Monira Khan
CS4641: Machine Learning

Unsupervised Learning and Dimensionality Reduction

The purpose of this assignment to explore unsupervised learning algorithms that include clustering and dimensionality reduction. The clustering algorithms we are to explore are k-means clusters and expectation maximization and the dimensionality reduction algorithms are PCA, ICA, randomized projections, and LDA. Everything is coded in python and is heavily based on Chad Maron's GitHub code.

For this assignment, I am using the same two datasets I used before in assignment 1: the chess game dataset and the breast cancer dataset. This dataset follows a chess game of king and rook team against a king and pawn team. The database was generated and described by Alen Sharprio and supplied by the Turing Institute in Glasgow. The dataset is multivariate and categorical. This dataset is composed of three thousand ninety-six instances with 36 attributes. The classification of the data is either white can win or white cannot win. The white team is the king and rook and the black team is the king and pawn team. The white is deemed to lose if the black pawn can safely advance. The breast cancer dataset consists 699 instances of data obtained from breast cancer databases from the University of Wisconsin hospitals. Each attribute correlates to an aspect of a breast tissue cell. The aspects are: clump thickness, uniformity of the cell size, uniformity of cell shape, marginal adhesion, single epithelial cell size, bare nuclei, bland chromatin, normal nucleoli, and mitoses. The classification of the instance is either benign or malignant.

K-Means Clustering & Expectation Maximization

K-means clustering is an unsupervised algorithm that separates data points into k number of clusters/classifications. Each cluster has a center that draws the points nearby–or by which ever metric of similarity that is chosen. Those points are then assigned to that cluster, and the center is reevaluated, and the process repeated until convergence.

For the clustering algorithm used, I simply used Euclidean distance as a metric of similarity. For the breast cancer data, k is equal to 2 because the data is either classified as benign or malignant and the same goes to the chess game data–win or no win. I graphed the number of clusters against their accuracies:

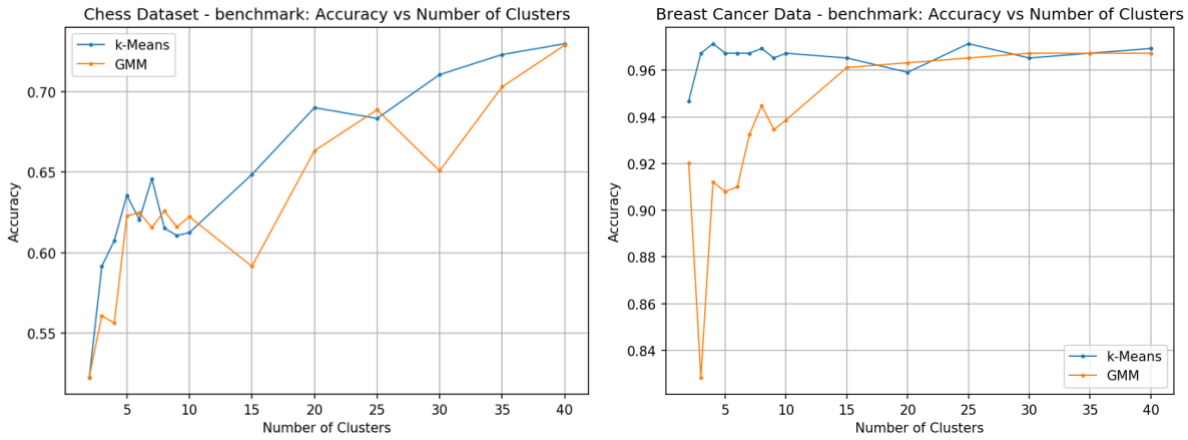


Figure 1: Accuracy vs Number of Clusters

For the chess dataset, it seems accuracy increases with the number of clusters. It seems that the clustering method performs very well with the breast cancer dataset, even with a low number of clusters. This makes sense, because clustering methods is often used for the visualization of cancer research. With how the cancer data behaves, all the attributes contribute to whether or not the cancer is benign or malignant. Cells that are similar will most likely have the same classification. But this is not the same case for the chess dataset. In the chess dataset, the games could go almost the same, but one move would cause the white team to win or lose. Only one move would change the entire game. I believe for that reason clustering works best for the breast cancer data and not for the chess game dataset.

Another interesting trend to note is the silhouette of the clusters quantifies the similarity of the data points to their cluster. The scores are graphed below:

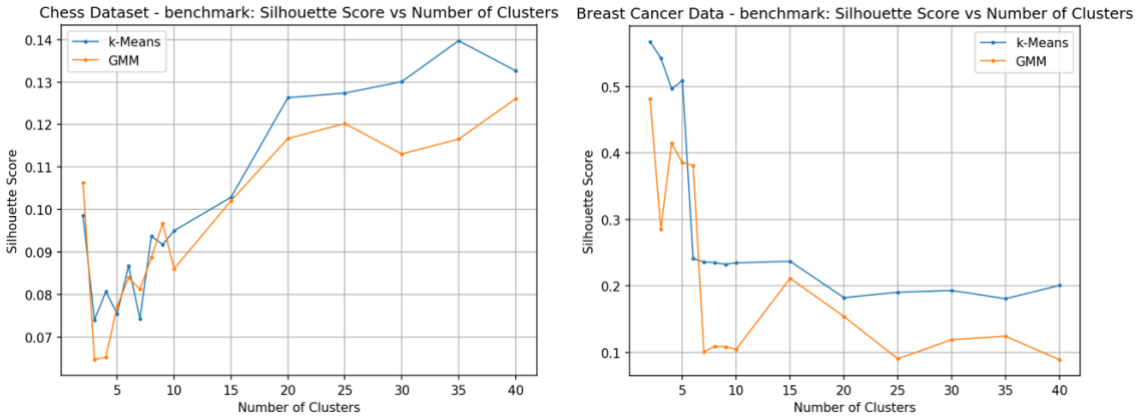


Figure 2: Silhouette Score vs Number of Clusters

For the chess game data, when clusters increase, the silhouette score increases while for the breast cancer data, the score decreases. What's happening with the breast cancer dataset is quite

counterintuitive–as more clusters are made, the clusters are seemingly attracting data points that are less and less similar to the cluster itself. For the chess data set, the clusters are getting better and better as the number increases.

In order to better visualize the clusters, I graphed the clusters by t-distributed stochastic neighbor embedding. The visualization algorithm models the data points in a way where similar points are close and dissimilar points are distant. The visualization of the clusters is provided below:

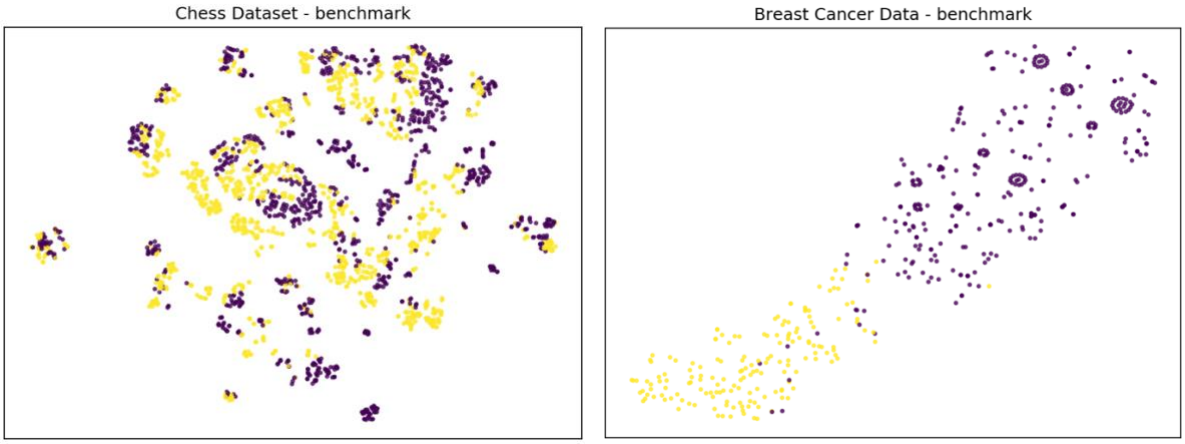


Figure 3: t-SNE clusters

As can be seen from the images above, the breast cancer clusters are much more separated out than the chess game dataset clusters. The chess game clusters are intertwined and doesn't look to be easily separable. While the chess game dataset clusters look similar to data that would be linearly separable. This makes sense, along with the fact that the clustering works well with the breast cancer data in general in relation to accuracy because k-means clustering weakness lies in noisy data, data with outliers, and nonlinear data.

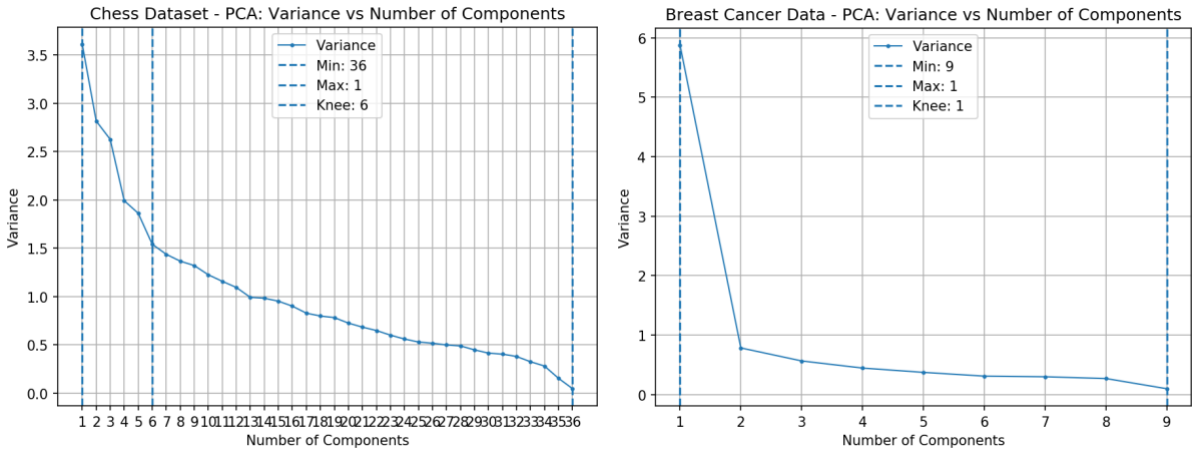
For *expectation maximization*, the algorithm starts by deciding the best clustering/grouping method for the data. The algorithm repeats two steps: expectation and maximization. In the expectation step, random parameters are used as estimated parameters which are used to generate a log-likelihood function. In the maximization step, the algorithm reevaluates the parameters to maximize the function of expected log-likelihood found in the expectation step. These two steps are repeated until the change in the function is below a small threshold. EM allow data points to technically belong to more than one cluster–an ability that k-means clustering does not have. EM does this by allowing a "degree" of membership to certain clusters.

You can see the performance of EM on the data from the graphs given above–Figure 1 and Figure 2. For both datasets, in general, it seems that k-means clustering performs better than EM in terms of accuracy and the silhouette score. For both data sets, EM and k-means accuracies converged around 40 clusters, but the EM clusters are not as good as the k-means clusters due to the silhouette score. The k-means clusters are purer than the EM clusters.

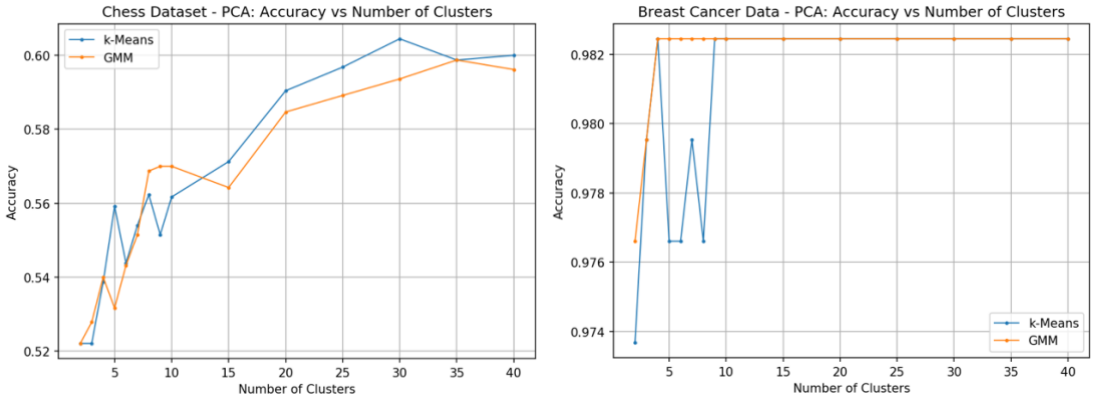
Principle Component Analysis

PCA is dimensionality reduction algorithm that analyzes linear relationships between variables. The statistical procedure utilizes an orthogonal transformation to convert of a set of observations of possibly correlated attributes into a set of linearly uncorrelated variable principles, termed principle variables.

Running PCA produced the graph that I provided below:



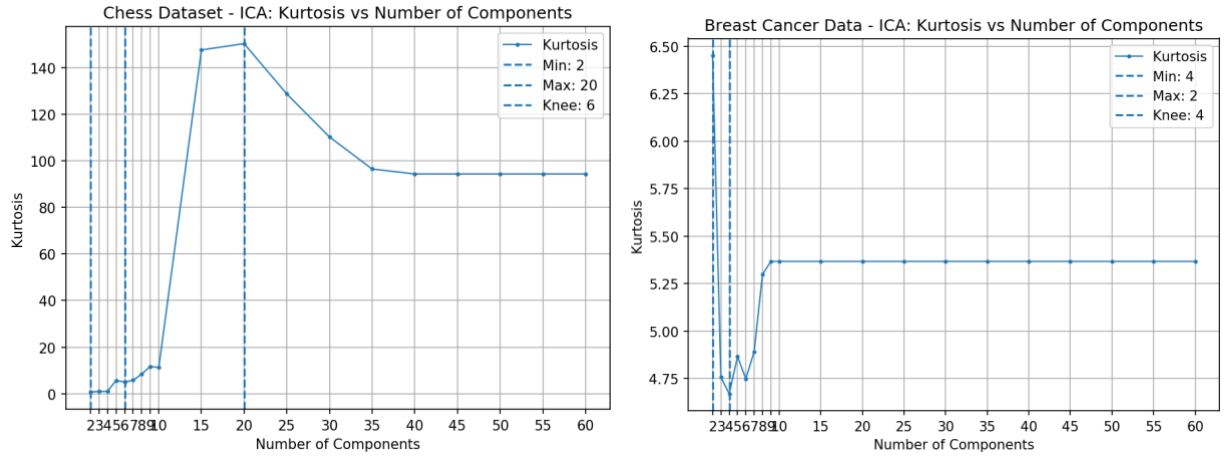
For the chess dataset, there is a gradual drop in variance, while for the breast cancer dataset, the drop-in variance is extremely sharp from 1 component to 2. The optimal values from the reduction is where the variance is the least, the min. So, taking the min value and running the clustering algorithms again produces the following results:



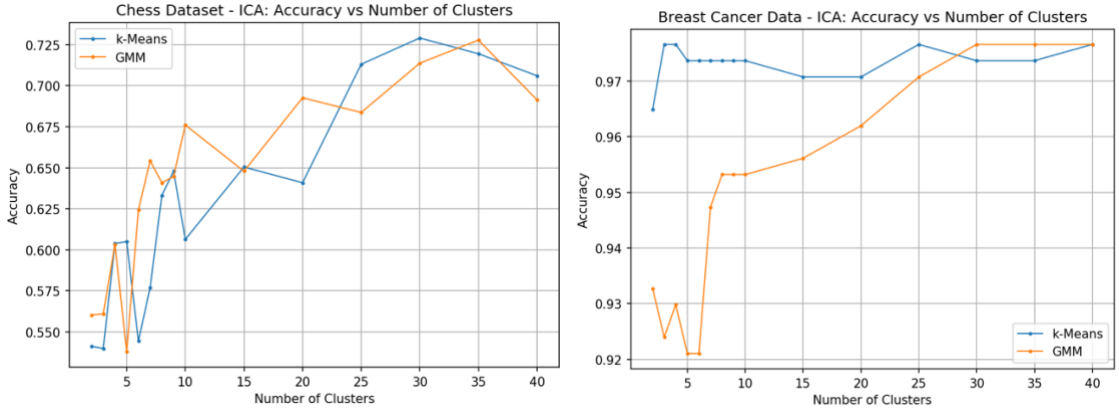
Comparing these figures to figure 1, it seems for the breast cancer dataset, accuracy has increased–especially for the EM method, but for the chess dataset, accuracy has decreased, but the same climbing pattern remains.

Independent Component Analysis

ICA works to separate linearly combined components into subcomponents, by assuming that they are statistically independent of one another. How ICA works can be compared to the "cocktail party problem" where people try to discern one signal in a noisy room. This works best with the data sources are linearly combined and has non-Gaussian distributions. Running ICA on the datasets produced the following figures:



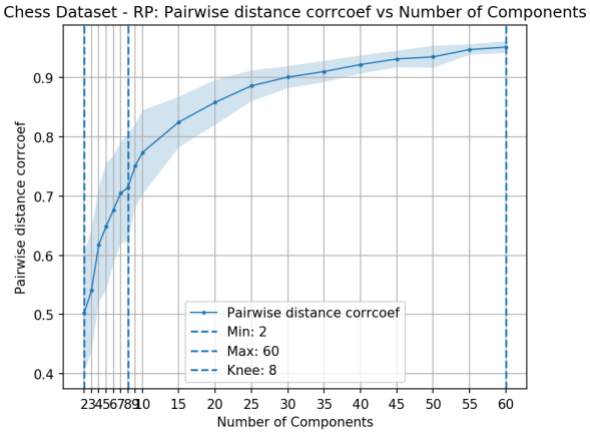
Observing the kurtosis behavior is quite interesting, so they are quite different for both datasets. The chess dataset seems to have extremely high kurtosis–or tailedness–in comparison to the breast cancer dataset. Taking the optimal number of components and running the clustering algorithms again produces the following graphs:

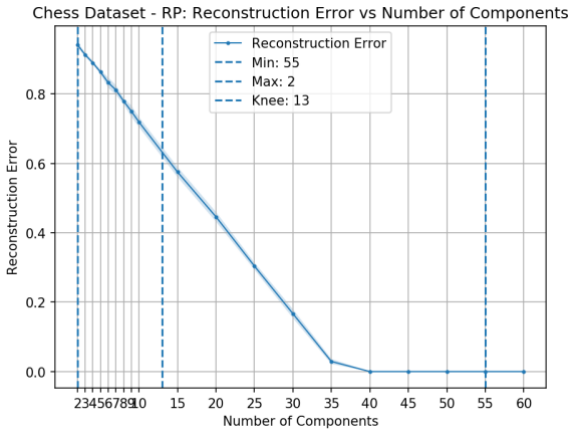


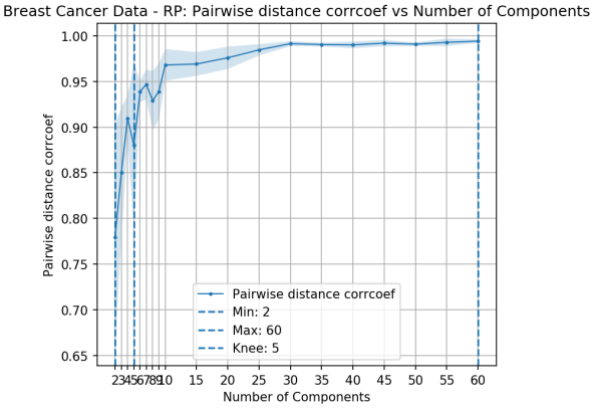
It seems that the clustering algorithms performed slightly worse on the chess data set in general and slightly better on the breast cancer data set in relation to Figure 1.

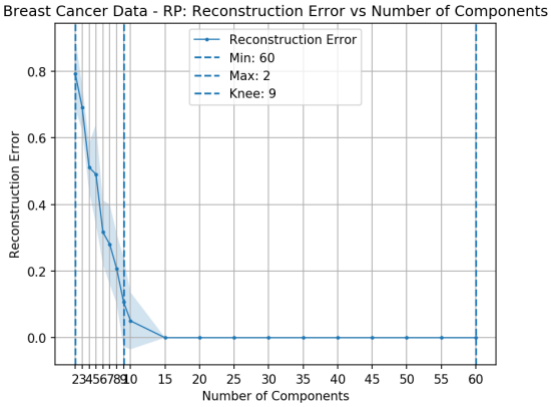
Randomized Projections

The idea behind randomized projects is project high-dimensional data onto a lower dimensional space–in reducing dimensionality and preserving distances among the points. Running RP on my datasets produced the following graphs:

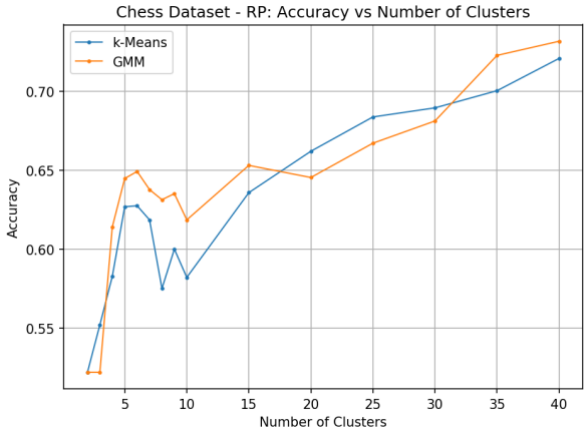


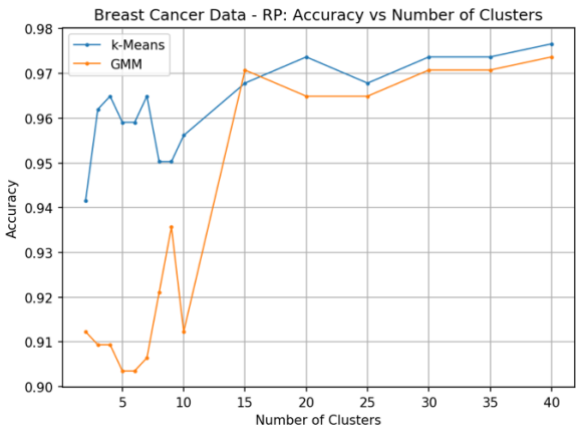






Reconstruction error–the error of the new lower dimension–decreases to 0 as the number of components increase. Plugging in the optimal number of components for RP and running the clustering algorithms again produces the following graphs:





It seems like the clustering algorithms performed nearly the same, even with dimensionality reduced. This is quite interesting–that the perform neither got way better or way worse in relation to Figure 1.

Re-running ANN on Newly Projected Data

Rerunning the neural networks on the newly projected data according to ICA, it is observed that for both data sets, the neural network took much less time to run. Significantly less time to run, but still took a while, due to the nature of the neural network, I presume. But, the accuracy of the of the neural network suffered for both datasets. Neural networks take advantage of numerous dimensions and attributes in order to achieve maximal accuracy and reducing the dimensionality of the data takes away from that.

Conclusion

It seems, for these certain datasets, that reducing dimensionality worked towards no improvement. This may be because the data sets are already quite simple and have all the data attributes and dimensionality necessary for accuracy. It was extremely interesting, though, the differing impacts of clustering on the two data sets. Clustering was quite successful with the breast cancer data set which is quite telling of the data, while not so much with the chess game.