CS7641 Assignment#3

Unsupervised Learning and Dimensionality Reduction

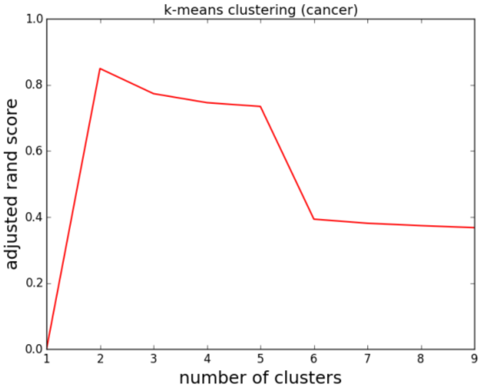
Yancheng Liu (yliu723@gatech.edu)

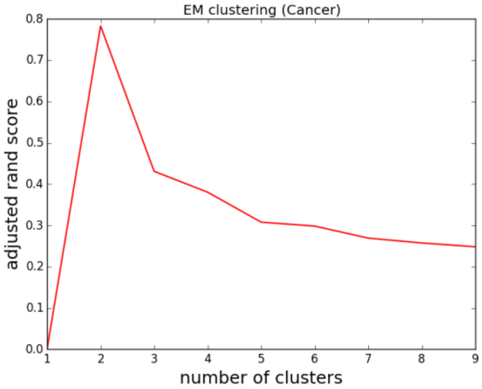
Datasets Used in This Study

The first dataset I chose to look at is the Breast Cancer Wisconsin Dataset (obtained from UCI Machine Learning Repository). Breast cancer is the most common cancer among women and the second leading cause of cancer death. In 2014, an estimated 232,670 new cases of invasive breast cancer were expected to be diagnosed in women in the U.S. and about 1 in 8 women will develop invasive breast cancer over the course of her lifetime (CDC). The data in this dataset are real cancer patient data obtained from the University of Wisconsin Hospitals in 1990s. There are 699 samples in this dataset and 11 attributes, including cell size, cell shape, clump thickness etc., all represented by digits from 1-10 (except the ID numbers which will be excluded from fitting). I use this dataset to practice what I learnt about unsupervised learning: basically grouping samples into clusters and comparing with the given labels (whether a tumor is benign or malignant).

The second dataset I examined is the CoverForest Dataset. The dataset include data about forest cover type in the Roosevelt National Forest of northern Colorado obtained from US Geological Survey. The forest cover dataset has important practical implication for ecological studies. Since cover type are with minimal human-caused disturbance, knowledge about how ecological factors can contribute to forest cover type can be learned by analyzing this dataset. There are more than 500,000 samples in this dataset!!! The huge amount of data makes the clustering task very challenging. Moreover, the dataset contains 54 attributes. Thus it is a perfect dataset to test the effect of dimensionality reduction on clustering accuracy and running time.

I. Clustering Algorithms





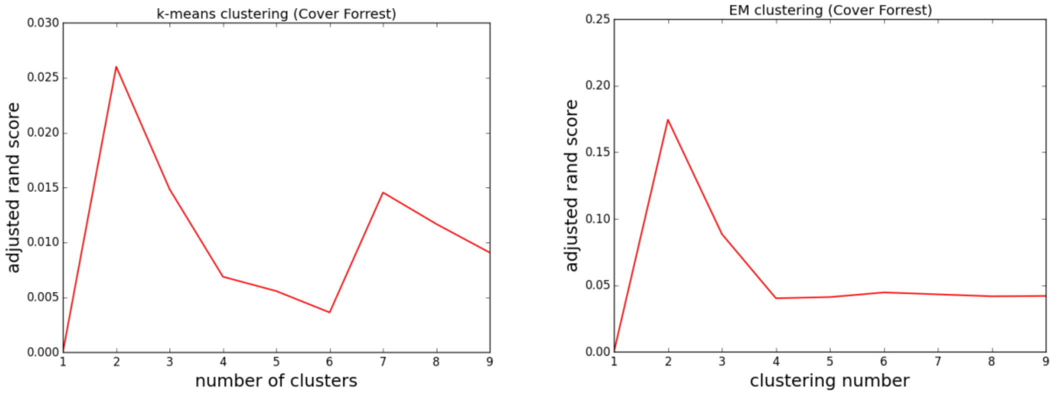


Figure 1. Choose the appropriate number of clusters.

Table 1. Performance summary of the K-means and EM algorithms.

Cancer					
	ARI	AMI	homogeneity_score	completeness_score	V_measure
K-means (k-means++)	0.839	0.732	0.732	0.74	0.736
K-means (random)	0.839	0.732	0.732	0.74	0.736
EM	0.783	0.703	0.734	0.703	0.718
CoverForrest					
	ARI	AMI	homogeneity_score	completeness_score	V_measure
K-means (k-means++)	0.187	nan	0.161	0.658	0.259
K-means (random)	0.026	nan	0.067	0.089	0.077
EM	0.175	nan	0.116	0.275	0.163

Clustering is an unsupervised data exploration method that groups samples with similar characteristics together to facilitate further analysis. Two popular clustering algorithms, K-means and Expectation-Maximizations (EM), are used here to interrogate the cancer and coverForest datasets. The K-means algorithm starts by choosing initial centroids, and then iterates between two major steps until converge. First, all samples are assigned to their nearest centroids. Second, new centroids are recomputed by calculate the mean of all the samples belonging to each old centroid. Although K-means will always converge, it may stuck at a local minimum depends on the initial positions of centroids. One way to address this problem is to do random start with different initial values of centroids. Another way is to maximize the initial distance between centroids, which has been suggested to have better performance than random initialization (1). In the present experiments, I compared these two methods by using the “k-means++” (select distant centroids) and “random” (random initialization) schemes implemented in scikit-learn. The results are shown in Table1. For the cancer dataset, both methods performed well and have the same score in metrics measures. But for the coverForest dataset, the “k-means++" scheme clearly out-performed the random initialization method, suggesting that selecting distant centroids is a better initialization scheme than random start for more complex datasets.

Both K-means and EM require the number of clusters to be pre-specified. Finding the appropriate number is a difficult process, especially for unlabeled dataset where there is no gold standard to determine what is the "correct" clustering. However, because both datasets are reused from assignment #1 and both have classification labels, I can use these labels as the standard to decide what will be the appropriate number of clusters to use in K-means and EM. As shown in Figure 1, the performance of the clustering with different number of clusters was evaluated using the Adjusted Rand Index (ARI). The ARI is the chance-normalized measurement of the similarity of the two clustering assignments and ignoring permutations. The clustering of the cancer data showed the highest ARI score when K=2 (Figure 1). This clustering scheme makes perfect sense and lines up well with the classification of the two tumor types (malignant and benign). In contrast, the clustering of coverForest data did not line up with the labels. Both K-means and EM showed the peak ARI score when the number of clusters was set at 2, suggesting these samples naturally separated into 2 clusters (Figure 1). But the labels indicate there are 7 different cover types. This inconsistency is intriguing and probably reflects the skewed sample distribution across the 7 cover forest types in the dataset. In support of this explanation, when inspected the sample distribution in detail, I found that type 1 and 2 are way over-represented, and type 3-7 are under-represented (Figure 2).

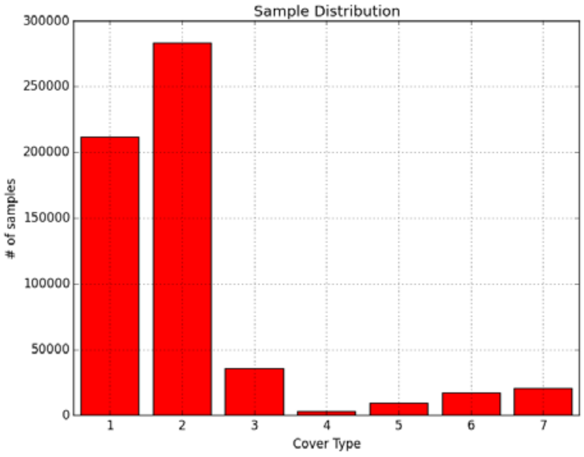


Figure 2. Sample Distribution of Cover Forest Dataset.

Another interesting observation is that K-means consistently achieved better performance than EM in both datasets when measuring the similarity of the clustering assignments with the labels. This is likely due to nature of the problem I chose. Both datasets are classification problems and the labels are discrete values, therefore K-means which does hard partitioning and assigns each sample to exactly one cluster will models the data better than the soft-clustering algorithm EM which uses probabilistic assignment.

II. Dimensionality Reduction Algorithms

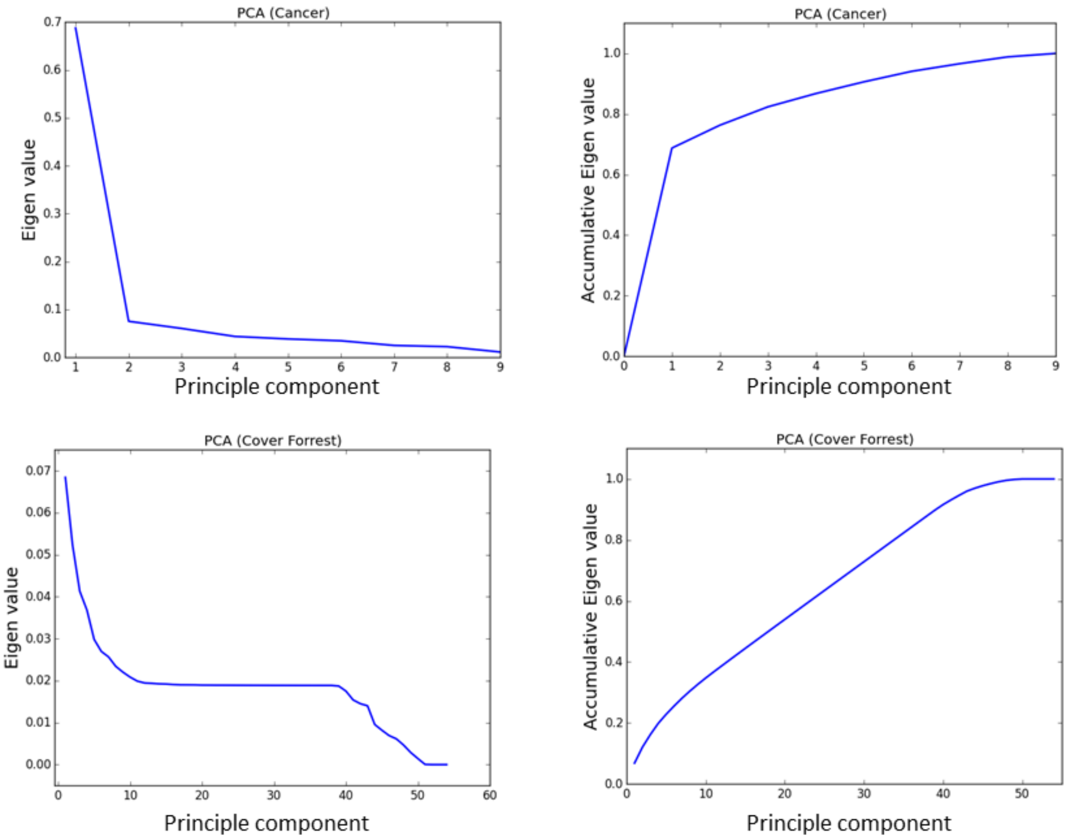


Figure 3. Eigenvalue distribution for PCA components.

In this study, different dimensionality reduction algorithms are applied to the two datasets and results are compared. Principal Component Analysis (PCA) aims to retain as much as possible the variation present between samples when reducing the dimensionality. It does so by finding the component dimensions that maximize the variance of the data. In practice, this equals to find the eigenvectors with largest eigenvalues. Therefore, I calculated the eigenvalue for each component dimension and plotted the distribution in Figure 3. As shown, the cancer dataset showed a distinct eigenvalue “elbow” with 2~4 components, suggesting that we’ll be able to reduce the dimension from 9 to 2~4 while at the same time retain most of the variance. Thus 3 principle components were selected to perform dimensionality reduction. As shown in Figure 4, plotting the data in the new space with the 3 principle components as directions naturally divided the samples into two clusters, and the two clusters correlated very well with the labels. This result demonstrates that we can successfully map the cancer data to a lower-dimensional space and meanwhile preserve the interrelationship between samples. In contrast, the “elbow” of the coverForest dataset is much harder to identify on the graph. There is no single or a few components that can explain the majority of variance. The eigenvalue of the first ~40 components are all in a comparable range. For the ease of visualization and comparison with the cancer dataset, the number of components was also set at 3 for dimensionality reduction. As expected, plotting the samples in the new space with the 3 principle components improved the separation between the 7 forest types, but not a whole lot (Figure 5).

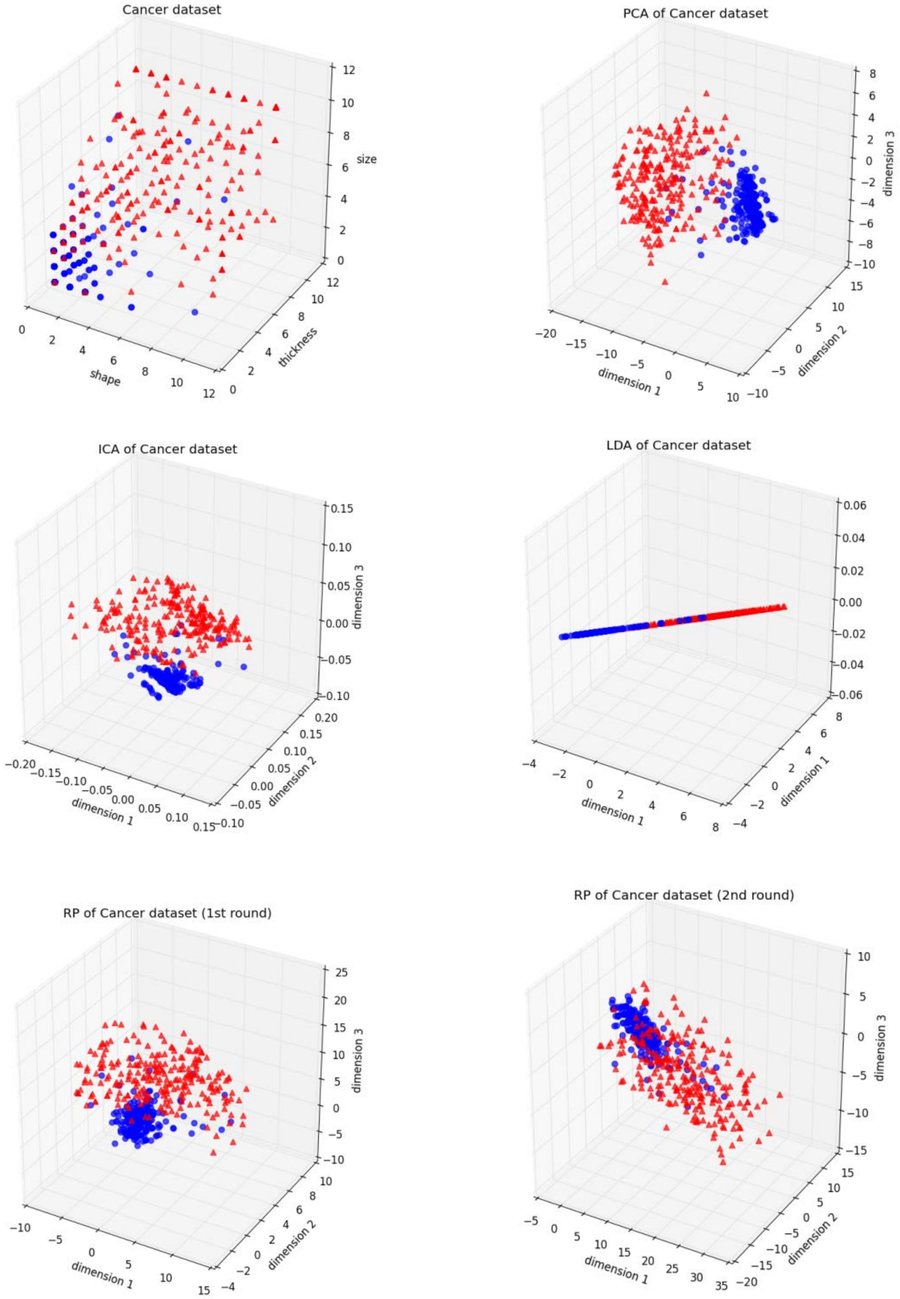


Figure 4. Sample distributions of the cancer dataset in the new spaces with reduced dimensions.

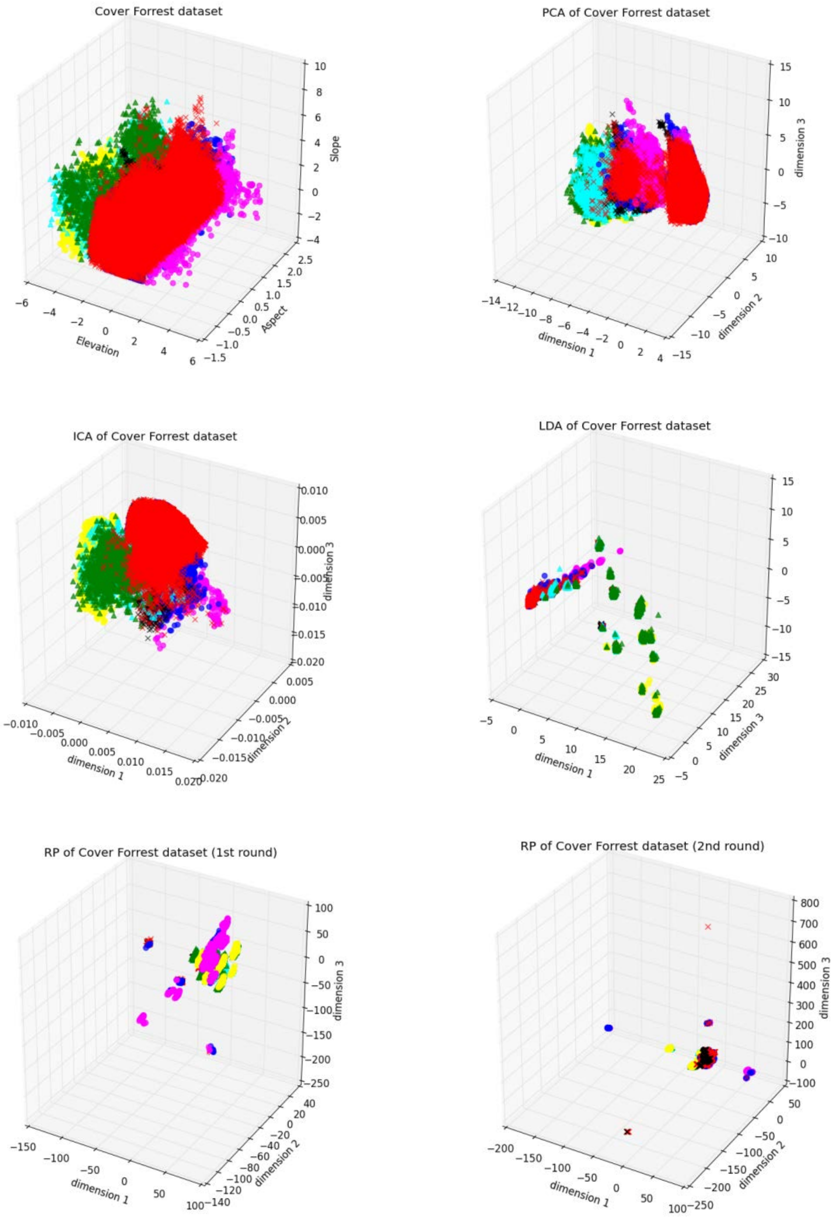


Figure 5. Sample distributions of the coverForest dataset in the new spaces with reduced dimensions.

Next, dimensionality reduction was performed using three other algorithms and the results were plotted in the new lower-dimensional space (Figure 4 and 5). Independent Component Analysis (ICA) aims to separate a superimposed signal into individual components which are not only uncorrelated but also maximally independent. The separation process assumes that the individual components are non-Gaussian. Therefore, after dimensionality reduction, the distribution of samples in the new space is less-Gaussian when compared to PCA.

Linear Discriminant Analysis (LDA) is similar to PCA. Both are linear transformation techniques. But instead of finding the dimensions with maximum variance, LDA takes in the class labels and searches for the dimensions that maximize the distance between different classes. This is clearly showed in Figure 4, where LDA only needs one dimension to retain the class-separation information for the cancer dataset. However, LDA struggled to find the correct dimensions to maximize the distance between classes in the coverForest dataset. This is probably due to the complex nature of the dataset, which may not contain enough information necessary for class discrimination, or the distance between classes is inherently small. Random Projection (RP) is another popular dimensionality reduction algorithm for mapping Gaussian mixtures into lower-dimensional space. Instead of mathematically select components like PCA, RP just project the data onto a random lower-dimensional space. This approach is significantly less expensive in computation and also preserves well the distance metrics between data. However, due to its random nature, it can generate totally different distributions if it was run multiple times, but the interrelationship between samples are mostly preserved (Figure 4 and 5).

III. Reproduce Clustering Experiments with Data Preprocessed by Dimensionality Reduction

Table 2. Performance summary of clustering using data preprocessed by dimensionality reduction.

Cancer-kmeans						
	ARI	AMI	homogeneity_score	completeness_score	V_measure	time (s)
non-reduced	0.839	0.732	0.732	0.74	0.736	0.131
PCA	0.85	0.746	0.747	0.752	0.749	0.0875
ICA	0.584	0.464	0.465	0.632	0.496	0.464
RP-1st	0.333	0.25	0.251	0.364	0.297	0.453
RP-2nd	0.699	0.574	0.575	0.617	0.595	0.428
RP-3rd	0.755	0.629	0.629	0.648	0.639	0.438
LDA	0.844	0.739	0.739	0.75	0.744	0.1
Cancer-EM						
	ARI	AMI	homogeneity_score	completeness_score	V_measure	time (s)
non-reduced	0.783	0.703	0.734	0.703	0.718	0.471
PCA	0.861	0.783	0.805	0.783	0.794	0.465
ICA	0.0183	0.0154	0.0167	0.192	0.0307	0.808
RP-1st	0.532	0.422	0.445	0.423	0.433	0.813
RP-2nd	0.439	0.306	0.307	0.33	0.318	0.813
RP-3rd	0.422	0.291	0.292	0.326	0.308	0.812
LDA	0.861	0.783	0.806	0.783	0.7894	0.467
CoverForrest-kmeans						

	ARI	AMI	homogeneity_score	completeness_score	V_measure	time (s)
non-reduced	0.187	nan	0.161	0.658	0.259	4.59
PCA	0.184	nan	0.159	0.664	0.257	2.53
ICA	0.186	nan	0.16	0.663	0.258	2.032
RP-1st	0.024	nan	0.011	0.055	0.018	0.68
RP-2nd	0.0065	nan	0.0043	0.0057	0.0049	1.256
RP-3rd	0.07	nan	0.062	0.435	0.109	0.974
LDA	0.131	nan	0.141	0.874	0.243	0.59

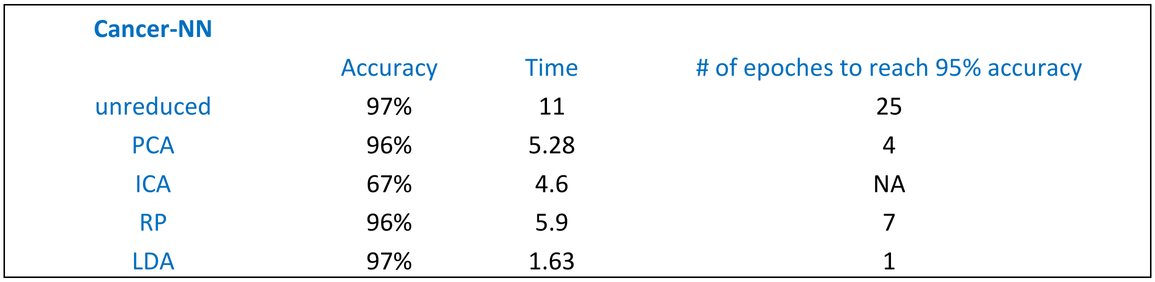
CoverForrest-EM

	ARI	AMI	homogeneity_score	completeness_score	V_measure	time (s)
non-reduced	0.174	nan	0.116	0.274	0.163	4.83
PCA	0.193	nan	0.157	0.577	0.247	3.83
ICA	0	nan	0	1	0	2.36
RP-1st	0.122	nan	0.084	0.357	0.136	1.351
RP-2nd	0.062	nan	0.019	0.086	0.031	2.05
RP-3rd	0.169	nan	0.104	0.245	1.46	1.38
LDA	0.211	nan	0.164	0.577	0.255	1.59

In this section, four dimensionality-reduction techniques were used to preprocess the data points for the two datasets, and then the processed data were used for clustering using K-means and EM algorithms. The performance was summarized in Table 2. In general, one would expect clustering using data with less dimensionality should be more time-efficient than running the same clustering technique with dimensionality un-reduced data. This is clearly the case when running clustering with coverForest data (Table 2). But the improvement in wall-clock time is less evident with the cancer dataset preprocessed by PCA and LDA (Table 2). The original dimensionality of the coverForest dataset is much higher than that of the cancer dataset (55 vs 9). Therefore the time improvement is more evident with the coverForest dataset. Interestingly, the running time for ICA and RP reduced data is longer than running without any dimensionality reduction. One possibility is that preprocessing with ICA and RP actually cost significant loss in information about the correlation between data points; therefore it took longer for the clustering algorithms to converge. In line with this possibility, the performance of both K-means and EM decreased when using ICA and RP preprocessed data (Table 2). The performance is measured by the similarity between the clustering assignment and the label-defined classes using ARI, AMI and V-measure. Therefore the low score in these measures indicate that the data were clustered differently from the original data after dimensionality reduction with ICA and RP.

IV. Neural Network Analysis Using Dataset Preprocessed by Dimensionality Reduction

Table 3. Performance summary of ANN using data preprocessed by dimensionality reduction

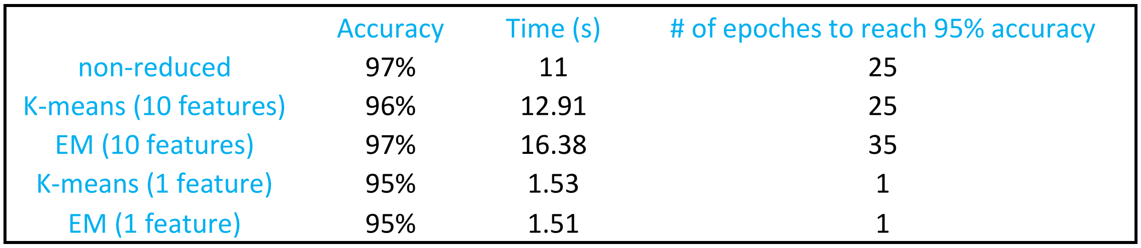


Cancer-NN			
	Accuracy	Time	# of epoches to reach 95% accuracy
unreduced	97%	11	25
PCA	96%	5.28	4
ICA	67%	4.6	NA
RP	96%	5.9	7
LDA	97%	1.63	1

In this set of experiments, four dimensionality-reduction techniques were used to preprocess the cancer dataset, and then the processed data were used for classification using neural network learner. The performance was summarized in Table 3. Clearly, all four techniques reduced the running time and the number of epochs for training the neural network to reach maximum accuracy. This is because the computational cost of training the learner is lower using data points with fewer dimensions. Among these techniques, LDA is the clear winner as it was the fastest and took only one epoch reach >95% accuracy. Meanwhile, feeding the learner with the data preprocessed by PCA, RP and LDA produced comparable accuracy score as using the original non-reduced data. The only exception is ICA, which degrades the accuracy of the neural network learner to less than 70% even with many epochs of training. This is an indication that dimensionality reduction using ICA failed to preserve the similarity/distance information between data points.

V. Neural Network Analysis Using Dataset Preprocessed by Clustering

Table 4. Performance summary of ANN using data preprocessed by Clustering



	Accuracy	Time (s)	# of epoches to reach 95% accuracy
non-reduced	97%	11	25
K-means (10 features)	96%	12.91	25
EM (10 features)	97%	16.38	35
K-means (1 feature)	95%	1.53	1
EM (1 feature)	95%	1.51	1

In this section, two clustering algorithms, K-means and EM, were used to preprocess the cancer dataset, and then the resulting data were for classification using neural network learner. After clustering, two separate experiments were carried out: first, using the original 9 features plus the clustering results as a new feature to train the neural network (10-feature training); second, using only the clustering results as a new feature to train the learner (1-feature training). The results were compared with the learner trained with the original 9 features (Table 4). As shown, training using 10 features produced similar accuracy but required slightly longer training time, which is likely due to adding 1 extra feature to the data points. In sharp contrast, training with the clustering results as the only feature greatly reduced the training time and only needed 1 epoch to reach >95% accuracy. These results demonstrate that reduce the dimensionality of data points by clustering can lead to significant improvement in speed for learners with high computational cost.

VI. Conclusions

In conclusion, dimensionality reduction methods, such as PCA, ICA, RP, LDA, serve to convert data into a lower-dimensional space while at the same time tries to minimize the information loss. With a loose definition, clustering methods such as K-means and EM can also be considered as dimensional reduction techniques. A major advantage of using dimensionality reduction for data preprocessing is to circumvent the curse of dimensionality and therefore reduce the computational cost for further analysis such as classification. However, one should take caution when doing so because, as shown in section III and IV, in some cases dimensionality reduction could result in significant information loss and negatively affects the accuracy of downstream analyses.

Reference

1. David Arthur and Sergei Vassilvitskii. *k-means++: The advantages of careful seeding*. *Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, Society for Industrial and Applied Mathematics (2007).