same data using different algorithms (Topchy and Jain, 2004), and ensemble of clusterings from multiple component datasets sampled on the same dataset (Domeniconi and Al-Razgan, 2009). Subspace ensemble clustering is a useful strategy to find robust clusters from sparse high dimensional data.

Recently, two methods for generating low dimensional component data are used in subspace ensemble clustering of high dimensional data. One method generates low dimensional data by randomly sampling different features. The other method generates low dimensional component data by using a random projection (RP) matrix to project the original high dimensional data onto a low dimensional space. We call the former random sampling (RS) and the later random projection. Different flavours of RP are available (Kumar, 2011; Zhu et al., 2012). In this work, we use Fern and Brodley (2003) RP method for generation of multiple component datasets in ensemble clustering of high dimensional data. These methods benefit ensemble clustering for high dimensional sparse data. For ensemble clustering with principal component analysis (PCA), we propose random sub-sampling on principal dimensions generated with PCA from high dimensional data. Random subsampling on principle dimensions is performed with replacement. However, the drawback of these methods is that they cannot well preserve the clustering structure of the original data in their generated low dimensional component data, which increases discrepancy of the clustering structures in component datasets, thus affecting the performance of ensemble clustering for high dimensional data.

In this paper, we present a new low dimensional component data generation method by FastMap (Faloutsos and Lin, 1995) for ensemble clustering, an algorithm that is used to generate different low dimensional component datasets of high dimensional data. Given the mutual Euclidean distance matrix of \( N \) objects, FastMap uses the well known Cosine Law to compute \( k \)-dimensional data representation of the \( N \) objects. Each iteration of FastMap process selects a pair of points with a comparatively large distance from \( N \) objects, called pivots, to compute the coordinates of the \( N \) objects that are projected to the line of the pivot objects. By removing the distance component of the new generated dimension, a new set of coordinates are computed. In this way, after \( k \) iterations, the \( N \) objects are presented as a \( k \)-dimensional data matrix. The randomly selection of pivot objects produces diverse low dimensional component datasets. The generation of diverse component datasets by FastMap improves the performance in ensemble clustering of high dimensional data. The advantage of FastMap projection (FP) in comparison with RS, RP and PCA is that it can better preserve the clustering structure of the original data in its generated component datasets. Due to this reason, the performance of ensemble clustering is improved significantly.

We propose three methods to measure preservation of the clustering structure of the original data in the generated component datasets. We conducted comparison studies on six real world high dimensional datasets. The comparison results have shown that FastMap preserved the clustering structure better than other three methods. We have also conducted the ensemble clustering experiments with four component data generation methods and three consensus functions to ensemble clustering results. \( k \)-means algorithm was used to generate component clusterings. The results have shown that the ensemble clustering with FP outperformed the ensemble clusterings with RS, RP, and PAC on all six datasets. The overall performance of FastMap was the best among the three methods and the \( k \)-means algorithm.
2 Framework for subspace ensemble clustering

Ensemble clustering of a dataset $X$ is a process to integrate multiple clustering results produced by one or more clustering algorithms from component datasets sampled from $X$ into a single clustering of $X$ with a result that is usually much better than the results of individual clusterings on $X$ (Strehl and Ghosh, 2002). The generic subspace ensemble clustering framework consists of the following steps.

- **Step 1**: Use a component data generation method to generate $K$ different subspace component datasets $\{C_1, C_2, \ldots, C_K\}$ from $X$.
- **Step 2**: Use one or more clustering algorithms to cluster the $K$ component datasets to produce $K$ component clusterings $\{\pi_1, \pi_2, \ldots, \pi_K\}$ independently.
- **Step 3**: Ensemble $K$ component clusterings into a single clustering $\pi$ using an ensemble method called a consensus function.

Figure 1 shows the generic framework of subspace ensemble clustering.

![Generic framework of subspace ensemble clustering](image)

2.1 Subspace component generation

In ensemble clustering of high dimensional data, the better performance of ensemble clustering on high dimensional data can be achieved by generating low dimensional component datasets that can better preserve the clustering structure of the original data. Recently, RP and RS are most commonly used methods for low dimensional component data generation. We review these two methods and PCA briefly below.

2.1.1 Random projection

In random projection, a dataset $X_{N \times m}$ with $m$ dimensions and $N$ objects is projected to a $p$-dimensional ($p \ll m$) subspace to generate a low dimensional dataset $Y_{N \times p}$, using a random matrix $R_{m \times p}$, with each column unit length (Bingham and Mannila, 2001). The project is written as

$$Y_{N \times p} = X_{N \times m} \times R_{m \times p}$$

(1)
FastMap in dimensionality reduction

$Y_{N \times p}$ is the projection of the data onto a low $p$-dimensional subspace. The idea of RP came from Johnson-Lindenstrauss Lemma (Johnson et al., 1986). The computational cost of RP is $O(d \times m \times N)$.

The selection of random matrix $R$ is an interesting point. The value of each element $r_{ij}$ in matrix $R$ should follow Gaussian distribution. Gaussian distribution was replaced with very simple distributions by Achlioptas (2001) as

$$r_{ij} = \sqrt{3} \begin{cases} +1 & \text{with probability } \frac{1}{6} \\ 0 & \text{with probability } \frac{2}{3} \\ -1 & \text{with probability } \frac{1}{6} \end{cases}$$

where the columns of matrix $R$ are zero-mean normal variables with unit length. In our experiments, we have used equations (1) and (2) for generation of component datasets.

2.1.2 Random sampling

A few methods are available for generation of low dimensional component datasets by randomly selection of features from the original space. These methods are distinguished in the way to select dimensions from the original high dimensional dataset. In literature, a most commonly used method is RS that uses a threshold on a feature or a set of features, and a feature is selected to add in the component dataset if the corresponding value exceeds the threshold value. The features can be selected randomly by using any of the following proposed methods like Gini index, Quinlan’s information gain ratio or Mingers’s G statistic. In our experiments, we have used Quinlan’s information gain ratio for feature selection.

2.1.3 Principal component analysis

Principal component analysis (PCA) is a statistical based method that uses orthogonal transformation to transform a set of all possible correlated variables into a set of linearly uncorrelated variables called principal components (PC). The total number of generated components is usually less than or may be equal to the total number of variables. The decomposition of eigenvalues from the data covariance matrix is generated as $E\{XX^T\} = E\Lambda E^T$. In matrix $E$, each column is considered as an eigenvector of the data covariance matrix $E\{XX^T\}$ and matrix $\Lambda$ contains eigenvalues. The dimensionality of a dataset $X$ can be reduced by considering the most important eigenvectors.

$$X^{PCA} = E_k^t X$$

where matrix $E_k$ contains $k$ eigenvectors. The computational cost of eigenvalue decomposition of the data covariance matrix is very expensive. The time complexity of PCA is $O(d_2N) + O(d_3)$ (Golub and van Loan, 1983). PCA is the best in the sense of mean-square error and linear dimensionality reduction (Jackson, 1991; Jolliffe, 1986).

In recent days, PCA is the most commonly used dimension reduction technique. The maximised data variance is a critical factor for PCA projection. However, the variance criterion used by PCA cannot guarantee a good low-dimensional representation of data for clustering (Fukunaga, 1990). Also, PCA is a deterministic approach. In order to use PCA for ensemble clustering of high dimensional data, we consider a random sub-sampling method with PCA. That is, after the component dimensions are obtained, we randomly sample the component dimensions with replacement to generate low dimensional component datasets, each with different combinations of component dimensions.
2.2 Component data clustering

The clustering of a low dimensional component dataset can be performed using any clustering algorithm. The popular clustering algorithms include $k$-means, subspace $k$-means and hierarchical clustering methods. Each clustering algorithm has its pros and cons. The advantage of the $k$-means (MacQueen, 1967) algorithm is its efficiency in handling large data. In this work, we used $k$-means. Quite often, different clustering algorithms were used to generate different component clustering results for ensemble clustering. However, there is no clear guidance how the different clustering algorithms should be used. In practice, it is more convenient to use one clustering algorithm for ensemble clustering, rather than multiple clustering algorithms. The clustering process of $k$-means is to minimise the following objective function.

\[
F = \sum_{j=1}^{k} \sum_{i=1}^{n} \| x^{(j)}_i - c_j \|^2 
\]

(4)

where $\| x^{(j)}_i - c_j \|^2$ is the distance between an object $x^{(j)}_i$ and the centroid $c_j$.

2.3 Ensemble component clusterings

A component generation method is used to generate multiple low dimensional component datasets. Multiple component clusterings are assembled into a single clustering as the final clustering result. An ensemble method, often called consensus function, is used to ensemble multiple component clusterings. Several consensus functions have been proposed with different strategies and methods to ensemble component clustering results. Below, we briefly review three ensemble methods that were used in this work.

2.3.1 Similarity-based consensus function

A clustering signifies a relationship between objects in the same cluster and can thus be used to establish a measure of pairwise similarity (Strehl and Ghosh, 2002). For each component clustering, a similarity matrix is constructed. In the similarity matrix, the element indexing two objects in the same cluster is assigned value 1, otherwise, the element has value 0 if the two objects are in different clusters. After computation of $K$ similarity matrices, a final matrix is obtained as the average of corresponding cells of all similarity matrices. The METIS algorithm (Karypis and Kumar, 1998) is then applied to the resultant similarity matrix to produce the final clustering ensemble.

2.3.2 Hyper graph-based consensus function

In hyper graph-based consensus function (HGPA), an ensemble problem is formulated as partitioning the hypergraph by cutting a minimal number of hyperedges (Strehl and Ghosh, 2002). The hyper graph is constructed by considering objects of a dataset $X$ as $N$ vertices, and hyper-edges with the same weight are used to connect a set of vertices by using $K$ component clusterings. The algorithm HMETIS (Karypis et al., 1999) is used to partition the hyper-graph into unconnected components by cutting a minimum number of hyper-edges.
2.3.3 Meta cluster-based consensus function

This method was introduced in the meta-clustering algorithm (MCLA) (Strehl and Ghosh, 2002). Similar to HGPA, a hyper graph is constructed by considering clusters of component clusterings as vertices and edges are used to connect these vertices. The weight between two vertices (clusters) is computed by using the following binary Jaccard distance equation.

\[
\text{JacSim}(C_x, C_y) = \frac{C'_x C_y}{\|C_x\|^2 + \|C_y\|^2 - C'_x C'_y}
\]

where \(C'_x\) and \(C'_y\) are two vectors of \(N\) elements representing two clusters where \(N\) is the number of objects in the dataset \(X\). Each element in the vector corresponds to one object. If the cluster contains the object, the corresponding element is assigned to 1. Otherwise, the element is 0. \(C'_x\) is the transpose of \(C_x\). METIS is used to partition this hyper graph to identify \(K\) meta-clusters. Finally, a voting method is used to assign each data point to its most associated meta-cluster.

3 FastMap projection for component data generation

3.1 FastMap projection

FastMap (Faloutsos and Lin, 1995) is an efficient algorithm to generate \(k\) dimensional coordinates of \(N\) objects from a dissimilarity matrix of the \(N\) objects. Given a high dimensional data \(X\) of \(m\) dimensions and \(N\) objects, we use a distance function to compute the distance matrix \(D_{N \times N}\). The distance function should satisfy the following properties of triangular inequality.

- \(d(O_a, O_b) = 0\),
- \(d(O_a, O_b) = d(O_b, O_a)\),
- \(d(O_i, O_b) \leq d(O_i, O_a) + d(O_b, O_a)\)

We select two objects \(O_a\) and \(O_b\) with a large distance as pivot objects and take the straight line passing the two objects as the projection axis of the first dimension coordinates \(J_1\). The coordinate of object \(O_i\) in the first dimension is computed by using the following cosine equation.

\[
x_{1,i} = \frac{d^2(O_a, O_i) + d^2(O_a, O_b) - d^2(O_b, O_i)}{d^2(O_a, O_b)}
\]

where \(d^2(O_a, O_b)\) is the distance between the pivot objects \(O_a\) and \(O_b\), \(d^2(O_a, O_i)\) is the distance between the pivot object \(O_a\) and object \(O_i\) and \(d^2(O_b, O_i)\) is the distance between the pivot object \(O_b\) and object \(O_i\).

After the coordinates of all \(N\) objects are computed, we compute the reduced distance matrix \(D'\) of \(N\) objects according to Lemma 1 in Faloutsos and Lin (1995) as

\[
d'(O_i, O_j)^2 = \sqrt{d(O_i, O_j)^2 - (x_i - x_j)^2}
\]
where \( d' \) is the reduced distance in \( D'_{N\times N} \), \( d \) is the distance in \( D_{N\times N} \) and the last term on the right is the squared distance in the new dimension.

Given \( D'_{N\times N} \), we can choose a new pair of pivot objects and use equation (6) to compute the coordinates of the second dimension. We repeat this process \( k \) times to generate a \( k \)-dimensional data of \( X \). The pivots objects are chosen by the following given Algorithm 1.

Algorithm 1: Heuristic to choose two distinct objects for iteration \( i \)

**Data:** \( X, \text{dist()} \)

**Result:** pivot objects \( O_a \) and \( O_b \)

1) Choose arbitrarily as object, and declare it to be the second pivot object \( O_b \)
2) Set \( O_a = (\text{the object that is farthest apart from } O_b ) \)
3) Set \( O_b = (\text{the object that is farthest apart from } O_a ) \)

4 Experiments

In this section, we present different experiments on real world datasets to evaluate the performance of ensemble clusterings with FP. The FastMap ensemble clustering results are compared with the results of RS, RP, and PCA. For experiments, six high dimensional datasets were selected from different application domains. The datasets were downloaded from web sources of UCI machine learning repository and feature selection datasets at Arizona state university. The characteristics of these datasets are listed in Table 1.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>#Instances</th>
<th>#Features</th>
<th>Source</th>
<th>#Classes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DBWorld emails</td>
<td>64</td>
<td>4702</td>
<td>Text</td>
<td>02</td>
</tr>
<tr>
<td>GLI-85</td>
<td>85</td>
<td>22,283</td>
<td>Microarray</td>
<td>02</td>
</tr>
<tr>
<td>Orlraws10P</td>
<td>100</td>
<td>10,000</td>
<td>Image</td>
<td>10</td>
</tr>
<tr>
<td>Internet Ad</td>
<td>1000</td>
<td>1558</td>
<td>Multivariate</td>
<td>02</td>
</tr>
<tr>
<td>CLL-SUB-111</td>
<td>111</td>
<td>11,340</td>
<td>Microarray</td>
<td>03</td>
</tr>
<tr>
<td>SRBCT</td>
<td>83</td>
<td>2308</td>
<td>Microarray</td>
<td>04</td>
</tr>
</tbody>
</table>

4.1 Experiment settings

We used four methods to generate component datasets, i.e., RS, RP, PCA, and FP. For component datasets, the \( k \)-means algorithm was used for clustering. The number of clusters \( k \) was given as the number of classes in the dataset. For ensemble clustering, we used the three consensus functions in Section 3, i.e., hyper graph based consensus function (HGPA), similarity-based consensus function (CSPA) and meta cluster-based consensus function (MCLA). By combining the four component data generation methods and the three consensus functions, we produced 12 ensemble clustering results from each dataset. We denote these 12 ensemble clustering methods as RS-CSPA, RP-CSPA, PCA-CSPA, FM-CSPA, RS-MCLA, RP-MCLA, PCA-MCLA, FM-MCLA, RS-HGPA, RP-HGPA, PCA-HGPA and FM-HGPA, respectively.
For comparison, different component datasets with the same number of dimensions were generated from the original dataset by each component generation method. Each ensemble clustering was generated from 10 component clusterings produced with the \( k \)-means algorithm. The clustering results of the original dataset by the \( k \)-means algorithm were also presented.

### 4.2 Evaluation of component data generation

In this section, we discuss three approaches to analyse preservation of the clustering structure of the original data in generated component datasets for ensemble clustering.

#### 4.2.1 Intrinsic dimensionality

The clustering of high dimensional data is negatively impacted by a phenomena called the curse of dimensionality. We use intrinsic dimensionality to measure the curse of dimensionality of a dataset as in Avez and Navarro (2001). A low value of intrinsic dimensionality depicts the existence of tight clusters in data. A high value of intrinsic dimensionality shows that dataset is poorly intrinsically structured.

Given a high dimensional dataset \( X_{N \times m} \), we use a component data generation function \( \Phi(X, \theta) \) to generate a subspace data \( Y_{N \times p} \), i.e., \( \Phi(X, \theta) = Y_{N \times p} \) where \( \theta \) are input parameters to produce different \( Y \)'s from \( X \). Let \( Y = \{Y_1, \ldots, Y_K\} \) be a set of \( K \) component datasets all in \( p \) dimensions and \( D = \{D_1, \ldots, D_K\} \) the set of \( K \) distance matrices computed from \( Y \). Given a distance matrix \( D_i \), we take the upper half mutual distances of \( D_i \) and plot the histogram of the mutual distances. Large mean and small variance of the histogram distribution of \( D_i \) represent a problem of curse of dimensionality.

**Definition 1:** The intrinsic dimensionality of a dataset in a metric space is defined as \( \rho = \frac{\mu^2}{2\sigma^2} \) where \( \mu \) and \( \sigma \) are the mean and variance of its histogram of distances (Avez and Navarro, 2001).

We use intrinsic dimensionality \( \rho \) to evaluate a method \( \Phi(X, \theta) \). For each component \( Y_i \), we compute \( \rho_i \). Then, we compute the average \( \bar{\rho} \) of \( \rho_i \) of \( K \) component datasets in \( Y \). The smaller \( \bar{\rho} \) the better the method \( \Phi(X, \theta) \).

#### 4.2.2 Distance preservation

The projection methods are required to preserve the distance between data points in the projected low dimensional space. Otherwise a problem may exist since the triangular inequality may not be maintained. Let \( D \) be the distance matrix of high dimensional data \( X_{N \times m} \) and \( \{D_1, \ldots, D_K\} \) the set of \( K \) distance matrices computed from \( Y \). Given \( D \) and \( D_i \) from \( Y_i \), we compute

\[
\text{Dist} = \sqrt{\frac{\sum_{i=1}^{\bar{N}} \sum_{j=1}^{N} (d_s(o_i, o_j) - d_o(o_i, o_j))^2}{\sum_{i=1}^{\bar{N}} \sum_{j=1}^{N} d_o^2(o_i, o_j)}}
\]

(8)

where \( \bar{N} \) is the total number of objects in the datasets, \( d_o \) is the distance between two objects in dataset \( X_{N \times m} \) and \( d_s \) is the distance between corresponding two objects in the subspace component dataset \( Y_i \). \( \text{Dist} \) is a measure of distance preservation of component dataset.
Y generated from X. The smaller Dist, the better the component dataset Y in preserving the mutual distances of objects in X. Dasgupta (2000) has shown that RP preserves the separation among Gaussian clusters having large variances.

### 4.2.3 Sum of square error

The sum of square error (SSE) is a commonly used statistical technique to evaluate the overall clustering structure of a given dataset. The SSE is a summation of squared differences between each actual observation and its group mean value.

\[
\text{SSE} = \sum_{i=1}^{n} (x_i - \bar{x}_i)^2
\]

where \( x_i \) is an actual observation, \( \bar{x}_i \) is a group’s mean of that observation. If all the clusters in a given dataset are identical, the SSE value will be equal to 0. We used SSE as a measure on preservation of component datasets by FastMap. If FastMap preserves the original clustering structure of data in its generated subspace component datasets, the SSE difference of clusterings between the original dataset D and FastMap generated subspace component dataset \( Y_i \) must be smaller than those by RS and RP.

### 4.3 Comparisons of FastMap projection vs. random sampling, random projection and PCA

In our experiments, we used the four component data generation methods: RS, RP, PCA and FP to generate component datasets from six real world datasets in Table 1. We computed three measures to evaluate the preservations of the original data clustering structures in the component datasets. The RP is a computationally inexpensive method while preserving the distances of data vectors so does PCA (Bingham and Mannila, 2001).

Table 2 depicts the comparisons of three methods in intrinsic dimensionality. The evaluation measure was computed in three different dimensions of component datasets. The intrinsic dimensionality values in FastMap columns are surprisingly lower than those in RP, RS and PCA columns. The overall results show that FastMap can generate better component datasets than RS, RP and PCA. For GLI85 dataset with 85 objects and 22,283 features, FP performance is a little worse than RP and PCA in intrinsic dimensionality. Even with this result, we will show in the next section that ensemble clustering results of FP in this dataset are still better than the results from other three methods.

Table 3 depicts the comparisons of three methods in distance preservation. The distance preservation values in FastMap columns are quite lower than the values of RP, RS, and PCA.

<table>
<thead>
<tr>
<th>Datasets</th>
<th>5-dimensional space</th>
<th>10-dimensional space</th>
<th>15-dimensional space</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FM</td>
<td>RP</td>
<td>RS</td>
</tr>
<tr>
<td>DBWorld</td>
<td>0.013</td>
<td>0.057</td>
<td>2.618</td>
</tr>
<tr>
<td>Internet Ad</td>
<td>3E-05</td>
<td>5E-05</td>
<td>0.526</td>
</tr>
<tr>
<td>CLLSUB111</td>
<td>9E-13</td>
<td>4E-05</td>
<td>0.526</td>
</tr>
<tr>
<td>Oodraws10P</td>
<td>1E-06</td>
<td>5E-05</td>
<td>1E-03</td>
</tr>
<tr>
<td>SRBCT</td>
<td>0.006</td>
<td>0.009</td>
<td>0.383</td>
</tr>
<tr>
<td>GLI85</td>
<td>5E-11</td>
<td>2E-11</td>
<td>3E-09</td>
</tr>
</tbody>
</table>

Table 2 Intrinsic dimensionality

Table 3 Intrinsic dimensionality
columns. These results demonstrate that FP has better distance preservations than other three methods. For CLLSUB111 dataset with 111 objects and 11340 features, FastMap preserves the distance very well with a huge difference than those of RS, RP, and PCA.

Table 3 Distance preservation

<table>
<thead>
<tr>
<th>Datasets</th>
<th>5-dimensional space</th>
<th>10-dimensional space</th>
<th>15-dimensional space</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FM</td>
<td>RP</td>
<td>RS</td>
</tr>
<tr>
<td>DBWorld</td>
<td>0.461</td>
<td>0.471</td>
<td>0.948</td>
</tr>
<tr>
<td>Internet Ad</td>
<td>0.681</td>
<td>0.739</td>
<td>1.812</td>
</tr>
<tr>
<td>CLLSUB111</td>
<td>0.568</td>
<td>0.662</td>
<td>0.995</td>
</tr>
<tr>
<td>Orlraws10P</td>
<td>0.479</td>
<td>0.519</td>
<td>1.238</td>
</tr>
<tr>
<td>SRBCT</td>
<td>0.639</td>
<td>0.659</td>
<td>0.908</td>
</tr>
<tr>
<td>GLI85</td>
<td>0.596</td>
<td>0.639</td>
<td>0.960</td>
</tr>
</tbody>
</table>

The SSE difference of clusterings between the original dataset and the component dataset is shown in Table 4. The SSE difference shows that FastMap component datasets clustering structure is very close to the original clustering structure of data. The data clustering of each low dimensional component dataset shows a unique underlaying original clustering structure of data. Better clustering resulted were achieved by ensemble clustering of different component clusterings by FastMap as shown below.

Table 4 SSE difference

<table>
<thead>
<tr>
<th>Datasets</th>
<th>5-dimensional space</th>
<th>10-dimensional space</th>
<th>15-dimensional space</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>FM</td>
<td>RP</td>
<td>RS</td>
</tr>
<tr>
<td>DBWorld</td>
<td>0.16</td>
<td>0.18</td>
<td>0.93</td>
</tr>
<tr>
<td>Internet Ad</td>
<td>0.67</td>
<td>0.51</td>
<td>1.52</td>
</tr>
<tr>
<td>CLLSUB111</td>
<td>1.18</td>
<td>4.36</td>
<td>10.0</td>
</tr>
<tr>
<td>Orlraws10P</td>
<td>2.41</td>
<td>2.96</td>
<td>2.89</td>
</tr>
<tr>
<td>SRBCT</td>
<td>3.11</td>
<td>5.92</td>
<td>4.16</td>
</tr>
<tr>
<td>GLI85</td>
<td>1.64</td>
<td>1.64</td>
<td>1.69</td>
</tr>
</tbody>
</table>

4.4 Sensitivity of $p$ dimensions

The size of dimensions of component datasets has great impact upon the overall performance of clustering. The optimal selection of dimension size $p$ for generation of multiple component datasets is a critical issue. It is desirable to find an optimum value of $p$ to capture all necessary structure information of the original data. We use entropy measure to find the best $p$ value based on the following entropy equation.

$$ \text{Entropy} = - \sum_{l=1}^{k} p(C_l) \log_2 p(C_l) $$

where $k$ is the number of clusters, $p(C_l)$ is the probability of a cluster $l$ with respect to the total number of instance $N$, i.e., $p(C_l) = \frac{|C_l|}{N}$. The optimum $p$ value is found by the difference between the entropy measure of the original data clustering and the average entropy measure of different component datasets. The minimum entropy measure difference gives the best value of $p$ for component data generation.
4.5 Evaluation methods

We used four evaluation methods to evaluate the performance of ensemble clustering methods. One is an unsupervised method and three are supervised methods defined as follows.

The unsupervised evaluation method is cluster compactness (CP) calculated as

$$CP = \frac{1}{n} \sum_{x=1}^{n} n_x \left( \frac{\sum_{o_i, o_j \in C_x} d(o_i, o_j)}{n_x(n_x - 1/2)} \right)$$

(11)

where $d(o_i, o_j)$ is the distance between objects $o_i$ and $o_j$ in cluster $C_x$, $n_x$ is the number of objects in cluster $C_x$. The smaller the value of CP, the better the clustering result.

The three supervised evaluation methods are normalised mutual information (NMI), adjusted rand index (ARI) and clustering accuracy (CA), calculated as follows.

$$CA = \frac{1}{n} \sum_{x=1}^{k} \max_y n_{x,y}$$

(12)

$$ARI = \frac{\sum_{x=1}^{k} \sum_{y=1}^{k} (n_{x,y}) - s_3}{\frac{1}{2}(s_1 + s_2) - s_3}$$

(13)

$$NMI = \sqrt{\sum_{x=1}^{k} n_x \log \frac{2n_x}{n} \sum_{y=1}^{k} n_y \log \frac{2n_y}{n}}$$

(14)

where $n_x$ and $n_y$ are the total numbers of objects in cluster $x$ and class $y$ respectively, $n_{x,y}$ is the total number of objects in both cluster $x$ and class $y$, $n$ is the total number of objects in the given dataset, $s_1 = \sum_{x=1}^{k} \left( \frac{n_x}{2} \right)$, $s_2 = \sum_{y=1}^{k} \left( \frac{n_y}{2} \right)$ and $s_3 = \frac{2s_1s_2}{n(n-1)}$. The larger the values of these measures, the better the clustering result.

4.6 Experimental results

Table 5 shows the ensemble clustering results of six datasets produced with 12 ensemble clustering methods. The results were evaluated with one unsupervised and three supervised evaluation methods. The good results were marked in bold. The performance of ensemble clustering with FP under clustering accuracy evaluation is the best in all datasets compared with the ensemble clusterings with other two methods with the same consensus function. The highest clustering accuracy was obtained in all datasets by the ensemble clustering with FP among all ensemble clustering methods. The improvements in datasets DBWorld emails and Internet Advertisements were significant compared with other three methods. For instance, the highest clustering accuracy of clustering by FM-CSPA is 92%, much higher than ensemble clustering with other three methods, which are less than 65%. It is also higher than the result of k-means that is 87%.

The results of ensemble clustering with FP were also better than those with other two methods under NMI evaluation in all datasets. The majority best results were also obtained with FastMap method, except for two cases of GLI-85 and CLLSUB111. However, the difference was not very significant.

Under CP and ARI evaluations, the ensemble clustering with FP also outperformed the ensemble clusterings with other three methods in most datasets. The majority
best results also occurred in FastMap method. These results demonstrated that FP for component dataset generation improved the performance of ensemble clustering of high dimensional data.

### Table 5  Ensemble clustering result comparison

<table>
<thead>
<tr>
<th>Methods</th>
<th>BDWorld emails</th>
<th>GLI-85</th>
<th>Ofravo10P</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CP</td>
<td>NMI</td>
<td>ARI</td>
</tr>
<tr>
<td>RS-CSPA</td>
<td>19.2</td>
<td>0.29</td>
<td>0.50</td>
</tr>
<tr>
<td>RP-CSPA</td>
<td>18.2</td>
<td>0.48</td>
<td>0.50</td>
</tr>
<tr>
<td>PCA-CSPA</td>
<td>18.1</td>
<td>0.68</td>
<td>0.59</td>
</tr>
<tr>
<td>FM-CSPA</td>
<td>17.9</td>
<td>0.76</td>
<td>0.82</td>
</tr>
<tr>
<td>RS-HGPA</td>
<td>19.2</td>
<td>0.11</td>
<td>0.51</td>
</tr>
<tr>
<td>RP-HGPA</td>
<td>18.3</td>
<td>0.15</td>
<td>0.50</td>
</tr>
<tr>
<td>PCA-HGPA</td>
<td>18.5</td>
<td>0.14</td>
<td>0.53</td>
</tr>
<tr>
<td>FM-HGPA</td>
<td>18.3</td>
<td>0.16</td>
<td>0.55</td>
</tr>
<tr>
<td>RS-MLCA</td>
<td>18.0</td>
<td>0.50</td>
<td>0.54</td>
</tr>
<tr>
<td>RP-MLCA</td>
<td>18.2</td>
<td>0.60</td>
<td>0.50</td>
</tr>
<tr>
<td>PCA-MLCA</td>
<td>18.3</td>
<td>0.63</td>
<td>0.51</td>
</tr>
<tr>
<td>FM-MLCA</td>
<td>17.9</td>
<td>0.73</td>
<td>0.77</td>
</tr>
</tbody>
</table>

### 5 Conclusions

In this paper, we have presented the FP method to generate low dimensional component datasets in ensemble clustering. We have analysed and demonstrated that FP can better preserve the clustering structure of the original data than RS, RP and PCA. The experiments on six real world datasets have shown that the ensemble clustering with FP outperformed ensemble clusterings with other three methods. Beside better performance, another advantage of FastMap is that it is efficient in handling big data and flexible in component data generation.

### Acknowledgements

This research work was supported by the National Natural Science Foundation of China under Grants No. 61175123, and the Shenzhen New Industry Development Fund under grant No. JCYJ2012061720716224.
References


