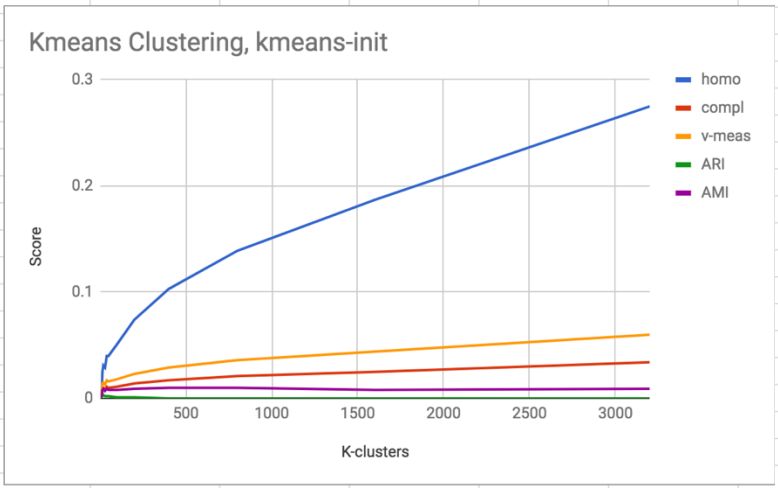
CS4641 Project 3 Analysis Report

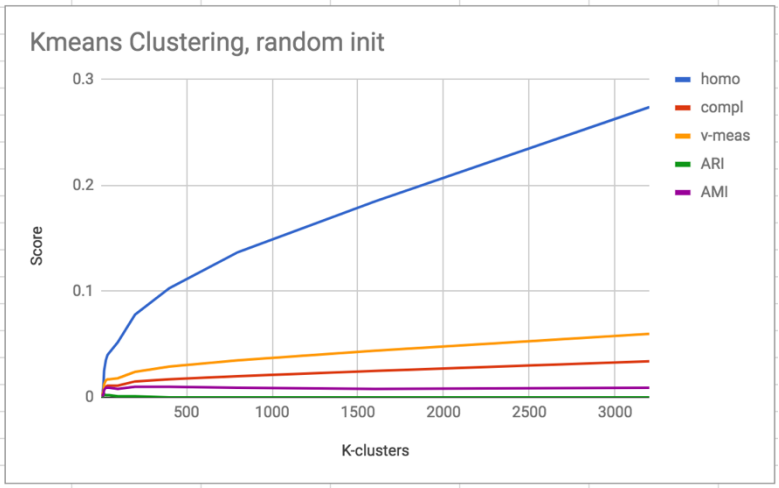
Kristian Suhartono / 903392481

Datasets:

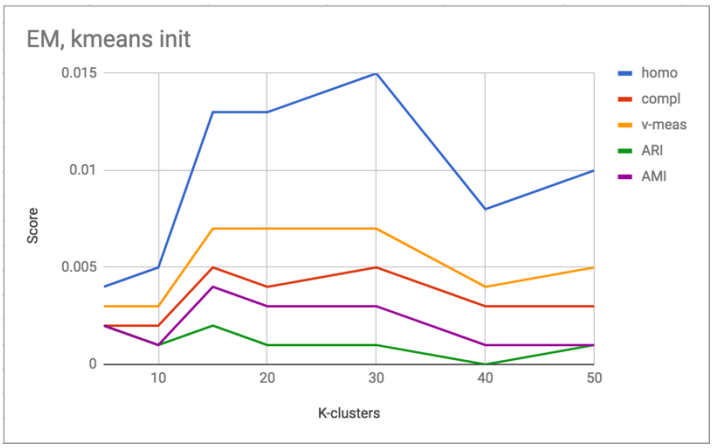
I reused both my datasets from project 1. The poker dataset, contains information about 5 poker cards, and labels for what kind of hand it is. The point is to classify what kind of hand is the 5 cards. The Dota2 dataset contains information about heroes picked and some extra information with labels for which team won. The point is to predict which team will win the game.

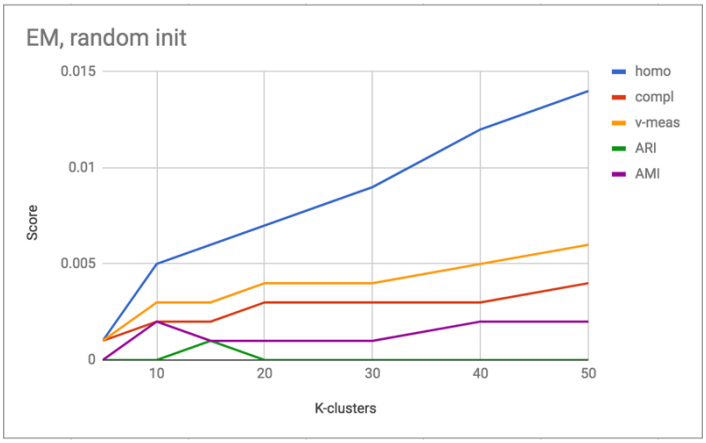
Poker:



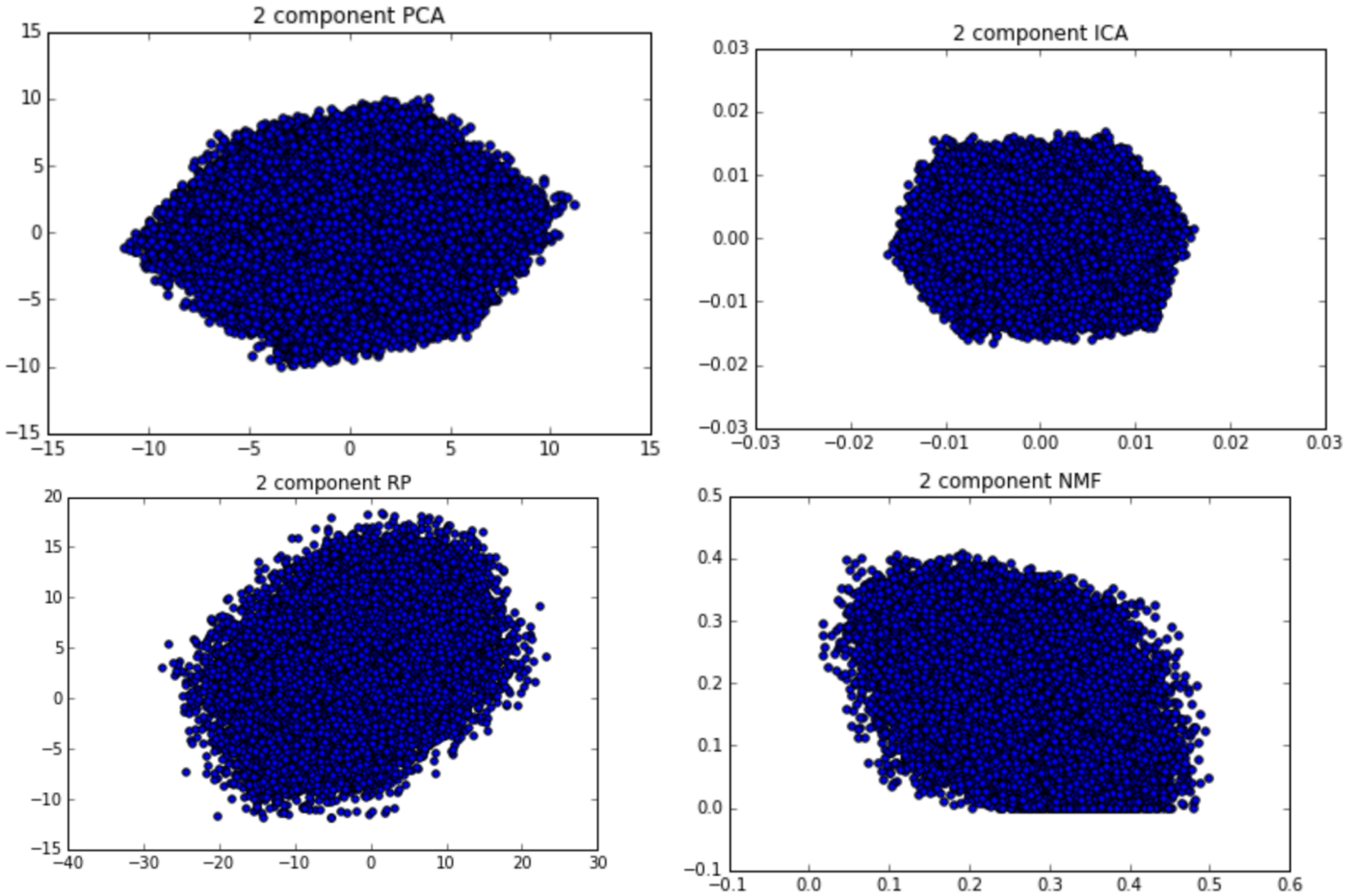


The above are metrics on the performance of KMeans clustering with a standard kmeans approach initialization and a random init. As can be seen from the results there is no difference either way in the initialization, meaning that the dataset is not affected too much by a randomized initialization. This is maybe due to the fact of the underlying distribution of the data. The metrics show a very low score in similarity metrics like ARI , AMI, and V-measure. This implies that clustering is not very suitable for the dataset, the data didn't cluster well with the Euclidean distance metric. The homogeneity results increases as K increases which is to be expected. My final K value is chosen based on the best ARI and AMI. Results might still improve with more K clusters, but it takes too long as the amount of clusters increase, and it only improves homo-score by a lot.

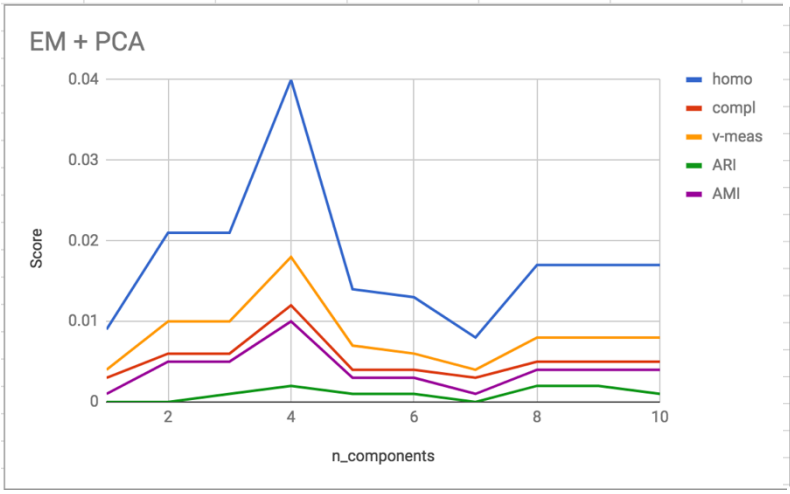


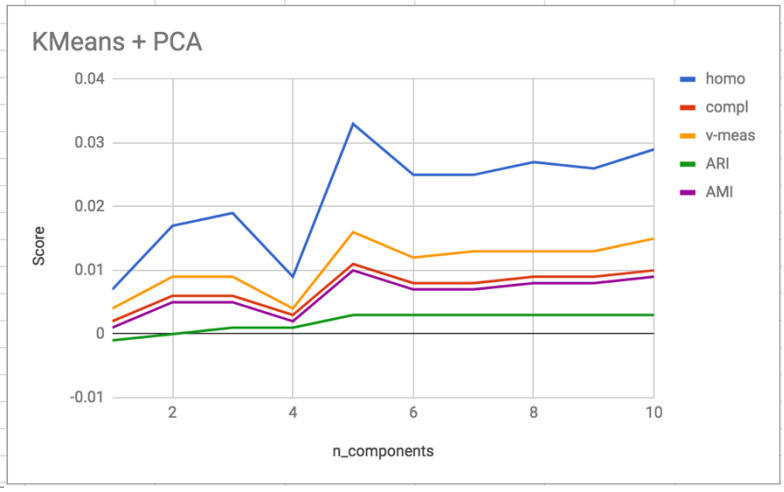


EM has a better performance with a smaller number of clusters than KMeans, this is probably due to parts of the dataset having a smaller number of examples, thus EM was able to predict the results better. However, again the results are very low which again enforces our belief that the dataset is not very clusterable.

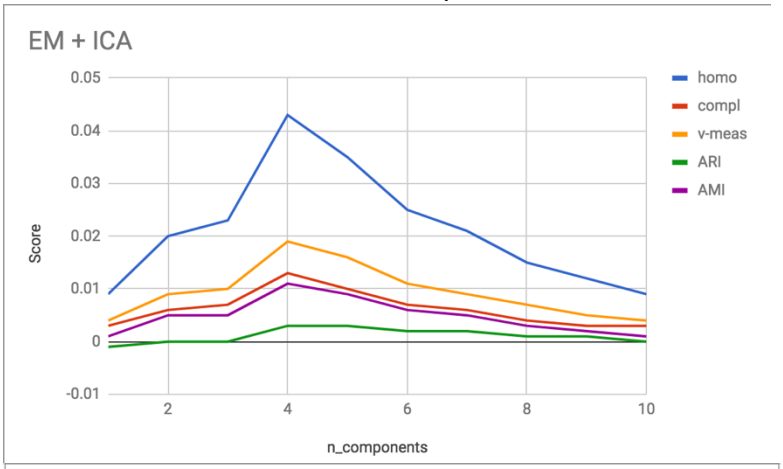


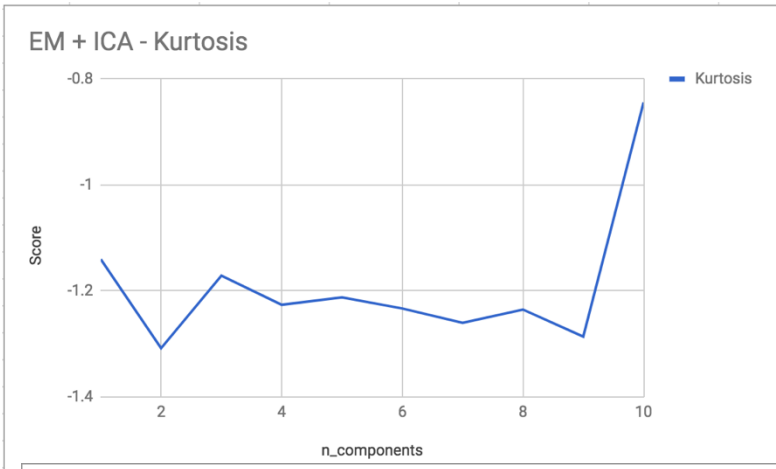
Above are scatterplots of the data with 4 different algorithms. All are compressed to 2 components, such that the results can be visualized here. PCA had clumped up results, however the results seem to be middle heavy and it sharpens as it gets further away, making it similar to the uniform distribution. This can be caused by the fact that there are a lot of correlations between the features. ICA is very squished, everything is basically spread with very small differences, this can be caused by the fact that separately, the features of the dataset doesn't really mean anything, causing ICA to perform poorly. RP was a larger spread, however it doesn't look like the results would show any interesting results. Probably due to the fact that projecting to a lower dimension would only help the computation here as RP attempts to preserve distances. The last algorithm I chose is NMF, non-negative matrix factorization, again it is very clumped up together, suggesting NMF is also unable to find interesting results from the dataset. I would guess that this is due to NMF doing dimensionality reduction and clustering at the same time. Since the dataset isn't very clusterable it performed poorly here.

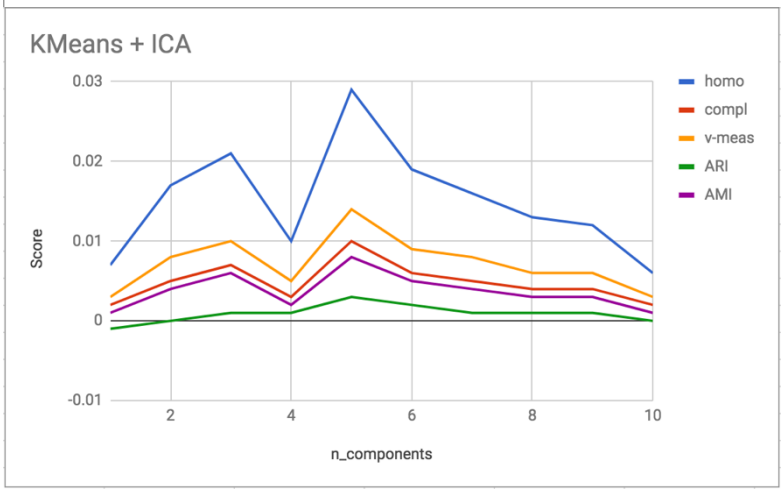


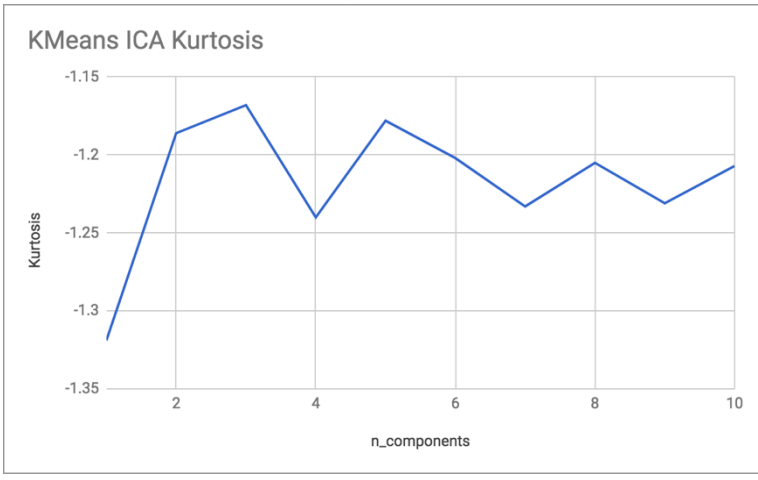


Clustering on data where PCA is applied seems to have increased results. AMI values seem to be better than what it was before, ARI is still poor though. This would again be attributable to the fact that PCA works really well theoretically with the dataset, however the dataset is just not clusterable. Speed wise, it is much faster to do the clustering with less amounts of features, which is expected as there is a smaller number of operations to do.

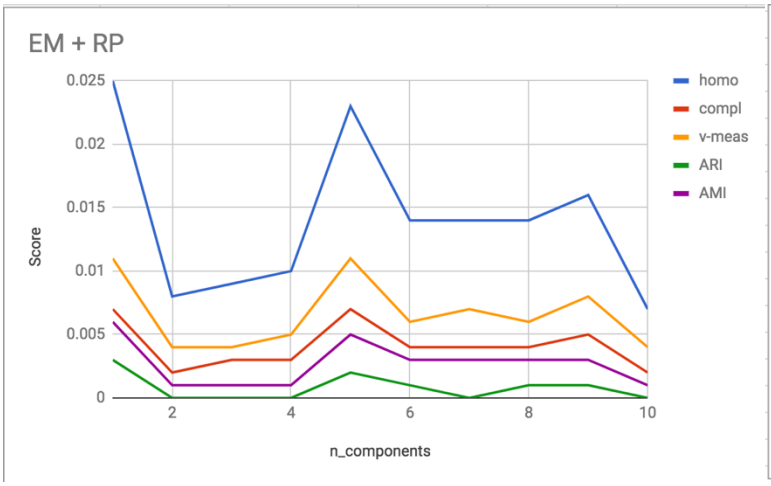


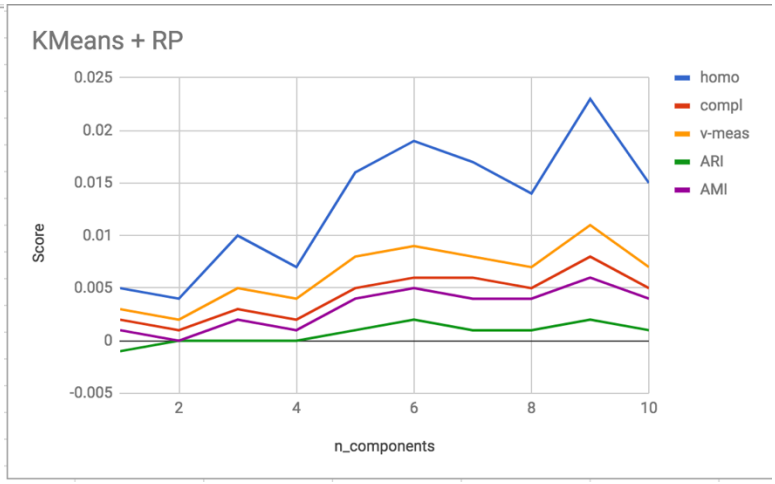




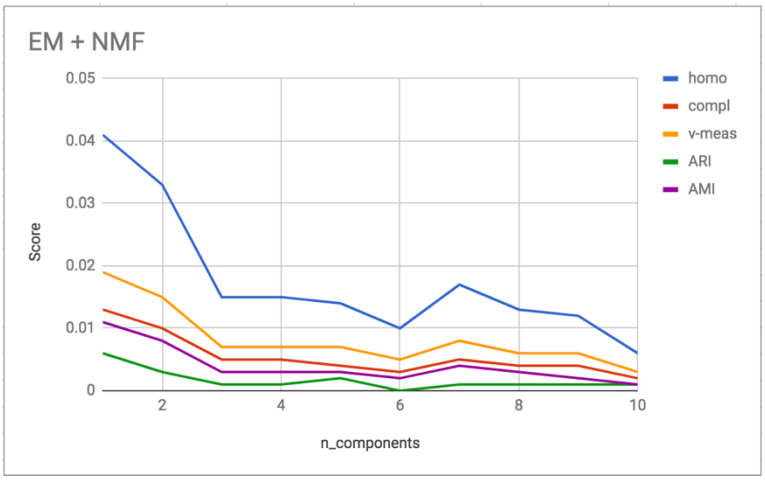


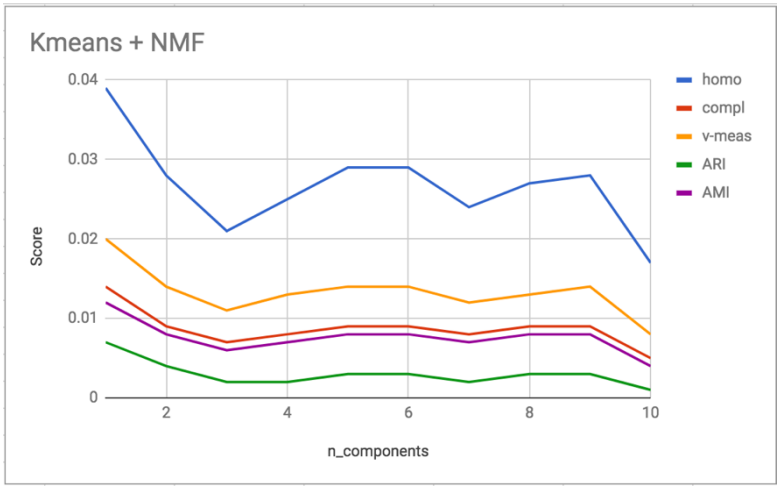
For ICA, the results are similar to PCA, however the average results are smaller than PCA, this shows the expected outcome that PCA is better than ICA due to the underlying correlation with the data. From the kurtosis it can be seen that most of the distributions are more uniform like. Which is supported by the graphs from before.



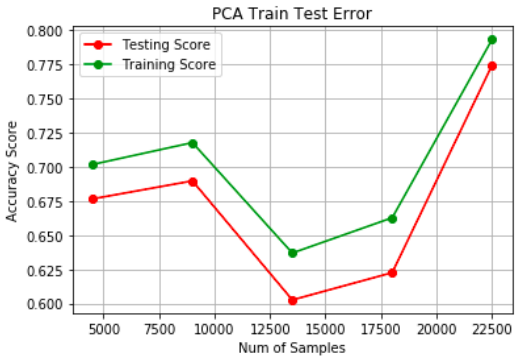


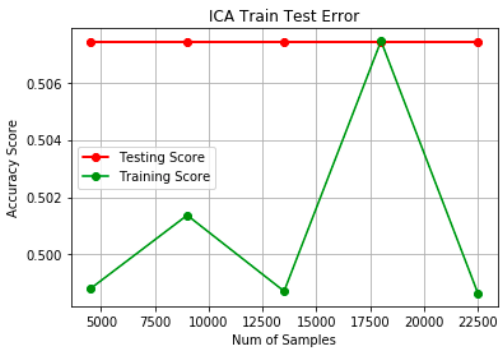
RP had an on average lower score compared to previous dimensionality reduction algorithms, but still has a comparable performance. This is probably due to the fact that the dataset just has a small number of features, making RP not so suitable for this.

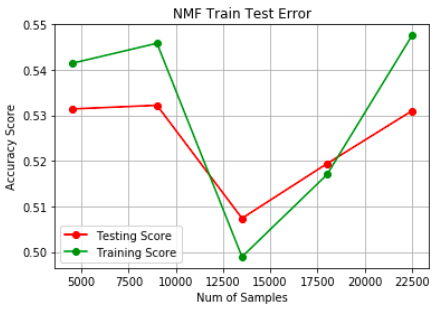


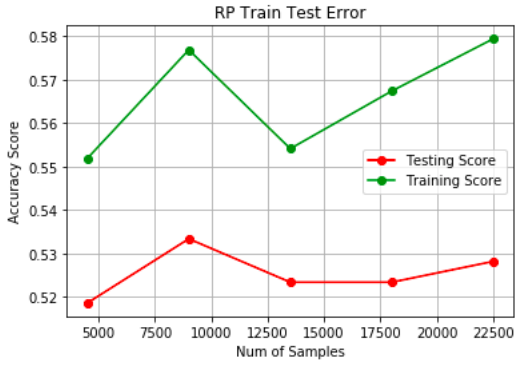


NMF has a much better comparison with a single component compared to all the other DRAs. I would attribute this to the matrix operations in the algorithm that is able to squeeze together all the info to one good attribute that is able to determine stuff better than the others, it also helps the clustering performance too.

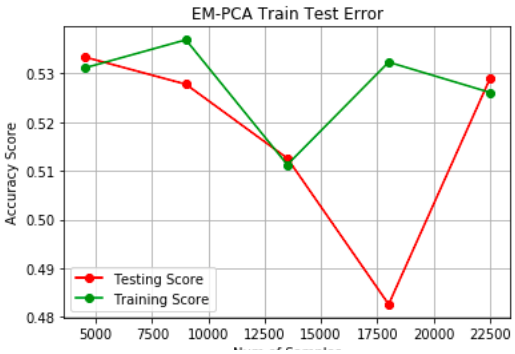


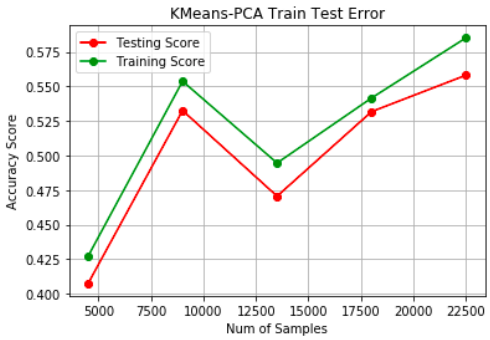


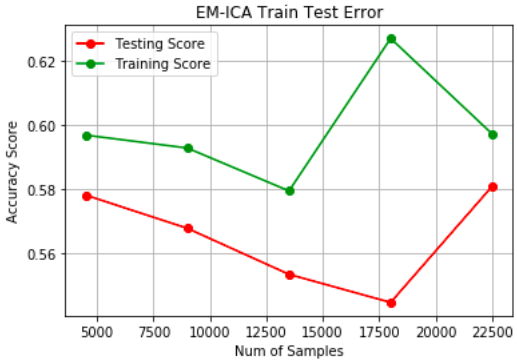


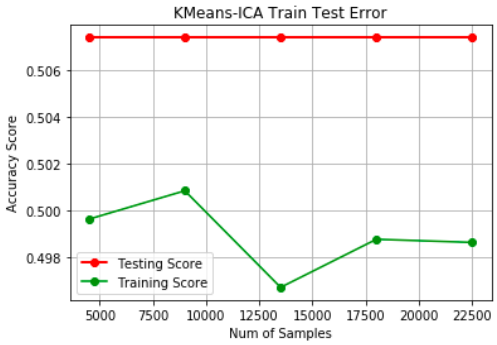


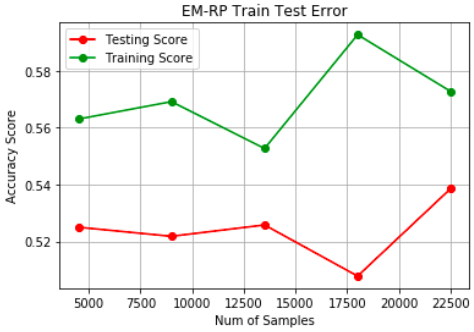
Shown in the graphs are the train test accuracies for the neural net, as a comparison, from project 1, the accuracy was highest at 74%. Whereas here, we can see that PCA best helps the neural net find some correlations that is clearly there in a poker hand. The other DRAs performed poorly, which can be attributed to the fact that they're not compatible with the dataset.

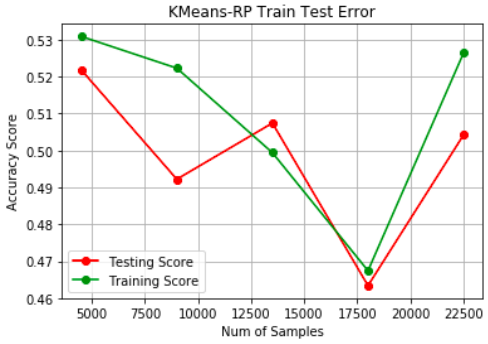


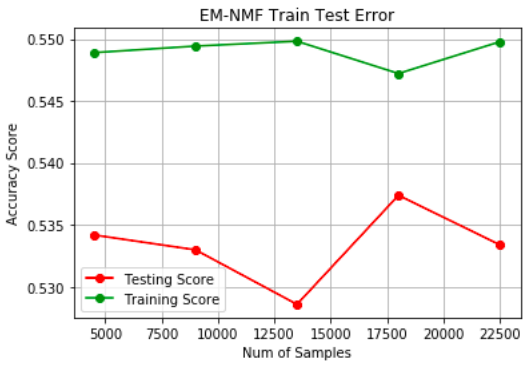


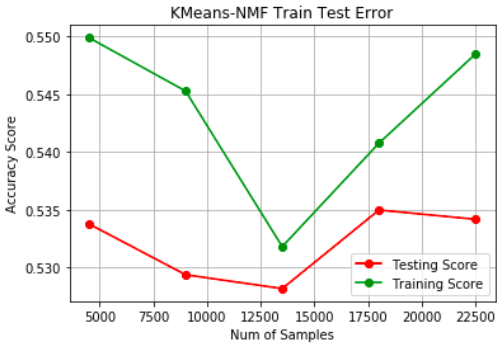






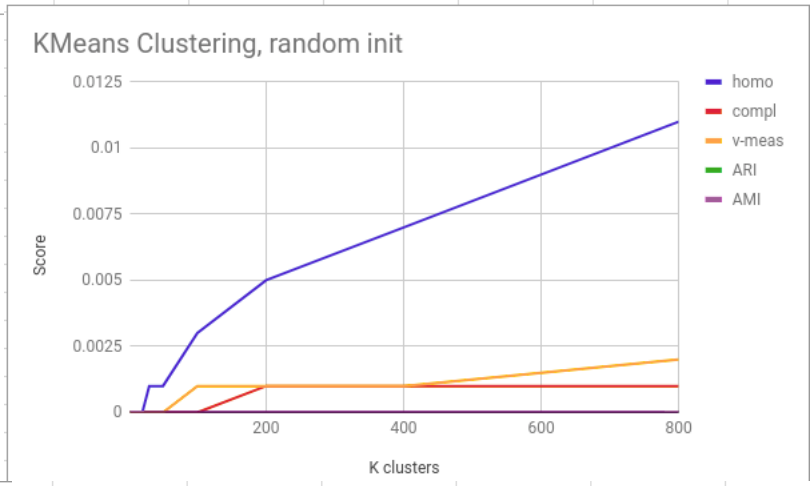


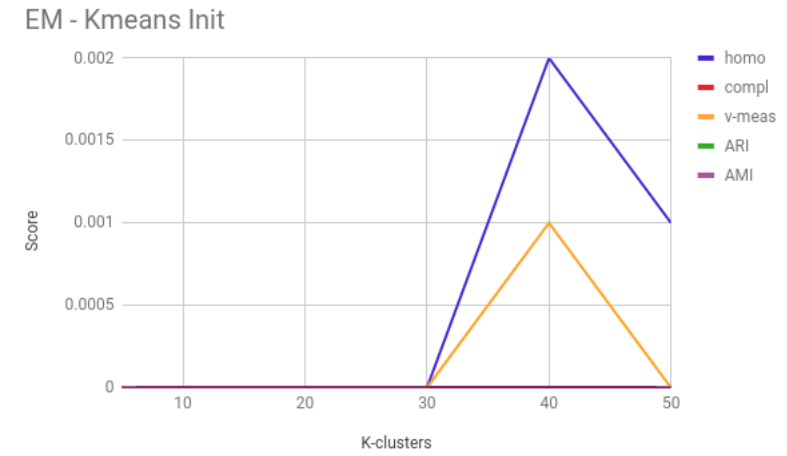


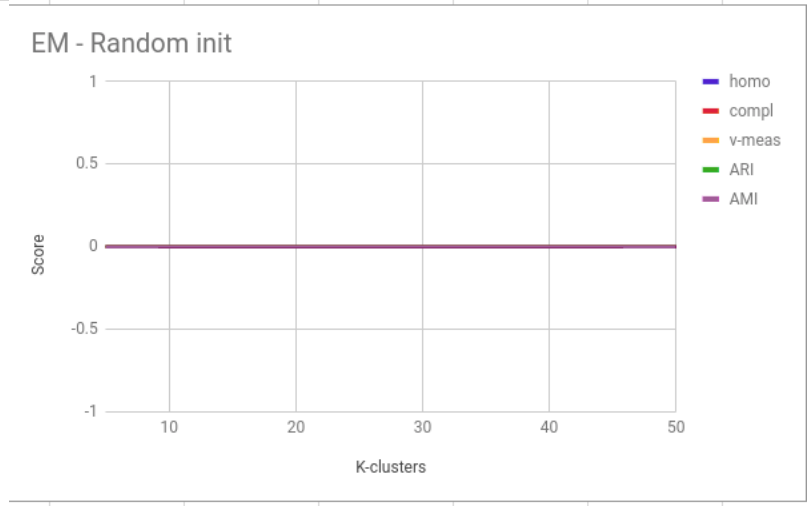


All the graphs describe the performance of the neural net with a dimensionality reduced dataset, which then got clustered on. Unsurprisingly all of them performed worse, even the one with PCA. This is clearly because of the fact that the dataset isn't able to be clustered. The distances are just not descriptive enough for this. A possible way to improve on this is to change the distance metric to something more suitable to describing the dataset.

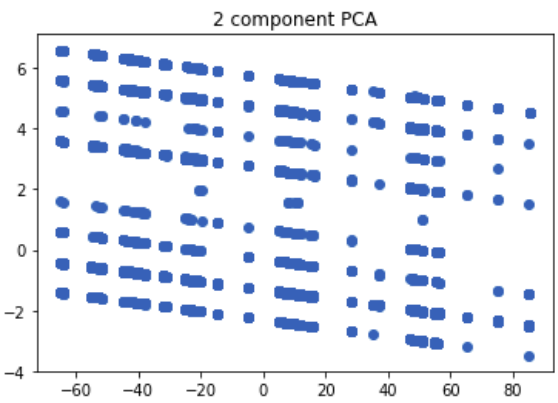
Dota 2:

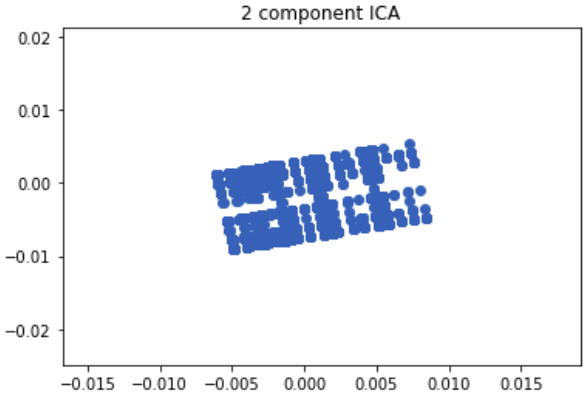


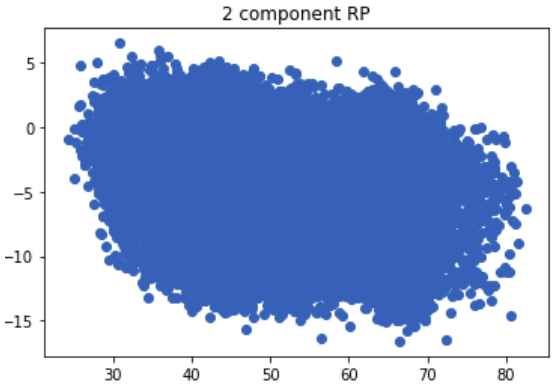


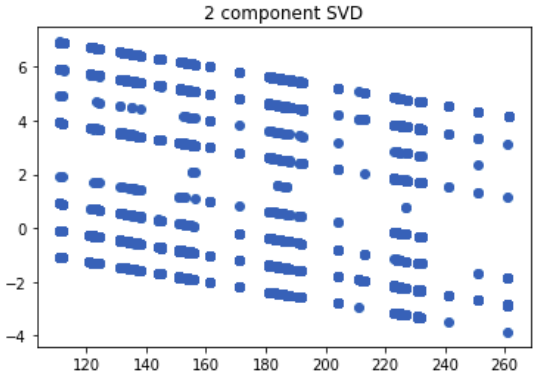


For the dota2 dataset, it would seem that clustering just simply doesn't work. The data has a lot of features, thus a too large number of clusters takes a lot of resources to compute (space wise, any number of clusters larger than 1600 seems to crash computers, my laptop ran out of RAM while trying to compute 3200 clusters). And the results are all 0 on most metrics except homogeneity. Even when there is an increase, it is just a small increase. The random start also clearly affects EM, as can be seen it has an all 0 result with EM, which is just bad. Therefore clustering seems to just doesn't work for the dataset. Maybe with a different distance metric it might work, but there isn't a clear definable distance function unlike for Poker. There might be an increase with more K, but my computer simply can't compute it.

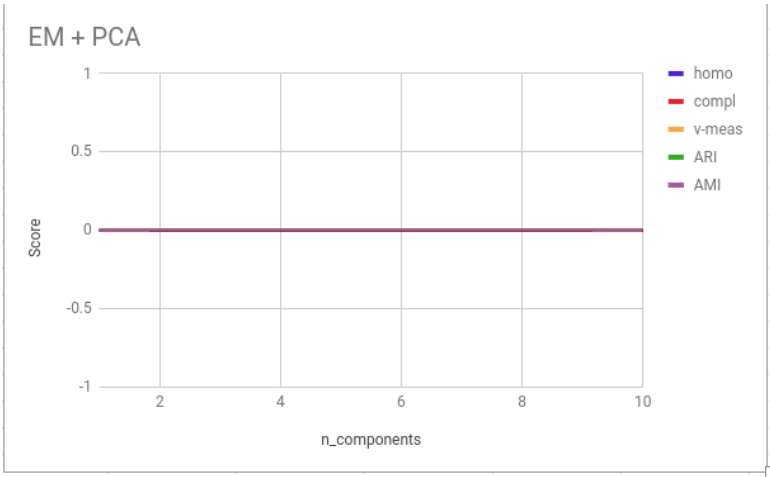


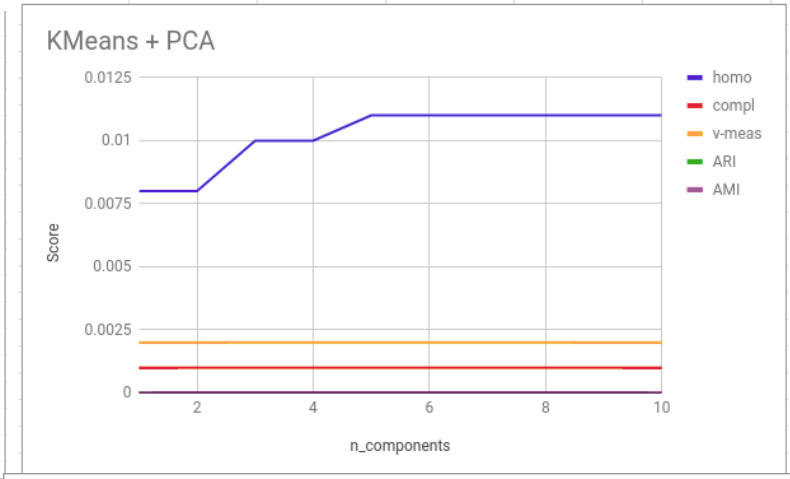


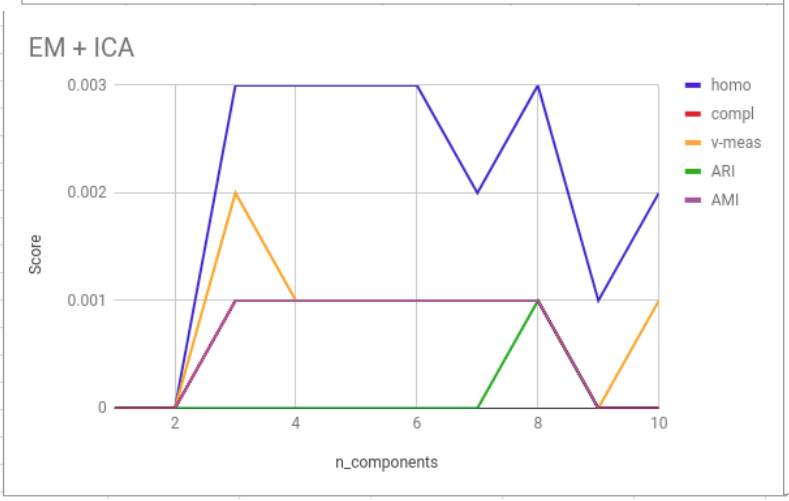


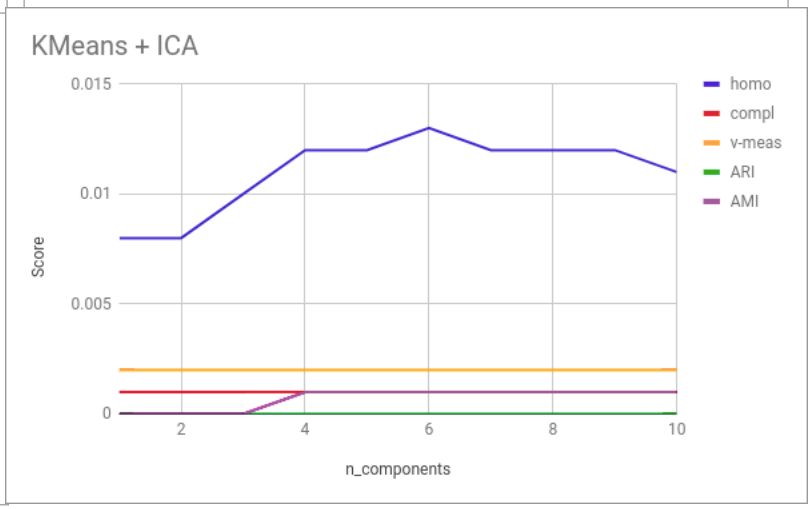


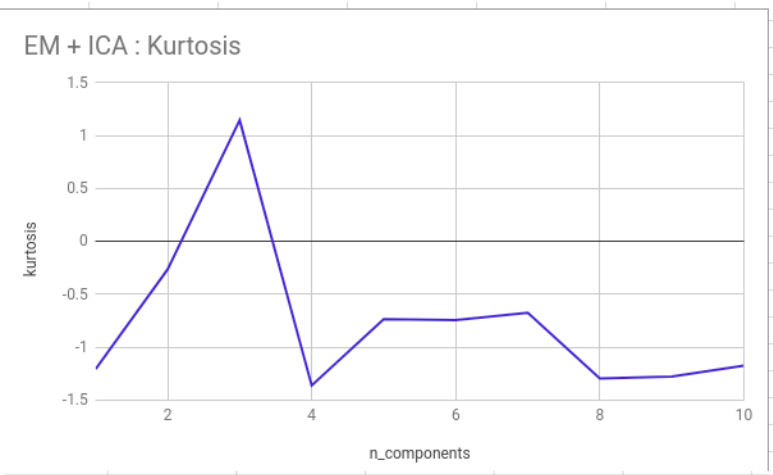
The 4th algorithm I chose here is TruncatedSVD, I chose it simply because of the fact that I was wondering if there would be any difference with PCA (the documentation mentioned a difference in how they were implemented). The outputs of both PCA and SVD seem interesting, they're both spread out and seem to form mini clusters, I'm not entirely sure what causes this, perhaps the data contained some interesting patterns that was represented like this. ICA also had a similar mini cluster form, however it is much more compacted, minimal differences between elements, this suggests that there isn't much point in examining the local components. RP seems to be the more spread out one, I guess due to the fact that it was projecting to a lower dimension, this was the expected output.

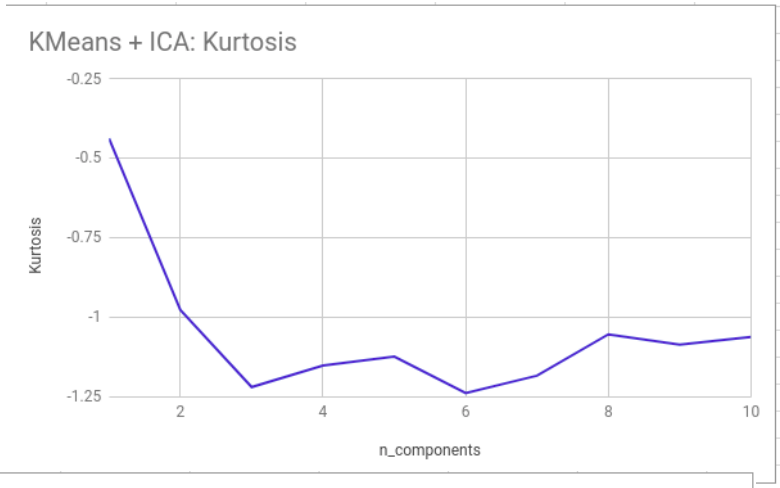


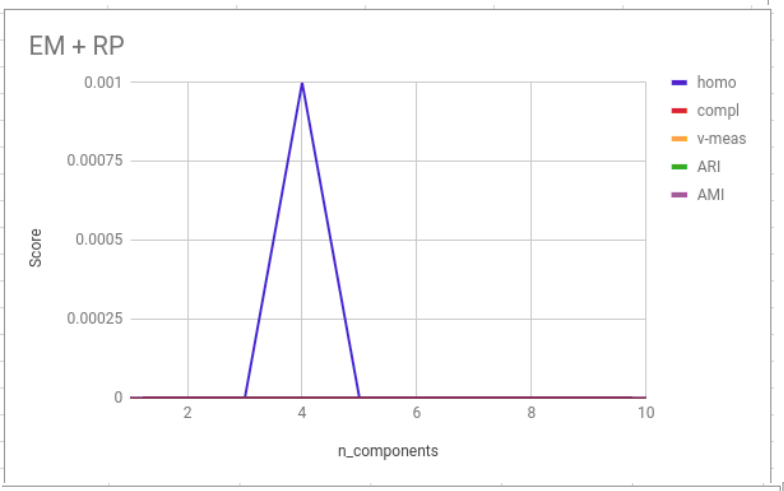


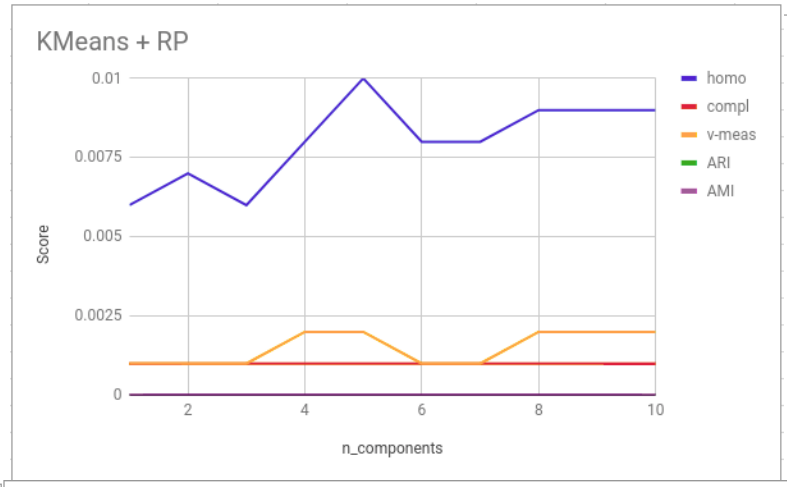


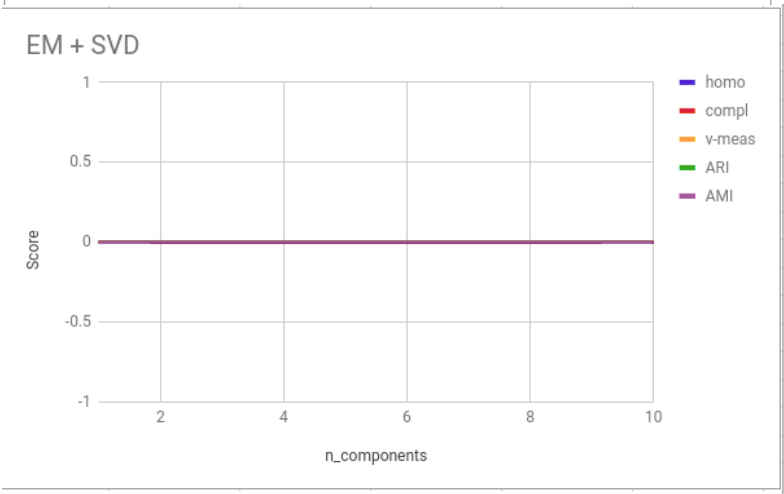


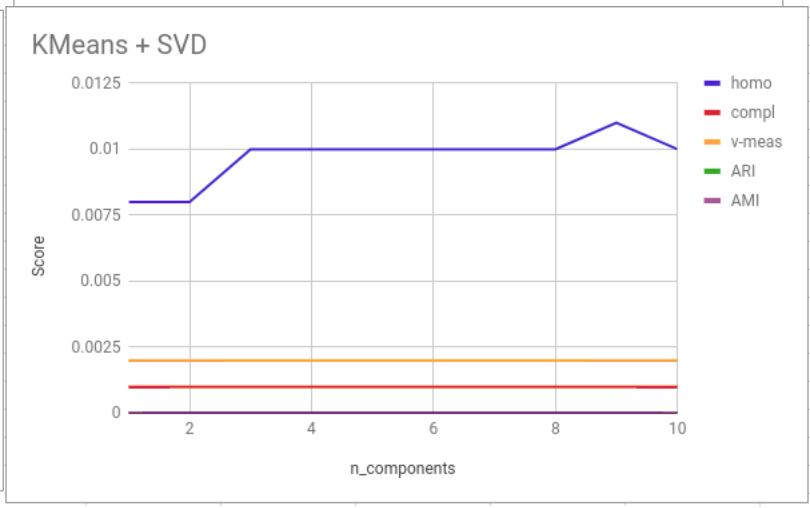




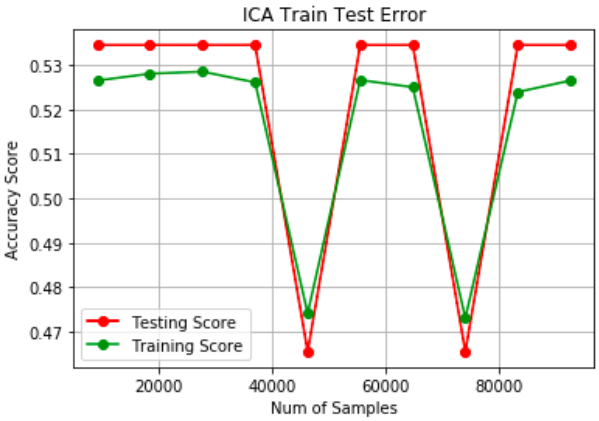


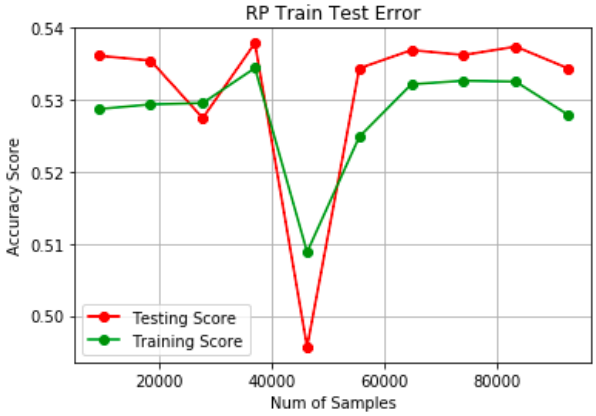


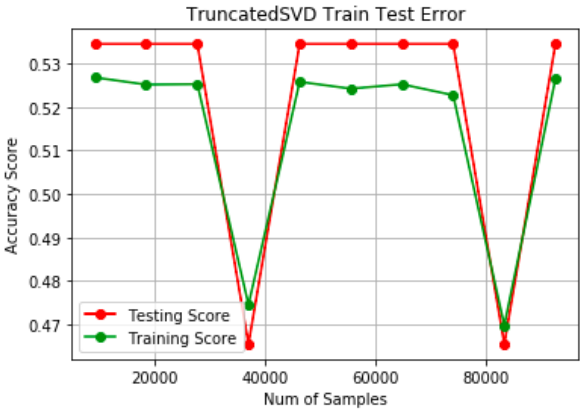




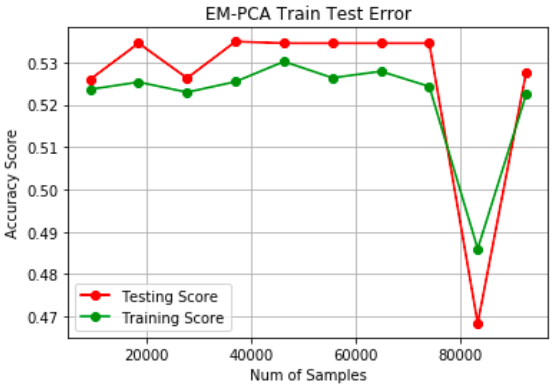
Expected results from the algorithms, most of them had no surprising results, all were just very near the 0 mark. Clearly caused by the data being not clusterable. The DRAs were unable to really help the results here.

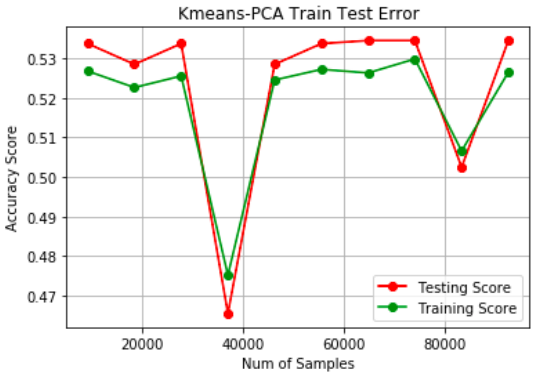


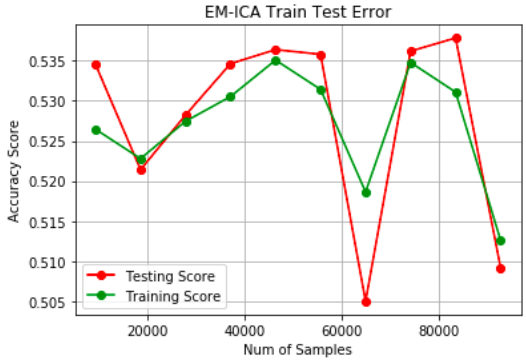


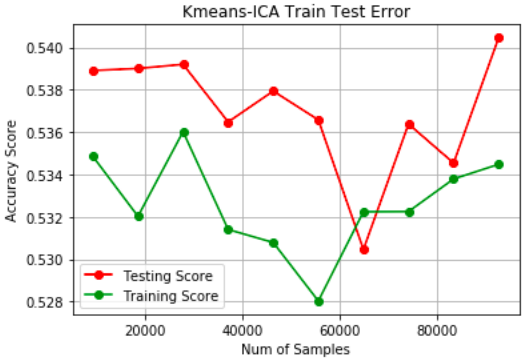


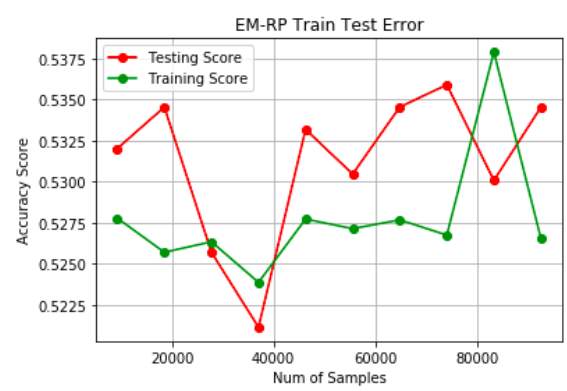
The DRAs all caused a decrease in the accuracy of the NeuralNet as compared to the results from Project 1 (60% accuracy). This would probably be because a too large reduction of features. With a larger number of features, these may be able to get a better accuracy. But currently DRAs to a small number of features worsen the accuracy.

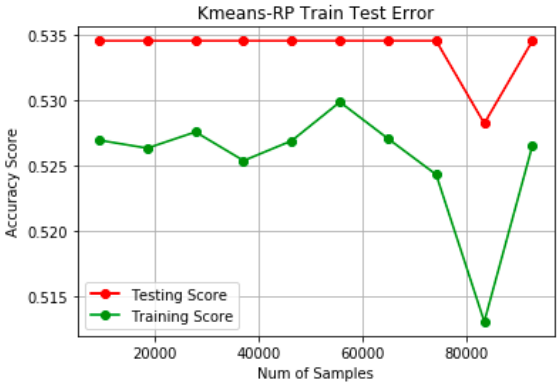


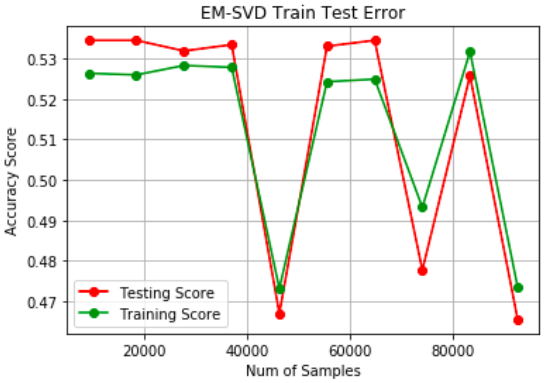


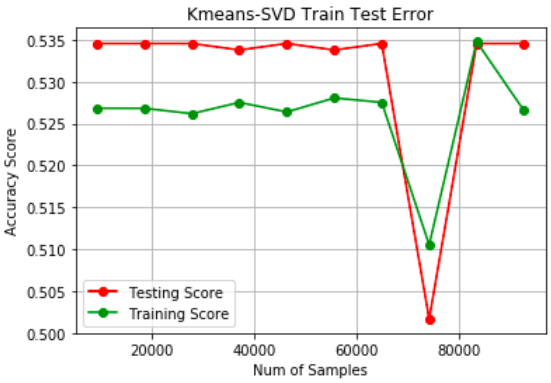












And again, unsurprisingly, there is different result with the results of doing clustering on the reduced data. Clustering doesn't play well with this data. DRAs may be able to work better with more n_components, which can serve as a point to improve on to help increase the results. But current results are worse than training NeuralNet with raw data.