CS-7641 Assignment 3 Report – Unsupervised Learning

Puneeth Kumar Chana Reddy
preddy61@gatech.edu

Abstract—**In this report, we will be exploring two clustering algorithms namely k-Means, Expectation Maximization. Four dimensionality reduction algorithms namely Principal component analysis, Independent component analysis, Random Projection, and Extra Trees Classifier on two classification data sets to analyse their performance. Furthermore, the reduced feature set from the above algorithms will be used as the input to the neural network classifier to compare and discuss their effectiveness and shortcomings.**

I. FIFA 19 COMPLETE PLAYER DATASET

This is multi class classification data set which contains 18207 entries with 89 columns. Several features are combined based on their positions into four classes namely ”Midfielder”, ”Attacker”, ”Defender” and ”Goal Keeper”. This transformation reduces the feature space to 29 columns and the predictable classes are more or less evenly distributed in the entire dataset. Furthermore, the feature spaces consists of 29 continuous variables.

A. Data set preprocessing

- The missing data in all the column were removed, as they accounted for just 0.3% of the data set.
- All the numerical features are scaled using ”StandardScaler” to bring them on to the same scale before feeding them to algorithms

II. STROKE PREDICTION DATASET

This is binary classification data set which contains 4908 rows and 11 columns. About 95% of the the data are ”no stroke” (Class 0) and the remaining are ”stroke” (Class 1) which makes this data set a highly imbalanced one. Furthermore, the feature space consists of 5 continuous, 4 categorical variables.

A. Data set preprocessing

- The missing data in the ”bmi” column were removed. They account for about 4% of the entire dataset.
- The categorical column ”smoking_status” contains ”never_smoked” and ”unknown” that were converted to ”never_smoked” based on the other features of the instances.
- The categorical columns in the feature space are One Hot Encoded for the algorithms to make fair predictions without introducing any bias.
- The numerical columns like ”bmi” and ”avg_glucose_level” are on different scales, hence all the numerical feature are scaled using ”StandardScaler” to bring them on to the same scale before feeding them to algorithms
- ”no stroke” (Class 0) is under-sampled to make it a balanced data set before applying clustering and dimensionality reduction algorithms

Both the data sets are then split into 70% training/validation set and the remaining 30% is kept aside as a testing set for checking the performance of the final model. To make better use of data, a 5 fold stratified cross validation is performed.

III. CLUSTERING - FIFA 19 COMPLETE PLAYER DATASET

A. K-Means

The elbow method showed a monotonically decreasing sum of squared error for different cluster sizes without giving much information or ”elbow” to pick the cluster size.

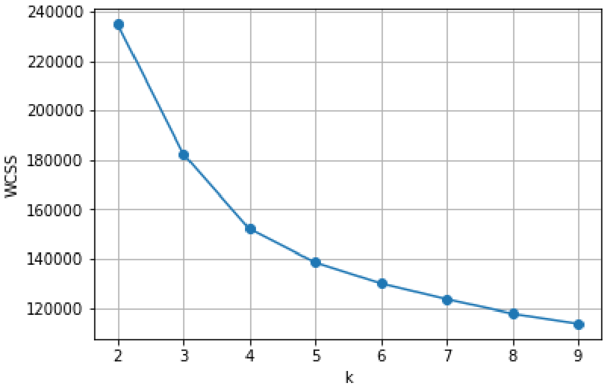


Fig. 1: Elbow Method

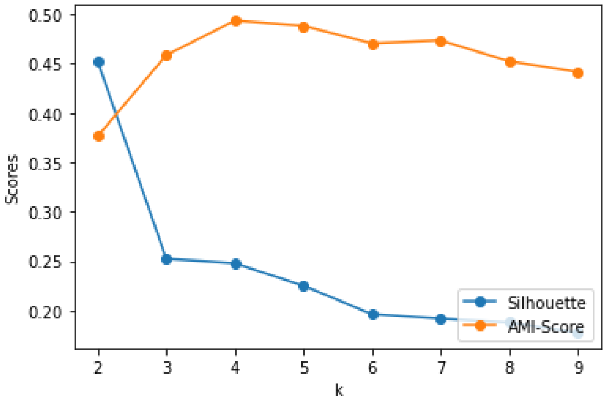
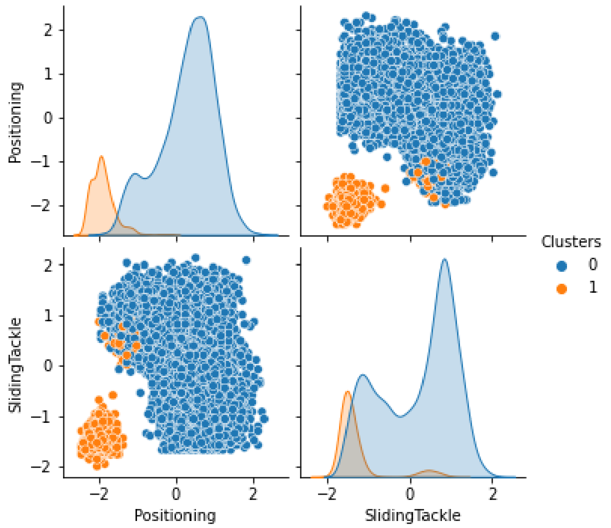


Fig. 2: Silhouette analysis

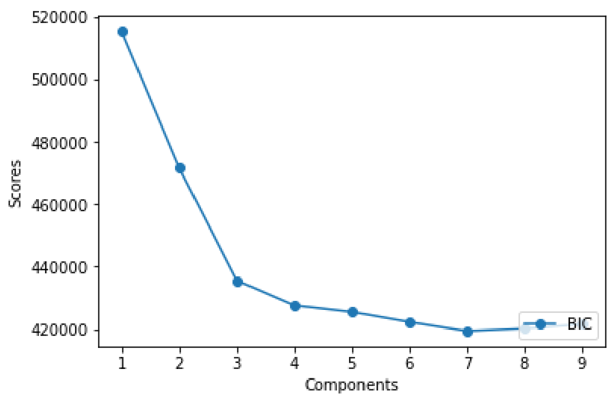
Silhouette analysis on the data set gave me a highest score of 0.45 for a cluster size of 2. Evaluating the same with adjusted mutual information score for different clusters shows that the highest score is at k=4 which matches the ground truth but not with the silhouette analysis.
As there are 4 ground truth labels in this data set but silhouette finds a highest score for only two clusters, I ran a classifier to fetch the feature importance and plotted a pair plot of the clustered data against two best features.



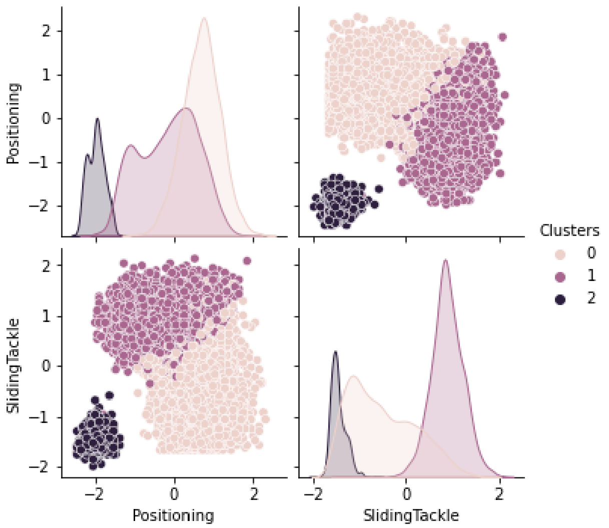
As from the figure, the samples of 3 classes were similar in terms of euclidean distance which k-means grouped together into one group. This clear distinction in the clusters explains high silhouette score for k=2. Furthermore, I analysed the cluster size by using Calinski-Harabasz and Davies-Bouldin Indices and all agree with the size of 2.

B. Expectation Maximization

GaussianMixture from sklearn with ”full” covariance type was used for the cluster analysis. BIC curve was plotted for different cluster sizes. Normally, model with lower BIC scores predict better, but this method also penalizes model with higher clusters to avoid over fitting. By this, choosing a high cluster value with low BIC score would not be good idea.



From the figure, BIC curves monotonically decreases with different slopes for different cluster sizes. The slope of the curves decreased rapidly until cluster size of 3 and gradually after that. I decided to choose a cluster size of 3 as there was not much of a change in the slope for k greater than 4. Evaluating the same with adjusted mutual information score for different cluster sizes shows that the highest score at k=3 agrees with the selected cluster size.



Like in the previous method I plotted a pair plot of the clustered data against two best features. The plot shows that GaussianMixture was able align the clusters mostly with the ground truth better than k-Means.

IV. Clustering - Stroke Prediction Dataset

A. K-Means

The elbow method showed a monotonically decreasing sum of squared error for different cluster sizes without giving much information to pick the cluster size.

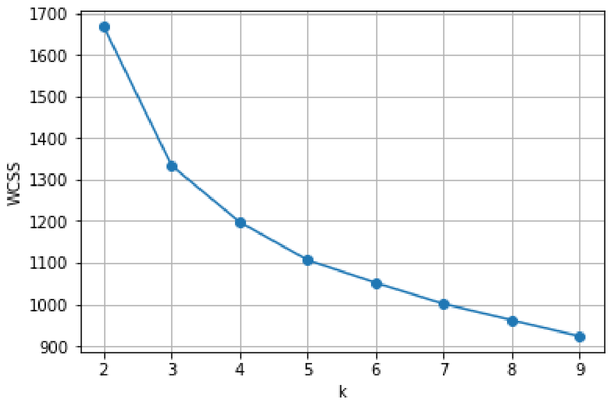


Fig. 3: Elbow Method

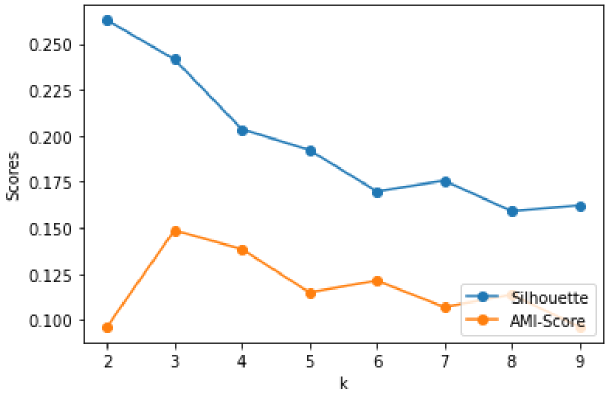
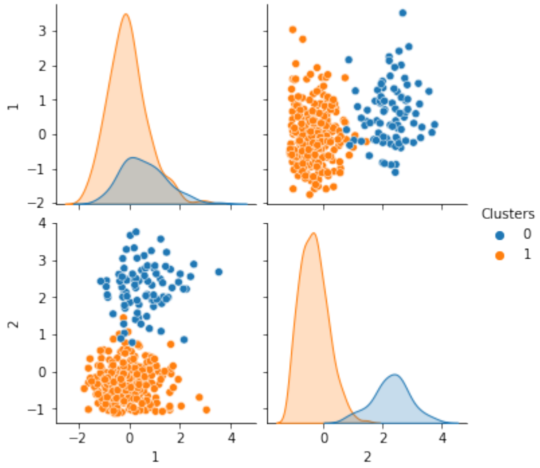


Fig. 4: Silhouette analysis



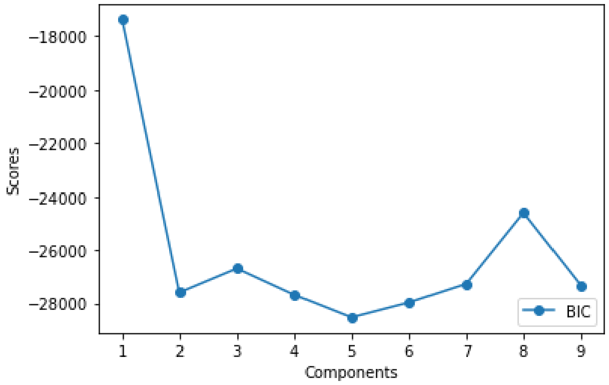
I then performed Silhouette analysis which gave me a highest score of 0.26 for a cluster size of 2. Evaluating the same with adjusted mutual information score for different clusters shows that the highest score at k=3 which doesn't agree with chosen cluster size of 2.

Also, plotting a pair plot of the clustered data against two best features also confirms the high silhouette score for a value of 2 as there are clearly distinguishable clusters.

Unlike Fifa data set, this data set has categorical variables whose euclidean distance doesn't make sense. I analysed their effect on clustering by probing the feature importances using a decision tree classifier which found top features to be 'age', 'bmi', 'avg_glucose_level' which all happened to be numerical. So a one hot encoded categorical variables did not have much impact on the clustering.

B. *Expectation Maximization*

Like in the previous data set, I used GaussianMixture from sklearn with "full" covariance type for the cluster analysis and plotted BIC curve for different cluster sizes



From the figure, BIC curve decreases sharply until cluster size of 2 and fluctuates after that. I decided to choose cluster size of 2 which has the lowest BIC score. Evaluating the same with adjusted mutual information score for different clusters shows that the highest score is at k=4 which do not agree with the ground truth labels. Also, pair plot like in k-Means shows a similar picture with two clearly defined clusters.

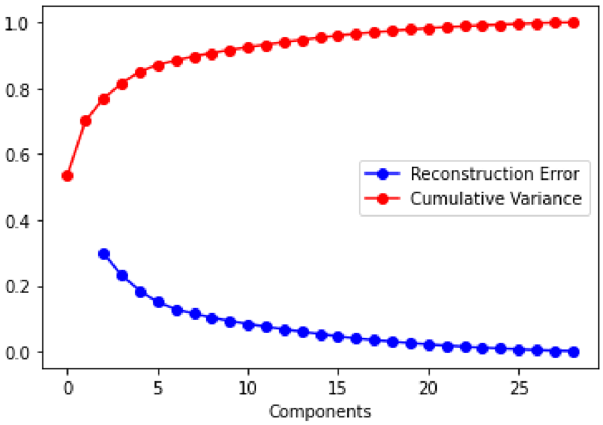
V. Dimensionality Reduction - FIFA 19 complete player Dataset

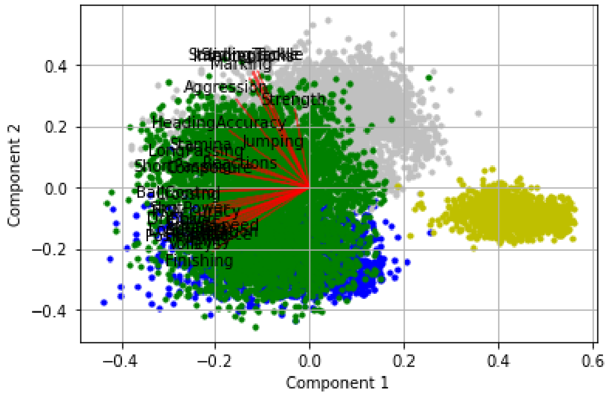
A. *Principal Component Analysis*

This data set has around 18000 samples and 29 features. I plotted the correlation matrix for the features and I noticed that there are some strongly correlated features like "StandingTackle" and "Marking" that could be linearly transformed to reduce the feature space and improve the training time of the model.

PCA linearly transforms original features into new components which are uncorrelated and most of the information is captured into the few principal components.

I ran PCA and plotted a cumulative variance for different principal components and also the reconstruction error. From the below figure, 15 components capture 95.4% of the variance and reduce the reconstruction error to less than 6%.

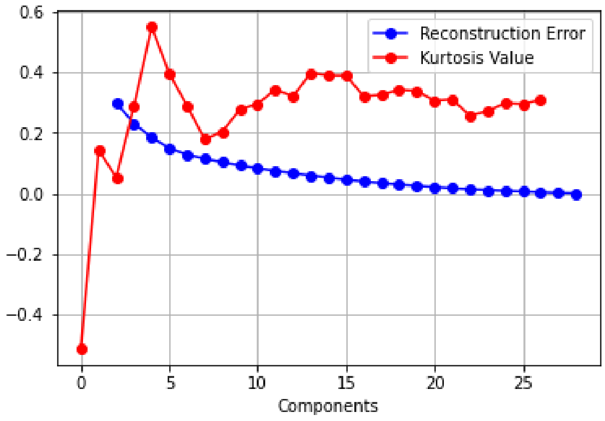


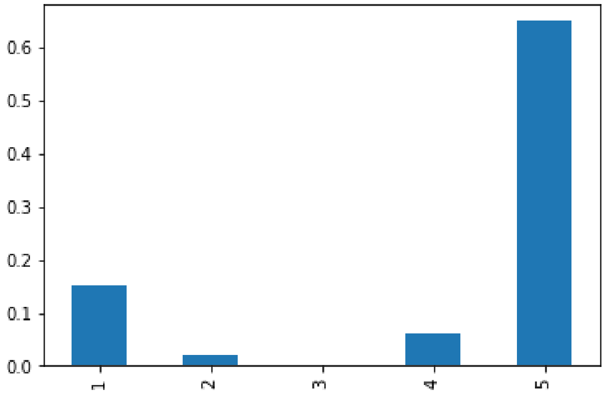


PCA bi-plot of the first two principal components showed that ”StandingTackle” and ”Marking” contributed to component 1 and are highly correlated to each other. Similarly, the length of the vector of other features shows the contribution to each of the components. Vectors aligning in the same or opposite direction were correlated positively or negatively and those orthogonal were uncorrelated.

B. Independent Component Analysis

ICA transforms the original features into new components which are statistically independent. Unlike PCA, in addition to removing the correlation between the variables it also removes higher order dependency between them. I ran FastICA from sklearn on the data and plotted a normalized mean kurtosis value for different sizes of the components.

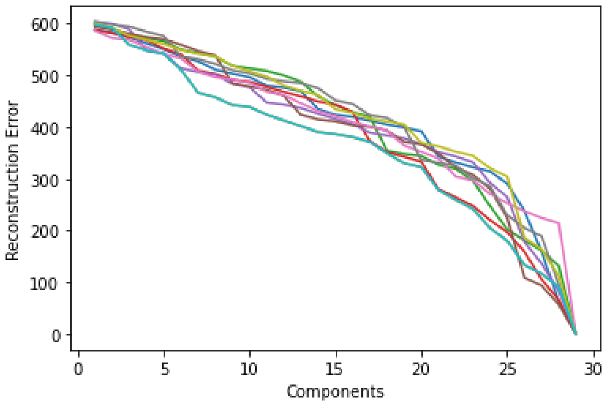




I chose a value of 6 components which has the highest mean kurtosis and reconstruction error of less than 13%. Also, I plotted the mutual information between the first component and other 5 components. It shows that they are still not independent after all, as there exists some mutual information between them.

C. Random Projection

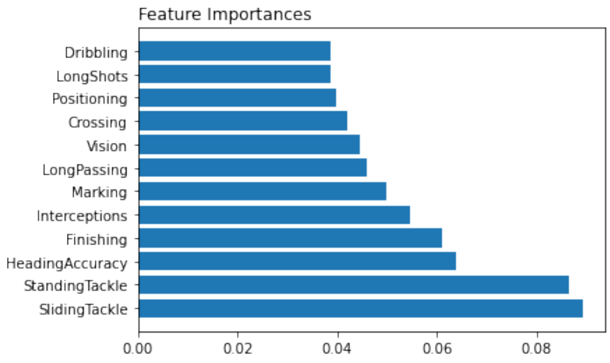
RP transforms the original features into new sub spaces which approximately preserves the pair wise distance between the samples in the data set. This is achieved by trading the error in the distances to smaller dimensions. I ran GaussianRandomProjection from sklearn on the data and plotted a reconstruction error for different sizes of the components. Algorithm was run several times with different random seeds and the mean reconstruction error for each seed was captured.



Components could not be reduced to a number which satisfies Johnson-Lindenstrauss lemma. This method would be applicable on a data set with large number features. But in this case, I chose a value of 22 components based on convergence of reconstruction error for different random seeds.

D. Extra Tree Classifier

All the above methods were transforming the original feature space to sub feature space without discarding them. On the other hand, feature selection techniques select good features from the original feature space instead of the transforming every feature. I used ExtraTreesClassifier from sklearn, an ensemble method with 100 estimators. It gave me 12 important features out of 29 from the data set.

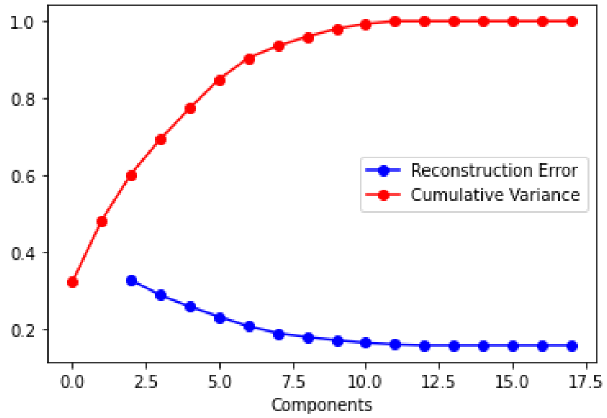


Above figure shows the features and their importances.

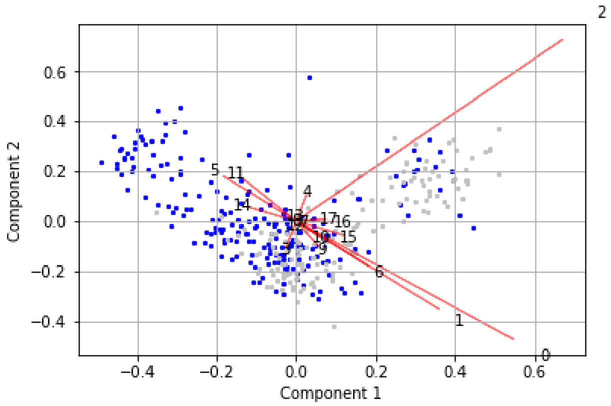
VI. Dimensionality Reduction - Stroke Prediction Dataset

A. Principal Component Analysis

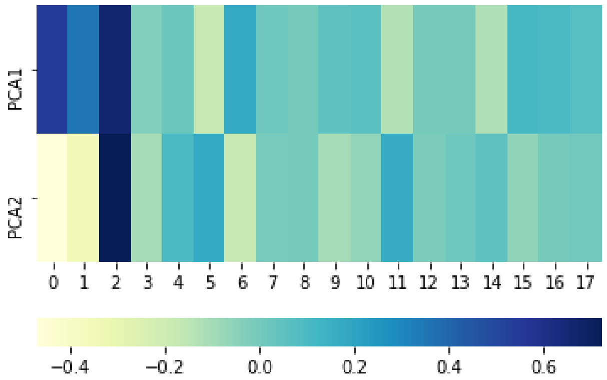
This data set has around 500 samples evenly distributed among ”stroke” and ”no stroke” classes and 18 features. PCA linearly transforms original features into new components which are uncorrelated and most of the information is captured by few principal components.
I ran PCA and plotted a cumulative variance for different principal components.



From the above figure, 93.4% of the variance is captured in just 8 components and also reduces the reconstruction error to less than 8%. Interestingly, the first 3 components account for 65% of variance.



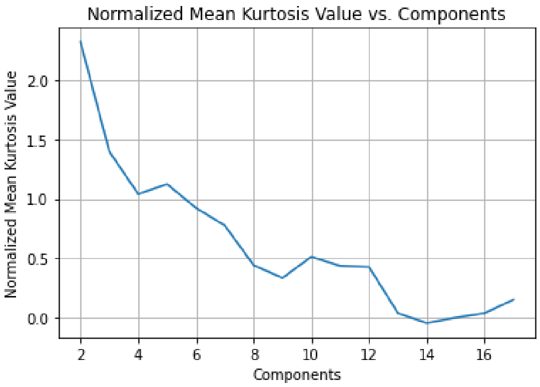
PCA biplot shows the composition of two principal components. The first two features have good contribution to component 1 and they are highly correlated. The third feature contributed more to the component 2 but not correlated to feature 1 and 2.



From the above figure, the first three numerical features contributed the most to two components compared to the categorical ones.

B. Independent Component Analysis

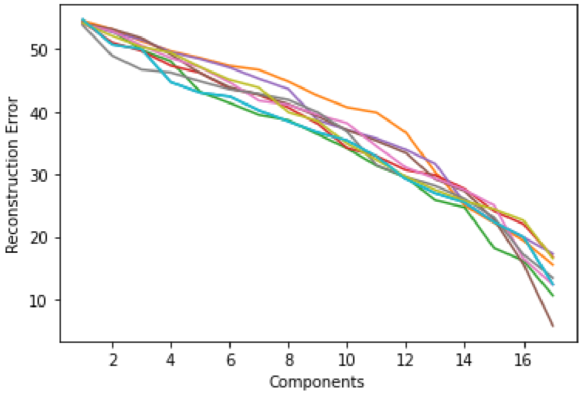
I plotted a normalized mean kurtosis value for different sizes of the components. From the below figure, I chose a value of 2 components which has the highest mean kurtosis and reconstruction error of less than 13%



The mutual information between the two components is zero in this case.

C. Random Projection

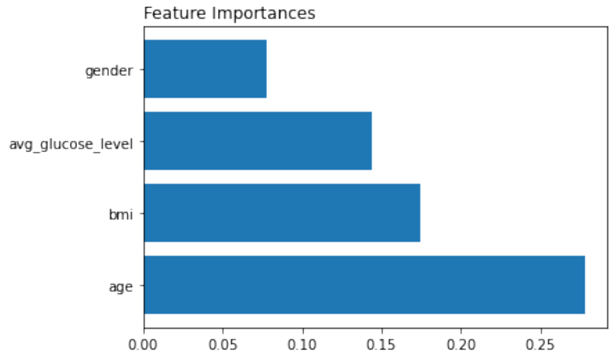
I ran GaussianRandomProjection from sklearn on the data and plotted a reconstruction error for different sizes of the components. Algorithm was run several times with different random seeds and reconstruction error was captured



Components could not be reduced to a number which satisfies Johnson-Lindenstrauss lemma. This method would be applicable on a data set with large number features. But in this case, I chose a value of 14 components based on convergence of reconstruction error for different random seeds.

D. Extra Tree Classifier

I ran ExtraTreesClassifier from sklearn, an ensemble method with 100 estimators. It gave me 4 important features out of 18 from the data set that included 3 numerical features and one categorical feature ”gender” as shown below.



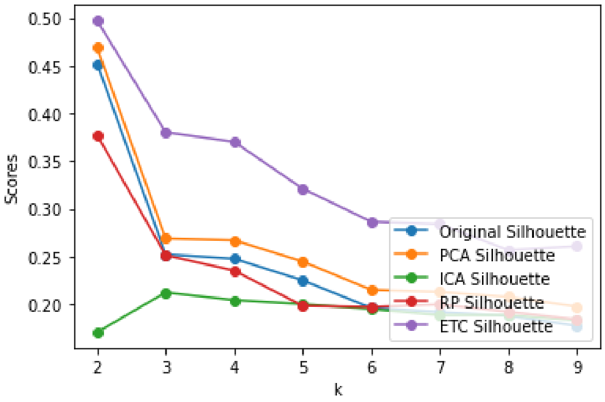
Above figure shows the features and their importances.

VII. Clustering with DR - FIFA 19 complete player Dataset

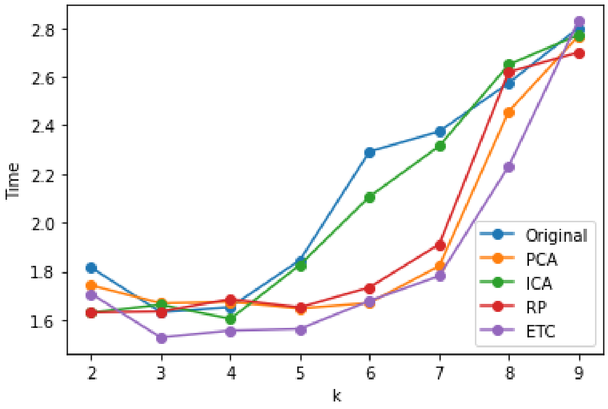
In this section, transformed data obtained after the dimensionality reduction is applied on k-Means and GMM models to evaluate their effect in terms of quality of the clusters and wall clock time they took on the reduced feature space.

A. K-Means

From the below figure, PCA and ETC improved the Silhouette score from the original data set but ICA and RP did not perform well for k=2



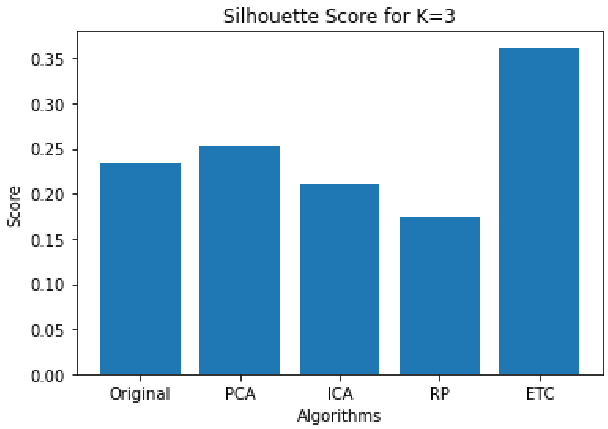
I compared the wall clock time of the clustering algorithms in reduced feature space to the original one. The plotted value are obtained by running the clustering algorithms several times and averaging the times to account for random initialization.



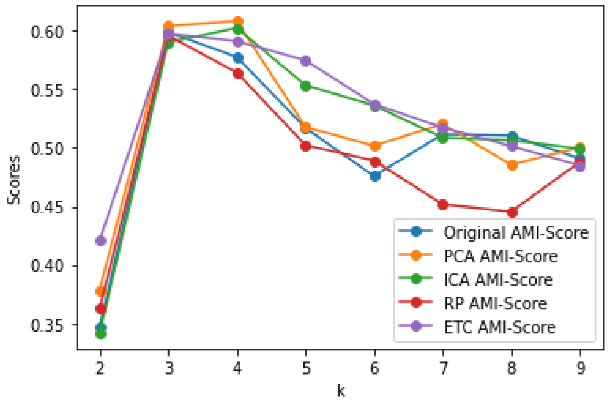
Every algorithm consistently took less time compared to the original feature spaces. Especially PCA, and ETC in addition to taking lower runtime they also improved the AMI score for k=2.

B. Expectation Maximization

I analysed the BIC curves for reduced feature space from different dimensionality reduction algorithms. All of them agreed with chosen k=3. Furthermore, I plotted the silhouette scores for a chosen cluster size of 3. ETC and PCA again scored higher than the original feature space.



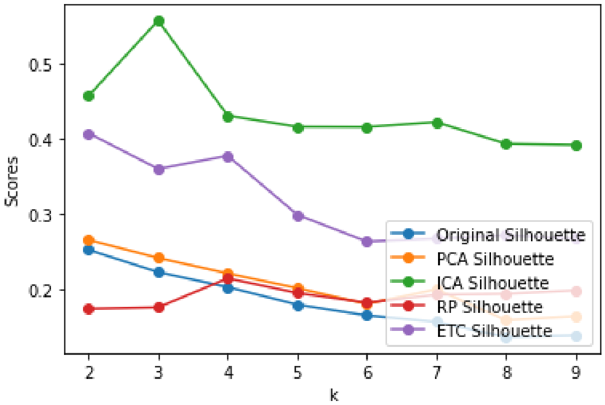
From the below AMI score figure, every dimensionality reduction maxed out at the same cluster size of 3 as we chose before using BIC analysis.



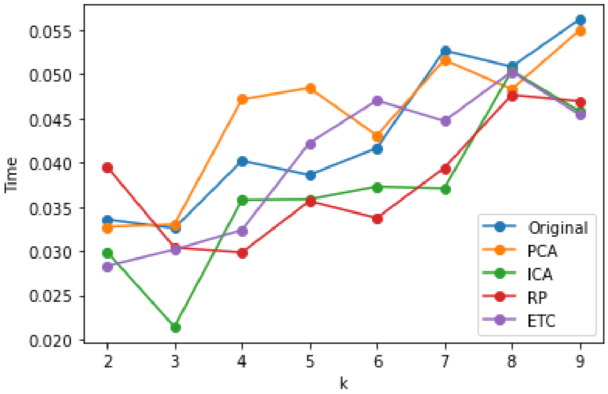
VIII. Clustering with DR - Stroke Prediction Dataset

A. K-Means

From the below figure, PCA and RP chose k=2 which is in agreement with the originally chosen cluster size but ETC shows high silhouette score at k=4 and ICA at k=3. But the absolute values of the score for ETC and ICA are higher than other algorithms. A high score for ICA can be attributed to the independent components.



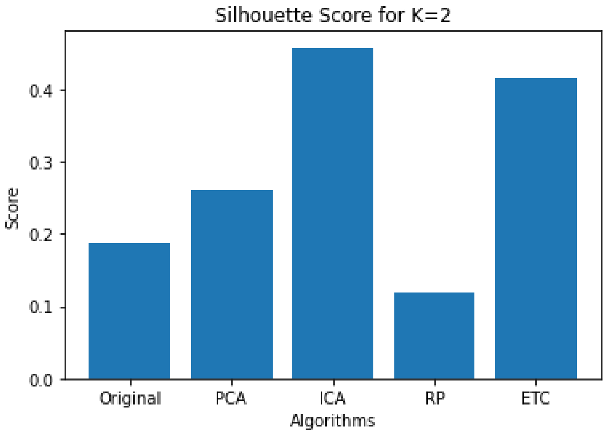
I compared the wall clock time of the clustering algorithms in reduced feature space to the original one. The plotted value are obtained by running the clustering algorithms several times and averaging the time to account for random initialization.



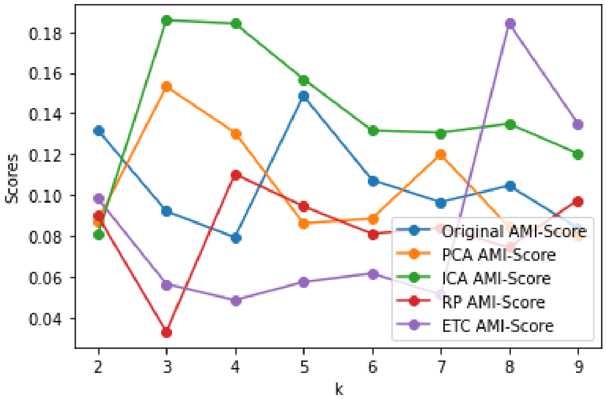
Every algorithm consistently except RP took less time compared to the original feature spaces.

B. Expectation Maximization

The BIC analysis for the reduced feature space from different dimensionality reduction algorithms agreed with k=2. ICA and ETC again scored higher than the original feature space for the same reason from the k-Means section above.



From the below figure different algorithms maxed out at different cluster sizes. PCA and ICA chose k=3, RP chose k=4, and ETC agrees with k=2 as in the BIC analysis before.

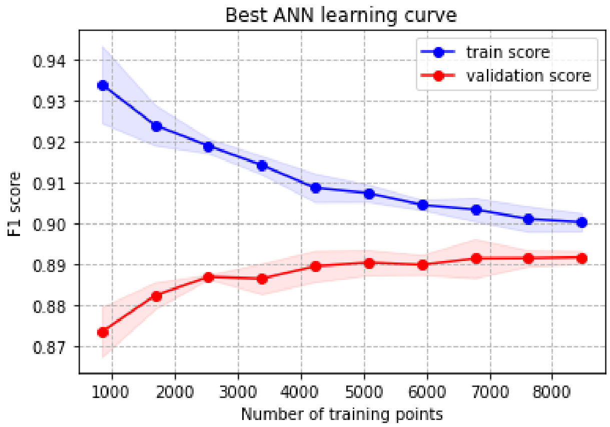


IX. Neural Network with DR - FIFA 19 complete player Dataset

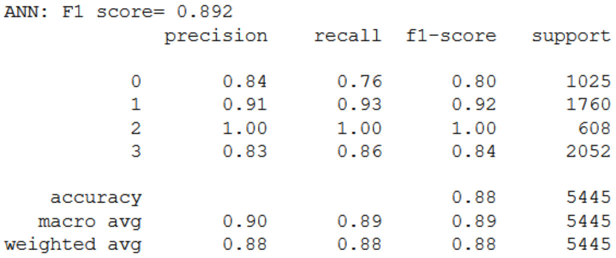
In this section, transformed data obtained after the dimensionality reduction is applied on the neural network from assignment 1 to evaluate its performance on the reduced feature space.

A. NN - Principal Component Analysis

I chose 15 components which contributed to 95% of explained variance. As the transformed feature set is reduced compared to the original feature space a exhaustive grid search of various parameters was performed to obtain *alpha=0.03* and *learning_rate=constant* and *hidden_layer_size=(15,)*. Interestingly, the *hidden_layer_size* was reduced to 15 from 25 because of reduced input layer.



Learning curve above after training the model with best parameters shows good generalization.

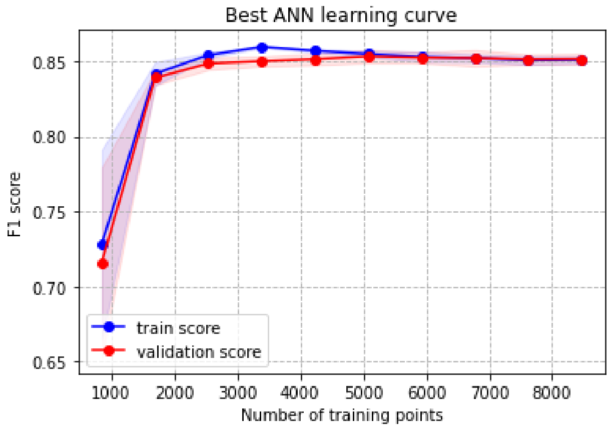


```
ANN: F1 score= 0.892
              precision    recall  f1-score   support

           0       0.84      0.76      0.80      1025
           1       0.91      0.93      0.92      1760
           2       1.00      1.00      1.00       608
           3       0.83      0.86      0.84      2052

    accuracy                           0.88      5445
   macro avg       0.90      0.89      0.89      5445
weighted avg       0.88      0.88      0.88      5445
```

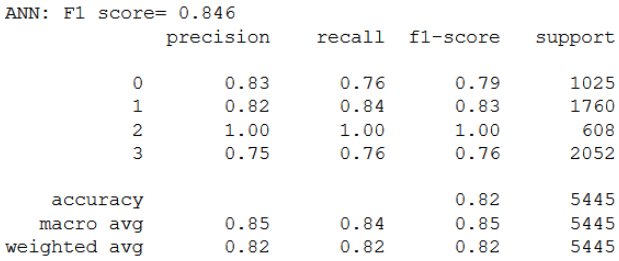
The F1 score of 89.2% is close to the value of 89.7% with the original feature space.

B. NN - Independent Component Analysis

I chose 6 components with high kurtosis value and transformed both train and test data. With the grid search and plotting model complexity curves for *hidden_layer_size* , I observed that *hidden_layer_size* kept reducing with fewer input features.In this case it was 10



Learning curve above shows that the model did not generalize well. The model could not learn from the training data as the training score was less for all sample sizes compared to PCA.

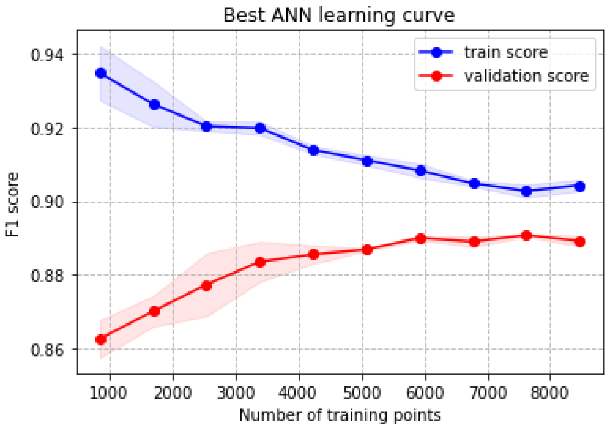


```
ANN: F1 score= 0.846
              precision    recall  f1-score   support

           0       0.83      0.76      0.79      1025
           1       0.82      0.84      0.83      1760
           2       1.00      1.00      1.00       608
           3       0.75      0.76      0.76      2052

    accuracy                           0.82      5445
   macro avg       0.85      0.84      0.85      5445
weighted avg       0.82      0.82      0.82      5445
```

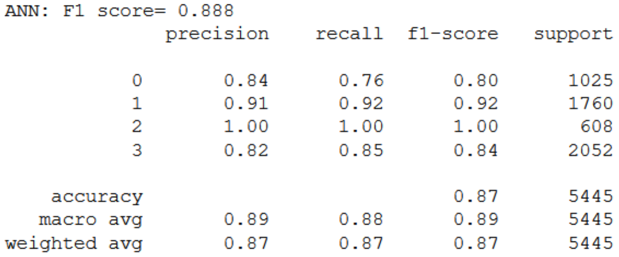
The F1 score of 84.4% is way less than the PCA. This is due to the fact that components still had some mutual information among them.

C. NN - Random Projection

I chose 22 components from random projection and transformed both train and test data. A grid search of various parameters then gave me *alpha=0.02* and *learning_rate=constant* and *hidden_layer_size=(20,)*. Also, the *hidden_layer_size* was reduced to 20.



Learning curve above after training the model with best parameters shows good generalization.

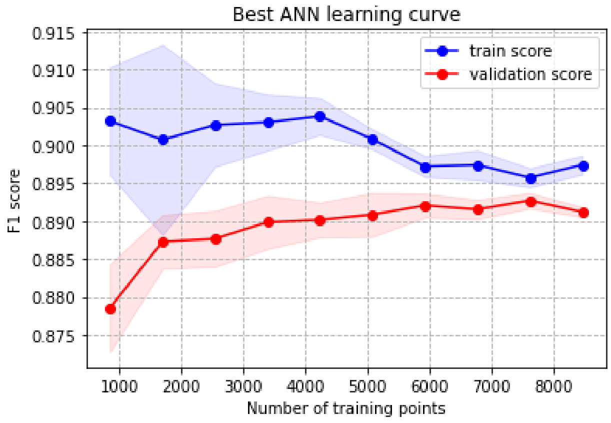


```
ANN: F1 score= 0.888
              precision    recall  f1-score   support

           0       0.84      0.76      0.80      1025
           1       0.91      0.92      0.92      1760
           2       1.00      1.00      1.00       608
           3       0.82      0.85      0.84      2052

    accuracy                           0.87      5445
   macro avg       0.89      0.88      0.89      5445
weighted avg       0.87      0.87      0.87      5445
```

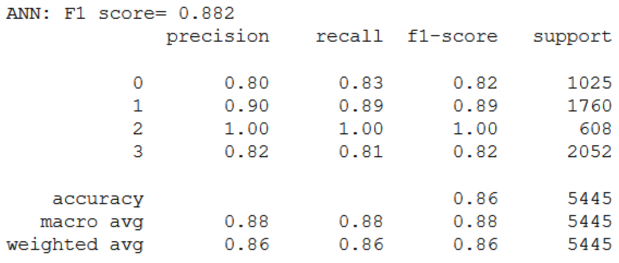
The F1 score of 88.8% is close to the value of 89.7% with the original feature space. Reducing the number of components made the model perform worse because of more distortion between the samples in the reduced feature space. This performed as good as PCA but with lot more components than PCA. I also, increased the number of component to 29 equal to number of features, the F1-Score then increased to 89.7% same as original feature space. The components acted as actual features as RP was not able to reduce to smaller sub space with out satisfying the Johnson-Lindenstrauss lemma.

D. NN - Extra Tree Classifier

This feature selection method found 12 prominent features from original feature space of 29. A grid search of various parameters then gave me *alpha=0.01* and *learning_rate=constant* and *hidden_layer_size=(10,)*. Also, the *hidden_layer_size* was reduced to 10.



The training and validation scores are less compared to PCA and RP because of feature removal. Information is lost because of discarded features.



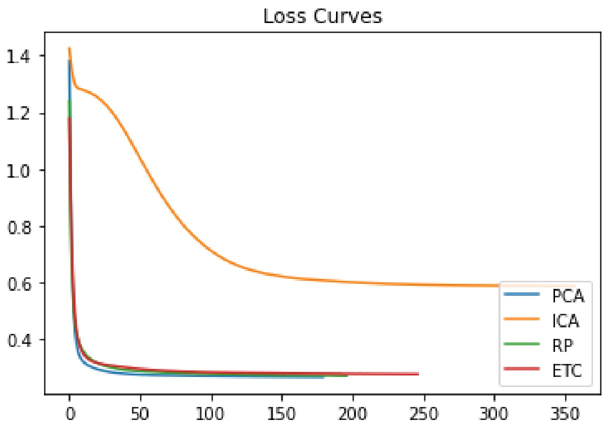
```
ANN: F1 score= 0.882
              precision    recall  f1-score   support

           0       0.80      0.83      0.82      1025
           1       0.90      0.89      0.89      1760
           2       1.00      1.00      1.00       608
           3       0.82      0.81      0.82      2052

    accuracy                           0.86      5445
   macro avg       0.88      0.88      0.88      5445
weighted avg       0.86      0.86      0.86      5445
```

The F1 score of 88.2% is close to the value of 89.7% with the original feature space. Learning curve above after training the model with best parameters shows that it cannot be improved further even with more data.

E. NN Performance Comparison

Algorithm	F1-score	Train Time(sec)
Original Features	89.7%	13
PCA	89.2%	10
ICA	84.8%	12
Random Projection	88.8%	12
Extra Tree Classifier	88.2%	7

All the dimensionality reduction algorithms took lesser time compared to the neural network classifier with original feature space. Fewer features reduces network complexity thereby reducing the number of weights leading to faster convergence.

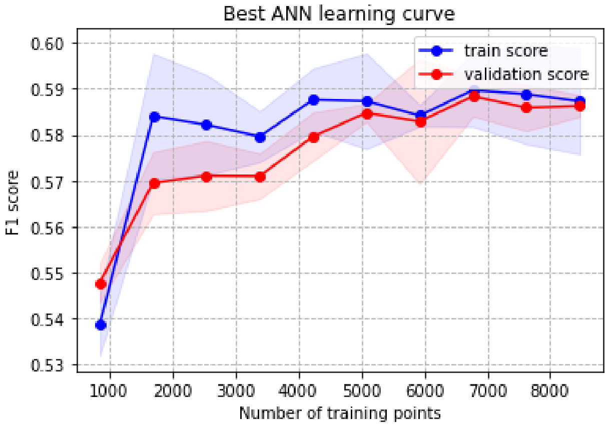


Comparing the loss curves above show that ICA could not reduce the loss as other methods which also reflects in a low F1 score compared to others.

X. Neural Network with Clustering - FIFA 19 Complete Player Dataset

A. NN - K-Means

From the k-Means section above, k value of 2 was chosen. Firstly, I added the clustered labels to the existing feature set as a new feature column.
Secondly, I used only clustered labels as my feature set and
Lastly, I used the distance of each sample to the centroids as new features which accounted to 2 new features and tuned the neural network classifier.
In the first case, there was no change in the models performance because of the additional feature.
In the second case too, a single feature of clustered labels doesn't give any information to train the model hence the score was very low.
For the third case with sample distances from the centroids as features, a grid search of various parameters was performed to obtain *alpha=0.1* and *learning_rate=constant* and *hidden_layer_size=(15,)*.



Learning curve above has low training and validation scores and providing more data would do no good here. The information in the features are not representative of the variation in underlying data set. The F1 score of 61.6% was achieved with 2 features.

B. *NN Performance Comparison*

Algorithm	F1-score	Train Time(sec)
Original Features	89.7%	13
Original+K-Means(labels)	89.6%	12
K-Means(labels)	39%	6
K-Means(Centroid distances)	61.6%	9

C. *NN - EMM*

From the EMM section above, k value of 3 was chosen. In a similar way as above

Firstly, I added the clustered labels to the existing feature set as a new feature column.

Secondly, I used only clustered labels as my feature set.

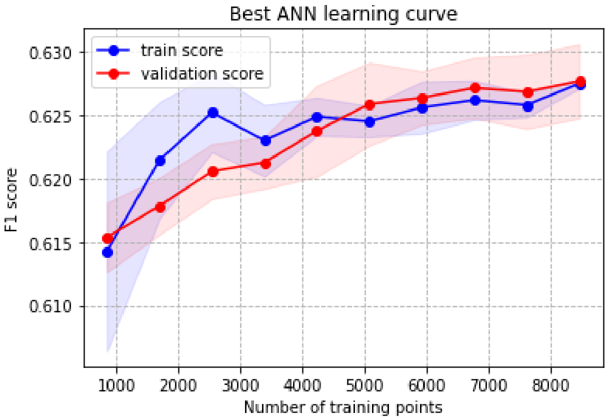
Thirdly, I used the probabilities of each sample to each of clusters as new features which accounted to 3 new features and

Lastly, I calculated the euclidean distance between samples and the mean of clusters and tuned the neural network classifier for every case.

In the first case, there was no change in the models performance because of the additional feature.

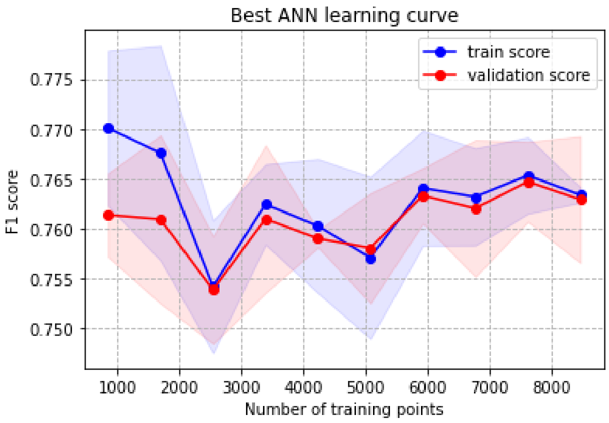
In the second case too, a single feature of clustered labels doesn't give any information to train the model hence the score was very low.

For the third case with sample probabilities for each of the cluster as features, a grid search of various parameters was performed to obtain *alpha=0.03* and *learning_rate=constant* and *hidden_layer_size=(10)*.

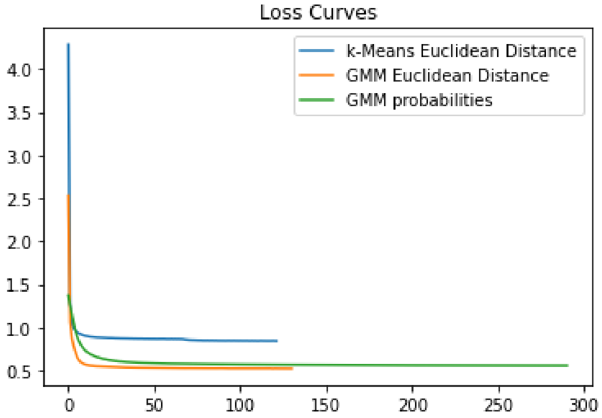


Similar argument as in k-Means is applicable here too, as there is little information in the features for the model to learn. The F1 score of 62.3% was achieved with 3 features but will not improve even with more data.

For the last case euclidean distance between samples and cluster means was used and a grid search of various parameters was performed to obtain *alpha=0.01* and *learning_rate=constant* and *hidden_layer_size=(10)*.



The training and the validation score was better than the previous case but it is still low. Comparing the loss curves of k-Means and GMM with features generated from clustering show that GMM performed relatively better that k-Means.



D. *NN Performance Comparison*

Algorithm	F1-score	Train Time(sec)
Original Features	89.7%	13
Original+GMM(labels)	89.7%	13
GMM(labels)	42%	5
GMM(Cluster probabilities)	62.3%	10
GMM(Distance to cluster means)	75.6%	10

References

[1] "Stroke-Prediction-Dataset", Kaggle Available: https://www.kaggle.com/fedesoriano/stroke-prediction-dataset. [Accessed: 01-Feb-2022]

[2] "FIFA 19 complete player Dataset," Kaggle Available: https:https://www.kaggle.com/karangadiya/fifa19. [Accessed: 01-Feb-2022]

[3] "scikit-learn documentation" Available: https://scikit-learn.org/stable/