## Dataset Descriptions:

The first problem deals with a person's income in relation to several attributes. The dataset "adult" consists of attributes age, work class, final weight*, education, marital-status, occupation, relationship status, race, sex, capital-gain, capital-loss, hours-per-week, and native country. The prediction task is trying to determine based on these attributes if the person makes either >50k or <=50k. The dataset is large with 32,561 instances, and 7% had missing values which were replaced with the mode (nominal) or mean (numeric) of the missing attribute (using a filter). The dataset is also quite imbalanced with approximately 76% in the <=50k class and 24% in the >50K class.

This problem is interesting in a practical sense as income is a measure of success in our society, and factors contributing to success should be study closely, as statistical advantages and disadvantages in certain groups can be evidence of societal issues. Analyzing data like this can provide insight into income inequality, and specifically the effect of race, nationality, and sex on income. From a machine learning perspective, I think this data is interesting as it is a large data set, consisting of 14 attributes with 8 nominal and 6 numeric.

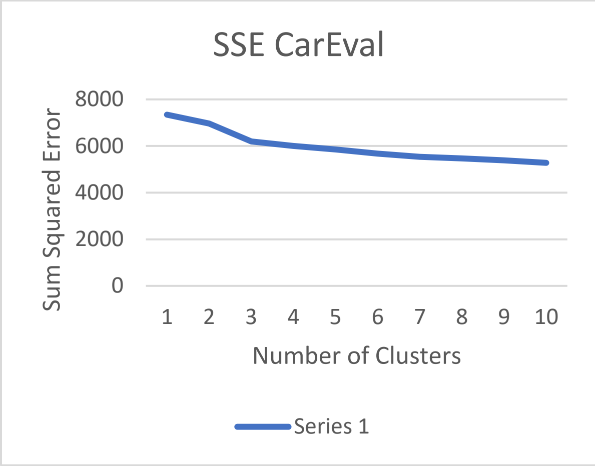
The second problem deals with evaluating cars as either unacceptable, acceptable, good, or very good. The dataset "car" consists of discrete attributes: buying price, maintenance price, doors, persons, lug_boot (trunk), and safety. This data set consists of 1,728 instances, and had no missing values. For this assignment, I split the dataset 70%/30% into a training set and a testing set, respectively. The data set is highly imbalanced with a split of 70%, 22%, 4%, and 4% between classes.

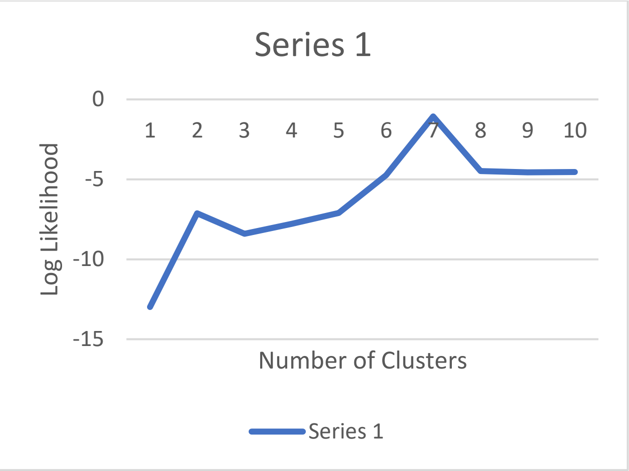
This problem is interesting mainly from a machine learning prespective. This dataset is much smaller than adult, which makes algorithm such as SVM and Neural Networks much faster, allowing all of the data to be used. The imbalance is also much higher with only 4% of the training data in the good and very good classes.

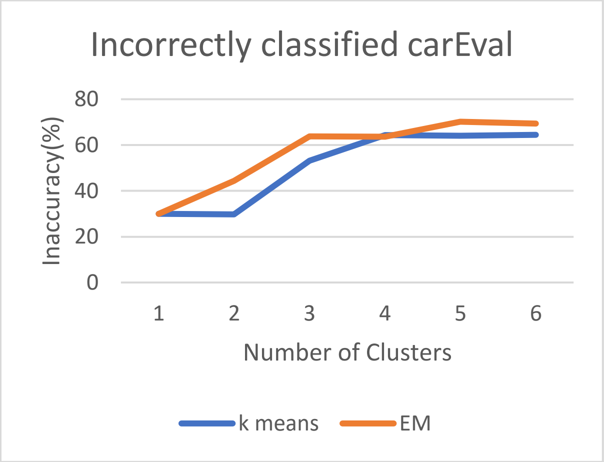
For this assignment, in order to maintain consistency, I ran each data set through Weka's NominalToBinary filter since PCA in Weka already does that. After which adult has 104 attributes and car has 21 attributes.

## Clustering:

For each Clustering Algorithm, I used Euclidean distance as the measure of similarity between instance. I graphed the effect of the number of clusters on the Sum Squared Error (for K means), the log likelihood (for EM), and the classification error for both datasets.





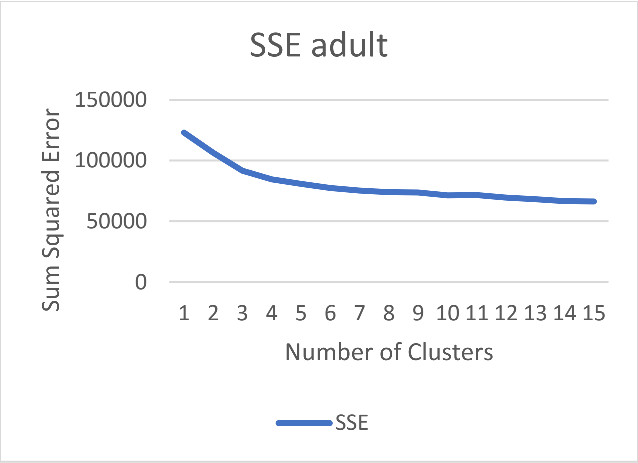


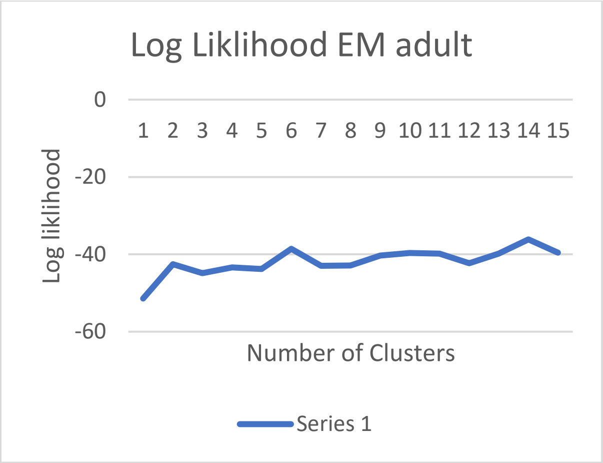
For the Car Evaluation dataset, we can see that as expected the sum squared error decreases and the log likelihood shows an upward trend as the number of clusters increases. Using the "elbow method," I chose 4 clusters to be "optimal" since the slope begins to decrease at that point, i.e. as you increase the number of clusters beyond 4 the improvement on the sum squared error becomes less and less per cluster. I chose the same number of clusters for EM, as to better directly compare the differences of the two algorithms.

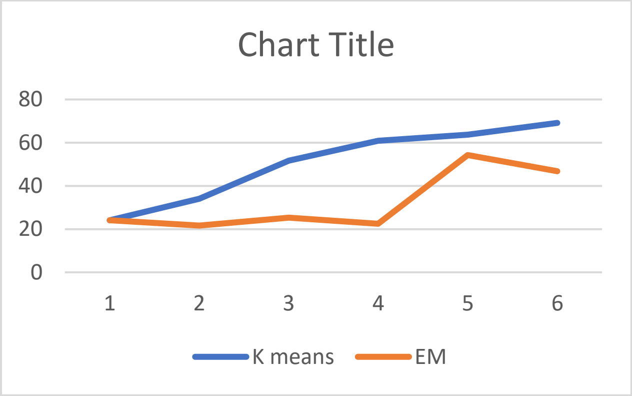
For the incorrectly classified graph, I do not believe these numbers are particularly meaningful as this dataset is quite unbalanced with almost 70% of the data in one class, which is why 1 cluster does so well at classification. Also, after the number of cluster becomes higher than the number of classes, Weka only assign each class to one cluster, so if the clusters are evenly spread the incorrectly classified percent will increase by a lot.

| Class (Kmeans, EM) | Cluster 1 | Cluster2 | Cluster3 | Cluster 4 |
|---|---|---|---|---|
| unacc | 100%, 100% | 56%, 62% | 54%, 48% | 100%, 100% |
| acc | | 32%, 31% | 34%, 35% | |
| good | | 6%, 7% | 6%, 5% | |
| vgood | | 6%, 0% | 6% 12% | |

We can see from the above table, which shows the class breakdown of each cluster, that the clusters did not line up particularly well with the classes, which explains the poor classification percentages. From a visual perspective, there seemed to be little distinction between the cluster in any 2-dimensional plot.







For the adult dataset, the same trends as described above are still present in the sum squared error and log likelihood. For K means, I chose somewhere between 3 and 6 clusters using the "elbow method;" however from a classification standpoint, 2 clusters would be a much better choice. The same can be said for EM, the loglikelihood graph shows 6 would be a good choice; however, the incorrectly classified graph has the incorrectly classified percentage to be much lower with only 2 clusters. I chose to go with 2 clusters so as to use the incorrectly classified statistic to further evaluate clusters after the dimensionality reduction.

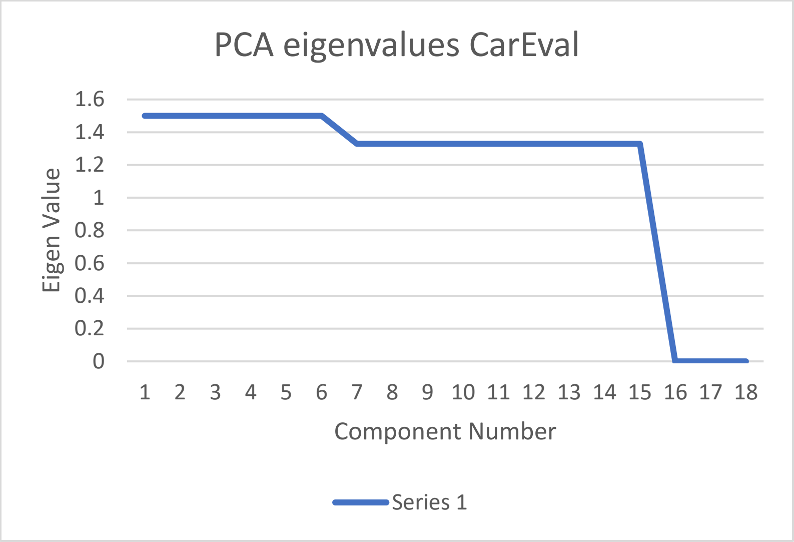
As with the car evaluation dataset the k means clusters did not line up particularly well with the classes; however, EM does quite a bit better at least having the >50K class as a majority in one cluster, this is reflected in the incorrectly classified percentage with EM at 21% and K means at 34%.

The table below shows the class breakdown of each cluster.

| Class (Kmeans, EM) | Cluster1 | Cluster2 |
|---|---|---|
| <=50K | 60%, 81% | 94%, 40% |
| >50K | 40%, 19% | 6%, 60% |

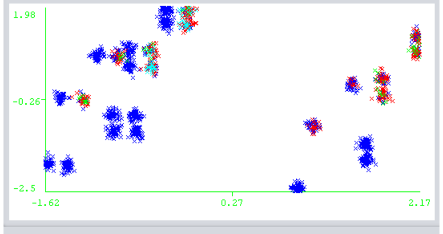
## Data Reduction and Clustering:

**Principal Component Analysis:**

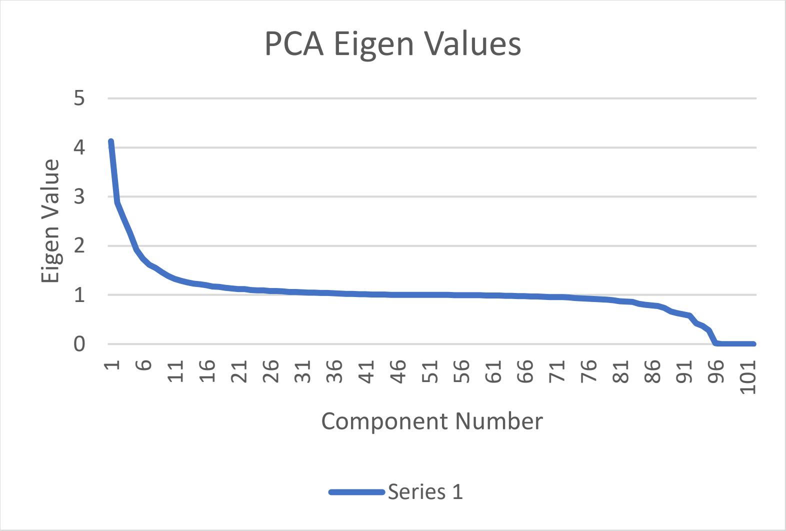


The graph above shows the eigen values of the principal components of the Car Evaluation dataset. We can see for principal components 16-18 the eigen values are almost 0, hence we gain very little from including these components in our reduction. So, I chose to reduce to the first 15 principal components. This covers of 0.95 of the variance of the data.

The plot below is of the first two principal components and the colors represent the classes, we can see the blue classes is almost easily separable from the other classes; however, the other three classes are all jumbled together, so it is pretty clear more principal components are need in order to accurately represent the data.

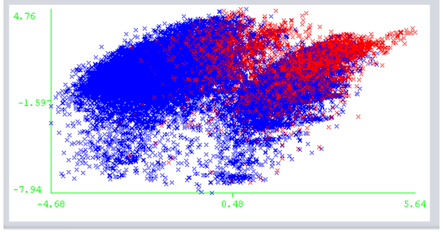


.



The graph above shows the distribution of the eigen values of the principal components of the adult dataset. We can see the eigen values go to almost 0 around component 97 so very little is gained from including those principal components. However, with this set I chose to reduce to components which cover 0.5 variance of the data which consists of the first 38 principal components. I chose to do this as to gain a more concise and reduced version of the data rather than data with 96 features.

The plot below is of the first two principal components, and the colors represent the two classes. We can see this actually shows a decent separation between the two classes with the red being mostly the upper right corner, and the blue making up the majority of the rest of the instances.



| Dataset | Clustering algorithm | # of Attributes | Incorrectly Classified | SSE/Log Likelihood |
|---|---|---|---|---|
| Car Eval | K means | 15 | 68.06% | 2176.08 |
| Car Eval | EM | 15 | 68.75% | -15.141 |
| Adult | K means | 38 | 27.78% | 6599.45 |
| Adult | EM | 38 | 42.78% | -51.744 |

The above table show the results of running the clustering algorithms on the PCA reduced data, using the same number of clusters as above. We can see there is little difference between the incorrectly classified after data reduction and the incorrectly classified on the original data, implying the clusters are very similar to the original clusters.

| Adult Class (K means, EM) | Cluster1 | Cluster2 |
|---|---|---|
| <=50K | 53%, 74% | 93%, 80% |
| >50K | 47%, 26% | 7%, 20% |

The table above shows the class breakdown of each cluster in the adult dataset. We can see that the clusters do not line up particularly well with the classes, although cluster 2 is majority class <=50K, cluster 1 is relatively evenly split. These are even worse clusters for the expectation maximization (the second cluster is no longer 60% >50K class). The K means clusters are quite close to the original clusters though only differing slightly in the first cluster (53/47 split instead of 60/40).

| Car Eval Class (K means, EM) | Cluster1 | Cluster2 | Cluster3 | Cluster4 |
|---|---|---|---|---|
| Unacc | 83%, 75% | 68%, 62% | 63%, 60% | 63%, 83% |
| Acc | 17%, 25% | 23%, 27% | 25%, 21% | 25%, 17% |
| Good | | 5%, 5% | 6%, 11% | 6%, 0% |
| Vgood | | 4%, 6% | 6%, 8% | 6%, 0% |

Again, these clusters line up very poorly with the classes and mostly just seem to represent relatively a random sample of the data. This is very similar to the original clusters though.

**Random Projections:**

For random projections, I performed the reduction three times with different random seeds (40, 42, 44), in order to account for the random nature of the projection. I also chose to reduce to the same number of features as I did in the principal component reduction so as to compare the methods of reduction better. The projections were performed using the Sparse1 distribution.

| Dataset | Clustering Algorithm | # of Attributes | Seed = 40, incorrectly classified | 40, SSE/Log likelihood | Seed = 42, incorrectly classified | 42, SSE/Log Likelihood | Seed = 44, incorrectly classified | 44, SSE/Log Likelihood |
|---|---|---|---|---|---|---|---|---|
| Car Eval | K means | 15 | 70.42% | 861.34 | 67.65% | 906.47 | 63.37% | 836.60 |
| Car Eval | EM | 15 | 69.16% | -29.77 | 65.97% | -27.40 | 64.81% | -30.48 |
| Adult | K means | 38 | 36.11% | 10303 | 48.14% | 11063 | 34.20% | 10369 |
| Adult | EM | 38 | 22.02% | -288.8 | 23.998% | -310.4 | 23.998% | -296.7 |

The above table shows a decent amount of variance in the different random projections for the car evaluation dataset, with the Car Evaluation k means inaccuracy ranging from 63%-70%, and the EM inaccuracy ranging from 64%-69%. For the adult dataset, there is a lot less variance in the EM inaccuracy with values in the range of 22%-24%; however, the K means inaccuracy has a lot more variance with values ranging from 34%-48%.

| Adult Class (K means, EM) | Cluster1 | Cluster2 |
|---|---|---|
| <=50K | 62%, 49% | 87%, 77% |
| >50K | 38%, 51% | 13%, 23% |

The table above shows the class breakdown of each cluster of the adult dataset, with random seed 44, as it had the lowest incorrectly classified percentage. It is again pretty clear that these do not line up well with the classes, cluster 2 is majority class <=50K and cluster 1 is again relatively evenly split between the class. The low incorrectly classified percent only occurs because of the imbalance of the classes. Although, unlike the PCA with EM cluster 2 is mostly composed of class >50K, which is why the incorrectly classified percentage is much lower (22% as opposed to the 42% of PCA).

| Car Eval Class (K means, EM) | Cluster1 | Cluster2 | Cluster3 | Cluster4 |
|---|---|---|---|---|
| Unacc | 78%, 56% | 52%, 58% | 52%, 91% | 96.6%, 76% |

|  |  |  |  |  |
|---|---|---|---|---|
| Acc | 16%, 30% | 31%, 34% | 40%, 8% | 3.2%, 18% |
| Good | 3%, 6% | 8%, 6% | 5%, 0.2% | 0.2%, 3% |
| Vgood | 3%, 8% | 9%, 2% | 3%, 0.8% | 0%, 3% |

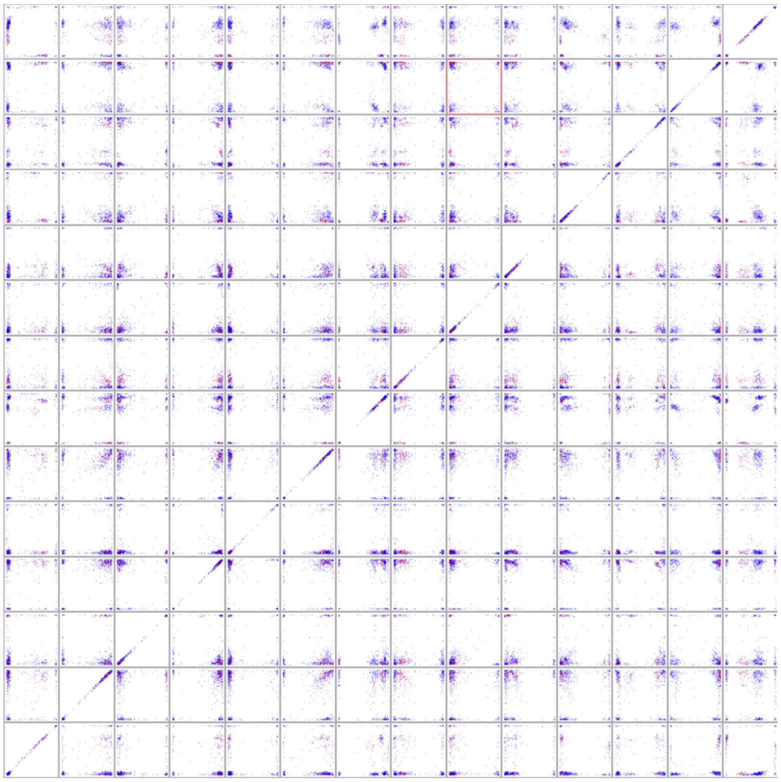
The above table shows the class breakdown of the clusters for the car evaluation dataset, with random seed 44, as it had the lowest inaccuracy. These clusters do not line up particularly well with the classes. The instances seem to be almost evenly spread between the clusters which clearly prevents the smaller classes from having the majority in any cluster when the data is imbalanced.

**MLPAutoencoder:**

For the MLPAutoencoder, I used the MLPAutoencoder Filter in Weka in the MultiLayerPerceptron Package. I decided to reduce to the same number of features as the PCA reduction in order to accurately compare the reduction algorithms.

| Dataset | Clustering Algorithm | # of Attributes | Incorrectly Classified | SSE/Log Likelihood |
|---|---|---|---|---|
| Car Eval | K means | 15 | 49.36% | 4938 |
| Car Eval | EM | 15 | 46.64% | -0.523 |
| Adult | K means | 38 | 32.81% | 145621 |
| Adult | EM | 38 | 28.26% | -7.821 |

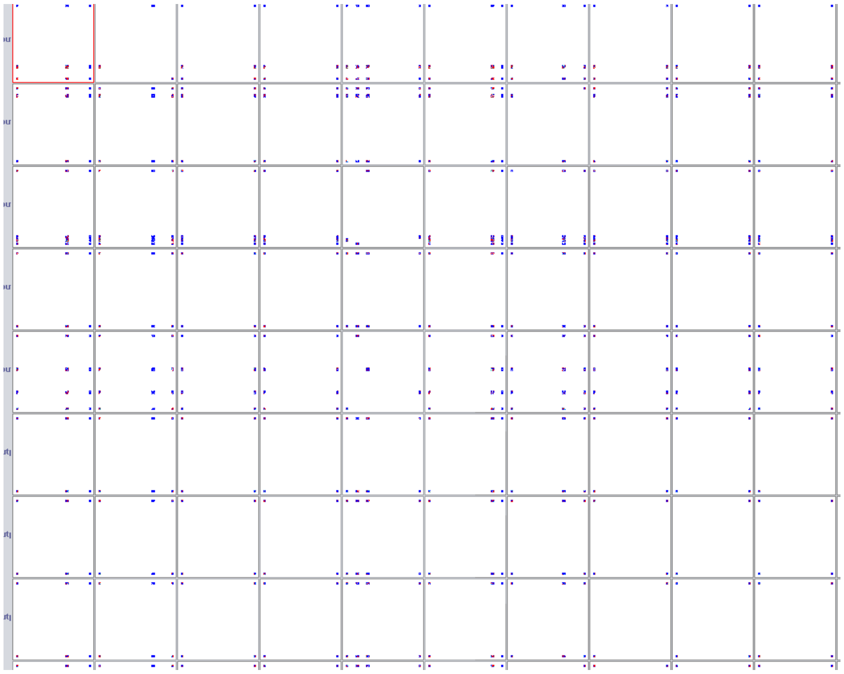
The plot below shows the distribution of the data after the reduction, the colors represent the different classes. The data is not particularly easily separable and is quite sparse, likely higher dimensional data would improve upon this.



Below is the class breakdown of the clusters for the adult data set after the reduction, like PCA these do a poor job of separating the classes, besides that cluster 2 for K means is overwhelming class <=50K and cluster 1 for EM is overwhelming class <=50K.

| Adult Class (K means, EM) | Cluster1 | Cluster2 |
|---|---|---|
| <=50K | 61%, 89% | 86%, 55% |
| >50K | 39%, 11% | 14%, 45% |

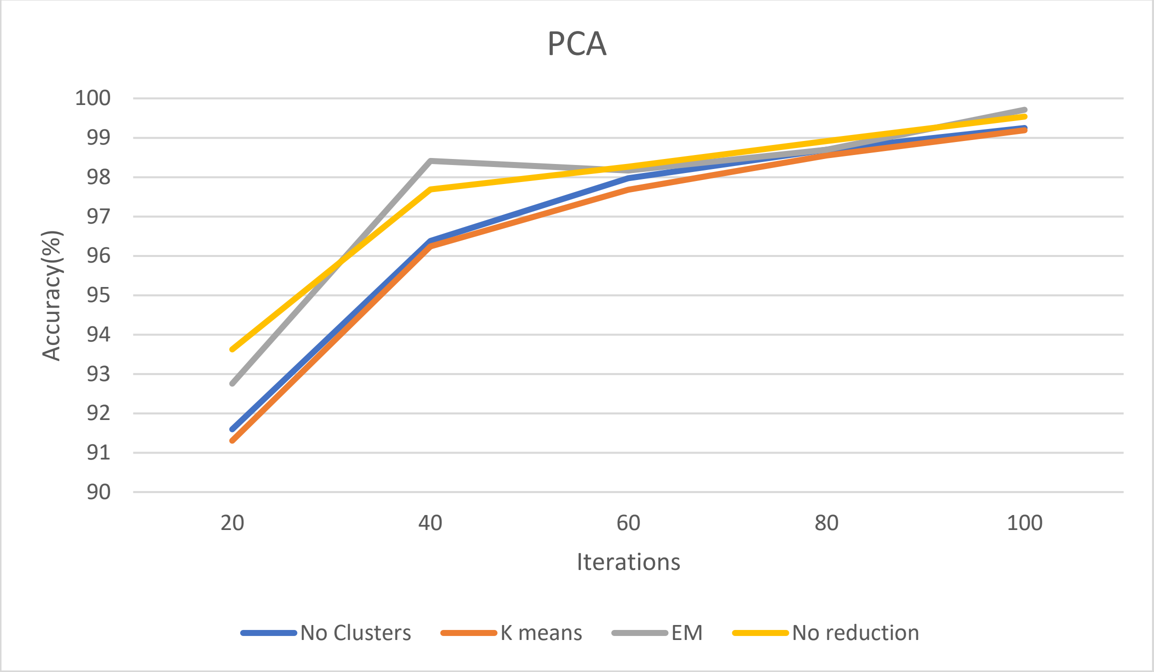
The plot below shows the distribution of the carevaluation dataset after the reduction, as we can see the data is quite sparse and easily separable. This is expected since with PCA we found a good representation of the data in 15 attributes covering the majority of the variance in the data.

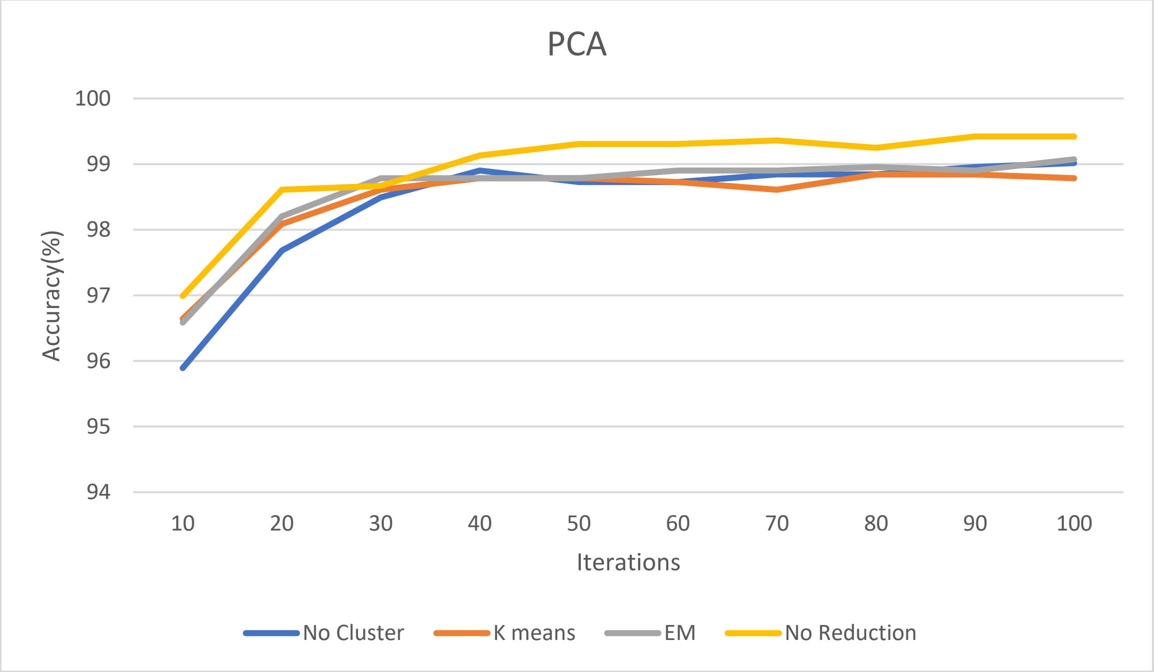


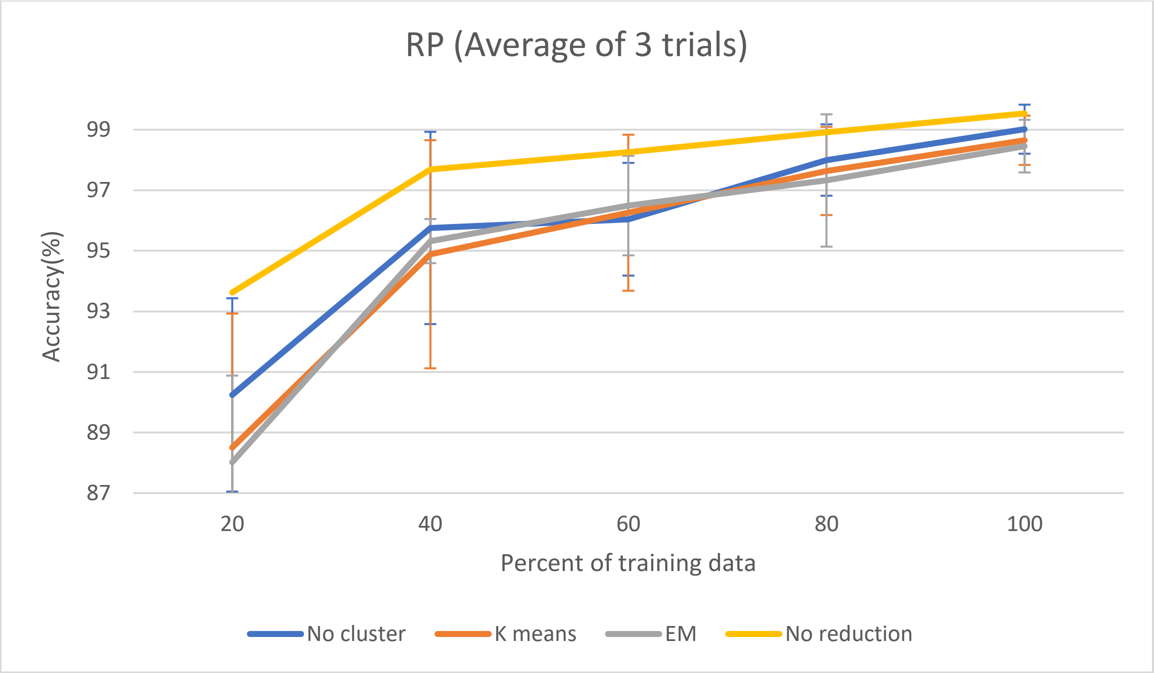
Below is the class breakdown of the clusters after the autoencoder reduction. As with all the other dimensionality reduction methods, these cluster do not seem to line up with the classes, which is why the incorrectly classified percentage is 49% for K means and 46% for EM.

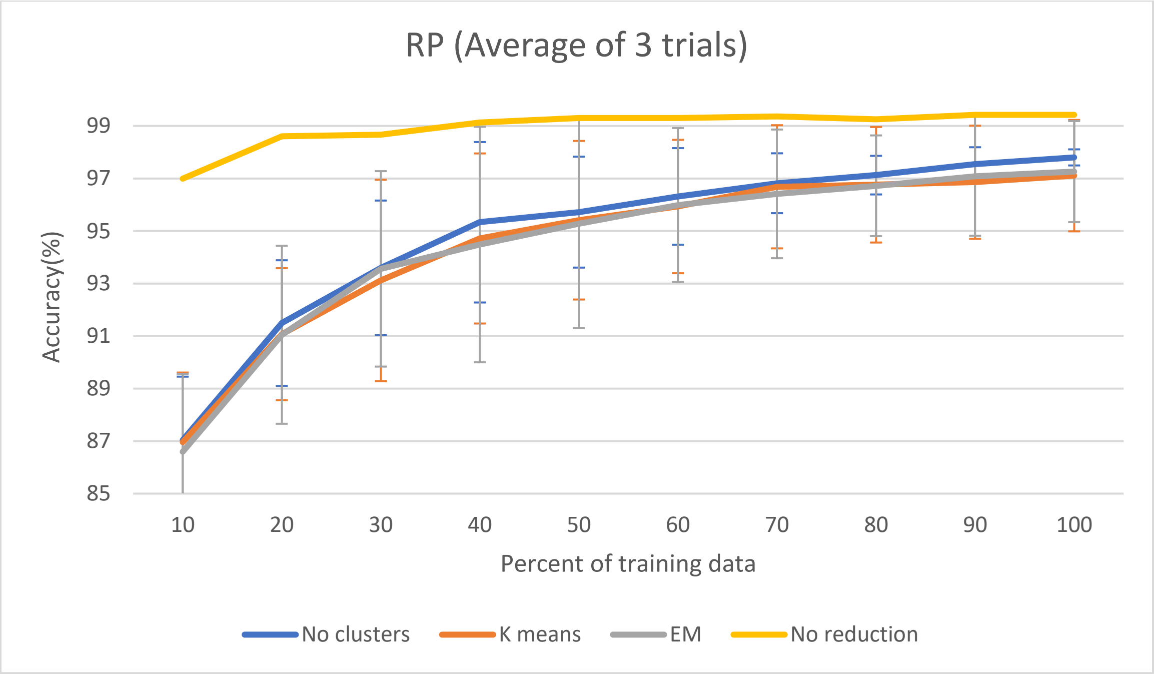
| Car Eval Class (K means, EM) | Cluster1 | Cluster2 | Cluster3 | Cluster4 |
|---|---|---|---|---|
| unacc | 64%, 83% | 69%, 62% | 68%, 62% | 74%, 73% |
| acc | 25%, 17% | 23%, 27% | 23%, 21% | 20%, 24% |
| good | 5% | 4%, 5% | 4%, 11% | 3%, 0% |
| vgood | 6% | 4%, 6% | 5%, 6% | 3%, 3% |

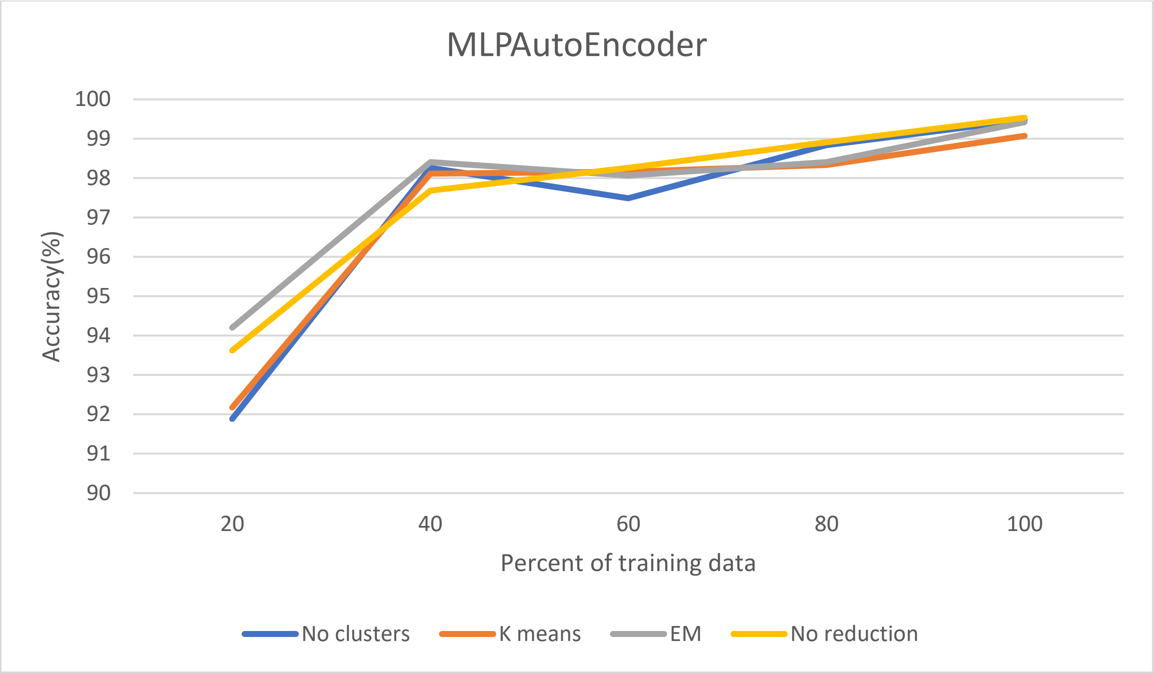
## Neural Network Training:

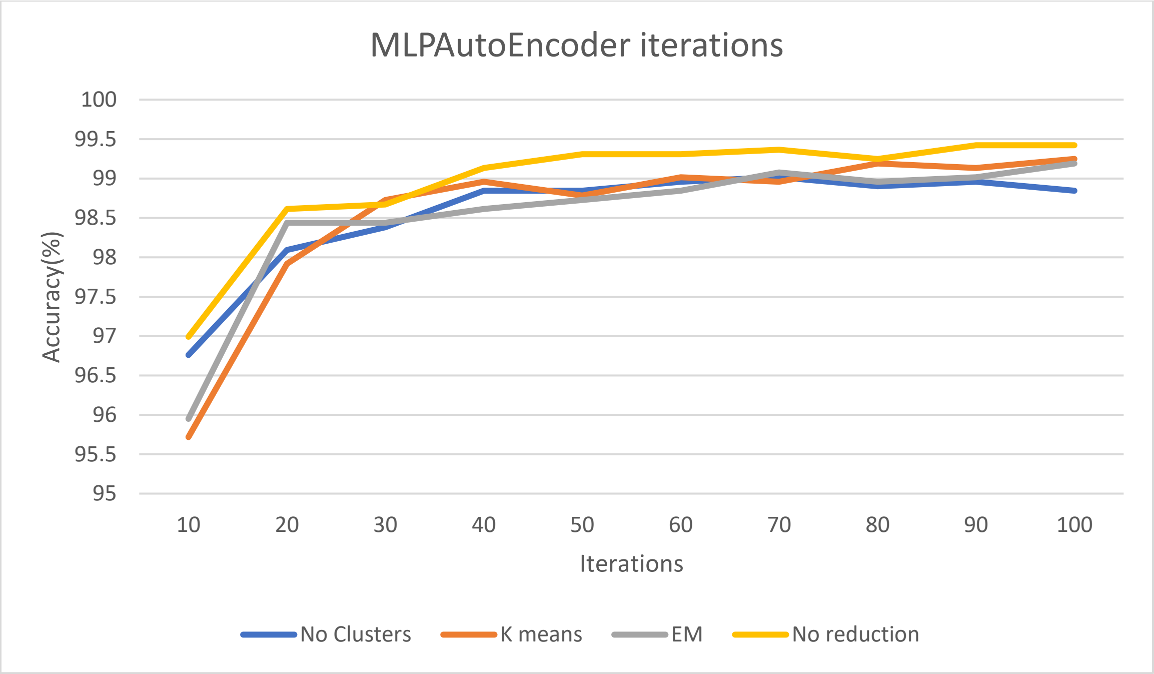





RP (Average of 3 trials)
Accuracy(%)
87
89
91
93
95
97
99
20
40
60
80
100
Percent of training data
No cluster
K means
EM
No reduction


RP (Average of 3 trials)
Accuracy(%)
85
87
89
91
93
95
97
99
10
20
30
40
50
60
70
80
90
100
Percent of training data
No clusters
K means
EM
No reduction





Each Neural Network was trained with a learning rate of 0.3 and a momentum of 0.2. The learning curves that vary the amount of training data were run for 500 epochs, and the curves that vary the number of iterations were run on 100% of the training data. The reductions and clustering were performed with parameters listed in the above sections. The graphs above show 10-fold cross validation accuracy.

We see, in general, there was little loss of accuracy from the reduction of the Car Evaluation dataset, so the reduced datasets do a good job of describing the data in a lower dimension. The addition of the clusters seemed to have very little effect on the accuracy, likely because it was so high anyway, and the clusters also don't particularly correlate well with the classes as stated in the above sections. So, the addition of these cluster attributes is unlikely to help since the say almost nothing about which class the instance is in. We do see, especially with the random projection reduction, that the reduced data seems to have lower accuracies for smaller amounts of training data and lower accuracies for fewer iterations.

We do see as expected the random projection perform worse than the principal component analysis with especially at lower numbers of iterations for instance at 100 iteration RP had an average accuracy of 97.8% compared to the 99% accuracy of PCA. The autoencoding and PCA perform quite similarly with both reach accuracy of over 99% with just 100 iterations on the full training set.

As the point of dimensionality reduction, is to see if our data can be represented in a more concise way (lower dimension), we can see the autoencoding and PCA reduction do exactly that hardly sacrifice any accuracy, as expected when the PCA cover 95% of the variance. This allows for shorter training times as there are less features. Also, we can see even the random projection on average did quite well 99% accuracy after 500 iterations on the full testing set. However, since these projections are suboptimal it is no surprise it did worse than PCA and Autoencoding.