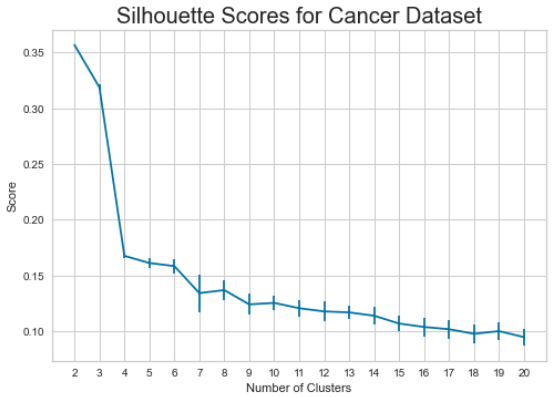
# CS 7641: Assignment 3 - Unsupervised Learning and Dimensionality Reduction

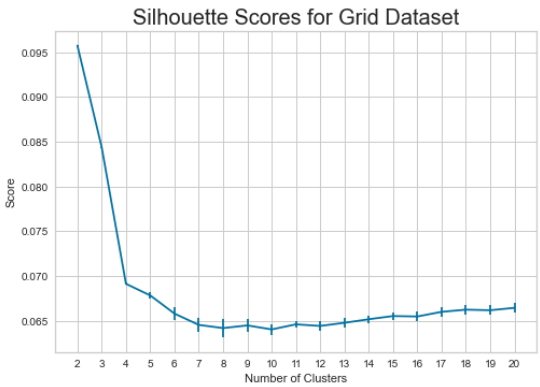
## Overview

This assignment consists of five parts: clustering algorithms, dimensionality reduction, dimensionality reduction and clustering, dimensionality reduction to refit the neural network, and clustering to refit the neural network. Parts 1-3 involved both the electric grid dataset and breast cancer dataset from the UC Irvine Machine Learning Repository. The target variable indicated whether the electric grid network is considered to be stable or not. The predictor variables consisted of the reaction time of the participant (tau), the power consumed (p), price elasticity (g), and maximal root of the equation (stab). The second dataset is the Wisconsin breast cancer dataset with 30 numeric predictor variables and 1 target variable. The target variable indicated whether the breast mass is benign or malignant. The dataset came with the target variable encoded so I didn't have to encode it myself. I processed, cleaned up, and standardized both datasets. Prior to running unsupervised learning methods and dimensionality reduction, I split each dataset into the training set (80%) and the testing set (20%). Parts 4-5 focused on just the breast cancer dataset. This entire assignment required the use of Python and its libraries such as sklearn, matplotlib, and yellowbrick.

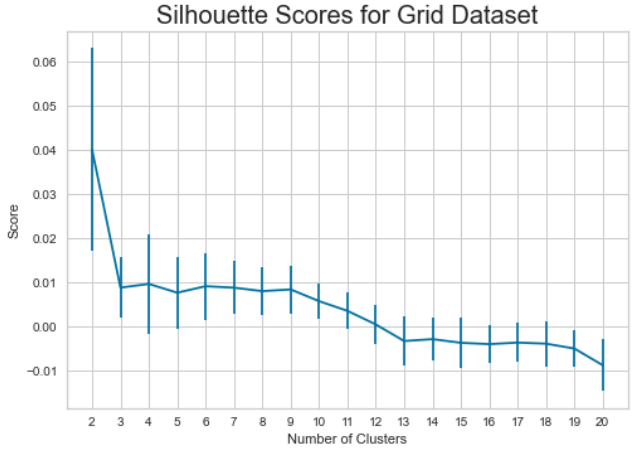
## PART ONE: Clustering Algorithms

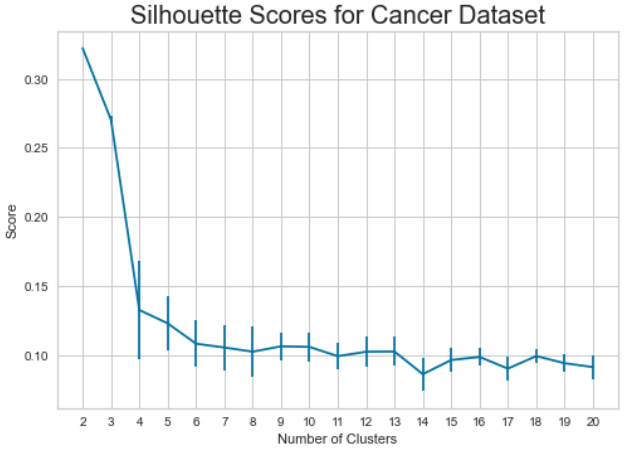
For this section, I examined both KMeans and expectation maximization (EM). I applied the elbow method to the silhouette score plots below to discern the ideal number of clusters with kmeans clustering, which turned out to be 4 clusters for both datasets.




For EM, it is apparent with the Elbow method that the ideal number of clusters for the grid dataset is 3, while 4 clusters are ideal for the cancer dataset.



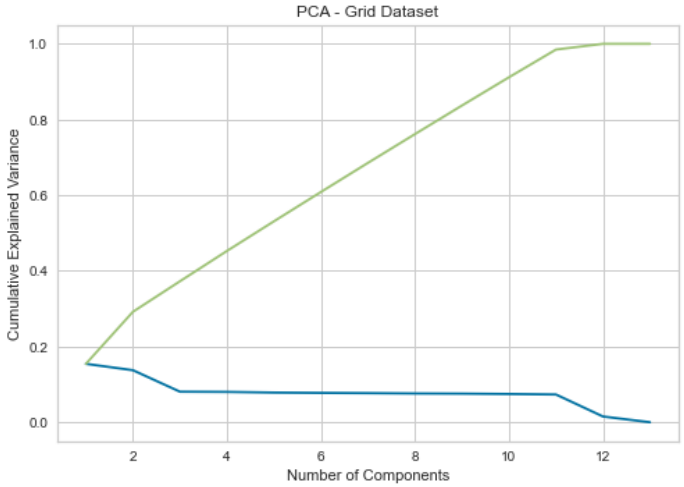



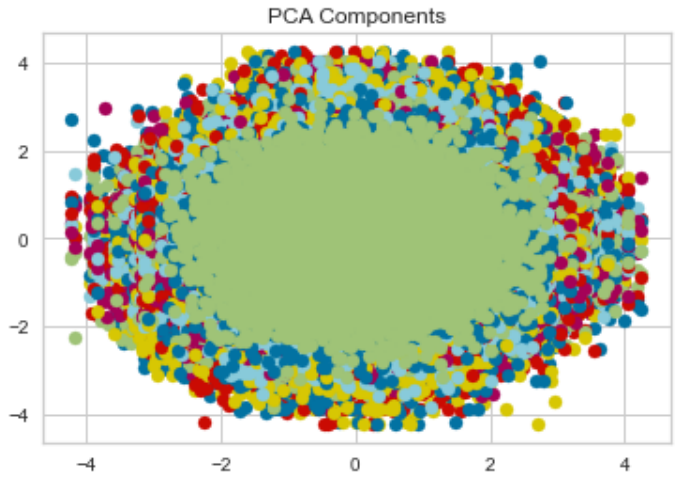

| | Part 1: Clustering Algorithms | | | |
|---|---|---|---|---|
| | KMeans Clustering | | Expected Maximization | |
| | Grid | Cancer | Grid | Cancer |
| Train Time (s) | 1.75 | 1.62 | 4.39 | 1.7 |
| Query Time (s) | 0.16 | 0.18 | 0 | 0 |

The table above reflects the training and query times for each clustering algorithm and dataset. The query time was slower for kmeans than EM for both datasets. This is understandable as the kmeans algorithm iteratively refines the centroids, while the EM algorithm is more efficient by utilizing probabilistic modeling to assign data points to clusters. The train time was similar for all datasets and algorithms except the grid dataset with EM that was much slower than the others. The similarity in training times is to be expected since both kmeans and EM clustering algorithms are rather comparable in efficiency and convergence speeds. However, the grid dataset being much slower in training for EM could be attributed to the distribution of the dataset.

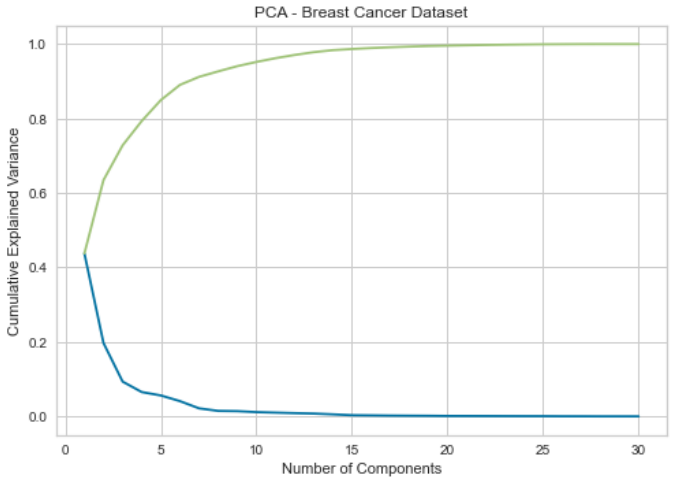
**PART TWO: Dimensionality Reduction**

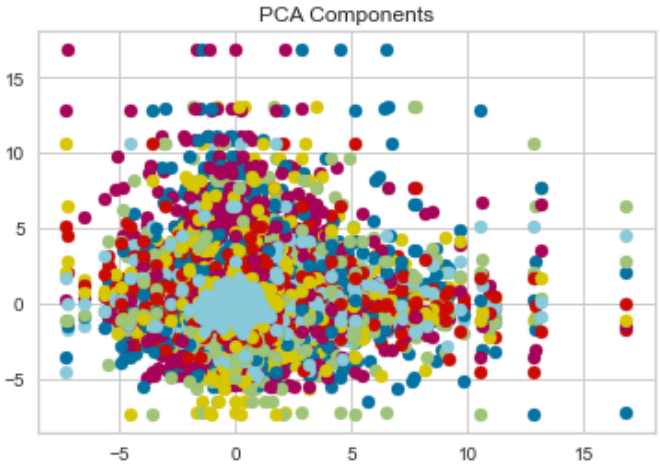
For this section, I applied the following dimensionality reduction techniques to both datasets: Principal Component Analysis (PCA), Independent Component Analysis (ICA), Randomized Projection (RP), and Singular Value Decomposition (SVD). For PCA, I outputted the optimal number of components along with the cumulative variance plot and decomposed components scatterplot.



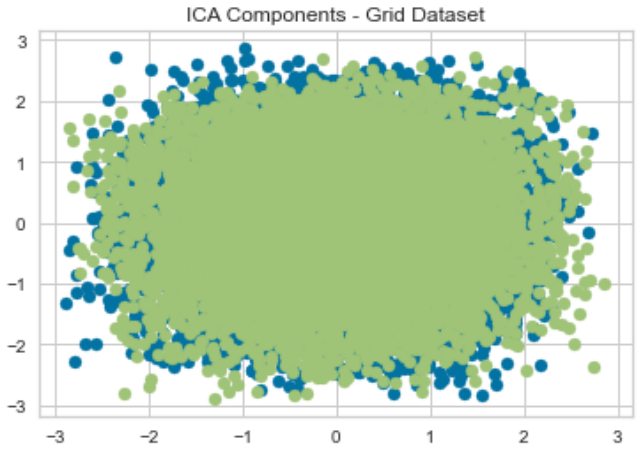

The grid dataset originally had 13 components. As you can see above, the cumulative explained variance plot (the green line) indicates that nearly 100% of the variance can be explained by 11 components for the grid dataset. The scatterplot on the right does not show the decomposed components very neatly, as it is clear that there is substantial correlation between the components.

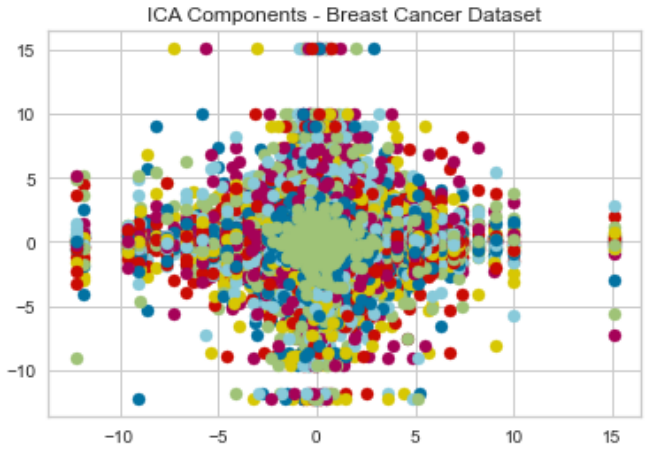


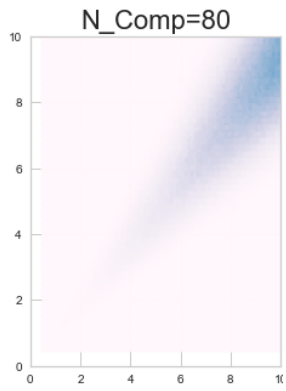

The breast cancer dataset originally had 30 components, but the ideal number of components was determined to be 10. The variance plot indicates that around 95% of the variance can be explained by 10 components. The scatterplot of the decomposed components on the right indicates weak separation of the different components. This may indicate that PCA may not be the ideal dimensionality reduction method for the cancer dataset. This can also occur from nonlinear relationships in the dataset.

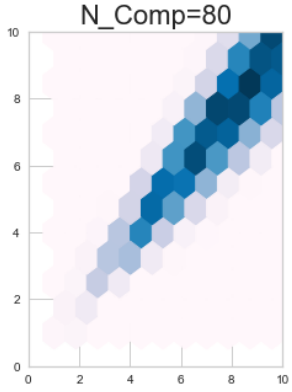
Next, I found the ideal number of components for ICA to be 2 for the grid dataset and 10 for the cancer dataset. Then, I produced scatterplots of the decomposed components for each dataset as shown below.



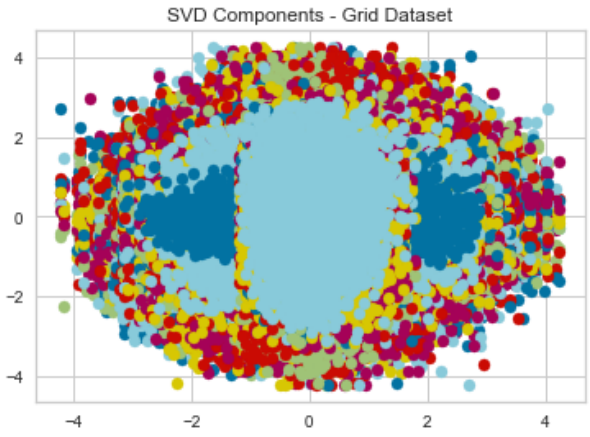

The scatterplot on the left indicates that two components are nicely separated and defined for the grid dataset. This shows much better separation than the decomposed components scatterplot for the PCA method. The improvement in separation in the ICA plot might be attributed to how ICA is better equipped to deal with non-normal distributions especially since PCA assumes Gaussianity and isn't as effective in separating components. The scatterplot on the right indicates poor separation of the components for the cancer dataset, similar to the PCA method. The cancer dataset experienced poor component separation for both PCA and ICA, which may indicate that the dataset is lacking in dominant factors or linear separability.

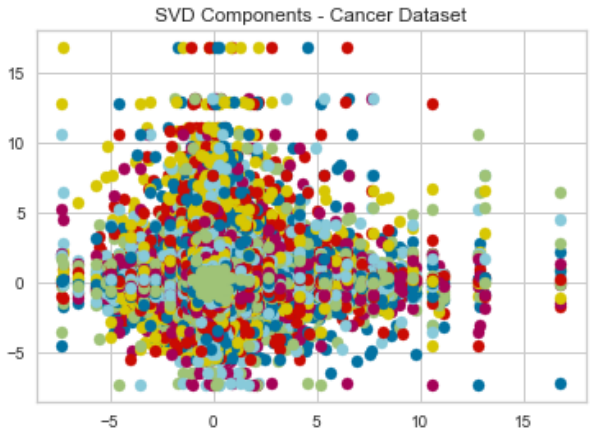



For RP, the MSE was calculated along with hexbin plots to compare pairwise Euclidean distances between data points in the original and projected spaces with varying numbers of components. The ideal number of components according to the lowest MSEs for both datasets was found to be 80. The hexbin plot on the left reflects the grid dataset, while the plot on the right reflects the cancer dataset. It is apparent that at 80 components for the RP method, the grid dataset exhibits lower density in the hexbin plot compared to the cancer dataset. Low density plots imply that the data points in the original and projected space do not cluster closely together. RP for the grid dataset has a MSE value of 18.67, while the RP for the cancer dataset has a MSE value of 118.79. The grid dataset having a lower MSE affirms that the RP technique preserved the relationships between the data points well.

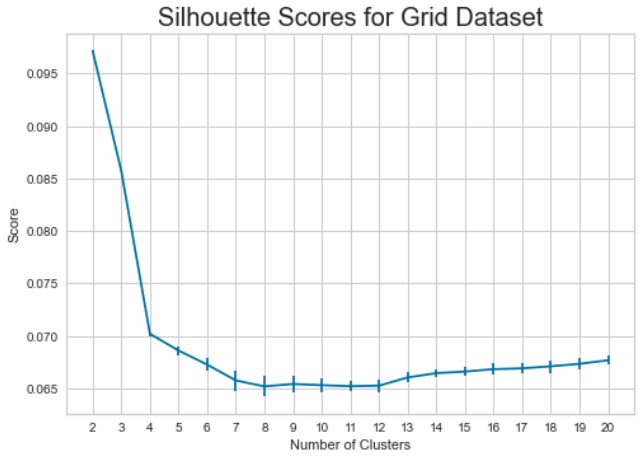
For SVD, 12 components is ideal for the grid dataset and 17 components for the cancer dataset as it can explain nearly 100% of the variance. The scatterplots of the decomposed components are displayed before. With the scatterplot for the grid dataset, it is clear that the components are mostly separated. However, the scatterplot for the cancer dataset depicts components that are not cleanly separated.

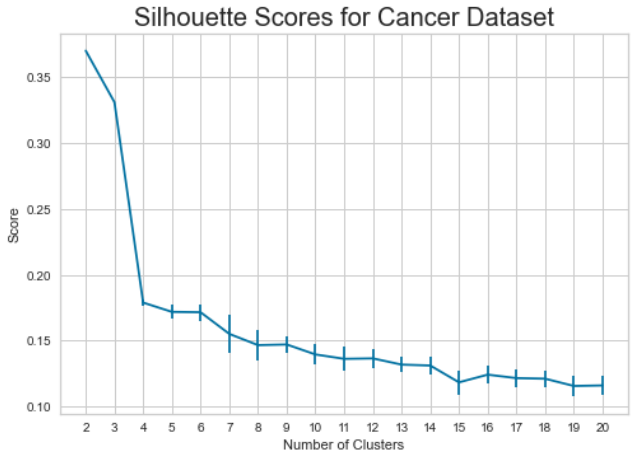


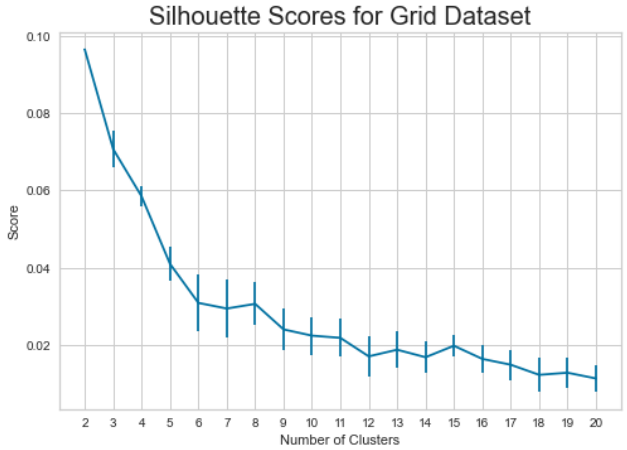
**PART THREE: Applying Dimensionality Reduction to Clustering Algorithms**

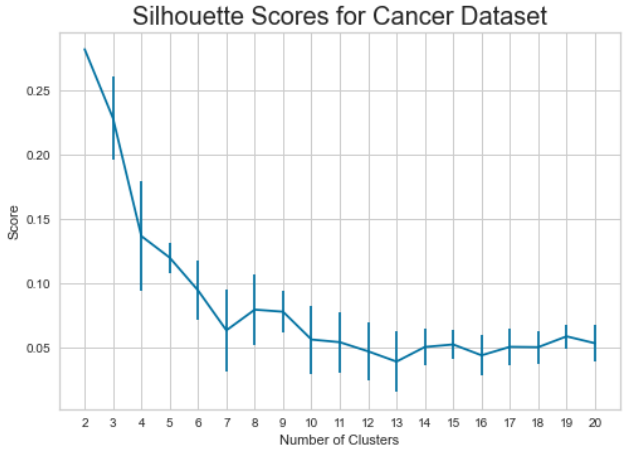
The dimensionality reduction techniques utilized in part two determined the optimal number of components necessary for PCA, ICA, RP, and SVD. I utilized these optimal values for the kmeans and EM clustering algorithms. Then, I produced silhouette score plots.



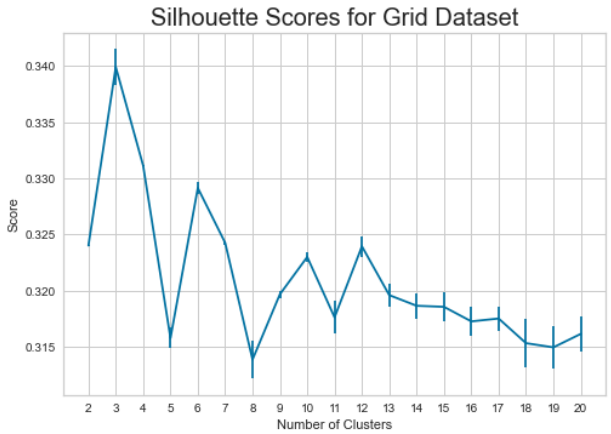



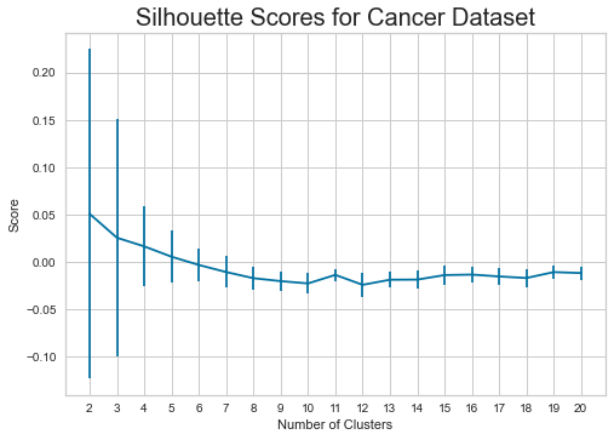

The silhouette score plots above are for KMeans clustering after using PCA to transform the datasets. As per the previous section, I used 11 components for the grid dataset and 10 for the cancer dataset.



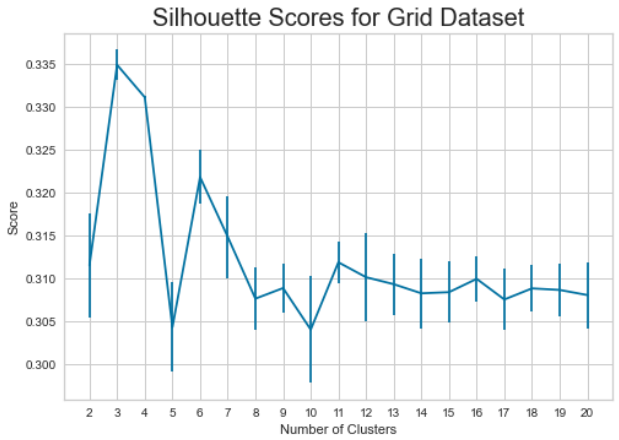



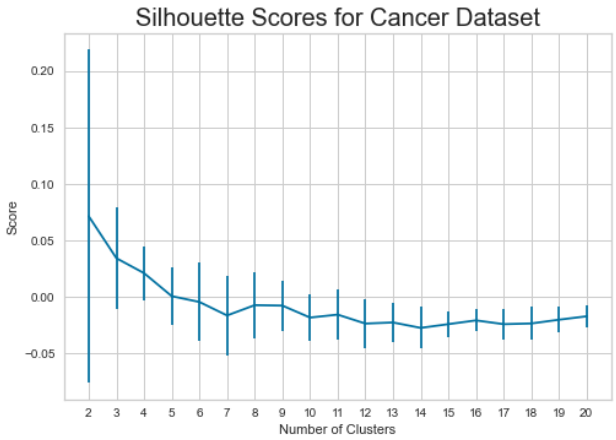

The silhouette score plots above are for EM clustering and PCA.



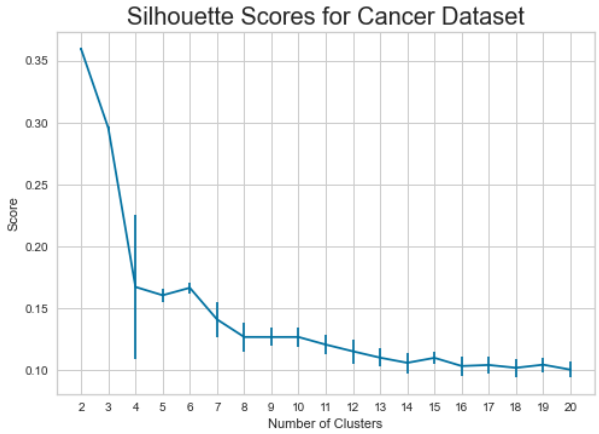



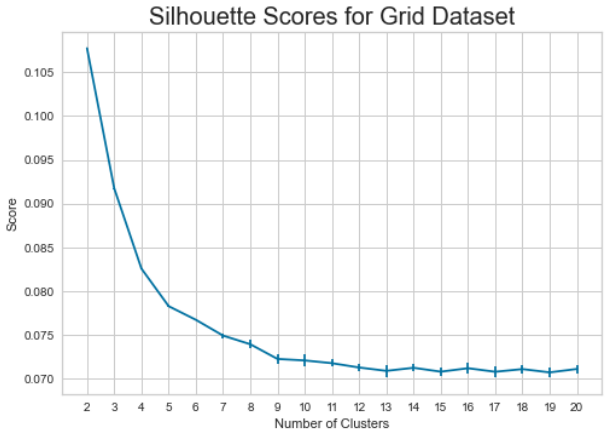

The silhouette score plots above are for KMeans and ICA. I used 2 components for the grid dataset and 29 components for the cancer dataset.



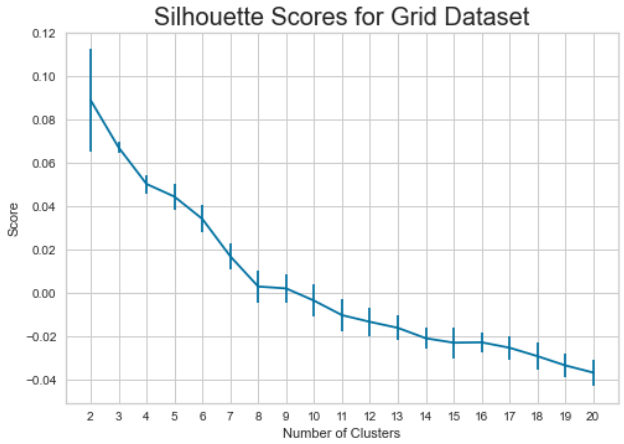



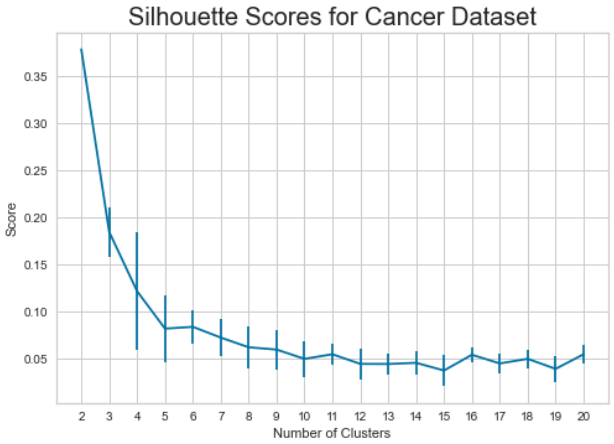

The silhouette score plots above are for EM clustering and ICA.



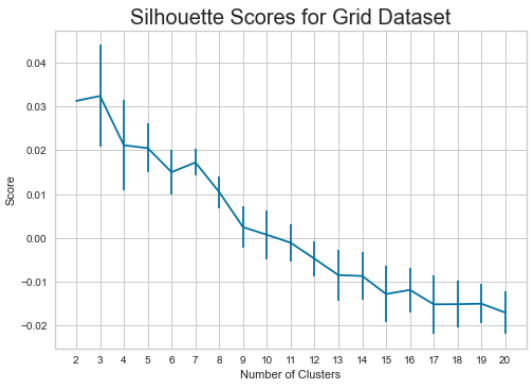



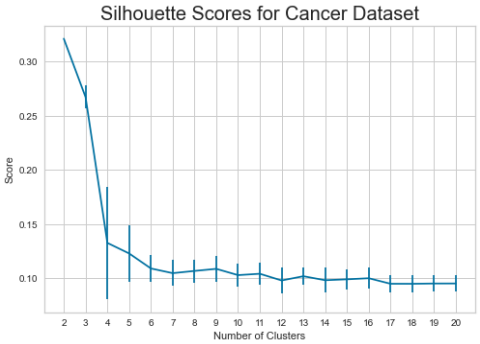

The silhouette score plots above are for KMeans and RP. I used 80 components for both datasets.



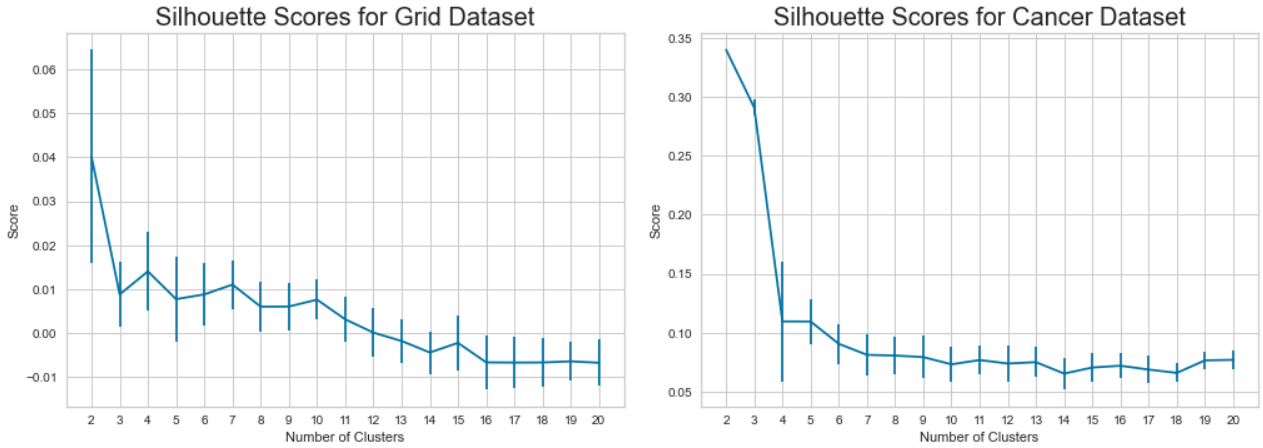




The silhouette score plots above are for EM clustering and RP.



The silhouette score plots above are for KMeans and SVD. I used 12 components for the grid dataset and 17 components for the cancer dataset. The silhouette score plots below are for EM clustering and SVD.




Just to recap part one of this assignment, 4 clusters were found to be ideal for both datasets for the KMeans clustering algorithm (prior to dimensionality reduction). I again used the Elbow method to discern the ideal number of clusters from each silhouette score plot. For the grid dataset with KMeans clustering, the ideal number of clusters are as follows: 4 for PCA, 13 for ICA, 8 for RP, and 8 for SVD. For the cancer dataset with KMeans clustering, the ideal number of clusters are as follows: 7 for PCA, 6 for ICA, 5 for RP, and 7 for SVD. From part one, 3 clusters were found to be ideal for the grid dataset and 4 clusters for the cancer dataset for the EM clustering algorithm. Again, this is prior to any dimensionality reduction. For the grid dataset with EM clustering, the ideal number of clusters are as follows: 6 for PCA, 11 for ICA, 13 for RP, and 12 for SVD. For the cancer dataset with EM clustering, the ideal number of clusters are as follows: 10 for PCA, 8 for ICA, 5 for RP, and 6 for SVD.

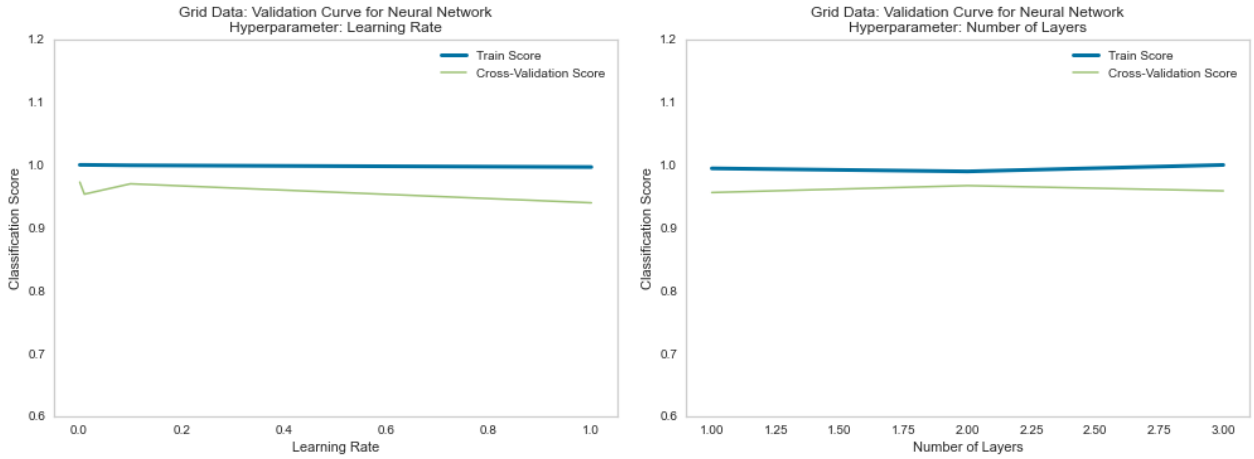
| **Part 3: Expectation-Maximization with Dim Red** | | | | | | |
|---|---|---|---|---|---|---|
| **Dataset Used:** | **Time (s)** | **PCA** | **ICA** | **Random Projection** | **SVD** | **Average** |
| Grid | Train | 1.45 | 0.16 | 0.74 | 1.23 | 0.895 |
| Grid | Query | 0.009 | 0.002 | 0.007 | 0.01 | 0.007 |
| Cancer | Train | 0.35 | 0.39 | 0.23 | 0.38 | 0.3375 |
| Cancer | Query | 0.002 | 0.003 | 0.004 | 0.002 | 0.00275 |
| | **Average** | 0.45275 | 0.13875 | 0.24525 | 0.4055 | |
| | | | | | | |
| **Part 3: KMeans Clustering with Dim Red** | | | | | | |
| **Dataset Used:** | **Time (s)** | **PCA** | **ICA** | **Random Projection** | **SVD** | **average** |
| Grid | Train | 1.94 | 1.26 | 1.59 | 1.8 | 1.6475 |
| Grid | Query | 0.11 | 0.11 | 0.09 | 0.11 | 0.105 |
| Cancer | Train | 1.11 | 1.4 | 1.06 | 1.03 | 1.15 |
| Cancer | Query | 0.1 | 0.11 | 0.09 | 0.11 | 0.1025 |
| | **Average** | 0.815 | 0.72 | 0.7075 | 0.7625 | |

The two tables above display the time it takes to train and query the dataset for each dimensionality reduction technique applied to either KMeans clustering or EM clustering. Here, it is clear that KMeans clustering is

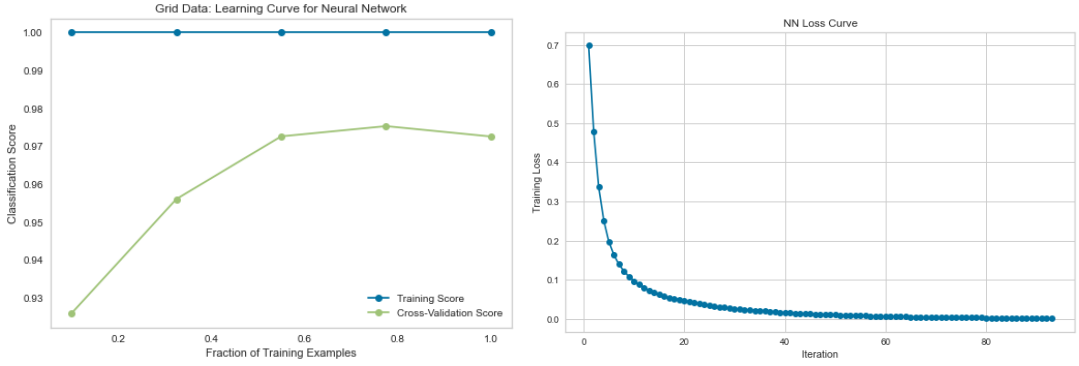
generally slower in training and querying the datasets compared to EM clustering. Dimensionality reduction techniques can affect the separability of data points, which can lead to an unsuitable clustering structure for KMeans. This may be why training/querying are slower for KMeans than EM. Training and querying times are similar across all dimensionality reduction techniques for both clustering models, which underscores the similarity in computational efficiency across all the techniques.

**PART FOUR: Refit Neural Network with Dimensionality Reduction**

For part four and five, I examined the cancer dataset. I compared the results of this assignment to the results of the Neural Network model in assignment 1. In assignment one, the best hyperparameters were found to be 1 for the hidden layer size and 0.01 for the learning rate. For the cancer dataset with dimensionality reduction, the best hyperparameters are 3 hidden layer sizes and 0.001 learning rate. With dimensionality reduction, the number of features decreased for all the techniques except for Random Projection. Instead, the cancer dataset increased from 30 features to 80 components with the RP technique (mentioned in part two). Eighty components is incredibly high, especially considering how small the dataset is. This might have thrown off the fit of the neural network, thus producing a slower learning rate to accommodate 80 components from RP. Moreover, it is apparent that the number of hidden layers increased from 1 in assignment one to 3 in this assignment. Dimensionality reduction can lead to a loss of information from reducing the number of components, so a more complex NN may be required to account for this. Below are the two validation curves for the learning rate (left plot) and number of layers (right plot).




Interestingly, the NN model in assignment one (without dim. red.) had a much faster prediction time at 0.00073 seconds while the NN model with dimensionality reduction took 0.00135 seconds. The NN model without dimensionality reduction had an accuracy of 96%, while the NN model with dim reduction has an accuracy of 42%. I was surprised to see a drastic drop in accuracy, especially given that dimensionality reduction helps to reduce the number of features. To me, this signifies that there must have been a profound loss of information which can affect the NN model fit and ability to capture relationships in the dataset. Below on the left is the learning curve plot, while the plot on the right is the loss curve for the NN model.

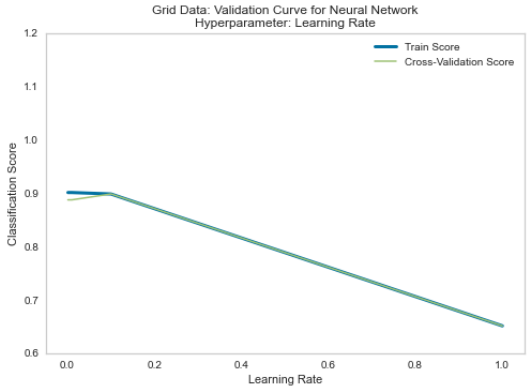


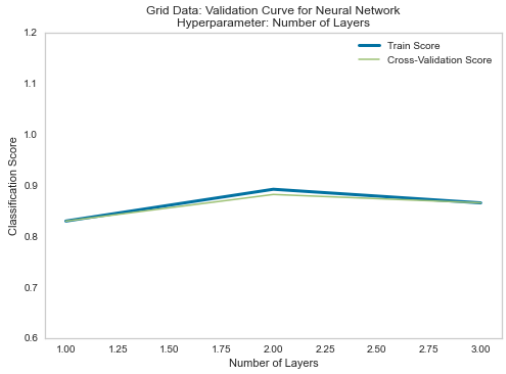

The learning curve for the training set is mostly flat, which indicates that the NN model failed to learn the training data. The loss curve plot shows a healthy learning rate for the model. However, it is apparent from

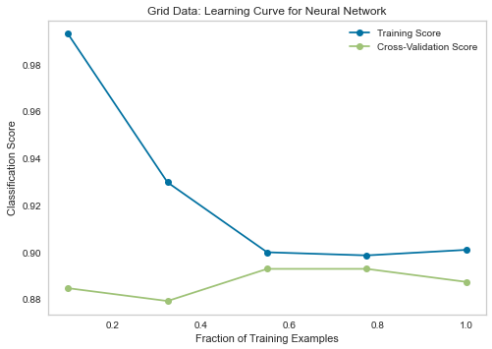
various metrics that dimensionality reduction does more harm than good for the breast cancer dataset, so it might be wise to avoid certain techniques (particularly the randomized projection one).

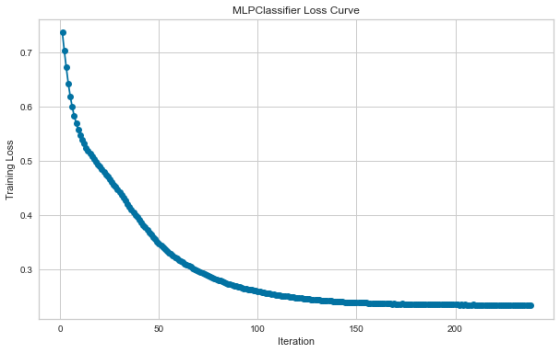
**PART FIVE: Refit Neural Network on Input Space from Part One**

I refitted the neural network based on the input space from conducting EM and KMeans clustering models.




The ideal hyperparameters for this NN model is 2 hidden layer sizes and 0.01 learning rate. The learning curve plot here points to possible overfitting, with the validation curve decreasing before increasing again. The learning curve for the training set decreases but increases slightly, again pointing to overfit. The structure of the loss curve is a downward curve, which indicates a good learning rate for the NN model. The accuracy for this model is 92%, which is slightly less than the accuracy for the NN model in assignment 1 but still fairly high. It is clear from part five that the NN model refitted based on the input space gleaned from conducting KMeans and EM clustering models performed much better than the NN model fitted on the data gleaned from dimensionality reduction techniques. This could be attributed to how unsupervised clustering models group data points based on similarity, thus capturing patterns in the data well. This would result in a well-structured input space for the neural network to learn from. Dimensionality reduction techniques run the risk of improperly preserving the structure and patterns in the data. There is also the risk of information loss, which can make it difficult for the NN model to accurately learn.

**Works Cited:**


https://cs231n.github.io/neural-networks-3/
https://www.analyticsvidhya.com/blog/2018/08/dimensionality-reduction-techniques-python/#h-3-9-independent-component-analysis
https://neptune.ai/blog/clustering-algorithms
https://towardsdatascience.com/gaussian-mixture-model-clusterization-how-to-select-the-number-of-components-clusters-553bef45f6e4
https://vitalflux.com/elbow-method-silhouette-score-which-better/