Analysis of Unsupervised Learning for Clustering and Dimensionality Reduction

Andrew Shi

CS 7641

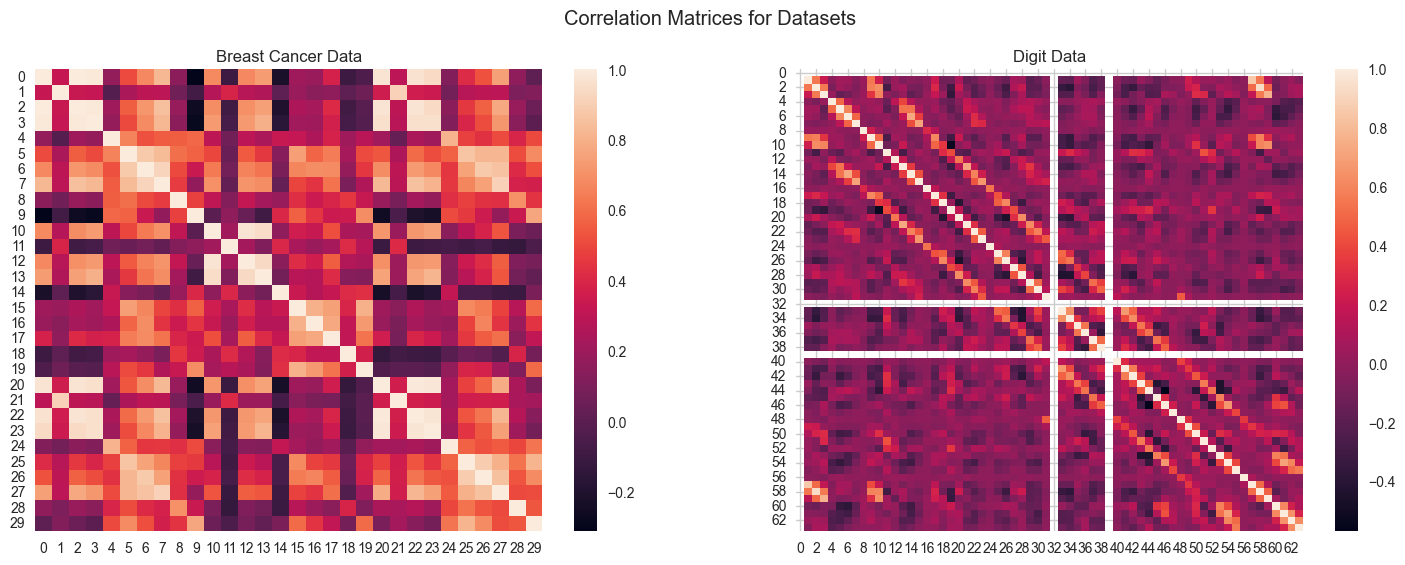
November 7, 2023

1. **Introduction**

In this assignment, we will explore clustering and dimensionality reduction algorithms. Two clustering algorithms – expectation maximization and k-means – will be fit with two chosen datasets; then, four dimensionality reduction algorithms (DRA’s) – principal component analysis (PCA), independent component analysis (ICA), randomized projections (RP) and Isomap - will be applied to the two datasets, resulting in eight reduced datasets, which will be fit using the two clustering algorithms Finally, two reduced datasets will be selected to be fitted for a neural network (NN), and then refit when the clusters of step one are included as features.

1. **Datasets**

Two new datasets were selected for this assignment, both from sklearn libraries. In order to explore the full extent of DRA’s, I found it best to select two datasets that exhibits high and low correlation, respectively between their features. Hence, the two datasets I selected were the breast cancer and digit data. Figure 1 shows the correlation matrices of the two sets.



*Figure 1. Correlation matrices for both datasets.*

The correlation between every pair of features is shown, hence the high correlation along the main diagonal and the symmetry across it. Note the relatively high contrast in the correlation matrix of the first dataset, with many pockets of values above 0.6, compared to the relatively low variation in correlation for the second dataset. A rough aggregate measure of correlation was taken by averaging the absolute value of the upper triangle of each matrix, resulting in 0.190867 for the breast cancer dataset, and 0.054845 for the digit dataset.

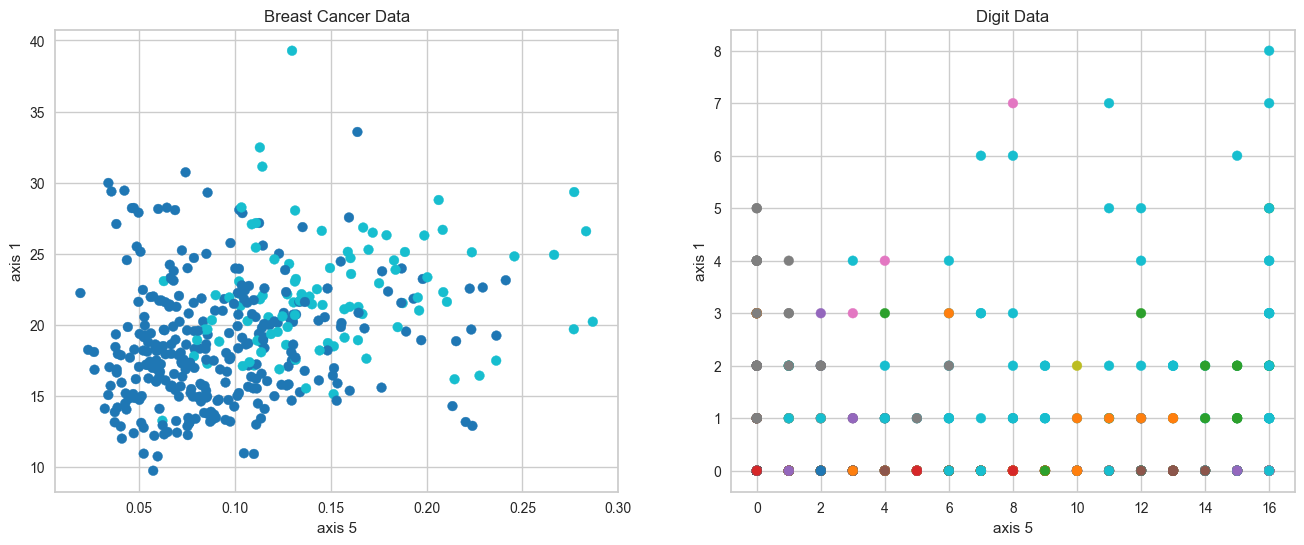
The breast cancer dataset contains 569 instances with 30 features describing characteristics of cell nuclei in an image of a breast mass. Of all instances, 357 are classified as benign, and 212 are classified as malignant. All 30 features are characterized by continuous positive real numbers, and are fall into 10 categories, such as radius, texture, perimeter, etc. over three cell nuclei per instance. The repetitive measurement over several cell nuclei in a cluster make this dataset a prime subject for dimension reduction.

The digit dataset contains 1797 instances with 64 features each an integer from 0-16 representing the greyscale value of a pixel in an 8-by-8 square. The instances are separated into 10 classes, based on what numerical digit the square represents. In addition to having relatively low correlation between features of this dataset, I thought it would be interesting to analyze the dimensionality reduction algorithms on a high-dimensional dataset. Unlike the first dataset, the features in this set are discretized, which may lead to interesting results when comparing DRA’s.

1. **Clustering**

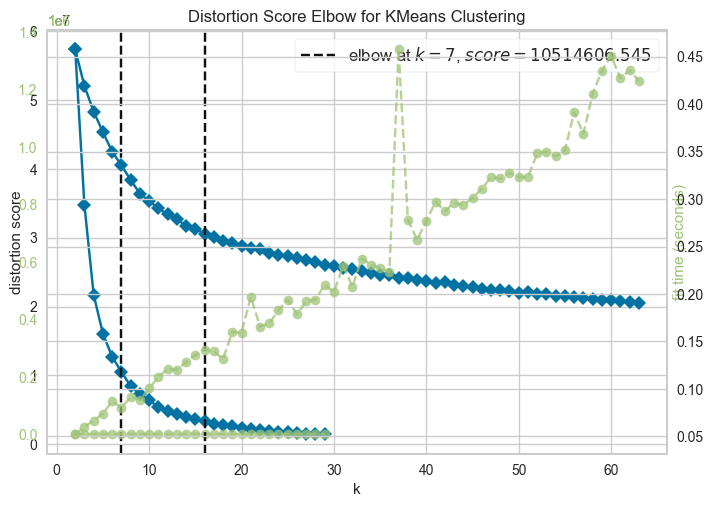
In this assignment, two clustering algorithms will be analyzed: expectation maximization and k-means. Expectation maximization (EM) alternates between two steps: an expectation stage, where an estimate of log-likelihood is made given current clustering, and a maximization stage, where log-likelihood is maximized by changing clustering parameters. This section was done in an unsupervised fashion hence ground truth labels are not considered in dimension selection. Future sections of this assignment analyze supervised learning on a neural network, so this until then, all results are captured after separating out 30% of the dataset as a test set.

I used the Gaussian Mixture Model implementation in sklearn. To determine the number of clusters, the algorithm was run with cluster numbers from 2 up to the number of features in each dataset. Then, the silhouette score is calculated for each model, where silhouette score is calculated from the mean intra-cluster distance and the mean nearest-cluster distance for each sample. Values of silhouette score close to one represent good clustering with highly separated clusters is represented by a 1, and mislabeled data is represented by a -1. The number of clusters that produces the best silhouette score is then noted. For the two datasets, silhouette scores of 0.388170 and 0.175466, achieved with 2 and 11 clusters, respectively. This method perfectly predicts the number of targets for the breast cancer data, and is quite close to the number of targets for the digit data. The error in the number of predicted clusters and the actual number of targets is due to the quantization of the digit data. Because the set contains almost 1800 instances, and each feature can only take on 16 values, there is a great deal of overlap between the true clusters themselves, compared to the continuous nature of the values allowed for the breast cancer data. Figure 2 shows the clustering produced by GMM fit using the suggested number of clusters. Note the discretization of dataset 2, and the high amount of overlap in data and clusters, hence the lower silhouette score.



*Figure 2. GMM clustering of datasets.*

The second of the two algorithms is k-means. Like EM, k-means also operates in two stages: k means are calculated to be at the mean of their clusters, and then each point is evaluated to be a part of the mean closest to it. For this assignment, I used the KMeans implementation from sklearn. To select the number of clusters, the distortion score was calculated for a range of number of clusters, and plotted. In this case, distortion score simply is an average of the squared distances from cluster centers to their respective clusters. Of course, the distortion score converges on zero when the number of clusters matches the number of instances, so an elbow value is taken as the optimal number of clusters. Figure 3 shows the elbow selection and distortion score of the two datasets.



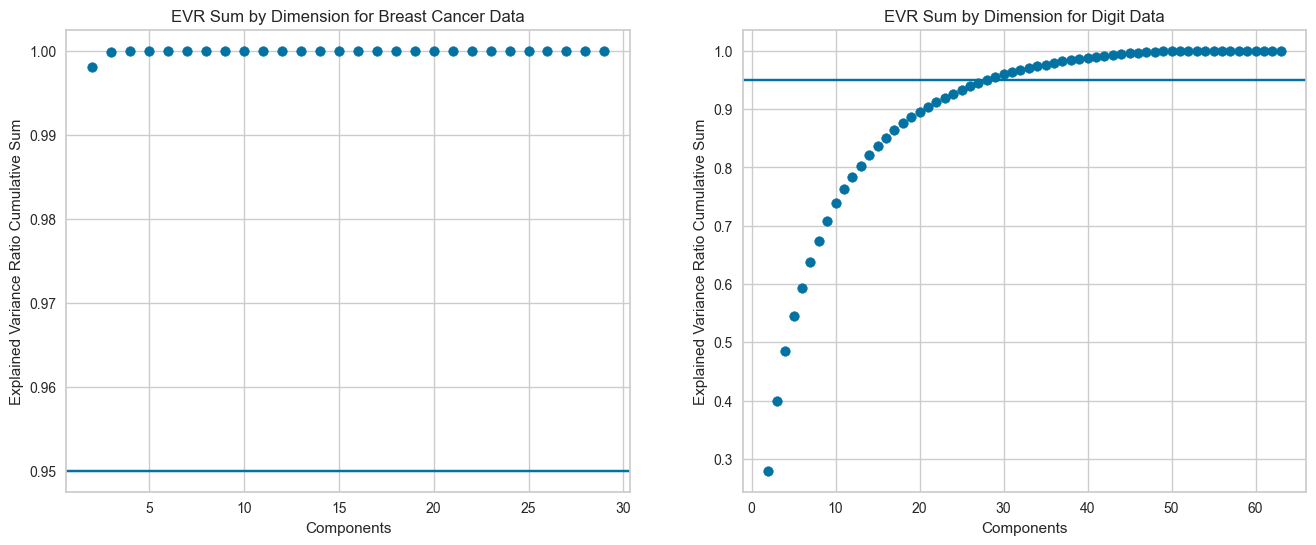
*Figure 3. Elbow plot and k-selection based on distortion score.*

The number of clusters selected for the two datasets is 7 and 16, respectively, both overestimations. One drawback of the elbow method is the lack of a clear elbow, evident in both graphs above, as the implementation use captures the point of maximum curvature. For the breast cancer dataset, increasing cluster numbers at low values drastically decreases the distribution score and the true target number is skipped, which is often the case with few target values.

1. **Dimensionality Reduction**

The aim of dimensionality reduction is to represent the datasets in a lower number of features while retain the meaningful properties of original data. Four dimensionality reduction algorithms were considered in this analysis. This section was done in an unsupervised fashion hence ground truth labels are not considered in dimension selection.

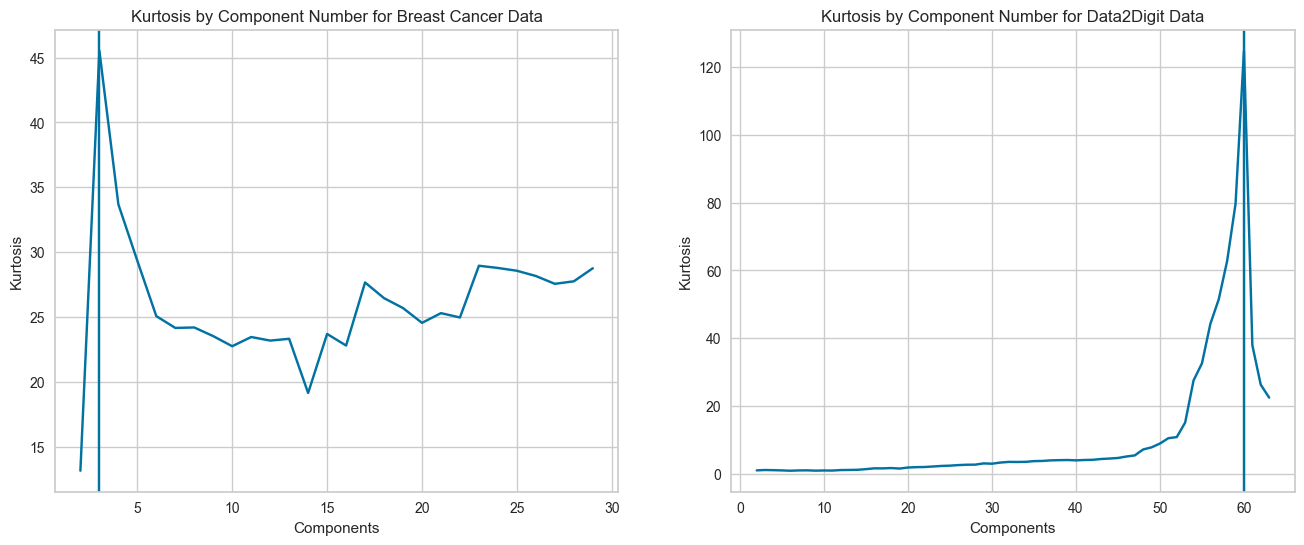
PCA is a dimensionality reduction technique that establishes an orthonormal basis that minimizes variance when compared to the original dataset. The cumulative variance explained using the first *n* eigenvectors can be plotted, where *n* is the number of components in the basis. The cumulative ratio of variance explained is shown for both datasets below.



*Figure 4. Explained Variance Ratio Cumulative Sum*

The blue line on each plot represents the threshold of explained variance ratio (EVR) cumulative sum, which was set to 95%, at which point the number of components necessary is satisfactory. This analysis yields a dimension reduction of 2 and 29 for the two datasets. Note that the EVR sum for the breast cancer data is adequately explained in a very small number of components, as expected from looking at its correlation matrix and hypothesizing on its redundant data. On the other hand, the digit data seems to have lower individual EVR’s per component, resulting in a slow increase of EVR cumulative sum. Again, this is expected from the low correlation.

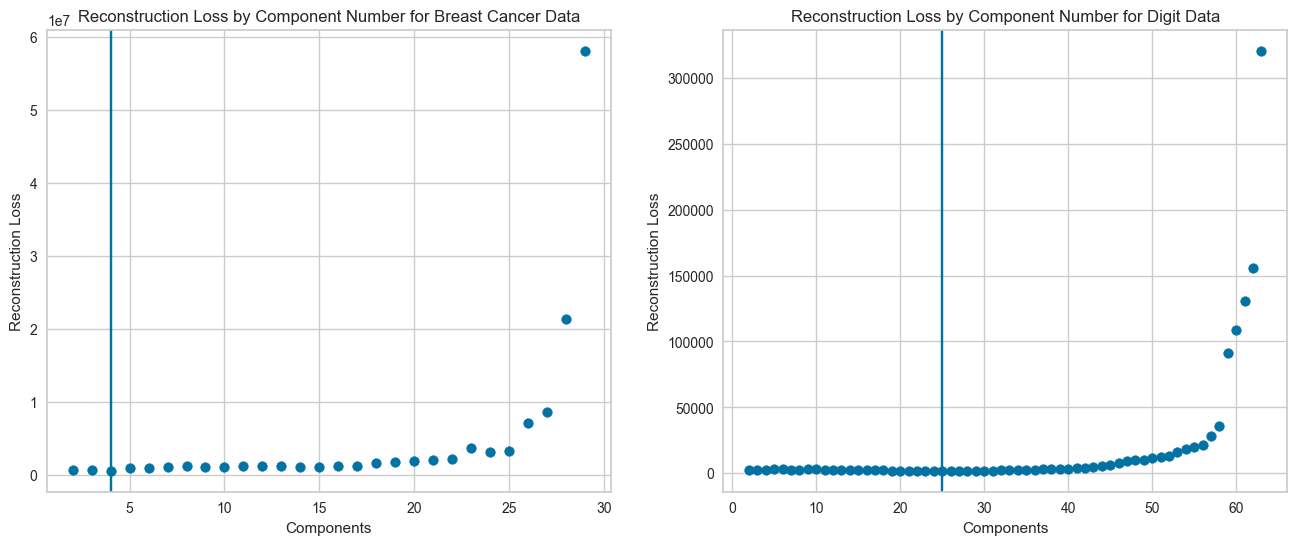
Unlike PCA, ICA is a DRA that attempts to separate a given signal into its independent subcomponents. Because ICA seeks to maximize statistical independence between subcomponents, it seeks to maximize the “non-gaussianness”, which is encapsulated by average magnitude of kurtosis of its components. Figure 5 shows the kurtosis scores as a function of number of components.



*Figure 5. Kurtosis score by number of components.*

The number of components that maximizes kurtosis is 3 and 60, respectively. For the first dataset, we again see that dimensionality reduction does a fantastic job isolating independent components, evident by the decrease from 30 original datapoints to 3 after the transformation. However, when compared to the second dataset, with low inter-feature correlation and a profusion of overlap between data, it seems difficult for kurtosis to identify independent components. In fact, it seems that almost every feature is independent of each other after reduction.

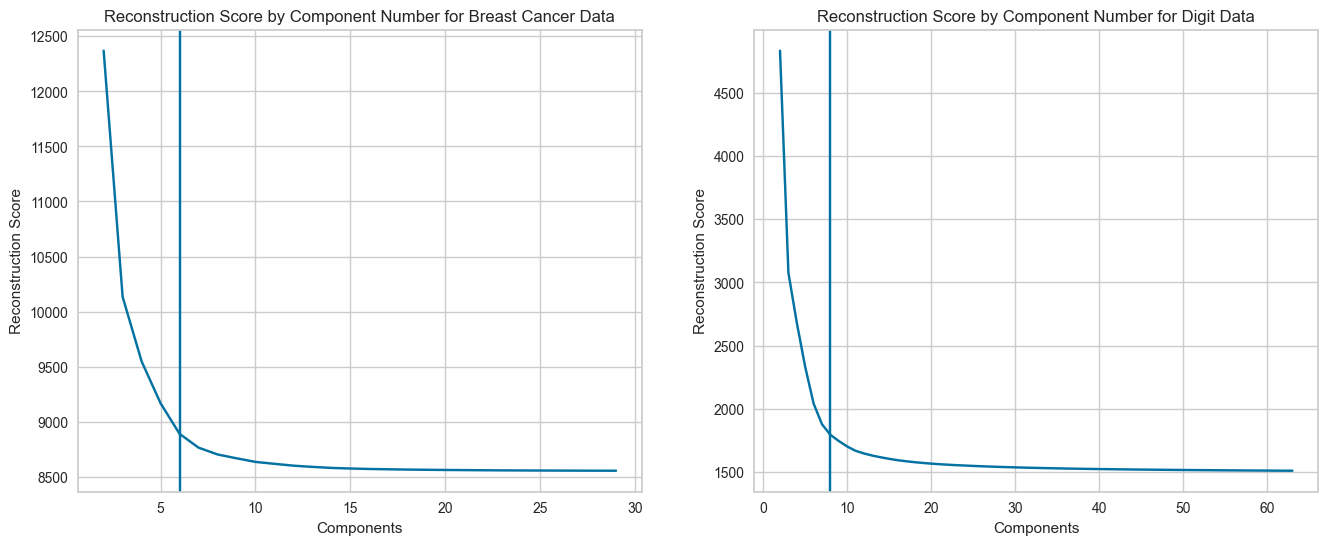
A Gaussian Random Projection is a DRA that, like its name suggests, uses a Gaussian distribution to generate a random matrix of mutually orthogonal vectors to project the original data. Johnson-Linderstrauss lemma posits that for data in a sufficiently high dimension, they can be projected into lower dimension in a way that distances are approximately preserved. For our high dimensional data, we would expect good preservation of data. To select the number of components to project to, the reconstruction score is calculated by “un-projecting” the data from the lower dimension back to the higher using the inverse transform matrix, then calculating the mean-squared error over the instance space.



*Figure 6. Reconstruction error as a function of number of components for both datasets. The minimal value is indicated.*

Figure 6 shows the reconstruction error as a function of number of components. The error shown is actually the average a number of random seeds, as randomness is a driving force behind RP. Unlike PCA, whose error strictly increases as the number of components increases, RP tends to de very poorly when the number of components approaches the number of features. This is evident by noting that when the number of components is equal to the number of features, PCA will select vectors that capture variance in the data, while RP selects vectors with no bias, and hence selects poorly. For the two datasets, the reconstruction loss is minimized when 4 and 25 components are used, respectively. For the first dataset, RP does very well, minimizing the reconstruction score only using 4 features, following the performance of PCA and ICA. For the second dataset, note that reconstruction loss is minimized with less components than both PCA and ICA, as would be expected from the Johnson-Linderstrauss lemma above.

The final DRA analyzed in this analysis is Isomap, which unlike the others, is a nonlinear DRA, meaning it has the capability to capture geodesic distances between all points. For datasets like the breast cancer one, with periodic correlation across features, we expect Isomap to be able to capture periodicity due to multiple cell nuclei measurements. The reconstruction score if found and plotted for each dataset in figure 7.

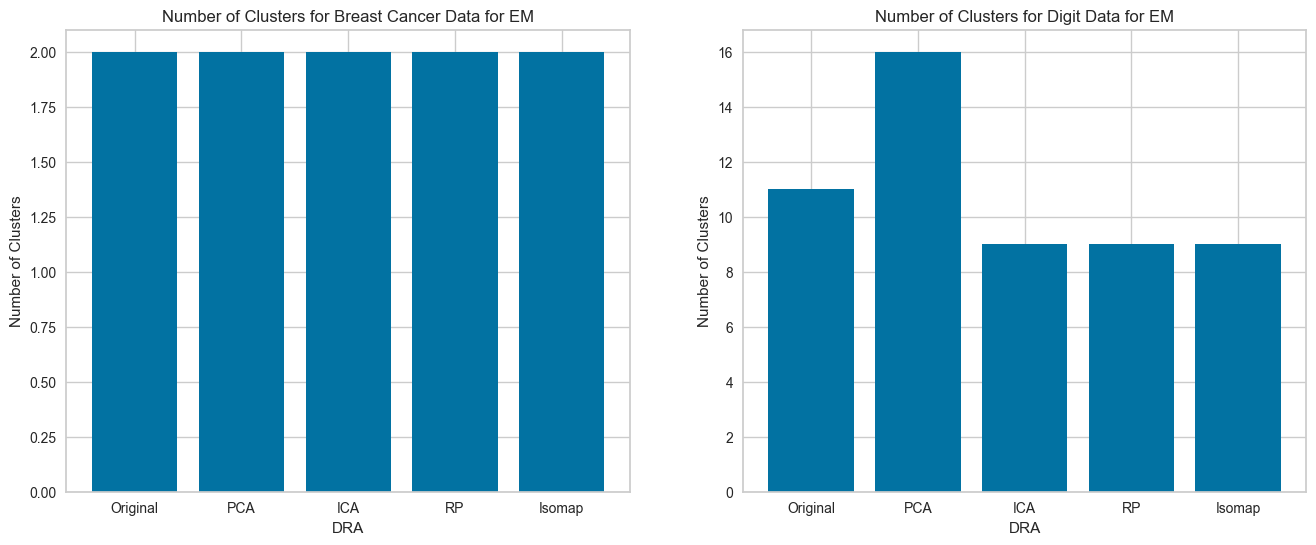
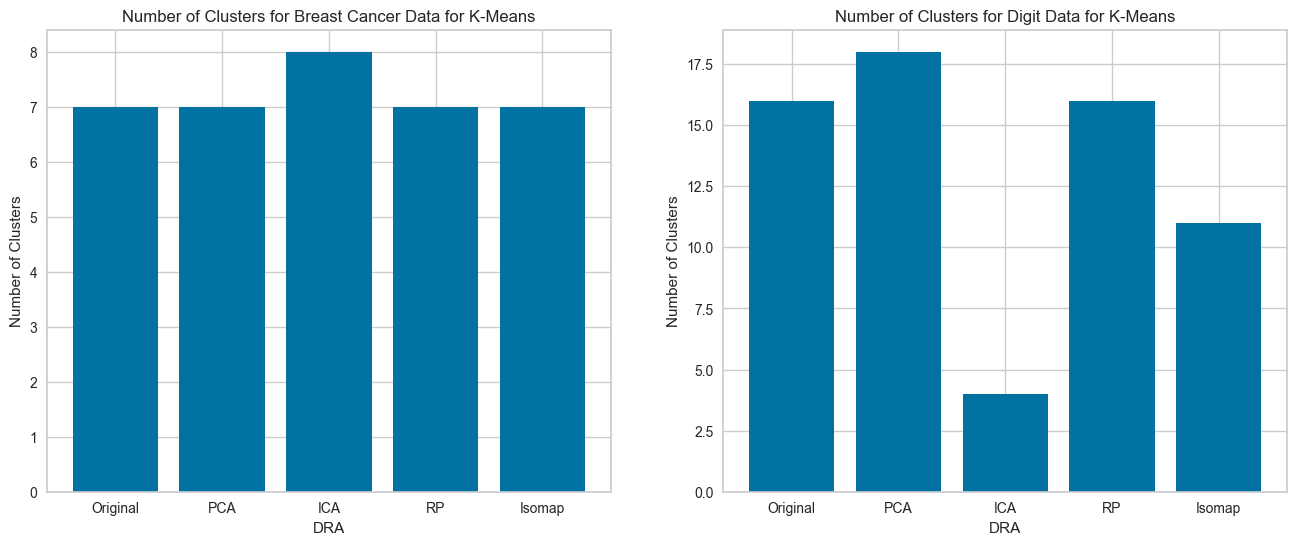


*Figure 8. Reconstruction score as a function of number of components for both datasets. The number of clusters selected is shown.*

The optimal number of clusters chosen by Isomap was visually determined by location of greatest curvature. For these two datasets, this is located at 6 and 8 components. Again, Isomap reduces the dataset into a low number of features, as with our prediction. However, its quite impressive to note the ability of Isomap to achieve a low reconstruction loss score with just 8 clusters, as opposed to 29, 61, and 25 with the previous DRA’s. This is likely due to the discretized nature of the digit dataset needing nonlinear boundaries to properly parse the manifold space, due to its ability to capture far more data complexity, especially when divisions between target spaces are fine.

1. **Re-clustering**

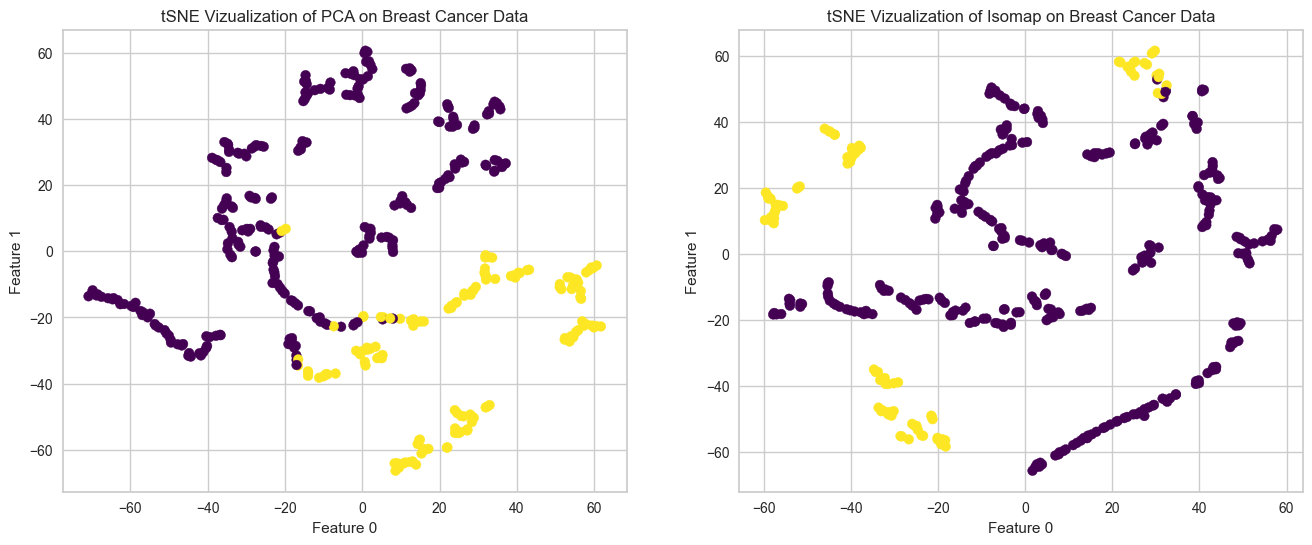
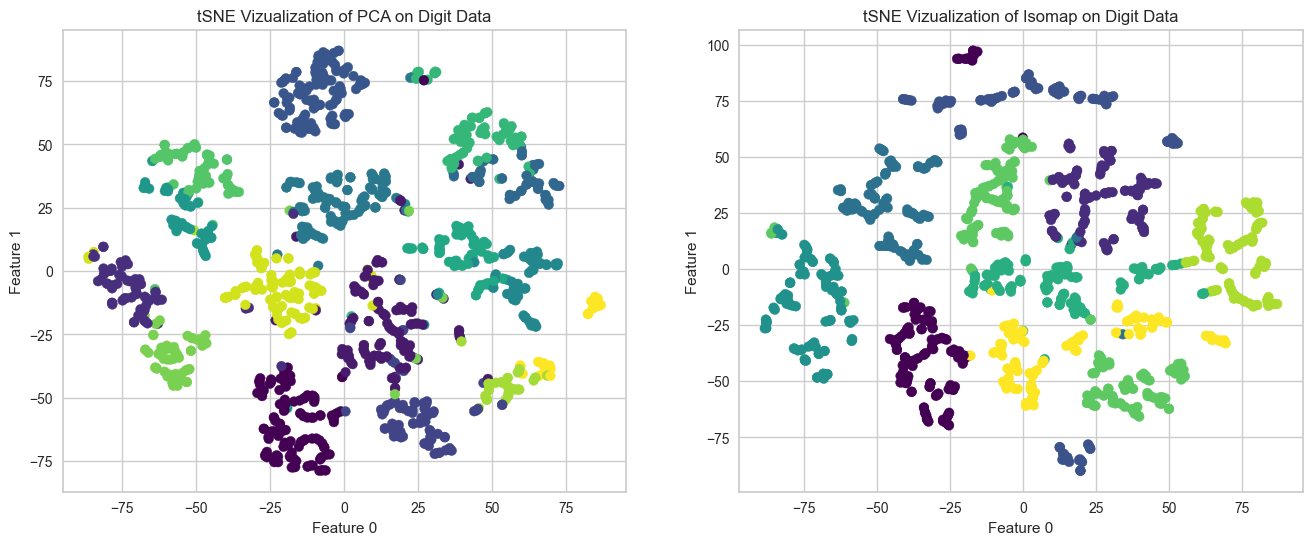
Once eight datasets have been obtained by running each of the two original datasets through each of the four DRA’s, we can re-cluster the data using each of the two clustering algorithms. The clustering resulted in the following numbers of clusters. This section was done in an unsupervised fashion hence ground truth labels are not considered in dimension selection.

*Figure 9. Clusters resulting from clustering on reduced datasets.*

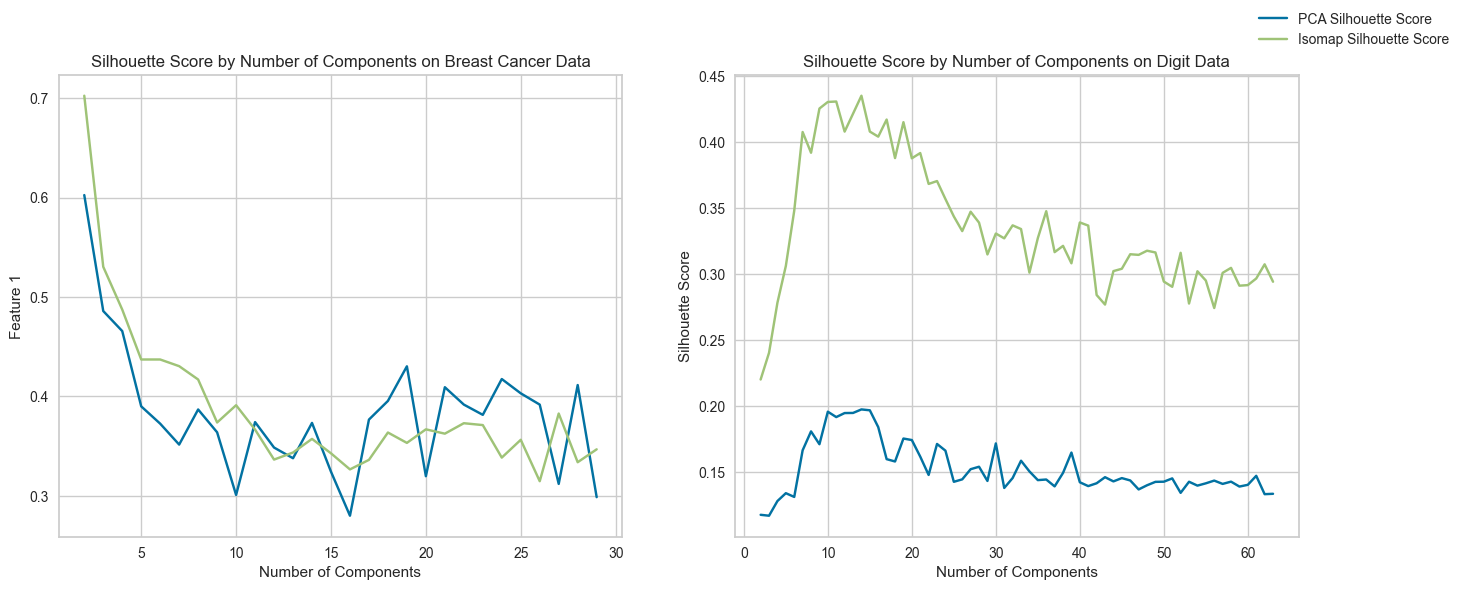
For the breast cancer dataset, EM finds 2 clusters for almost all datasets, as it does for the original dataset. K-means almost does the same, finding 7 clusters in all but one dataset, namely, the ICA dataset, likely due to slight inaccuracies in finding the exact elbow value. On the other hand, for the digit data, k-means finds a wildly low number of clusters, only 4, while the next lowest is only 11, found by k-means on the Isomap data. The ICA-reduced digit data seems be quite good with cluster finding; this is likely because of the ability to separate out distinct sources of data to eliminate crosstalk. PCA, on the other hand, seems to find more clusters than the original data does in both clusterings of the digit data. PCA seems to do poorly with the discreteness of the second dataset. Of course, because of the unsupervised nature of this analysis, the number of clusters is not directly a measure of how well the dimensionality reduction algorithms or clustering algorithms perform.

To better understand the results of these clustering algorithms on the reduced data, we will closely examine the behavior of EM on the PCA- and Isomap-reduced versions of our data. Recall that PCA and Isomap reduced the breast cancer dataset to 2 and 6 components for the breast cancer data, and 29 and 8 for the digit data. EM was selected for its ability to generate generally lower numbers of components. To visualize what these datasets look like, we can use a t-distributed Stochastic Neighbor Embedding (t-SNE) to reduce the Isomap data to 2 components. While it itself is an example of a manifold dimension reduction algorithm, it is only used here for visualization. Figure 11 shows the results of applying t-SNE to the reduced data, with colors representing targets that are predicted by EM. Though t-SNE adds bias, as we are looking at a reduction of our PCA- and Isomap- reduced data, we still see that in most cases, EM can clearly separate out clusters in the two datasets, and in many cases, fine boundaries between groups are established.

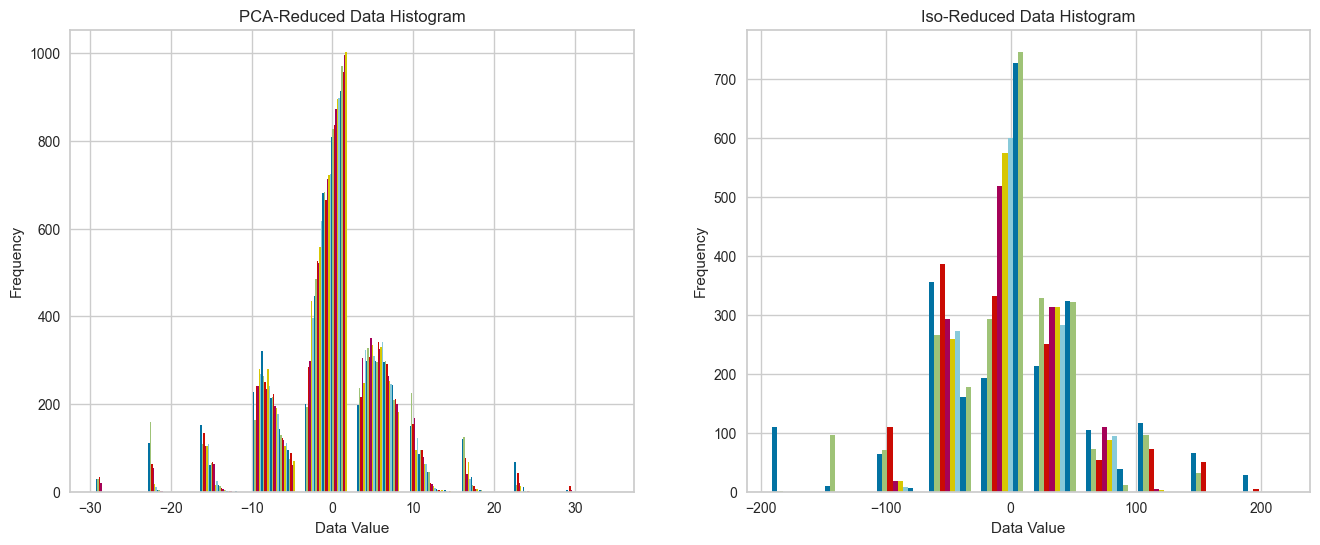
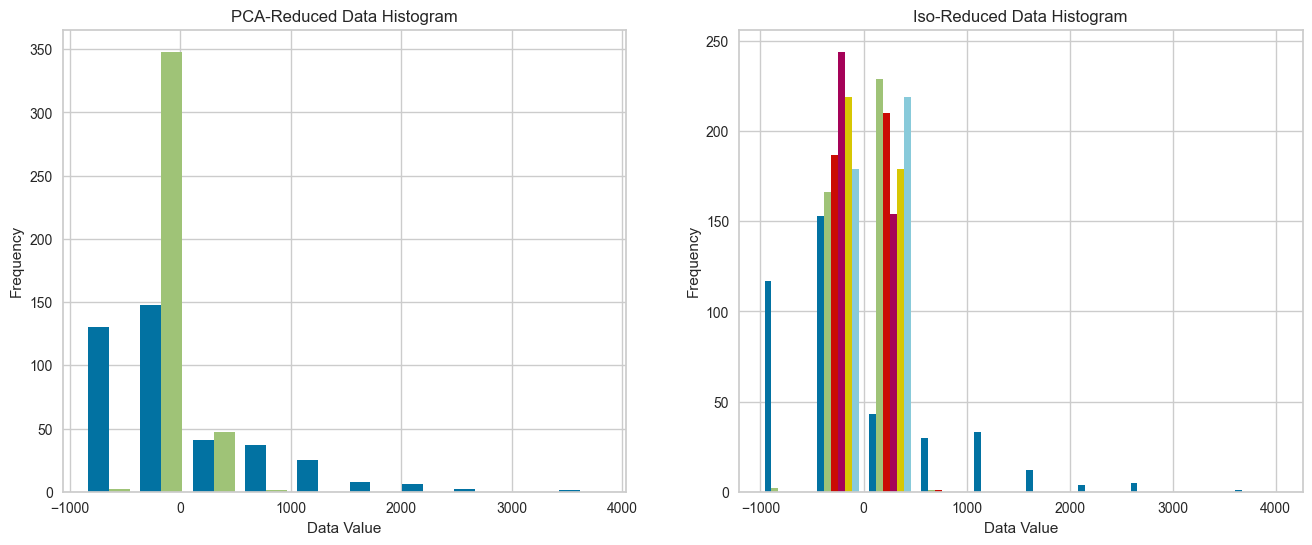
*Figure 10. t-SNE transformation of PCA- and Isomap-reduced data. t-SNE transformation of original data is shown for reference, with clustering.*

Because EM uses maximum silhouette score to find an appropriate number of clusters, it would be helpful to start exploring with the silhouette score calculated for a number of clusters. The following figure shows the silhouette score for these two datasets as a function of component number.



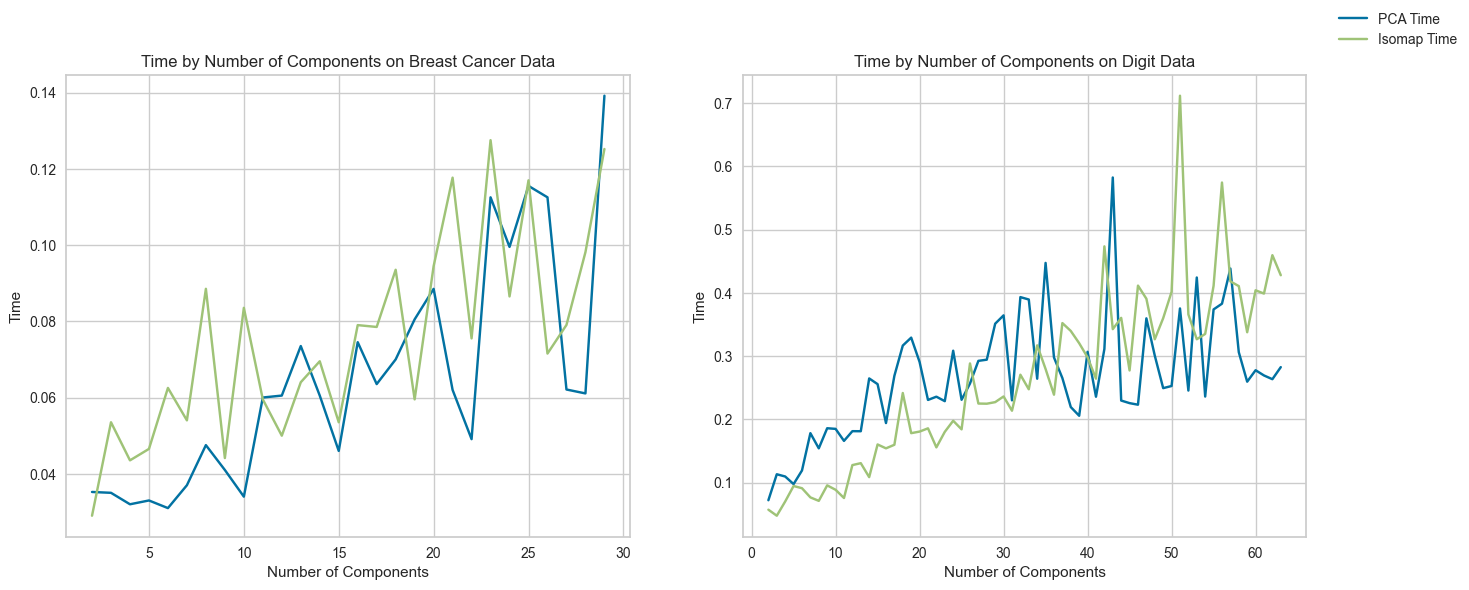
*Figure 11. Component selection for EM two datasets through silhouette score as a function of component number.*

We can see a steep decrease in silhouette scores for the breast cancer data, mainly sure to the large clumping interconnected nature of the data; with more than two clusters, more overlap between clusters arises, resulting into smaller values of silhouette score. We can also note the relatively similar scores achieved by the PCA and Isomap data, though Isomap outperforms slightly at lower component values, and vice versa at higher ones, crossing around a cluster number of about 13. During feature reduction, Isomap likely separates out individual clusters well enough, such that when fit onto EM, there is higher amounts of cluster overlap at high numbers of clusters. For the digit dataset, however, Isomap seems to outperform PCA in every number of components, by a considerable and consistent margin, despite reaching maximum silhouette scores at approximately the same number of components. Because Isomap reduced the digit data to less than a third of the features that PCA does, it is likely that far less overlap of clusters exists; again, this is because of the non-linearity of Isomap being able to separate data values more sparsely. We can visualize this by looking at the distribution of values that the data in PCA and Isomap take on, shown in figure 12 (Right). The wider distribution of the Isomap-reduced digit data is immediately evident from this graph. In fact, the standard deviation of the PCA digit data is only 6.278290, compared to the Isomap digit data, which was calculated to be 51.948746. Contrasted with the similarity between breast cancer data deviation, which exhibits standard deviations of 502.906520 and 335.954768, respectively. This shows the robustness of Isomap on heavily overlapped and discrete valued datasets.



*Figure 12. Distribution of data values in PCA- and Isomap-reduced data. (Left) Breast cancer data. (Right) Digit data.*

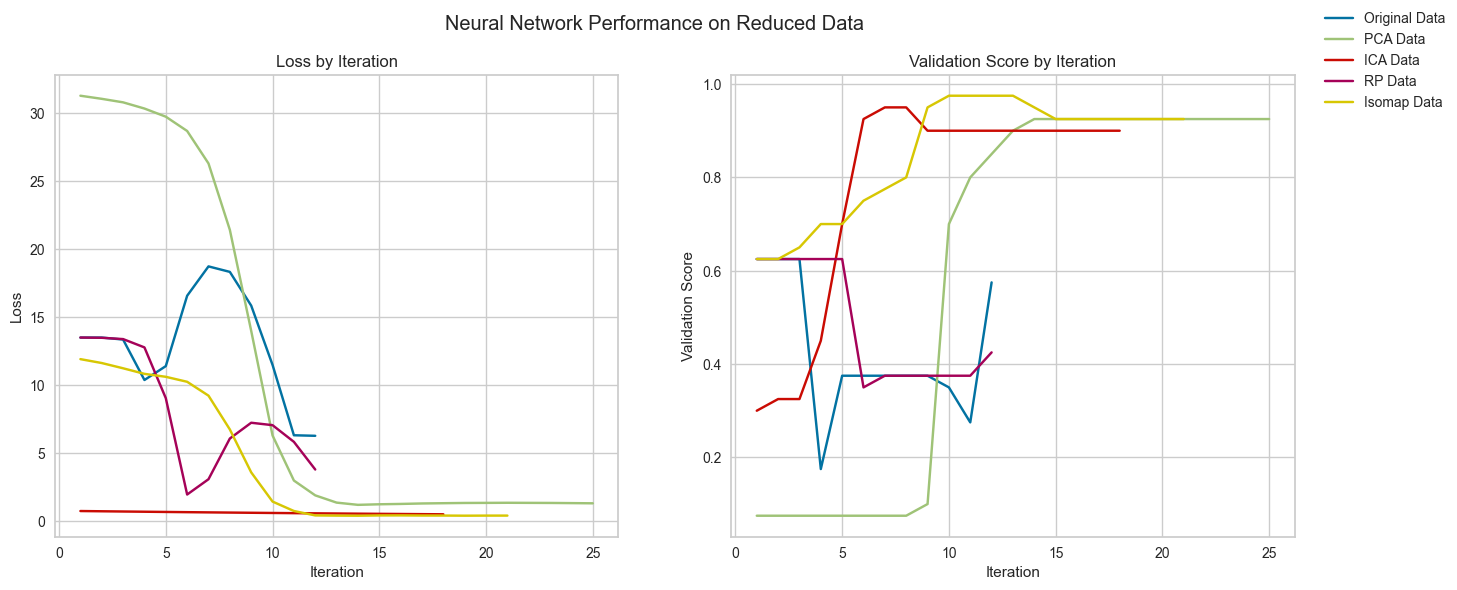
The fitting times to determine the ideal number of clusters can also be found plotted. There is not significant difference in the performance of EM over the PCA-reduced and Isomap-reduced data on either dataset, as both increase somewhat linearly with the number of components (with considerable random noise). Averaging over many random iterations could remove much of the random noise.



*Figure 13. EM fitting time as a function of component number for both datasets with both DRA’s*

1. **Neural Network Fitting**

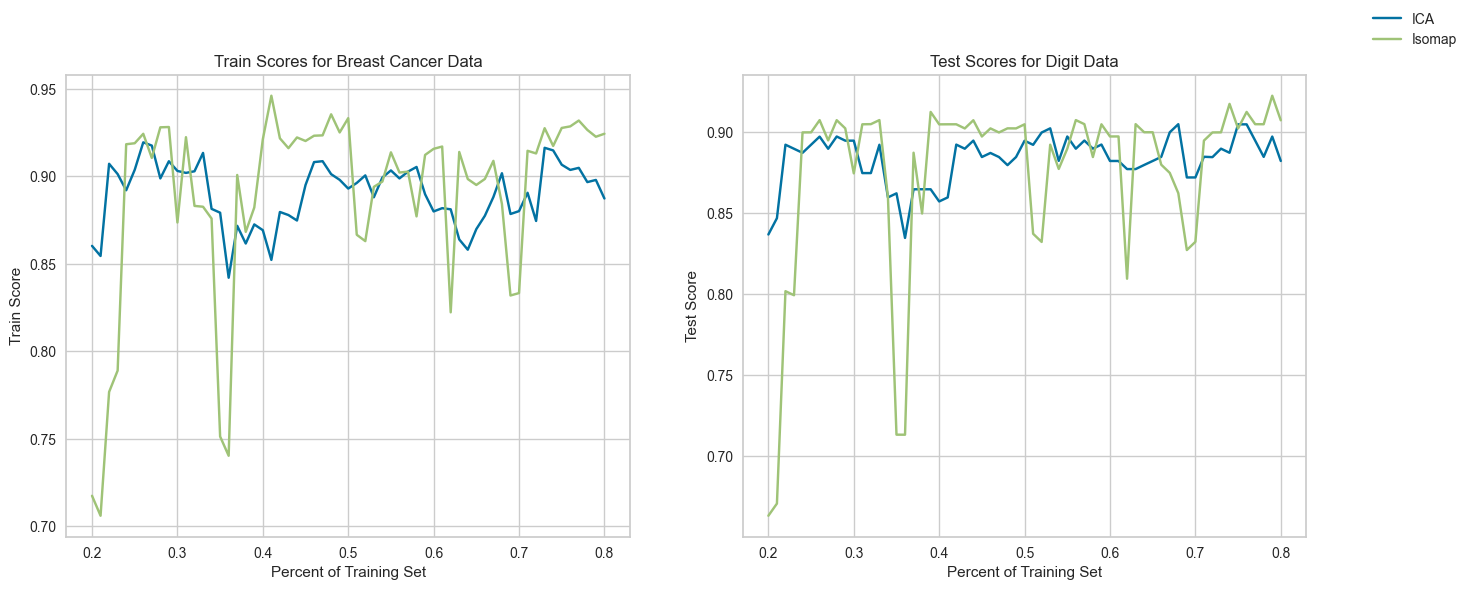
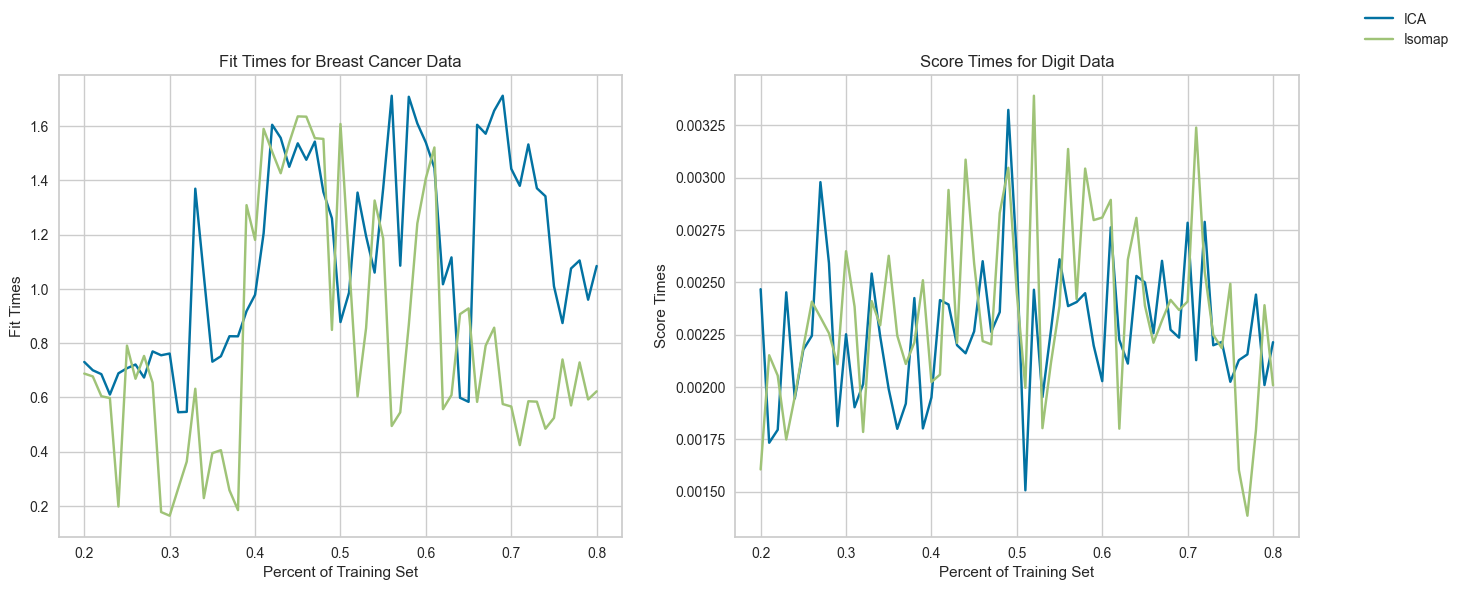
In this section, we will analyze results when using training a multi-layer perceptron model with the data we have reduced, specifically, the reduced datasets from the four DRA’s on our breast cancer data. The breast cancer dataset was selected, as it is known to have a high degree of reducibility. This section was done in a supervised fashion, meaning a testing set was transformed on each DRA as well, after being fit with the respective model’s training set. Figure 14 shows the loss curve and validation scores at each iteration for the neural net with each dataset as an input, including the original data.



*Figure 14. Loss curve and validation score by iteration over each dataset.*

By looking at the validation curve for the datasets, PCA, ICA, and Isomap seem to perform extremely well, each boding about 90% accuracy at termination. We can note that ICA is the first to get near its max, but exceeds 90% and falls down to it after a series of iterations; Isomap also does this, yet it approaches perfect validation score slightly closer, and takes a bit longer to settle. PCA, the last to get to about 90% accuracy, just slowly approaches, after a large jump. The drop in Isomap and ICA is due to an alteration in the network structure that improves the score of training set, but in fact harms the score of the testing set. This is a known problem for neural networks, and is due to a mismatch between training and testing set. One way to reconcile this error is to split multiple sets an perform multiple trainings and validations. The original data and the RP, on the other hand, fail to reach as low loss scores as the other three algorithms, and instead, have a number of iterations where loss increases drastically. This is because an early stop condition was reached for both of mentioned datasets during training, where a validation set score did not increase for a number of iterations, even though training set loss decreased. The Johnson-Linderstrauss lemma tells us that RP would have performed better with the reduced-digit dataset.

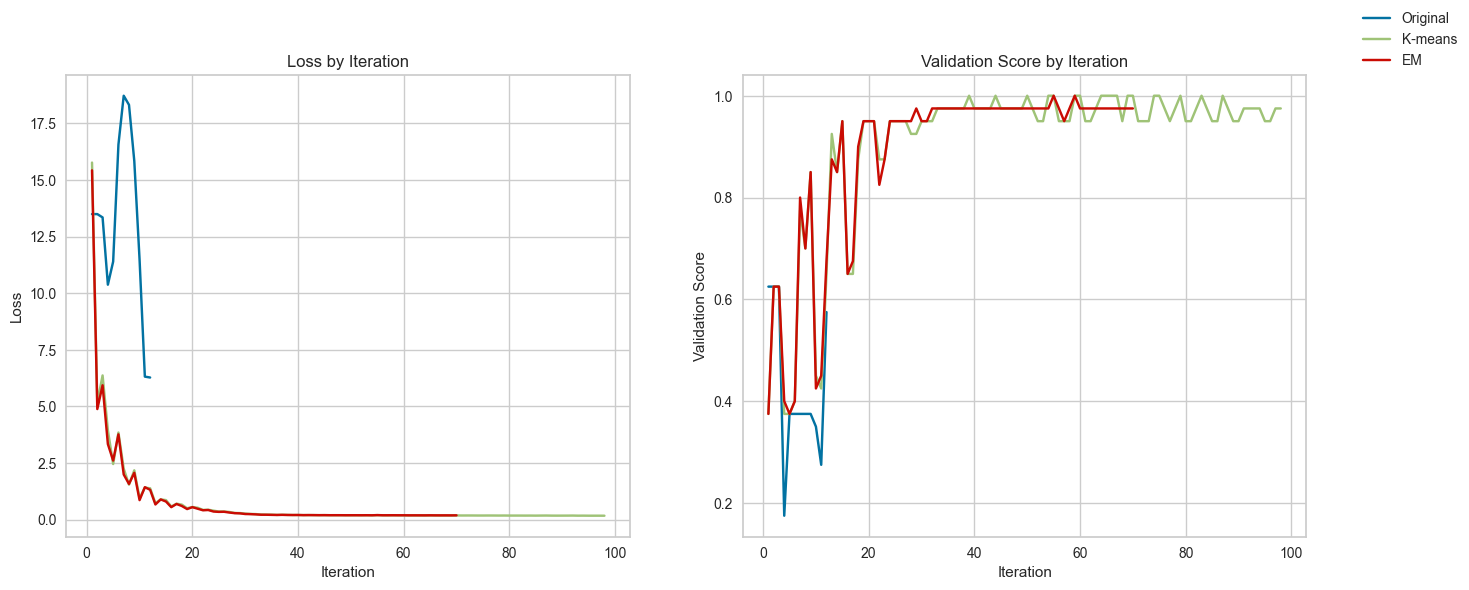
Looking at the learning curves of ICA and Isomap can show more detail about the behavior of the two reduced datasets. Figure 15 shows the training scores and test scores of the ICA- and Isomap-reduced datasets. The generation of these learning curves employs 5-cross validation, to slightly mitigate effects of random selection. At first glance, the scores of both reduced datasets seem similar, both hovering around 90 %. However, the variation of Isomap seems to be greater, with large drops in accuracy at multiple points, first around 0.35, then again at 0.51, 0.62, and 0.69. Because of the large variation of the Isomap dataset, this variation in scores could be due to separation of the five datasets. Luckily, Isomap seems to return to the incumbent score shortly after its dips. Isomap also seems to struggle with small training sets, which again could be due to the large variation in the set.

 **

*Figure 15. (Left) Learning curves of the two reduced datasets. (Right) Fit times and score times for the two datasets.*

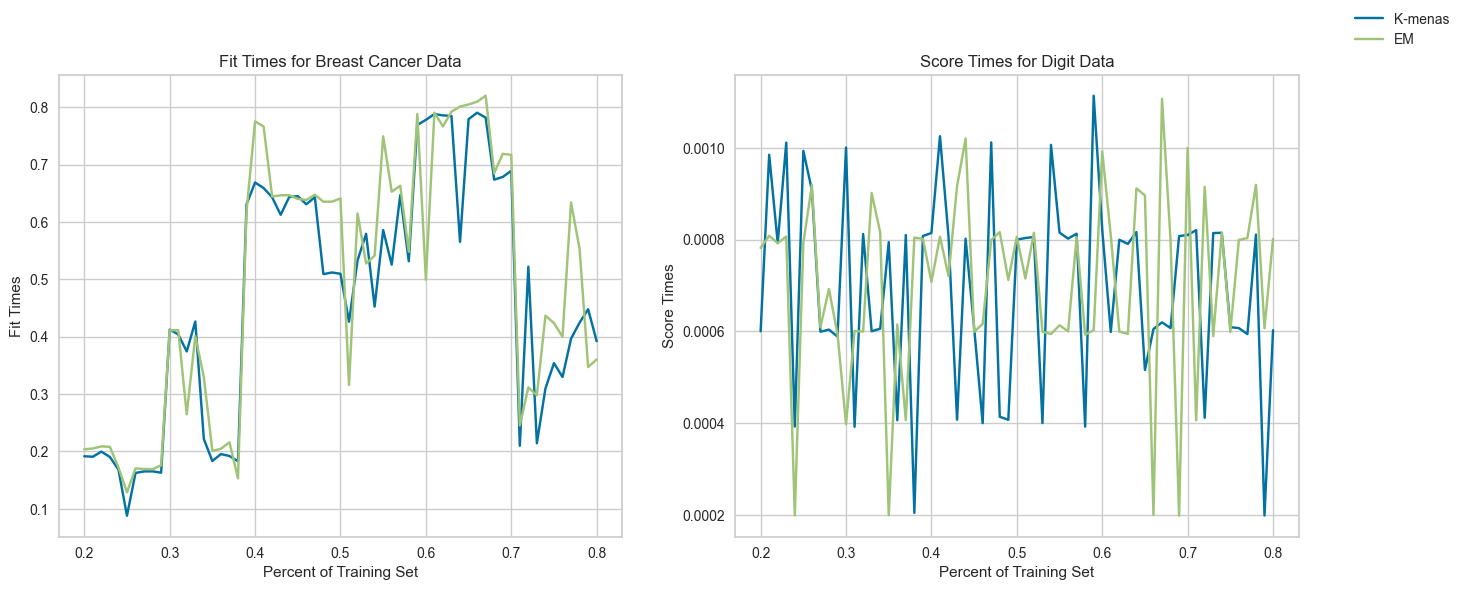
Figure 15 shows the fit and score times of the neural network on the ICA- and Isomap-reduced datasets. The score times of the two datasets is quite similar, with average times of 2.270 milliseconds for ICA and 2.356 milliseconds for Isomap. The fit times, however, differ by more, with average times of 1.119 seconds for ICA and 0.811 seconds for Isomap. The faster fit time may be because of better isolation of features due to the nonlinearity of Isomap.

We will now analyze the performance of the neural network with the cluster results inputted as features. The dataset will be the original breast cancer dataset, and will be supplemented with both EM and k-means original fittings. Recall that EM and k-means fitted the data with 2 and 7 clusters. Figure 16 shows the loss curve and validation score by iteration over both datasets.



*Figure 16. Loss curve and validation score by iteration over both datasets.*

The original data loss curve and validation score are shown for reference. It is evident that the addition of clustering results an addition feature drastically increases the performance of the neural network classifier. The loss for both datasets quickly approach zero, and validation score quickly approaches a perfect validation score, both settling at around 96 %. The validation score of k-means seems to fluctuate quite a bit, before settling. K-means also seems to take about 15 iterations more than EM does, and is likely due to the overestimate of clusters that k-means makes in the original fitting.

*Figure 17. (Left) Learning curves of the two boosted datasets. (Right) Fit times and score times for the two datasets.*

By inspection, there seems to be very little difference between the performances of k-means and EM. Because large amounts of noise are present in all plots, due to random set selection variation, the general shape will be analyzed. Both training and test scores seem to increase somewhat linearly by increasing the percent of data in the training set. More interesting than the scores, though, the fit times for both the k-means and EM data seem to match to a spectacular level, with synchronized increases and decreases in fit time at seemingly arbitrary times, since it would be expected that fit times of k-means would be marginally higher because of more states that the k-means feature takes on. Not much meaning can be extracted from the scoring times of digit data, aside from their general similarity, since the amount of data being scored is identical.

1. **Conclusion**

In this assignment, we analyzed the how clustering and dimensionality reduction algorithms perform on different datasets. The two datasets selected for this assignment were deliberate, demonstrating different degrees of correlation among the features. In initial clustering, EM seems to find a lower number of clusters for both datasets compared to k-means. When run with DRA’s, the high correlation within features of the breast cancer dataset allows for a drastic reduction of features in every case, while the lower correlation of the digit data does not reduce quite as well. This led to a more consistent clustering of the reduced breast cancer data compared to the reduced digit data, whose number of clusters found varies greatly. When these reduced datasets are fit using a neural network, we see a drastic improvement in the validation scores achieved, indicating the effectiveness of the reduction techniques to remove non-important features. Finally, when the original clusters are combined with EM and k-means classifications, a drastic increase in performance is observed, indicating effectiveness of clustering techniques.