Assignment 3
Unsupervised Learning
Amisha Buch - abuch6

1. DATASETS

Diabetic Retinopathy (DR) Debrecen Data Set

Diabetic retinopathy is caused by damage to the blood vessels in the tissue at the back of the eye (retina). This data set contains features extracted from the Messidor image set to predict whether an image contains signs of diabetic retinopathy or not. All features represent either a detected lesion, a descriptive feature of an anatomical part or an image-level descriptor. The data is complete with 19 attributes, out of which 18 are features and one is the Result (0,1) variable which determines if the image contains sign of DR. There are 1151 samples in total. There are no missing values. The attributes in the dataset were normalised using the StandarScaler pre-processing library in python.

One of the main reason I feel this that accuracy of supervised learning algorithms for this dataset Is low is that the value of the various features are sparsely and uniformly spread, making it difficult to split into hyperplanes or get a perfect splitting criteria. It would be interesting to see how clustering can be performed on such data and if dimensionality reduction applied will give any different results.

Website Phishing Data Set

Phishing is the fraudulent attempt to obtain sensitive information or data, such as usernames, passwords and credit card details or other sensitive details, by impersonating oneself as a trustworthy entity in a digital communication. We have a dataset where different features related to legitimate and phishy websites have been identified. There are total 1353 samples in the data set out of which 702 are phishing URL's and 103 are suspicious URL's. The rest are legitimate ones. The phishing websites were collected from Phishtank (www.phishtank.com) data archive and the legitimate ones were collected from Yahoo. There are 10 attributes in the data set, 9 features and 1 Result set which classified the url's as Legitimate, Suspicious or Phishy. When a website is considered SUSPICIOUS it means it has some legit and phishy features both. The data is complete with no missing values. The attributes in the dataset were normalised using the StandarScaler pre-processing library in python.

Supervised learning techniques performed better than the diabetes dataset on the phishing dataset. One reason is I think that it was easier to find boundaries within data due to its non-fractional values. This could be an advantage for clustering.

Clustering

We standardise the data before feature selection and transformation. Also, we do not split the data into training and testing set for the clustering and dimensionality algorithms (except neural networks at the end), as we don't want to build a model, just test how accurately an algorithm performs on unlabelled data.

K-Means Clustering: K-means algorithm identifies *k* number of centroids, and then allocates every data point in the dataset to the nearest cluster, while keeping the centroids as small as possible.

The distance metric we have used here is the Euclidean distance. The cluster range i.e k ranges from 1 to 50, for each dataset to find the best value of k. There are various ways which you can determine the best value of k. One is the elbow method and one is the silhouette method. In the elbow method, we try to find a point from where the accuracy or score starts descending within a cluster. In the second method, silhouette value indicates how similar (or different) a data point is to its cluster compared to others. Higher the value, more is the accuracy of a cluster. I have use both methods and then taken the best of the two results to determine best value of k. Lets apply k-means to both datasets and see the results

Expectation Maximisation: The EM model works on probability. EM creates a probability density function, and each point can belong to many clusters with different probabilities. EM tries to find the hypothesis function which maximises this probability by computing the clustering expectation and then take the mean of the clusters. We look at log likelihood to determine how similar data points are clustered together. Number of clusters are chosen such that adding more cluster doesn't give improvement in performance.

Let us now apply these clustering mechanisms on the two data sets and see some results. First is the Diabetes dataset.

Figure 5 and 6 show the accuracy obtained from Diabetes dataset for both clustering algorithms with varying number of clusters. Although the dataset has 2 labels, accuracies are extremely low for both methods, showing that dataset cannot be straightaway segregated into clusters representing different classes in higher dimensions .

The Figure 2 and 3 above show that the silhouette score shows that the k value should be around 2 or 3 after which it starts decreasing, which validates our data as we have 2 labels. The elbow method shows that the SSE graph also shows 2 as the highest point. Elbow method is not very accurate in such datasets, as the continuous decreasing graph does not help much to determine the k value. Therefore, a k-value of 2 is selected for this algorithm.

For EM, log likelihood value increase with cluster size. The silhouette score is maximum again at 2 and 4. The Adjusted MI graph also shows 2 as the highest point after which it dips. Therefore, a k-value of 2 again is selected for this algorithm. So, using both the methods, optimal K for this dataset would be approximately between 2-4 clusters in higher dimension.

For the phishing dataset, as you can see, we achieve much more accuracy than the diabetes dataset. One reason for this is that the features represent the data well, as they

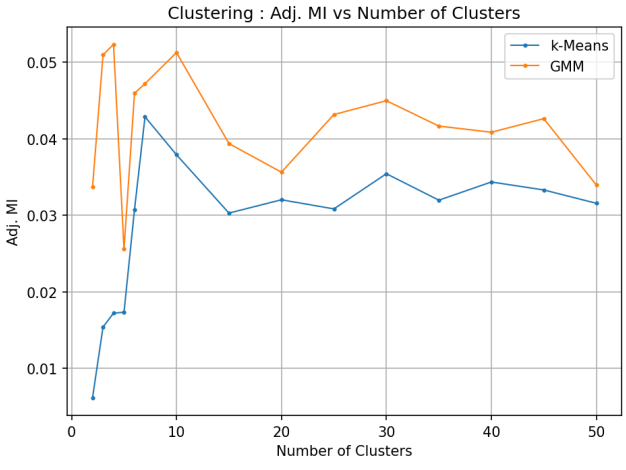


Fig 1

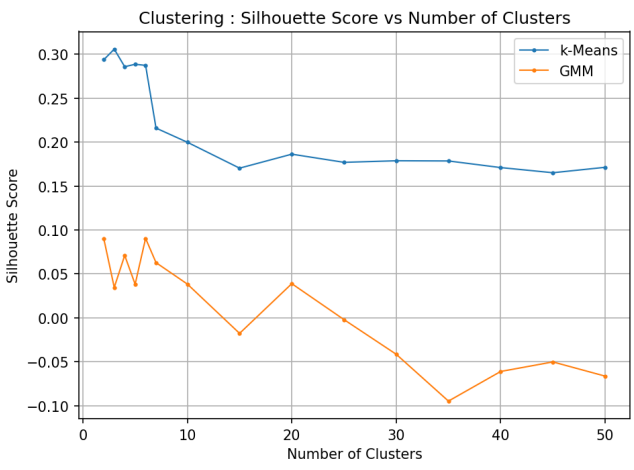


Fig 2

Fig 3

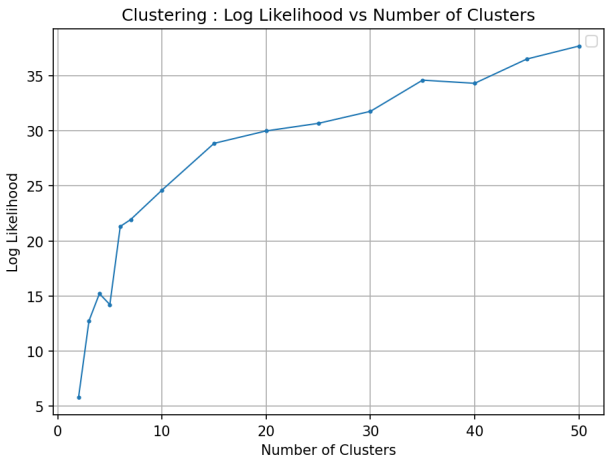


Fig 4

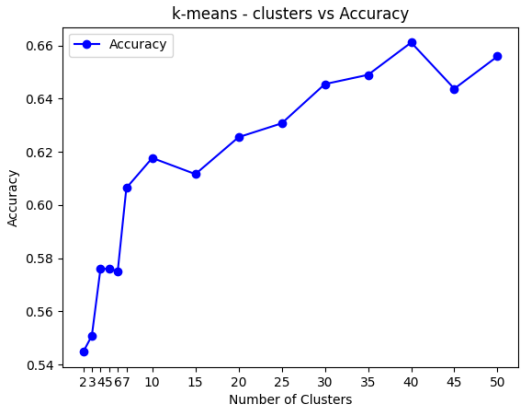


Fig 5

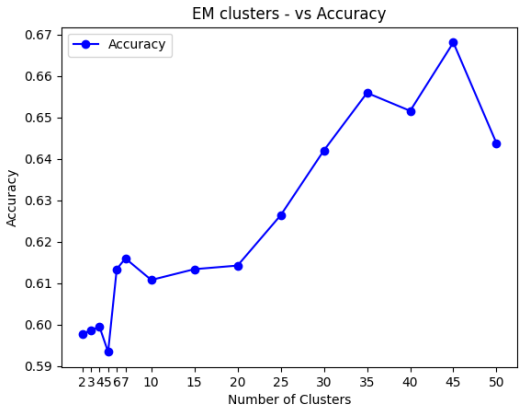


Fig 6

have binary values. The diabetes set had real values with less variance, but still different labels, which resulted in more misclassifications.

We use the same metrics to get the k-value for the phishing dataset. The silhouette graph shows that the optimum k value is 3, which is exactly the number of labels. The BIC graph confirms this. Using the elbow rule, k-value should be 2-3 as in the shown the SSE and BIC graph, as that is where it shows diminishing return.The adjMI plot shows that the k value should be around 2, but as the number of labels are 3, and the silhouette method confirms this, we take k=3 for the k-means clustering.

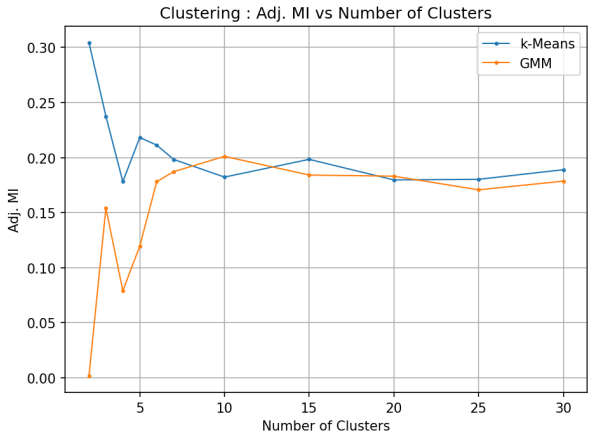


Fig 1

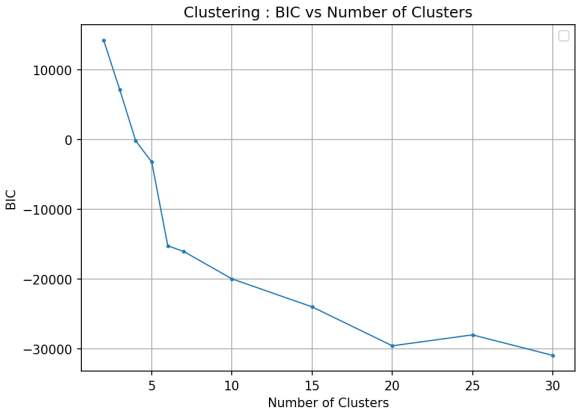


Fig 2

Overall I found the silhouette method to be more useful in representing data for the k-means algorithm.

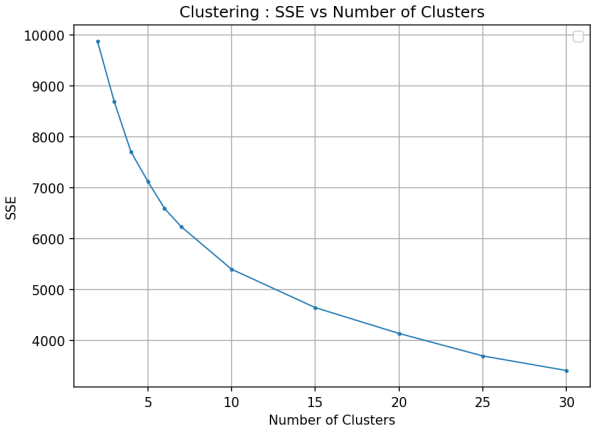


Fig 3

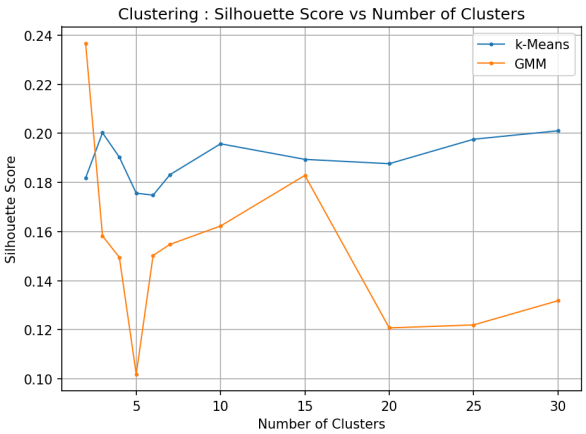


Fig 4

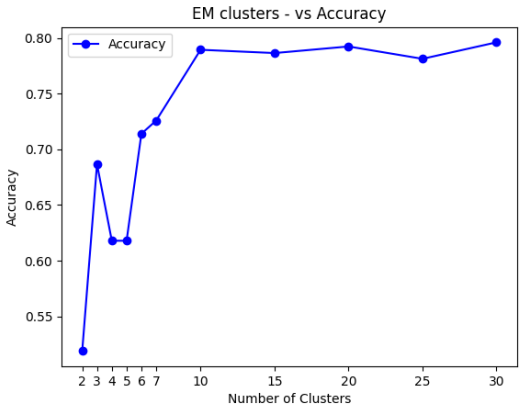


Fig 5

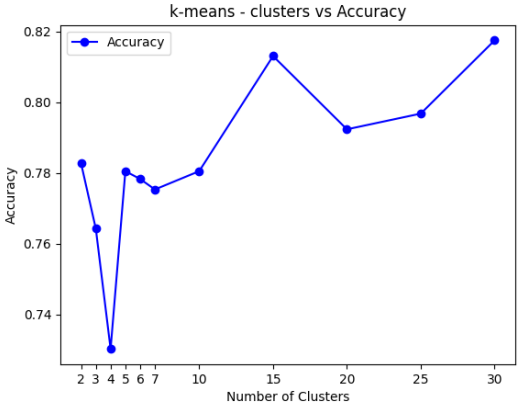


Fig 6

For EM, with gaussian mixture, the adjMi and the silhouette show different results. The adjMI peaks at 3, while silhouette peaks at 2, dips and 5 and then increases. This is reflected in the accuracy vd cluster graph as well (Fig 5.) It was difficult to determine a k-value using EM.

The diabetes dataset seems to have much lower adjMI compared to the phishing dataset for the k cluster range values.. This makes it difficult to cluster, as features are not seemed to be related to one another and do not provide information regarding various clusters when used together, in simple language. The k-means seemed to be performing better at clustering data rather than EM for the phishing dataset.

Dimensionality Reduction

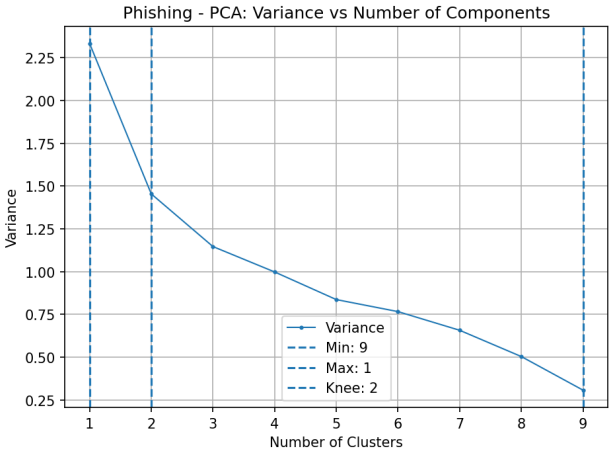
PCA, ICA, RP and RFE are implemented as dimensionality reduction algorithms and then re-slustered with k-means and EM.

PCA

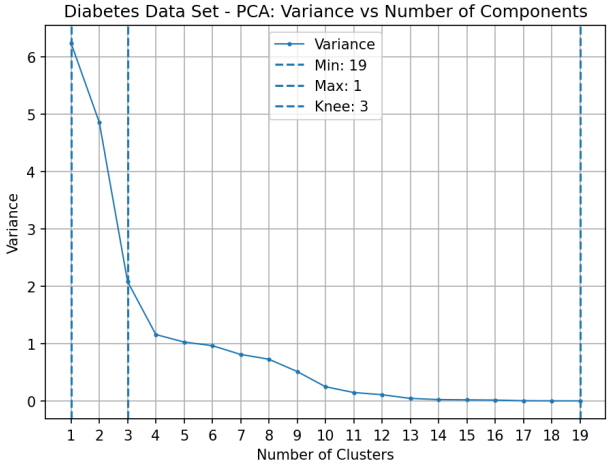
Principal Component Analysis, is a method that is often used to reduce the dimensionality of large data sets, by transforming a large set of variables into a smaller one that still contains most of the information in the large set. It uses orthogonal transformations to maximise the variance and generating independent components known as principal components.
In order to estimate the components to keep, we use a “knee value”. A knee or inflection point in the scree plot is used to determine the number of components to keep.

As shown in the below figure, the PCA knew value for the diabetes value is 3, while the knee value for phishing data is 2. The values are interchanged, but still they are representative of the data. For the diabetes dataset, the value of knee 3 represents about 60% of the variance. (Graph not included here due to space constraint). Similarly, for phishing around 70% is represented by knee = 2.

Maximum variance is present among the first few right angle projections, indicating that most of the dimensions are redundant in terms of information and they can be reduced to a smaller number of set. There is not much variation in the accuracy after we increase the dimension after the dimension of that more informative reduced feature set except small variation due to noise.



PCA 1



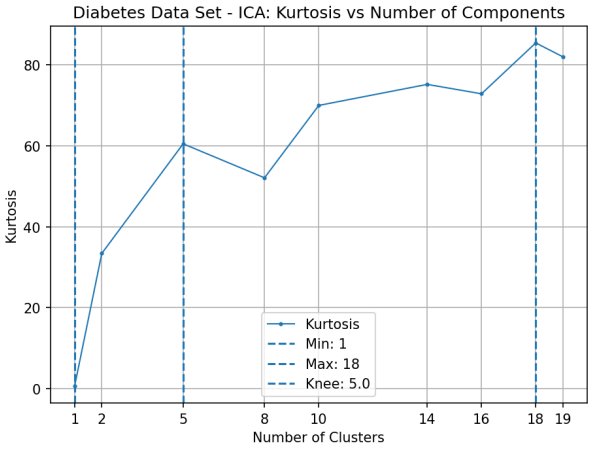
PCA 2

ICA

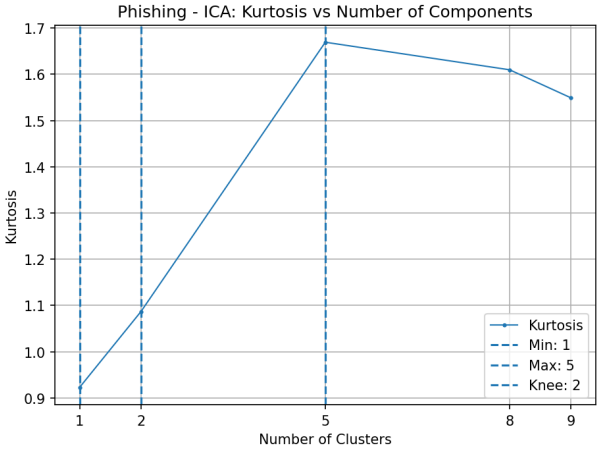
ICA is used to reconstruct a new feature space of independent components through linear transformation, to maximize non-Gaussianity. Kurtosis is used to measure this. It is the sharpness of the peak of a frequency-distribution curve. Data sets with high kurtosis tend to have heavy tails, or outliers.

Applying ICA to the two datasets, we get the following results from the graph below.

For the diabetes dataset, the knee value comes to be 5, and the maximum kurtosis is 18, which is actually representative of the data, as we have total 19 features. For the phishing dataset, the knee value is 2, with maximum kurtosis at 5 and minimum at 2. The number of components for the phishing dataset is 9, so a kurtosis of 5 is quite representative of



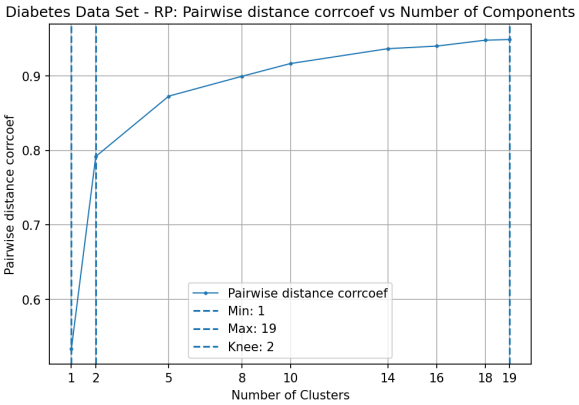
Caption



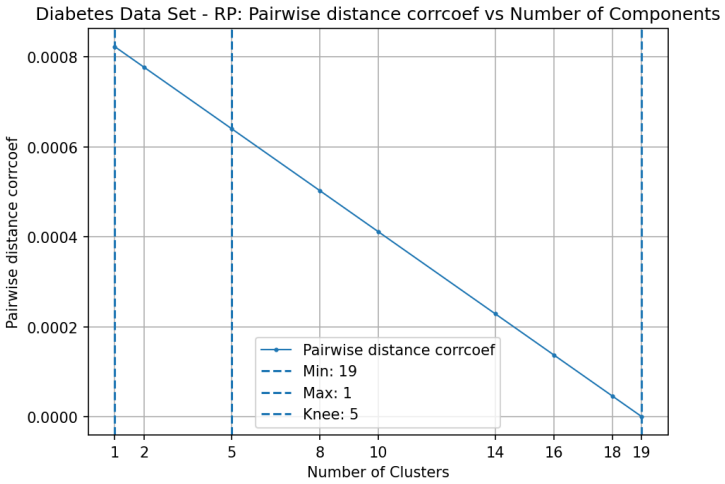
Caption

the data. However the kurtosis is a little lower than expected, indicating that there might not be an underlying distribution.

RP



Pairwise Distance



Reconstruction Error

Randomized Projections are known to best work on classification problems. In RP, arbitrary directions are generated and the dataset is projected onto these directions in a lower dimensionality space. In Random projection, distance based pairwise distance between any two instance of dataset is preserved. This is done by varying dimensions and distributions of random projections matrices. The pairwise distance coefficient plot gives the accuracy or optimum value of the knee.

From the pairwise distance plot, we can see that the knee value is predicted as 2 for the diabetes set which is very accurate as it has two labels. The reconstruction error starts decreasing after cluster size 2, with knee as 5, showing after 5 the reductions do not add any value.

For the phishing dataset, the plots show similar trends, but the knee value for the phishing set is 2. RP is not expected to perform as well as PCA, as it is more random, but in this case performs better for the diabetes dataset. It is also much faster. (Plots for phishing data set are not included as they show similar trends.

RFE

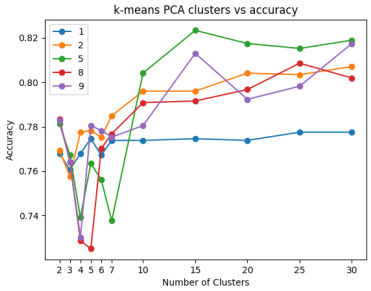
Recursive Feature Elimination is easy to configure and use as it selects those features in a dataset that are most relevant in predicting the target variable. There are two configuration options for RFE: the number of features to select and the algorithm used to help select features. Both of these params can be experimented with, although the performance is not only dependent on these hyper parameters being rightly selected.

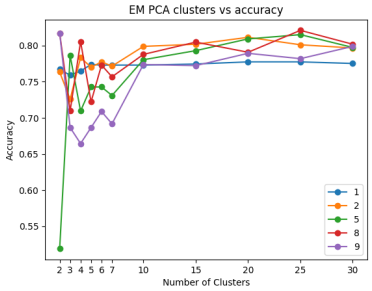
We use SVM as an estimator here with a linear kernel. For both datasets, the knee value from RFE comes out to be 2. This is accurate for diabetes data, but not exactly for the phishing dataset. I actually missed plotting the graphs for RFE alone, and did it for RFE + clustering.

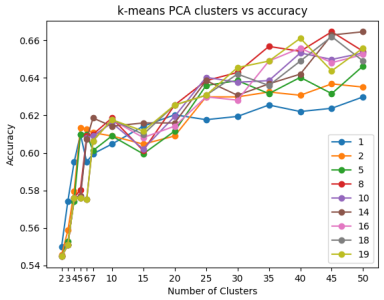
CLUSTERING & DIMENSIONALITY REDUCTION:

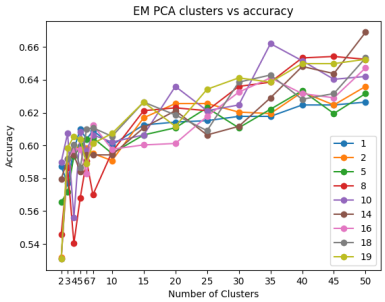
Let's look at the performance of various DR algorithms after cluster for k-means and EM.

PCA + Cluster - For Phishing and Diabetes Dataset Respectively

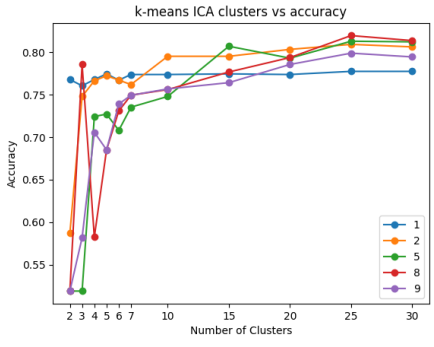


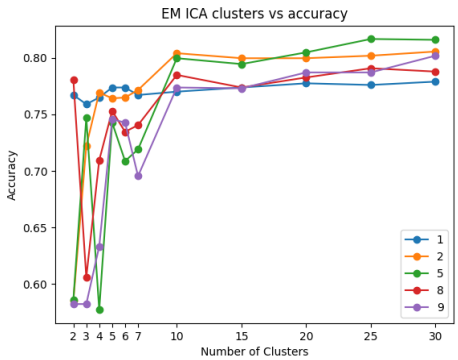


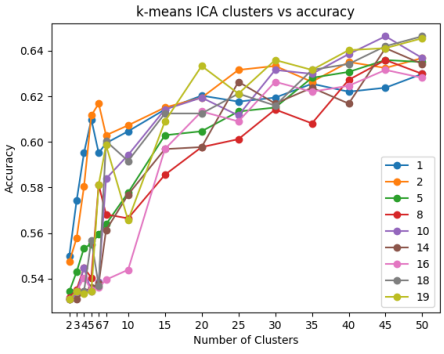


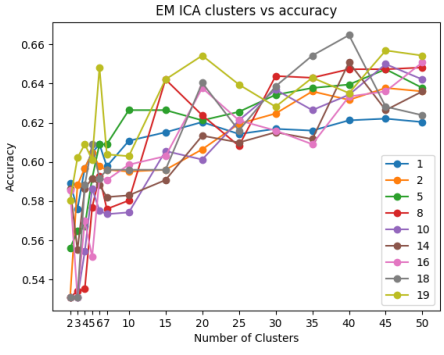


ICA + Cluster - For Phishing and Diabetes Dataset Respectively

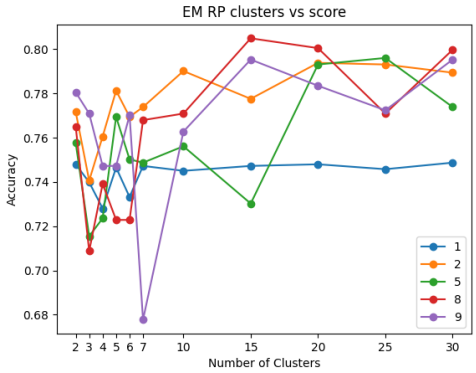


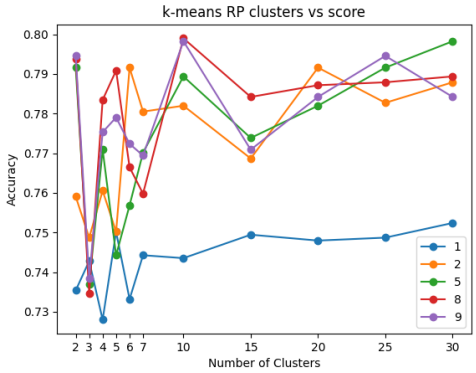


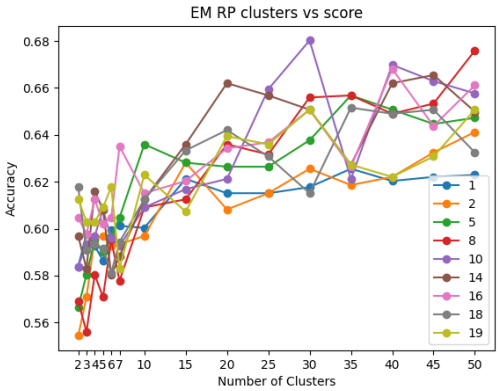


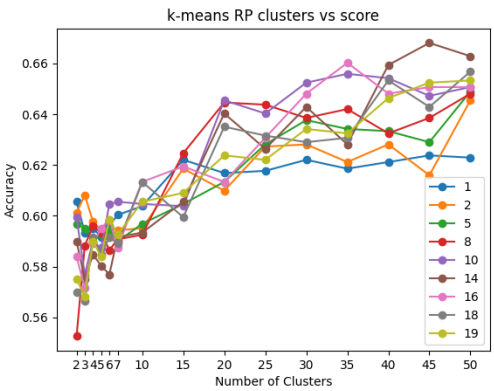


RP + Cluster - For Phishing and Diabetes Dataset Respectively

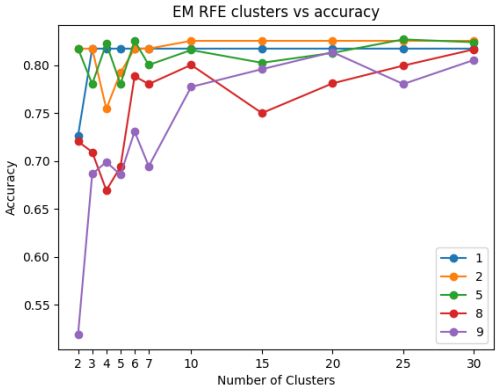


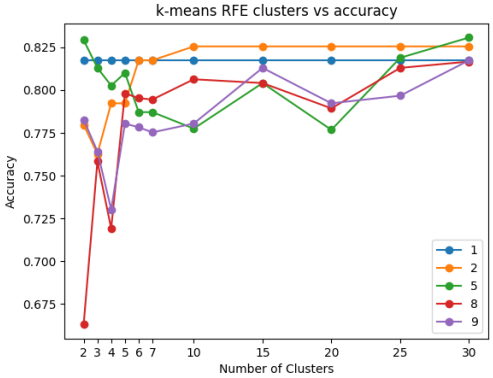


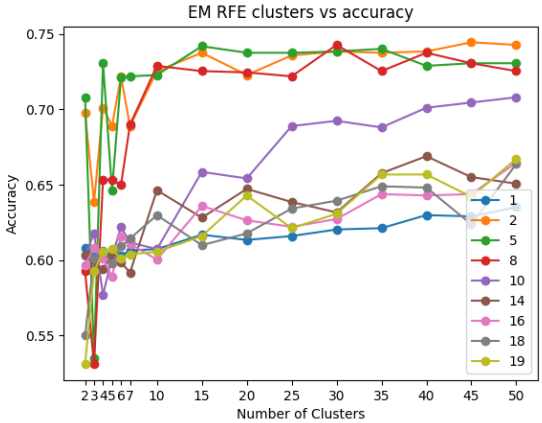


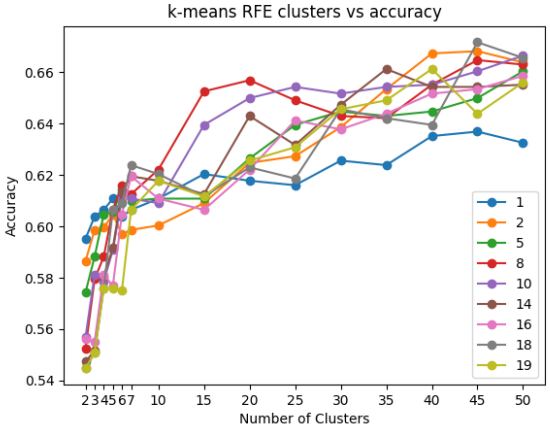


RFE + Cluster - For Phishing and Diabetes Dataset Respectively









Diabetes Dataset: Clustering with dimensionality gave varied results as you can see from the accuracy plots for k-means and EM for different DR algos. ICA and PCA gave a result similar to the results without clustering. RP performed a little worse for k-means and EM, giving component values as around 18 for the diabetes dataset which is not optimal and do not technically do any DR. RFE was surprising as it gave really good results, with best accuracy at k=2, which is equal to the number of labels. This is because SVM performed best on the diabetes dataset in Assignment 1. RFE with a SVM linear kernel represents that. In terms of performance, RFE actually gives the best accuracy (~75%) across all the altos, while the other ones have around 68%. This indicates that RFE creates good clusters with this dataset.

Phishing Dataset: For this dataset, dimensionality reduction methods and clustering actually increased the overall accuracy, for all the DR algorithms, for both k-means and EM.Again, PCA and ICA gave similar results as before, (without clustering)> But, RP performed better than the diabetes dataset, suggesting randomised transformations worked. This could be because the data is well clustered as well has less outliers. The adjusted MI across the plots suggest that the peak for both EM and K-means seem to be closer to 5, which validates with earlier values, but with increased accuracy. In order to improve performance in the future, I can try running with larger k-values.

DIMENSIONALITY REDUCTION WITH NEURAL NETWORK:

	NN - Original -Without clustering	PCA	ICA	RP	RFE
Fit & Predict time	3.02	2.24	4.42	2.345	1.67
Train Accuracy	92%	62%	59.4%	68.4%	71.45%
Test Accuracy	90%	60.345%	54.6%	65%	69.78%

CLUSTERING WITH NEURAL NETWORK:

For this part of the assignment, I am going to use the Diabetes dataset. We are going to apply a neural network, using clustering as an extra attribute. We compare the results of the accuracy and time scores for NN with clustering and without clustering. The optimal value of k was set to 2 for k-means and 3 for EM for the diabetes dataset.

EM	NN - Original -Without clustering	PCA	ICA	RP	RFE
Fit & Predict time	3.02	1.24	2.45	2.3	1.56
Train Accuracy	92%	42%	61.8%	70%	79%
Test Accuracy	90%	39%	59%	65%	76%

The benchmarked neural network still performs the best among all algorithms, without any clustering or DR. For EM, RFE performed the best and for K-means, RP performed the best in terms of accuracy. Still, they could not achieve the 90% accuracy that the neural nets achieve. The slowest fit and predict time was for ICA. PCA performed the

worst for both k-means and EM.From the above result it seems that no pre-processing of data is required for applying neural network algorithms. Neural network which runs directly on the dataset with clustering, then the time taken of neural network to run as compared to time taken to run on dimensionally reduced dataset would be much higher as the dataset would have all the features while the reduced dataset would have less number of features.

OBSERVATIONS:

1. The clustering accuracy is quite low, compared to NN, which shows that the data is not very well cluster able in high dimensions.

2. Silhouette method is more reliable than the elbow method, when determining k-value, as the clustering is not very good.

3. The phishing dataset performs better with clustering rather than the diabetes dataset.

4. In case of ICA, RP and RFE, the cluster accuracies are higher before and after applying DR.

5. Most variation is present in the first few components.

6. Reconstruction error is quite low for both datasets.

7. Reconstruction error decreases suddenly after the first few components, because they are the most important.

8. Neural networks perform much better than both clustering and dimensionality reduction.

9. Overall I observed that EM works best for clustering, while RFE worked best among the dimensionality reduction algorithms.

REFERENCES:

The plotting code for the data is referenced from

https://github.com/bryanmarthin/Gatech-CS-7641-Assignment-3

https://github.com/huhu42/Unsupervised-Learning/tree/e36526786cb8c6e72b10eaecf9ae2f2e0445db11

Other references are mentioned in the README.txt due to space constraints.