LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-TR-531131

On the selection of dimension reduction techniques for scientific applications

Y. J. Fan, C. Kamath

February 22, 2012

Disclaimer

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

On the Selection of Dimension Reduction Techniques for Scientific Applications

Ya Ju Fan and Chandrika Kamath

Center for Applied Scientific Computing, Lawrence Livermore National Laboratory, Livermore, CA
{fan4,kamath2}@llnl.gov

Abstract. Many dimension reduction methods have been proposed to discover the intrinsic, lower dimensional structure of a high-dimensional dataset. However, determining critical features in datasets that consist of a large number of features is still a challenge. In this paper, through a series of carefully designed experiments on real-world datasets, we investigate the performance of different dimension reduction techniques, ranging from feature subset selection to methods that transform the features into a lower dimensional space. We also discuss methods that calculate the intrinsic dimensionality of a dataset in order to understand the reduced dimension. Using several evaluation strategies, we show how these different methods can provide useful insights into the data. These comparisons enable us to provide guidance to a user on the selection of a technique for their dataset.

1 Background and Motivation

It is a challenge to understand, interpret, and analyze high-dimensional data, where each example or instance is described by many features. Often, only a few features are important to the analysis task, or the data naturally lie on a lower-dimensional manifold. To reduce the dimension of the dataset, we can either identify a subset of features as important using techniques such as filters [1] and wrappers [2]. Or, we can transform the data into a reduced dimensional representation while preserving meaningful structures in the data. These methods include linear projections, such as principal component analysis (PCA) [3], as well as several non-linear methods that have been proposed recently [4].

To fully benefit from this wealth of dimension reduction techniques, we need to understand their strengths and weaknesses better so we can determine a method appropriate for a dataset and task, select the parameters for the method suitably, and interpret the results correctly to provide insights into the data. Some techniques, such as PCA, filters, and wrappers, have been studied extensively and applied to real problems. Others, such as the recent non-linear dimension reduction (NLDR) techniques, have been explained, and their benefits demonstrated, through the use of synthetic datasets, such as the three-dimensional Swiss roll data. While the simplicity of these data sets is useful in

visually explaining the techniques, it is unclear how they perform in real problems where the dimensionality is too high for visualization. Additional guidance is needed to ascertain if these newer techniques are more appropriate than other approaches in their ability to represent the data in the lower-dimensional space, their computational cost, and the interpretability of the results.

In this paper, we present a series of carefully designed experiments with real datasets to gain insights into the different dimension reduction methods. We consider data from three science domains: astronomy, wind power generation, and remote sensing, where these techniques are used to identify features important to the phenomenon being observed, to build more accurate predictive models, to reduce the number of features that need to be measured, and to reduce the number of samples required to explore the feature space of a problem.

To provide guidance to a practitioner, we focus on three aspects of the task of dimension reduction. First, we evaluate the techniques using datasets with properties that arise commonly in practice, such as data with noise features, with labeling based on different criterion, or with very high dimensionality. These data may also have other unknown properties, such as inherent lower dimensional manifolds. Second, we consider the task of setting the dimensionality of the lower dimensional space. This important issue is rarely discussed in the context of real datasets whose high dimensionality prevents visualization to understand their properties. And finally, we consider ways in which we might interpret the results obtained using the different methods.

This paper is organized as follows: we start by briefly describing data transformation methods and feature subset selection techniques in Sections 2 and 3, respectively . Next, in Section 4, we discuss how we can obtain the intrinsic dimensionality of the data by exploiting the information provided by these methods. In Section 5, we describe the scientific problems of interest, followed by our evaluation methodology for the dimension reduction techniques in Section 6. The experimental results are discussed in Section 7. In Section 8 we describe related work and conclude in Section 9 by summarizing our guidance for practitioners.

The notation used in this paper is as follows: $X \in \mathbb{R}^{n \times D}$ represents the dataset in the high-dimensional space, that is, X consists of n data points, X_i, each of length D, the dimension of the data. We want to reduce the dimension of these points resulting in the dataset, $Y \in \mathbb{R}^{n \times d}$, where $d < D$.

2 Dimension Reduction Using Transformation

We next briefly describe the transform-based techniques, including PCA and four popular NLDR techniques: Isomap, Locally Linear Embedding (LLE), Laplacian Eigenmaps, and Local Tangent Space Alignment (LTSA) [5, 6]. These methods share the use of an eigendecomposition to obtain a lower-dimensional embedding of the data that is guaranteed to provide global optimality.

Principal Component Analysis (PCA): PCA [3] is a linear technique that preserves the largest variance in the data while decorrelating the transformed

dataset. An eigenvalue problem to the data covariance matrix, C, is formulated as $CM = \lambda M$. The eigenvectors, M, corresponding to the significant eigenvalues, λ, form a basis for linear transformation that optimally maximizes the variance in the data. The low-dimensional representation is expressed by $Y = XM$ and the eigenvalues can be used to determine the lower dimensionality, d.

PCA does not require any parameter to be set. It has a computational cost of $O(D^3)$ and requires $O(D^2)$ memory.

Isomap: The Isomap method [7] preserves pairwise geodesic distances between data points. It starts by constructing an adjacency graph based on the neighbors of each point in the input space. These neighbors can be either the k-nearest neighbors or points which lie within an ϵ-neighborhood. Next, the geodesic distances [8, 9] between all pairs of points are estimated by computing their shortest path distances over the graph. Let $D_G = \{d_G(i,j)\}_{i,j=1,\ldots,n}$ be the matrix of geodesic distances, where $d_G(i,j)$ is the distance between point i and j. Isomap then constructs an embedding in a d-dimensional Euclidean space such that the pair-wise Euclidean distances between points in this space approximate the geodesic distances in the input space. Let $D_Y = \{d_Y(i,j)\}_{i,j=1,\ldots,n}$ be the Euclidean distance matrix and $d_Y(i,j) = \|Y_i - Y_j\|_2$. The goal is to minimize the cost function $\|\tau(D_G) - \tau(D_Y)\|_2$, where the function τ performs double centering on the matrix to support efficient optimization. The optimal solution is found by solving the eigen-decomposition of $\tau(D_G)$. The Y coordinates are then computed based on the d largest eigenvalues and their corresponding eigenvectors.

Isomap requires one parameter k (or ϵ), has a computational cost of $O(n^3)$ and requires $O(n^2)$ memory.

Locally Linear Embedding (LLE): The LLE method [10] preserves the reconstruction weights ω_{ij} that are used to describe a data point X_i as a linear combination of its neighbors $X_j, j \in \mathcal{N}(i)$, where $\mathcal{N}(i)$ is the set of neighbors of point i. The optimal weights for each i are obtained by minimizing the cost function, $\min_{\omega} \{\|X_i - \sum_{j\in\mathcal{N}(i)} \omega_{ij} X_j\|^2 \Big| \sum_{j\in\mathcal{N}(i)} \omega_{ij} = 1\}$. LLE assumes that the manifold is locally linear and hence the reconstruction weights are invariant in the low-dimensional space. The embedding Y of LLE is obtained from the eigenvectors corresponding to the smallest d nonzero eigenvalues of the embedding matrix, defined as $M = (I - W)^{\mathsf{T}}(I - W)$, where W is the reconstruction weight matrix with elements $W_{ij} = 0$ if $j \notin \mathcal{N}(i)$; $W_{ij} = \omega_{ij}$ otherwise; and I is an identity matrix.

LLE requires one parameter k (or ϵ), has a computational cost of $O(pn^2)$ and requires $O(pn^2)$ memory, where p is the fraction of non-zero elements in the sparse matrix.

Laplacian Eigenmaps: This method provides a low-dimensional representation in which the weighted distances between a data point and other points within an ϵ-neighborhood (or k-nearest neighbors) are minimized [11]. The distances to the neighbors are weighted using the Laplacian operator $W_{ij} = e^{-\frac{\|X_i - X_j\|^2}{t}}$.

Here, $t = 2\sigma^2$, where σ is the standard deviation of the Gaussian kernel. The representation of Y is computed by solving the generalized eigenvector problem: $(S - W)v = \lambda Sv$, where $S_{ii} = \sum_j W_{ij}$. Only the eigenvectors (v) corresponding to the smallest nonzero eigenvalues (λ) are used for the embedding.

Laplacian Eigenmaps requires two parameters k (or ϵ) and t, has a computational cost of $O(pn^2)$ and requires $O(pn^2)$ memory.

Local Tangent Space Alignment (LTSA): The LTSA method [12] applies PCA on the neighborhood of each data point, forming a local tangent space that represents the local geometry. Those local tangent spaces are then aligned to construct the global coordinate system of the underlying manifold. LTSA requires one parameter k and the determination of d before applying the method. It has a computational cost of $O(pn^2)$ and requires $O(pn^2)$ memory.

3 Dimension Reduction Using Feature Subset Selection

We consider four methods which are applicable when the dataset is labeled.

Stump Filter: A stump is a decision tree with only the root node; the stump filter ranks features using the same process as the one used to create the root node. Decision trees split the data by examining each feature and finding the split that optimizes an impurity measure. To search for the optimal split for a numeric feature x, the feature values are sorted ($x_1 < x_2 < ... < x_n$) and all mid-points $(x_i + x_{i+1})/2$ are evaluated as possible splits using a given impurity measure. The features are then ranked according to their optimal impurity measures. In our work, we use the Gini index [13] as a measure of the impurity.

Distance Filter This filter calculates the class separability of each feature using the Kullback-Leibler (KL) distance between histograms of feature values. For each feature, there is one histogram for each class. In a two class problem, if a feature has a large distance between the histograms for the two classes, then it is likely to be an important feature in differentiating between the classes. We discretized numeric features using $\sqrt{|n|}/2$ equally-spaced bins, where $|n|$ is the size of the data. Let $p_j(d = i|c = a)$ be an estimate of the probability that the j-th feature takes a value in the i-th bin of the histogram given a class a. For each feature j, we calculate the class separability as $\Delta_j = \sum_{a=1}^{c} \sum_{b=1}^{c} \delta_j(a, b)$, where c is the number of classes and $\delta_j(a, b)$ is the KL distance between histograms corresponding to classes a and b: $\delta_j(a, b) = \sum_{i=1}^{B} p_j(d = i|c = a) \log\left(\frac{p_j(d=i|c=a)}{p_j(d=i|c=b)}\right)$, where B is the number of bins in the histograms. The features are ranked simply by sorting them in descending order of the distances Δ_j (larger distances mean better separability).

Chi-squared Filter: The Chi-squared filter computes the Chi-square statistics from contingency tables for every feature. The contingency tables have one row for every class label and the columns correspond to possible values of the feature

Table 1. A 2×3 contingency table, with observed and expected frequencies (in parenthesis) of a fictitious feature f1 that takes on 3 possible values (=1, 2, and 3).

Class	f1=1	f1=2	f1=3	Total
0	31 (22.5)	20 (21)	11 (18.5)	62
1	14 (22.5)	22 (21)	26 (18.5)	62
Total	45	42	37	124

(see Table 1, adapted from [14]). Numeric features are represented by histograms, so the columns of the contingency table are the histogram bins. The Chi-square statistic for feature j is $\chi^2_j = \sum_i \frac{(o_i - e_i)^2}{e_i}$, where the sum is over all the cells in the $r \times c$ contingency table, where r is the number of rows and c is the number of columns; o_i stands for the observed value (the count of the items corresponding to the cell i in the contingency table); and e_i is the expected frequency of items calculated as: $e_i = \frac{(\text{column total}) \times (\text{row total})}{\text{grand total}}$. The variables are ranked by sorting them in descending order of their χ^2 statistics.

ReliefF: ReliefF [15] estimates the quality of features by calculating how well they distinguish between instances close to each other. It starts by taking an instance i at random and identifies its nearest k hits (H_i) and misses (M_i), which are the closest instances of the same and different classes, respectively. Then, it obtains the quality estimate of a feature s, which for a two-class dataset is defined as: $Q_s = \sum_{i=1}^{n} \left\{ \sum_{m \in M_i} \frac{\|X_{is} - X_{ms}\|}{nk} - \sum_{h \in H_i} \frac{\|X_{is} - X_{hs}\|}{nk} \right\}$ where X_{is} is the value of feature s for instance i. By increasing the quality estimate when the selected point and its misses have different values of feature s, and decreasing it when the point and its hits have different values of the feature, ReliefF ranks the features based on their ability to distinguish between instances of the same and different classes.

4 Determining the Intrinsic Dimensionality of the Data

An important issue in dimension reduction is the choice of the number of dimensions for the low-dimensional solution. The intrinsic dimension of the data is the minimum number of variables necessary to represent the observed properties of the data. While many algorithms require the intrinsic dimensionality of the embedding be explicitly set, only a few provide an estimate of this number.

In feature subset selection methods, we can easily identify the number of features to select by considering the metric used to order the features and disregarding features ranked lower than a certain threshold value. Or, we can include a noise feature and disregard any features ranked lower than this noise feature.

In the case of PCA, an adequate number of principal components is identified by ordering the eigenvalues and selecting the top d significant principal components, with the remainder describes the reconstruction error: $E_d = \sum_{j=d+1}^{D} \lambda_j$

[16]. Many selection critera have been developed based on the magnitude of eigenvalues. In our work, we use the number of eigenvalues that exceed a fixed percentage of the largest eigenvalue [17]. For example, we use d^{10} to indicate the number of eigenvalues that exceed 10 percent of the largest eigenvalue.

For nonlinear methods, the use of the eigen-spectrum only works when the data lie on a linear manifold [18]; so, we need to consider other methods. One such approach applicable to Isomap and LLE is based on the *elbow test* using a lack-of-fit measure. We first determine the property that the NLDR technique is trying to preserve. The deviation between the property in the low-dimensional space and the input space is plot against the dimensionality and the intrinsic dimension is chosen at the "elbow" in the plot where after a certain number of dimensions, the lack-of-fit value is not reduced substantially. For Isomap the lack-of-fit measure is the residual variances of the two geodesic distance matrices evaluated in the representation space and in the input space. For LLE, we use the reconstruction error. The reconstruction weights are updated using the embedding vectors Y_i and then applied to the input data X_i. The intrinsic dimensionality d can be estimated by the values of reciprocal cost function [10], defined as $f(W^{(d)}) = \sum_i \|X_i - \sum_j W_{ij}^{(d)} X_j\|^2$, where $W^{(d)}$ is the reconstruction weight matrix computed using the d-dimensional representation vectors Y_i.

An alternate approach is to determine the locally linear scale using simple box counting. Let $C(r)$ indicate the number of data points that are contained within a ball of radius r centered on a data point. If the data are sampled over a d-dimensional manifold, then $C(r)$ is proportional to r^d for small r. The intrinsic dimensionality at the locally linear scale is $d = \frac{\partial \ln C(r)}{\partial \ln r}$. Since datasets have finite samples in practice, we can obtain the estimate by plotting $\ln C(r)$ versus $\ln r$ and measuring the slope of the linear part of the curve [19].

And finally, we can use the statistical estimation of intrinsic dimensionality [20], which is based on the assumption that the topological hypersurface in a local region can be approximated by a linear hypersurface of the same dimensionality. We start by calculating the distances between all points. Then, for each point i, we find the closest neighbor j_0; the vector connecting i to j_0 forms a subspace of dimension one. We then consider the next closest neighbor j_1 to i, and consider the angle between the vector connecting i and j_1 and the subspace. These vectors connecting i and its l closest neighbors form an l-dimensional space. We continue increasing the size of l until, for a certain dimension, d, the mean of the angles taken over all points is less than a threshold.

5 Datasets Used in the Evaluation

We evaluate the dimension reduction techniques using classification problems in three science domains: astronomy, wind energy, and remote sensing.

5.1 Astronomy Dataset

This dataset is used to build a model to classify radio-emitting galaxies into two classes - one with a bent-double morphology (called 'bents') and the other

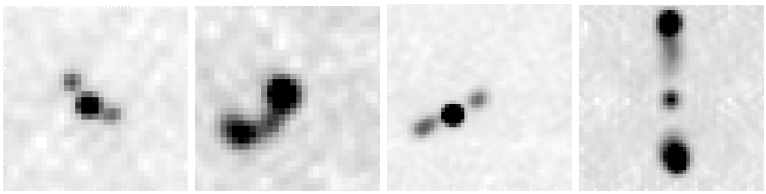


Fig. 1. Examples of bent-double (left two) and non-bent double (right two) radio-emitting galaxies.

without (called 'non-bents') (Figure 1). These data are from the Faint Images of the Radio Sky at Twenty-cm (FIRST) survey [21]. The astronomers first processed the raw image data to create a 'catalog' by fitting two-dimensional, elliptic Gaussians to each galaxy. Each entry in the catalog corresponds to a Gaussian and includes information such as the location and size of the Gaussian, the major and minor axes of the ellipse, and the peak flux. This catalog was then processed to group nearby Gaussians into galaxies and extract features, such as angles and distances, that represented each galaxy. The focus was on galaxies composed of three Gaussians and the features included those obtained by considering each Gaussian individually, considering the Gaussians taken two at a time, and considering all three Gaussians.

This dataset, which we refer to as the *First* dataset, is quite small, consisting of 195 examples, with 167 bents and 28 non-bents, each described by 99 features, of which 9 are non-numeric. In addition, we also consider a derived dataset, which we refer to as *FirstTriples*, containing only the 20 numeric features for all three Gaussians. The astronomers thought this subset to be a better representation of the bent galaxies.

5.2 Wind Energy Dataset

Our next application area is wind power generation. The task is one of using the weather conditions provided by meteorological towers in the region of the wind farms to classify days which will have ramp events. A ramp event occurs when the wind power generation suddenly increases or decreases by a large amount in a short time (Figure 2). These events make it difficult for the control room operators to schedule wind energy on the power grid. If we can use the weather conditions to predict if a day will have a ramp event, the grid operators can be better prepared to keep the grid balanced in the presence of these events.

In this dataset, we have 731 examples representing the data for the days in 2007-2008. The features are the daily averages of different variables, such as wind speed, wind direction, and temperature, at three meteorological towers in the Tehachapi Pass region. Each tower provides 7 features, for a total of 21 features. Each day is assigned a binary class variable, indicating if a ramp event exceeding a certain magnitude occurred in any 1 hour interval during that day. There are two datasets, *Wind115* and *Wind150*, which correspond to ramps with magnitudes exceeding 115 MW and 150 MW, respectively. That is, in the

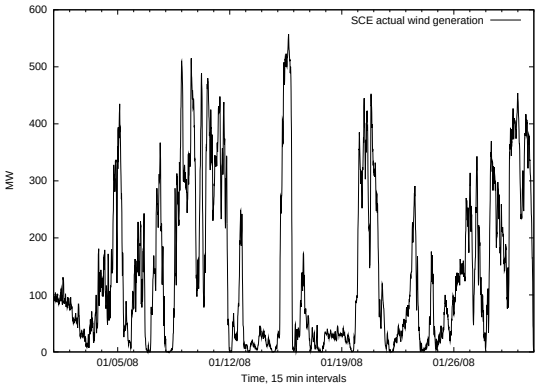


Fig. 2. Wind power generation from the wind farms in the Tehachapi Pass region in Southern California for January 2008.

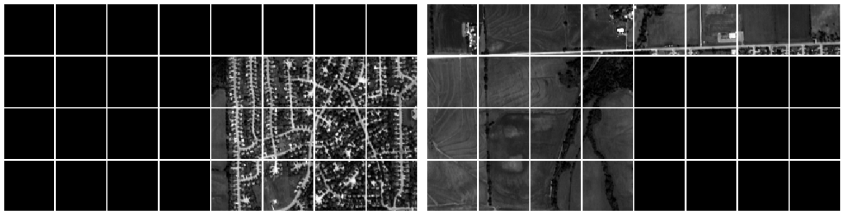


Fig. 3. Example of a region in satellite imagery illustrating the ground truth with inhabited tiles (left) and uninhabited tiles (right). Original satellite image by GeoEye (formerly Space Imaging).

Wind115 dataset, a day is assigned a label of 1 if during any one hour interval, the wind power generation increased or decreased by more than 115 MW.

5.3 Remote Sensing Dataset

Our third application area is remote sensing, where the task is to classify tiles in satellite images of the earth as being inhabited or uninhabited (Figure 3; data from the IKONOS satellite (www.geoeye.com)). The data are available as 4-band multi-spectral (near-infrared, red, green, and blue) images at 4 meter ground sample distance. An image is divided into non-overlapping tiles of size 64×64 pixels. Each tile is represented by several texture features as the domain experts believed that texture could indicate man-made structures, such as houses or parking lots where there is certain regular structure that can be represented as a 'texture'. However, as they were not sure which texture feature was the most appropriate, they extracted several, including the Grey Level Co-occurrence Matrix (GLCM), the power spectrum texture features, the wavelet texture features, and the Gabor texture features [22–24]. Further, as it was not clear which of the 4-bands had the most relevant information, the domain experts extracted the texture features for each band and concatenated them, resulting in a long feature vector of 496 features (124 from each band), representing a tile. Since this

dataset was very large, both in the number of examples and in the number of features, we created a smaller subset, which we refer to as *RemoteSmall*, for use in our experiments. This subset contains all the features from the *Remote* dataset, but has only 2000 examples, distributed equally among inhabited and uninhabited tiles.

6 Evaluation Methodology

We evaluate the effectiveness of the dimension reduction methods using the classification accuracy of the transformed or selected features relative to the accuracy using all the original features. In our work, we consider decision tree classifiers as their results, being easily interpreted, can be explained to domain scientists. Also, decision tree classifiers utilize the order of the significance of features [25], making them suitable for our use as the features in the lower dimensional space are ordered using either the magnitude of eigenvalues or a metric that determines the discriminating ability of a feature. We could have also used other classifiers such as support vector machines or neural networks, but their results are not as easily interpreted. We could have also used sparse methods which incorporate feature selection [26, 27], but they are more suitable for regression problems.

In our work, we used the ensemble approach proposed in [28] as it gives more accurate results than bagging or boosting. This approach creates ensembles by introducing randomization at each node of the tree in two ways. It first randomly samples the examples at a node and selects a fraction (we use 0.7) for further consideration. Then, for each feature, instead of sorting these examples based on the values of the feature, it creates a histogram, evaluates the splitting criterion (we use Gini [13]) at the mid-point of each bin of the histogram, identifies the best bin, and then selects the split point randomly in this bin. The randomization is introduced both in the sampling and in the choice of the split point. The use of the histograms speeds up the creation of each tree in the ensemble. We use 10 trees in the ensemble. Using the first d transformed features, we report the percentage error rate obtained for five-fold cross validation repeated five times and evaluate how this error rate changes as the number of features is increased.

We observe that our use of a classifier for evaluation may favor the feature selection methods as they exploit the class label in the ordering of features by importance. In contrast, data transformation methods are unsupervised, trying to find hidden structure of the data without knowledge of the class labels. Consequently, the feature selection methods may have an advantage with our evaluation criterion, though we expect that in comparison to the error rate using all original features, the transform based methods should provide an improvement.

In addition to classification accuracy, we also evaluate the dimension reduction methods on the insights they can provide into the data. The major advantage of feature subset selection is that the methods identify the important original features, which can be used to understand scientific phenomenon.

In contrast, for the data tranformation methods, it is not easy to explain what forms the features in the new space. In the case of PCA, since it transforms the original data using linear combinations of the top d eigenvectors, we can consider the values of elements of the eigenvectors for insights. The absolute values of elements in the eigenvector weigh the importance of the original features for the corresponding principal component, while the sign of the elements indicates the correlation among the features. We use a biplot [29] to interpret PCA results, although it is limited to top two or three features on the plot.

For the nonlinear transformation methods, the reduced dimension has been explained in the case of datasets such as visual perception, movement and handwriting [7]. The data points are displayed as images that are interpolations along straight lines in the representing coordinate space. This task becomes impossible for scientific data sets that consists of a large number of features extracted from low-level data and are not necessarily images. Hence, we are limited to evaluating the linear correlation between the projected dimensions and the original features to gain insights into the data.

7 Experimental Results and Discussion

We next present the experimental results for the four feature selection methods and the five transform methods on the datasets from the three problem domains. For the low-dimensional data representations using the four NLDR techniques, we experimented with several parameter settings. Isomap, LLE and Laplacian Eigenmaps have a parameter k or ϵ, depending on whether we consider the k-nearest neighbors or an ϵ-neighborhood. Laplacian Eigenmaps has an additional parameter t used in the Gaussian kernel. LTSA has only a parameter k, but requires a determination of d before applying the method. We tested $k = 3, 5, 7, \ldots, 29$, ϵ that ranges from 1.2 to 20.0, and $t = 1, 5$ and 10. We then obtained the percentage error rates for the decision tree ensemble classifier as outlined earlier in Section 6. The same approach was used for the four feature subset selection methods, where we obtained the percentage error rate using the first d features. In the classification error rate plots presented in the rest of this section, we include the best results for the four NLDR methods, the results for PCA and the four feature subset selection methods, and the error rate for the decision tree ensemble applied to the whole, original input data, which is displayed as a constant horizontal line on the plots.

Table 2 summarizes the intrinsic dimension estimation using eigenvalue spectrum of PCA on all datasets. They are discussed together with the results of all other intrinsic dimensionality estimations in the following.

7.1 Experiments on *First* and *FirstTriples* Datasets

Figure 4 presents the classification accuracy of dimension reduction methods applied on the *First* dataset. We observe that using the reduced representations from the five data transformation techniques is not guaranteed to provide better

Table 2. Intrinsic dimension using PCA.

Dataset	d^{10}	d^5	d^1
First	21	26	36
FirstTriples	9	12	13
Wind115 & Wind150	5	8	15
RemoteSmall	2	4	11

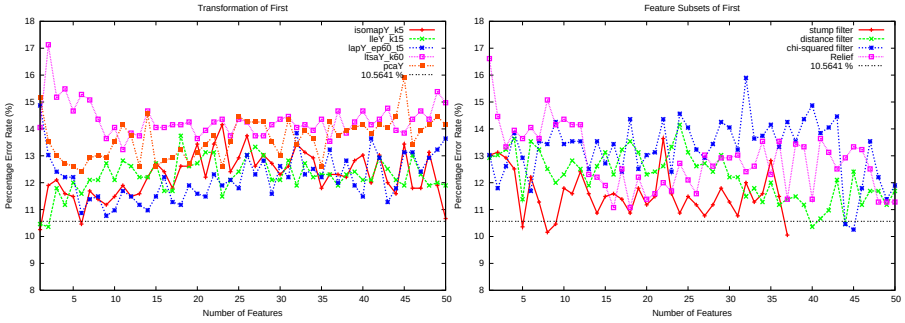
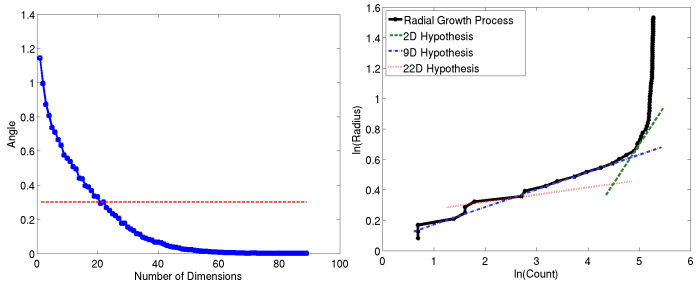


Fig. 4. Classification error rates using decision tree classifiers on the transformed features (left) and the selected features (right) for the *First* dataset.

classification performance than using the original input data. Only the representation of Isomap with $k = 5$ and LLE with $k = 15$ gives a smaller error rate than the original input data when using the first few features. On the other hand, in the results with the feature subset selection methods, Relief fails at selecting useful subsets for *First* dataset, while the other three methods give error rates below the horizontal line, indicating an improvement over using all original features.

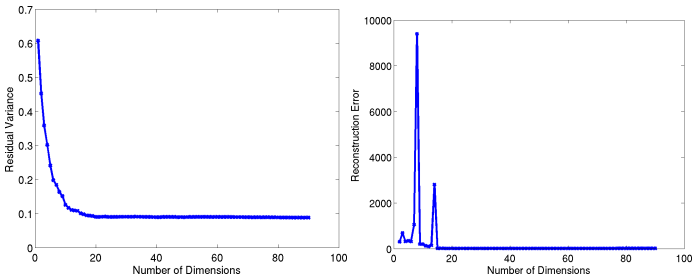
Figure 5 shows the intrinsic dimensionality of the *First* dataset estimated by the four different methods. This dataset contains 90 features and there is some variation in the estimates. In Figure 5(a), the angles obtained by the statistical approach are plotted against the number of dimensions. The dotted line is a threshold; we estimate the dimension as the value where the angle falls below the threshold, that is at $d = 21$. This is the same as PCA d^{10}. The locally linear scale in Figure 5(b) indicates that the intrinsic dimensionality falls approximately between 9 to 22 dimensions. Using the elbow test on residual variances of Isomap, the estimate is about 18. The plot of reconstruction error in Figure 5(d) obtained by LLE does not have an elbow shape, making it difficult to identify the intrinsic dimensionality.

In contrast to *First*, the results for *FirstTriples* shown in Figure 6 indicates that PCA, Isomap, LLE, and Laplacian Eigenmaps improve the error rates. The best performance is obtained using the top 2 features from PCA and from



(a) Statistical Approach ($d = 21$)

(b) Locally Linear Scale ($d \approx 9 \sim 22$)



(c) Isomap with $k = 5$. ($d \approx 18$)

(d) LLE with $k = 15$. ($d \approx 15$)

Fig. 5. Intrinsic dimensionality estimation on *First* dataset

Isomap, top 12 features from LLE, and top 7 features from Laplacian Eigenmaps. However, none of the NLDR techniques perform better than the four feature subset selection methods. In addition, since the *FirstTriples* dataset is derived from *First*, and all methods give lower error rates on *FirstTriples*, it emphasizes that the dataset is less noisy, confirming the scientists expectation.

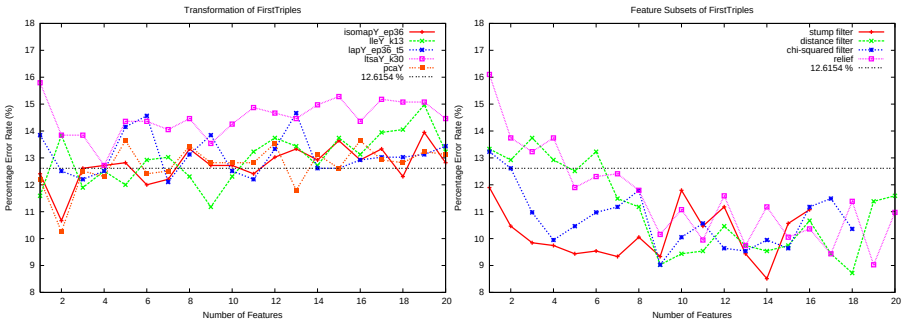
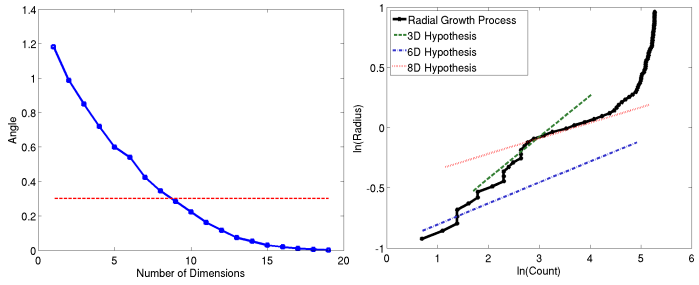
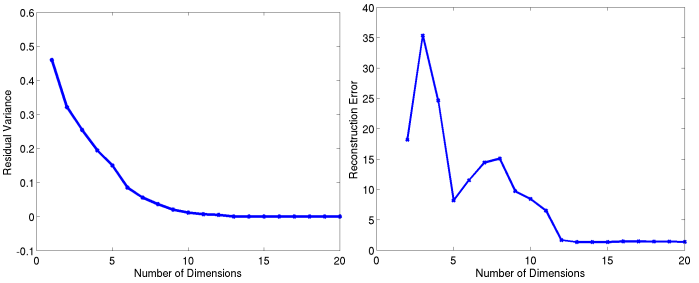


Fig. 6. Classification error rates using decision tree classifiers on the transformed features (left) and selected features (right) for the *FirstTriples* dataset.



(a) Statistical Approach ($d = 9$) (b) Locally Linear Scale ($d \approx 8$)



(c) Isomap with $\epsilon = 3.6$ ($d \approx 9$) (d) LLE with $k = 12$ ($d \approx 12$)

Fig. 7. Intrinsic dimensionality estimation on *FirstTriples* dataset

We observe that the *FirstTriples* dataset, with fewer features, has a smaller variance in the estimation of the intrinsic dimensionality in comparison to the *First* dataset. This may due to the small ratio of the set cardinality n to the number of dimensions D. In order to obtain an accurate estimation of the dimensionality, it has been proven that the inequality $D < 2\log_{10} n$ should be satisfied [30]. The number of data points needed to accurately estimate the dimension of a D-dimensional data set is at least $10^{\frac{D}{2}}$. So, in practice, if the sample size of a dataset is small, we should try reducing the number of features using domain information prior to determining its intrinsic dimensionality.

Figure 8 is a biplot of the *First* dataset. Points shown in the plots are observations represented by the top two dimensions of PCA, and lines reflect the projections of the original features on to the new space. The length of the lines approximates the importance of the features. To avoid a large number of overlaps, only features whose elements in the eigenvector have absolute values that are larger than 0.2 are shown in the plot.

Feature CoreAngl (indicated as 8 in the plot) has positive projection on to both the first and the second dimensions. There is a negative correlation between Feature CoreAngl and a subset of features related to angles (9, 10 and 13) and symmetry (20). This subset of features have negative projection on to both the first and the second dimensions. The bents observations are clustered at low values of the distance features (45, 16, 90 and 33). The non-bents observations

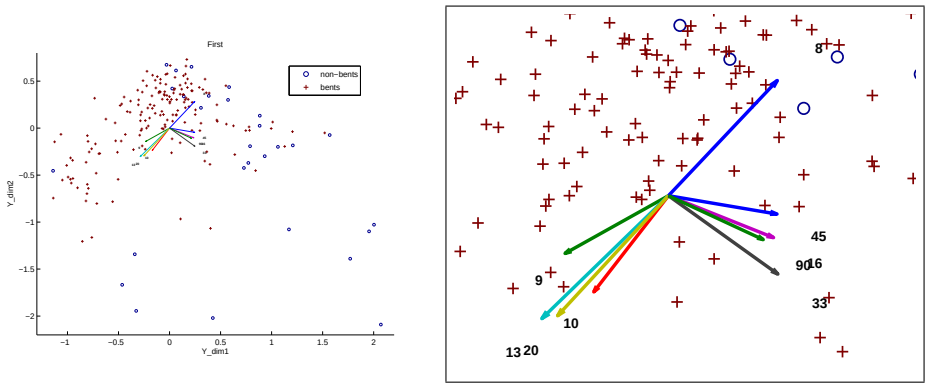


Fig. 8. Left: PCA biplot of the *First* dataset. Right: zoomed-in view.

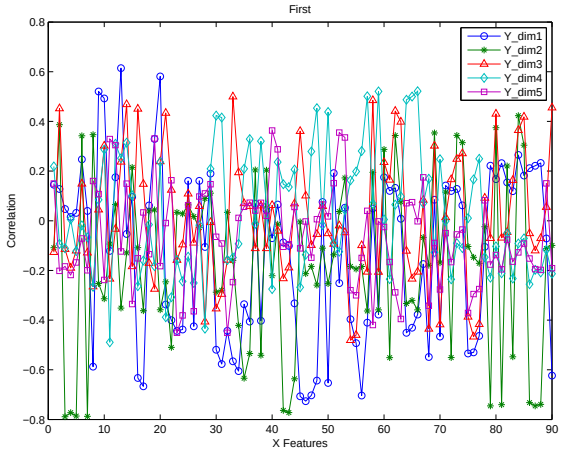


Fig. 9. Correlation between top five Isomap reduced dimensions and all original features for *First* dataset.

forms a cluster at the near highest CoreAngl values, near lowest symmetries (20) and near lowest other angle values (at 9, 10 and 13) as well as the high distances (at 45, 16, 90 and 33). We may interpret both the first and second PC dimensions as distance-angle dimensions. These observations support the visual labeling process used by astronomers, where symmetry is an important feature of bent-doubles, and angles are an important discriminating feature.

Figure 9 shows the linear correlation between Isomap dimensions and the original features. Seven out of the top 20 features that are highly correlated to the first Isomap dimension are also among the top 9 features PCA highly rated. Although many features are highly correlated to the second Isomap dimension, none of them are among the top PCA features. These are the linear relationships that we can explain. Nonlinear relationships among the features are still

unknown. In the decision tree classification, using only one dimension of Isomap can give a better classification performance than the original dataset. It indicates that the first dimension of Isomap captures a property of the data that reflects the class labels.

The feature subset selection methods rank highly the features of the *First* dataset which are related to symmetries and angles, consistent with what PCA has captured for the top two PCs.

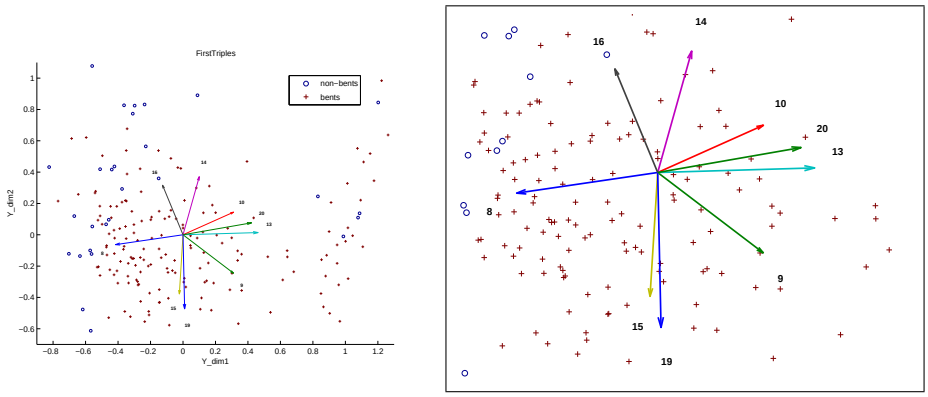


Fig. 10. Left: PCA biplot of the *FirstTriples* dataset. Right: zoomed-in view.

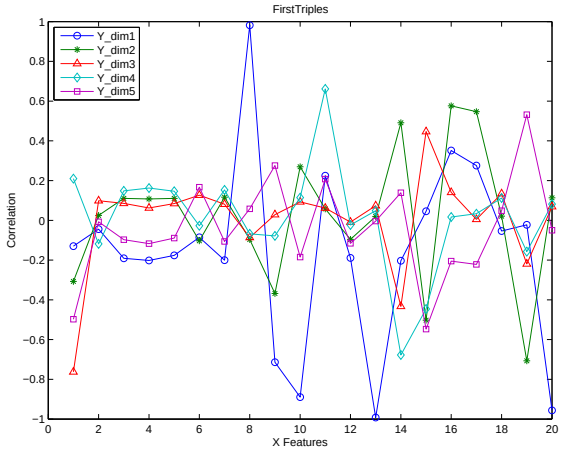


Fig. 11. Correlation between top five Isomap reduced dimensions and all original features for *FirstTriples* dataset.

Figure 10 displays the PCA biplot of *FirstTriples*. Seven out of the nine features whose elements are significant in either one of the top two eigenvectors, are also among those that PCA chooses for the *First* dataset. These features are as well consistent with the highly ranked features from filter methods. The result emphasizes that PCA can be a good measure for removing noise features, although the PCA representations of *First* data do not improve the classification.

There is again a negative correlation between feature CoreAngl (8) and a subset of features related to angles (9,10,13 and 14) and symmetry (20). These features are parallel to the first PC coordinate, which tells us that the first dimension is an angle coordinate. We can also see clusters of non-bents fall around the extreme values of angle features, while clusters of bents have medium angle values. The second dimension means the AriSym (19) v.s. ABAngleSide (14) and SumComDist (16). We can see a cluster of bents that has small values of SumComDist (16), large values of AriSym (19) and medium values of angles. There is also a non-bent cluster at the near highest CoreAngl values, near lowest symmetries (20) and near lowest other angle values (at 9,10 and 13). This fact is similar to what PCA gets from *First* dataset. The two PCs together shows that there exist no non-bents at the corner area where the AriSym is high and CorAngl is low.

Figure 11 shows that Isomap dimensions have linear correlations with a subset of features that are similar to what PCA dimensions have. The first dimension of Isomap has significantly linear correlations with a subset of features. Similar to PCA, in the first dimension there is a negative correlation between feature CoreAngl (8) and a subset of features, containing features AngleAB (9), AngleAC (10), AriAgl (13) and AnotherSym (20). The second dimension of Isomap tells its linear correlation with feature AriSym (19), which is also a main feature in the second dimension of PCA. In the decision tree classification shown in Figure 6, both PCA and Isomap obtain best classification performance using their top two features. This similarity in performance of PCA and Isomap strengthens the possibility that Isomap captures the linear properties in the *FirstTriples* dataset, and it is unlikely there is a nonlinear manifold underlying the data.

Finally, we observe that the subset selection methods, PCA, and Isomap all selected features of the *FirstTriples* dataset related to symmetries and angles.

7.2 Experiments on *Wind115* and *Wind150* Datasets

Figure 12 displays the classification results for the *Wind115* dataset. It is significant that the feature subset selection techniques outperform the data transformation methods. All data transformation techniques do not reach the accuracy of the original data. This can due to the bad labeling on the data.

The performance on *Wind150* dataset, which is labeled differently, is shown in Figure 13. We observed that all methods give lower error rates than on *Wind115*, which indicates that labeling the data according to 150 MW ramps can help identify events more significantly.

Again, the feature selection methods outperform the data transformation techniques. Isomap and PCA are the only two data transformation techniques

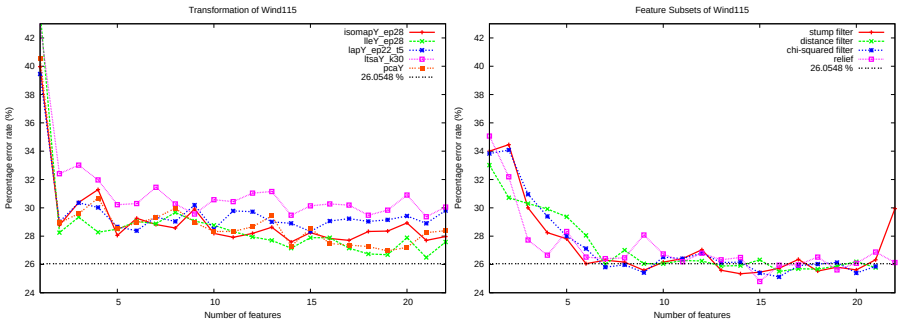


Fig. 12. Classification error rates using decision tree classifiers on the (left) transformed features and (right) selected features for the *Wind115* dataset.

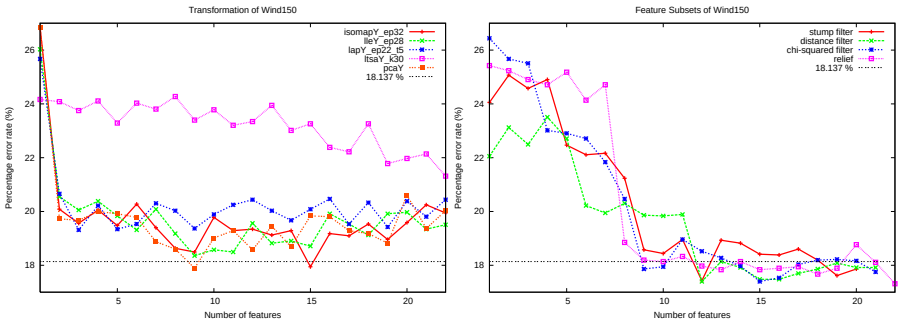
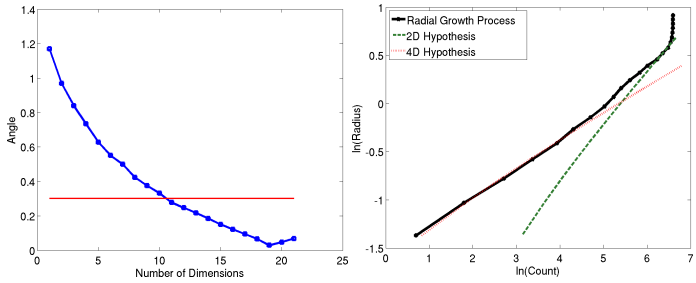


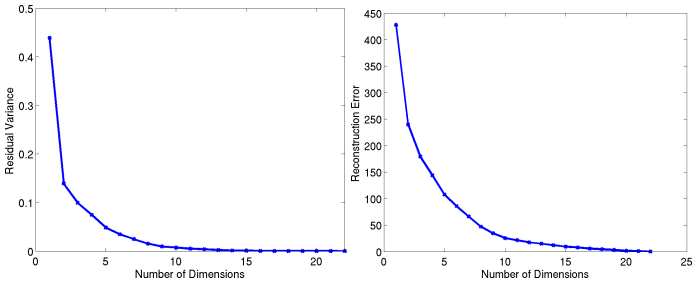
Fig. 13. Classification error rates using decision tree classifiers on the (left) transformed features and (right) selected features for the *Wind150* dataset.

that, in comparison to the original data, slightly improve the classification. The best performance of PCA is at $d = 9$, and the best of Isomap is at $d = 15$ and $\epsilon = 3.2$ (displayed as isomapY_ep32). At $d = 12$ the stump filter and the distance filter reach their lowest error rates, and at $d = 15$ the chi-squared filter has the lowest error rate. However, we observe that when the number of features is less than 9, the data transformation methods are more accurate than the feature selection methods.

Both *Wind115* and *Wind150* are the same dataset, but with different labeling criteria. Hence, they have the same intrinsic dimensionality shown in Figure 14. The estimate of statistical approach is $d = 11$, while locally linear scales give $d = 4$. The elbow test on residual variances of Isomap gives $d \approx 9$, close to the statistical approach. The dimensionality according to the elbow test on reconstruction error of LLE gives $d \approx 10$ to 15. All are near the range of $d \approx 5$ to 15 that PCA estimates.



(a) Statistical Approach ($d = 11$) (b) Locally Linear Scale ($d = 4$)



(c) Isomap with $\epsilon = 3.2$ ($d \approx 9$) (d) LLE with $\epsilon = 3.2$ ($d \approx 10 - 15$)

Fig. 14. Intrinsic dimensionality estimation on *Wind115* and *Wind150* datasets

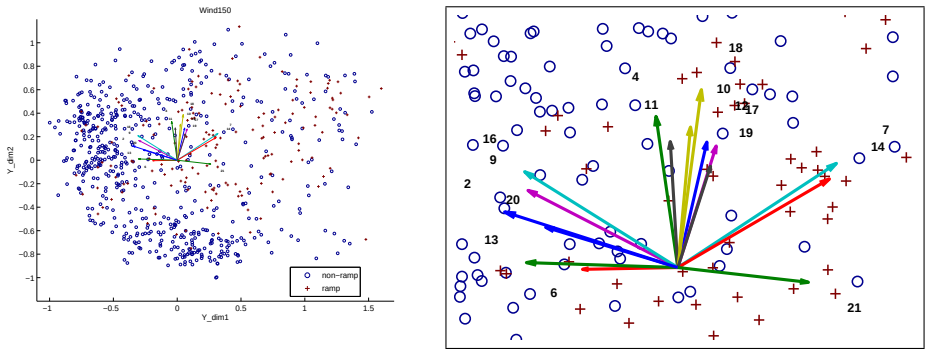


Fig. 15. Left: PCA biplot of the *Wind150* dataset. Right: zoomed-in view.

The PCA biplot of *Wind150* shown in Figure 15 shows that the first coordinate has two subsets of features that are negatively correlated. One is the humidity features at three weather sites (7, 14, 21). The other subset contains the temperature features (6, 13, 20) and the solar radiation features (2, 9, 16) at three weather sites. The second principal component is about wind direction vector (4, 11, 18) and speed (10, 12, 17, 19) that are positively correlated.

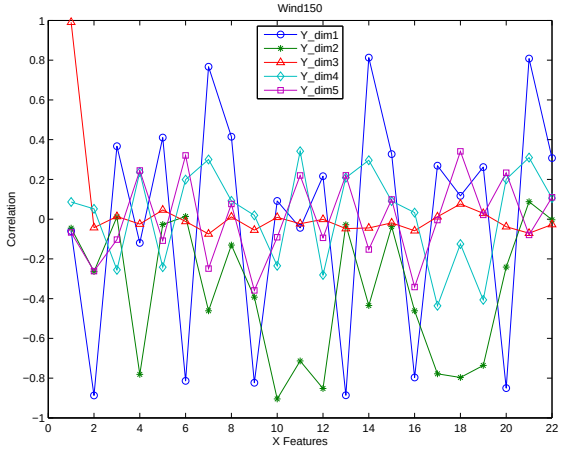


Fig. 16. Correlation between top five Isomap dimensions and all original features for *Wind150* dataset.

There are two clusters that are dense. One contains observations that have low wind speeds and low wind direction vector degrees. The other cluster is at high temperature, high solar radiation and low humidity. These charactors represent non-ramp events, which are consistant with the labels shown on the graph. There exits no clusters of ramp events that are obviously dense.

The linear correlation between the first five Isomap dimensions and all original features for *Wind150* is shown in Figure 16. Like PCA, the first dimension of Isomap is linearly correlated to features of humidity, temperature, and solar radiation at three weather sites. Similarly, the second dimension of Isomap is linearly correlated to wind direction vector degrees and speeds. This implies that the isomap captures the linear relations of the data. It is not straight-forward to determine the existence of any nonlinear relations.

Finally, the top six common features that are ranked highly by all three filters are also the wind speed, temperature and humidity. This consistancy shows the success of filters, PCA, and Isomap in dimension reduction.

7.3 Experiments on *RemoteSmall* Dataset

Figure 17 shows the classification error rates for the *RemoteSmall* data set. Only 50 of the 496 features are displayed because the rates become almost constant when large numbers of features are used for all methods.

Though the feature subset selection methods still outperform the data transformation techniques, all methods perform well on *RemoteSmall*. Isomap, LLE and PCA have similar performance and reach the minimum error rate at $6-9$ dimensions. In contrast, Laplacian Eigenmaps reaches its best performance at

$d = 34$ and LTSA at $d = 26$. The result could be due to the actually high-dimensional data with a large number of samples ($n = 2000$). Thus, the lower dimensional structure exist and the data transformation methods can find them.

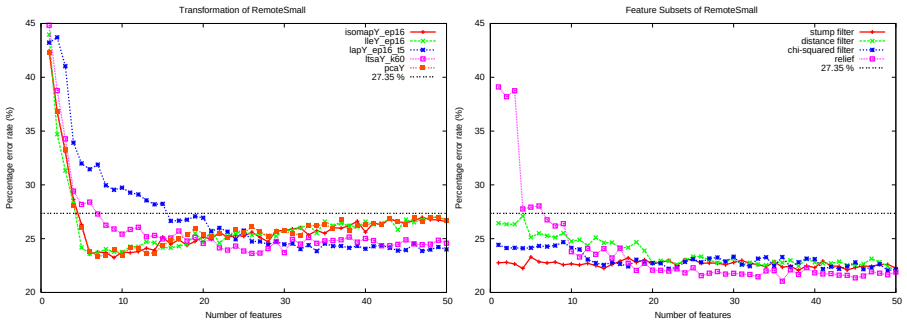
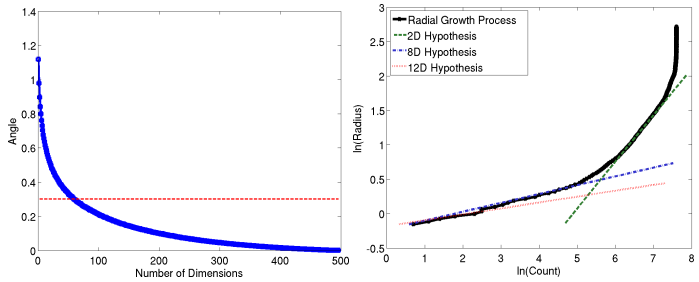
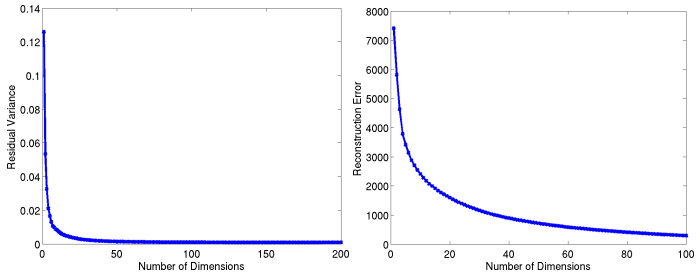


Fig. 17. Classification error rates using decision tree classifiers on the (left) transformed features and (right) selected features for the *RemoteSmall* dataset.



(a) Statistical Approach ($d = 60$)

(b) Locally Linear Scale ($d = 8$)



(c) Isomap with $\epsilon = 16$ ($d \approx 6$)

(d) LLE with $\epsilon = 16$

Fig. 18. Intrinsic dimensionality estimation on *RemoteSmall* dataset

For *RemoteSmall* dataset, the statistical approach gives an intrinsic dimensionality estimate of $d = 60$, which is quite different from $d = 8$ estimated using locally linear scale. PCA gives small numbers of estimation as well. Elbow test on Figure 18(c) shows that $d = 6$ is right below the cliff and the flat region begins at around $d = 20$. Combining the results given in Figure 17 and Figure 18(c), we can see that Isomap with $\epsilon = 16$ gives the minimum error rate at dimensionality close to the estimate of $d \approx 6$. LLE reconstruction error seems not a reliable indicator for estimating intrinsic dimentionality.

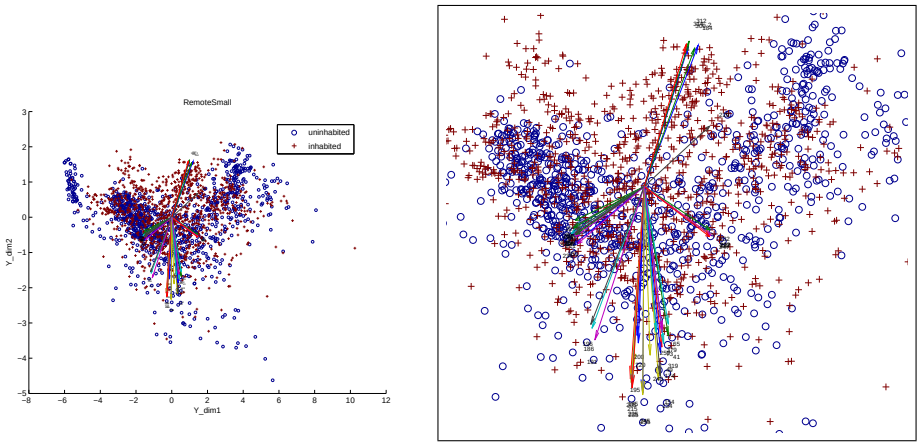


Fig. 19. Left: PCA biplot of the *RemoteSmall* dataset. Right: zoomed-in view. Vectors on the graph are 20 times larger than their original sizes for easier identification.

In Figure 19 the PCA biplot of *RemoteSmall* shows a large number of features that project observations in the first two PCs. Features pointing to the bottom right are all features from entropy in GLCM category, and most are green bands. Features pointing to the left are all features from inverse difference moment in GLCM category with green, blue and red bands. These two subsets of features are negatively correlated. They determine the first coordinate. Most features that point to the top are features from Gabor and wavelet categories. They are all features of near-infrared bands. Features pointing to the bottom are features of GLCM category with near-infrared, green, blue and red bands. They can be used to explain the second PC. The observations form a funnel on the plot, indicating that one dimension affects the variance of another orthogonal dimension. It means that high GLCM values are similar in their entropy and inverse difference moment, while low GLCM values are more varied.

The feature selection methods rank highly the features in the green and near-infrared bands rather than the blue and red bands. The majority of the top ten features are from the GLCM category, while the wavelet and Gabor features are selected less frequently. Power spectrum features are rarely selected. The GLCM features selected most often in top ten are entropy and inverse difference

moment. These results agree with what PCA suggests. The linear correlations between Isomap dimensions and the original features are again similar to PCA.

8 Related Work

In this paper, we have focused on a few popular data transformation methods for dimension reduction: PCA, Isomap, LLE, Laplacian Eigenmaps and LTSA. Many other techniques have also been proposed, including Hessian Eigenmaps [31], neighborhood preserving methods [32, 33], diffusion maps [34], and local tangent space analysis [35], as well as techniques that reduce the data to two dimensions for visualization, such as t-distributed stochastic neighbor embedding (tSNE) [36], self-organizing maps [37], and neural network-based approaches [38].

Much of the work in NLDR techniques has focused on the algorithmic aspects, with experiments on artificial datasets illustrating the benefits of these methods. However, a recent comparative study [6] on several NLDR techniques applied to both artificial and real datasets concluded that the strong performance of these techniques on the artificial Swiss roll data does not generalize to more complex, artificial datasets, such as those with disconnected manifolds or manifolds with high intrinsic dimensionality. In addition, most nonlinear techniques do not outperform PCA on real data sets. Another comparative study of dimension reduction techniques [39] also shows that for data visualization purposes, NLDR techniques generally perform better on the synthetic data than on the real-world data, and the overall best performing algorithm is Isomap.

Our study does not focus on data visualization, but on practical scientific data analysis. The experiments presented in this paper also support the conclusions from these comparative studies. However, there are successful applications of NLDR on real world datasets [5], and methods, such as tSNE, when used for visualization, have been shown to provide insights into the inherent structure in high-dimensional data [36]. It appears that the best NLDR technique depends on the nature of the input data and on the use of the reduced representation [5].

9 Conclusions

In this paper, we describe a series of carefully-designed experiments that test, in a useful and impartial manner, how dimension reduction methods work in practice. We investigate two types of techniques: data transformation methods and feature subset selection techniques. Using classification problems in five scientific datasets, each exhibiting different data properties, we compare the error rates for the original dataset with those obtained for the reduced representations resulting from the data transformaion methods as well as feature selection techniques. We also evaluate the intrinsic dimensionality of the data using estimates obtained from PCA and two of the NLDR methods (Isomap and LLE), in addition to two classical techniques, one based on a statistical approach and the other on a locally linear scale.

Our experiments indicate that, while the supervised feature subset selection techniques consistently improve the classification of all datasets, the data transformation methods do not. However, it is possible to use them to find properties of the data related to class labels. Our experiments show that both PCA and Isomap are able to find representations that improve data classification. Since both PCA and Isomap employ the eigenvectors corresponding to the largest eigenvalues, they seem to perform better than methods which use the eigenvectors corresponding to the smallest non-zero eigenvalues, such as LLE, Laplacian Eigenmaps and LTSA. Like PCA, when the data tend to have strong linear properties, Isomap can identify these properties. Isomap can also capture some kind of nonlinear properties that PCA can not find. Although there exists applications indicating that PCA is better than Isomap in terms of classification [6], our experiments indicate a different conclusion. We also observe that the ability to interpret the reduced dimension made by data tranformation methods is very limited.

Since feature subset selection techniques are computationally inexpensive, we suggest using them first, especially as they could provide insights into the dataset by indicating which of the original features are important. If a dataset contains noise features, the use of feature subset selection techniques to identify and remove possible noise features prior to the application of the data transformation methods could also be helpful. Among the feature subset selection techniques, the filter-based methods give more consistent results. The estimation of intrinsic dimensionality of the dataset may vary, depending on the method used. However, the estimate could be meaningful if it is close to the number of features that give the best performance. For an NLDR method, this may also imply that the method finds the lower-dimensional manifold on which the data lie, something which is not possible with linear feature subset selection.

References

1. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. Journal of Machine Learning Research **3** (2003) 1157–1182
2. Kohavi, R., John, G.: Wrappers for feature subset selection. Artificial Intelligence **97** (1997) 273–324
3. Pearson, K.: On lines and planes of closest fit to systems of points in space. Philosophical Magazine **2** (1901) 559–572
4. Lee, J.A., Verleysen, M.: Nonlinear Dimensionality Reduction. Springer, New York, NY, USA (2007)
5. Niskanen, M., Silvn, O.: Comparison of dimensionality reduction methods for wood surface inspection. In: Proceedings of the 6 th International Conference on Quality Control by Artificial Vision. (2003) 178–188
6. van der Maaten, L., Postma, E., van den Herik, J.: Dimensionality reduction: A comparative review. Technical Report TiCC TR 2009–005, Tilburg University (2009)
7. Tenenbaum, J.B., Silva, V., Langford, J.C.: A Global Geometric Framework for Nonlinear Dimensionality Reduction. Science **290** (2000) 2319–2323

8. Dijkstra, E.W.: A note on two problems in connexion with graphs. Numerische Mathematik **1** (1959) 269–271
9. Floyd, R.W.: Algorithm 97: Shortest path. Commun. ACM **5** (1962) 345–
10. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. Science **290** (2000) 2323–2326
11. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation **15** (2003) 1373–1396
12. Zhang, Z., Zha, H.: Principal manifolds and nonlinear dimension reduction via local tangent space alignment. SIAM Journal of Scientific Computing **26** (2002) 313–338
13. Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.J.: Classification and Regression Trees. CRC Press, Boca Raton, FL (1984)
14. Huang, S.H.: Dimensionality reduction on automatic knowledge acquisition: a simple greedy search appr oach. IEEE Transactions on Knowledge and Data Engineering **15** (2003) 1364–1373
15. Robnik-Sikonja, M., Kononenko, I.: Theoretical and empirical analysis of relieff and rrelieff. Machine Learning **53** (2003) 23–69
16. Valle, S., Li, W., Qin, S.J.: Selection of the number of principal components: the variance of the reconstruction error criterion with a comparison to other methods. Industrial & Engineering Chemistry Research **38** (1999) 4389–4401
17. Fukunaga, K., Olsen, D.: An algorithm for finding intrinsic dimensionality of data. IEEE Transactions on Computers **C-20** (1971) 176 – 183
18. Saul, L.K., Roweis, S.T., Singer, Y.: Think globally, fit locally: Unsupervised learning of low dimensional manifolds. Journal of Machine Learning Research **4** (2003) 119–155
19. Kegl, B.: Intrinsic dimension estimation using packing numbers. In: Advances in Neural Information Processing Systems: NIPS, MIT Press (2002) 681–688
20. Trunk, G.V.: Statistical estimation of the intrinsic dimensionality of a noisy signal collection. Computers, IEEE Transactions on **C-25** (1976) 165 –171
21. Becker, R.H., White, R.L., Helfand, D.J.: The FIRST survey: Faint Images of the Radio Sky at Twenty-cm. Astrophysical Journal **450** (1995) 559
22. Haralick, R.M., Shanmugam, K., Dinstein, I.: Textural features for image classification. IEEE Transactions on Systems, Man, and Cybernetics **3** (1973) 610–621
23. Manjunath, B.S., Ma, W.Y.: Texture features for browsing and retrieval of image data. IEEE Transactions on Pattern Analysis and Machine Intelligence **18** (1996) 837–842
24. Newsam, S., Kamath, C.: Retrieval using texture features in high-resolution, multispectral satellite imagery. In: Data Mining and Knowledge Discovery: Theory, Tools, and Technology, VI, Proceedings of SPIE Vol. 5433, SPIE Press (2004) 21–32
25. Sabato, S., Shalev-Shwartz, S.: Ranking categorical features using generalization properties. J. Mach. Learn. Res. **9** (2008) 1083–1114
26. Tibshirani, R.: Regression shrinkage and selection via the lasso. Journal of the Royal Statistical Society, Series B **58** (1996) 267–288
27. Zou, H., Hastie, T.: Regularization and variable selection via the elastic net. Journal of the Royal Statistical Society, Series B **67** (2005) 301–320
28. Kamath, C., Cantú-Paz, E., Littau, D.: Approximate splitting for ensembles of trees using histograms. In: Proceedings, Second SIAM International Conference on Data Mining. (2002) 370–383
29. Gabriel, K.R.: The biplot graphic display of matrices with application to principal component analysis. Biometrika **58** (1971) 453–467

30. Smith, L.A.: Intrinsic limits on dimension calculations. Physics Letters A **133** (1988) 283 – 288
31. Donoho, D.L., Grimes, C.: Hessian eigenmaps: Locally linear embedding techniques for high-dimensional data. PNAS **100** (2003) 5591–5596
32. He, X., Cai, D., Yan, S., Zhang, H.J.: Neighborhood preserving embedding. In: Computer Vision, Tenth IEEE International Conference on. Volume 2. (2005) 1208 –1213
33. Kokiopoulou, E., Saad, Y.: Orthogonal neighborhood preserving projections: A projection-based dimensionality reduction technique. IEEE Transactions on Pattern Analysis and Machine Intelligence **29** (2007) 2143–2156
34. Coifman, R.R., Lafon, S.: Diffusion maps. Applied and Computational Harmonic Analysis **21** (2006) 5 – 30
35. Zhang, Z., Zha, H.: Principal manifolds and nonlinear dimension reduction via local tangent space alignment. SIAM Journal of Scientific Computing **26** (2004) 313–338
36. van der Maaten, L., Hinton, G.: Visualizing Data using t-SNE. Journal of Machine Learning Research **9** (2008) 2579–2605
37. Kohonen, T.: Self-organized formation of topologically correct feature maps. Biological Cybernetics **43** (1982) 59–69
38. Haykin, S.: Neural Networks: A Comprehensive Foundation. 2nd edn. Prentice Hall PTR, Upper Saddle River, NJ, USA (1998)
39. Tsai, F.S.: Comparative study of dimensionality reduction techniques for data visualization. Journal of Artificial Intelligence **3** (2010) 119–134