CS 7641 Machine Learning HW3

Hao-Jen Wang

Abstract

Unsupervised Learning Methods have many useful applications, such as classifying data into clusters, or enhancing the richness of features. I will focus on K-means and GMM for clustering and 4 dimensionality reduction methods: PCA, ICA, RCA, and LDA. Intuitions will be drawn based on the experimental results.

1. Introduction

In this project, I will evaluate 2 clustering methods and 4 dim reduction methods on the two datasets: Wisconsin breast-cancer dataset and MNIST hand-writing dataset. Both datasets are interesting to me, since I hope to provide solutions to improve modern medical analysis or street number classification one day.

2. Datasets

Wisconsin breast-cancer dataset consists of 569 samples in total, with binary labels and 30 distinct features. MNIST has 70,000 samples with 10 labels and 784 features. To evaluate, I split the breast cancer data into 455 training and 114 test, and MNIST into 60,000 for training and 10,000 for test. Standard Scalar is applied for the two datasets to avoid scalar issues.

3. Experiments

According to the hyper-parameters tuned from previous assignments, the Decision Tree (DT) yields 100.0% training and 91.23% test accuracies and overfits the breast cancer data. The neural network (NN) model yields 99.56% training and 95.61% test for breast cancer, and 98.99% training and 96.85% test for MNIST.

A. K-means – Breast Cancer

I will first run K-means on the training set and measure the optimal K for this data. With different K ranged from 1 to 200, I will measure the process time and performance metrics such as Silhouette score and Inertia. Since these two metrics have different range (Silhouette score is less than 1, whereas Inertia can be huge as 10^3) and the scale does not matter, I will normalize them by applying a min-max scalar for a better measurement. The result is shown in Figure 1.

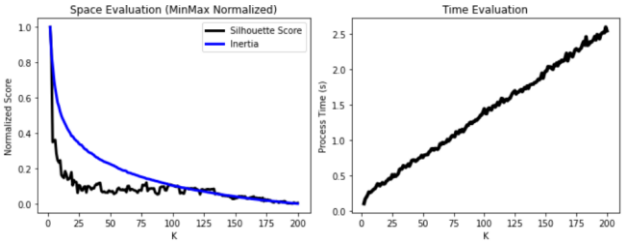

Figure 1. [Breast] K-means Evaluations

While Inertia drops slowly, Silhouette score decreased significantly after K=10. The reason may be better explained from a scatter plot. It is hard to visualize the clusters in 30-dimensional space, so I ran 2D PCA to project the clusters to a 2D plane. The explained variance is 0.63 and should be enough to draw intuitions behind it. The result is shown in Figure 2.

Note that Silhouette and Inertia have been normalized into a range [0, 1], so K=2 always has the value of 1. The upper left figure is the original scatter plot based on labels, whereas the remaining parts are cluster assignments given different K. When K=4, it starts to separate the

cluster which is originally an optimal split. It increases the inter-cluster distance unreasonably. Since Silhouette score considered such an issue, it is a better measure than Inertia. In this regard, the optimal split is K=2. In Figure 2, we can see that K=2 is closed to the label assignment.

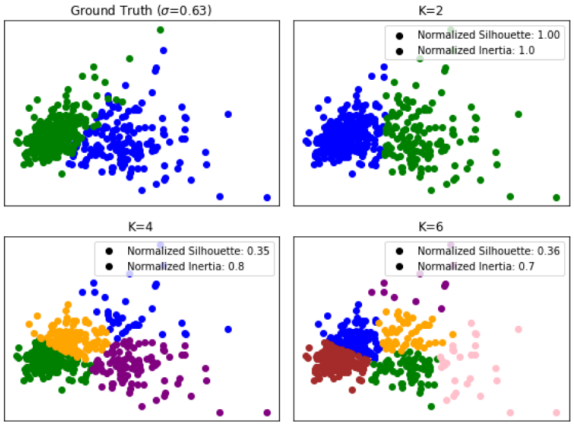

Figure 2. [Breast] K-means Assignment

One application of K-means is to enrich the features of supervised model by adding an extra feature based on the cluster ID. K=2 is optimal, and I will first apply 2-means to data, append the cluster ID to the feature, and then train it on NN. Compared to the NN trained from scratch, the training score slightly drops from 99.56% to 99.12%, but the test from 95.61% to 98.25%. Thus, adding K-means feature improves NN!

Yet another application of clustering is the compression. By assigning feature vectors to their nearest centroids, the representation can be simplified. Here I train DT and NN models on compressed K-means features and compare to the uncompressed model. Notice that the test set is unobservable, so K-means should be fit only from the training set. The result is in Figure 3. K-means is unable to improve the NN, yet it can reduce the overfitting of DT, as the test curve is closer to the training curve. An optimal K is 62 yielding 97.37% test accuracy.

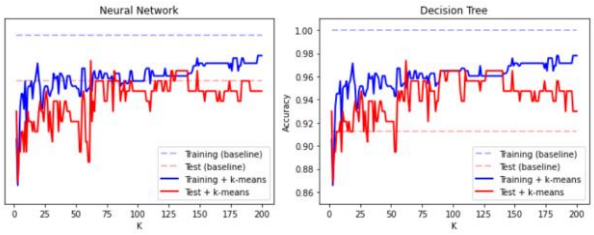

Figure 3. [Breast] K-means on NN and DT

Such a fact might be explained by a pair plot given some combinations of selected features. Figure 4 shows 3 different feature pairs on the columns. The first row is decision boundaries from DT which is trained from scratch, and the second row is from the DT trained on 62-means features. We can see the original DT overfits on the test data points on the border, and 62-means features make DT to draw simpler decision lines. However, K-means is unsuitable for NN since it can capture non-linear decision boundaries and thus avoid overfitting. Running K-means may force NN to neglect some important features.

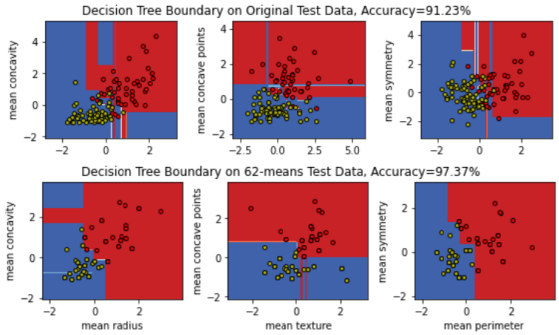

Figure 4. [Breast] K-means + DT Pair Plots

B. GMM – Breast Cancer

To compared with K-means, I again use the Silhouette and process time on different N. The result in Figure 3 shows Silhouette also quickly drops with small K, meaning a few components are sufficient to represent the whole data.

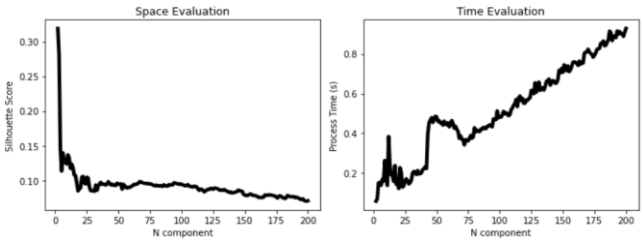

Figure 5. GMM Evaluations

To visualize it, I use 2D PCA and show the cluster result in Figure 6. The result is similar to K-means when N is small, yet a large N induces less confident assignment.

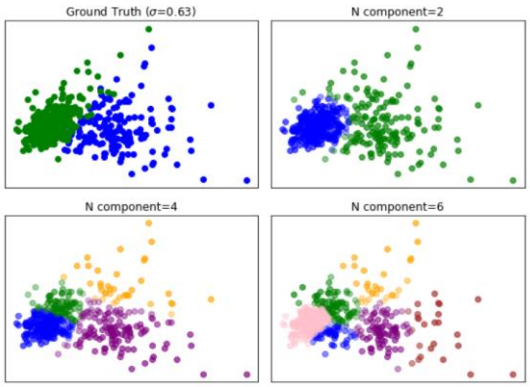

Figure 6. [Breast] GMM Assignment

To have a deeper look, I visualize the K=2 result in Figure 7. It shows centroids of 2-means are slightly closer to the label, and GMM yields the means slightly closer to the boundary. It is reasonable since it predicts based on probability. Thresholding forces it to discard the knowledge about probabilities and is therefore unsuitable, especially when N is large. Despite it, 2 cluster methods still yield very closed means at N=2.
We may again regard the GMM assignment as a new feature, which is continuous in range [0,1]. Unsurprisingly, it yields similar results to the K-means, with training 99.12% and test 98.25%. Whatever clustering methods, both outperform the model trained on the original feature only.

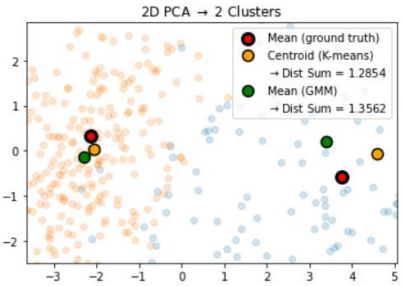

Figure 7. [Breast] 2-means vs. 2-GMM

Now we have an intuition that GMM is not good at tasks requiring thresholding when N is large. Figure 8 shows reducing representation of features by assigning to the means is extremely unsuitable with a large N. DT trained on those reduced features losses too much information about the original data distribution and is thus overfitting on the training data distribution. Pair plot study shows when N is large, two test data points with different labels may ended up being assigned to the same centroid. Note that GMM here is fitted from the training set. All evidence indicates GMM with large N is not suitable for discarding its probability information.

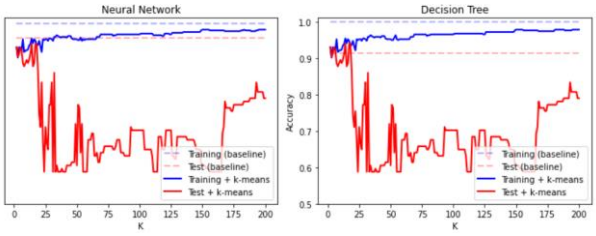

Figure 8. [Breast] GMM on NN and DT

C. PCA – Breast Cancer

Since PCA builds components based on the orthogonality, PCA with larger N is a superset of PCA with small N, and thus always yield a larger sum of explained variance. The learning curve is in Figure 9. Notice that increasing the variance is related to the singular value.

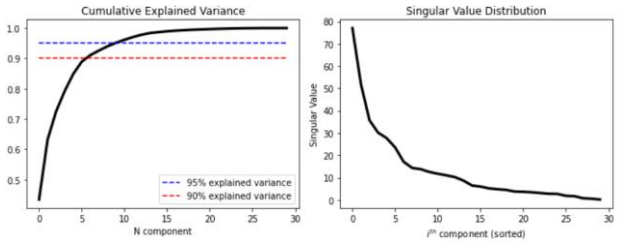

Figure 9. [Breast] PCA Evaluations

As N=2, the sum of variance exceeds 50%. N=7 reaches 90%, and N=10 reaches 95%. So, retaining 10 components is enough to represent the whole data distribution. We may have a look on the coefficient distribution on the component. The result is shown in Figure 10, where the left graph has an unsorted x-axis and vice versa.

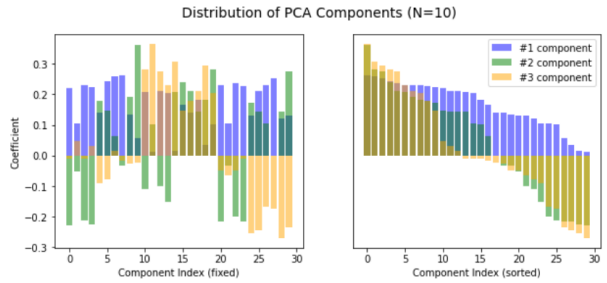

Figure 10. [Breast] PCA Feature Importance (1)

The right figure shows that magnitudes are almost non-zero and linearly decreasing. Figure 11 shows a detail of the 5 largest coefficients on the top 2 components. The first one seems to suggest “concave”, and the second one suggest “fractal” keyword. PCA seems to capture some individual objects underlying the data!

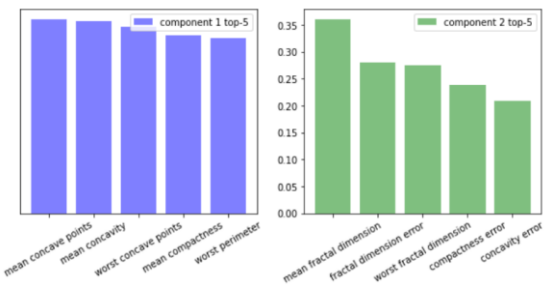

Figure 11. [Breast] PCA Feature Importance (2)

Next, I measure how different N affects the reconstruction error. Given the matrices trained on the training data, PCA can first transform the training and test sets and then reconstruct them back. The curve with different N is identical to ICA, so I will skip the plot here and will explain in the next section. As N is more than 10 (90% variance), the mean squared error become less than 2.5 and reaches the global optimum. The test curve is not farther away from the training curve, meaning the components are sharable between two sets from the same distribution.

Now consider the combination of PCA with clustering. We may conduct the PCA before or after clustering. If PCA is done after clustering, the centroids in the original space are projected based on PCA components. By setting K=2, we may visualize which method is better by seeing how close it is to the ground truth, (the average point location). Results in Figure 12 shows the result of 2D PCA and 2-means.

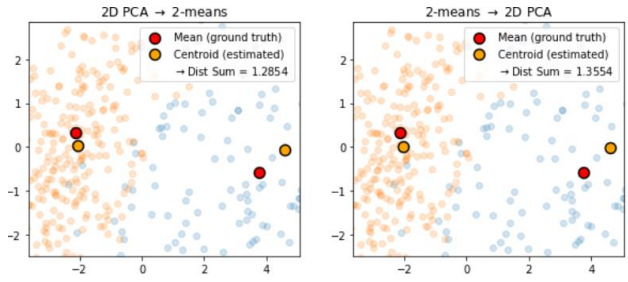

Figure 12. [Breast] PCA + K-means Scatter Plot

It shows that PCA before K-means yields a lower sum of distance from centroids to labels. Actually, 2-GMM after 2D PCA also yields a lower sum of distance than the reversed order, and replacing PCA with ICA also indicates that dimensionality should be prioritized. Given this knowledge, I can compare K-means with GMM in the reduced feature space and visualize them,

which has already been shown in Figure 7.

D. ICA – Breast Cancer

ICA tries to find independent components of data. In experiment, it is however harder to train and may sometimes cause convergence warning. With a proper setting, the average kurtosis and reconstruction error curves are in Figure 13.

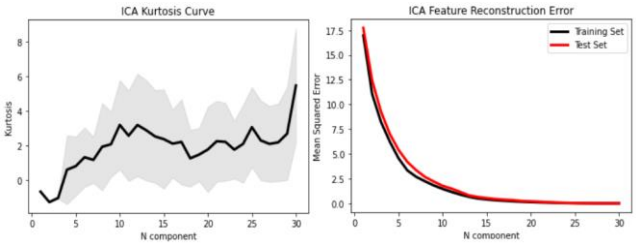

Figure 13. [Breast] ICA Evaluations

Kurtosis is not linearly increasing, so too large N may cause ICA to find some unwanted dependent components. The local optimal value around the middle 3.18 as N=10. On the other hand, the reconstruction error is monotonically decreasing like PCA, and it turns out ICA and PCA have identical reconstruction curve! This fact may be explained by that ICA components are not only independent but also orthogonal to each other here. To verify the orthogonality, I take combinations of the inner product from 2 distinct components, averaging them to see if they are nearly zero. Result shows that when N is smaller than 25, the mean values turn out be 0, meaning that the components in ICA are nearly orthogonal in the breast cancer data!

We may have a look at the ICA component distribution, which is in Figure 14. It shows that different components do not overlap much like PCA, yet the magnitudes of the coefficients are sparser and mostly zeros around.

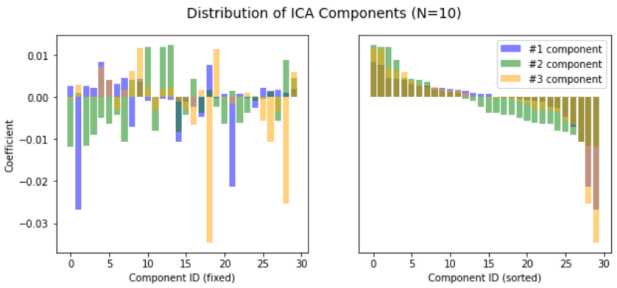

Figure 14. [Breast] ICA Feature Importance

Figure 15 shows a detailed distribution. ICA tends to capture the attributes instead of the individual objects in the data. For example, the first component capture “mean” and “symmetry” keyword, and the second one capture “error” or “worst”. While PCA can detect objects, ICA is able to extract certain attributes in all population.

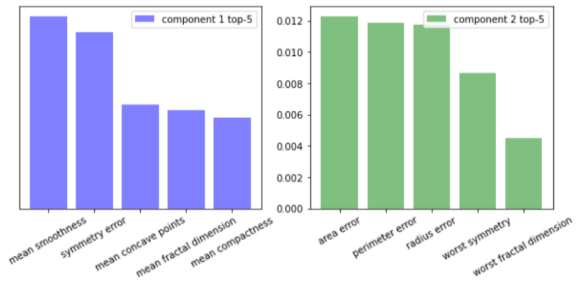

Figure 15. [Breast] ICA Feature Importance (2)

E. RCA – Breast

RCA generates random projection matrices, so it is expected to yield a higher reconstruction error compared to PCA and ICA. The error plot and the projected scatter plot are in Figure 16.

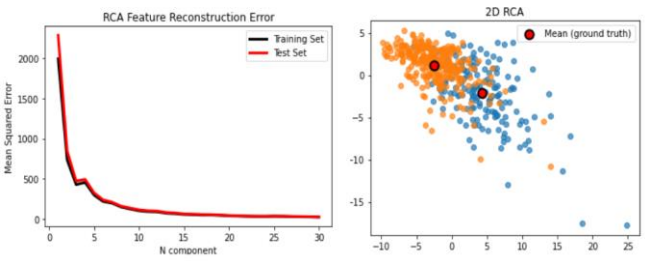

Figure 16. [Breast] RCA Error and Scatter Plots

The minimum training and test errors are 27.29 and 29.08 at N=30, which are even worse than the worse cases of PCA and ICA. While RCA is ineffective of reconstructing the data, it runs very fast. Also, the right scatter plot shows it somehow maintain the original distribution of data. We may see the component distribution in Figure 17, which is similar to PCA. By looking at the topmost values, it turns out that the first component "by chance" captures the keyword "smoothness", and the other captures "concave". Such a fact also support that RCA components are not so ineffective to explain the data.

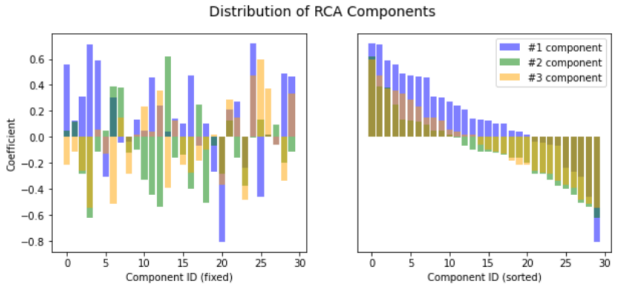

Figure 17. [Breast] RCA Feature Importance

F. LDA – Breast Cancer

LDA requires information about the label, so it is expected to be too most capable method to explain the data than the other unsupervised methods. The maximal N for LDA is restricted to be not greater than the label counts, so there is only one possible outcome in such a binary classification problem. N=1 yields a trivial 1.0 explained variance. We may compare PCA and LDA by visualizing how well they separate the data in a 1D projected space in Figure 18. Two colors represent distinct labels, and we can see that LDA has fewer overlapping data points.

However, a better explained variance is not guaranteed to have a lower reconstruction error. By fitting LDA on the training data, it has 30.17 reconstruction error on training and 32.42 on test, even worse than PCA and ICA when N=1! The reconstruction requires computing pseudo-inverse of component if not full ranked, which makes inaccurate reconstruction. Hence, LDA is good at explaining variances, not reconstruction.

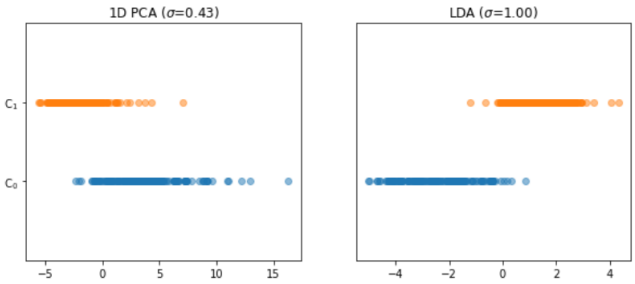

Figure 18. [Breast] PCA vs. LDA Scatter Plot

G. Cluster + Dim Reduction – Breast Cancer

Since 4 dimensionality reduction methods have different evaluation metrices, one way to compare fairly is to look at their reconstruction errors. To summarize the previous experiments, PCA and ICA have the lowest, identical curves. LDA only has a single component with a high error and is unable to be improved further. RCA has a high error at first but finally outperforms LDA with sufficient large N. A common fact of the 4 methods is that their reconstruction error reaches the optimal when N is the largest. With such a setting, we may compare how the cluster methods performs on 4 methods in different N. Results are in Figure X, where the left figure is the K-means result, and the right is the GMM. Here LDA has the highest inter-cluster distance based on the Silhouette, while ICA become the worst when N is large. Again, it indicates LDA is very good at maintaining the data variance.

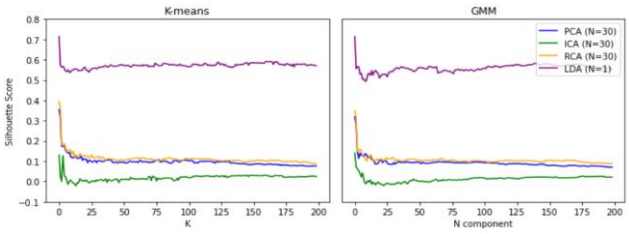

Figure 19. [Breast] Clustering + Dim Reduction

H. K-means – MNIST

The evaluation is shown in Figure 20. Again, Silhouette and Inertia has different ranges, so I will normalize them in [0,1]. Like breast cancer dataset, Inertia drops slowly while Silhouette has two peaks at 5 and 25.

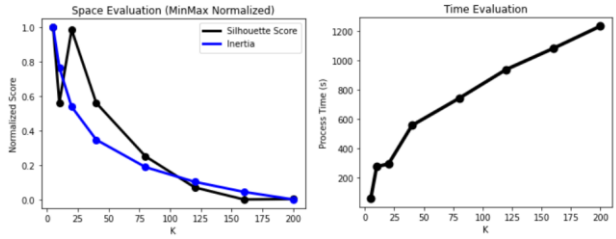

Figure 20. [MNIST] K-means Evaluations

It is worth to investigate why the clustering result of K=10 is not satisfying, since we know an ideal split should be 10 (it is the number of label). Given the clustering assignment, we can check how many points within a cluster belong to some labels in Figure 21. The x-axis is the cluster ID with a centroid image, and the y-axis is the true label. Clusters in columns 7 and 10 cannot effectively distinguish 4, 7 from digit 9, and clusters in columns 5 and 9 cannot separate digits 3 or 8. Their centroids are ambiguous, so the overall Inertia is high, and Silhouette is low.

From the breast cancer experiment we know K-means can reduce the overfitting of DT. Here I evaluate if it is still true in MNIST. The curve is shown in 22, and we can see that a sufficient K=120 (about 15% of the original) can prevent DT from overfitting while boost the accuracy, which connects the breast cancer result.

Figure 21. [MNIST] 10-means Heatmap

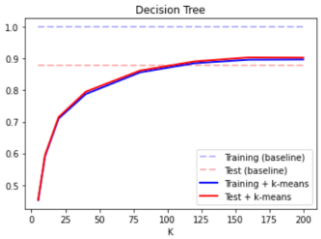

Figure 22. [MNIST] K-means on NN

I. GMM – MNIST

GMM turns out to be very slow on MNIST, and even running on N=2 requires 1.67 hours! Based on the previous result, it is unnecessary to have a large N for GMM. Within the range of 20, the result in Figure 23 shows N=2 yields an optimal Silhouette and runtime. Running GMM on such a huge dataset is however impractical.

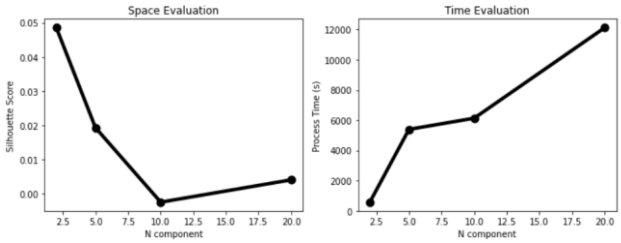

Figure 23. [MNIST] GMM Evaluations

Notice that N=10 has the lowest Silhouette score. Compared with the 10-means result, we may find some reasoning behind it in Figure 24. When N=10, GMM is not only hard to classify

digits 4 ad 5, but the mean points even become more obscure than 10-means. Thresholding is absolutely not a suitable method for GMM, just like the result in Figure 7 also shows the sum of distance is farther away from the true mean due to the embedded information about probabilities.

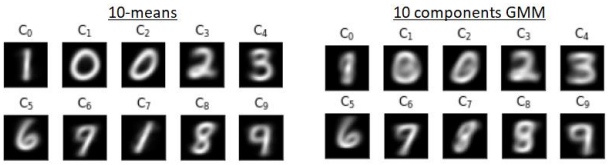

Figure 24. [MNIST] K-means vs. GMM

J. PCA – MNIST

The evaluations of explained variance and the singular values are shown in Figure 25. As N=11, the sum of variance has already exceeded 50%. As N=88 (11% of the original dimension), it yields 90% variance, and N=155 (20%) yields 95% variance. PCA can effectively reduce some redundant features in MNIST based on that fact.

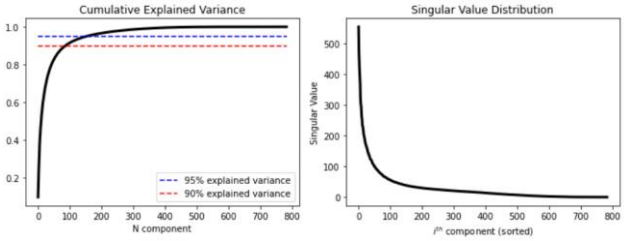

Figure 25. [MNIST] PCA Evaluations

From previous experiment, we found PCA can extract individual objects. It is interesting to visualize the top-10 components (in Figure 26). Each component seems to extract some basic block from digit contours (they seem like some naïve digits). Such a fact connects to the breast cancer case.

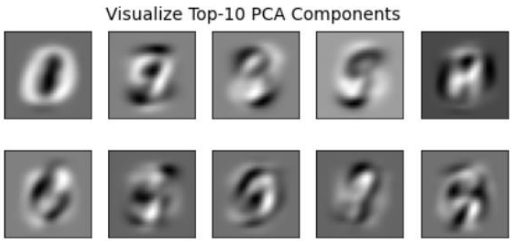

Figure 26. [MNIST] PCA Top-10 Components

We may further analyze the distribution of PCA components in Figure 27. Again, most of the magnitudes of coefficients are non-zero and almost linearly decreasing. Deeper investigation on some largest coefficients shows nearby pixel values from the largest one is also large, so they can capture neighborhood areas and comprises some individual components in MNIST.

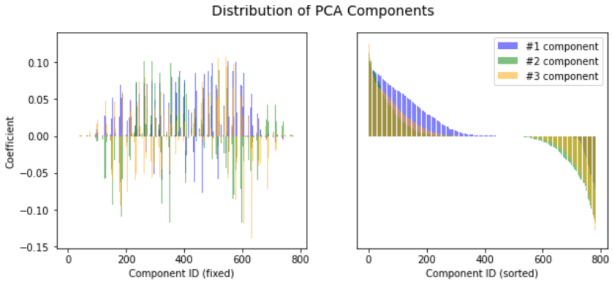

Figure 27. [MNIST] PCA Feature Importance

As PCA can reduce the number of features while maintaining the variance, it is important to know if it can also save the running time and reduce overfitting problems for some supervised learning methods. Since DT tends to overfit to large-scale data like MNIST, it is worth to see how PCA affect the DT performances in Figure 28. Note the dash yellow line on the right figure represents the DT fit time on the original data. The overall runtime consists of construction of PCA features and the fit time on DT. Sadly, the accuracy is not improved, but the running time becomes even slower than the model trained on the original 784-D features when N exceeds 100.

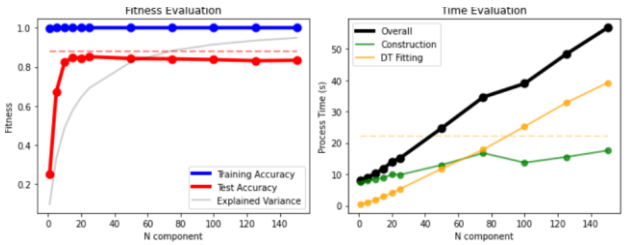

Figure 28. [MNIST] PCA + DT Performance

Figure 29 even shows that tree becomes deeper than the original one when N exceeds 50. PCA may keep some important features that are not ignorable by each decision rounds. The right figure shows the feature importance is slowly increasing on the original feature space, while faster on the PCA features with larger N.

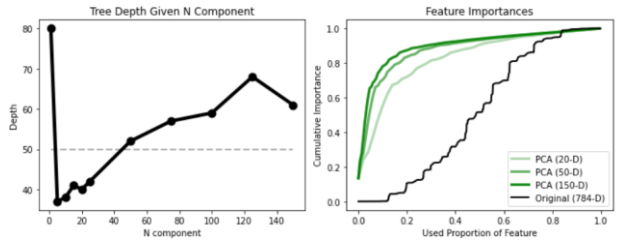

Figure 29. [MNIST] PCA + DT Performance

How about NN? With some specific N, we can see though the test accuracy is not improved, the runtime can be reduced two times from the origin! Since the number of parameters in NN is fixed, and the decision boundaries of NN can be non-linear, running PCA on NN is thus more suitable with sufficient N than on DT.

Table 1. [MNIST] PCA + NN Performance

Factor / Method	**Explain Variance**	**Training Accuracy**	**Test Accuracy**	**NN Fit Time**
Baseline	(X)	98.99%	**<u>96.85%</u>**	223.83
[PCA] N=11	50%	94.28%	93.62%	**<u>27.72</u>**
[PCA] N=88	90%	**<u>99.05%</u>**	96.66%	162.92
[PCA] N=155	95%	98.91%	96.8%	103.48

K. ICA – MNIST

The learning curve for ICA is in Figure 30. Unlike the breast cancer case, here Kurtosis is monotonically increasing before N=600. Since MNIST is more complex, it may require more components to capture independent features on the data. However, too much N may force ICA to capture unwanted dependent variables, just like the result on the breast cancer dataset.

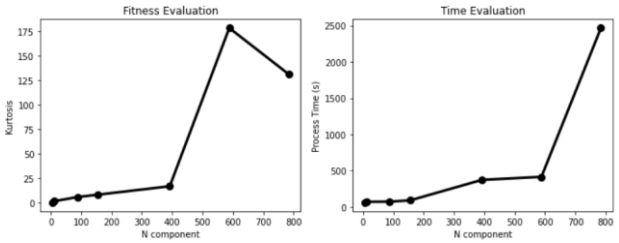

Figure 30. [MNIST] ICA Evaluations

The reconstruction error for ICA in MNIST requires at least N=600, unlike PCA only needs at least N=400. Maybe the orthogonality aspect is more judgmental than the independence.

We may further analyze the distribution of ICA components with optimal N=600 in Figure 31. It is like the breast cancer case: most values are small with only a few peaks, and the largest values do not correspond to some nearby pixels. Since the pixel distribution is extreme and hard to visualize, further investigation is required to understand what these components are.

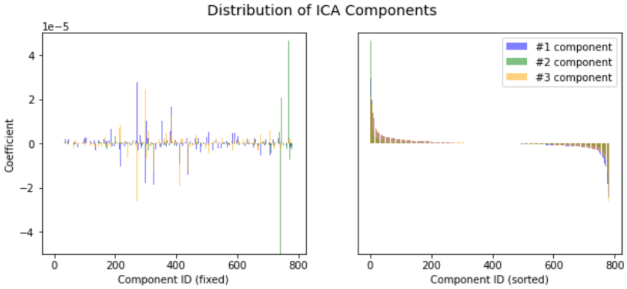

Figure 31. [MNIST] ICA Feature Importance

L. RCA – MNIST

The evaluation curves of RCA are in Figure 32. It has a minimal reconstruction error 11.7 on N=784, which is higher than the PCA error (0) and the ICA error (0.12). However, the running time is very fast and requires only few seconds.

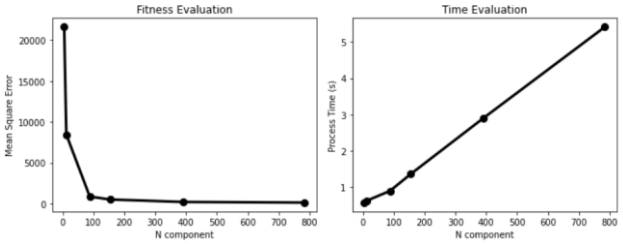

Figure 32. [MNIST] RCA Evaluations

There is no specific distribution pattern for RCA components since it uses random matrices. The visualization of components are just noises.

M. LDA – MNIST

Since MNIST has 10 labels, the maximal N for LDA is also 10. The evaluation curve and component distribution are shown in Figure 33, where N=8 yields 90% variance and N=9 has 95% of it. The component distribution is even more sparse than ICA and thus reduce lots of redundant information embedded in MNIST.

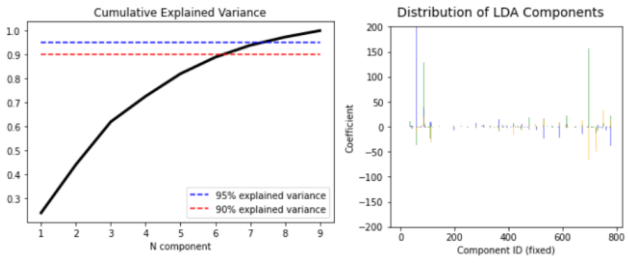

Figure 33. [MNIST] LDA Evaluations

4. Conclusions

Here are some brief conclusions draw from the previous experimental results:

K-means

- Silhouette measures better than Inertia since it changes more rapidly.
- Assigning the cluster ID as a new feature can boost the test accuracy of NN.
- Replacing features with the nearest centroid can reduce the overfitting of DT.
- 10-means is insufficient to detect MNIST.

Gaussian Mixture Model (GMM)

- Hard assignment on data is unsuitable when N components is not small.
- Runtime overhead is huge for large datasets.

Principle Component Analysis (PCA)

- Each component captures individual objects
- Component coefficients are mostly non-zero and linearly decreasing.
- DT trained on PCA-projected features does not run faster since all features are important.
- NN trained on PCA-projected features runs faster without losing test accuracy.

Independent Component Analysis (ICA)

- Each component captures attributes that are shared in the population.
- Component coefficients are mostly zeros.
- Clustering on the projected-ICA is the worst among the other dim reduction methods.

Randomized Component Analysis (RCA)

- It performs very fast and requires sufficient N to reduce the reconstruction error.

Linear Discriminant Analysis (LDA)

- It is good at maintaining data distribution in a small dimensional space.
- It is not good at data reconstruction.