Mikey Ling
mling7
903278121

Unsupervised Learning and Dimensionality Reduction

1 Introduction

This report covers the implementation of two clustering algorithms (K-Means and Expectation Maximization) and four dimensionality reduction algorithms (Principal Component Analysis, Independent Component Analysis, Random Projections, and Information Gain). The motivation behind the report is to compare and analyze the algorithms by including them in the data pre-processing step of training a neural network.

2 Datasets

The Wisconsin Breast Cancer and Letter Recognition datasets were chosen for this assignment. Note these two datasets are different than the ones used in Assignment 1. The switch to these datasets was made because both datasets used (Abalone and Adult Salary from the UCI database) in Assignments 1 and 2 had significant amounts of overlap.

2.1 Wisconsin Breast Cancer Dataset

This dataset contains 669 instances with 10 features containing information for female patients at the University of Wisconsin Hospital. Class labels of the data set describe potentially cancerous growths found during data acquisition: 458 (65.5%) instances were labeled as benign, and 241 (34.5%) instances were labeled as malignant.

2.2 Letter Recognition Dataset

The Letter Recognition Dataset contains 20,000 instances of handwritten (single) letters from A to Z with 16 features. Clearly, there are 26 different classes an instance can be assigned to. The character images were based on 20 different fonts and each letter within these 20 fonts was randomly distorted to produce all the samples. The Breast Cancer dataset is a binary classification problem based on 10 features variables, while the Letter Recognition dataset is a 26-label classification problem with 16 features. The differing feature to class label ratios between the two datasets will hopefully provide for interesting analysis and results.

3 Clustering

Clustering is the process of separating instances by assigning them to different groups, or clusters. The criteria upon which the instances are clustered greatly depend on the present features within the dataset. The pre-processing step of the workflow is integral in running an effective clustering algorithm. To shed light on the effects pre-processing can have on clustering, K-Means and Expectation Maximization (EM) clustering are first performed without any pre-processing of the datasets.

3.1 K-Means Clustering

K-Means clustering is a centroid-based clustering model. For N clusters, N random points are first designated as centroids for the clusters and assigns instances to these clusters based on user-specified distance metrics, ie Euclidean or Manhattan distances. The algorithm iteratively changes the centroid locations until convergence occurs.

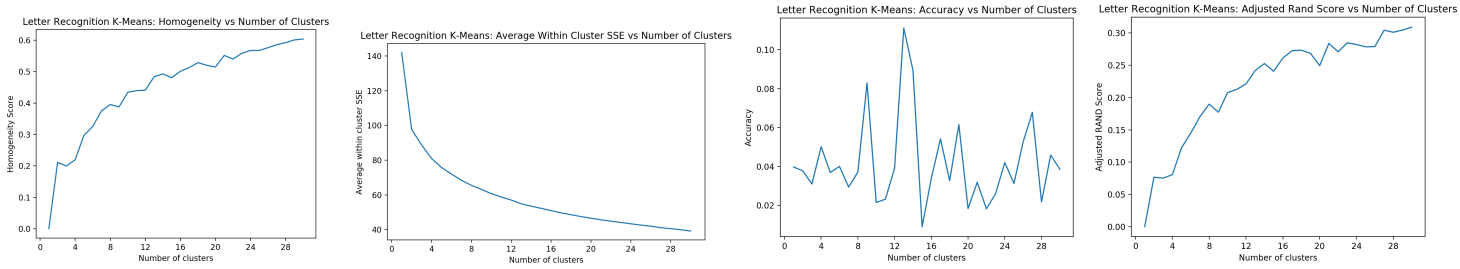


Figure 1 - Letter Recognition K-Means Clustering Results

Figure 2 - Breast Cancer K-Means Clustering Results

Analysis

The K-Means implementation in Scikit-Learn was used to create the visualizations in this section of the report, and it's important to note these results will be used as the baseline for comparison when using the preprocessing techniques described later in this report.

Letter Recognition - As you can see in Figure 1, the homogeneity of the clusters increases as the number of clusters increases. This follows the intuition of the closer the number of clusters reaches the number of classes in the dataset the more the homogeneity of the clusters will increase. However, it's worth noting the homogeneity doesn't reach its maximum value when there are 26 clusters (which is equal to the number of classes in the dataset). It's also interesting to note the accuracy of the clusters does not reach a maximum when the number of clusters equals the number of classes. Why does this occur? My thoughts are the features in the dataset do not offer enough information for the vanilla K-Means algorithms to effectively differentiate between the 26 different letters of the alphabet. Unlike the accuracy graph, the adjusted Rand graph follows a bit more intuition. The adjusted Rand score, which measures how similar clusters are to each other (in our case we compare the test labels to the true labels from the dataset), increases and starts to flatten around 26-28 clusters. In other words, when the number of clusters gets closer to the number of classes in the dataset, the similarity between test and true clusters increases.

Breast Cancer - Figure 2 shows the K-Means results when the number of clusters formed varies from 1 to 9. Unlike the Letter Recognition dataset, the accuracy vs number of clusters graph (second from the right) follows our intuition of the highest clustering accuracy occurring when the number of clusters equals the number of classes in the dataset, which is two. It's reasonable to assume we achieve more intuitive results with the Breast Cancer dataset because it contains less classes than the Letter Recognition dataset. Perhaps the K-Means algorithm has a harder time differentiating between 26 differing groups of datapoint as opposed to two. The homogeneity and SSE graphs are similar for both datasets (though the SSE scale is different, the overall shape of the graph is nearly identical). It makes sense for the homogeneity of the Breast Cancer dataset to be higher than that of the Letter Recognition dataset because, again, there are less classes. In other words, there are less classes to "pollute" the clusters when measuring homogeneity.

3.2 Expectation Maximization

A problem with K-Means comes to light when a datapoint is equidistant from two (or more) clusters. The hard-clustering nature of K-Means deems this datapoint can only belong to one cluster at a time, so the point will randomly be assigned to one of the clusters between labelling iterations. Expectation Maximization is a soft-clustering algorithm: a datapoint is shared between clusters, and the label given to the datapoint is determined via probability distributions and maximum likelihood estimators.

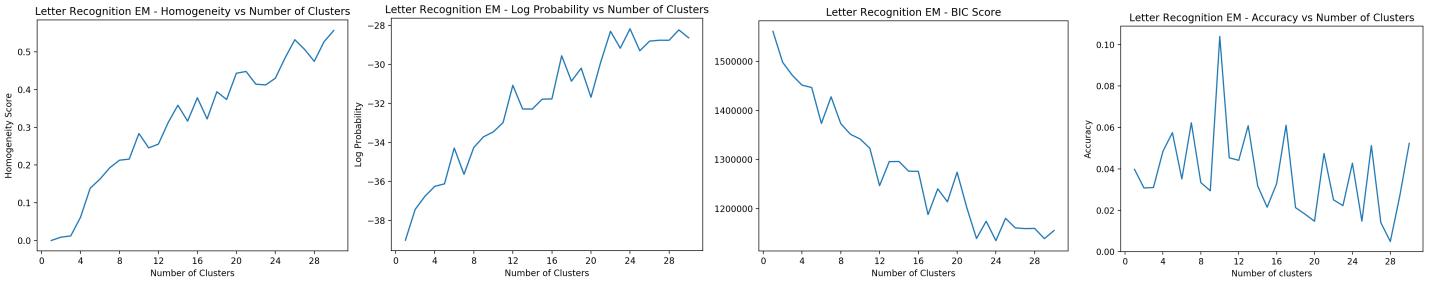


Figure 3 - Letter Recognition EM Clustering Results

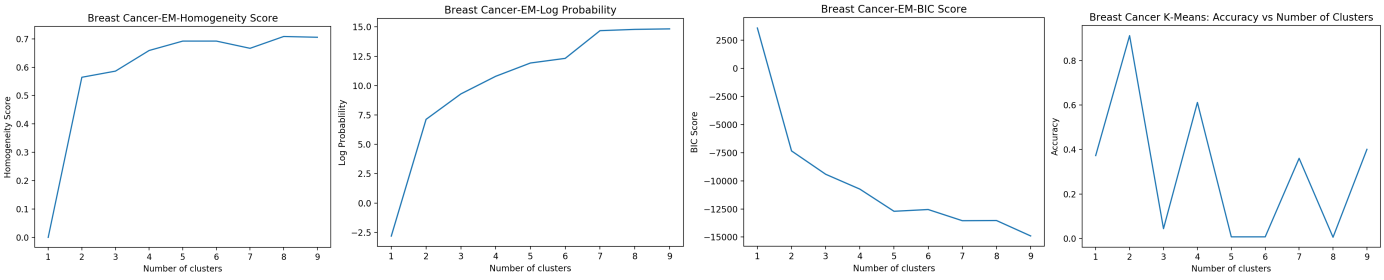


Figure 4 - Breast Cancer EM Clustering Results

Analysis

Letter Recognition - Figure 3 shows, yet again, the Letter Recognition clustering yields an accuracy vs number of clusters relationship that's a bit counterintuitive. The accuracy peaks when the number of clusters is around 10, and the shape of the graph is very similar to the graph K-Means clustering produced. And although the shapes are similar, the highest accuracy achieved in the EM graph for the Letter Recognition dataset is *lower* than that in the K-Means graph. Perhaps the "soft" nature of EM produces less accurate results compared to the "hard" clustering of K-Means. It will be interesting to see how the BIC Score is affected preprocessing the data, which will be performed in the next section of the report. My hypothesis is the BIC Score will decrease for *both* datasets when using EM (lower BIC scores are preferred when choosing models).

Breast Cancer - The rightmost graph in Figure 4 reveals EM clustering obtains the highest accuracy when two clusters are formed, which is to be expected after seeing the K-Means results of the dataset. One notion I find very interesting is in the Log Probability graph in Figure 4 (the second from the left). This measurement reflects the probability of the data given the estimated model parameters, so the higher the Log Probability the better the model. However, too high a Log Probability can leave to overfitting. As you can see, the Log Probability starts to flatten out when more than 2 clusters are created. You can see this is where overfitting starts to occur because the accuracy starts to *decrease* despite the Log Probability increasing.

4 Dimensionality Reduction and Clustering

Dimensionality Reduction focuses on decreasing the number of features in the dataset prior to classifying it with a learning algorithm. We explore four dimensionality reduction algorithms in this section: principal component analysis (PCA), independent component analysis (ICA), random projections (RP), and InfoGain (my algorithm of choice).

4.1 Principal Component Analysis (PCA)

Principal Component Analysis is a well-studied filtering Eigenproblem that chooses the best feature vector based on which one maximizes the variance within the dataset. The following principal components follow the same criteria, but they are orthogonal to each other. All the heavy lifting was performed using the Weka GUI. The output of the GUI was fed through the same Python scripts that produced the clusters and visualizations in part 3 of the report.

Analysis of Letter Recognition Dataset

Figure 5 - PCA K-Means Letter Recognition Clustering Results

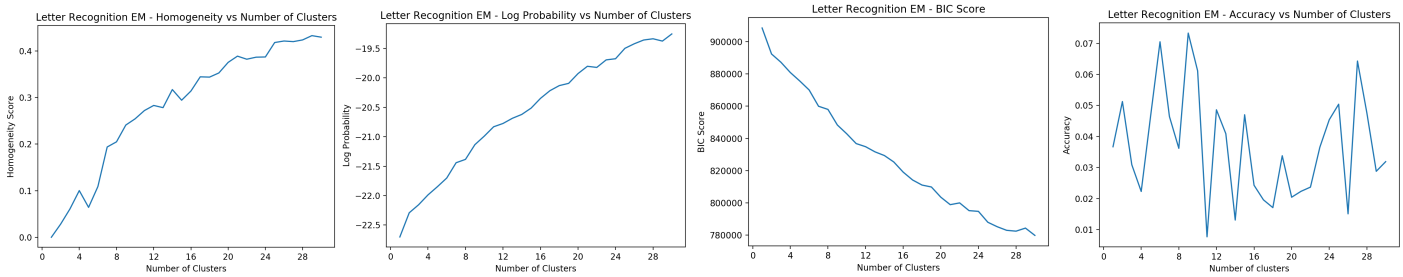


Figure 6 - PCA EM Letter Recognition Clustering Results

If the results of clustering the Letter Recognition Dataset with PCA preprocessing are compared to the baseline results featured in figures 1 and 3, it's apparent performing PCA on this particular dataset does not improve the performance of clustering. The accuracy of both K-Means and EM when PCA is used to preprocess the data is around 4% *less* than when no preprocessing is used. In addition to the accuracy, the three other performance measurements as a result of PCA are all *less* than the results of vanilla K-Means and EM. The important question to ask is why does applying PCA to the data set *decrease* the accuracy? Do these results/clusters make sense? I believe PCA changed the dataset in such a way that clustering the data into 26 classes that are more accurate than the default K-Means and EM results cannot be achieved. As I said before, there may be too many classes that have too many overlapping features for accurate clusters to be formed. The letters of the alphabet as described by the features in the dataset have too many similarities for PCA effectively alter the dataset.

Despite achieve less accurate results than vanilla K-means and EM, PCA does reveal a very useful insight with regards to choosing when to use EM versus K-means. As previously stated, EM is a soft clustering algorithm while K-means is a hard clustering algorithm. K-Means will perform well when a dataset contains clusters that have clear separation between them, whereas EM will perform better when a dataset contains clusters that do *not* have a clear separation between them. As you can see here, the accuracy of EM is greater than K-means when the number of clusters is equal to 26 (the number of classes in the dataset). This makes sense because the clusters in the Letter Recognition dataset do *not* have clear separation between them.

Analysis of Breast Cancer Dataset

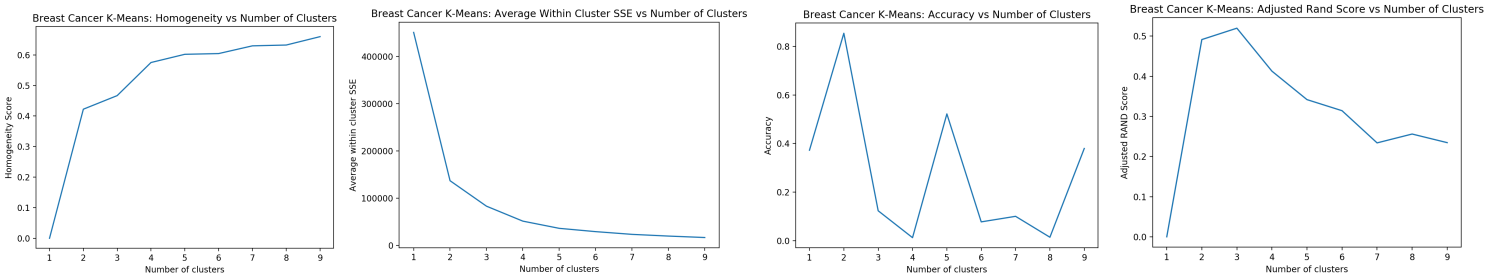


Figure 7 - PCA K-Means Breast Cancer Clustering Results

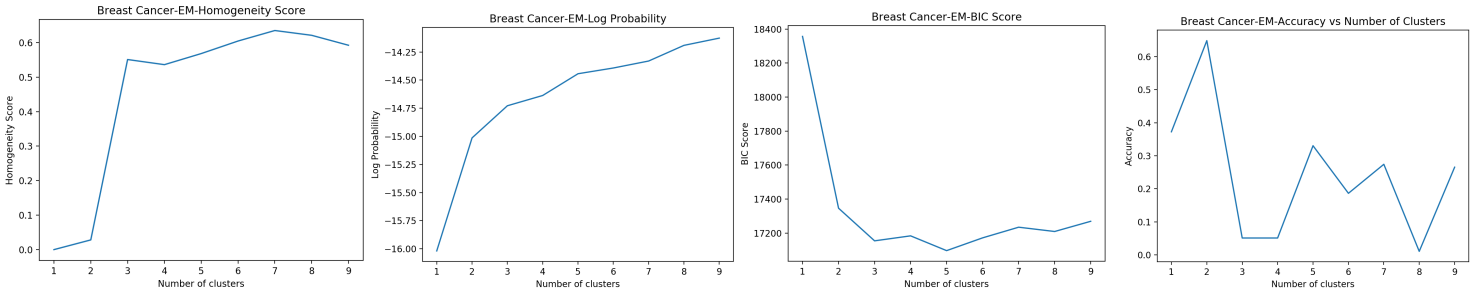


Figure 8 - PCA EM Breast Cancer Clustering Results

Performing PCA on the Breast Cancer dataset *improved* the performance of K-Means and EM clustering, unlike what happened when PCA is performed on the Letter Recognition dataset. The maximum homogeneity of both methods of clustering increased by about 5%, and, interestingly enough, only the accuracy of the K-Means clustering increased from 65% to an impressive 85%. PCA seems to alter the input dataset in such a way that favors increasing the accuracy of K-Means over EM (85% vs 65%). Why? EM clustering is to be used when there are clusters that exist in the dataset, but there is not any clear separation between them, hence why soft clustering would be useful in this situation. So, given the results communicated by Figures 7 and 8, it seems like the Breast Cancer dataset contains data that *has* features that can be combined in such a way that separability becomes more concrete.

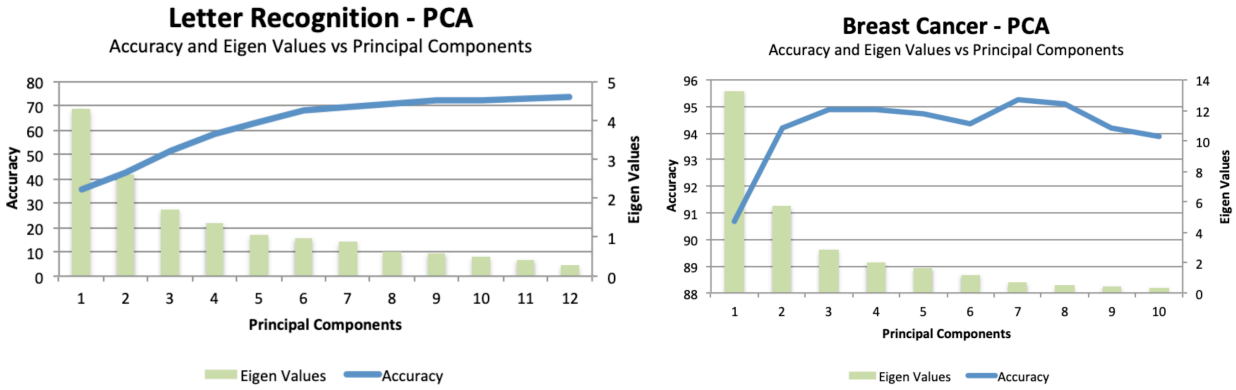


Figure 9 - Eigen Values vs Principal Components for Both Datasets with PCA Preprocessing

Figure 9 shows the relationships between the Eigen Value distribution of the PCA-preprocessed dataset and the accuracy of the model to the number of principal components in PCA. From left to right, the principal components decrease in the amount of variability they produce in the dataset. For example, a principal component of 1 (the leftmost bar in both graphs in Figure 9) means the component that produced the highest variability was used. The next bar to the right used the two highest variability-producing components, and so on.

4.2 Independent Component Analysis (ICA)

Independent Component Analysis aims to restructure the input data by increasing the separation between components by finding the basis vectors within the dataset that are statistically independent. The Weka GUI was used to implement the FastICA algorithm. This preprocessed data was then fed to the model that created the original clustering results in Section 3 of the report.

Analysis of Letter Recognition Dataset

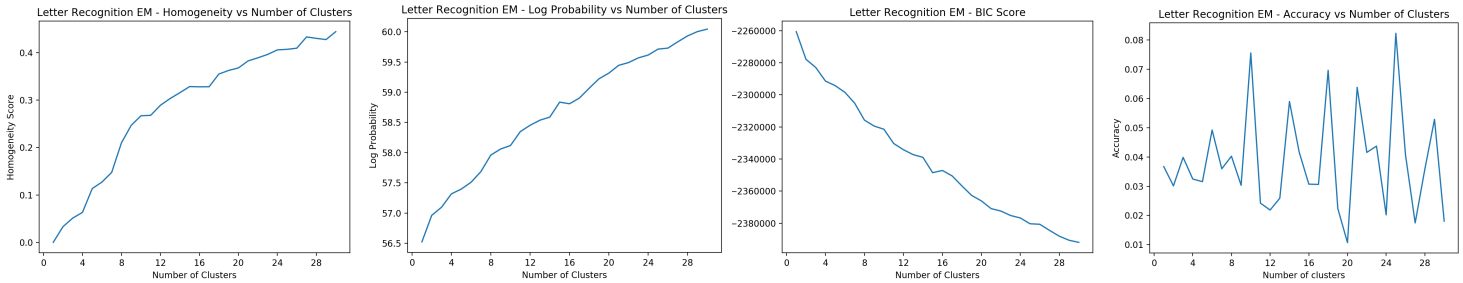


Figure 10 - ICA EM Letter Recognition Clustering Results

Figure 11 - ICA K-Means Letter Recognition Clustering Results

Performing ICA on the Letter Recognition Dataset yielded the Accuracy vs. Number of Clusters graph (rightmost graph in Figure 10). Compared to the baseline results shown in Figure 3, the maximum accuracy is achieved at a different number of clusters, 26 instead of 10. This follows the intuition of achieving maximum accuracy when the number of clusters is equal to the number of classes in the dataset. The Y-axis of the BIC score and Log Probability graphs are also *much* higher in scale compared to those in the baseline experiment as well; however, the homogeneity of the clusters decreased. These observations can be made when both K-Means and EM are used to preprocess the dataset, so this leads me to believe performing ICA on this dataset does *not* improve the performance. Why? I believe the overlapping nature of the dataset does not allow ICA to reconstruct the input data with components of high degrees of independence. The ability to find basis vectors that are independent components of the original data is hindered because the letters of the alphabet can look similar to each other under the filtering used to create the dataset.

Analysis of Breast Cancer Dataset

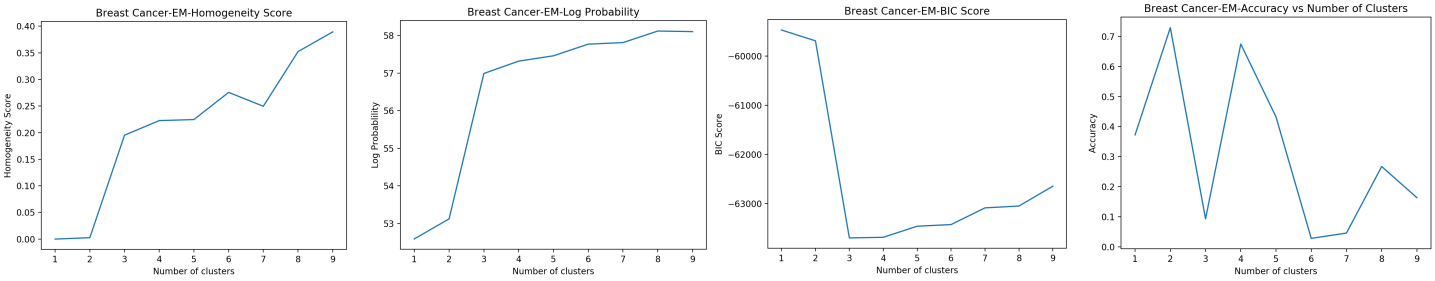


Figure 12 - ICA EM Breast Cancer Clustering Results

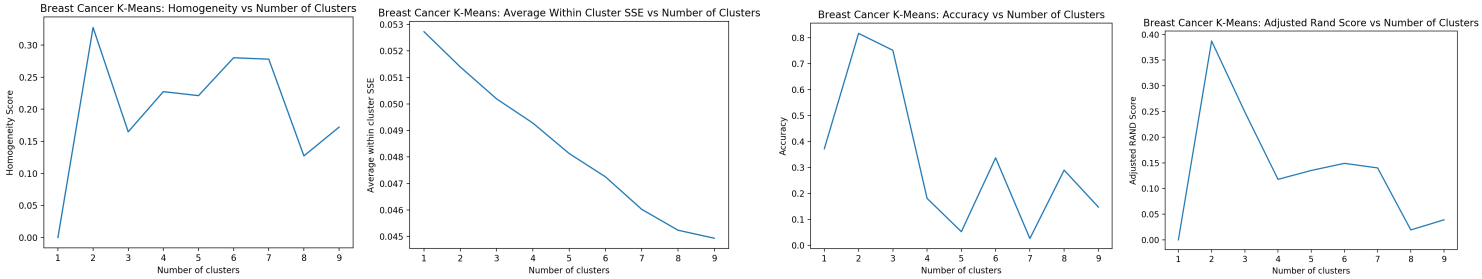


Figure 13 - ICA K-Means Breast Cancer Clustering Results

Comparing Figure 12 to Figure 4 helps visualize the effects ICA has on the Breast Cancer Dataset when EM is used. The homogeneity score is significantly lower when ICA is used, yet the accuracy of the model is higher when there are 4 to 5 clusters. This leads to me to believe the model has a hard time splitting the dataset into two clusters with like-labeled data. Unlike what we saw in the leftmost graph in Figure 4, it seems like the homogeneity of the clusters when ICA is used alongside with EM does not start to flatten out. The log probability is much higher, yet the BIC score is much lower. Both of these conditions are usually desired when choosing one model over the other, but this may not be the case because the accuracy of the model is not higher than that of the vanilla EM cluster in section 3. To my surprise, performing ICA in conjunction with K-Means clustering does *not* improve the performance of the model compared not performing any preprocessing. The homogeneity of the clusters decreases from 0.4 to 0.3 when two clusters are formed, the accuracy decreases from

about 82% to 80%, and the adjusted Rand score decrease from 0.5 to 0.4. The biggest difference between the graphs in Figure 2 and Figure 13 is the shape of the Average Within Cluster SSE vs Number of Clusters plot. Vanilla K-Means showed an exponentially decaying function while ICA K-Means showed a linearly decaying function. The sharp decrease in SSE in Figure 2 when two clusters are formed is vastly different than the SSE in Figure 13. In other words, the distance between data points when the number of clusters increases decreases at a much faster rate when vanilla K-Means was performed.

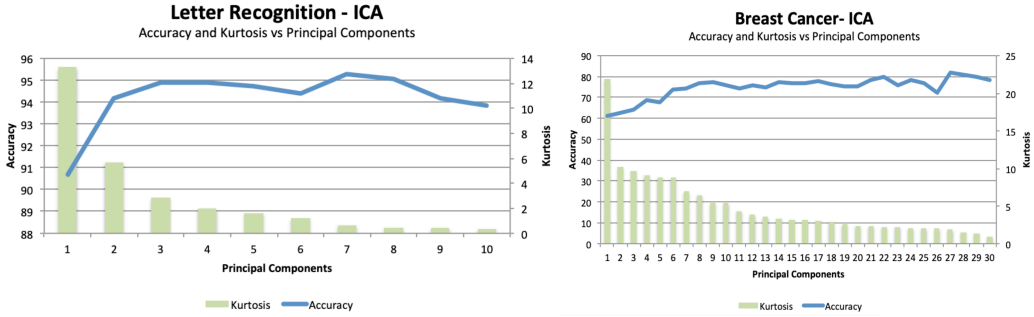


Figure 14 - Accuracy and Kurtosis for Both Datasets with ICA Preprocessing

Figure 14 shows the relationships between the accuracy of the model and the kurtosis of the distribution to the number of principal components used to produce the clusters. Kurtosis is simply a measurement of how sharp the peak of a frequency-distribution curve is. The higher the kurtosis, the sharper the peak and the lighter the tails. In other words, a curve with higher kurtosis is less outlier prone than one with a lower kurtosis. As you can see with the Letter Recognition Dataset, the accuracy of the model seems to negatively mirror that of the kurtosis values of the principal components. On the other hand, the Breast Cancer Dataset's accuracy doesn't seem to be heavily affected by the kurtosis of the distributions formed with ICA. ICA attempts to find linear combinations of sources that are as non-Gaussian as possible (the kurtosis of a normal Gaussian distribution is 3) yet still can be combined to form the original, more-Gaussian signal. The *more* Gaussian a combination of sources is the *less* help it provides when creating the clusters. This is reflected in Figure 14. As the kurtosis of the principal components approaches 3 (more towards a Gaussian distribution), the accuracy seems to change less.

4.3 Random Projection (RP)

Random Projection (RP) is another dimensionality reduction method that's used to project N total attributes onto a K-dimensional space where K is much less than N. This helps alleviate the Curse of Dimensionality and save computation costs. Depending on the random projection, the accuracy of the data may (or may not) be significantly affected. To produce the graphs in Figures 15 and 16, the Weka GUI was used to preprocess the datasets. The results of the preprocessing were fed into the same Python script that was used to create the graphs featured in Section 3 of this report.

Analysis for Letter Recognition Dataset

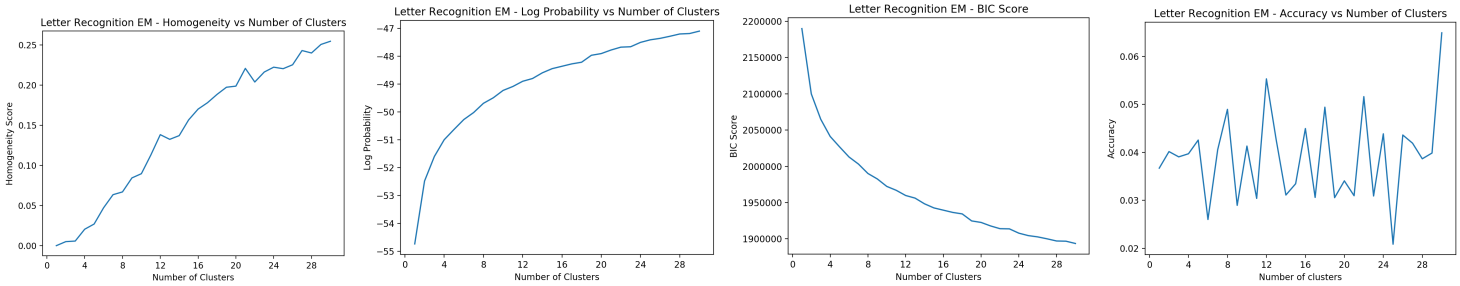


Figure 15 - RP EM Letter Recognition Clustering Results

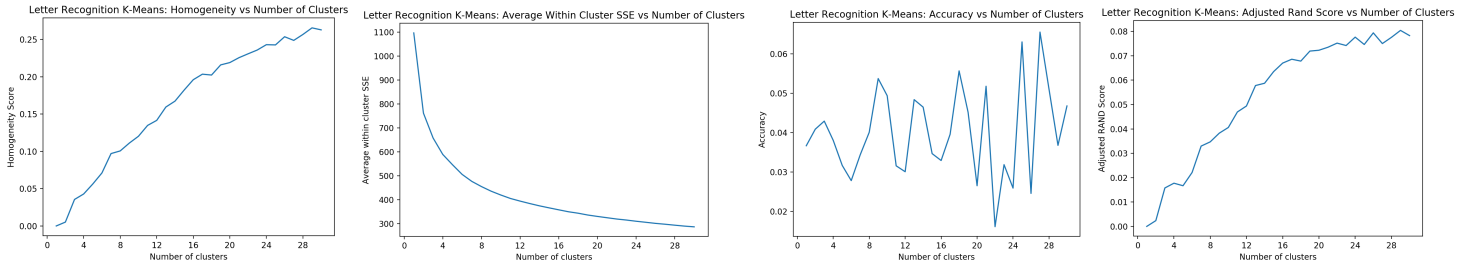


Figure 16 - RP K-Means Letter Recognition Clustering Results

The accuracy of the data was not significantly affected by preprocessing it with RP. As you can see in Figure 15 and 16, the performance of the model *decreased* for all measurements depicted in the visualizations. The homogeneity of the clusters is nearly half of what's shown in Figure 3 and Figure 2. This means the clusters are "polluted" with nearly double the incorrectly labeled data points! The accuracy is also about 0.4 *lower* when RP is used for this dataset. It can quickly be determined using RP for the Letter Recognition does *not* improve performance. However, the amount of time it took to preprocess and train the model was faster than that of regular, PCA, and ICA K-Means and EM clustering.

Analysis for Breast Cancer Dataset

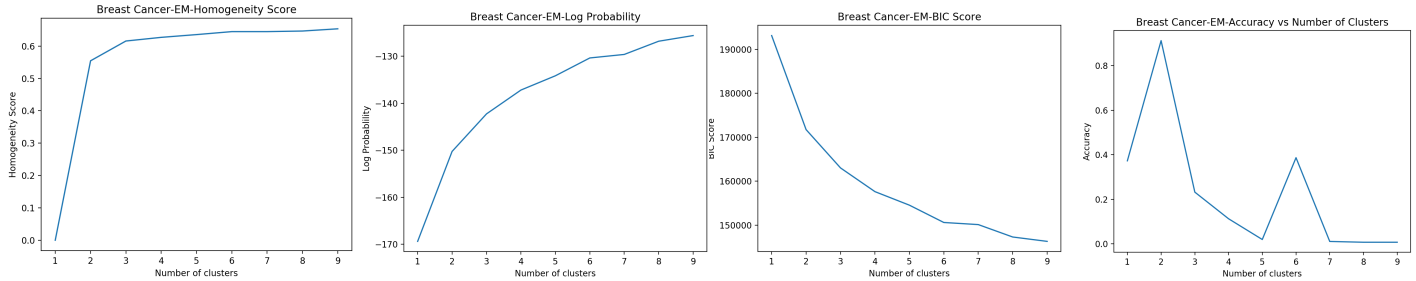


Figure 17 - RP EM Breast Cancer Clustering Results

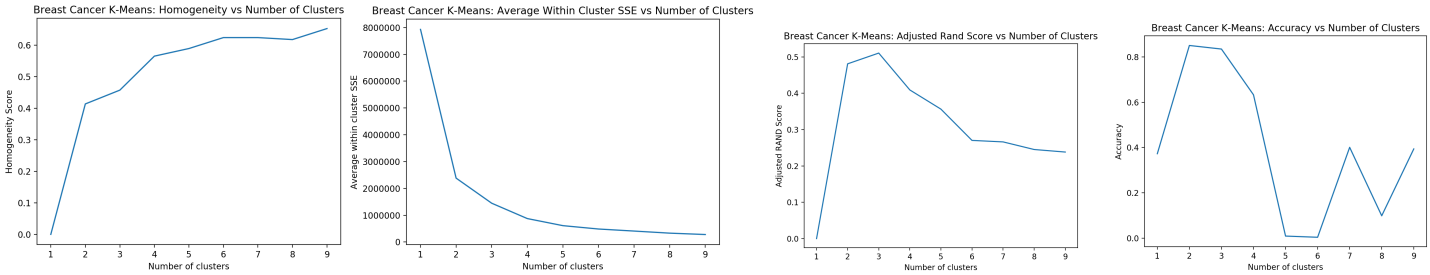


Figure 18 - RP K-Means Breast Cancer Clustering Results

Unlike the Letter Recognition Dataset, the Breast Cancer dataset *did*, in fact, experience some performance enhancements when RP was used for preprocessing. The homogeneity value when two clusters are formed is equal to that in Figure 4, and the accuracy seems a be a little bit *higher* than that in Figure 4 as well. While these improvements seem miniscule and incremental, the amount of time it took to create the clusters was less than that of vanilla K-means and EM. An important question to ask is why Random Projection works better with the Breast Cancer dataset over the Letter Recognition Dataset? Well, RP tends to do better with datasets of higher dimensions. That being said, it's no surprise RP performs better on the Breast Cancer dataset, considering the Breast Cancer dataset has nearly double the features used in the Letter Recognition dataset. PCA tends to work better on relatively low dimensional data.

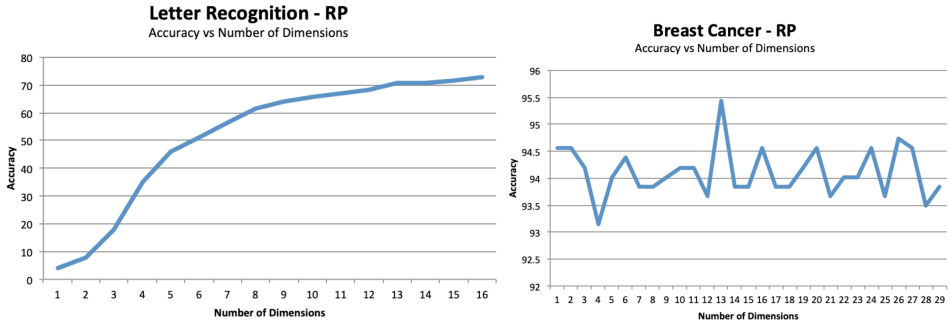


Figure 19 - Accuracy vs Number of Dimensions for Both Datasets with RP Processing

Figure 19 shows the relationship between the accuracy of the model vs the number of dimensions when RP is used for preprocessing the datasets. One would expect the accuracy to increase as the number of dimensions increases because, assuming sufficient amounts of data are provided, the more dimensions used can yield more accurate results. This is seen in the Letter Recognition dataset (the left graph in Figure 19). You can see a steady increase in accuracy as the number of dimensions used increases. However, this is *not* seen for the Breast Cancer dataset. The results shown in Figure 19 are actually the average values taken when running RP 10 times. The variation of the accuracy for the Letter Recognition dataset was as large as the variation witnessed in the Breast Cancer dataset (as reflected in the shapes of the graphs). Why does the Letter Recognition dataset show *less* variation than the Breast Cancer dataset when RP is used? I believe this occurs because the overlapping nature of the LR dataset is less susceptible to huge swings in accuracy, so random projections may not alter the accuracy of the model as much as one might think. However, for a binary classification problem like the Breast Cancer dataset reflects, where the features allow for very accurate predictions, random projections can have a huge effect on the accuracy of the model.

4.4 Information Gain (IG)

Information Gain is implemented based on information theory. Again, Weka was used to perform the preprocessing, and the data was fed into the same Python scripts that produced the graphs in Section 3 of this report. The InfoGain attribute evaluator is an internal Weka attribute selection algorithm that evaluates the "worthiness" of an attribute based on the information gain of the attribute with respect to the class of that particular data point.

InfoGain(Class,Attribute) = H(Class) - H(Class | Attribute)

This is the same concept used when decision trees evaluate information gain to determine which attribute is to be used for the top node of the tree (the attribute with the highest information gain). In simple words, information gain can be thought of as the measurement of how well an attribute splits the data.

Analysis for Letter Recognition Dataset

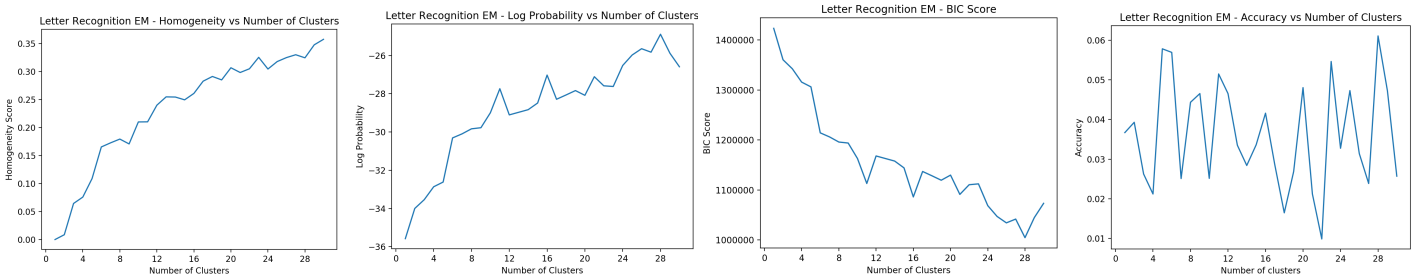


Figure 20 - IG EM Letter Recognition Clustering Results

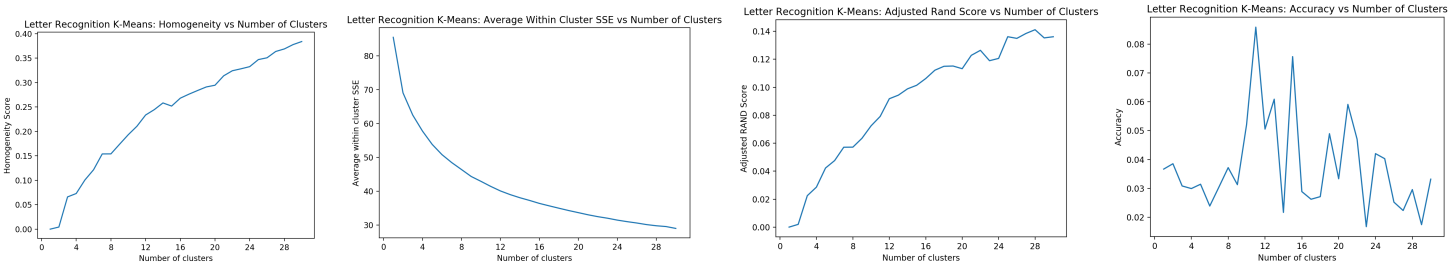


Figure 21 - IG K-Means Letter Recognition Clustering Results

Comparing Figure 20 to Figure 4, we can see the results of using IG reveal some interesting insights. The maximum homogeneity decreases from 0.5 to 0.35 when the number of clusters formed is 26. And although the global maximum for the accuracy graph in Figure 3 is much higher than the global maximum shown in Figure 20, we can see the accuracy in Figure 20 is actually *higher* than that in Figure 3 when the number of clusters formed is 26. The Log Probability between the two is very similar, so the models can be considered to have the same amount (or absence of) overfitting/underfitting. As for K-Means, the maximum homogeneity decreases from 0.6 to 0.4, and the accuracy of the model with 26 clusters *decreases* from 0.06 to 0.04. Overall, the performance of the K-Means algorithm is hindered by the use of IG preprocessing. My guess is the IG preprocessing of separating features via information gain can be thought of as another way to perform "hard" class labeling, much like a decision tree. Since this dataset has a lot of overlapping data, trying to classify data points based on information gain will not yield better results.

Analysis for Breast Cancer Dataset

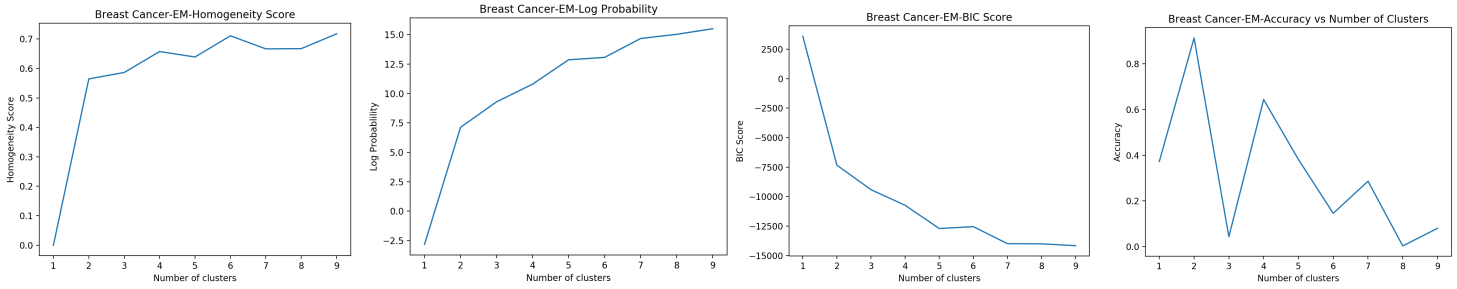


Figure 22 - IG EM Breast Cancer Clustering Results

Figure 23 - IG K-Means Breast Cancer Clustering Results

Comparing Figure 22 to Figure 3, we can see IG doesn't affect the performance of EM at all. The results are nearly identical! All four graphs featured in Figure 22 are very similar to their counterparts in Figure 3. The homogeneity of the IG-EM graph makes a large positive leap from 0.0 to about 0.56 between clusters one and two; the log probability jumps to 7.5 and starts to flatten out to 15.0, and the accuracy seems to be slightly higher than that in Figure 4 (reaching an accuracy of 0.85). The same observations can be made when comparing Figure 23 to Figure 2. Why does IG produce nearly the same exact results as vanilla K-Means and EM? I believe this happens because both IG and vanilla K-Means and EM provide the near optimal results as far as performance goes for this particular dataset. Although the InfoGain algorithm provides some data preprocessing that is absent in vanilla K-Means and EM, it may simply manipulate the dataset in such a way that the K-means and EM clustering algorithms produce clusters similar to those that would have been produced if no preprocessing were used at all. However, the time it took to create the clusters via IG was faster than when it was not used. This could be due to IG ignoring some of the features that don't provide significant amounts of information about the class labels.

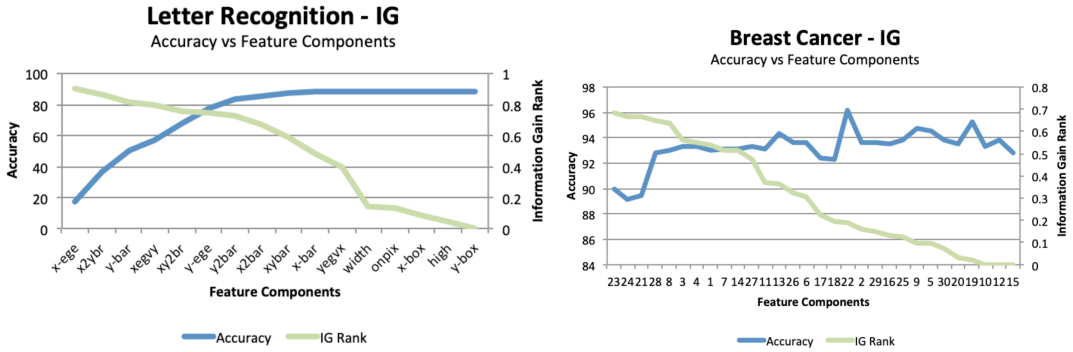
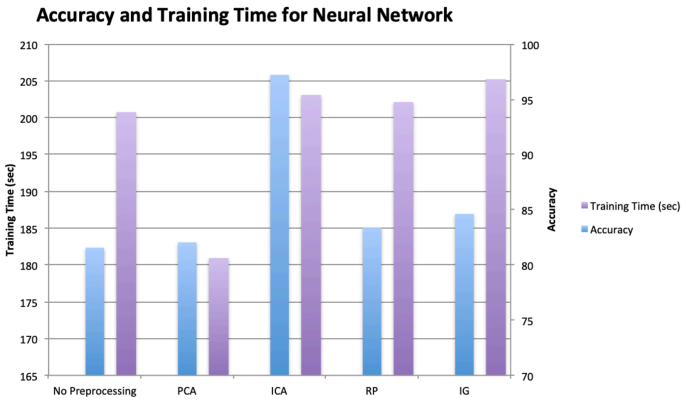


Figure 24 - Accuracy vs Feature Components for Both Datasets with IG Preprocessing

5 Neural Network Performance

Dimensionality Reduction

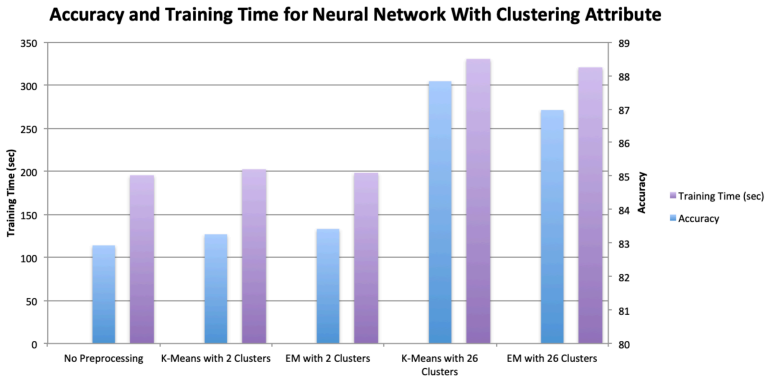
For the last portion of this report, we use the preprocessed data we obtained using the Weka GUI (PCA, ICA, etc.) to train a neural network (we also used the Weka GUI to train and test the neural network using its "multilayer perception" implementation). The network has one hidden layer containing 21 nodes (0.5 * [# of attributes + # of classes]). A momentum of 0.2 and a learning rate of 0.3, the default settings for Weka, were used for this experiment. The purpose of this experiment is to see how the neural network performs with and without preprocessing, and the results of the experiment are described in the bar graph below:



As you can see, PCA has the fastest training time of all the preprocessed datasets; however, it also has the lowest accuracy. Conversely, ICA has a much higher accuracy than any other preprocessing algorithm. Why is ICA so much better (in terms of accuracy) than the other methods? I believe ICA works better in the Letter Recognition dataset because the problem statement of the dataset itself is similar to the Blind Source Separation problem. The original 26 letters of the alphabet are distorted in different ways to create the 20,000 samples in the Letter Recognition dataset. In other words, the features of each data instance are linearly mixed to create a "new" data points to cluster. ICA is able to take these new data points and recover the original, unaltered data features. It's important to note that, in theory, ICA can only extract sources that are combined linearly. If, for example, the samples in the dataset were combined and altered in a non-linear fashion, preprocessing the data with ICA most likely would not have improved performance.

Clustering

In this experiment, we utilize Weka's AddCluster method located in the Preprocess mode of the GUI to add different numbers of clusters as attributes to the dataset. We then use this dataset to cluster the dataset into the number of clusters we see fit. The default number of clusters Weka uses is 2, so we use this value to compare the performance of clustering when the correct number of clusters is used, 26.



As you can see in the chart above, we can see a vast improvement in accuracy when the number of clusters changes from 2 to 26, though the training time increases significantly as well. The accuracy of K-Means is slightly higher than the accuracy of EM, which was experience in the previous sections of the report.

6 Conclusion

Dimensionality reduction provides certain benefits when running K-Means and Expectation Maximization clustering algorithms. Depending on the characteristics of the dataset (dimensionality, how separable the features make the class labels, etc), improvements in accuracy, runtime, and sample complexity can all be realized if the right dimensionality process is matched with the correct clustering algorithm:

Clustering

K-Means

- A hard clustering algorithm that assigns a data point to the cluster whose centroid is closest to it
- The "hard" clustering of K-Means does not make it the best clustering algorithm to run on the Letter Recognition dataset because the dataset contains data samples that are too similar to always be classified one way or another.
- This algorithm performed better on the Breast Cancer dataset, especially when it's paired with PCA preprocessing.

Expectation Maximization

- A soft clustering algorithm that assigns a probability distribution to each of the data samples. This distribution determines which cluster the data sample belongs to, which can change between queries. This allows data points to be assigned to more than one cluster, which is a main difference when comparing EM to K-Means clustering.
- This algorithm performed better on the Letter Recognition dataset because it allows data samples to belong to multiple clusters (letters/class labels).

Dimensionality Reduction

We used four different dimensionality reduction concepts in this assignment: Principal Component Analysis, Independent Component Analysis, Random Projections, and Information Gain.

Principal Component Analysis

- PCA's advantage lies in its tendency to lead to shorter training times.

Independent Component Analysis

- ICA produced the most accurate results for both datasets, and, in turn, also produced the *longest* training times. The significant increase in training time is a tradeoff I believe many people will often take when such an increase in accuracy is possible.

Random Projections

- RP produced surprisingly good results for both datasets, and because it's a relatively computationally "light" algorithm, the training times were lower than the other dimensionality reduction methods.

Information Gain

- IG helps identify the attributes that "split" the dataset the best. In other words, IG reconstructs the dataset as if it were a decision tree. This can lead to quicker training times as a result of some less "important" features being ignored during training.

Overall this report provided some extremely interesting insights towards the type of datasets and problems different clustering and dimensionality reduction techniques will produce the best results for. The two datasets chosen allowed for visualizations to communicate the tradeoffs of the mentioned algorithms. The Breast Cancer dataset yielded more accurate results from clustering, but I believe this happened because it is a "simpler" dataset. Nonetheless, this report served as a great introduction to the world of unsupervised learning, and I look forward to diving deeper into the subject on my own.

7 References

1. Data Mining Practical Machine Learning Tools and Techniques – WEKA, Ian W., Eibe F.
2. http://cs229.stanford.edu/notes/cs229-notes10.pdf
3. http://cs229.stanford.edu/notes/cs229-notes11.pdf