Dimensionality Reduction

1^{st} Yao An Lee
Department of Mechanical and Aerospace Engineering, University of Florida, Gainesville, FL 32611 USA
danielyaoanlee@gmail.com

***Abstract*—Over the years, Machine Learning(ML) has been widely used in several industries and usually need to deal with large data set. This study proposes several dimensionality reduction algorithms over handwritten digit dataset to reduced the dimensionality of data, compare the projection space and feed the reduced data into classification model to experiment their performance. In my study, I compare the performance between original data and reduced data (done by PCA) using logistic regression as classifier. Performance between data with MDS, Isomap and LLE are also done using logistic regression. I also compare the feature selecting result under RFE with Logistic Regression and SVM Classifier as the estimators. The 2-D projection of reduced data is compared between t-SNE, LDA and PCA. The metric to compare model performance is determined by the accuracy of classification, the classification report and confusion matrix. All models are tuned using hyperparameter tuning techniques to maximize performance of each model.**

***Index Terms*—Machine Learning, Dimensionality Reduction, RFE, PCA, LDA, t-SNE, MDS, Isomap, LLE, Logistic Regression, Random Forest Classifier.**

I. Introduction

Machine learning has developed into a widely used framework for approximating statistical and algebraic relations over a given set of data for decades. With large dataset, we can reduce the influence of overfitting but often result in high computaional expenses. Dimensionality reduction algorithms are introduced to project the data into lower dimensional space and select subset of features to increase the computation speed and not sacrifice too much of the classification performances.

A. Data

The dataset is collected by students in University of Florida. It contains 9600 samples with 90000 pixels as features and 10 classes (Fig.1) are included in the data. Among these data, both higher and lower cases are presented for alphabets. The train and test split is considered to be 70:30 for all algorithms.

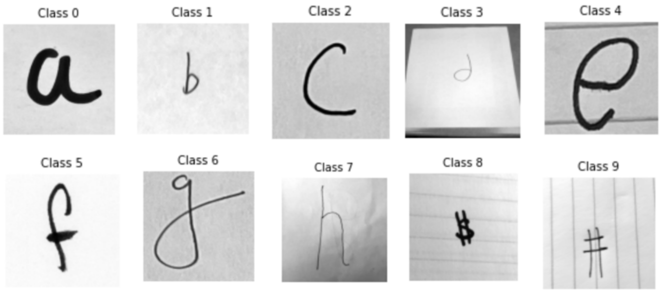


Fig. 1. Handwritten digit

B. Logistic Regression and SVM Classifier

Logistic Regression and SVM Classifier are two widely used and simple classification algorithms. Both of them can be implemented using sklearn library. In this study, I use both of these classification algorithms to train the data and as the estimators of dimensionality reduction algorithms.

C. Feature Selection: RFE

Recursive Feature Elimination (RFE) is a feature selection algorithm. It automatically select a subset of features that are most relevant to the problem with a minimum decay in performance to improve computational efficiency or reduce the generalization error of the model by removing irrelevant features or noise.

D. Feature Extraction: PCA, LDA

Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) are linear feature extraction algorithms. Feature extraction derive information from the feature set to construct a new feature subspace. For PCA, it describes the direction of maximum variance in data and is unsupervised thus not able to deal with labeled and non-linear data. Unlike PCA, LDA describes the direction of maximum separability in data and is supervised thus requires class label and be able to deal with labeled data.

E. Manifold Learning: t-SNE, MDS, Isomap, LLE

T-distributed stochastic neighbor embedding (t-SNE), Multi-Dimension Scaling (MDS), Isometric mapping (Isomap), Locally Linear Embedding (LLE) are manifold learning algorithms and do not assume the data is linear. t-SNE is usually used when visualization, it not only captures the local structure of the higher dimension but also preserves the global structures of the data. However, t-SNE does not learn a general function that can be effectively applied to unseen data and thus is not great at pre-processing features for prediction. MDS finds a low-dimensional projection of the data such as to preserve the pairwise distances between data points, and involves finding the eigenvectors of the distance matrix and thus is a generalization of PCA and Isomap. Isomap project the data to a lower-dimensional space using MDS, but the distance/dissimilarities are defined in terms of the geodesic distances measured along the manifold. LLE first computes the set of coefficients that best reconstructs each data point from its neighbors. These coefficients are arranged to be invariant to rotations, translations, and scalings of that data point and its neighbors, and hence they characterize the local geometrical properties of the neighborhood. Among these algorithms, t-SNE preserves both global and local structure, MDS and Isomap preserve global structure, LLE only preserves local data structure.

II. Experiment

A. Data preprocessing

Data is formed by 90000 pixels in each 300x300 image and thus the value in each feature are between 0 to 255. Due to the large dataset, in order to reduce the computational expenses, I import cv2 library and downsample the dataset from 300x300 to 50x50. Note that downsampling will result in low accuracy. Since there is no missing data or outliers, We can then just implement MinMaxScaler from sklearn library to scale the data.

B. Hyperparameter tuning

Due to the objective of this project is to study the effect of different dimensionality reduction algorithms, I choose to use the default value of Logistic Regression in sklearn library and only number of components for each manifold learning algorithms are considered as hyperparameter to be tuned. The hyperparameter is selected using Cross Validation(CV) Grid search.

C. Feature selection using RFE

In this project, logistic regression and SVM classifier are selected as estimators for RFE to select subset features and visualize the mask examples from the training dataset.

D. PCA

For PCA, I set the number of components and visualize the cumulative summation of explained variance. The goal is to select the number of components that explain at least 90 percent of the explained variance and compare the performance and run time between original and reduced data. Top ten components and reconstruct image are also visualized to provide deeper understanding.

E. 2-D projection: LDA, t-SNE and PCA

Implement LDA and t-SNE and set number of components equals 2 to reduce the dataset to a 2-D surface. Visualization is presented to show the projection and to compare with 2-D projection from PCA.

F. Dimensionality reduction: MDS, Isomap, LLE

MDS, Isomap and LLE are included in pipelines together with logistic regression. Hyperparameter tuning are implemented to find the best number of embedded features reduced by these manifold learning algorithms. Visualization of first two dimensions and comparison between results of each algorithm are provided.

III. Results and discussion

A. RFE: Mask example and test data transformation

For RFE with logistic regression as estimator, number of feature to select (Fig.2) is 2330. For RFE with SVM as estimator, number of feature to select is 2490. The best accuracy we can derive in training data is 92.43 percent with SVM as estimator. Compared to training accuracy, test accuracy in both estimators are 35.49, 36.98 percent, respectively.

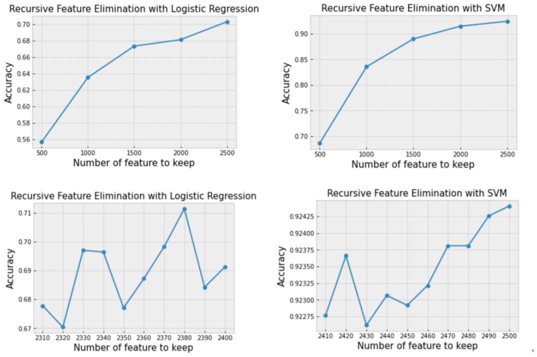


Fig. 2. Mask example for RFE

B. PCA: Performance of original and reduced data

In order to get reduced data with minimum complexity and explain at least 90 percent of explained variance, we perform PCA with a large number of components and plot the cumulative summation of explained variance (Fig.3) to help me select the correct number of components. Based on the result, we can see that the number of components that explain at least 90 percent of the explained variance is 183.

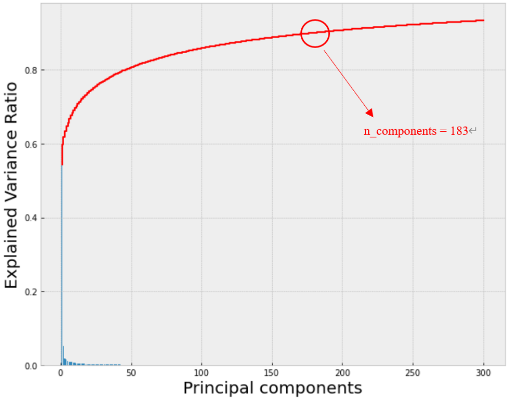


Fig. 3. Cumulative summation of explained variance

- Training time for original data with hyperparameter tuned logistic regression is 81.75 secs. Training time for reduced data with hyperparameter tuned logistic regression is 4.84 secs. Thus training for reduced data is much more faster than original data.
- Eigenvectors like PC1 and PC2 show the region that most characters exist is in the middle square area. Eigenvectors

PC3 and PC4 show that in the middle square area, some characters tend to have a shape like ellipse in the middle. Eigenvectors PC5, PC6, PC7 and PC8 show that some characters are located slightly closer to up, down, left and right. Eigenvectors PC9 and PC10 show there exist some situation that character edges are complex but still mostly in the middle of the image.

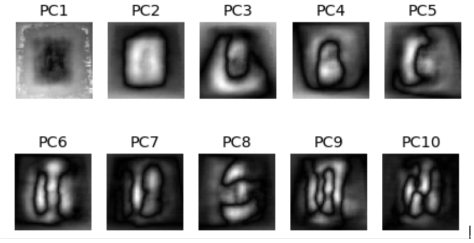


Fig. 4. Top 10 eigenvectors for reduced data

- Reconstruct image (Fig.5) is a little blur but can still visually identify the characters.

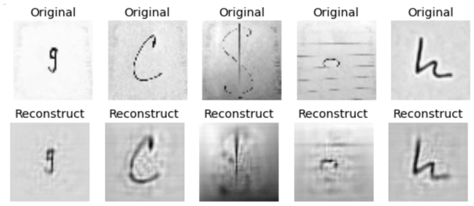
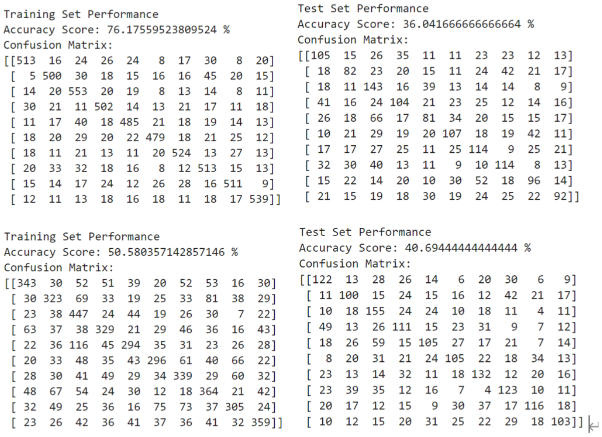


Fig. 5. Reconstruct image

- Due to the influence of downsampling, we can expect the accuracy of downsampled data is lower. In performance measurement (Fig.6), we can see that although train accuracy for reduced data is a lower than original data, test accuracy for reduced data is higher than original data. This result shows that reduced data with 90 percent of explained variance can still be representative and cost lower computational expenses. The reason that accuracy are all very low and obviously overfitting is because PCA in unsupervised, it's not able to deal with non-linear and labeled data well.

C. 2-D projection: LDA, t-SNE and PCA

PCA and t-SNE are unsupervised and PCA is not able to visualize non-linear data well. Usually, t-SNE is able to visualize non-linear data into clear clusters and preserve local and global relationship, but in the result I present (Fig.7), overlapping in PCA and t-SNE are severe. Overlapping in t-SNE means that there is high possibility that the performance of classifier will be really bad when selecting 2 components. As for LDA, LDA is supervised and thus is able to visually separate the data better than PCA, but there still exists some overlap and thus we can conclude that the performance when only select 2 components will not be good. Based on the result derived in section PCA, at least 183 components need to be preserve to have a better generalization. As for the visualization of transformation in test set, 2 dimension is not enough to represent the whole data and thus have severe overlapping in all projection.



```
Training Set Performance
Accuracy Score: 76.17559523809524 %
Confusion Matrix:
[[513  16  24  26  24   8  17  30   8  20]
 [  5 500  30  18  15  16  16  45  20  15]
 [ 14  20 553  20  19   8  13  14   8  11]
 [ 30  21  11 502  14  13  21  17  11  18]
 [ 11  17  40  18 485  21  18  19  14  13]
 [ 18  20  29  20  22 479  18  21  25  12]
 [ 18  11  21  13  11  20 524  13  27  13]
 [ 20  33  32  18  16   8  12 513  15  13]
 [ 15  14  17  24  12  26  28  16 511   9]
 [ 12  11  13  18  16  18  11  18  17 539]]
```

```
Test Set Performance
Accuracy Score: 36.041666666666664 %
Confusion Matrix:
[[105  15  26  35  11  11  23  23  12  13]
 [ 18  82  23  20  15  11  24  42  21  17]
 [ 18  11 143  16  39  13  14  14   8   9]
 [ 41  16  24 104  21  23  25  12  14  16]
 [ 26  18  66  17  81  34  20  15  15  17]
 [ 10  21  29  19  20 107  18  19  42  11]
 [ 17  17  27  25  11  25 114   9  25  21]
 [ 32  30  40  13  11   9  10 114   8  13]
 [ 15  22  14  20  10  30  52  18  96  14]
 [ 21  15  19  18  30  19  24  25  22  92]]
```

```
Training Set Performance
Accuracy Score: 50.580357142857146 %
Confusion Matrix:
[[343  30  52  51  39  20  52  53  16  30]
 [ 30 323  69  33  19  25  33  81  38  29]
 [ 23  38 447  24  44  19  26  30   7  22]
 [ 63  37  38 329  21  29  46  36  16  43]
 [ 22  36 116  45 294  35  31  23  26  28]
 [ 20  33  48  35  43 296  61  40  66  22]
 [ 28  30  41  49  29  34 339  29  60  32]
 [ 48  67  54  24  30  12  18 364  21  42]
 [ 32  49  25  36  16  75  73  37 305  24]
 [ 23  26  42  36  41  37  36  41  32 359]]
```

```
Test Set Performance
Accuracy Score: 40.69444444444444 %
Confusion Matrix:
[[122  13  28  26  14   6  20  30   6   9]
 [ 11 100  15  24  15  16  12  42  21  17]
 [ 10  18 155  24  24  10  18  11   4  11]
 [ 49  13  26 111  15  23  31   9   7  12]
 [ 18  26  59  15 105  27  17  21   7  14]
 [  8  20  31  21  24 105  22  18  34  13]
 [ 23  13  14  32  11  18 132  12  20  16]
 [ 23  39  35  12  16   7   4 123  10  11]
 [ 20  17  12  15   9  30  37  17 116  18]
 [ 10  12  15  20  31  25  22  29  18 103]]
```

Fig. 6. Performance for original (up) and reduced (down) data

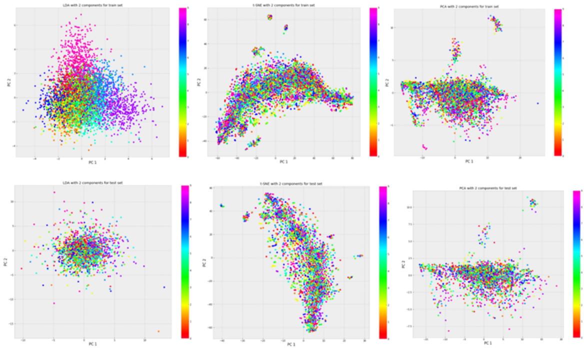


Fig. 7. 2-D projection for LDA, t-SNE and PCA (left to right) in training (up) data and test (down) data

D. Performance of data reduced by MDS, Isomap, LLE

In the visualization and interpretation (Fig.8) and performance table (Table.1). We can see that LLE with number of components 8 only preserves local data structure thus has a bad representation with low accuracy and not able to visually identify the feature it preserve. For MDS, its number of components is 6. We can visually identify that digit in first two dimensions tend to have bigger size in the left and smaller size in the right with thicker edges presented in the diagonal area. However, the use of euclidean distance makes it difficult to correctly unfold the manifold thus lead to a bad accuracy. As for Isomap, its number of components is 17. it makes use of geodesic distance and thus be able to properly unfold the manifold with better representation and higher accuracy. As a result, Isomap is a better manifold learning algorithm in this problem.

ACKNOWLEDGMENT

The authors would like to thank Dr. Catia Silva for providing this opportunity to experiment Machine Learning dimen-

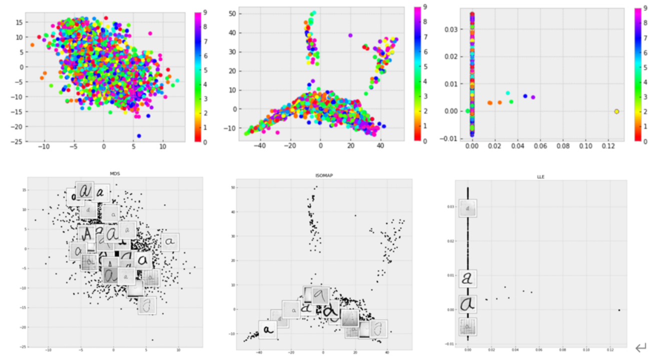


Fig. 8. Visualization and Interpretation of first 2-D for train set. MDS, Isomap, LLE (left to right)

TABLE I
PERFORMANCE WHEN USING MDS, ISOMAP, LLE

Manifold Learning	Accuracy
MDS	Training set: 9.43 percent, Test set: 11.70 percent
Isomap	Training set: 19.84 percent, Test set: 16.70 percent
LLE	Training set: 11.65 percent, Test set: 10.35 percent

sionality reduction methods.