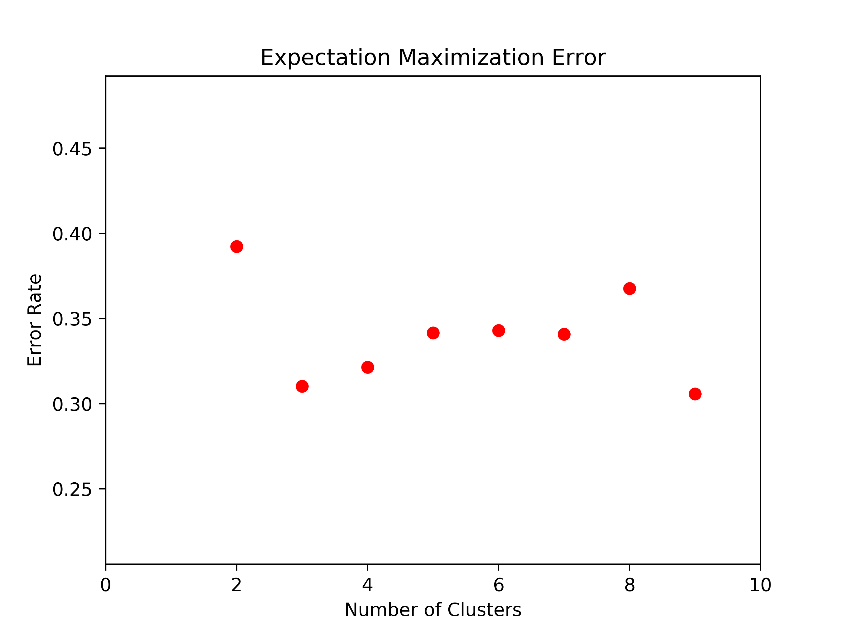
**Datasets**

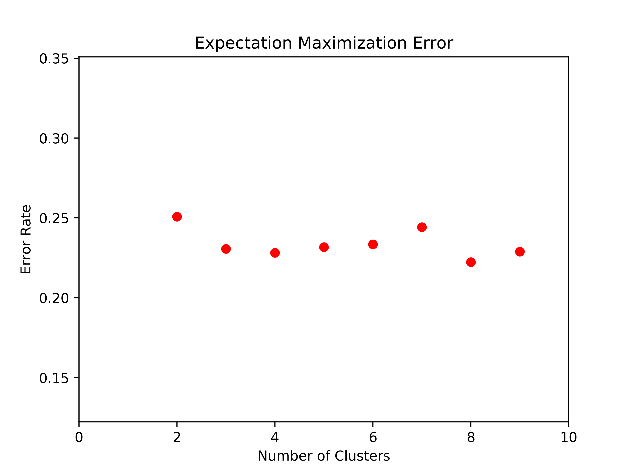
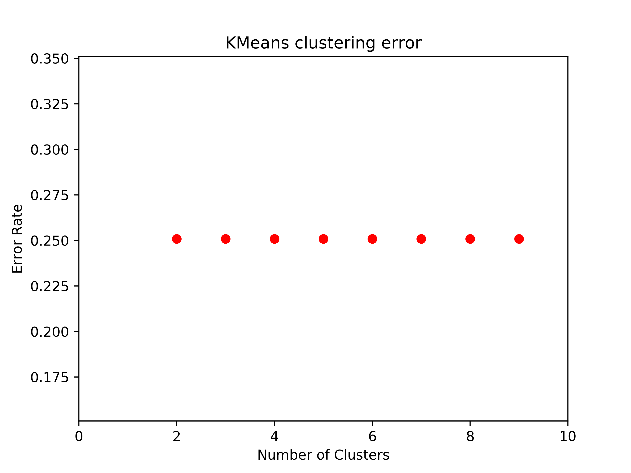
For this assignment, I am using the same datasets as used before in assignment 1. The two datasets chosen for this project are the Spambase Data Set and Adult Data Set from UCI’s machine learn repository. The spam data set contains 4601 instances gathered from work and personal emails and 1813 were categorized as spam by the individual, using a binary classification of 0 and 1. Each instance is made up of 57 attributes which were either a continuous real number or integer. Example attributes include frequencies of different words, characters, and the length of certain continuous sequence of characters. The adult data set contains 48842 instances gathered from the 1994 Census database. Each instance consists of 14 attributes and a binary classification of labels '>50K' and ‘<=50K’. The attributes were a mix of integer values and text values such as race, occupation, sex etc. The spam dataset contains many attributes which would make it a great candidate for applying dimensionality reduction algorithms.

**Clustering: K-Means & Expectation Maximization**

I first ran the K-Means algorithm on both datasets using the Euclidean distance between points as the distance metric. For the parameter k, I chose k to equal the number of possible values for classification which is 2 for both datasets. Due to domain knowledge, we know that 2 is the true number of clusters for both datasets. Those results I used as the *ground truth* and I used that to evaluate performance and calculate the error rate of the resulting clusters created. My approach was to run KM with differing values of k and compare and calculate the error of the resulting labeled clusters each datapoint belong to with the *ground truth.* The graphs below show the results after running KM and EM on the spam dataset



The graphs below show the results after running KM and EM on the adult dataset



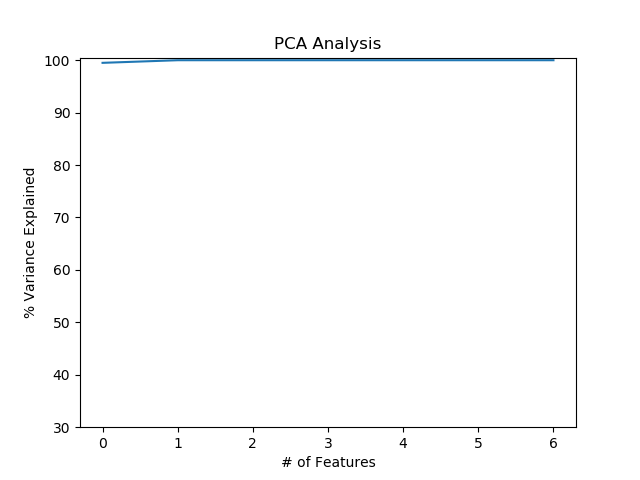
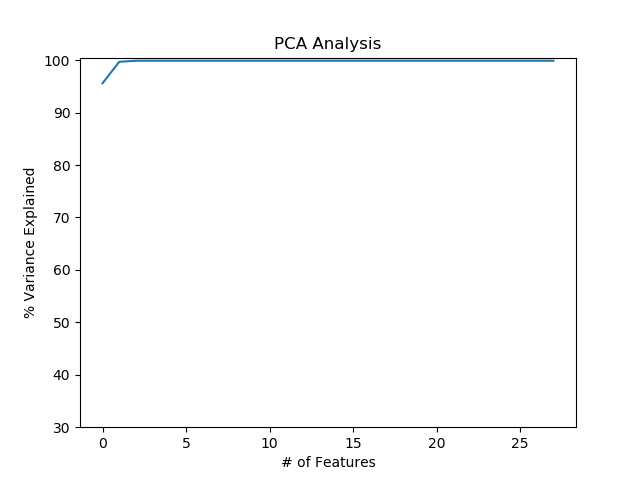
As seen in the graph for KM, increasing the number of clusters showed a steady decrease in error rate. Despite the fact that there are only 2 classes in the spam dataset, 2 clusters gave the highest error rate and 7 seems to be the optimal number of clusters. This tells us that the spam dataset causes the KM algorithm to find more specific differences and nuances in among the data points and group them accordingly. This also tells us that the dataset contains numerous irrelevant attributes that don’t contain important information for classification that KM takes into account anyways. EM gives slightly better results as 3 clusters produced the lowest error rate and increasing the clusters produces high error rates before drastically lowering for 9 clusters. The improved results seen from EM compared to those of KM can be due how KM is more biased towards spherical, evenly sized clusters while EM is able to accommodate clusters of more variable sizes. The spam dataset does not have an even split of classes and a large amount of attribute which contributes to differences in KM and EM’s results. There is also the curse of dimensionality as the large of attributes will cause KM and EM to perform worse. For the adult dataset, KM performed the same no matter the number of clusters. This could be due to that fact that the data is more separable. Clustering on the adult dataset had lower error compared to the spam dataset since there are les attributes and thus a lower dimensionality, making clusters more differentiable. The spam dataset could have attributes that were more correlated resulting in more overlapping clusters.

**Dimensionality Reduction**

Applying dimensionality reduction should see better clustering results as the algorithms will select relevant attributes that will make clustering easier. The four dimensionality reduction algorithms used were PCA, ICA, randomized projections, and K Best.

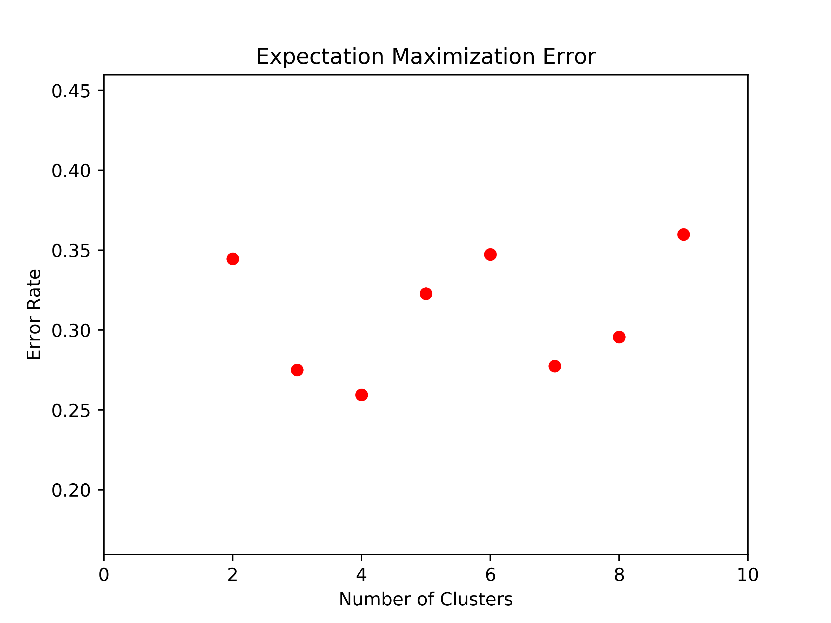
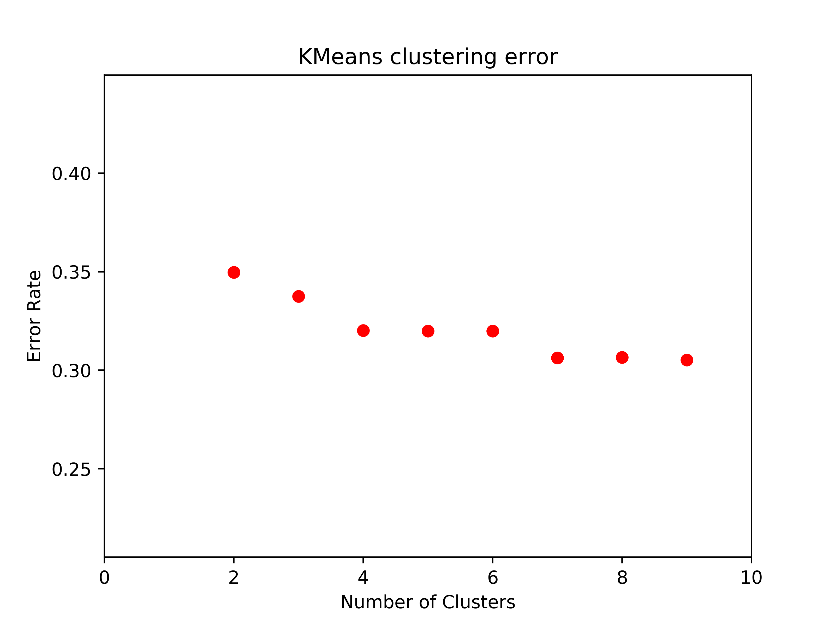
**PCA**

Spam Adult



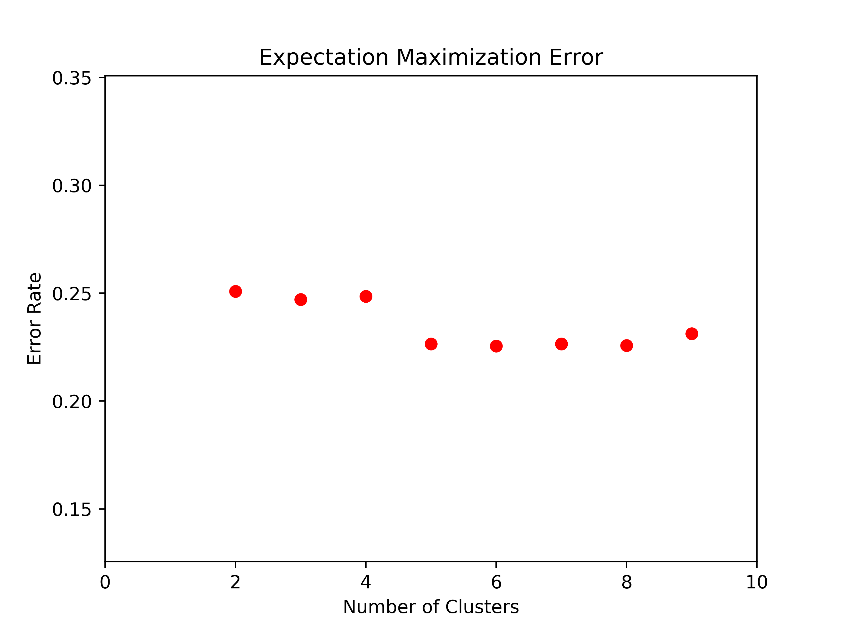
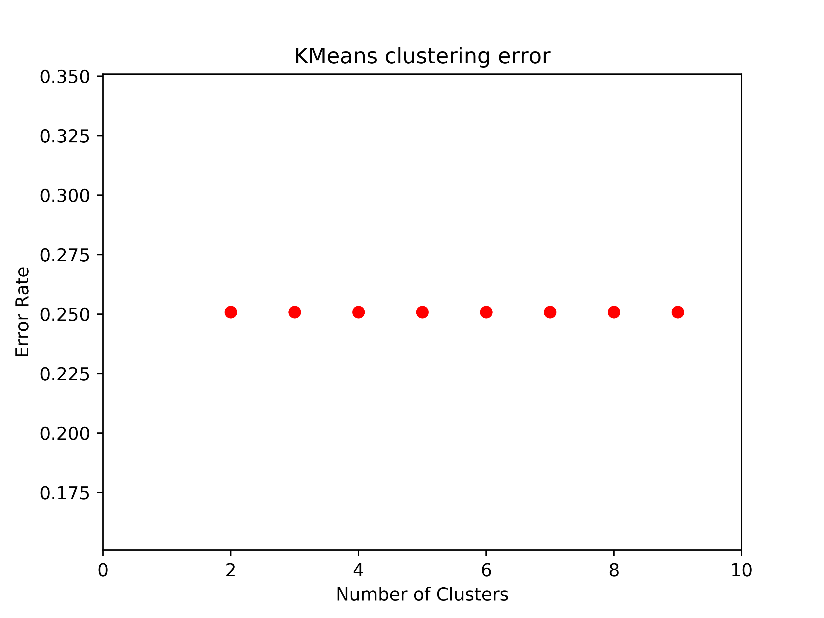
For applying PCA on the datasets, I used sklearn’s PCA algorithm and varied the number of components to keep. Plotting the number of features kept along with the variance in data shows a high percentage of variance that is covered by a very small number of features. For spam, the principal component covers 95.63% of variance and the second component covers 99.76% of variance. Each additional component shows diminishing returns as the amount of variance covers becomes minuscule.

Spam



After applying the clustering algorithms on the dimensionality reduced dataset, I got the above results for the spam dataset. For KM, I obtained the same error rate for the clusters as before. PCA is not good for discrete values which makes sense as the spam dataset contains all discrete values. Applying EM after PCA on the spam dataset shows that the feature transformed dataset has overall a lower error rate. But as a whole, PCA did not improve the accuracy of the clustering algorithms by much.

Adult

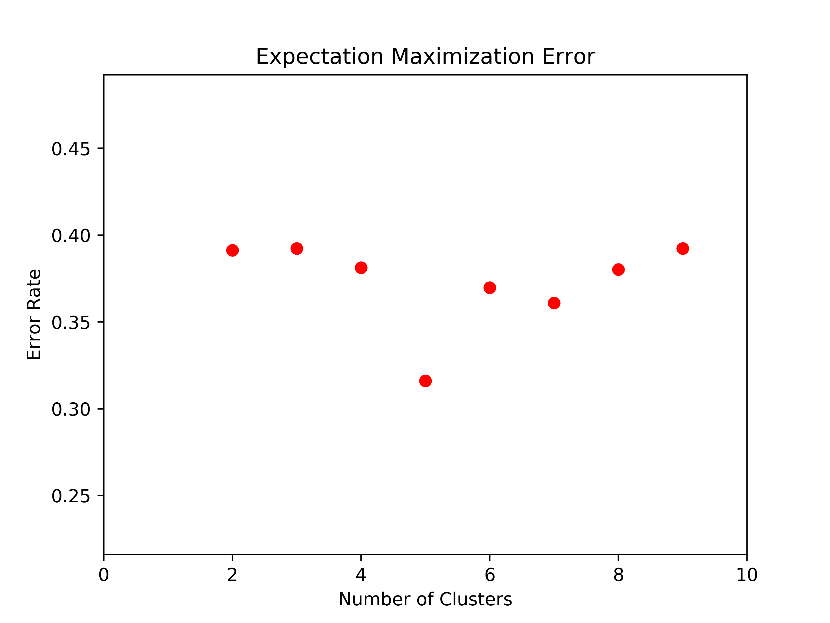


For the adult dataset, the error rate remain consistent for KM and EM but with a drop in error rate for EM. PCA generally performs better for non discrete values as the adult dataset attributes were mainly categorical values.

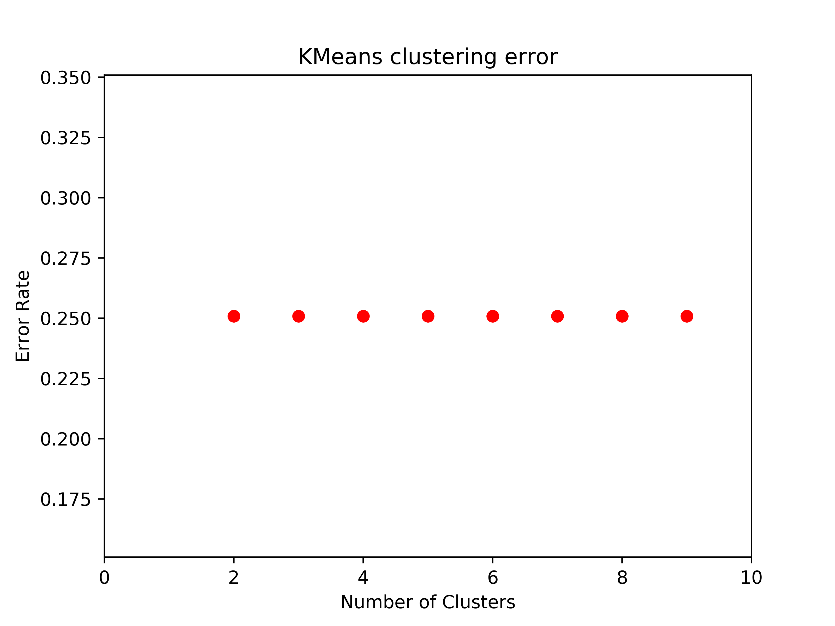
**ICA**

Applying ICA and running the cluster algorithms on the spam dataset produced the following graphs.

Spam



Adult



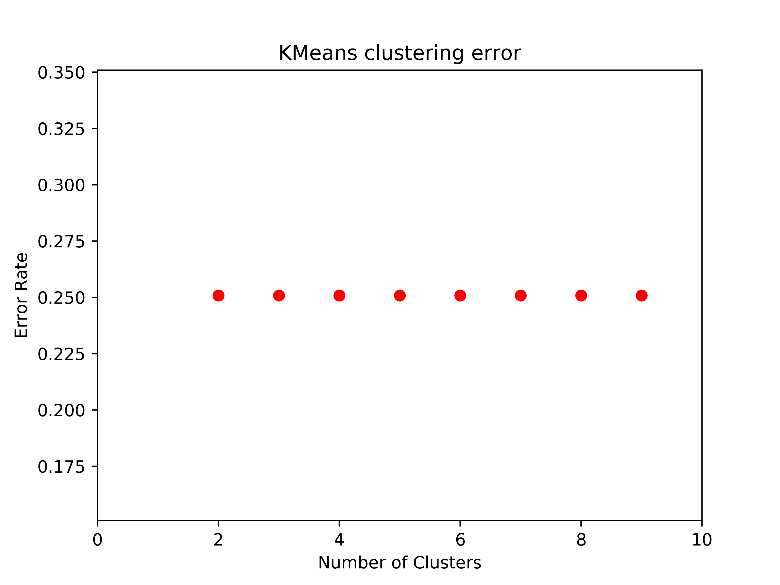
The kurtosis values produced for spam and adult datasets differed greatly. The spam data produced much higher kurtosis values compared to the adult dataset with the max value being 52.012075481962 versus 8.936061133735953. The spam dataset has several kurtotic attributes that varied along with other attributes which the adult dataset had more uniformly distributed attributes that produced independent components that were more Gaussian in nature.

**Randomized Projections**

Spam



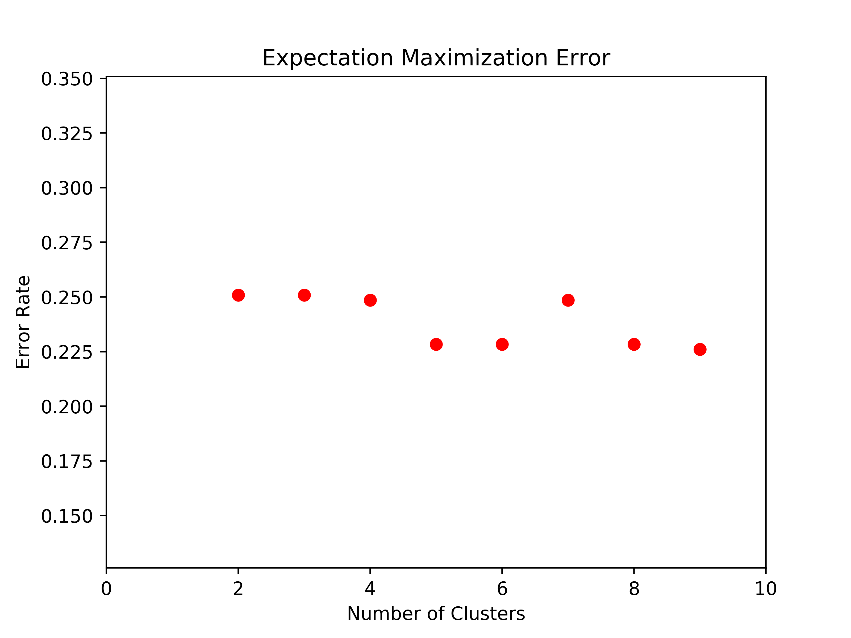
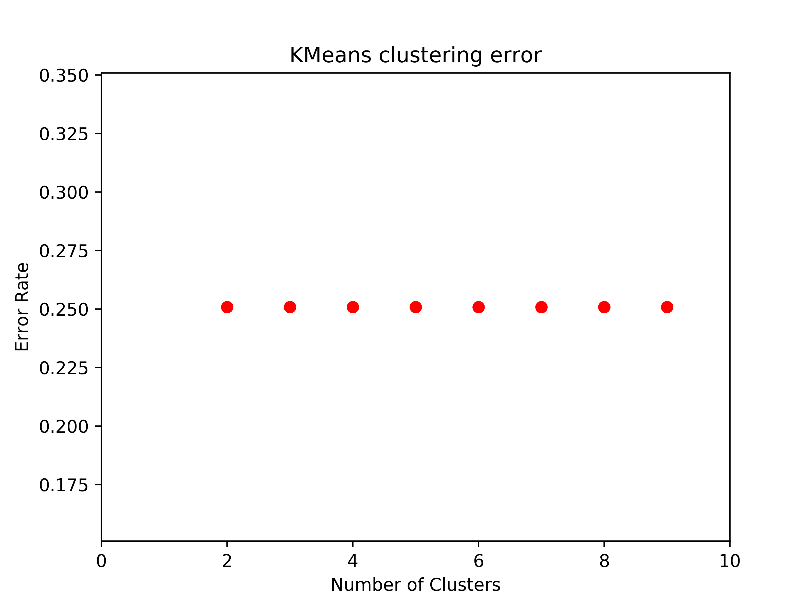
Adult



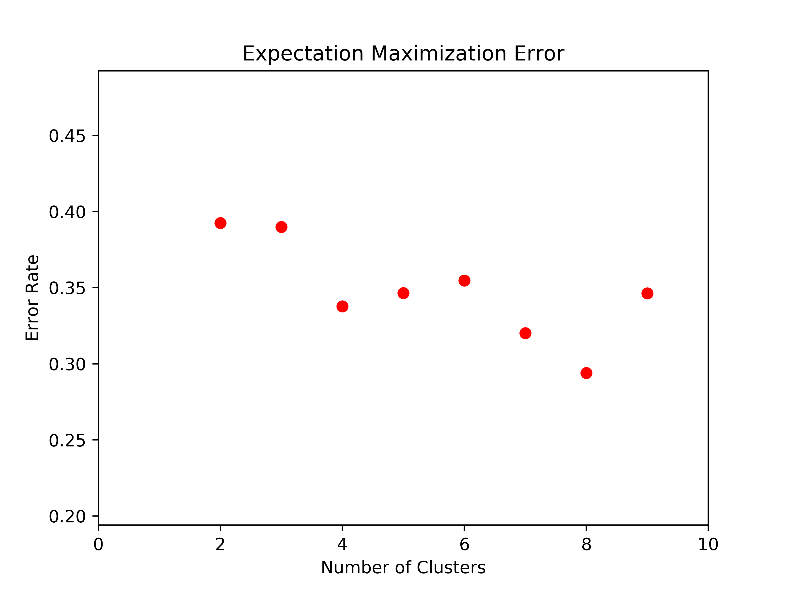
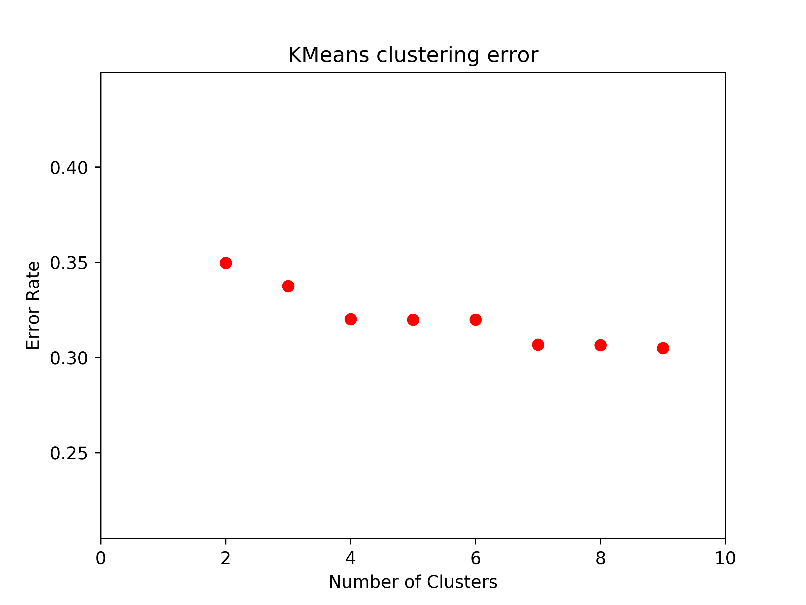
Random projection did the worse for the spam dataset out of the algorithms.

**K Best**

Adult



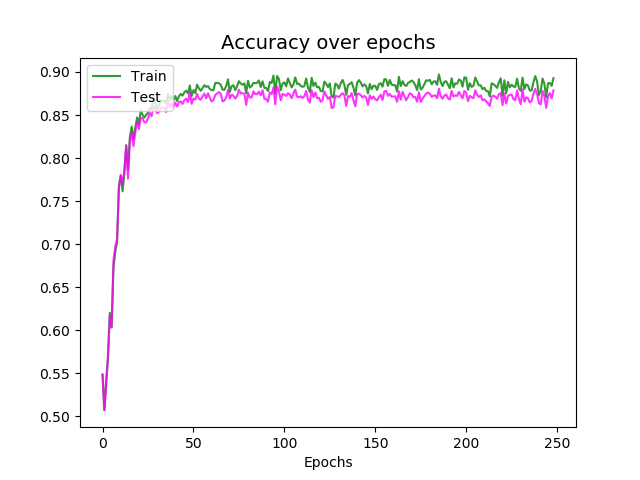
Spam



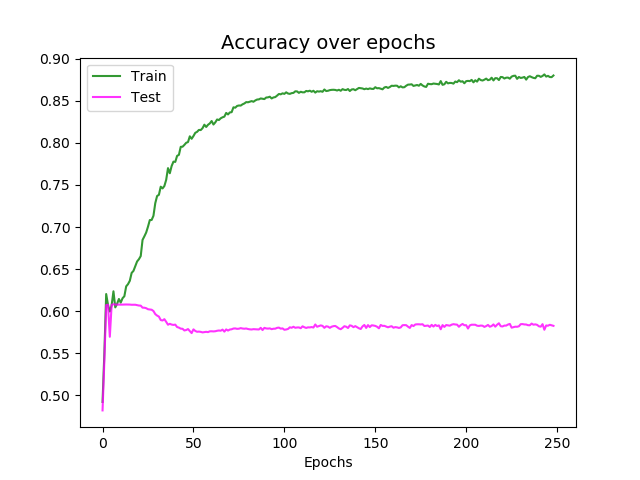
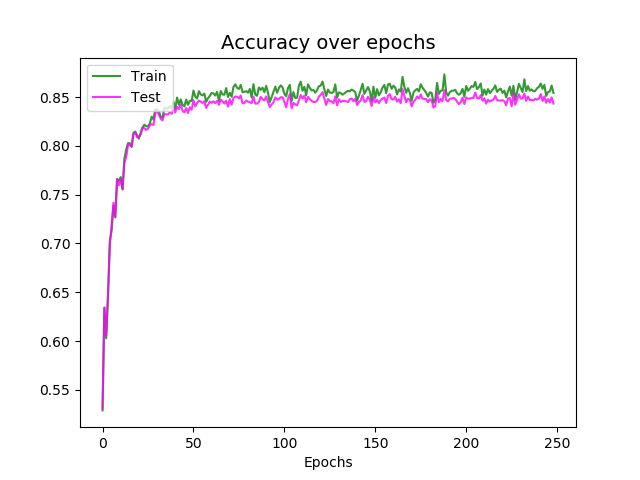
K-Best keeps the top scoring features measured by the chi square test which measures dependence between variables. It selects the features with the highest values relative to the classes and thus weeds out features that are deemed independent of class and irrelevant for classification.

**Dimensionality Reduction + Neural Network**

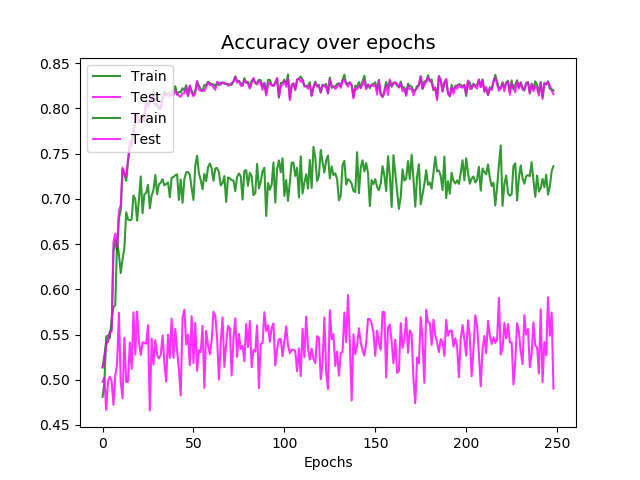
Neural Network

****

PCA ICA



KBest (Top) RP (Bottom)

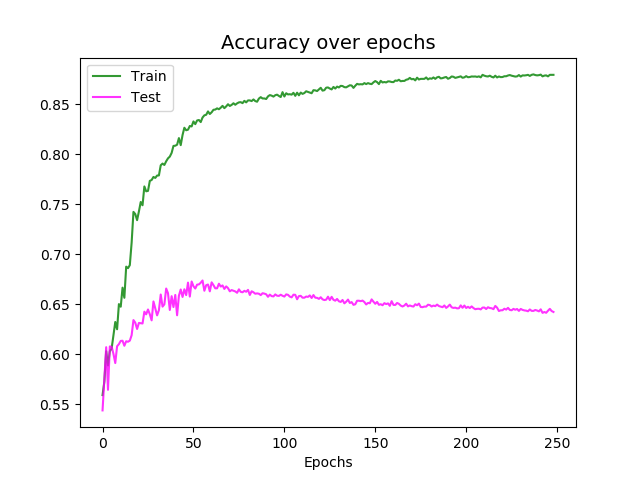
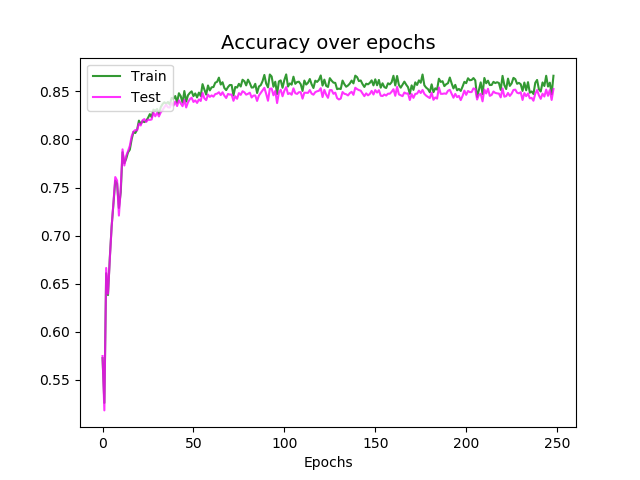


|  |  |
| --- | --- |
| PCA | 1223.16200018 sec |
| ICA | 2486.98800015 sec |
| RP | 712.198999882 sec |
| KBest | 1109.75899982 sec |

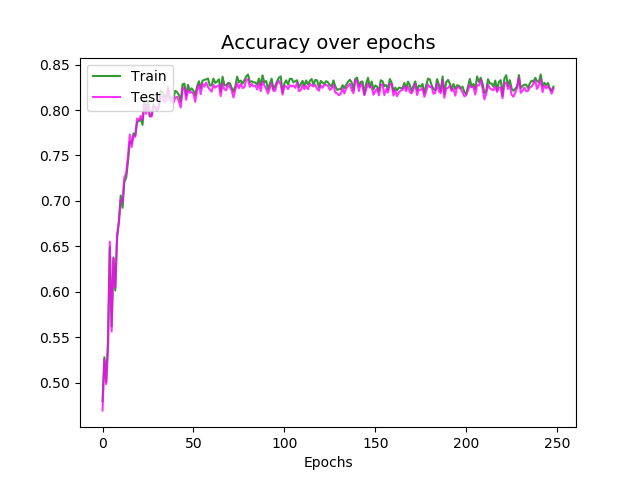
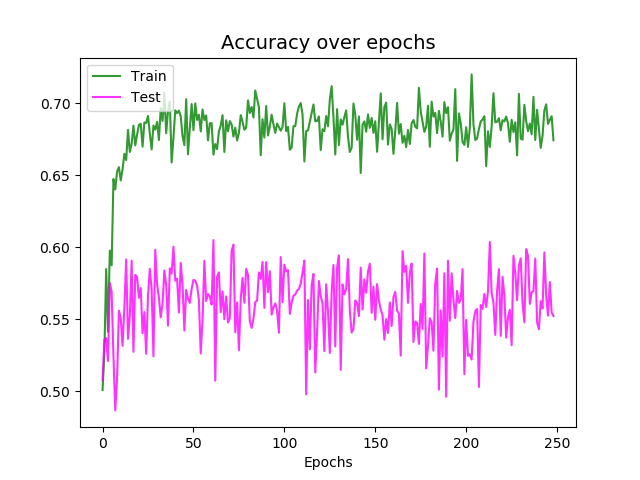
I’ve run the neural network after applying the dimensionality reduction algorithms. Overall, PCA and K-Best seemed to performed the best with the neural network since PCA keeps the features with the greatest eigenvalues and variance explained and K-Best keeps the top features. ICA and RP both had high training accuracy but low testing accuracy which indicates overfitting which is strange. RP perform relatively worse out of the four algorithms which might be because it chooses attributes that aren’t as relevant to classification due to its random nature. Although those features may best separate the data, there is a loss of information that causes it to ignore the combination of attributes that are best for training. Overall, training and testing took less time with dimensionality reduction preprocessing than without. Since those algorithms reduces the number of features and redundancy, the hypothesis is smaller as a result. However, there is a tradeoff because aggressively reducing features can hurt accuracy as it can remove useful information for classification. When comparing the results to the neural network performance without any dimensionality reduction, all four algorithms had lower accuracies. This could be because useful and discriminative information may be part of components with low variance thus resulting in poorer performance.

**Clustering + Neural Network**

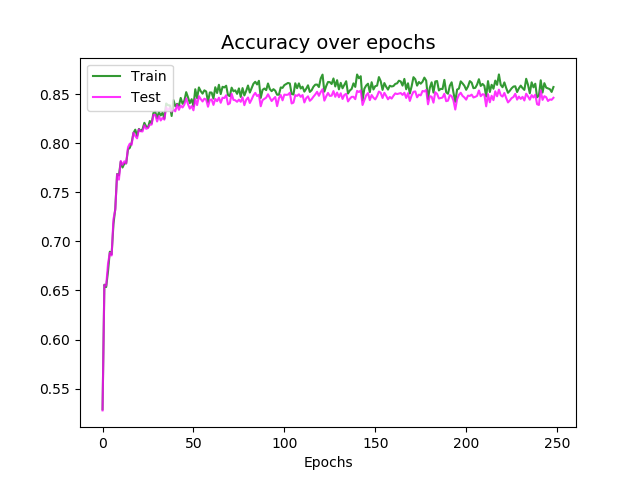
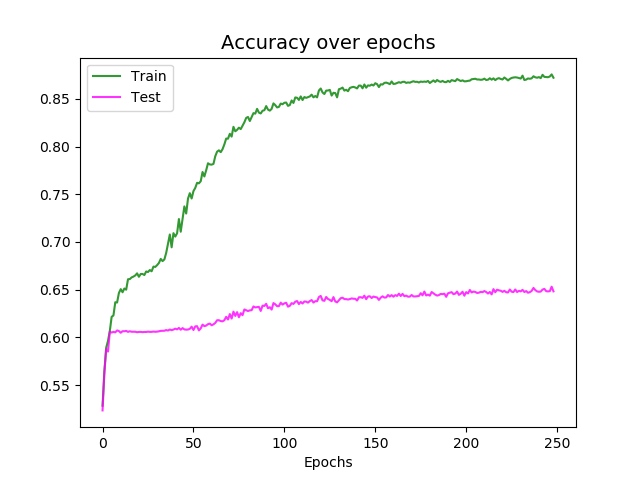
EM + ICA EM + PCA

****

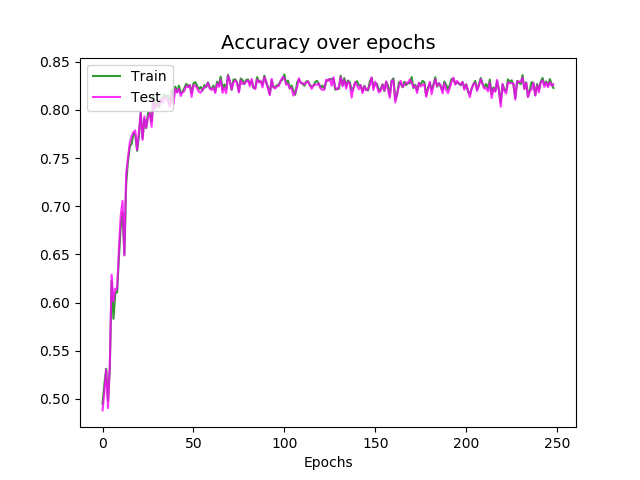
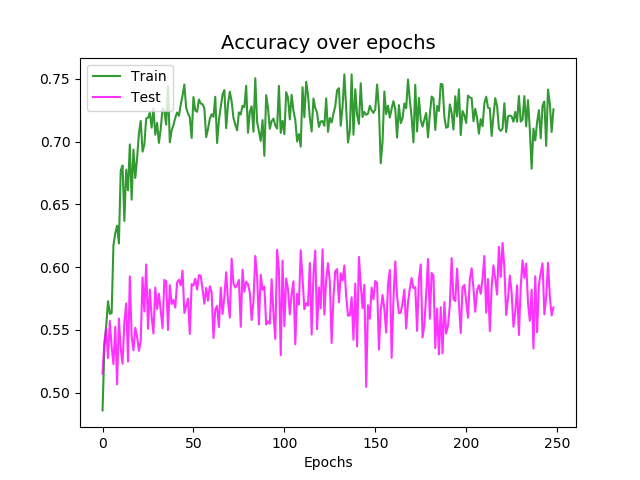
EM + RP EM + KBest



KM + ICA KM + PCA



KM + RP KM + KBest



Adding clustering to the dimensionality reduced datasets and running the neural network on it produced the graphs above. Clustering adds structure to the data which along with dimensionality reduction, reduces the size of the neural network and time needed for computation. Again, adding clustering did not give better results compared to just the neural network. Performance compared to just adding dimensionality reduction looked overall similar and slightly better for ICA and RP. The training time was slighter higher when clustering was included but was still lower compared to without clustering and dimensionality reduction.