Abstract

In this paper, we shall implement ten techniques which can be used to reduce the dimensionality of a data set. These include random projection (RP), principal component analysis (PCA), the variance approach (Var), the combined approach (CA), the direct approach (DA), the new random approach version 1 (NRA v1), the new random approach version 2 (NRA v2), and three novel approaches (Nov App1, Nov App2 and Nov App3) proposed by Baba, Nsang and Adeseye[1]. We shall investigate the relative effective performances of these techniques based on four different criteria: classification preservation, variance preservation, interpoint distance preservation and total run time. We shall then describe hierarchical clustering, and compare the ten dimensionality reduction techniques mentioned above for hierarchical clustering preservation.

Index Terms— dimensionality reduction, clustering, the rand index

1. Introduction

Dimensionality Reduction is an interesting and important concept in Machine Learning. It enables us to reduce a data set containing \( n \) points in high dimensional space into a lower dimensional space, making sure the result obtained by working in the lower dimensional space is a good approximation to the result obtained by working in the original high dimensional space[2]. In other words, given a \( p \)-dimensional data set \( X = (x_1, \ldots, x_n)^T \), \( s \in \mathbb{R}^p \), we are to find a lower dimensional representation of it, \( S = (s_1, \ldots, s_k)^T, s \in \mathbb{R}^k \), with \( k \leq p \), that captures the content in the original data, according to some criterion.

Dimensionality reduction has several advantages, the most important of which is the fact that with dimensionality reduction, we could drastically speed up the execution of an algorithm whose runtime depends exponentially on the dimensions of the working space[2]. At the same time, the solution found by working in the low dimensional space is a good approximation to the solution in the original high dimensional space.

Dimensionality reduction can be applied in several domains which include: text retrieval, image compression, nearest neighbor search, similarity search in a time series database, learning, clustering/classification, signal processing etc. [2]

In this paper, we implement ten dimensionality reduction techniques and then compare their relative effective performances based on four criteria: classification preservation, variance preservation, interpoint distance preservation and total run time. The rest of this paper is organized as follows. In Section 2, we shall briefly describe the ten dimensionality reduction approaches we have implemented. In Section 3, we shall compare the relative effective performances of these approaches based on the four criteria described above. In Section 4, we shall define clustering, and discuss the various distance metrics that can be used to group the points of a dataset into different clusters. We shall also discuss the hierarchical clustering algorithm, and compare the ten dimensionality reduction techniques discussed in the paper for hierarchical clustering preservation. Then we shall conclude this paper in Section 5.

2. The Dimensionality Reduction Techniques

The dimensionality reduction techniques which shall be implemented in this paper include:

2.1 Random Projection (RP)

In Random Projection, the original \( d \)-dimensional data is projected to a \( k \)-dimensional \( (k << d) \) subspace through the origin, using a random \( d \times k \) matrix \( R \) whose columns have unit lengths [3]. If \( X_{n,d} \) is the original set of \( n \) \( d \)-dimensional observations, then

\[
X_{n,d}^{RP} = X_{n,d}R_{d \times k}
\]

is the projection of the data onto a lower \( k \)-dimensional subspace.
2.2 Principal Component Analysis (PCA)

Given \( n \) data points in \( \mathbb{R}^p \) as an \( n \times p \) matrix \( X \), we want to find the best \( q \)-dimensional approximation for the data (\( q < p \)). The PCA approach achieves this by first computing the Singular Value Decomposition of \( X \). In other words, it finds matrices \( U, D \) and \( V \) such that \( X = UDV^T \) where:

- \( U \) is an \( n \times n \) orthogonal matrix
- \( V \) is a \( p \times p \) orthogonal matrix and
- \( D \) is an \( n \times p \) diagonal matrix
- Define \( U_q \) to be the matrix consisting of the first \( q \) columns of \( U \)

The transformed matrix is given by:

\[
X_{\text{SVD}} = X^T U_q [3]
\]

2.3 The Variance Approach [4]

With the Variance approach, to reduce a dataset \( D \) to a data set \( D_k \) we start with an empty set, \( I \), and then add dimensions of \( D \) to this set in decreasing order of their variances. That means that a set of \( I \) dimensions will contain the dimensions of top \( r \) variances. Thus, let \( I = [i_1, \ldots, i_k] \subset [1, \ldots, n] \), the collection of dimensions corresponding to the top \( r \) variances. The reduced data base is obtained by extracting the data corresponding to the selected dimensions. That is,

\[
D_k = D(:, I)
\]

2.4 The Combined Approach [4]

Like the two previous approaches, the Combined Approach is one approach which reduces a dataset \( D \) to a subset of the original attribute set.

To reduce a dataset \( D_{\text{exp}} \) to a dataset containing \( k \) columns, the Combined Approach selects the combination of \( k \) attributes which best preserve the interpoint distances, and reduces the dataset to a dataset containing only those \( k \) attributes. To do so, it first determines the extent to which each attribute preserves the interpoint distances. In other words, for each attribute, \( x \), in \( D \), it computes \( g_{m} \) and \( g_{M} \) given by:

\[
g_{m} = \min \left\{ \frac{|| f(u) - f(v) ||^2}{|| u - v ||^2} \right\}
\]

\[
g_{M} = \max \left\{ \frac{|| f(u) - f(v) ||^2}{|| u - v ||^2} \right\}
\]

where \( u \) and \( v \) are any two rows of \( D \), and \( f(u) \) and \( f(v) \) are the corresponding rows in the dataset reduced to the single attribute \( x \). The average distance preservation for the attribute \( x \) is then computed as:

\[
g_{\text{mid}} = \frac{(g_{m} + g_{M})}{2}
\]

To reduce the dataset \( D \) from \( p \) columns to \( k \) columns, this approach then finds the combination of \( k \) attributes whose average value of \( g_{\text{mid}} \) is maximum.

2.5 The Direct Approach [4]

As with the Combined Approach, to reduce a dataset \( D_{\text{exp}} \) to a dataset containing \( k \) columns, the Direct Approach selects the combination of \( k \) attributes which best preserve the interpoint distances, and reduces the original dataset to a dataset containing only those \( k \) attributes. To do so, it first generates all possible combinations of \( k \) attributes from the original \( p \) attributes. Then, for each combination, \( C \), it computes \( g_{m} \) and \( g_{M} \) given by:

\[
g_{m} = \min \left\{ \frac{|| f(u) - f(v) ||^2}{|| u - v ||^2} \right\}
\]

\[
g_{M} = \max \left\{ \frac{|| f(u) - f(v) ||^2}{|| u - v ||^2} \right\}
\]

where \( u \) and \( v \) are any two rows of \( D \), and \( f(u) \) and \( f(v) \) are the corresponding rows in the dataset reduced to the attributes in \( C \). The average distance preservation for this combination of attributes is then computed as:

\[
g_{\text{mid}} = \frac{(g_{m} + g_{M})}{2}
\]

To reduce the dataset \( D \) from \( p \) attributes to \( k \) attributes, this approach then finds the combination of \( k \) attributes whose value of \( g_{\text{mid}} \) is maximum.

2.6 The New Random Approach

This is a technique suggested by Nsang, Maikori, Oguntoyinbo and Yusuf in [6]. With this technique, to reduce a data set \( D \) of dimensionality \( d \) to one of dimensionality \( k \), a set \( S_{k} \) is formed consisting of \( k \) numbers selected at random from the set \( S \) given by:

\[
S = \{ x \in N \mid 1 \leq x \leq d \}
\]

Then, our reduced set, \( D_{\text{R}} \), will be given by:

\[
D_{\text{R}} = D(:, S_{k})
\]

That is, \( D_{\text{R}} \) is a data set having the same number of rows as \( D \) and if \( A_{i} \) is the \( i^{th} \) attribute of \( D_{\text{R}} \), then \( A_{i} \) is the \( i^{th} \) attribute of \( D \) if \( j \) is the \( i^{th} \) element of \( S_{k} \).

2.7 The Modified New Random Approach [7]

This technique is suggested as an improvement of the New Random Approach discussed in Section 2.6 above. To reduce a dataset \( D_{\text{exp}} \) from \( p \) attributes to \( k \) attributes using the modified version of the New Random Approach, we use the algorithm below. Clearly, the idea here is to generate a result which is less random (and thus more efficient) than the result of the New Random Approach. Note that \( m \) in the algorithm is the number of times the execution of the New Random Approach is repeated.
Algorithm

M = []
for i = 1 to m do
  - Run the New Random Approach to generate k numbers at random in the range 1..p
  - Store the list of numbers generated as the i-th row of M
end

Generate the one-dimensional matrix M1 with p entries such that M1[p] holds the frequency of the number p in the matrix M
Finally, generate the matrix Result which contains the k entries in M of highest frequency, arranged in ascending order

Thus, if D is the original dataset, the result of reducing D using the modified version of the New Random Approach is given by:

\[ D_R = D(:, \text{Result}) \]

2.8 First Novel Approach [1]

To reduce a dataset \( D_{n \times p} \) to a dataset containing k columns, it first determines the approximate extent to which each attribute preserves the interpoint distances by computing \( g_{x_{\text{mid}}} \) for each attribute \( x \).

For a reduction of a data set \( D \) from \( p \) attributes to \( k \) attributes, the \( k \) attributes of \( D \) with largest \( g_{x_{\text{mid}}} \) value are selected.

2.9 Second Novel Approach [1]

Like the first novel approach, this approach first determines the extent to which each attribute preserves the interpoint distances. This time, however, the actual distance preservation for each attribute is computed.

If \( x \) is an attribute of the dataset \( D \), the actual distance preservation of \( x \) is computed as:

\[
\text{adp}_x = \sum_{u=1}^{n} \sum_{v=1}^{n} \left( \frac{\| f(u) - f(v) \|^2}{\| u - v \|^2} \right) / n_r
\]

where \( n \) is the number of rows of \( D \), \( u \) and \( v \) are any two rows of \( D \), and \( f(u) \) and \( f(v) \) are the corresponding rows in the dataset reduced to the single attribute \( x \).

The term \( n_r \) in this equation is the number of pairs of rows of \( D \) computed as:

\[
n_r = C_r = n(n-1)/2
\]

To reduce the dataset \( D \) from \( p \) columns to \( k \) columns, it selects the \( k \) attributes of \( D \) with largest \( \text{adp}_x \) value.

2.10 Third Novel Approach [1]

To reduce a dataset \( D_{n \times p} \) from \( p \) attributes to \( k \) attributes, the Third Novel Approach first computes the extent to which each attribute of \( D \) preserves \( k\)-means clustering. It then selects the \( k \) attributes which best preserve \( k\)-means clustering.

3. Relative Effective Performance of Each Approach

The dataset students.txt (Table 1) was reduced using the ten dimensionality reduction techniques described in the last section, and the relative effective performances of these approaches were compared using the following four criteria: classification preservation, variance preservation, interpoint distance preservation and total run time. The results obtained are shown in Tables 3 and 4 below.

Table 1: Contains the scores of 15 students recorded in the file Students.txt. The last but one column contains the average score of each student, and the last column classifies each student into one of eight classes following the criteria in Table 2.

<table>
<thead>
<tr>
<th>Range</th>
<th>Classification</th>
</tr>
</thead>
<tbody>
<tr>
<td>0 - 29</td>
<td>Very Dull</td>
</tr>
<tr>
<td>30 - 39</td>
<td>Dull</td>
</tr>
<tr>
<td>40 - 49</td>
<td>Below Average</td>
</tr>
<tr>
<td>50 - 54</td>
<td>Average</td>
</tr>
<tr>
<td>55 - 59</td>
<td>Above Average</td>
</tr>
<tr>
<td>60 - 74</td>
<td>Bright</td>
</tr>
<tr>
<td>75 - 84</td>
<td>Very Bright</td>
</tr>
<tr>
<td>85 - 100</td>
<td>Genius</td>
</tr>
</tbody>
</table>

Table 2: Classification of the 15 students in Table 1
Table 3: Results obtained when the Dataset Students.txt (Table 1) was reduced from 15 attributes to 12 attributes using ten Dimensionality Reduction Techniques, and the Relative Effective Performances of these techniques compared using the four criteria specified in the table.

<table>
<thead>
<tr>
<th>DR Technique</th>
<th>Classification Preservation</th>
<th>Interpoint Distance Preservation</th>
<th>Variance Preservation</th>
<th>Time Taken</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. RP</td>
<td>93.13%</td>
<td>93.44%</td>
<td>93.61%</td>
<td>0.003</td>
</tr>
<tr>
<td>2. PCA</td>
<td>93.32%</td>
<td>93.14%</td>
<td>93.34%</td>
<td>0.004</td>
</tr>
<tr>
<td>3. Var</td>
<td>93.32%</td>
<td>87.45%</td>
<td>93.45%</td>
<td>0.036</td>
</tr>
<tr>
<td>4. Comb</td>
<td>93.33%</td>
<td>73.85%</td>
<td>93.34%</td>
<td>0.003</td>
</tr>
<tr>
<td>5. Dir</td>
<td>93.33%</td>
<td>69.25%</td>
<td>93.34%</td>
<td>0.063</td>
</tr>
<tr>
<td>6. NRA (v1)</td>
<td>93.53%</td>
<td>68.96%</td>
<td>93.61%</td>
<td>0.004</td>
</tr>
<tr>
<td>7. NRA (v2)</td>
<td>93.53%</td>
<td>68.96%</td>
<td>93.61%</td>
<td>0.004</td>
</tr>
<tr>
<td>8. Nor App 1</td>
<td>93.53%</td>
<td>68.96%</td>
<td>93.61%</td>
<td>0.004</td>
</tr>
<tr>
<td>9. Nor App 2</td>
<td>93.53%</td>
<td>68.96%</td>
<td>93.61%</td>
<td>0.004</td>
</tr>
<tr>
<td>10. Nor App 3</td>
<td>93.53%</td>
<td>68.96%</td>
<td>93.61%</td>
<td>0.004</td>
</tr>
</tbody>
</table>

Remark:
The extent to which the classification of the original dataset was preserved by each dimensionality reduction technique was computed using the rand index [4].

The above results show that:

- The Combined and Direct Approaches have the worst performance in terms of computational runtime. All the other approaches have very low runtime complexities.
- The three random approaches (RP, NRA (version 1) and NRA (version 2)) have the best performance in terms of computational runtime.
- The Combined and Direct Approaches are the best in preserving the classification of the students in original dataset, while RP and PCA have the worst performance in this regard. All the other approaches are almost as good as each other in preserving the classification of the students in original dataset.

- PCA is the best approach in preserving the interpoint distances, and RP is the worst. All the other approaches are almost as good as each other in preserving the interpoint distances.
- Also, PCA is the best approach in preserving the variances between the entries in the original dataset, and RP is the worst. All the other approaches are almost as good as each other in preserving variances between the entries in the original dataset.

4. Clustering

Clustering is the assignment of a set of observations into subsets (called clusters) so that observations in the same cluster are similar in some sense. Usually, similarity is measured by considering the distance between two observations. Thus two observations are “similar” if the distance between them is very small.

In a cluster, the distance between neighboring points is typically smaller than the distance between points of different clusters. Therefore the choice of a suitable distance metric for points (that is, for objects to be grouped) and for clusters is of fundamental importance.

Various distance metrics are defined for the distance $d$ between two vectors $x$ and $y$ in $\mathbb{R}^n$. They include:

i) Euclidean distance:
$$d_e(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

ii) Sum of squared distances:
$$d_s(x, y) = \sum_{i=1}^{n} (x_i - y_i)^2$$

iii) Manhattan distance:
$$d_m(x, y) = \sum_{i=1}^{n} |x_i - y_i|$$

iv) Distance of the maximum component:
$$d_m(x, y) = \max_{i=1,2,...,n} |x_i - y_i|$$

v) Inverse of normalized projection of the vectors $x$ and $y$ on each other:
$$d_i(x, y) = \frac{\|x\| \|y\|}{x \cdot y}$$
4.1. Hierarchical Clustering [1]

In hierarchical clustering we begin with \( n \) clusters consisting of one point each.

Then the nearest neighbor clusters are combined until all points have been combined into a single cluster, or until a termination criterion has been reached.

**Algorithm:**

The process above translates into the following algorithm:

\[
\text{HierarchicalClustering}(x_1, \ldots, x_k) \\
\text{initialize } C_1 = \{x_1, \ldots, x_k\} = \{x_k\} \\
\text{Repeat} \\
\quad \text{Find two clusters } C_i \text{ and } C_j \text{ with the smallest distance} \\
\quad \text{Combine } C_i \text{ and } C_j \\
\text{Until Termination condition reached} \\
\text{Return} \{\text{new with clusters}\}
\]

The termination condition could be chosen as, for example, a desired number of clusters or a minimum distance between clusters.

It is so far unclear how the distances between the clusters are calculated. A convenient and often used metric is the distance between the two closest points in the two clusters \( C_i \) and \( C_j \):

\[
d_{\min}(C_i, C_j) = \min_{x \in C_i, y \in C_j} d(x, y)
\]

where \( d(x, y) \) is computed using one of the distance metrics above.

4.2 Preservation of Hierarchical Clustering

We compare the ten dimensionality reduction techniques described in this paper for hierarchical clustering preservation using the \textit{rand} index. In other words, for each dimensionality reduction technique, we would like to determine the extent to which the clustering results of a dataset is maintained when the dataset is reduced by this technique and the hierarchical clustering algorithm applied to the reduced dataset, and make comparisons with each other.

To do so, we reduce three datasets \textit{Weather}, \textit{Student} and \textit{Inosphere} datasets (obtained from MATLAB) using each of these ten techniques to varying numbers of attributes. The results obtained are shown in Tables 5, 6 and 7 below.
These results show that:

➢ All ten reduction techniques are very efficient in preserving the hierarchical clustering of the original datasets
➢ However, on the average, RP performs slightly worse than all the other approaches, while PCA has the best performance.

5. Conclusion

In this paper, we implemented ten dimensionality reduction techniques, which include the three novel approaches proposed by Baba, Adeseye and Nsang in [1]. Then we investigated the relative effective performance of each approach with respect to classification preservation, interpoint distance preservation, variance preservation and time taken to reduce the datasets.

We also discussed the hierarchical clustering algorithm, and compared the ten approaches implemented in this paper by the extent to which they preserve hierarchical clustering. We realized that all ten techniques were very efficient in preserving the hierarchical clustering of the original datasets; and that RP performs slightly worse than the others, while PCA has the best performance.

References


