Ragy Haddad

Unsupervised Learning

Content Map:
- Data Overview/Features/Sources.
- Clustering Algorithms
- PCA on two datasets
- PCA And Neural Network
- PCA And Clustering

Data Sources:

Dataset I: Network packet data used in KDD cup 1999 to classify malicious connections.

Dataset II: Optical recognition

http://www.kdd.org/kdd-cup/view/kdd-cup-1999/Data

https://archive.ics.uci.edu/ml/datasets/optical+recognition+of+handwritten+digits

Feature Explanation And Pre-Processing for Dataset I:

The dataset contains packet data and a label of the attack type the packet data corresponds to.

So a row in the dataset looks like the following table:

Total number of features: ***39*** *+* ***general label.***

Total number of rows: ***300,000***

Noise: ***The dataset includes some noise where the same features occasionally correspond to different labels.***

F1	F2	Fn	Label

Handling Categorical Features:

Categorical features were transformed to *Integer* labels using *sci-kit learn's OneHotEncoder*

Feature Explanation And Pre-Processing for Dataset II:

The second dataset used is the letters dataset, and I thought it would be a good way to gain a solid foundation in unsupervised learning and PCA, in addition to my growing interest in computer vision applications.

Total number of features: ***64*** *+* ***general label.***

Total number of rows: ***1800***

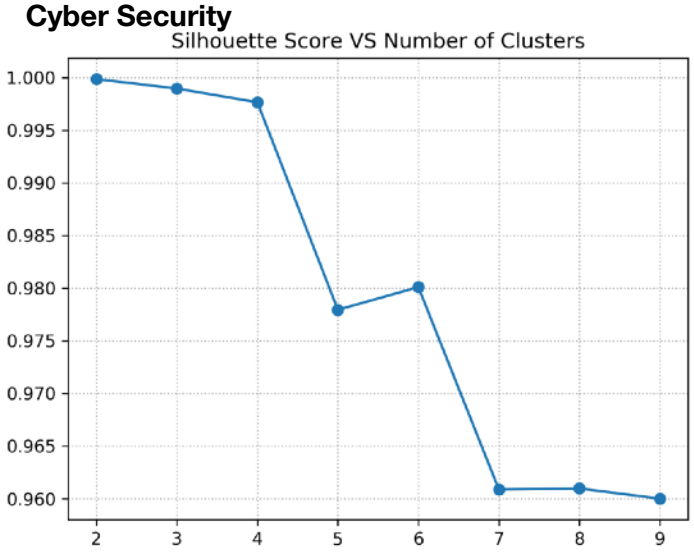
Classes: 10

Clustering Algorithms:

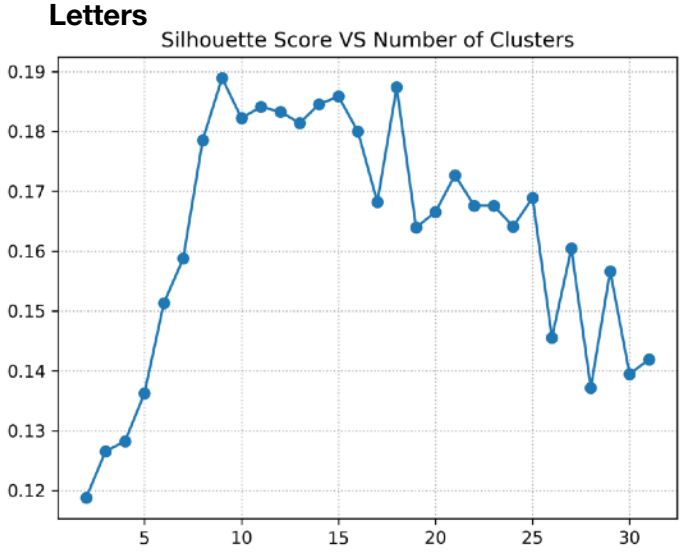
K-Means Clustering:

K-means is an unsupervised learning algorithm which instantiates cluster based on the mean of the data, it then iteratively adjusts the cluster means and gathers the data around the clusters that each cluster encompasses. For our initial testing of k-means we will vary our number of clusters and compute the silhouette score. The silhouette score is calculated using the mean intra cluster distance and the mean nearest cluster distance. A silhouette score of 1 is best and worst is 0. The lower the silhouette score the more overlap there is between samples. This is similar to the elbow method and allows us to non trivially select our number from the results we can see the number of

Cyber Security



Letters

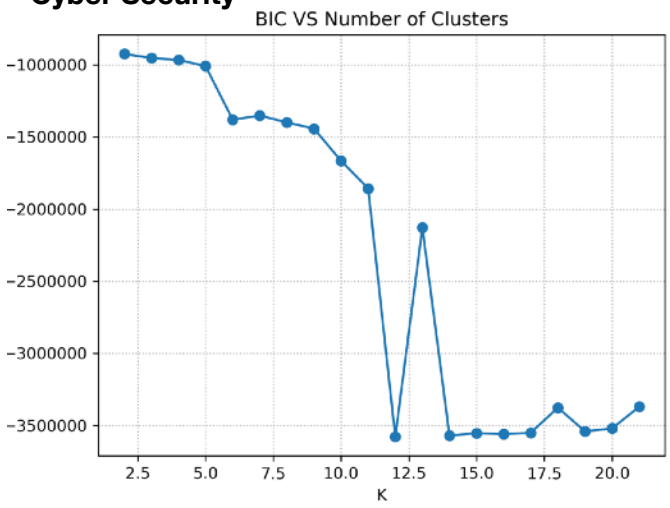


clusters needed with minimal overlap. For the cyber security attacks (2 classes) we can see that we have a high score silhouette score for 2 clusters which makes sense. However for the Letters dataset (10 classes) we can see that the best score is for 9 clusters, this could be because of high overlaps between the data. A normalization could be applied to the letters data set to adjust this overlap.

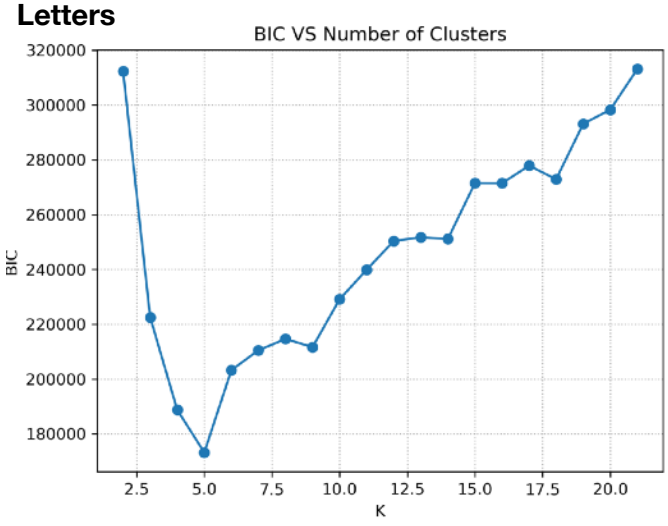
GMM Clustering:

Gaussian mixture models is a soft clustering algorithm as opposed to hard clustering in K-means. So instead of clustering a data point to a certain cluster. GMMs are capable of telling us a probability to which a data point is assigned to a cluster. The probability assigned is based on the gaussian(normal) distribution of each cluster.

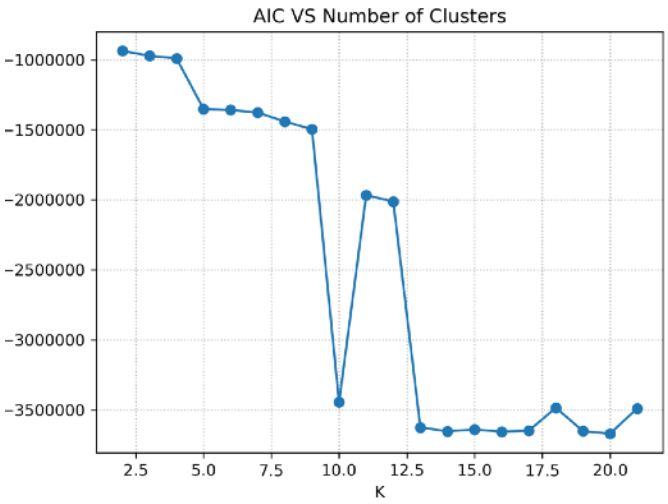
Cyber Security



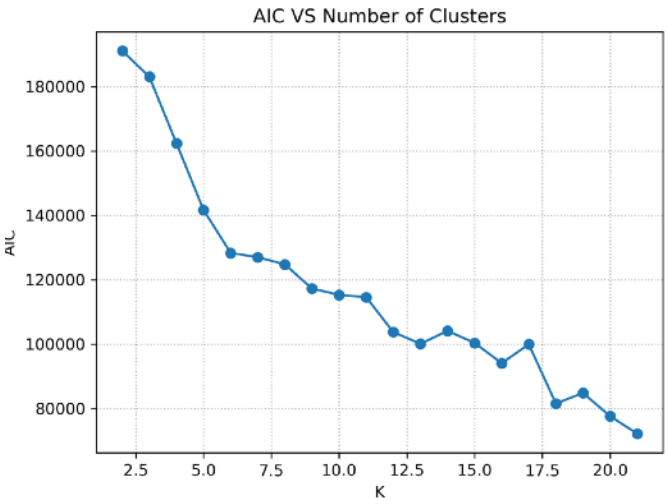
Letters



Cyber Security



Letters

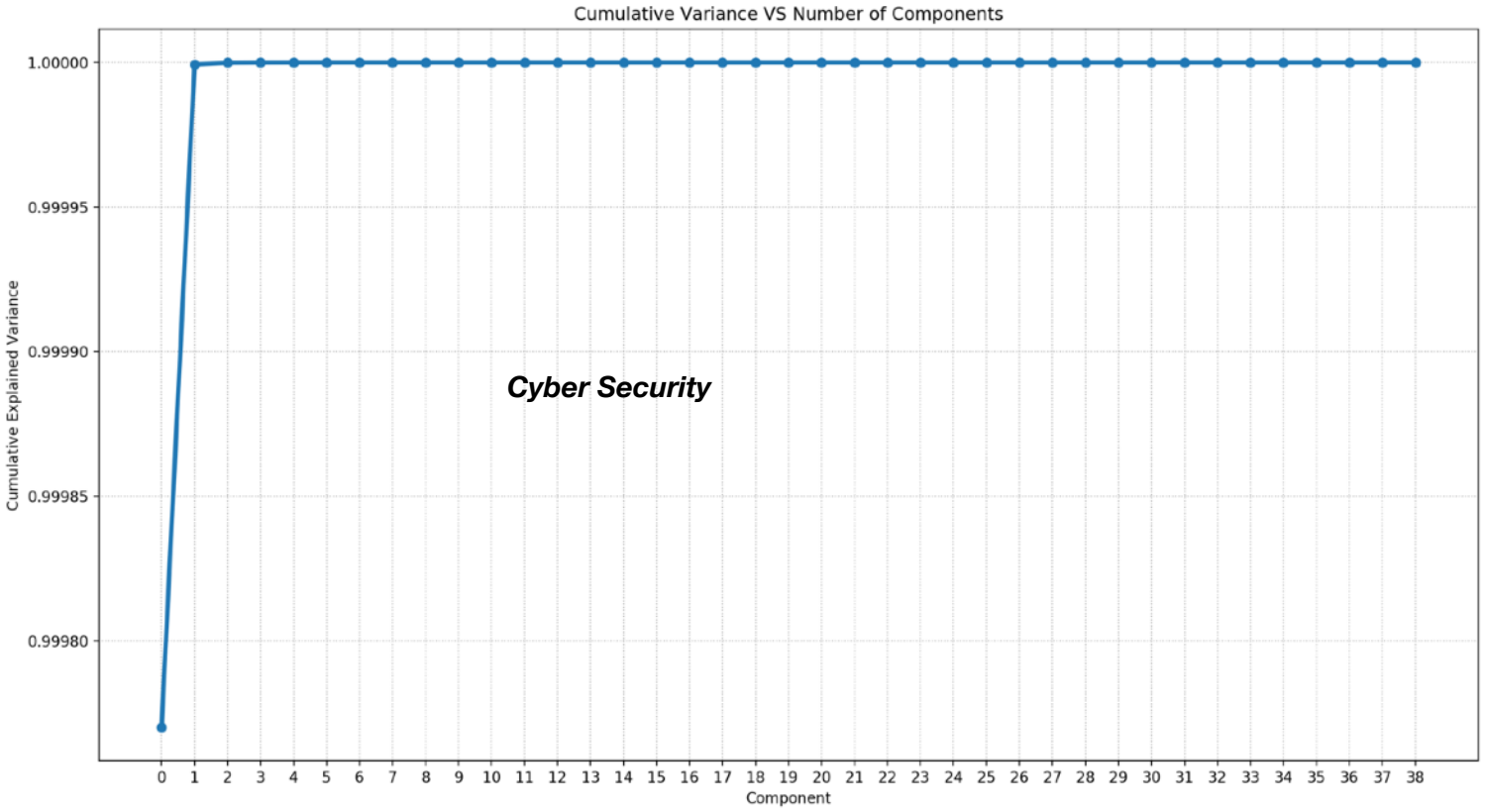


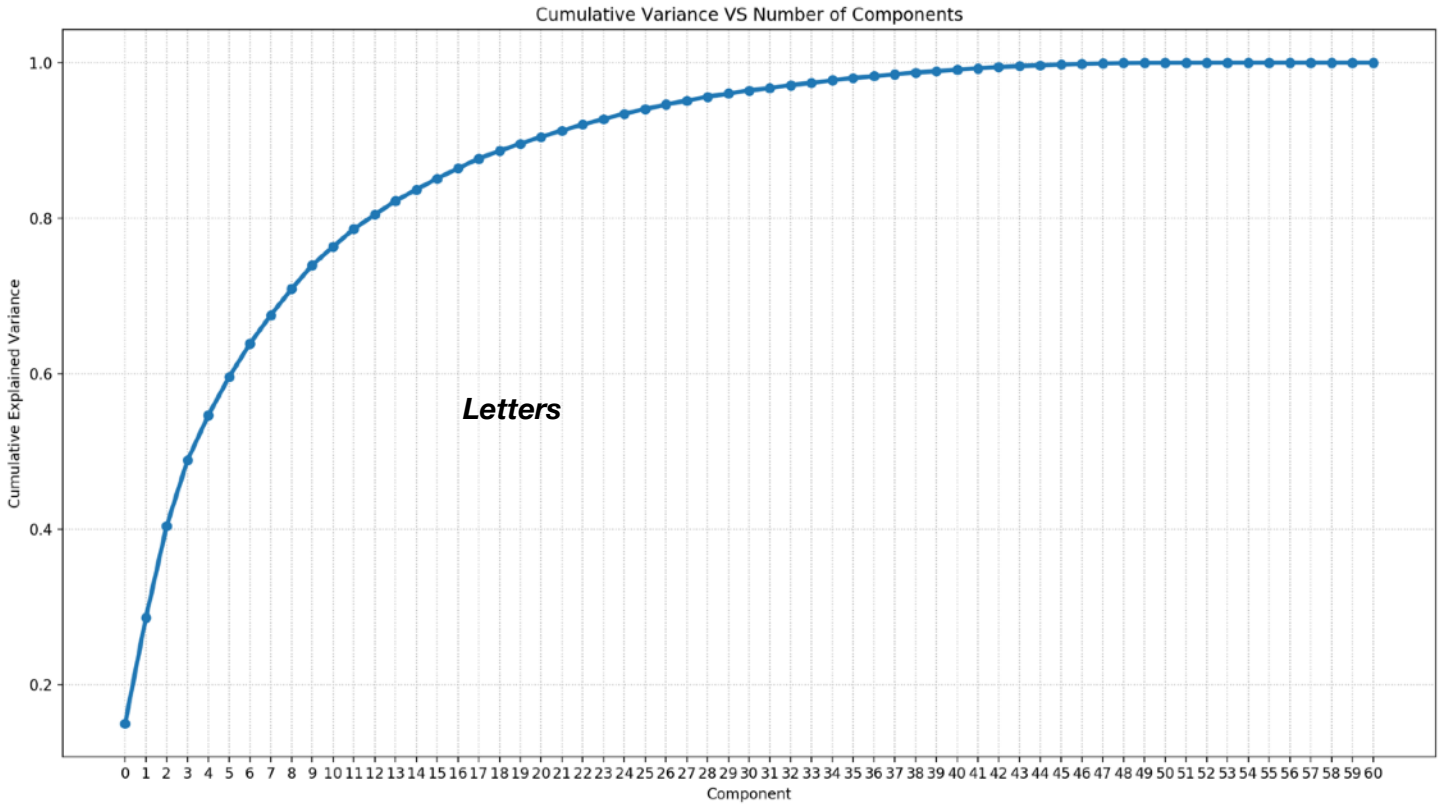
From the figures above we are plotting the AIC and BIC depending on the number of clusters. We use these metrics to select our best model, the lower the values of AIC and BIC the better the model. However, there is more depth to these metrics. The change in AIC is what is more important than just the value. So the relative change in values of AIC tells us how good our model is, on the left measuring the AIC for cyber security for example the biggest relative difference in AIC is between 2.5 and 10 which helps us see that there is something significant about the model

Dataset	**K-Means**	**GMMs**
Cyber Security	K = 2	K = approx(2)
Letters	K = 9	K = 5

Principal Components Analysis:

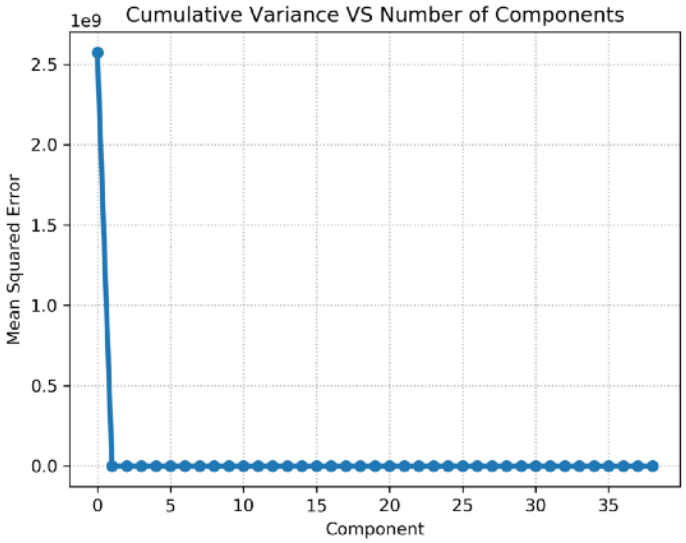
Now we start exploring another unsupervised learning technique, PCA. Principal component analysis helps us reduce the dimensionality of our data and helps us point out the dimensions that cause the most variation of our data thus having more influence on the variation of characteristics within our data. Another way to think about this is "How many characteristics do you actually need to describe data". The first approach we will explore for PCA is finding the number of components needed to explain our data, we do this by calculating the cumulative explained variance of our data and finding the number of components that cumulatively address almost 100% variation in data.

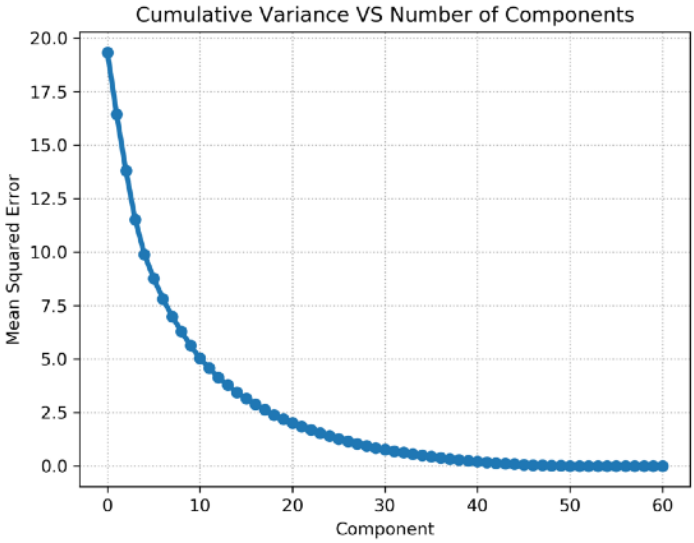




It is often perceived that selecting the number of components is trivial but you can actually get a nice estimate to the number of components needed to explain your data using the cumulative sum of the explained variance. In this analysis we see that from the total sum of attributes for cyber security it appears that the first two components account for a very large variation of our data. Which means they are important dimensions and cause a lot of variation. In the letters dataset you can see that the first 42 dimensions can help us explain almost 100% of the variation of the data. You only level of at 1.00 explain variance of course because you are still doing a dimensionality reduction and you will lose a minute amount of information from the data.

PCA Reconstruction:
Now let us reconstruct our data from the optimal number of components and measure the Mean Squared Error we do this by performing a back projection to the original data from the the reduced data.





From the figures you can see that the error is very low when we approach the number of components that we have concluded to be optimal from the cumulative variance exploration. Which makes sense and proves that these are good parameters for the number of components needed to study the data. However the data for Cyber security appears to rely heavily on the first two components. We will explore this anomaly further during our next steps.

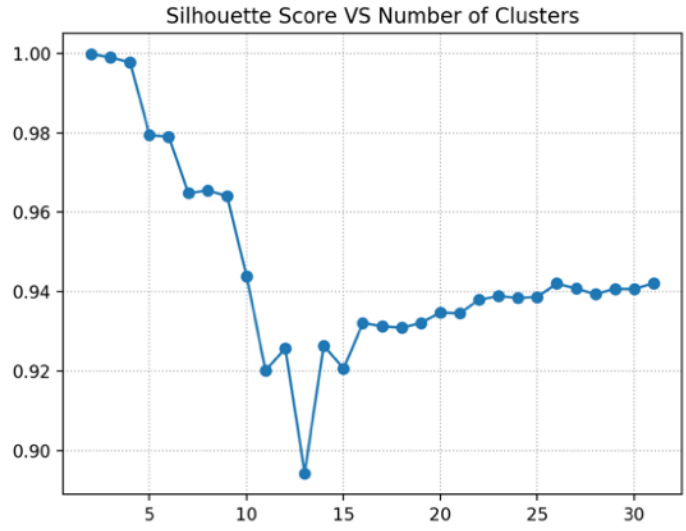
Dataset	Optimal Number of Components
Cyber Security	~2 - 5
Letters	~ 37 - 45

Now using our finding about the optimal number of components, we will re run our clustering algorithms with the new number of components and see if we get better or worse clusters.

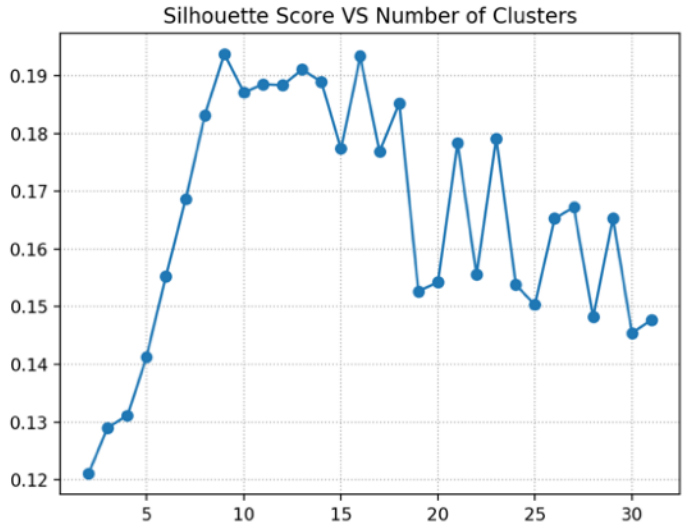
Post PCA Clustering:

We now will address the clustering metrics for Kmeans and GMMs after we run our PCA (Using reduced data set) For cyber security we will use k = 3 and for Letters we will use k = 37. First lets look at K-means

Cyber Security comp = 3

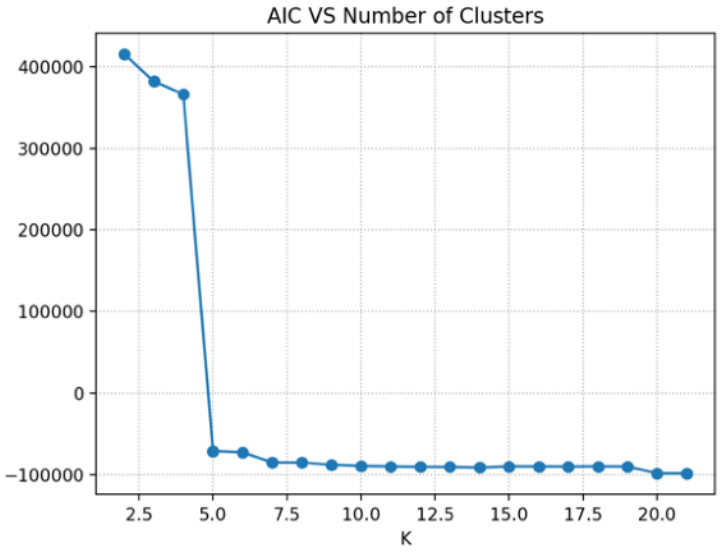


Letters Comp = 37

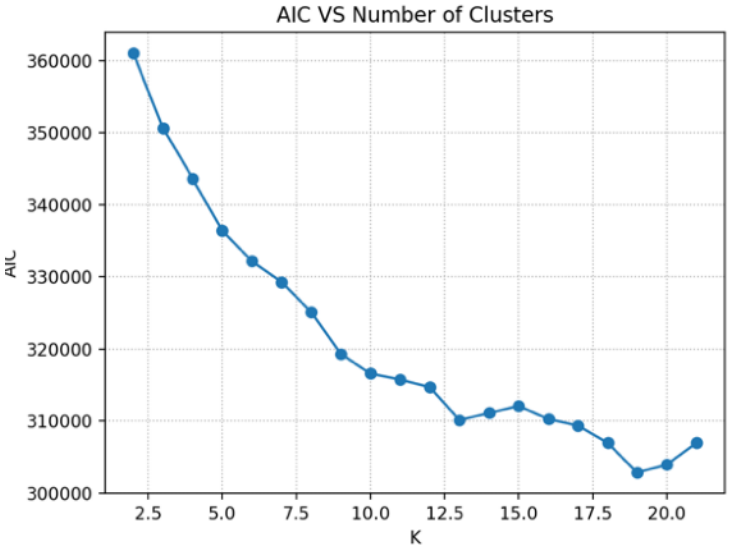


From the figures you can see that for cyber security the increase in Silhouette score remains minimal, so the effect of PCA education on forming better clusters was not that large. On the other hand for the letters dataset we get a significant improvement of silhouette score which means that data reduction for letters helped us create more separated clusters. Now we will apply the same technique but we will use GMM as our clustering algorithm. We will use AIC as our metric for the goodness of clusters.

Cyber Security comp = 3

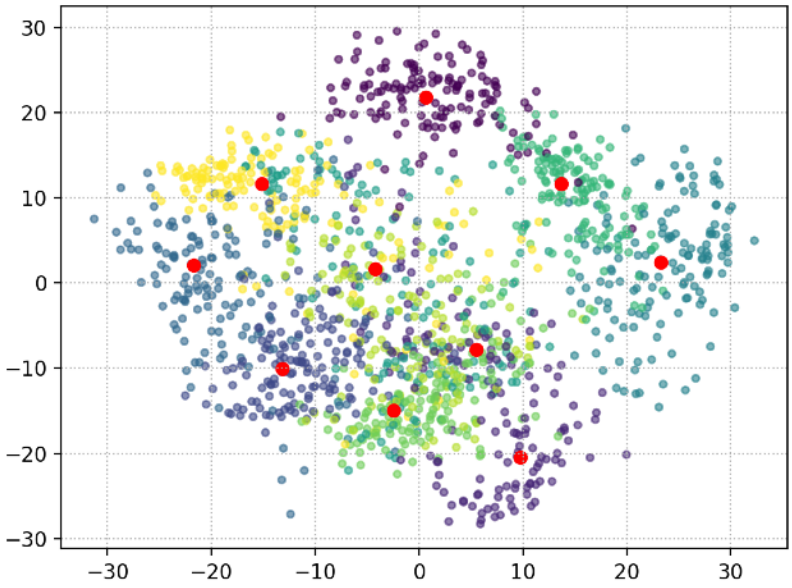


Letters Comp = 37

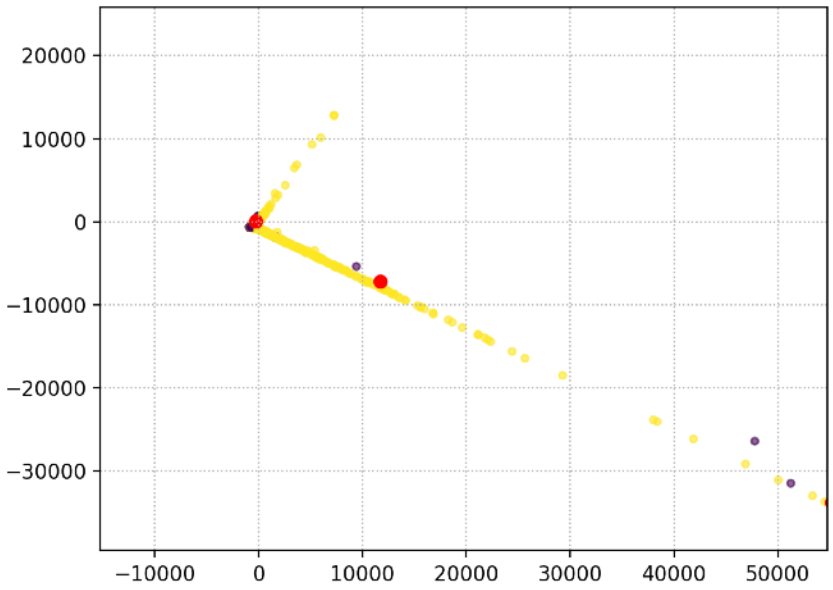


From the above figures you can see that our AIC improved for GMMs, it is noticeable that the data reduction led to a higher variance for AIC compared to the AIC we got without PCA. Still the improvement for cyber security was not as highly noticeable. But it was noticeable for Letters dataset.

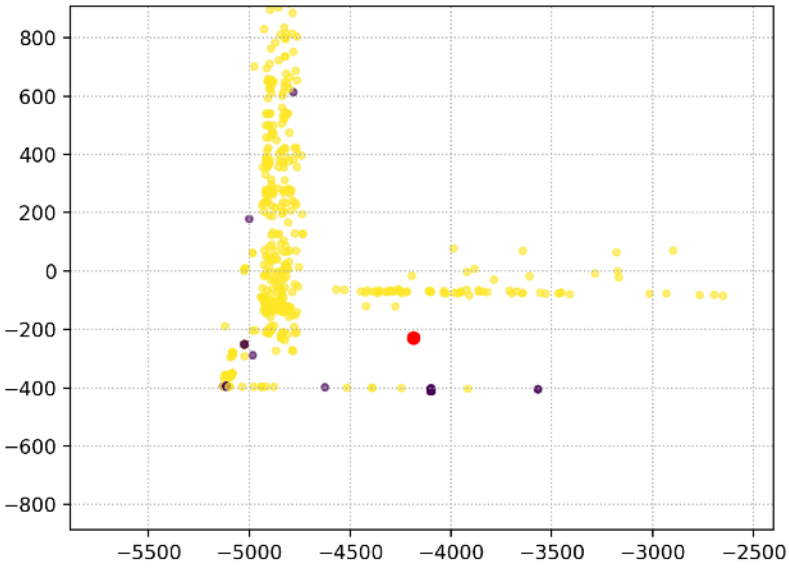
Letters Clustering in 2D



Cyber Security Clusters in 2D space



Zoomed in View



PCA + Neural Network:

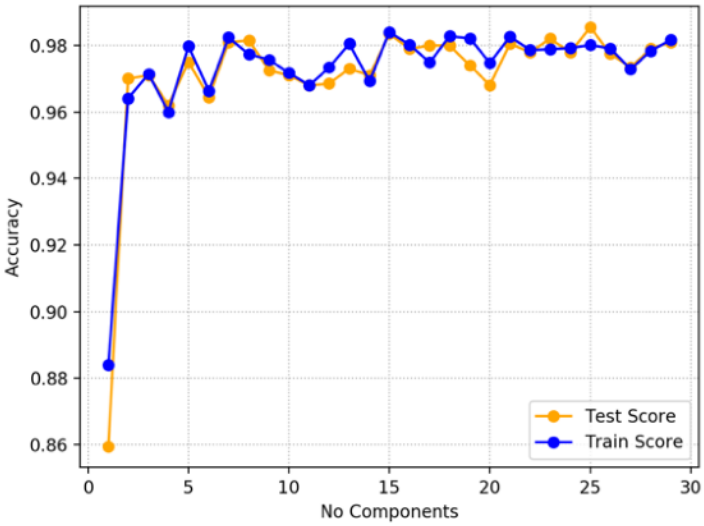
We now fit our PCA using our training set, then once we learn our projection from the training set we transform both our training and test set and use them to train and test a neural network. We expect to see that our optimal components would result in a high relative accuracy and below that number of components we would get lower accuracy since we would be having a high information loss, while the optimal number of components provide enough information for prediction.

Neural Network Architecture:

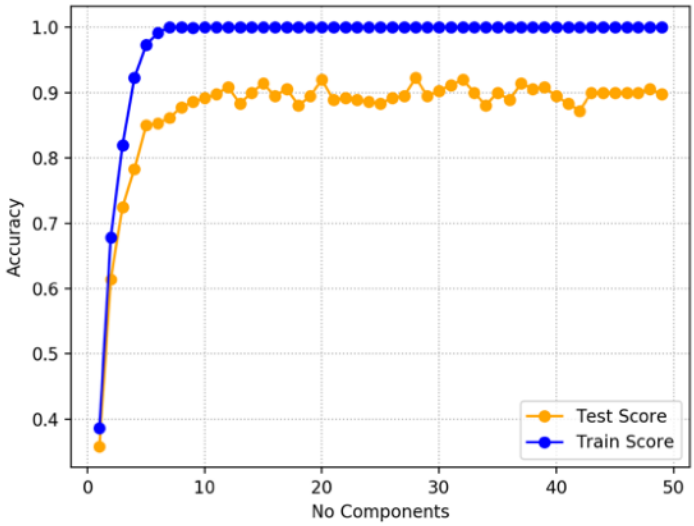
2 hidden layers, 20 nodes each

Activation function: 'relu'

Cyber Security



Letters



From the results shown above you can see that for the cyber security dataset, around 3 components were enough to prove a relatively high accuracy for the model. For the letters data set you can see that around 32 components were enough to provide a relative high accuracy for our predictions. This builds off nicely from our PCA results the studies the number of components that minimizes the mean squared error and levels off for cumulative explained variance ratio.

Classification Report Cyber Security with Optimal Component number (3):

Class	Precision	Recall	F1
Attack	0.99	0.93	0.96
Normal	0.79	0.97	0.87

Classification Report Letters with Optimal Component number (37):

Class	Precision	Recall	F1
0	0.94	0.94	0.94
1	0.89	0.94	0.91
2	1	0.95	0.97
3	0.70	0.93	0.80

4	0.89	0.89	0.89
5	1.00	0.93	0.96
6	0.95	0.92	0.93
7	0.89	0.94	0.91
8	0.88	0.78	0.83
9	0.89	0.82	0.86

From the classification report we could see a good overall F1 score that allows us to classify the data using a reduced number of dimensions. However the reduction for the cyber security is still comparable to with non reduced data. However the run time is significantly less.

Using Clustering Transformation as Neural Network features for prediction: