DEPARTMENT OF COMPUTER SCIENCE, GEORGIA INSTITUTE OF TECHNOLOGY

Unsupervised Learning and Dimensionality Reduction

Joseph Su

January 6, 2019

1 INTRODUCTION

This paper considers two unsupervised clustering algorithms, k-mean clustering and Expectation Maximization, and the following dimensionality reduction algorithms: Principal Components Analysis (PCA), Independent Components Analysis (ICA), Randomized Projections (RP), and Feature Agglomeration (FA). Two datasets consisting of well-known, fixed clusters of faces and handwritten digits are adopted. We explore and apply the aforesaid algorithms to the datasets, and compare and contrast the results with respect to each algorithm's theoretical origins and empirical studies in literature. Lastly we choose our wines dataset from our prior assignment and pre-process it using the same clustering and reduction algorithms. We apply a convolutional neural network to this dataset to assess its learned effects.

COMPUTING FACILITIES This work makes use of various libraries, toolsets, and modules in Python to arrive at its studies and results: `scikit-learn 1.19.dev0`, `sk-nn`, [theano], `numpy`, `scipy`, `IPython`, and `panda`. All of the experiments are conducted on a 2015 Mac OSX 2.5 GHz Intel Core i7 with 16GB DDR3.

2 DATA

DIGITS DATASET This dataset comes from the well-known MNIST Database of handwritten digits [1], which consist of images 0 to 9. Fig. 2.1 displays the first 5 original and normalized digits, each having 8-by-8 grayscale pixels in a combined feature dimension of 64. Each input instance, x, is projected into a new dimensional vector space, $\mathbb{R}^{s\times f}$ where $s = 1791$ samples and $f = 64$ features. The target label is an one-hot-encoded bit string vector, $y \in [0\cdots 9]$, with the i-th label having 1 at the i^{th} bit of the string and 0 throughout.

FACES DATASET This dataset comes from the Database of Faces at AT&T Laboratories [2]. This dataset consists of 10 facial images from each of the 40 participants. Fig. 2.1 displays the first 5 original and normalized faces of one participant, with each image having 64-by-64 grayscale pixels in a combined feature dimension of 4096. Similar to the digits dataset each input, x, is transformed into a new space, $\mathbb{R}^{s\times f}$ where $s = 400$ samples and $f = 4096$ features. The target labels are one-hot-encoded in bit vectors, $y \in [0\cdots 39]$.

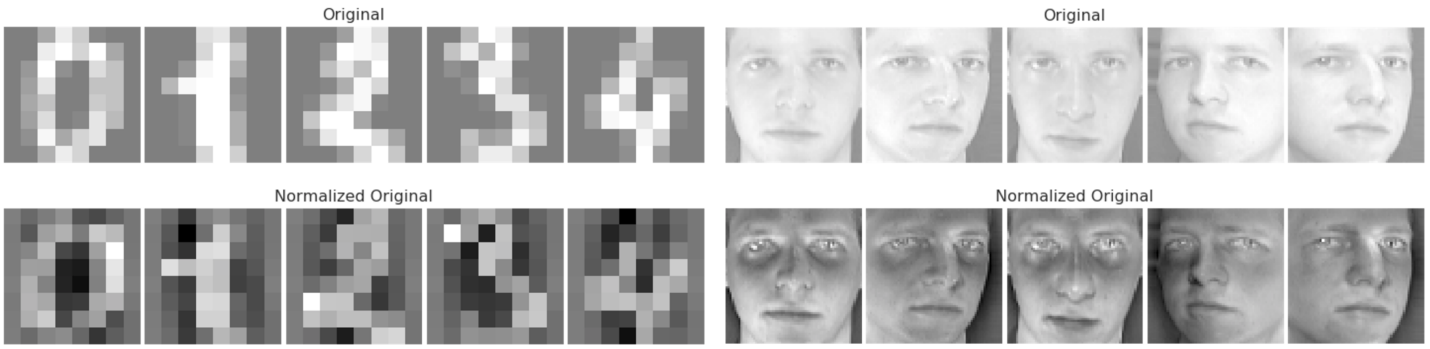

Figure 2.1: Digits Dataset with 10 Labels and Faces Dataset with 40 Labels

WINES DATASET This dataset comes from the prior assignment, retrieved from UCI's Wine Database with continuous, real-valued and multi-class features [3]. There are 13 features, a good enough fitting entry point to admit decent, expedient, exploratory analysis and comparisons of the algorithms in cross validating and tuning their efficacies. The dimension space for the Wine dataset is $\mathbb{R}^{s\times f}$, where $s = 178$ and $f = 13$. The classification labeling is multi-class, with each wine being one of the three types.

3 CLUSTERING METHOD: k-MEANS

k-means clustering is a method of vector quantization [4]. With a set of observations $(x_1, x_2, \cdots, x_n)$, we aim to partition them into $k \leq n$ sets $\mathbf{s} = (\mathbf{s_1}, \mathbf{s_2}, \cdots, \mathbf{s_k})$, minimizing the in-cluster sum of squares distance of each observation in the cluster to the k center. The objective is thus

$$\arg\min_{\mathbf{s}} \sum_{i=1}^{k} \sum_{x\in S_i} ||x - \mu_i||^2 \tag{3.1}$$

where μ_i is the mean of observations in the $\mathbf{s}$ set. We apply `scikit-learn`'s `k-means++` to optimally initialize the cluster centers, iterating the process 10 times with different centroid seeds and outputting the best run in terms of its inertia, or the sum of distances of samples to their closest cluster center.

MEASURES OF SIMILARITY/DISTANCE `k-means++` uses Euclidean distance metric, which works best when datasets are normally distributed and their distances linearly measurable. We apply several additional metrics, elbow and quality, to assess the goodness of fit of the cluster labels in relations to the ground truth.

k SELECTION ANALYSIS Elbow is an information criterion often used in data k-clustering to select k. We obtain the percentage of variance explained as a function of k. The first clusters gives the highest information gain which decreases along with increasing k. We employ the following quality metrics for added validation: homogeneity score, completeness score, v-measure, adjusted Rand index (ARI), and adjusted mutual information (AMI). A value of 1.0 generally connotes to a perfect fit (significant agreement) whereas 0.0 indicates dissimilarity or misalignment in a labeled dataset. In particular v-measure is computed as the harmonic mean of homogeneity and completeness measures.

4 CLUSTERING METHOD: EXPECTATION MAXIMIZATION

We employ `scikit-learn`'s `GaussianMixture`, which is based on EM involving an iterative process of model estimation and update. In the expectation "E" step, the algorithm computes for each data point a probability of being generated by each cluster; this is different than k-means where each data point is assigned a cluster label. In the maximization "M" step, the model's within-cluster parameters are updated to maximize the likelihood of the data given those assignments. EM always converges to a local optimum. For both datasets we

consider a univariate Gaussian mixture with G components, each having its own covariance matrix, making a Bayesian Information Criteria (BIC) estimate of G. We default our EM algorithm with 100 iterations, k-means to initialize weights, means and precisions, and a convergence threshold of 0.001.

5 Dimensionality Reduction Algorithms

We describe the four reduction algorithms, and their respective Python `scikit-learn` modules.

PCA This is a linear transformation algorithm to find the directions (eigenvectors) of maximum variance in highly dimensional data, projecting each of its input, $x \in \mathbb{R}^n$, into an orthogonal subspace with less dimensions $\mathbb{R}^m$, where $m << n$. We employ `PCA` and preprocess data with whitening to generate features that are less correlated but with the same variance.

ICA This is a fixed point algorithm using kurtosis computed over a large set of data to quickly converge our model, at a speed that is often 10 to 100 folds faster than other adaptive algorithms [5]. ICA selects features that are maximally independent from one another and their values form a non-Gaussian distribution. The objective is to minimize mutual information and maximize non-Gaussianity in the data. We employ `FastICA` and use it to survey the kurtotic landscape in the distribution to arrive at the best component used. We enable whitening, 200 iterations, and with a convergence threshold of 0.0001.

RP RP is introduced to speed up the projection computation for high-dimensionality datasets (n-dimensions) to their lower dimensional counterparts (m-dimensions) with sufficient accuracy [6]. With s samples, the projection is done via a random $n \times m$ matrix of unit-length columns and $n \times s$ data matrix into m dimensions. We apply `GaussianRandomProjection` for this task.

Feature Agglomeration This algorithm applies hierarchical clustering to aggregate data in the direction of feature similarity, by recursively extracting features that minimize the intra-class variance across clusters. `FeatureAgglomeration` in `sklearn`'s clustering module is applied.

6 Results

We present two categories of results herein:

- Unsupervised exploration of datasets with an ensemble of clustering and reduction algorithms.
- Supervised learning experiment on a much reduced and clustered wines dataset.

6.1 Clustering

k selection is performed for k-means. Two approaches, elbow and quality metrics, are taken. The former is a crude method involving visualizing the turning point where changes in variance ween off. The second involves quality assessment leveraging the domain knowledge (ground truth) from these datasets.

k Selection: Digits The elbow criterion for this data inconclusively conveys a range of 8 to 18 as observed in Fig.6.1. The assessment illustrates a few salient points. **Firstly**, homogeneity and completeness run in opposition of each other; increasing one decreases another, and vice versa. **Secondly**, scoring trends towards a convergence as k approaches 10. We confirm a similar trend with silhouette analysis. $k = 10$ for this dataset.

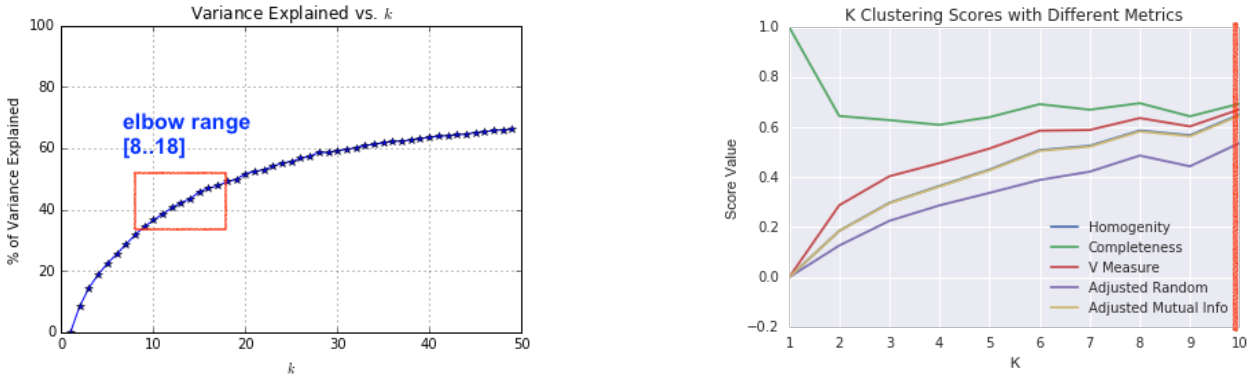

Figure 6.1: Digits: Elbow vs Quality Scores

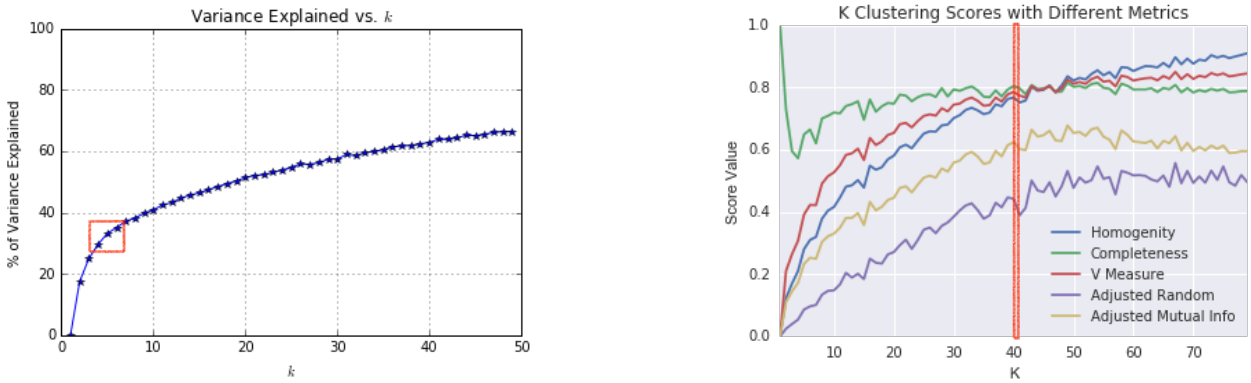

Figure 6.2: Faces: Elbow vs Quality Scores

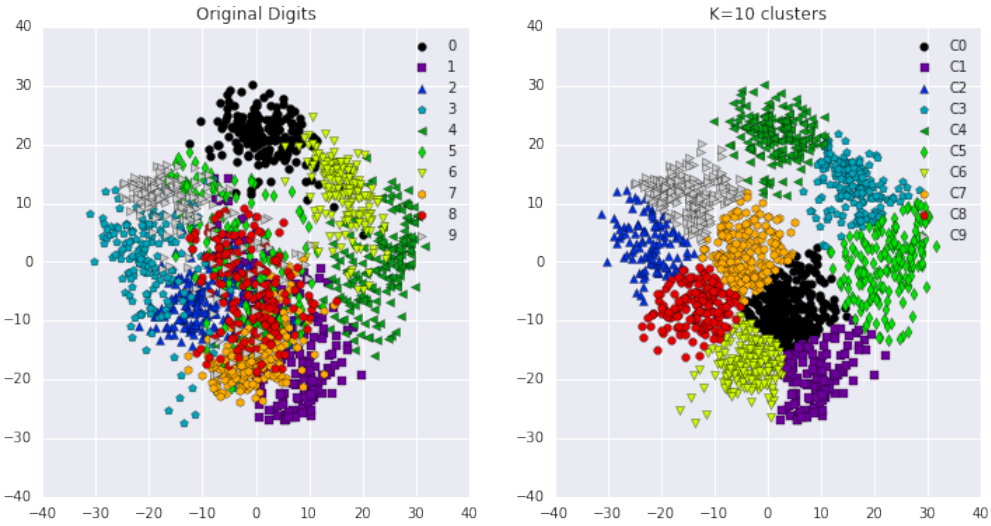

Figure 6.3: Digits: Original vs $k = 10$

Figure 6.4: Clustered Digits and Faces

k SELECTION: FACES The elbow criterion conveys a range of optimality, with k being 4 to 6 as seen in Fig.6.2. This result is inconclusive. With quality metrics, we observe scoring consistency and convergence as k crosses 40 to 42. For this dataset, $k = 40$. However, we choose $k = 10$ to ensure that forthcoming experiments yield the same number of clusters on both datasets, allowing us to remove the influence of k on the evaluation criteria.

k-MEANS AND EM Fig. 6.3 shows the original digit distribution vs. $k = 10$ distribution in 2D. One observes both the intra and inter-class relationships in a cluster of 10. Fig. 6.4 shows the clustered images and their respective training time for both datasets. The running time for EM is several-fold over k-means. EM yields visually different clusters than k-means' own.

6.2 DIMENSIONALITY REDUCTION

The first 5 images after running the various reduction algorithms are displayed in Fig. 6.5. The original images are included for comparison purposes. For PCA these images represent the first 5 principal components (PC).

PCA These PCs are eigenvectors sorted by the magnitude of their eigenvalues; instead of trending their values we generate their explained variances in Fig. 6.7, which shows both cumulative and individual variances with an increasing number of PCs. The first 20 components in either dataset explain approximately 80% of the variance. For faces, a fair amount of information gain levels off after 100+ components. To ascertain information loss and entropy gain visually, reduced datasets are reconstructed with PCA on their covariance matrices, yielding the results in Fig. 6.6. For digits reconstruction, we use the first 10 PCs that account for the 60% of the variance; for faces reconstruction, we use the first 30 PCs that account for the 83% of the variance.

ICA We survey the degree of non-Gaussianity across all components, observing both subgaussian and supergaussian components in our data. Kurtosis as a standardized fourth moment evaluates to 3 for gaussian. We maximize the least kurtotic component in our datasets by way of evaluating $max(abs(min(kurtosis-3)))$ across components. Fig. 6.8 shows ICA kurtosis vs components. For digits, the kurtotic distribution occurs at component = 11; for faces, the distribution occurs at 9. These values are used in our ICA experiments.

RP Projected data are shown in Fig. 6.5, which represents a culmination of several runs of RP. Reconstruction yields images with little resemblance to their original (Fig. 6.9).

FA Agglomerated data are shown in Fig. 6.5. Among all other reduction algorithms, FA takes the longest time to compute faces; the result is dismal (Fig. 6.5). FA is rather successful with digits.

6.3 DIMENSIONALITY REDUCTION + *k*-MEAN CLUSTERING ENSEMBLES

For brevity we display the results from running an ensemble of one of each reduction algorithms with k-means. For notation purpose, PCA -> k-means specifies that data is reduced with PCA, then clustered with

Figure 6.5: Reduced Digits & Faces Datasets: First 5 Components

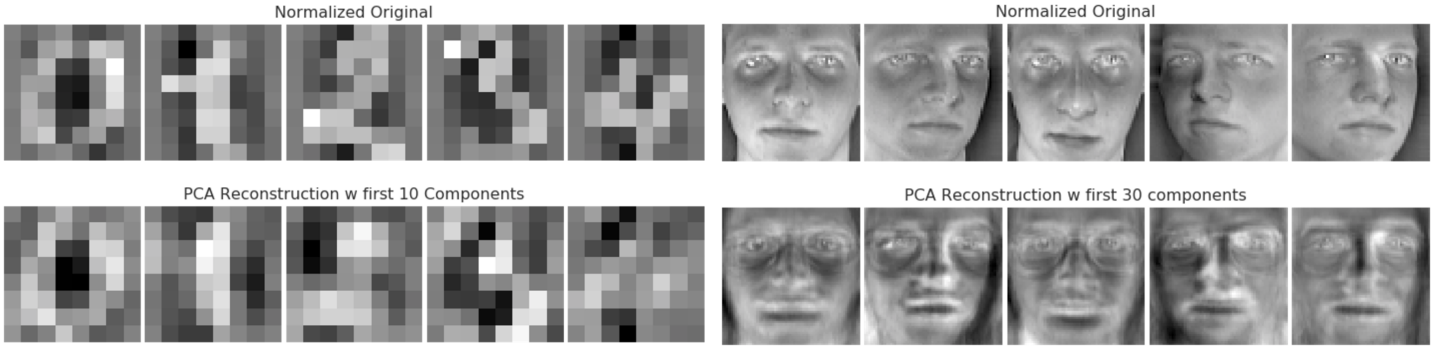

Figure 6.6: PCA Reconstruction vs Original

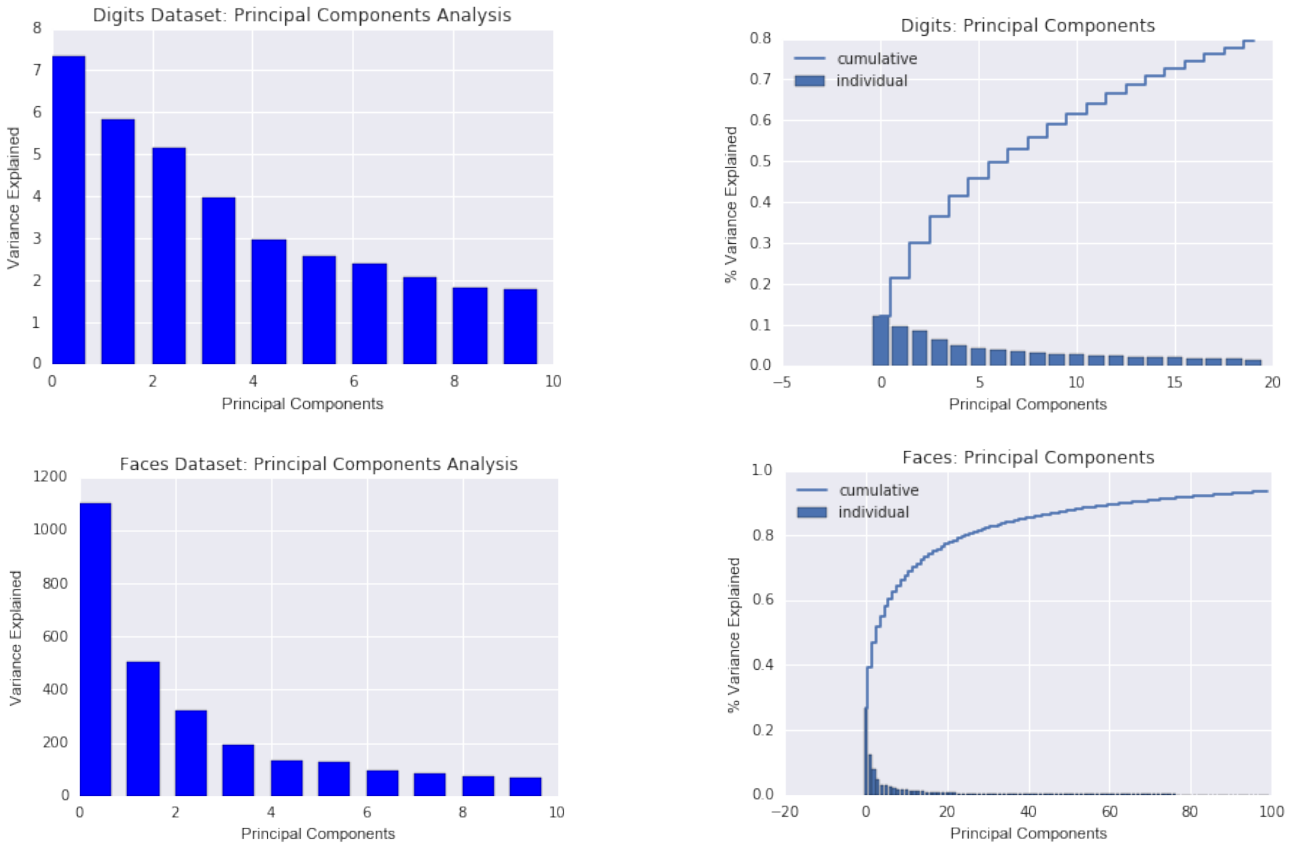

Figure 6.7: PCA Variance Explained: Digits vs Faces

k-means. We assess our results by looking at our inertia trends across all ensembles. Inertia, or within-cluster sum of squares, is to be minimized. Fig. 6.10 exhibits these trends for both datasets. Both k-means alone and RP->k-means yield the worst results, with ICA->k-means being the best indicator. Both PCA and FA assume the same negative slope as ICA. However, the former are at a much higher inertial baseline than the latter.

7 Supervised Learning: Neural Networks

For this experiment the wines dataset is used. Fig. 7.1 exhibits the effects of training time and $f1$ scoring, by applying clustering and dimensionality reduction on the dataset through a convolution neural network learner (CNN). We retain the same initializations used in the prior assignment for CNN with a 20/80 holdout and training split. Pre-processing the data with either clustering or dimension reduction techniques, or both,

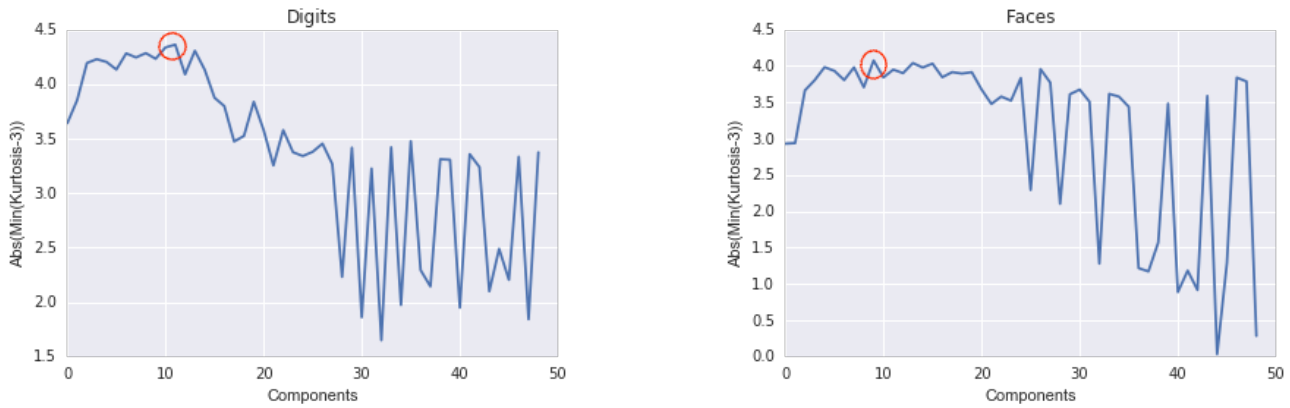

Figure 6.8: ICA Kurtosis vs Components

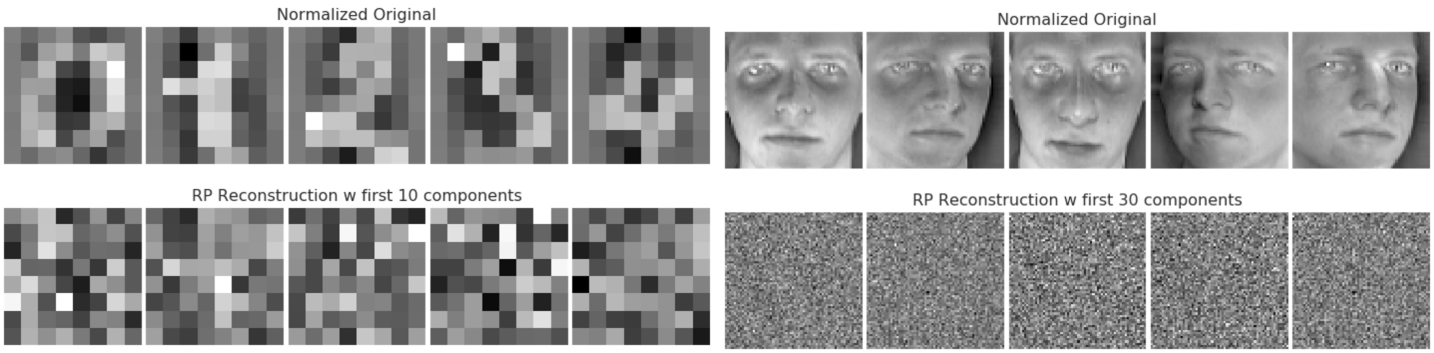

Figure 6.9: RP Reconstruction vs Original

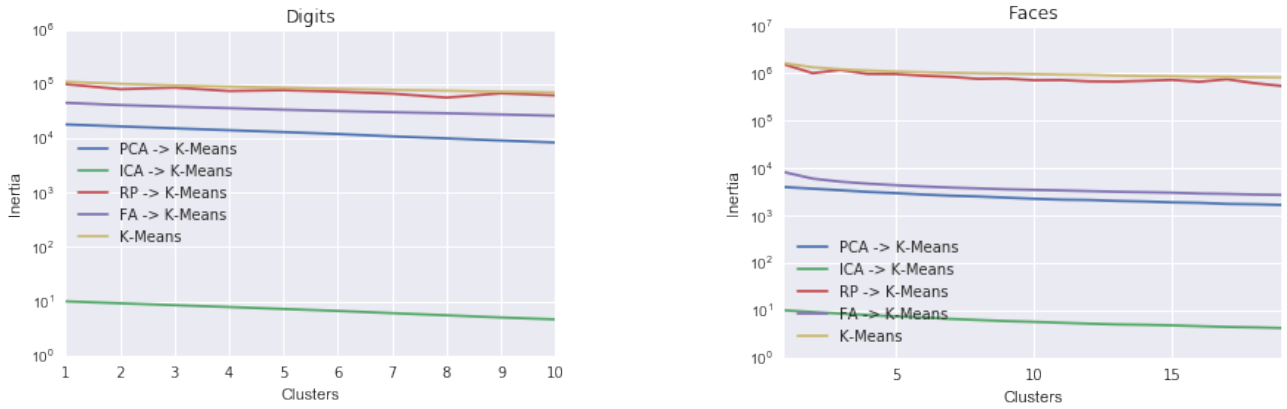

Figure 6.10: Inertia vs. Clustered and Reduced Data

yields overfitting. The best available scenario is when we do not apply any clustering and reduction algorithms to the data, as evident in Fig. 7.1. CNN alone yields the best time and scoring for both datasets. PCA has the second best $f1$, followed by ICA. However ICA takes the longest to process.

8 Discussions

We apply the elbow method on both normalized datasets and conclude that the method does not convey the optimal k very well. This means our source is loosely clustered as its variance explained is devoid of statistical significance.

Clustering k-means produces good, visually discernible clusters of digits such as 0, 3, 4, and 6 (Fig. 6.3). These digits exhibit high **intra − class** (e.g., digits 3 and 4) or low **inter − class** (e.g., digits 0 and 6) similarity with $k = 10$. We are cognizant of our k selection in this study. k-means is relatively efficient with a time of $O(skt)$, where s = # of samples, k= # of clusters, and t = # of iterations. EM is a Lloyds variant of k-means, but is more computationally intensive than the latter with no termination guarantee; EM has a time complexity of $O(p^2kt)$ and space of $O(sk)$, where p is the estimation of a probability that a data point is in cluster k. For digits, EM takes 10x longer than k-means. For faces, EM takes 100x longer than k-means.

Dimensionality Reduction As seen from Fig. 6.5, the 5 reduced images via PCA are sorted with the highest explained variance first. These images are the eigenvectors representing the directions in which the maximum amount of variations occurs, with the first capturing the main direction of such variations and subsequent ones being more complex and orthogonal to the ones prior. When dealing with a dataset with a cluster structure, most of the its between-cluster variance is captured by the first principal components. The digits dataset has a cluster structure corresponding to digits $0 \cdots 9$, therefore PCA yields eigenvectors that *approximately* correspond to these mixtures of cluster centroids. Thus our strongest set of principal components (PC) should

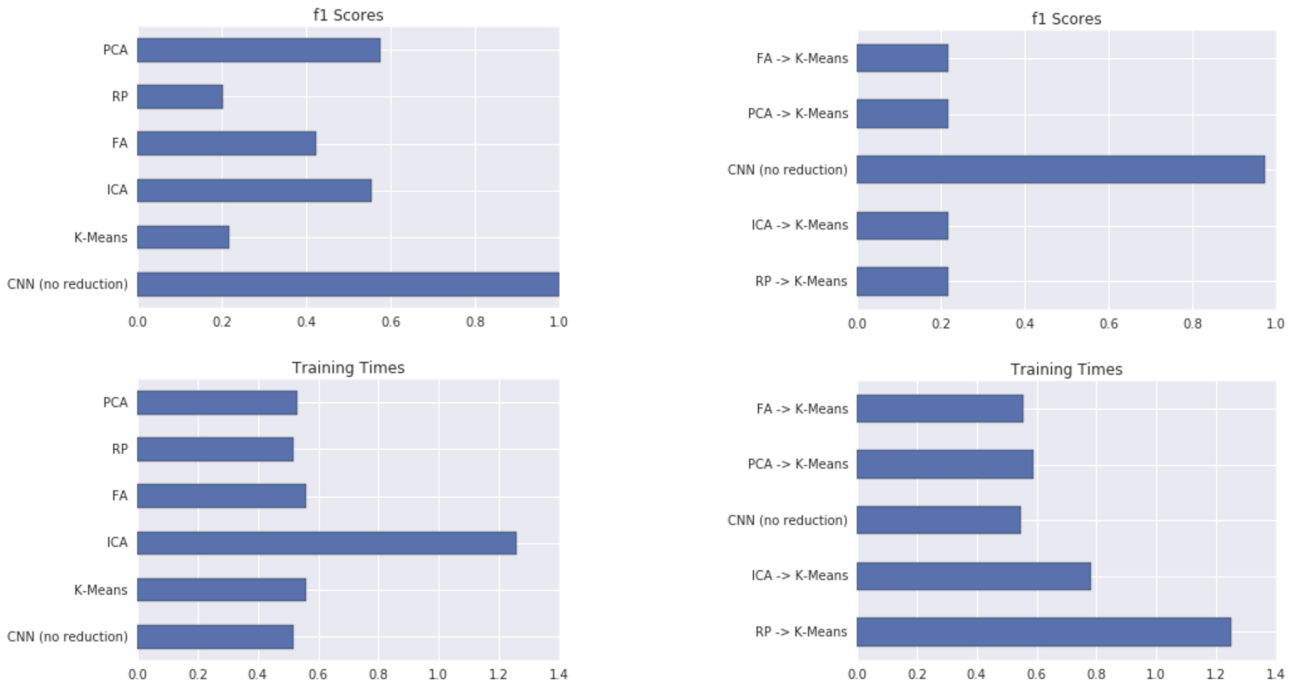

Figure 7.1: Convolutional Neural Network: Wines Dataset

"look" approximately, but not necessarily, like $0 \cdots 9$ in the order of their explained variances, as seen in Fig. 6.5. Zero-look component seems effective filtering digits such as 8, 9, and 6 which look like 0. Since the squared errors of projection are mean-square minimized in a m-dimensional space, PCA has a time complexity of $O(n^2 S) + O(n^3)$, where S is the # of samples and n is the dimensionality of input in $\mathbb{R}^n$.

One observes that ICA works on data with some sparsity. This is illustrated in Fig. 6.5 on faces, but not so much on digits which insinuate their high degree of gaussianity that does not work so well with the algorithm. ICA has a time complexity of $O(n(n+1)Si)$, where i is the # of iterations and the others are as defined. Empirically, ICA takes roughly 200-300% longer than PCA.

FA aggregates data in the direction of similar features, with an $O(n^2 log(n))$ time complexity. The agglomerated digits, however, bear extreme similarity to their original conveying the algorithm's efficacy on data with stronger inter-class affinity and higher intra-class distance.

RP is highly unstable and inconsistent as seen in Fig. 6.5; different random projects lead to very different clustering results. Repeated RP experiments yield sparse results. However RP's strength lies in its ability to preserve similarities of data vectors well. RP has a complexity of $O(nmS)$ after projecting data from n into m dimensions; thus time is linear and computation is among the fastest.

DIMENSIONALITY REDUCTION + CLUSTERING ENSEMBLES The best performance is with ICA->k-means. ICA maximizes the feature independence statistically, while both PCA and FA simplify the problem space through combining correlated components and features; such effect leads to a slight reduction in the inertia as seen in Fig. 6.10. RP, on the other hand, is almost as ineffective as performing only k-means clustering on raw data. One notes that projecting samples effectively without much distortion requires several thousands dimensions, agnostic of the number of features in a dataset [7]. The digits set is in $\mathbb{R}^{64}$; this dimensionality is too small for RP to be effective. When running RP on faces with a much higher dimensionality of $\mathbb{R}^{4096}$ over several runs, we observe its equally ineffective outcomes. This is in agreement with literature stating that RP is highly unstable and inconsistent; different random projects may lead to very different clustering results [8].

NEURAL NETWORK Clearly the best classification performance is achieved when no reduction or clustering is applied to the wines dataset. These added steps also increase a network's learning time as seen in Fig. 7.1, despite having a much reduced and more clustered data.

9 Conclusion

We explore and contrast various unsupervised learning algorithms with respect to their performance in reducing and clustering digital images. Both k-means and EM yield clusters with good intra and inter-class delineations. The latter produces non-deterministic results taking significantly more time than the former. k-means is clearly the superior choice if speed and output consistency are important.

Of the various algorithms studied, PCA reduces the complexity of the data quickly and can afford strong classifiers in unsupervised learning. However PCA does not do well with data having a large intra-class variation orthogonal to the inter-class variation and are non-linear in nature. Still other algorithms exist to reduce data dimensionality. FA is introduced to agglomerate data based on feature similarity; this method works well with our digits. ICA yields good results with data exhibiting strong nongaussianity; it appears that our faces dataset yields decent results with this method alone. RP is designed to work with data much higher in dimensionality than what ours call for, and such algorithm would be useful in data mining problems involving big data requiring expedient projections; none of our datasets fits this criteria. Unlike image reconstruction with RP, reconstruction works well with PCA, especially on digits with defined and delineated clusters.

Lastly, clustering and decomposition algorithms bring little, if any, intrinsic values to a neural network learner, which is quite sensitive to the undulating noises introduced to the network; ignoring such input noises may compromise the learner's efficacy. One concludes that removing components with insignificant variance, especially from a dataset that has a relatively small number of features (13 in wines), curtails a network's performance. Although in theory high dimensional input should be more ϵ regulated and scaled, we should be careful with how much aliasing to remove to improve the learned features.

References

[1] LeCun Y., Cortes, C., and Burges, C. J. C. *URL: http://yann.lecun.com/exdb/mnist/*. 2016.

[2] AT&T Laboratories. The Database of Faces. 2002.

[3] UCI Machine learning repository. Wine Data Set, UCI Machine learning repository *http://archive.ics.uci.edu/ml/datasets/wine)*. Web. 01 Sep. 2016.

[4] Lecture 6, Session 2003. Vector Quantization and Clustering. *https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-345-automatic-speech-recognition-spring-2003/lecture-notes/lecture6new.pdf*. 2003.

[5] X. Giannakopoulos, J. Karhunen, and E. Oja. Experimental comparison of neural ICA algorithms. *In Proc. Int. Conf. on Artificial Neural Networks (ICANN'98)*. pages 651-656, Skovde, Sweden, 1998.

[6] Bingham, Ella, and Heikki Mannila. Random projection in dimensionality reduction: applications to image and text data. *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2001.

[7] Wikipedia contributors. Johnson?Lindenstrauss lemma. *Wikipedia, The Free Encyclopedia. Wikipedia, The Free Encyclopedia*. Web. 10 Oct. 2016.

[8] Fern, X. Z., and Brodley, C. Random projection for high dimensional data clustering: A cluster ensemble approach. *ICML*. Vol. 3. 2003.