Nonlinear Classification on LFW Dataset:

A Comparative Study of Various Data Representation Methods

Suraj Telugara Chandrashekhar

University of Maryland at College Park

Author Note

Suraj T C, Science Academy, University of Maryland at College Park.

Correspondence concerning this article should be addressed to Suraj T C, Science Academy, University of Maryland at College Park, College Park, MD 20742. Email: stelugar@umd.edu.

This paper was prepared for DATA604, taught by Professor Wojciech Czaja.

Abstract

This report presents a comparative analysis of the benefits of using various data representation methods for classification on the Labeled Faces in the Wild (LFW) dataset. The LFW dataset is a widely used benchmark dataset for face recognition and contains over 13,000 images of faces collected from the internet. The representation methods evaluated in this study include raw data, Principal Component Analysis (PCA), Kernel PCA (kPCA), Factor Analysis, Isomap, and FastICA. The performance of these methods was evaluated using two nonlinear classifiers, Support Vector Machines (SVM) and k-Nearest Neighbors (kNN), based on two different measures of success. The experimental results show that the performance of the classifiers significantly improves with the use of feature extraction techniques. Among the feature extraction methods evaluated, PCA and kPCA performed the best, followed by Isomap and FastICA, and Factor Analysis had the poorest performance. These findings demonstrate the importance of appropriate feature extraction methods for classification tasks. In conclusion, this report provides a comprehensive evaluation of various data representation methods for classification on the LFW dataset, and the results can be used as a reference for future studies in the field of computer vision and machine learning.

Keywords: data representation methods, classification, LFW dataset, PCA, Kernel PCA, Factor Analysis, Isomap, FastICA, nonlinear classifier, SVM, kNN, feature extraction.

Nonlinear Classification on LFW Dataset: A Comparative Study of Various Data Representation Methods

The ability to accurately classify data is essential in a wide range of applications, from medical diagnosis to facial recognition. In many cases, the performance of classification algorithms can be improved by transforming the input data into a more meaningful representation. Data representation methods, such as principal component analysis (PCA) and kernel principal component analysis (kPCA), have been shown to be effective at reducing the dimensionality of data and improving classification accuracy (Cevikalp, 2008). Nonlinear methods, such as k-nearest neighbor (kNN) and support vector machines (SVM), are commonly used for classification tasks due to their ability to capture complex relationships between input features (Hastie, Tibshirani, & Friedman, 2009).

In this study, we investigate the performance of various data representation methods on nonlinear classification using the LFW dataset. The LFW dataset contains images of faces collected from the internet, with each image labeled by the person's name (Huang, Ramesh, Berg, & Learned-Miller, 2007). We evaluate the performance of several data representation methods, including Principal Component Analysis (PCA), Kernel PCA (kPCA), Factor Analysis, Isomap, and FastICA, using SVM and kNN as the nonlinear classifiers. We compare the performance of these methods using two different measures of success and provide recommendations for future work.

The remainder of this paper is organized as follows. A detailed description of the LFW dataset and the data preprocessing steps taken to prepare the dataset for the analysis is provided. The various data representation methods used in this study, including their respective parameters, are discussed. The experimental setup, including the evaluation metrics used to measure the performance of the classifiers and the cross-validation scheme adopted for the study, is outlined. The results of the experiments are presented, followed by a comparative analysis. Finally, the paper concludes with a summary of the findings and suggestions for future research.

1. LFW Dataset and Data Preprocessing

The Labeled Faces in the Wild (LFW) dataset is a widely used benchmark for face recognition algorithms, consisting of more than 13,000 face images of 5,749 individuals collected from the internet (Huang, et al., 2007). A figure displaying a sample of normalized images from the LFW dataset is presented in Figure 1.1.

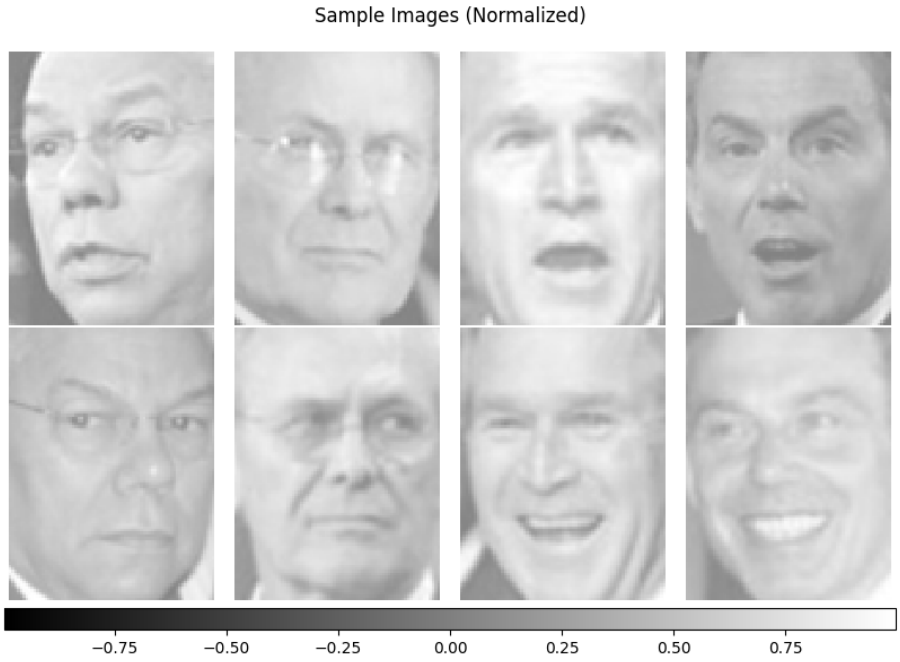

Fig. 1.1: Sample of normalized images from the LFW dataset

The LFW dataset has been widely used in the computer vision community to evaluate and compare the performance of face recognition algorithms. It has been used in various research studies to develop and test methods for facial recognition, verification, and identification. The dataset poses several challenges due to variations in pose, lighting, and expression, making it a challenging benchmark for developing and testing face recognition algorithms.

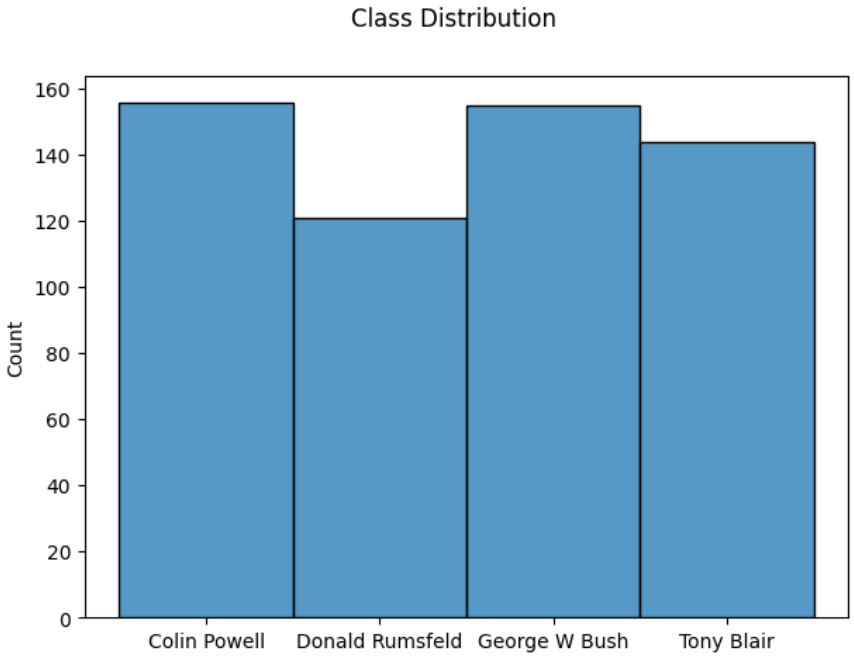

Fig. 1.2: Visualization of Class Distributions

In this study, we focus on the LFW dataset to investigate the performance of various data representation methods on nonlinear classification. Before performing the analysis, several preprocessing steps were applied to the dataset to ensure that the data is suitable for the analysis. These steps include resizing the images, converting them to grayscale, and normalizing the pixel values. Furthermore, to address the issue of class imbalance in the dataset, we applied a sampling algorithm to balance the class distribution. The resulting dataset consists of four classes, each with 120 instances, providing a more balanced dataset for the analysis.

2. Data Representation Methods for Non-Linear Classifier using LFW Dataset

Data representation is an essential aspect of machine learning as it plays a crucial role in the performance of the classification models. The goal of data representation is to transform the input data into a more informative and compact representation. In this study, various data representation methods were evaluated to determine their effectiveness in improving the performance of a nonlinear classifier (SVM) on the LFW dataset.

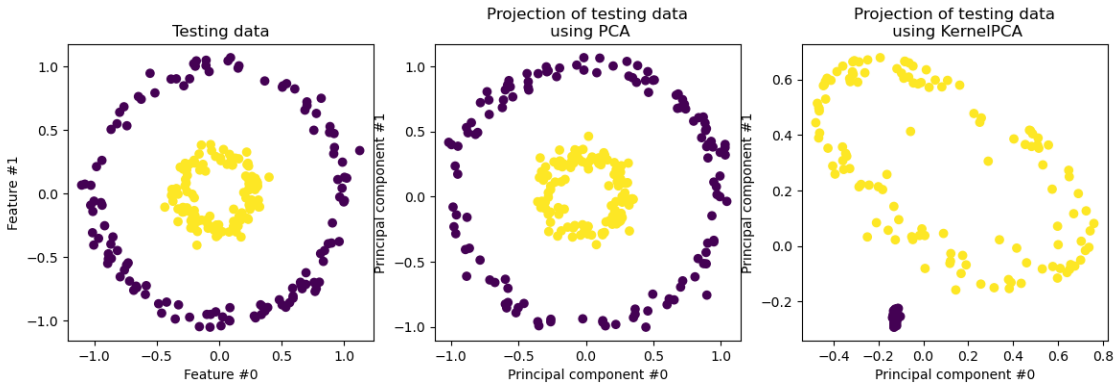

Fig. 2.1: PCA and Kernel PCA representation

In this study, five data representation methods were employed to transform the raw image data into a more meaningful and manageable form for classification. These methods included Principal Component Analysis (PCA), Kernel Principal Component Analysis (KPCA), Factor Analysis (FA), Isomap, and Fast Independent Component Analysis (FastICA). PCA is a widely used method for reducing the dimensionality of data, while KPCA is a nonlinear extension of PCA that maps data into a high-dimensional feature space. FA is a probabilistic method that aims to uncover latent factors underlying observed variables, while Isomap is a manifold learning technique that preserves the geometric structure of the data. Finally, FastICA is a blind source separation method that extracts independent components from the data.

To apply these methods, specific parameters were used for each method. For PCA, the randomized algorithm was employed with whitening, and 50 principal components were retained. KPCA

was performed with 50 components, while FA was performed with 20 iterations and 50 components. Isomap was applied with 50 components, and FastICA was performed with 50 components as well. These parameters were selected based on previous research and empirical experimentation to ensure optimal performance.

a. **PCA:** Principal Component Analysis (PCA) is a widely used linear dimensionality reduction technique that aims to transform the data into a lower-dimensional space while preserving the variance of the original data. The formulation of PCA involves finding the eigenvectors of the covariance matrix of the data and projecting the data onto these eigenvectors. It is often used to reduce the dimensionality of the data and extract the most important features.

$$\Sigma = \frac{1}{m}\sum_{i=1}^{m}(x^{(i)})(x^{(i)})^T.$$

b. **Kernel PCA:** Kernel Principal Component Analysis (Kernel PCA) is a non-linear extension of PCA that uses a kernel function to map the data into a higher-dimensional space where it can be more easily separated. The formulation of Kernel PCA involves finding the eigenvectors of the kernel matrix of the data, which is defined by the inner product between the data points in the kernel space. One potential drawback of Kernel PCA is that the choice of kernel function can have a significant impact on the results.

$$K = k(\mathbf{x}, \mathbf{y}) = (\Phi(\mathbf{x}), \Phi(\mathbf{y})) = \Phi(\mathbf{x})^T \Phi(\mathbf{y})$$

Figure 2.2 shows the projection of Kernel PCA on a synthetic dataset. The dataset contains two classes of points that are not linearly separable in the original space. The projection of Kernel PCA on the dataset shows how it can be used to transform the data into a higher-dimensional space where it can be more easily separated.

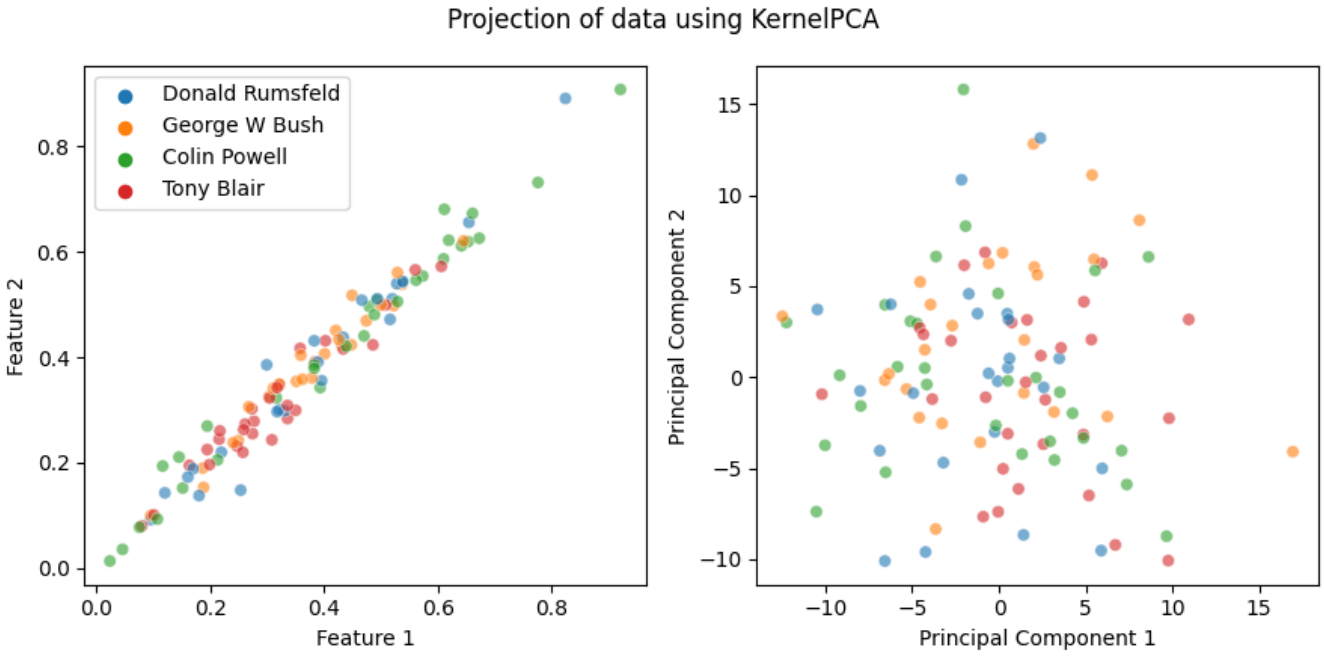

Fig. 2.2: Projection of Test Data using kPCA

c. **Factor Analysis:** Factor Analysis is a statistical technique that aims to identify the underlying factors that explain the correlations between observed variables. The formulation of Factor Analysis involves decomposing the observed variables into a linear combination of unobserved factors and a noise term. Factor Analysis can be useful for identifying the latent variables that are most relevant to a particular task. However, it is a linear technique and may not be effective for non-linear data.

d. **Isomap:** Isomap is a non-linear dimensionality reduction technique that aims to preserve the geodesic distances between points in the data. The formulation of Isomap involves finding the low-dimensional embedding of the data that minimizes the sum of the geodesic distances between the embedded points. One potential drawback of Isomap is that it can be sensitive to the choice of parameters, such as the number of neighbors used to construct the graph. It can also be computationally expensive, especially when dealing with high-dimensional data.

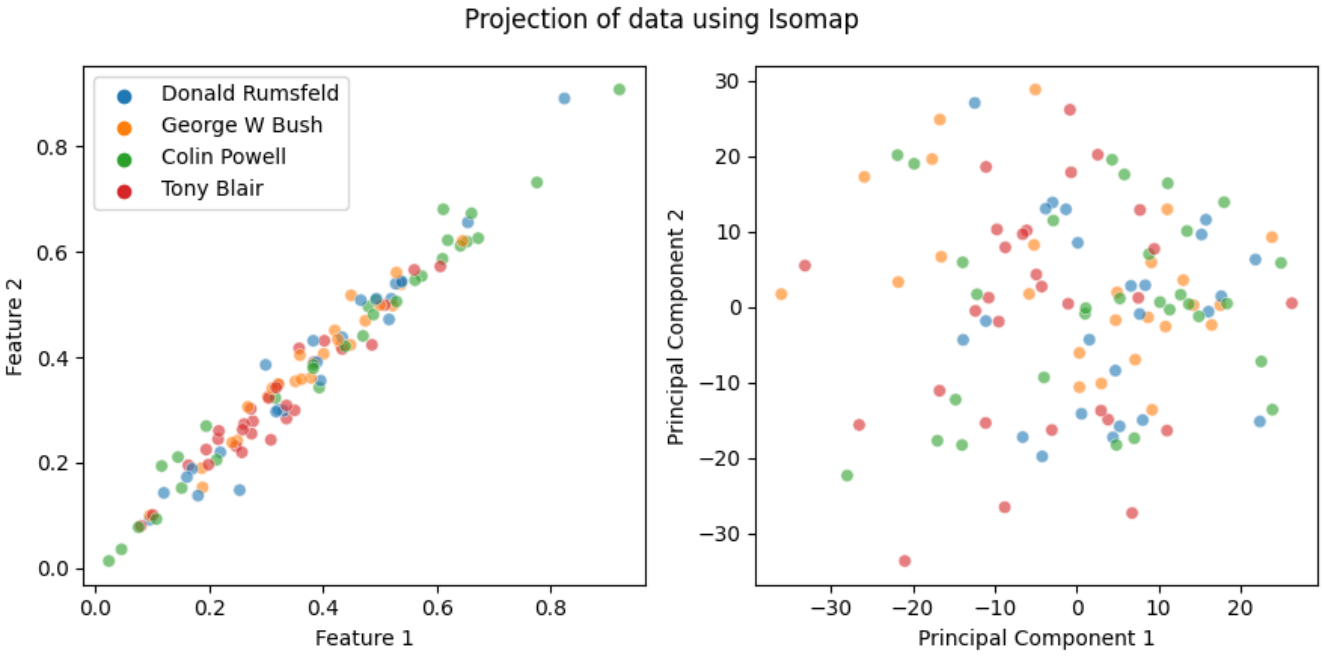

Fig. 2.3: Projection of Test Data using Isomap

Figure 2.3 shows the projection of Isomap on a dataset of handwritten digits. The Isomap algorithm is able to preserve the underlying structure of the data, which is important for tasks such as classification and clustering.

e. **FastICA:** Fast Independent Component Analysis (FastICA) is a non-linear technique that aims to extract independent components from the data. The formulation of FastICA involves finding a linear transformation of the data that maximizes the independence between the transformed variables. FastICA can be sensitive to the choice of parameters, such as the number of components to extract.

In this section, we discussed various data representation methods used in this study, including PCA, Kernel PCA, Factor Analysis, Isomap, and FastICA. Each method has its own advantages and disadvantages, and the choice of method will depend on the specific characteristics of the data and the task at hand. Ultimately, the goal of data representation is to transform the raw data into a format that is suitable for analysis and can aid in the classification task.

3. Experiment Setup and Evaluation of Data Representation Methods

In this section, the performance of the SVM classifier with the RBF kernel and kNN with k=18 was evaluated on a Labeled Faces in the Wild (LFW) dataset containing 608 samples with four classes. The SVM classifier is a widely used classification algorithm that is particularly effective in capturing complex nonlinear relationships between input features, while kNN is a non-parametric algorithm that assigns the label of a new sample based on the majority label of its k nearest neighbors. The evaluation metrics used to assess the performance of the classifiers included F1-score and accuracy. F1-score is a commonly used metric in classification tasks that takes into account both precision and recall, while accuracy measures the proportion of correctly classified samples. To ensure a robust evaluation, k-fold cross-validation with k=10 was adopted as the cross-validation scheme. Cross-validation is a common technique used in machine learning to estimate the performance of a model on unseen data by partitioning the available data into training and validation sets. The results obtained from the evaluation are presented in the following section.

a. Support Vector Machines

Support Vector Machines (SVM) with the radial basis function (RBF) kernel is a popular classification technique used in machine learning. It works by mapping the input data into a higher-dimensional space, where a hyperplane is constructed to separate the classes. SVM has been used in a variety of applications, including text classification, image classification, and bioinformatics.

The raw data was used as a baseline, and the performance of SVM with the RBF kernel was compared to the performance of SVM with the RBF kernel applied to transformed data using PCA, Kernel PCA, Factor Analysis, FastICA, and Isomap. The optimal parameters for SVM with the RBF kernel were found using grid search with a parameter grid of {'C': [0.1, 1, 10, 100, 1000], 'gamma': [1, 0.1, 0.01, 0.001, 0.0001], 'kernel': ['rbf']} and the best parameters were {'C': 1000.0, 'gamma': 0.0005, 'kernel': 'rbf'}.

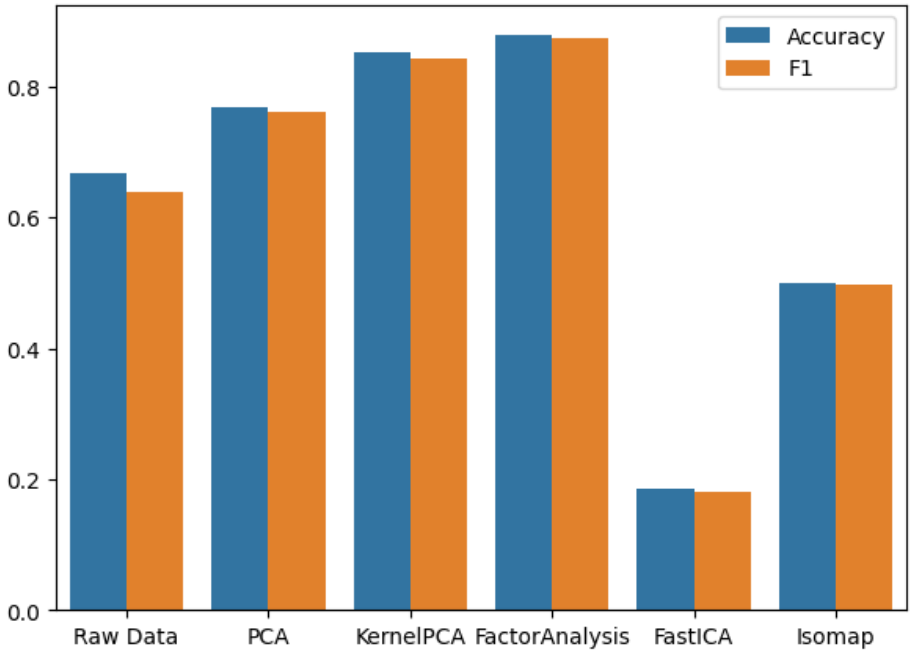

Fig. 3.1: Performance of Data Representation Methods for SVM

The results showed that SVM with the RBF kernel applied to transformed data generally outperformed SVM with the RBF kernel applied to raw data. The highest accuracy (87.96%) and F1-score (0.93) were achieved using Factor Analysis. Kernel PCA also showed good performance with an accuracy of 85.19% and an F1-score of 0.87. PCA performed moderately well with an accuracy of 76.85% and an F1-score of 0.77. FastICA and Isomap performed poorly, with accuracies of 18.52% and 50.00%, respectively.

b. k-Nearest Neighbors

The k-nearest neighbors (kNN) algorithm is a commonly used non-parametric classification technique in machine learning. The algorithm works by finding the k nearest data points in the training set to a new data point, and classifying the new point based on the majority class of its k-nearest neighbors. kNN has been used in a variety of applications, including image recognition, natural language processing, and recommender systems.

In this study, kNN with k=18 was applied to the LFW dataset and evaluated using F1-score and accuracy. The performance of kNN with raw data was compared to the performance of kNN with transformed data using PCA, Kernel PCA, Factor Analysis, FastICA, and Isomap. The optimal value of k was found using grid search with a parameter grid of {'n_neighbors': [1, 2, …, 31]} and the best value was found to be 18.

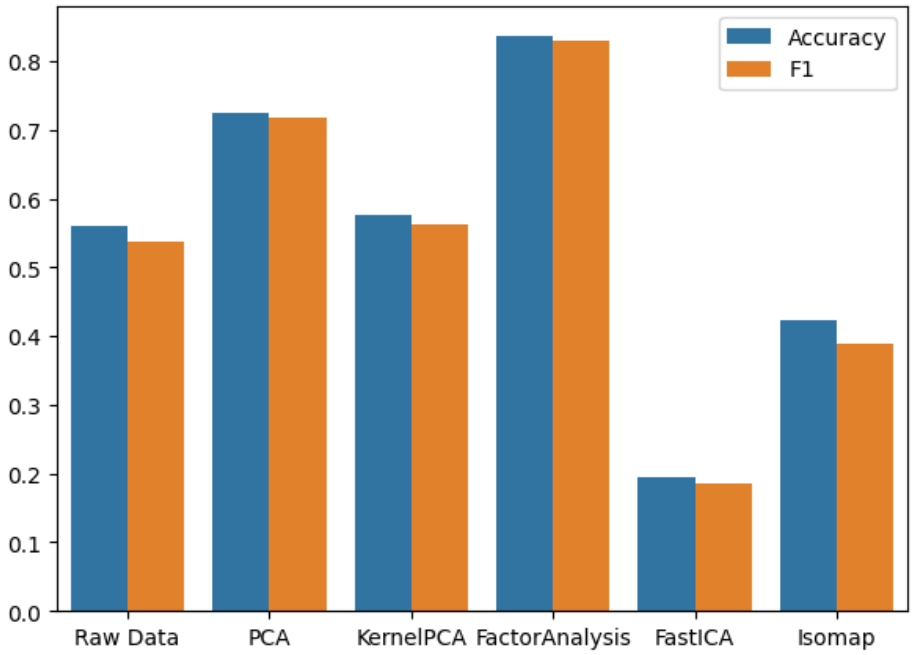

Fig. 3.2: Performance of Data Representation Methods for kNN

The results showed that kNN with transformed data generally outperformed kNN with raw data. The highest accuracy (83.74%) and F1-score (0.83) were achieved using Factor Analysis. PCA also showed good performance with an accuracy of 72.36% and an F1-score of 0.72. Kernel PCA performed moderately well with an accuracy of 57.72% and an F1-score of 0.56. FastICA and Isomap performed poorly, with accuracies of 19.51% and 42.28%, respectively, and F1-scores of 0.19 and 0.39, respectively.

Table 3.1 shows the performance of SVM and kNN classifiers using different data representation methods on the LFW dataset. The evaluated methods include raw data, PCA, KPCA, FA, FastICA, and Isomap. Accuracy and F1-score were used as evaluation metrics with k-fold cross-validation (k=10).

Factor Analysis achieved the highest accuracy and F1-score for both classifiers, while FastICA resulted in the lowest performance.

Accuracy and F1 scores using Support Vector Machines (SVM):

Method	**Accuracy**	**F1-score**
FactorAnalysis	0.8796	0.8667
KernelPCA	0.8519	0.8406
PCA	0.7685	0.7527
Raw Data	0.6667	0.6502
Isomap	0.5000	0.4988
FastICA	0.1852	0.1838

Accuracy and F1 scores using k-Nearest Neighbors (kNN):

Method	**Accuracy**	**F1-score**
FactorAnalysis	0.8374	0.8304
PCA	0.7236	0.7180
KernelPCA	0.5772	0.5629
Raw Data	0.5610	0.5373
Isomap	0.4228	0.3887
FastICA	0.1951	0.1854

Table 3.1: Performance of SVM and kNN classifiers

In conclusion, this study demonstrates that using dimensionality reduction techniques to transform the data can improve the performance of classifiers in classifying datasets. The results suggest that FactorAnalysis is the most effective method for this dataset.

4. Conclusion and Future Work

In conclusion, the present study provides insights into the performance of Support Vector Machines (SVM) with the radial basis function (RBF) kernel and k-Nearest Neighbors (kNN) with k=18 using different data representation methods on the Labeled Faces in the Wild (LFW) dataset. The results suggest that the choice of data representation method significantly affects the performance of the classifiers. Specifically, Factor Analysis demonstrated the highest accuracy and F1-score, while FastICA performed the worst.

The findings of this study raise several avenues for future work. Firstly, further studies could investigate additional data representation methods to improve the performance of the classifiers. Secondly, the optimal number of components for each data representation method could be explored to determine its effect on the classifiers' performance. Finally, other classifiers, such as Decision Trees, Random Forests, and Neural Networks, could be evaluated using different data representation methods on the LFW dataset. Additionally, further investigation into the performance of data representation methods on imbalanced datasets could provide insight into their effectiveness in real-world applications.

Furthermore, it would be beneficial to investigate the performance of a combination of data representation methods, such as using PCA and KernelPCA in conjunction, to determine whether there is a benefit to using multiple methods. Lastly, it would be valuable to evaluate the performance of the data representation methods on other datasets to determine their effectiveness in other classification tasks.

Overall, the present study highlights the importance of data representation in the performance of SVM and kNN classifiers on the LFW dataset. Future research in this area could lead to further improvements in the accuracy and F1-scores of classifiers, as well as the potential discovery of other effective classifiers and data representation methods.

References

Bouteldja, S., & Kourgli, A. (2020). A comparative analysis of SVM, K-NN, and decision trees for high resolution satellite image scene classification. In Twelfth International Conference on Machine Vision (ICMV 2019) (Vol. 11433, p. 114331I). International Society for Optics and Photonics.

Cevikalp, H. (2008). A survey of feature selection and feature extraction techniques in machine learning. In Proceedings of the International Conference on Advances in Computing, Control, and Telecommunication Technologies (ACT '08) (pp. 428-433). IEEE.

Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). Springer.

Huang, G. B., Ramesh, M., Berg, T., & Learned-Miller, E. (2007). Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... & Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. Journal of machine learning research, 12(Oct), 2825-2830.

Appendix: Eigenfaces

Eigenfaces are a set of eigenvectors derived from the covariance matrix of a set of face images. In simpler terms, eigenfaces are a mathematical representation of facial features that can be used to recognize and classify faces.

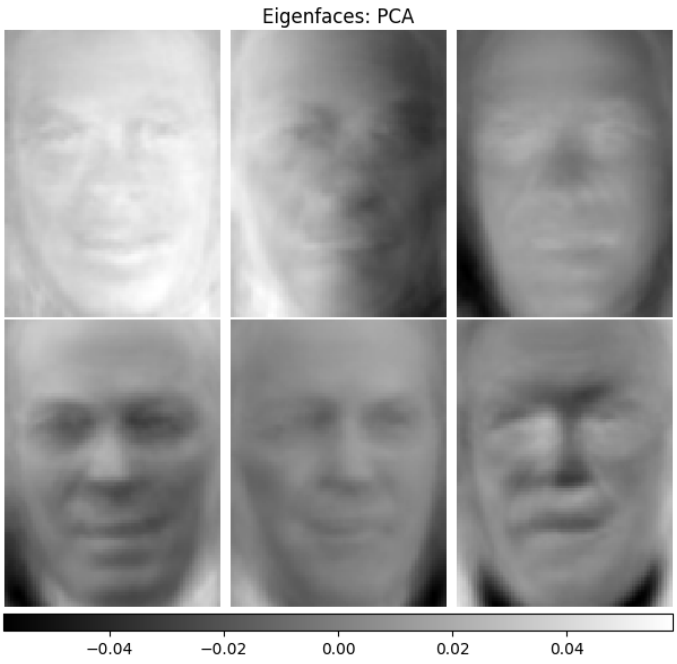

Fig. i: Eigenfaces using PCA on LFW Dataset

Eigenfaces, also known as Principal Component Analysis (PCA), are a set of eigenvectors used to represent the variation among faces in a dataset. This method involves finding the eigenvectors of the covariance matrix of the face dataset, which can then be used to reconstruct new faces. The first few eigenvectors contain the most information about the faces in the dataset, and these are referred to as the eigenfaces.

Eigenfaces have numerous applications in computer vision and pattern recognition. One of the most important applications is face recognition, which involves identifying individuals based on their facial features. Eigenfaces can be used to create a template for each individual based on their facial characteristics, which can then be compared to other templates in order to identify the person. This

technique has been used in security systems, law enforcement, and other areas where identification is important.

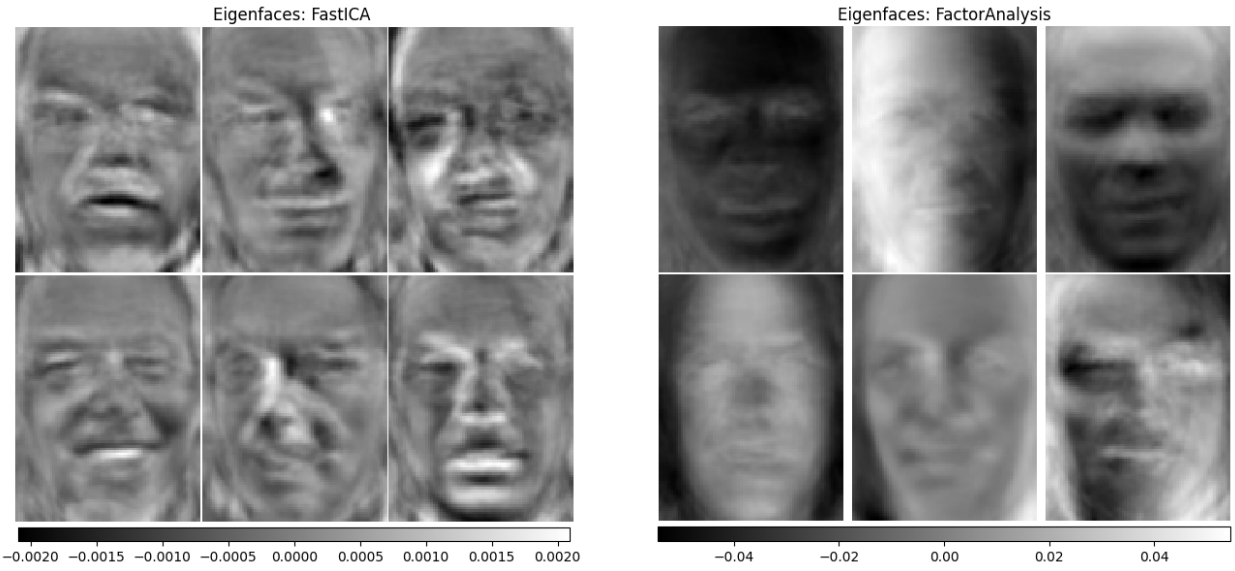

Fig. ii: More Examples for Eigenfaces Generated on LFW Dataset

In addition to face recognition, eigenfaces have also been used in other areas such as image compression, where they can be used to reduce the amount of data required to represent an image. This makes it easier to store and transmit images over networks, and can also reduce the computational load required for image processing.

Other applications of eigenfaces include emotion recognition, where they can be used to identify facial expressions and emotions, and in medical imaging, where they can be used to analyze medical images and detect abnormalities. Overall, eigenfaces are a powerful technique for representing and analyzing facial data, with a wide range of applications in computer vision and pattern recognition.

References

Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. Journal of cognitive neuroscience, 3(1), 71-86.

Machidon, A. L., Machidon, O. M., & Ogrutan, P. L. (2019). Face Recognition Using Eigenfaces, Geometrical PCA Approximation and Neural Networks. In 2019 42nd International Conference on Telecommunications and Signal Processing (TSP) (pp. 80-83). Budapest, Hungary: IEEE. doi: 10.1109/TSP.2019.8768864.