Assignment 3 (Unsupervised Learning and Dimensionality Reduction)

Cindy Nyoumsi
snyoumsi@gatech.edu

***Abstract*—** In this report, I will be implementing two unsupervised learning algorithms, namely K-means clustering and expectation maximization. I will also implement four dimensionality reduction algorithms, namely PCA, ICA, Randomized Projections and Lasso Selection. I will then combine these algorithms and use these new features created on the same datasets as in my previous assignment to see if it improves the performance of the Neural Network predictions on my classification problems.

1 DATASETS DESCRIPTION

I will be exploring using the two datasets described below to implement various clustering models as well as feature selection models before using Neural networks on the new features and clusters to resolve the classification problems also outlined below;

1.1 College vs Non-College Graduates Classification Description

For the first classification problem, I will be trying to classify adults into *college graduates* vs *non-college graduates* using census demographic data. This might be useful/interesting in a customer segmentation analysis where as a data science consultant, I want to help a college graduate career service company optimize their marketing investment by focusing only on college graduates in a large pool of potential clients.

1.2 Risky vs Non-Risky Lenders Classification Description

For the second classification problem I will be trying to classify adults into *risky* vs *non-risky lenders* using demographic and payment history data. This might be useful/interesting in a customer segmentation analysis where as a data science consultant, I want to help a loan company identify how risky their lending portfolio client base is.

NOTE: I will be discussing the analysis on the implementation of these algorithms at the dataset level to maximize the clarity and ease of comprehension of my report.

2 UNSUPERVISED LEARNING & DIMENSIONALITY REDUCTION ANALYSIS ON ADULT DATASET

The first unsupervised learning model I will be running on the adult dataset is K-means clustering.

2.1 K-means Clustering

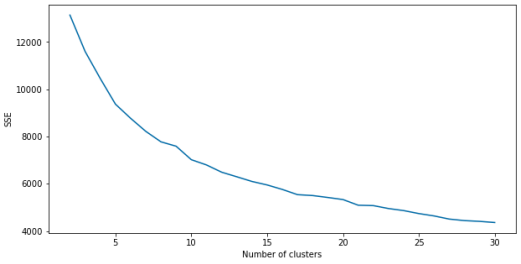
2.1.1 *Choosing K*

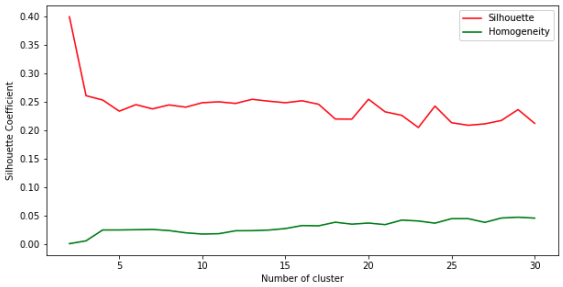
The first step towards implementing my K-means clustering on my dataset involved choosing the number k clusters I wanted formed. In order to make a data driven decision on the number of clusters to use, I decided to set up an inertia / Sum of Squared Errors (SSE) plot and use the elbow method to

identify the optimal number of clusters, but as you can see from the chart below, that was not very useful as there was no clear elbowing point unfortunately.

Since the elbow method didn't work, I decided to try using the silhouette score next to choose the number k. From my research, although the elbow method is a faster and more efficient way to identify the best k to choose, apparently the using the silhouette coefficient is more robust and produces more accurate results. Fortunately, since I know the ground truth labels of the datasets I am working with, I can combine using the silhouette scores with the homogeneity scores to identify the best k.

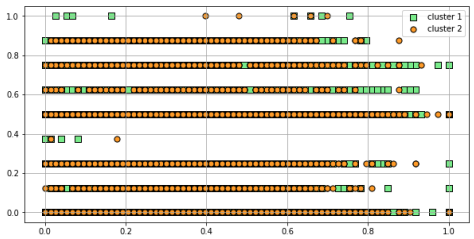
From the chart below showing the silhouette and homogeneity scores for different clusters below, I decided to land on k = 20 because of the balance between how high the silhouette and homogeneity scores were for that cluster number.





2.1.2 *Describing and Evaluating K-means Clusters*

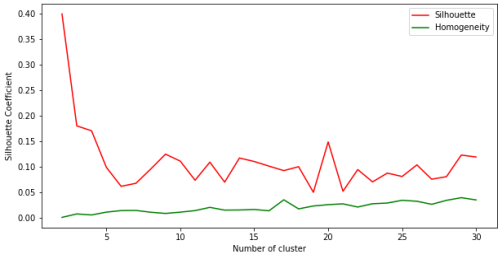
When reviewing the clusters for the Y variable produced as shown below we can see that the clusters are not distinct and show a lot of overlap between the points instead of being clearly separated as with the original data. I then used the kmeans object and fit it to feature variables before using it to predict the clusters for the y variable. Comparing the results from the k-means clusterings to the true labels, the accuracy was only 0.58. There are different reasons why this score is so low but key ones being the categorical nature of the data, which although encoded and normalized results in a great amount of information lost.



2.2 Expectation Maximization (EM)

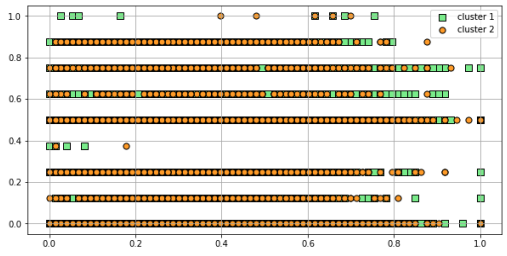
2.2.1 *Choosing number of EM K clusters*

Given the nature of the Expectation Maximization model, using silhouette score to choose K is the best option. So similar to with K-means above, I combined the analysis with



the homogeneity score and although the charts look different, they led me to choosing k = 20 in this case as well as we can see the silhouette score peak at 20.

2.2.2 *Describing and Evaluating EM Clusters*



A sample of the clusters produced for the y values of the data as shown below also shows a lot of overlap between the clusters and thus low quality of the clustering as eh clusters are not distinct.

Following the same steps as with K-means I then used the gaussian model object and fit it to feature variables before using it to predict the clusters for the y variable. Comparing the results from the EM clustering to the true labels, the accuracy from was 0.58. There are different reasons why this score is so low but key reasons being the categorical nature of the data at hand, which although encoded and normalized results in a great amount of information lost.

2.3 K-means Clustering vs Expectation Maximization Clustering

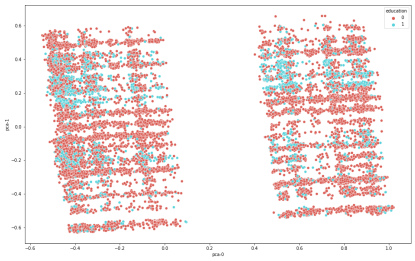
When comparing these two clustering algorithms, despite their silhouette charts taking different shapes, unsurprisingly, they both align pretty well on the number of clusters as well as how they only align fairly with the true labels of the data with both peaking at 20 clusters and only showing an accuracy of 0.58 when aligned with the true labels. Some changes I will make to the algorithms to improve performance include choosing datasets with less categorical data that needs encoding. I also would change the randomized starting points and expand it to a wider area.

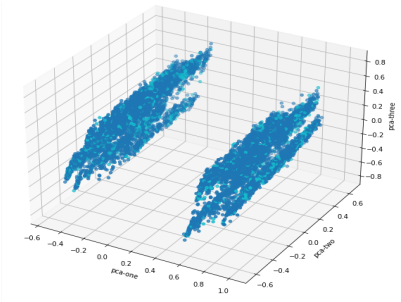
2.4 Principal Component Analysis (PCA)

The Adult dataset has a total of 13 explanatory feature variables and so after applying PCA to this dataset and running it from 2 to 13 components, It showed that the explained variance ratio was as high as 0.9 with 7 components with a reconstruction error of 0.26 which is not bad at all and reduces the dimension by a little over half from 12.

2.4.1 Description of data in new space and reconstruction error.

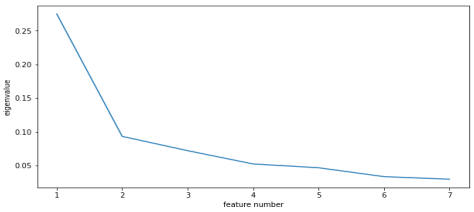
Zooming in on the two predicted components from the y values and projecting it in 2D and 3D space, we can see that we are not able to properly visualize the 3rd component as shown below and this can be attributed to the fact that the first two components alone already account for more than half of the data explained variance as will also be seen looking at the distribution of eigenvalues later. Therefore the most important components here are PCA1 and PCA2. Lastly the reconstruction errors for the projected 7 components averaged 0.267 which is a fairly good number in practice for how much information we are losing in general.





2.4.2 *Distribution of eigenvalues*

Looking at the distribution of the eigenvalues for the different components, we see that the first eigenvalue has the best fit among the data and the eigenvalues decrease with each additional eigenvalue being more and more useless.

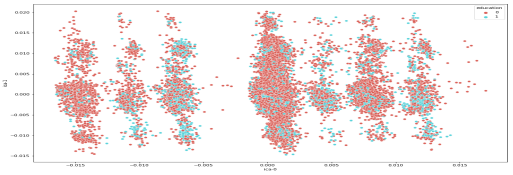


2.5 Independent Component Analysis (ICA)

Running ICA on the adult dataset, my intention was to find a new combination of the existing variables such that it creates new components that are each independent of each other. To identify how many independent components I wanted to create, similar to K-means, I used the elbow method to identify how many clusters of data could be formed, and landed on choosing 8 as that's where it elbowed.

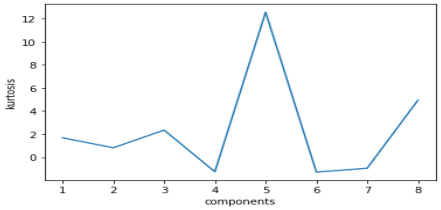
2.5.1 *Description of data in new space*

I then ran ICA on the dataset and obtained 3 different components which look as shown below when projected into a 2D and 3D space, and we can see how separated they are.



2.5.2 *Kurtosis Distribution of Components and meaningfulness of projected axes.*

Looking at the distribution of the kurtosis of the components as shown below, we see that component 5 is very super gaussian while interestingly components 4,6 and 7 around it are subgaussian. Even when comparing against the 2d projection we can see that the 5th projection carries most of the information of the underlying data unlike the 4th component which is barely visible.

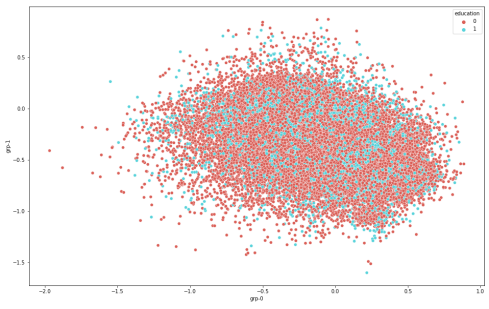


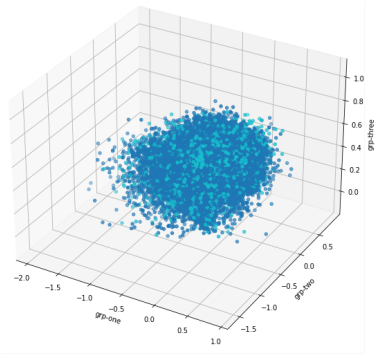
2.6 Randomized Projections (RP)

Similarly to PCA after applying RP to this dataset and running it from 2 to 13 components, It showed that the using 7 components maximized SSE using the elbow method.

2.6.1 Description of data in new space and reconstruction error.

Interestingly there is no clear separation between the components from the RP neither in 2D or 3D, and this is highly due to the fact that my data is very low dimensional and does not benefit from an algorithm like RP.





2.6.2 *Variation after re-running RP several times*

To evaluate the variation in my components when re-running RP several times, I used the silhouette score and homogeneity scores to identify changes, and noticed that both scores kept increasing at a very small rate and eventually converged after 30 runs..

2.7 Lasso Feature Selection

Running the Lasso model on the adult dataset, I was able to obtain 6 features as being most important, which is very close to the number of K components produced with the other dimension reduction algorithms that had 7 for both PCA & RP.

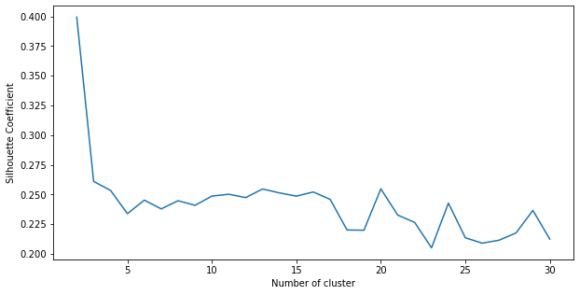
2.8 Clustering on Dimensionally Reduced Datasets

Running K-means and EM on every dimensionality reduction result demonstrated the following;

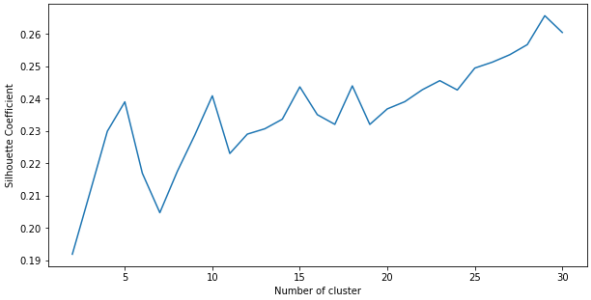
2.8.1 Differences between clusters before and after dimensionality reduction.

The silhouette scores across the board on the datasets starting from 5 components increased instead of decreasing unlike before. Especially for K-means before and after ICA. Which makes sense since ICA created more independently defined component variables that are thus easier to cluster. Also I noticed that neither K-means and EM stayed about the same after RP & Lasso interestingly. I suspect it is also because of the smaller dimensionality of the adult dataset.

K Means Before ICA



K Means After ICA



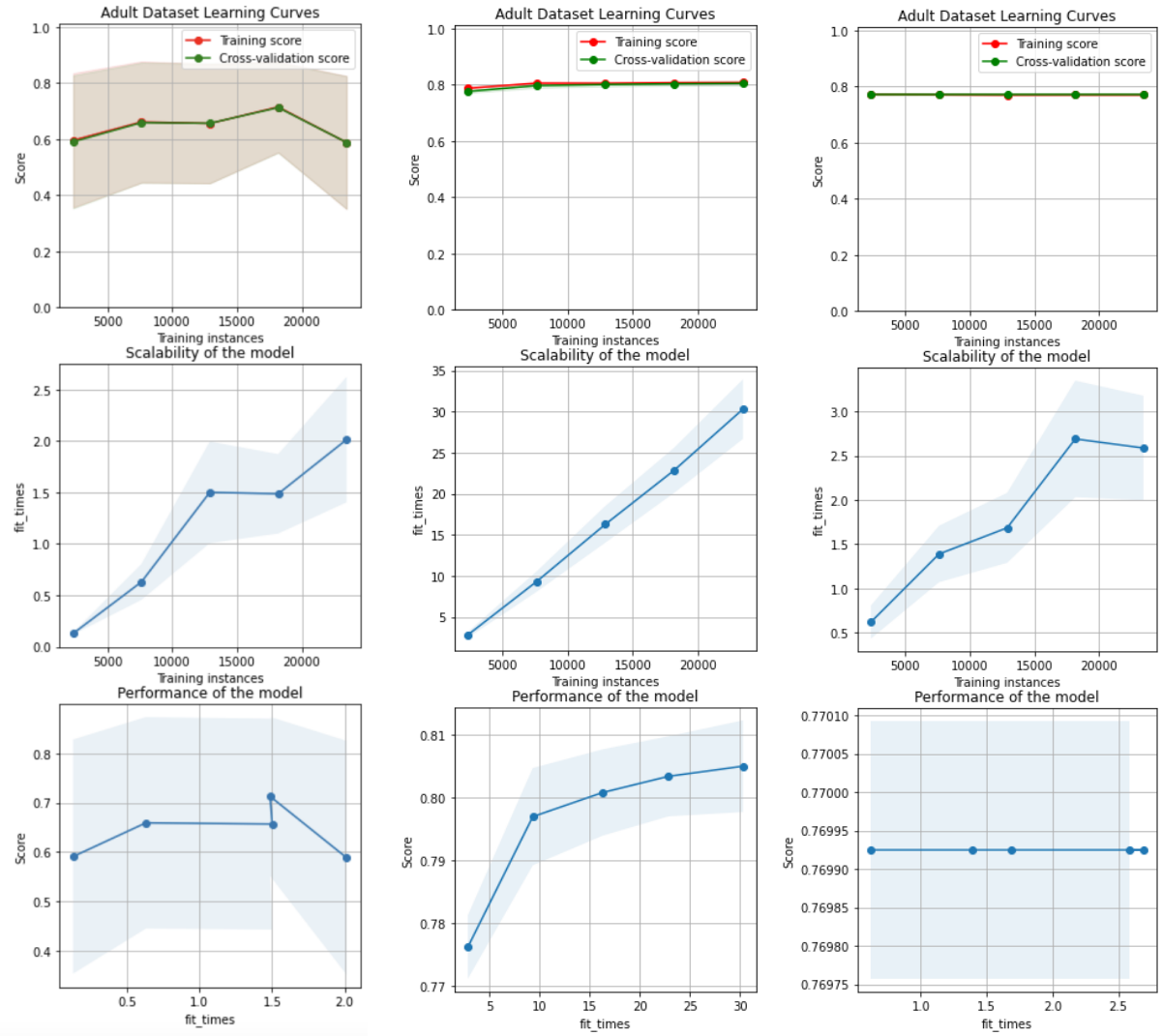
2.9 Neural Networks on Clustered and Dimensionally Reduced Datasets

Re-running Neural Networks on every K-means and EM and dimensionality reduced dataset result demonstrated the following;

2.9.1 Differences between Neural Networks before and after Clustering and dimensionality Reduction The The charts below do a great job of demonstrating the difference in performance between the NN classification results before and after each clustering and dimensionality reduction. I have summarized the best two NN performances below, where we can see that, that the NN After PCA & Kmeans did better in terms of Training vs Validation score as well as Accuracy score, but did worst in fit times, while

the NN After RP & EM I believe was overall best because it improved across the board from fit times, Training vs Validation score as well as Accuracy score, so I declare it the winner.

NN Before | NN After PCA & Kmeans | NN After RP & EM



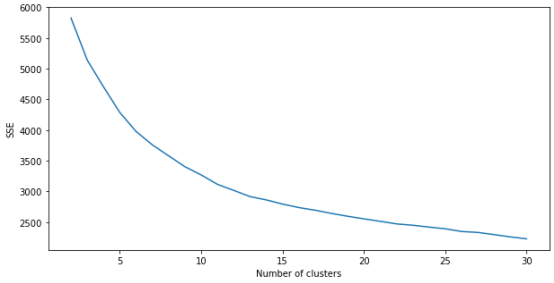
3 UNSUPERVISED LEARNING & DIMENSIONALITY REDUCTION ANALYSIS ON CREDIT DATASET

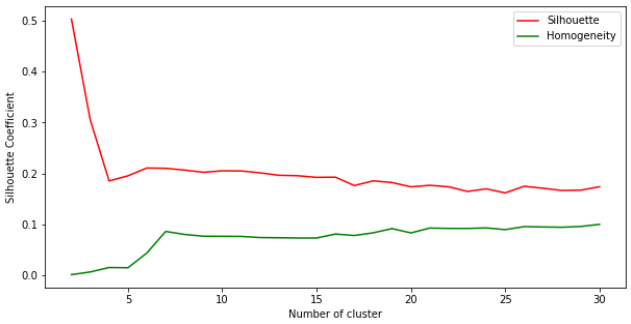
3.1 K-means Clustering

3.1.1 Choosing K

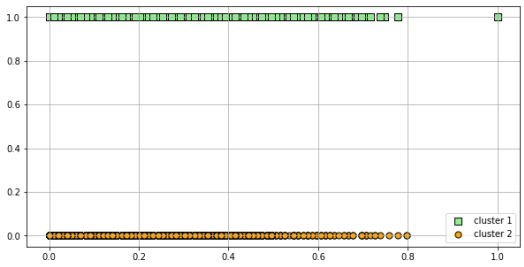
Similarly as with the adult dataset, the elbow method didn't work for the credit dataset either so I moved on to using the silhouette score to choose the number k. From the chart below showing the silhouette and

homogeneity scores for different clusters below, I decided to land on k = 7 because of the balance between how high the silhouette and homogeneity scores were for that cluster number.



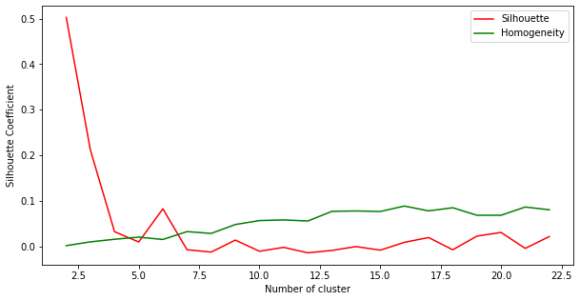


3.1.2 *Describing and Evaluating K-means Clusters*



When reviewing the clusters for the Y variable produced as shown to the left we can see that the clusters are very distinct unlike how they were with the adult dataset. They show no overlap between the points. Yet surprisingly, when comparing the results from the k-means clusterings to the true labels, the accuracy was only 0.57. There are different reasons why this score is so low but the quality of data being a combination of the categorical and float.

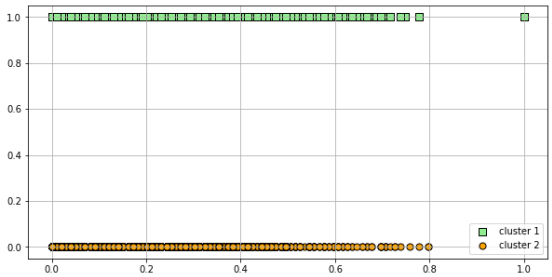
3.2 Expectation Maximization (EM)



3.2.1 *Choosing number of EM K clusters*

I continued using silhouette score to choose the best K. So similar to with K-means above, I combined the analysis with the homogeneity score and although the charts look different, they led me to choosing k = 6 in this case as well as we can see the silhouette score peak at 6.

3.2.2 *Describing and Evaluating EM Clusters*



A sample of the clusters produced for the y values of the data as shown below also shows no overlap as was the case with K-means.

Following the same steps as with K-means I then used the gaussian model object and fit it to feature variables before using it to predict the clusters for the y variable.

Comparing the results from the EM clustering to the true labels, the accuracy was also 0.57.

2.3 K-means Clustering vs Expectation Maximization Clustering

When comparing these two clustering algorithms, despite their silhouette charts taking different shapes, unsurprisingly, they both align pretty well on the number of clusters as well as how they only align fairly with the true labels of the data with peaking at 7 and 6 clusters which is very close and only showing an accuracy of 0.57 each when aligned with the true labels. Some changes I will make to the algorithms to improve performance include choosing datasets with less categorical data that needs encoding or choose more uniform floating or numerical data points.. I also would change the randomized starting points and expand it to a wider area.

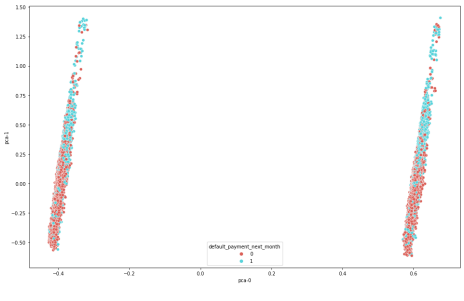
2.4 Principal Component Analysis (PCA)

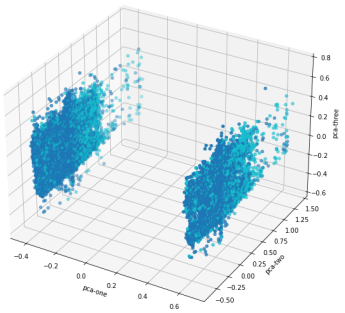
The Credit dataset has a total of 23 explanatory feature variables and so after applying PCA to this dataset and running it from 2 to 23 components, It showed that the explained variance ratio was as high as 0.99 with only 13 components with a reconstruction error of 0.05 even smaller than with the adult dataset which is due to the size that we are working with a higher dimension dataset and so have more information to gain than lose within components.

2.4.1 Description of data in new space and reconstruction error.

Zooming in on the two predicted components from the y values and projecting it in 2D and 3D space, we can see that we are not able to properly visualize the 3rd component as shown below and this can be attributed to the fact that the first two components alone already account for more than half of the data explained variance as will also be seen looking at the distribution of eigenvalues later. Therefore the most important components here are PCA1 and PCA2.

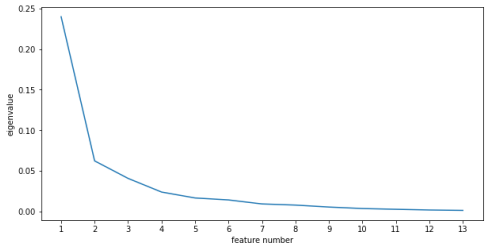
Also the reconstruction errors for the projected 7 components averaged 0.054 which is a very good number in practice for how much information we are losing in general.





2.4.2 Distribution of eigenvalues

Looking at the distribution of the eigenvalues for the different components, we see that the first eigenvalue has the best fit among the



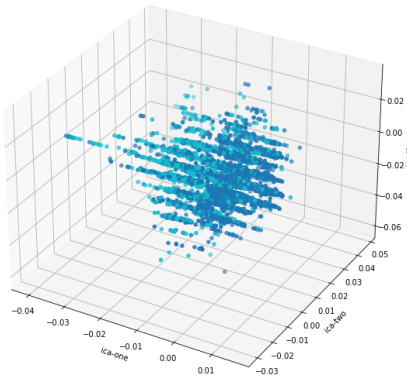
data and the eigenvalues decrease with each additional eigenvalue being more and more useless.

2.5 Independent Component Analysis (ICA)

To identify how many independent components I wanted to create, similar to K-means, I used the elbow method to identify how many clusters of data could be formed, and landed on choosing 8 as that's where it elbowed.

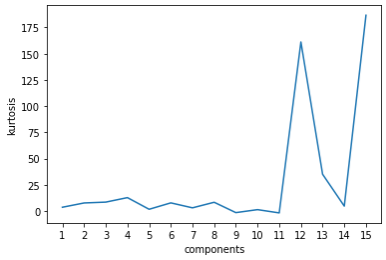
2.5.1 Description of data in new space

I then ran ICA on the dataset and obtained use the different components to predict the y values which look as shown below when projected into a 2D and 3D space, and we can see how overlapping they are across the ICA components which is not reflective of the overlying data and is caused by the underlying data having high inter covariance overall.



2.5.2 Kurtosis Distribution of Components and meaningfulness of projected axes.

Looking at the distribution of the kurtosis of the components as shown below, we see that component # 12 and # 15 are super gaussian.

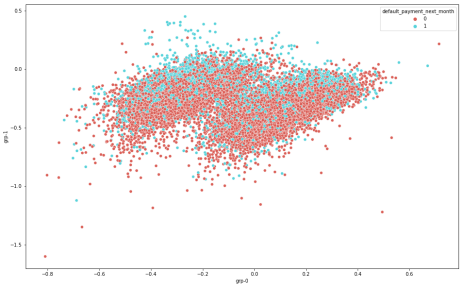


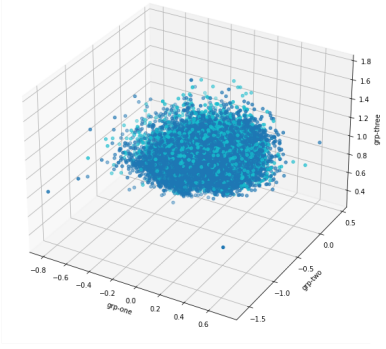
2.6 Randomized Projections (RP)

Similarly to PCA after applying RP to this dataset and running it from 2 to 23 components, It showed that the using 13 components maximized SSE using the elbow method.

2.6.1 Description of data in new space and reconstruction error.

Interestingly there is no clear separation between the components from the RP neither in 2D or 3D, and this is highly due to the fact that my data is very low dimensional and does not benefit from an algorithm like RP.





2.6.2 *Variation after re-running RP several times*

To evaluate the variation in my components when re-running RP several times, I used the silhouette score and homogeneity scores to identify changes, and noticed that both scores kept increasing at a very small rate and eventually converged after 30 runs..

2.7 Lasso Feature Selection

Running the Lasso model on the Credit dataset, I was able to obtain 8 features as being most important, which is a little further away from the number of K components produced with the other dimension reduction algorithms that had 13 and 15 for PCA & RP respectively.

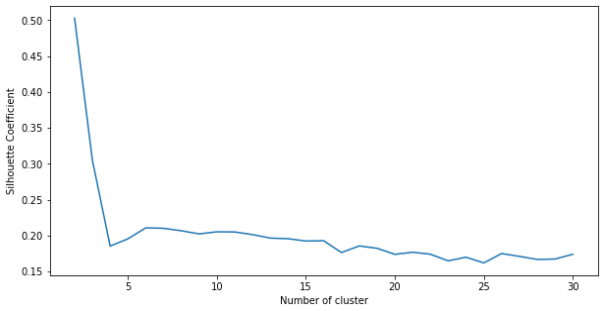
2.8 Clustering on Dimensionally Reduced Datasets

Running K-means and EM on every dimensionality reduction result demonstrated the following;

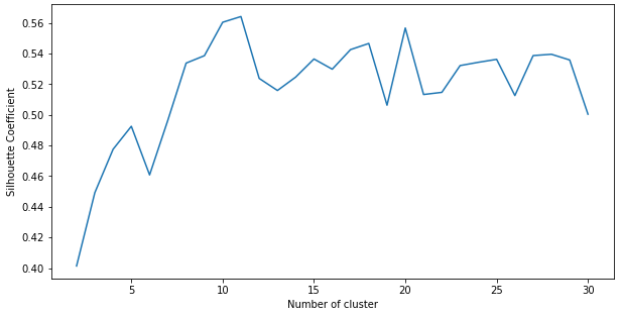
2.8.1 Differences between clusters before and after dimensionality reduction.

Unlike with the adult dataset, the silhouette scores across the board on the datasets did not all improve after combining both dimensionality reduction and clustering. The only two combinations to have shown improvement were K-Means and ICA and the most dramatic improvement, Kmeans and Feature Selection. I believe this makes sense given the high correlation between the variables of the credit dataset and the nature of Lasso algorithm.

K Means Before Lasso feature Selection



K Means After Lasso feature Selection



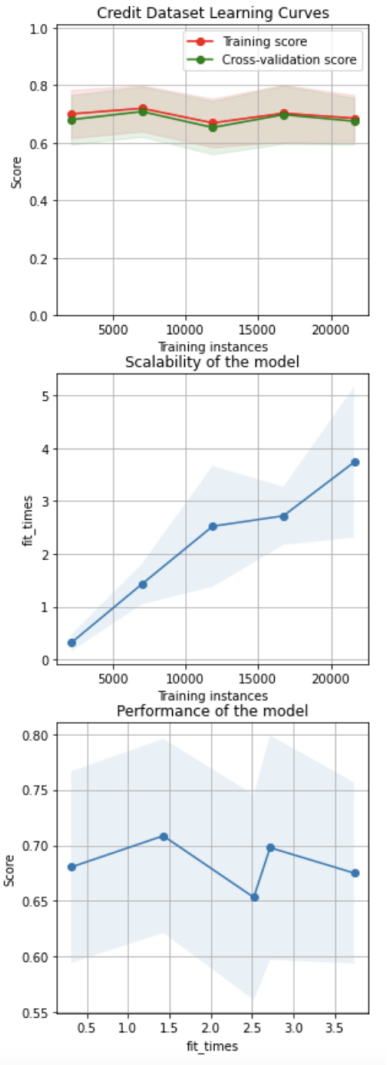
2.9 Neural Networks on Clustered and Dimensionally Reduced Datasets

Re-running Neural Networks on every K-means and EM and dimensionality reduced dataset result demonstrated the following;

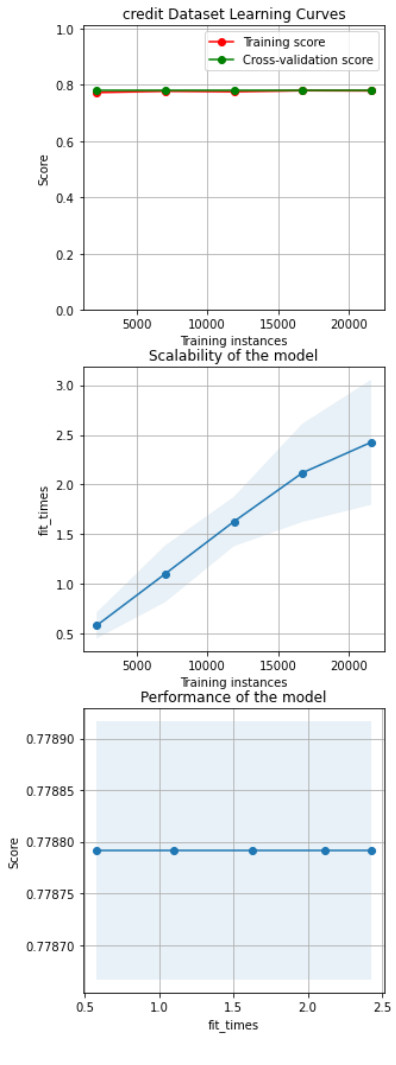
2.9.1 Differences between Neural Networks before and after Clustering and dimensionality Reduction The charts below do a great job of demonstrating the difference in performance between the NN classification results before and after each clustering and dimensionality reduction. I have summarized the best two NN performances below, where we can see that, that the NN After PCA & Kmeans did better in terms of Training vs Validation score as well as Accuracy score, but did worst in fit times, while the NN After ICA

& EM was overall best because it improved across all three dimensions on board from fit times, Training vs Validation score as well as Accuracy score, so I declare it the winner.

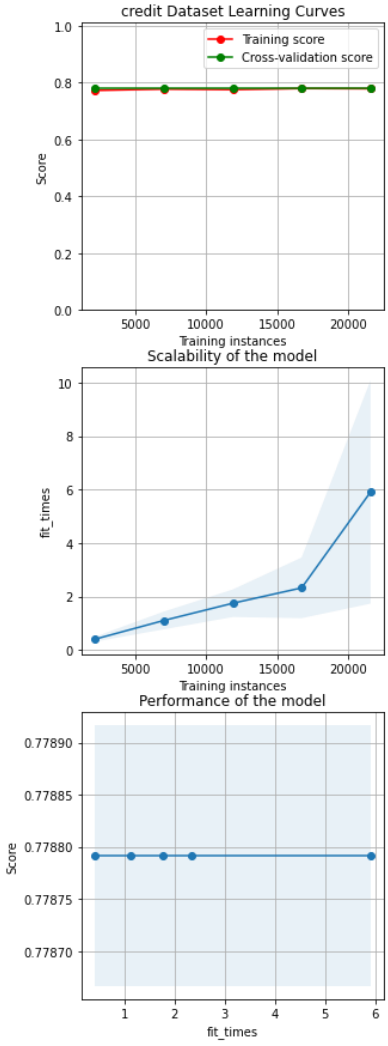
NN Before



NN After ICA & EM



NN After PCA & KMEANS



4 REFERENCES

1. UCI Adult Data Set: https://archive.ics.uci.edu/ml/datasets/Adult
2. UCI Credit Card Data Set from Taiwan with history payments and binary: https://archive.ics.uci.edu/ml/datasets/default+of+credit+card+clients
3. K Means Clustering Simplified in Python https://www.analyticsvidhya.com/blog/2021/04/k-means-clustering-simplified-in-python/

4. K-Means Clustering Explained with Python Example https://vitalflux.com/k-means-clustering-explained-with-python-example/
5. Silhouette Analysis in K-means Clustering https://medium.com/@cmukesh8688/silhouette-analysis-in-k-means-clustering-cefa9a7ad111
6. Principal Component Analysis (PCA) with Scikit-learn https://towardsdatascience.com/principal-component-analysis-pca-with-scikit-learn-1e84a0c731b0
7. PCA https://sebastianraschka.com/Articles/2015_pca_in_3_steps.html#pca-vs-lda
8. PCA step by step https://www.youtube.com/watch?v=FgakZw6K1QQ
9. Independent Component Analysis (ICA) in Python using sklearn https://towardsdatascience.com/independent-component-analysis-ica-in-python-a0ef0db0955e
10. Sklearn Random Projection https://scikit-learn.org/stable/modules/random_projection.html
11. Lasso Feature Selection using Python and Sklearn for Machine Learning https://towardsdatascience.com/feature-selection-in-machine-learning-using-lasso-regression-7809c7c2771a
12.
13.
14.