CS7641 Project 3 – Unsupervised Learning

Tugsa Gongor (tgongor3@gatech.edu)

I. Introduction

Two clustering algorithms (K-Means and Expectation Maximization) are used with four dimensionality reduction algorithms (PCA, ICA, Randomized Projections, SVD) on two different datasets with differing properties. We then use neural networks on the resulting datasets. Analyses of these methods are discussed.

II. Introduction to Datasets

A. *Epilepsy*

The epileptic seizure recognition dataset consists of data collected from an electroencephalography (EEG) recording of 500 different individuals over 23 seconds. Learning to predict epilepsy from similar datasets as the given one could potentially lead to much earlier interventions and improve the quality of life of those suffering from such disorders.

This dataset has 178 features. The y column had five different values, only one of which indicated seizure activity. Subjects falling in the other four categories (2-5) did not have seizure activity. As such, in preprocessing the data, the category column y was changed to be binary such that category of 1 indicated having epileptic seizure and categories 2-5 were considered not having epileptic seizure.

The input vectors contained continuous integer values in both positive and negative realms, and the classification was made to be binary. We also used standard scaling to standardize the inputs to unit variance.

The metric for optimizing the hyperparameters for each model was the f1 score, which calculates the f1 score for both classes and takes their weighted average. This is best suited for datasets that contain majority negative classes.

B. *GAMMA*

The gamma telescope dataset consists of 19,020 instances of Monte Carlo generated observations meant to simulate high energy gamma particle readings in a Cherenkov gamma telescope.

The dataset contains 11 columns in total, 10 of which are the input features and the 11th being the category of the input vector. Some features include the major and minor axes of the elliptical image, angle of major axis, and distance to center of ellipse. All of the input features are numerical and continuous.

There are 12,332 instances of the positive gamma signal case and 6688 instances of the negative hadronic noise case. In comparison to the epilepsy dataset, the positive instances outnumber the negative in this case by a factor of close to 1.8.

The metric for optimizing the hyperparameters for each model was the AUC score since identifying a negative class as positive is worse than identifying a positive class as negative. For this reason, we chose not to use the standard accuracy or weighted f1 scores, though they will be mentioned throughout the analysis. We also used standard scaling to standardize the inputs for usability.

In comparison with the epilepsy data, the GAMMA dataset offers us an unbalanced label set in the opposite direction. GAMMA also has only 10 features versus the 178 of the epilepsy dataset, and GAMMA is more difficult to predict. These differences offer us interesting comparisons for the chosen algorithms.

III. K-Means

The K-Means algorithm attempts to segregate a given dataset into K clusters by minimizing the sum of squared distances within a cluster. The centers of the clusters are recalculated according to how the samples were clustered. This is repeated until convergence.

A. *Epilepsy*

We initially tested the number of clusters between 2 and 20 to determine the best cluster size. The sum of squared errors (SSE) is presented in Figure 1. From this graph alone, it's difficult to determine an 'elbow' point, or a point where the inertia starts decreasing in a linear fashion.

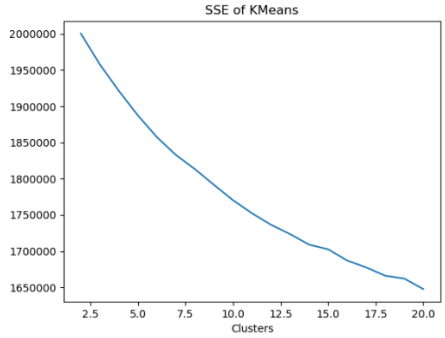


Figure 1

The homogeneity-silhouette curve is shown in Figure 2.

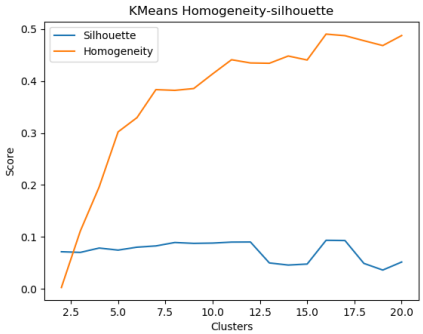


Figure 2

Homogeneity, which is the measure of how homogenous the labels in a cluster is, is monotonically increasing as expected. Silhouette score measures the similarity of an object to its own cluster compared to other clusters. It remains relatively stable at around 0.08. Taking these curves into account, we can determine the ideal K value at 16, which has the highest silhouette and homogeneity scores (0.094 and 0.491 respectively). It also has a relatively low SSE score. The V score, which is the harmonic mean of the homogeneity and completeness measures, was 0.271 at K=16.

The 178-feature input is very complex and includes data collected from 500 patients. The high K value likely tries to capture the complexity of the features. It also likely sees some similarity between patients' recordings as some patients are expected to have similar readings.

B. *GAMMA*

Using the same range of K values, we extracted the SSE values as presented in Figure 3. Again, it's difficult to determine an elbow point from this graph alone.

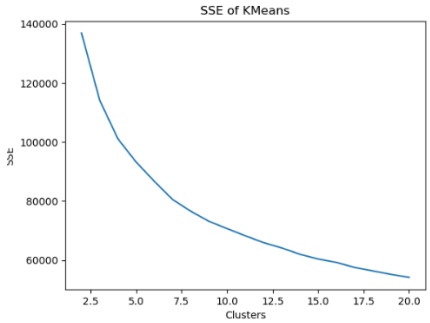


Figure 3

The homogeneity-silhouette curve is presented in Figure 4. Homogeneity is increasing almost monotonically here as well, but silhouette decreases as soon as the number of clusters increase above 2.

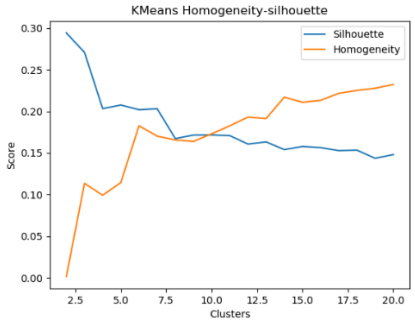


Figure 4

At K=2, silhouette is highest, but homogeneity is lowest. For a more balanced performance, K=10 is a good choice, but for the best silhouette score, K=2 is best. The V score at K=2 was 0.002, which is extremely low. At K=10, V score is 0.08. However, we chose the best K value as 2 since it had the highest silhouette score.

The input for this dataset consists of only 10 features, which is not as complex as the epilepsy dataset. As such, a low number of clusters got a high silhouette score. 10 clusters got a balanced performance.

IV. Expectation Maximization (EM)

EM is a probabilistic model that assumes all the data points are generated from a mixture of a finite number of Gaussian distributions. It can be thought of as a generalization of the K-Means algorithm to incorporate information about the covariance structure of the data.

A. *Epilepsy*

EM can give us different number of clusters because it takes into account the covariance of the data. As such, using SSE does not make much sense because of the different covariances. We tried different covariance types ('full', 'spherical', 'tied') as well as cluster sizes (in range 2 to 20).

We determined the best covariance type to be full. The BIC, or the Bayesian Information Criterion, is shown in Figure 5. The BIC is used to curtail complexity and overfitting by introducing penalties for the number of parameters in the model. The lowest BIC is at K=4, which we will use as the best number of clusters.

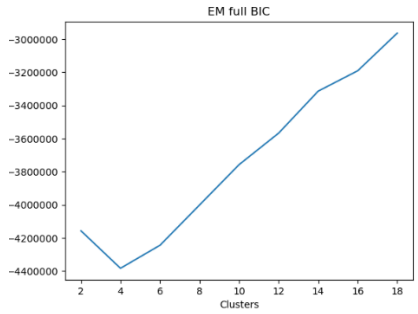


Figure 5

The V score at K=4 was 0.320. The homogeneity-silhouette curve is shown in Figure 6.

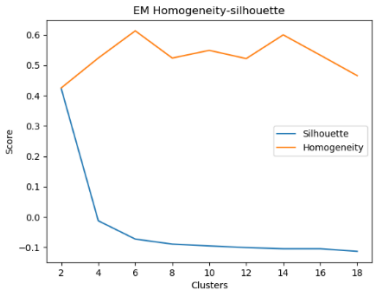


Figure 6

The silhouette score at K=4 is low, but the peak homogeneity is reached at 0.524. Considering these scores, we're comfortable in saying that the best number of clusters is 4.

B. *GAMMA*

Using the same parameter ranges, we determined the best covariance type to be full as in the epilepsy dataset. The BIC curve for this covariance type is presented in Figure 7. The lowest BIC was at 16 clusters. The V score was 0.135 at K=16.

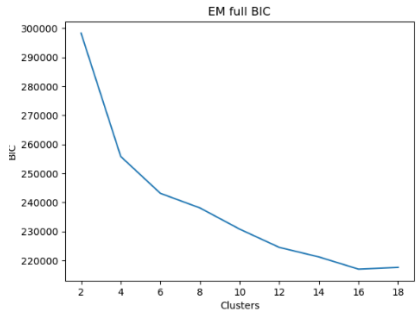


Figure 7

This result was surprising since the input only consist of 10 features and that we determined a cluster size of 2 was the best using K-Means. The homogeneity-silhouette curve is shown in Figure 8.

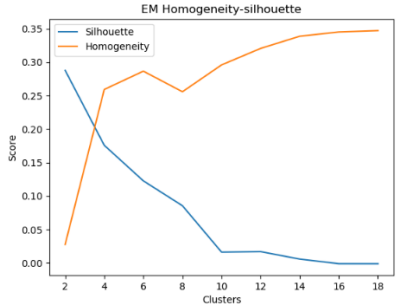


Figure 8

Cluster size of 4 gives us the most balanced performance. Silhouette and homogeneity are equal at 0.20, but the BIC is much higher. As such, we would be comfortable in saying that the best number of clusters to use is indeed 16 because it gives us the best V measures and lowest BIC.

Such a high number of clusters was likely chosen just to decrease the complexity of the model. Using K-Means, we determined the best K value was 2, but that was likely deemed very complex by EM, and thus 16 was chosen.

V. Dimensionality Reduction

A. *Principle Component Analysis (PCA)*

The PCA algorithm is used to decompose multivariate datasets using orthogonal components that explain a maximum amount of the variance. This method is sensitive to the scale of the features. As such, we used standard scaling.

1) Epilepsy

We searched for the number of features that would explain 0.95 of the variance, which was 39 dimensions. The eigenvalues, or the explained variance, is shown for each dimension in Figure 9. The first feature explains 10 percent of the variance.

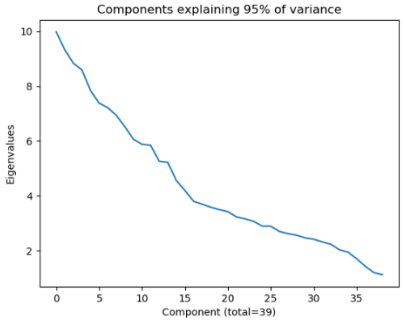


Figure 9

The number of dimensions of the input after reduction is only 22% of the original size. Figure 10 shows the projection of the first 2 components, which does not show good separation among the labels.

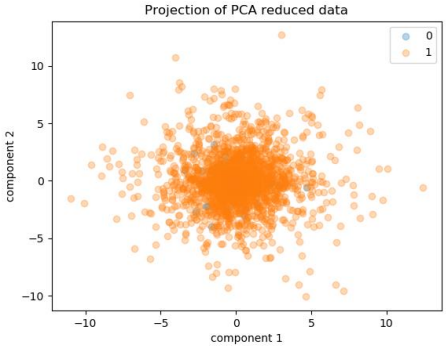


Figure 10

2) GAMMA

We again used 0.95 as the value of variance explained, which was determined to be 7 components. The eigenvalues for each component are shown in Figure 11. The amount of variance explained by the first component is 0.42.

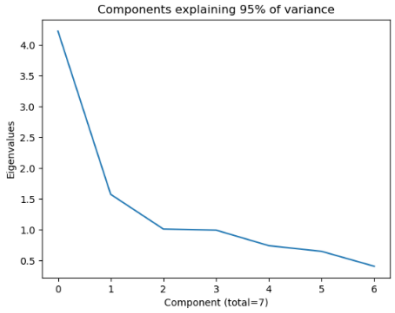


Figure 11

The number of dimensions after reduction was 7, which is 70% of the original size. Because the original input size is small, there isn't much room to reduce the dimensions of the input. Figure 12 shows the projection of the first 2 components of PCA, which shows decent separation of the labels.

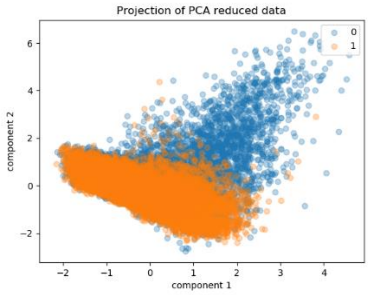


Figure 12

B. *Independent Component Analysis (ICA)*

ICA works by separating multivariate signals into additive subcomponents that are maximally independent. The underlying assumption is that each subcomponent is non-Gaussian distributed and are independent.

1) Epilepsy

We searched over a range of 2 through 0.80 of the total input dimension size. For this dataset, the upper limit of the dimensions we searched over was 142. We charted both reconstruction error and average kurtosis. Maximizing the kurtosis value means that we are

maximizing nongaussiantiy. This is shown in Figure 13. For 137 dimensions, kurtosis is highest at 145.034.

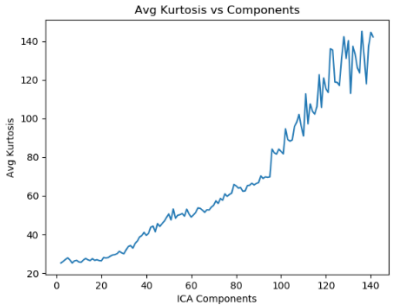


Figure 13

Figure 14 shows the reconstruction error. For high number of components, the error is very low. At 137 components, the error is virtually 0. For these reasons, we can comfortably say that 137 is the best number of components for this dataset.

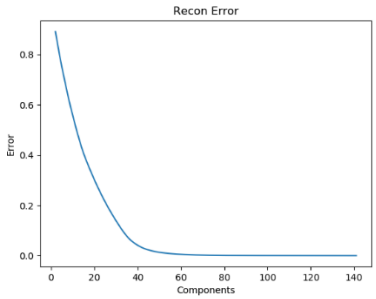


Figure 14

2) GAMMA

We searched over a similar range of input dimension sizes (2 to 0.80 of total size). The upper limit was 8 dimensions for this dataset. The average kurtosis for each dimension size is shown in Figure 15. Highest kurtosis of 10.118 was at 7 dimensions.

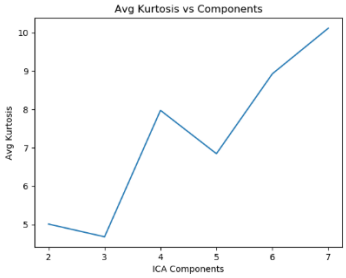


Figure 15

Figure 16 shows the reconstruction error. As expected, reconstruction error is lowest at 7 dimensions as well. As such, we can say that 7 components is the best choice for this dataset.

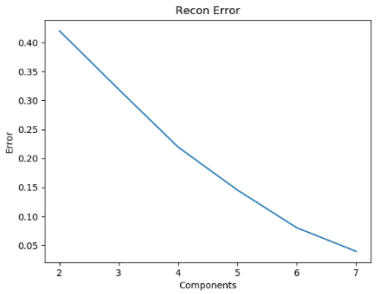


Figure 16

C. *Randomized Projections (RP)*

RP works by projecting the features on a randomly generated matrix, which is sampled from a Gaussian distribution. RP preserves the distances between any two data points. We ran RP 10 times and took the average for each metric.

1) Epilepsy

We searched over the entire input space to determine the best reduction size. Both average kurtosis and reconstruction errors were charted to determine this. Figure 17 shows the reconstruction error curve. The lowest error was at size 135.

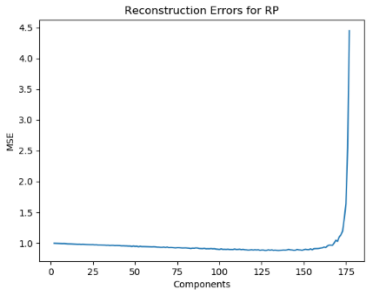


Figure 17

The average kurtosis is shown in Figure 18. As can be seen, the kurtosis does not provide a good indicator of the ideal number of dimensions as the curve oscillates around the same constant value. As such, we will use the lowest reconstruction error and choose 135 as the best component size.

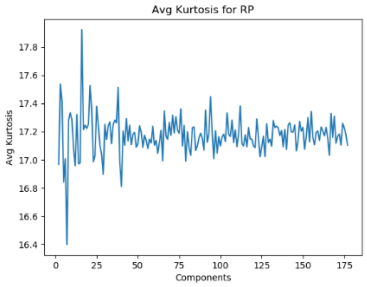


Figure 18

2) GAMMA

We again searched over the entire input feature space and tracked both kurtosis and reconstruction errors. Figure 19 shows the reconstruction error curve. The lowest error is at component size of 5.

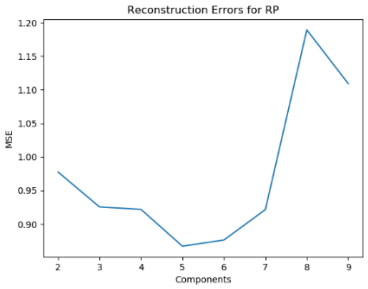


Figure 19

Figure 20 shows the average kurtosis values. Highest kurtosis was at sizes 2 and 3, but these gave higher reconstruction errors. For this reduction algorithm, we believed that reconstruction error was more important and as such, we chose 5 as the ideal size for this dataset.

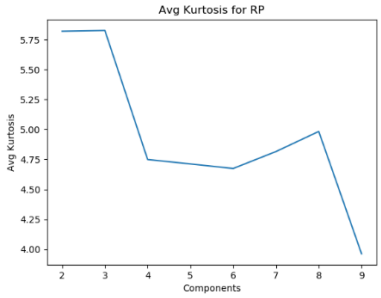


Figure 20

D. *Singular Value Decomposition (SVD)*

SVD works by calculating the k largest singular values, where k is a parameter that we have to specify. It works very similarly to the PCA algorithm, but with SVD, the input matrix does not need to be centered.

1) Epilepsy

We chose to use the entire input space for the k value and keep only the features that explain 0.95 of variance, similar to how we chose the input size for PCA. The components and what percentage of the variance they explain is presented in Figure 21. We determined that 38 components explain 0.95 of variance.

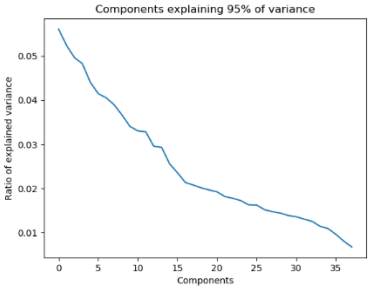


Figure 21

We charted the first two components, which can be seen in Figure 22. As can be seen, there is not good separation of the labels.

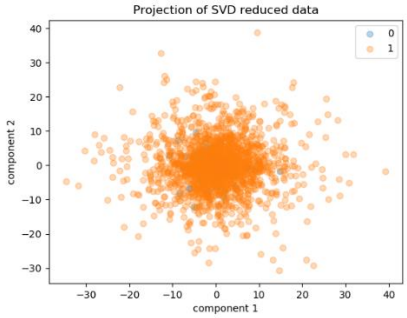


Figure 22

For this dataset, we chose 38 as the best dimension size because it explained 0.95 of variance.

2) GAMMA

We again searched over the entire input space and chose the dimension size that explained 0.95 of variance. The component curve is presented in Figure 23. We determined that 6 features explain 0.95 of variance.

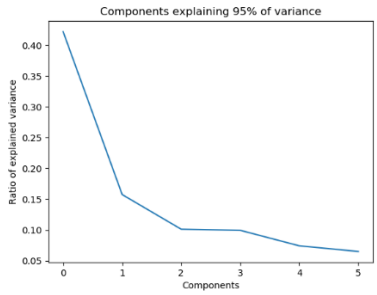


Figure 23

We charted the first two components, which can be seen in Figure 24. This as well as the previous component projection looks identical to that of PCA, which indicates that the same components have been chosen. As with PCA, there is decent separability between the labels.

As such, we choose 6 as the best dimension size.

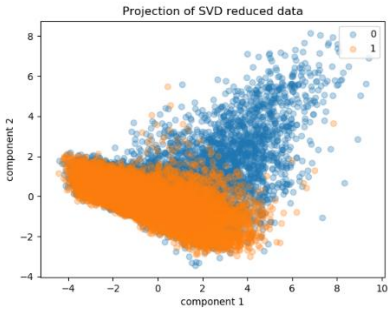


Figure 24

VI. Clustering after Reduction

We give brief overviews of using our two clustering algorithms on reduced data since there are 16 different combinations.

A. *K-Means*

1) PCA

a) Epilepsy

As explained before, we used dimension of 39. The homogeneity-silhouette curve is presented in Figure 25. The highest

silhouette score was 0.07 at K=17. We get a similar K value as with the full data, but we get a slightly worse score across the board. The V measure was 0.232 for our reduced data versus 0.271 for the full dataset. This is a similar clustering to the original clustering on the unreduced data. It is slightly different however.

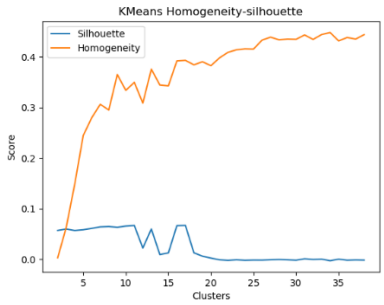


Figure 25

b) GAMMA

We used dimension size of 7. The homogeneity-silhouette curve is shown in Figure 26. The highest silhouette score was 0.199 at K=2. The curve shows a more balanced performance at higher dimensions. Similar to using the full dataset, we get the same K value with a lower silhouette. However, the V measure for the reduced data was 0.133 versus the 0.002 for the full dataset, which is significantly better. Using a reduced dataset gave us better results for the V measure. Again, we get a similar clustering here as with unreduced data.

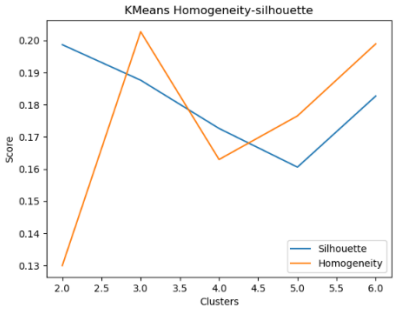


Figure 26

2) ICA

a) Epilepsy

ICA determined the best component size to be 137. The homogeneity-silhouette curve is in Figure 27. The best silhouette of 0.011 was at K=2. Even the V measure was significantly lower for the reduced data (0.0001 versus 0.271). Clustering on ICA was worse across the board.

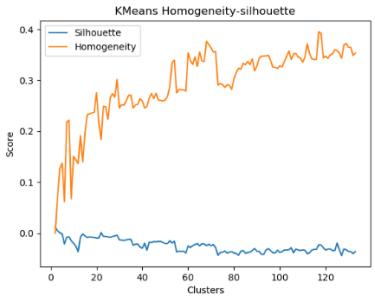


Figure 27

b) GAMMA

ICA's best dimension size was 7. The homogeneity-silhouette curve is in Figure 28. The highest silhouette was 0.199 at K=2. Though the silhouette was lower than the full dataset, the reduced clustering got a better V measure (0.133 reduced versus 0.002 full).

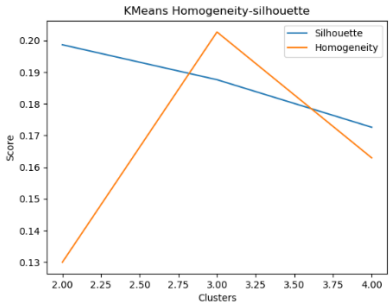


Figure 28

3) RP

a) Epilepsy

RP's best dimension size was 135. Figure 29 shows the relevant curves. Highest silhouette of 0.619 was at K=2, which is similar to ICA but with much better score. V measure was lower than the full dataset (0.183 reduced versus 0.271 full). The homogeneity is low for K=2, but it has a high completeness score. This clustering gives us overall better results than the full dataset but with some tradeoffs.

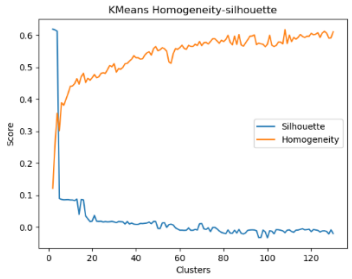


Figure 29

b) GAMMA

RP's best dimension was 5. The relevant figure is shown in Figure 30. K=3 was chosen as the best with silhouette score of 0.337. Overall, the reduced dataset gives much better scores across the board. V measure of reduced was 0.071 versus 0.002 of the full dataset. Using RP gives us better results than the full dataset.

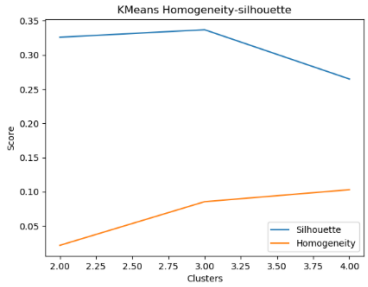


Figure 30

4) SVD

a) Epilepsy

SVD's best dimension size was 38. The relevant figure is in Figure 31. As expected, this curve is very similar to that of PCA's. The best K was 12 with silhouette of 0.094, almost identical to using the full dataset. V measure was comparable as well, but slightly lower. This clustering is most similar to the unreduced data, indicating that SVD has kept all the pertinent features and removed the ones with little relevance.

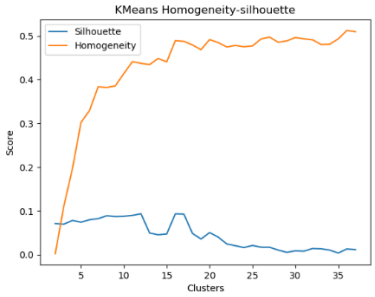


Figure 31

b) GAMMA

The best dimension size was 6. Figure 32 shows the relevant figure. The best K was 2 at silhouette of 0.295, again almost identical to using the full dataset. Other measures were identical as well, indicating that reducing using SVD was no different than using the full dataset. This performance is better than PCA. This closeting is much different from the original. It maintains a higher silhouette score, but lower homogeneity. Still, this is preferred because it gives us better clustering overall.

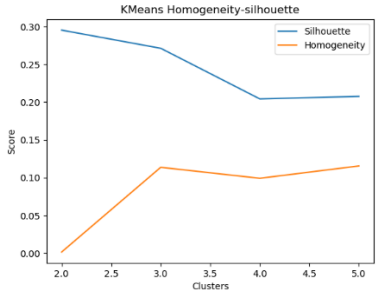


Figure 32

5) Comparison

Using RP tended to give us more consistently better results, whereas other reduction algorithms led to K-Means performing with tradeoffs. RP is likely the best choice for reduction for K-Means using these datasets.

As for likeness with clustering on unreduced data, SVD gives us similar clustering for epilepsy dataset, and PCA gives us similar results for GAMMA. Likeness is due to the reduction algorithm keeping the most pertinent features while removing the less relevant ones.

B. *Expectation Maximization*

1) PCA

a) Epilepsy

Using dimension size 39, we determined the best cluster was 8. The BIC curve is shown in Figure 33. The V measure for this cluster size was 0.358, higher than the full dataset, but silhouette was -0.124, lower than the full dataset. BIC was also higher with reduced data.

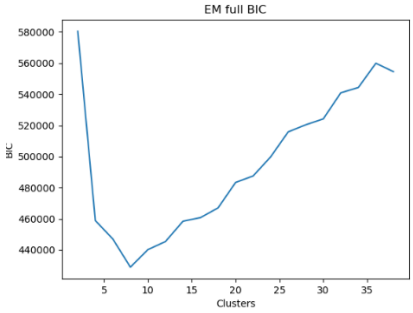


Figure 33

b) GAMMA

Using dimension size 7, we determined the best cluster was 6. The BIC is shown in Figure 34. Compared to full dataset, the number of clusters was much smaller, but BIC was slightly higher. V measure was slightly higher as well by 0.02, and silhouette by 0.073. Overall, using reduced data gave us better metrics except for BIC.

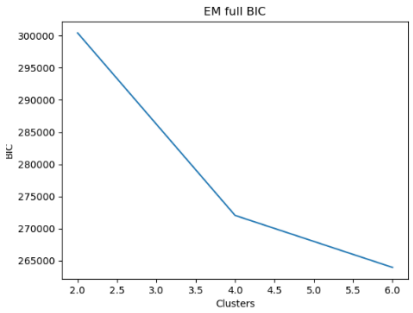


Figure 34

2) ICA

a) Epilepsy

Using dimension size 137, we determined the best cluster to be 4. BIC is shown in Figure 35. Performance on ICA reduced data is much better across the board than using the full dataset. We get much lower BIC, silhouette higher by 0.1, and V measure higher by 0.065.

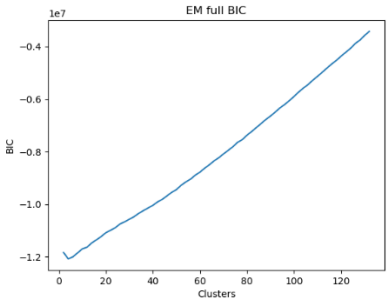


Figure 35

b) GAMMA

Using dimension size of 7, we got the best cluster as 4. The BIC is shown in Figure 36. We get a much lower BIC, a higher silhouette (0.115), and higher V measure (0.175) versus using the full dataset. This combined with the epilepsy outcome indicates that using EM clustering on ICA reduced data is better than using the full dataset in these cases.

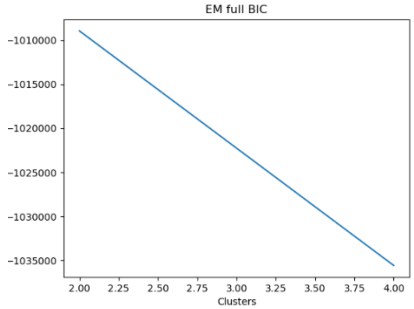


Figure 36

3) RP

a) Epilepsy

RP's best dimension was 135. The best cluster size was 4. The BIC is shown in Figure 37. We get a higher BIC and lower silhouette, but higher V measure versus the full dataset. Overall, the performance using the reduced dataset is worse.

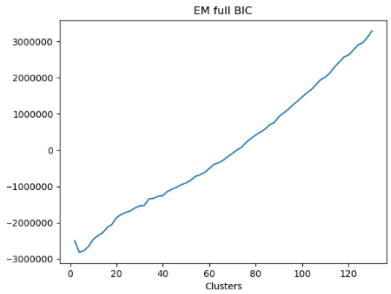


Figure 37

b) GAMMA

Using dimension size of 5, we got the best cluster to be 4. The BIC is shown in Figure 38. Using the reduced data got a much lower BIC and a much higher silhouette (0.135), but lower V measure (0.097). Since the V measure difference wasn't very high, we're comfortable in saying that using the reduced data gave us a better performance using EM clustering.

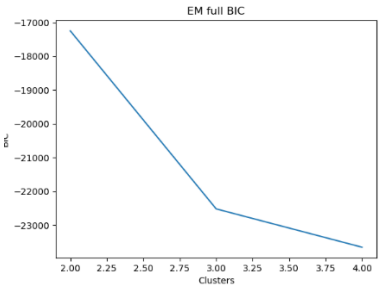


Figure 38

4) SVD

a) Epilepsy

Using dimension size of 38, we got the best cluster size as 4. The BIC is shown in Figure 39. Using the reduced data gave us comparable V measure and BIC as the full data, but worse silhouette score (-0.093). As such, using the SVD reduced data gave us worse performance overall.

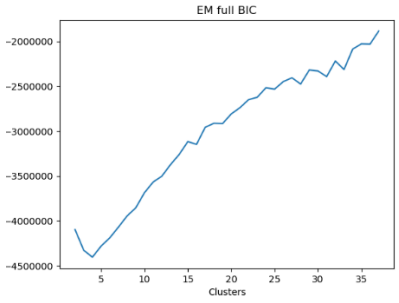


Figure 39

b) GAMMA

Using dimension size of 6, we got the best cluster size as 5. BIC is shown in Figure 40. We got a higher BIC than the full dataset, and comparable V measure (0.137), but better silhouette (0.097). As such, using SVD does not seem to be ideal for EM clustering on these datasets.

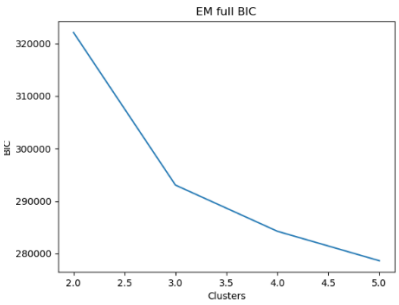


Figure 40

5) Comparison

Overall, using ICA reduction gave us better clustering performance using EM for both datasets. RP did well in the GAMMA dataset, but not in epilepsy. PCA and SVD had too many tradeoffs to be considered good.

RP and SVD both give very similar clustering to the original for epilepsy data, but SVD is closer to the original results. For GAMMA, PCA, SVD, and RP all give similar results with SVD once again giving us more alike results to the original clustering. This indicates that SVD and RP did the best in keeping the relevant features and discarding the rest.

VII. Neural Network after Reduction

After running both datasets, we chose to use the GAMMA dataset for this part because we generally get more interesting results. So, the following will all be discussions of the GAMMA dataset. As before, we consider AUC score to be more important than either f1 or accuracy, though we will also discuss these metrics as well. On unreduced data, the best model got accuracies of 0.84 and AUC of 0.90 against the test set. For the training sets, both validation and training accuracies were as high as 0.97. We will use these metrics for comparison.

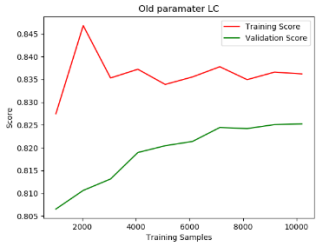
A. PCA

We used a dimension size of 7, which explained 0.95 of the variance. We ran the neural network using the best parameters we obtained in project 1 (hidden layer sizes = (10, 10), learning rate=0.001, logistic activation), which we will call old parameters. The learning curve to suffer from high bias. Against the unreduced data,

the accuracies here are very low, never reaching values above 0.825. AUC was also lower than using the full dataset at 0.88.

After optimizing for the reduced data, we got the best parameters as follows: hidden layer sizes = (15, 15), learning rate=0.01, relu activation. The learning curve and ROC are shown in Figure 41. The high bias problem is somewhat addressed with the new parameters, but it's still present. We see the variance has increased somewhat as well, which is to be expected.

We get slightly better results across the board, but they are still worse than using the full dataset. For reference, the best AUC is 0.89 for the reduced dataset, and f1 is 0.82. Recall and precision for both classes are also worse. This indicates that we've lost some features important for the neural net to learn.



Classification report:

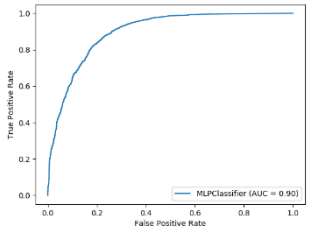
	precision	recall	f1-score
0	0.80	0.68	0.73
1	0.84	0.91	0.87
accuracy			0.83
macro avg	0.82	0.79	0.80
weighted avg	0.83	0.83	0.82

Figure 41

B. *ICA*

Using dimension size of 7 again, we compare the old and new best models. Using the 7 features selected by ICA, the model doesn't seem to be learning at all as the validation curve was completely stagnant. The training scores decrease with more samples, further supporting this idea. This could be an issue with picking a learning rate that is too high. AUC is very low as well at 0.82.

For sake of brevity, we will only discuss the optimized model for the reduced data. Figure 42 shows the classification report and ROC. It got a higher AUC score of 0.90, but it performed horribly in recognizing negative instances. As such, f1 was 0.78 and accuracy was similarly low at 0.79. It seems that ICA removed features that were important for the neural net to learn to classify on the dataset. As such, ICA reduction is not ideal.



Classification report:

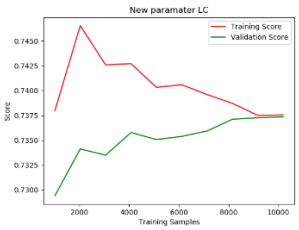
	precision	recall	f1-score
0	0.77	0.56	0.65
1	0.79	0.91	0.85
accuracy			0.79
macro avg	0.78	0.73	0.75
weighted avg	0.78	0.79	0.78

Figure 42

C. *RP*

This time, we used dimension size of 5. With the old model, we get a very high bias learning curve and a very low AUC score of 0.73. Negative recall for the old model was only 0.43, which is very low, and positive precision was also very low at 0.75.

Using an optimized model, we get a learning curve and classification report as shown in Figure 43. There is much more variance than with the old model, but there is still too much error to call this a good performance. The AUC improved to 0.74, but it still had problems identifying negative classes. This indicates that RP also lost some features important in correctly identifying negative classes. With such low accuracy and f1, it's difficult to call RP a good reduction algorithm in this instance.



Classification report:

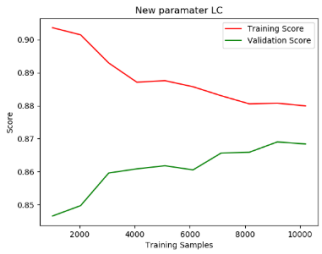
	precision	recall	f1-score
0	0.74	0.41	0.53
1	0.74	0.92	0.82
accuracy			0.74
macro avg	0.74	0.67	0.68
weighted avg	0.74	0.74	0.72

Figure 43

D. *SVD*

Using dimension size of 6 and the old model parameters, we got a learning curve with high bias and a very high AUC score of 0.93. Even the f1 and accuracy scores (both 0.87) against the test set was higher than with the full dataset. This was a very promising performance.

Using an optimized model, we got a performance that we can easily call the best among the reduced datasets. The learning curve and the classification reports are in Figure 45. The accuracy and f1 scores have improved and the AUC remains the same at 0.93. Compared to using the full dataset, we get much better and balanced performance across the board for both negative and positive classes. The learning curve shows higher variance than with the old model, which shows that it overcame its high bias.



Classification report:

	precision	recall	f1-score
0	0.87	0.75	0.81
1	0.87	0.94	0.91
accuracy			0.87
macro avg	0.87	0.85	0.86
weighted avg	0.87	0.87	0.87

Figure 45

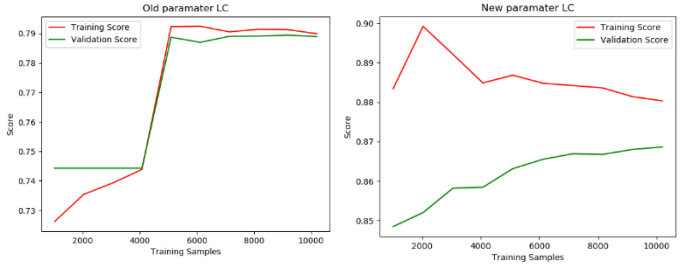
We can confidently say that SVD was the best reduction algorithm for this problem. It gave us better performance with fewer features, which indicates that it successfully cut out the features that were muddying the waters, as it were. In other words, SVD does a great job of keeping the very relevant features and discarding those that weren't.

VIII. Neural Network after Clustering

Again, we choose to discuss the GAMMA dataset as it's a more interesting one.

A. *K-Means*

We chose to use K=2, which we determined as the ideal cluster since it had the highest silhouette score. The old model got a low AUC of 0.84, but it gave us good performance on f1 and accuracy (0.86, 0.87 respectively). The learning curve (left below) showed very high bias and clearly shows the distinction between the two clusters.



The optimized model (right above) gives us a better result. With high enough model complexity, we're able to overcome the bias

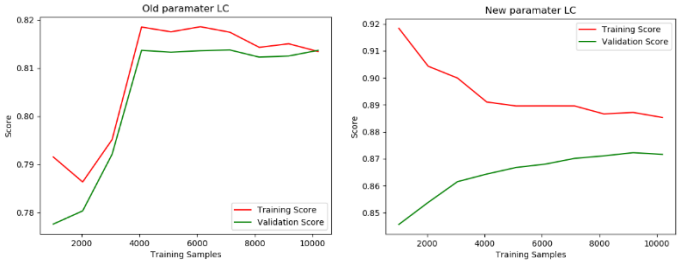
problem as the curve indicates. AUC was much better at 0.93, and it also gave us much better performance than using the non-clustered dataset on f1 and accuracy. It's a much more balanced performance and gives us insight into the advantages of using clustering first.

Classification report:

	precision	recall	f1-score
0	0.91	0.73	0.81
1	0.87	0.96	0.91
accuracy			0.88
macro avg	0.89	0.85	0.86
weighted avg	0.88	0.88	0.88

B. *EM*

The dimension size we used was 16 since this had the lowest BIC. The old model got an AUC of 0.88, slightly lower than without clustering, but we got excellent performance on accuracy, f1, and on individual classes. The learning curve (left below) is indicative of high bias once again.



After optimizing to the clustered dataset, the learning curve (right above) looks much better with sufficient model complexity. We get better scores across the board than both the un-clustered data and the old model. AUC for the optimized model was 0.94, the highest yet. This is very similar to the numbers we got from the K-Means dataset, and this excellent performance also shows the benefits of clustering as the terrible performance on negative classes from the un-clustered data is much better after clustering.

Classification report:

	precision	recall	f1-score
0	0.88	0.77	0.82
1	0.88	0.94	0.91
accuracy			0.88
macro avg	0.88	0.86	0.87
weighted avg	0.88	0.88	0.88

From these metrics, it's clear that clustering using EM first tends to give us much better results. Even the negative recall, which the old models were struggling with, is much better with this clustering. This might be because clustering gives the model a much clearer idea of how to classify each class.

IX. Clock Times

Neural network ran much faster on reduced and clustered datasets than the full dataset. This is because they run on lower space states.

EM generally took longer to cluster than K-Means as it is the more complex algorithm. All of the reduction algorithms took about the same time per execution except for RP, which was slightly faster than the other three.

X. Conclusions

Across both datasets, we've seen how dimension reduction can massively improve performance by eliminating potentially confounding and useless features. With epilepsy dataset, we saw that only 22% of the features explain 95% of the variance. We further saw that reducing the dimensions gave us much better results when using neural networks to classify. Clustering first also gave us much better results for both datasets as clustering seems to give the model a much clearer idea of how to go about classification.

For clustering on reduced datasets, we saw that using RP was best for K-Means, and ICA was best for using EM as they gave better results. We also saw that SVD was the ideal choice to reduce the GAMMA dataset when we use neural networks for classification. We finally saw that both K-Means and EM both gave similar performances when using optimized neural networks, but EM beat out K-Means by a few points in certain metrics. For both clustering datasets, we had to increase the model complexity to reduce high bias.

XI. Additional Notes

We ran the neural networks for clustered data a few times over to increase model complexity as the initial results showed very high bias. These results are not included for the sake of brevity.