Anastasia Psarou - Lab 3. Feature selection and extraction

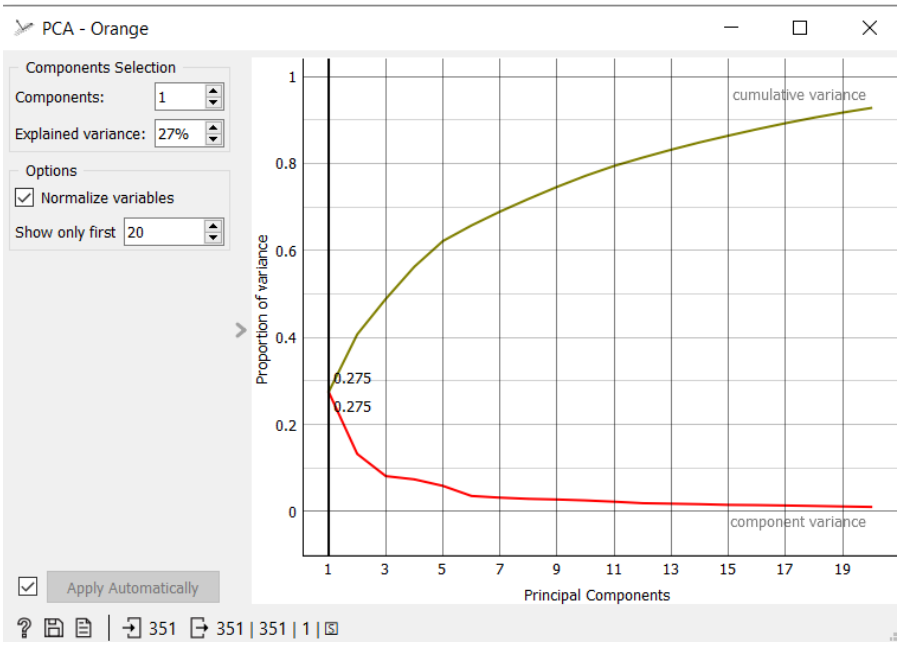
a) Select a dataset with the number of features greater than 10. Preferably more than 20 less than 50. Data should be prepared for classification task.

I selected the “Ionosphere” as it has 24 instances.

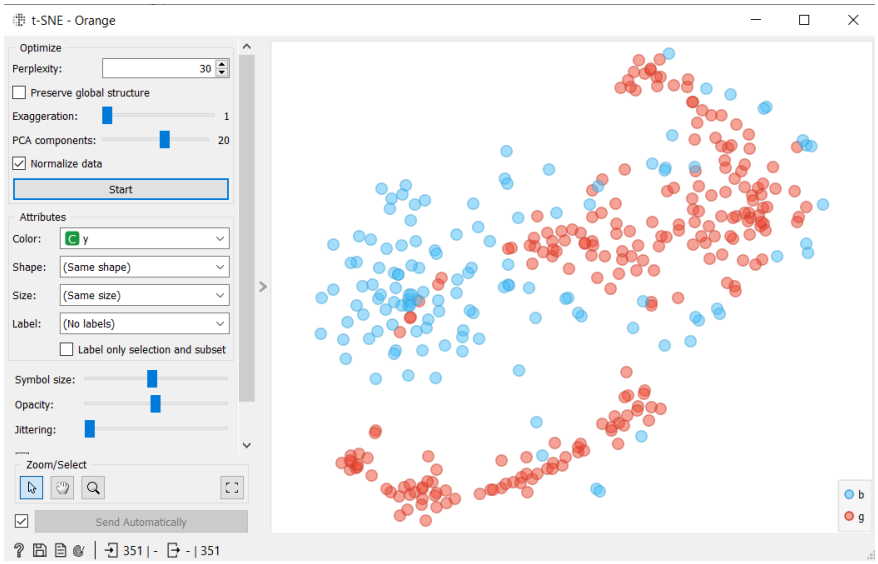
b) Read about this dataset and try to understand features.

d) Visualize the dataset, using PCA and t-SNE.

This is the PCA representation of the data:



This is the t-SNE representation of the data:

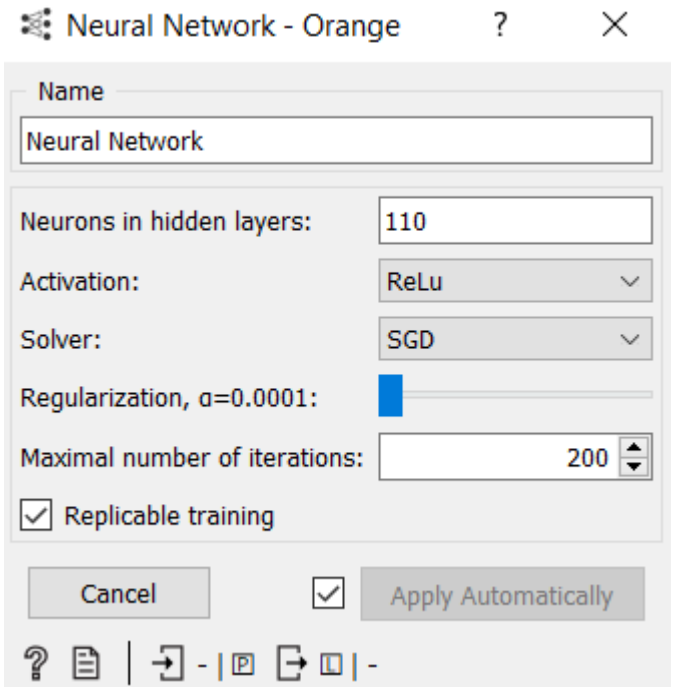


e) Use neural net classifier for finding classes. Decide about the values of the metaparameters of the NN classifier.

I aim for the best precision in the Test and Score widget. In order to achieve this, I made some tests in the Neural Network widget. This is the best I achieved

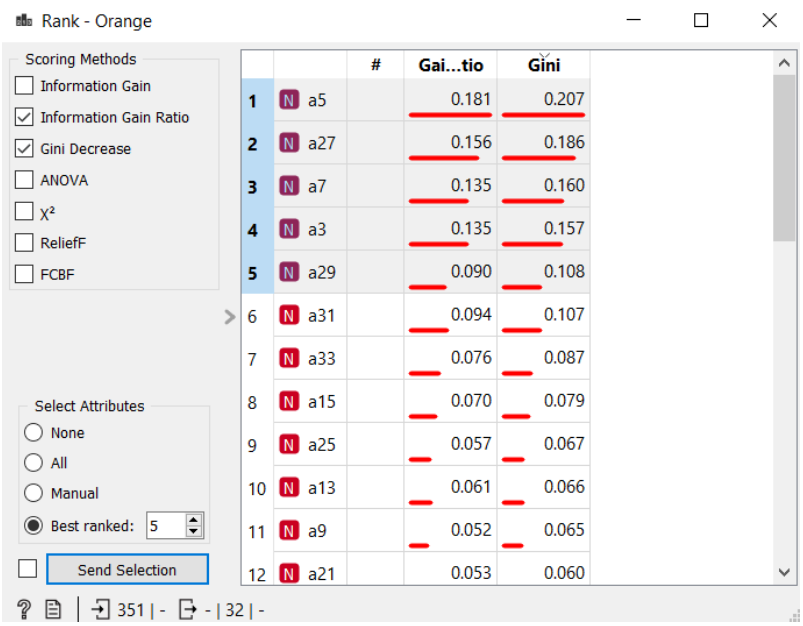
Model	Train time [s]	Test time [s]	AUC	CA	F1	Precision
Neural Network	7.023	0.129	0.936	0.892	0.889	0.894

I used the metaparameters below to achieve that:



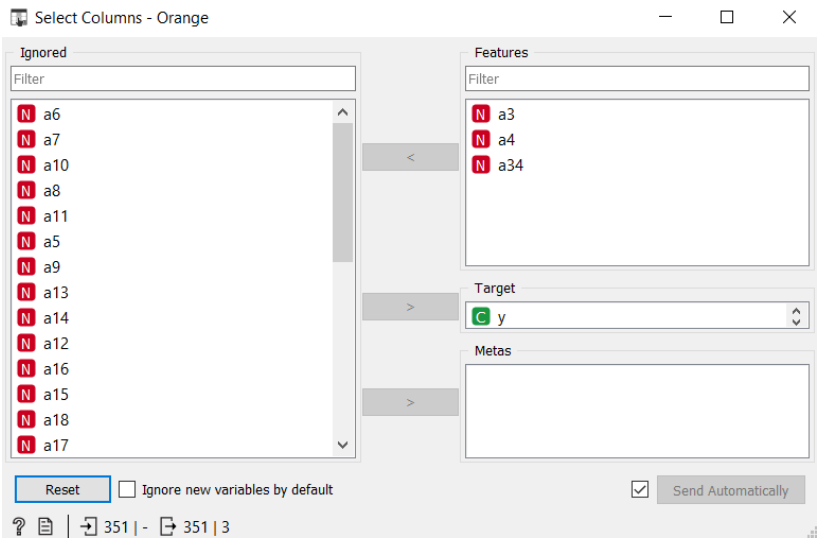
f) Based on features ranking 1) select a few best features (2-3) 2) Check how the accuracy produced by the classifier changes with increasing the number of features 3) Check how the accuracy of the classifier changes assuming first the whole set of features than decreasing the number of features removing the worst ones one by one.

1) From Rank widget we can see that the best features are:



So, I take a5, a27 and a7 as the 2-3 best features.

2) In the Select Columns widget I left only 3 features.

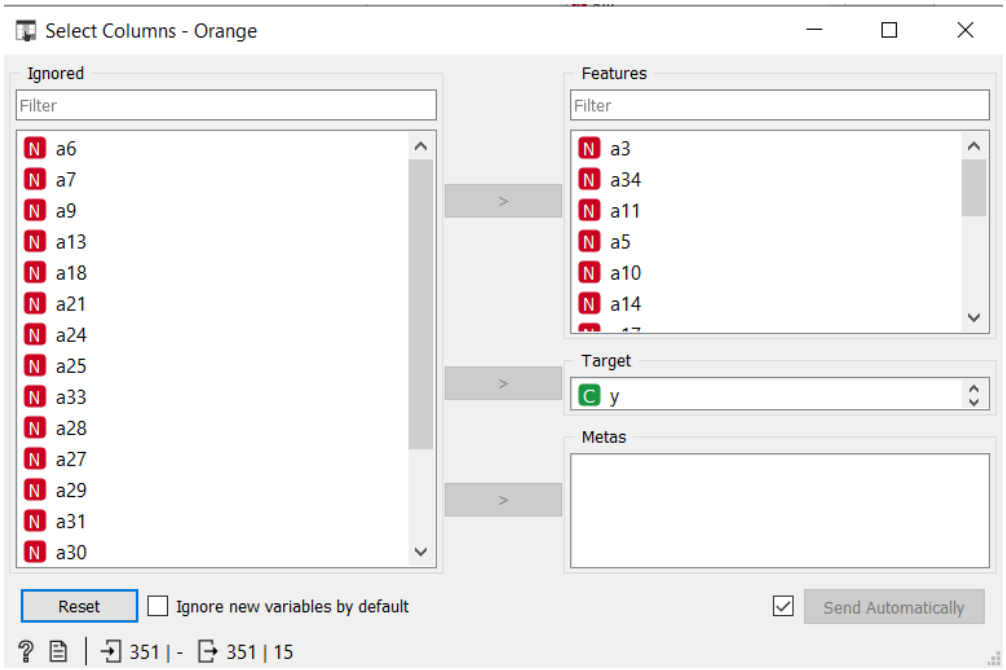


In this case we have CA = 0.821

Model	Train time [s]	Test time [s]	AUC	CA	F1	Precision	Recall
Neural Network	5.429	0.022	0.862	0.821	0.811	0.828	0.821

Then, I started increasing the number of features. The outcome was that the CA also started increasing.

For example, after adding some features:



We can see that CA has augmented.

Model	Train time [s]	Test time [s]	AUC	CA	F1	Precision	Recall
Neural Network	6.290	0.066	0.889	0.846	0.842	0.847	0.846

3) Through the Rank widget we can found the worst features and then in the Select Columns widget I diminish them one by one.

At the beginning, having all the features CA = 0.883 we have:

Model	Train time [s]	Test time [s]	AUC	CA
Neural Network	7.141	0.099	0.936	0.883

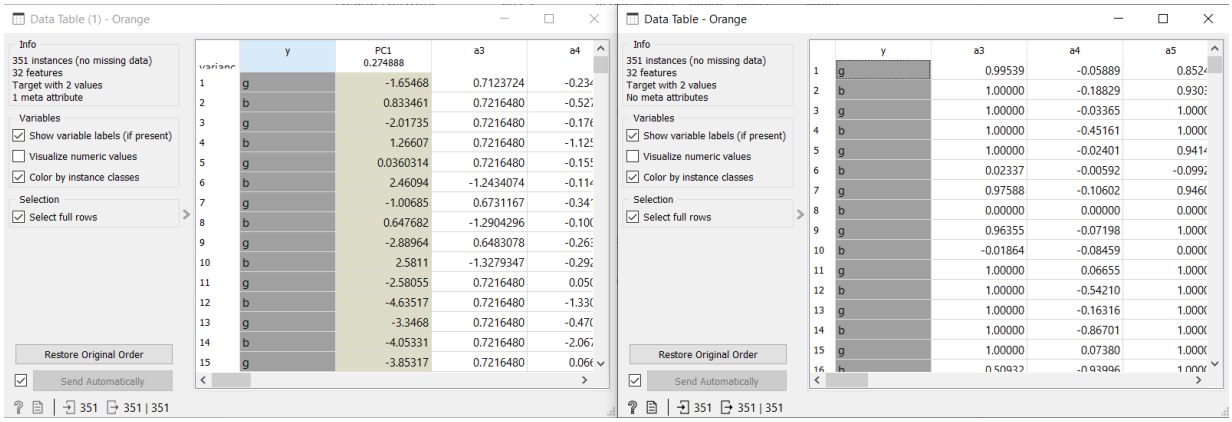
After, removing a couple of features I noticed that the CA got decreased. More specifically, after removing the 6 worst features the CA got to 0.866.

Model	Train time [s]	Test time [s]	AUC	CA	F1	Precision
Neural Network	6.729	0.105	0.931	0.866	0.864	0.865

The CA is getting decreased but with a small rate.

g) Check how the classifier behaves 1) after PCA transformation (all features), compare with that for all set of old features; 2) after decreasing the number of PCA features one by one starting from the feature with the smallest eigen value.

Comparing the Data Tables of the features before and after the PCA transformation it is clear that a lot of changes have been occurred, as the numbers have changed completely. We can see it below:

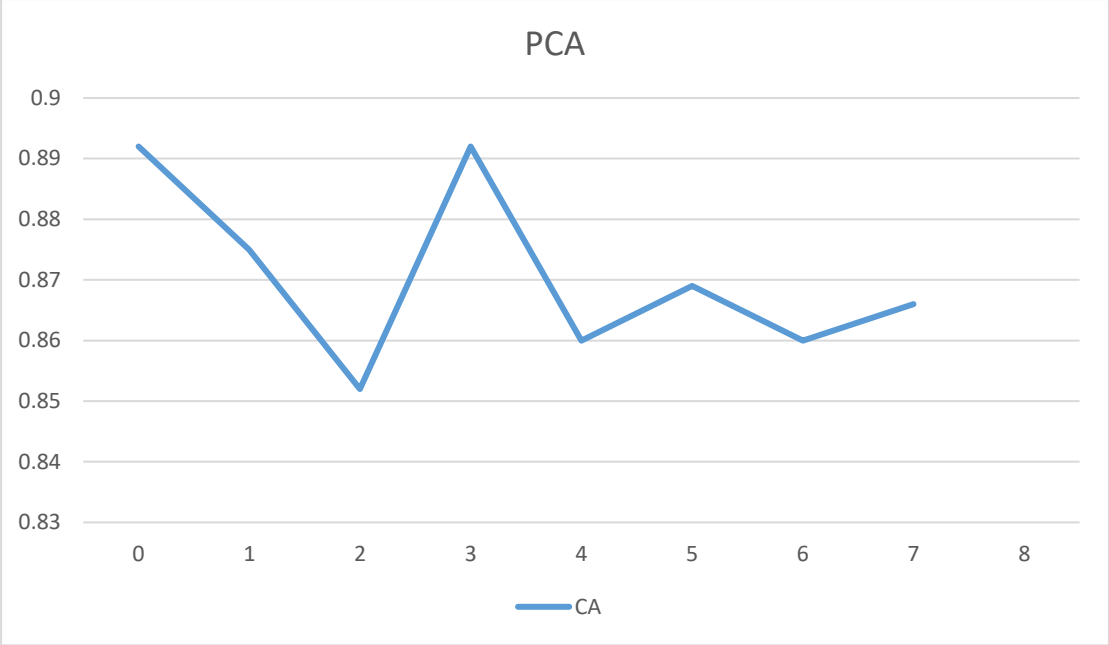


On the left are the features after the PCA transformation and on the right before this transformation.

As far as the CA value is concerned, they have the same one:

Model	Train time [s]	Test time [s]	AUC	CA
Neural Network	7.093	0.126	0.936	0.892

Model	Train time [s]	Test time [s]	AUC	CA
Neural Network	6.758	0.123	0.936	0.892

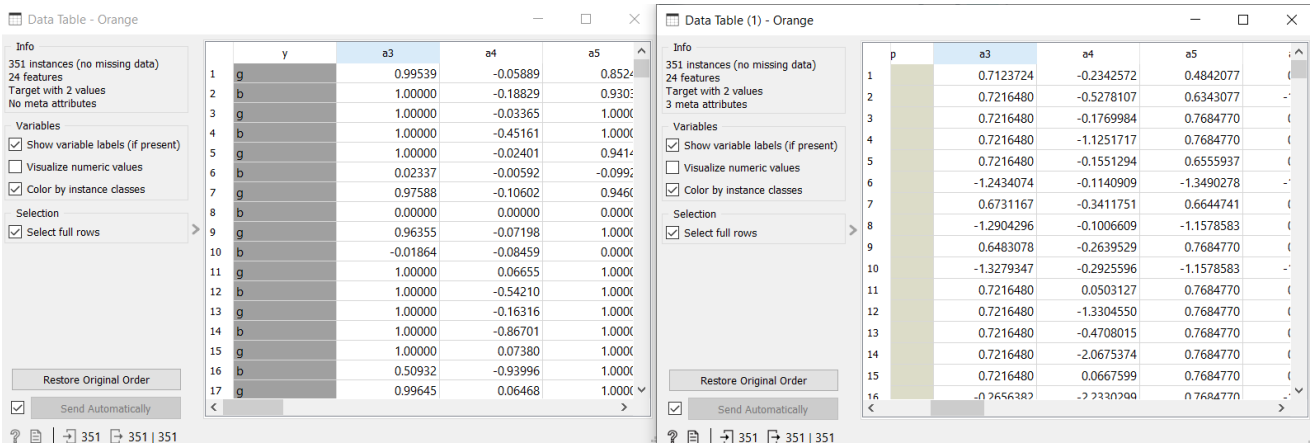


This is the graph I got after decreasing 0, 1, 2, 3, 4, 5, 6, 7 features from the dataset with the smallest eigen value after the PCA transformation.

h) Do the same as g) for MDS or t-SNE transformations.

MDS

Comparing the Data Tables of the features before and after the MDS transformation it is clear that a lot of changes have been occurred, as the numbers have changed completely. We can see it below:

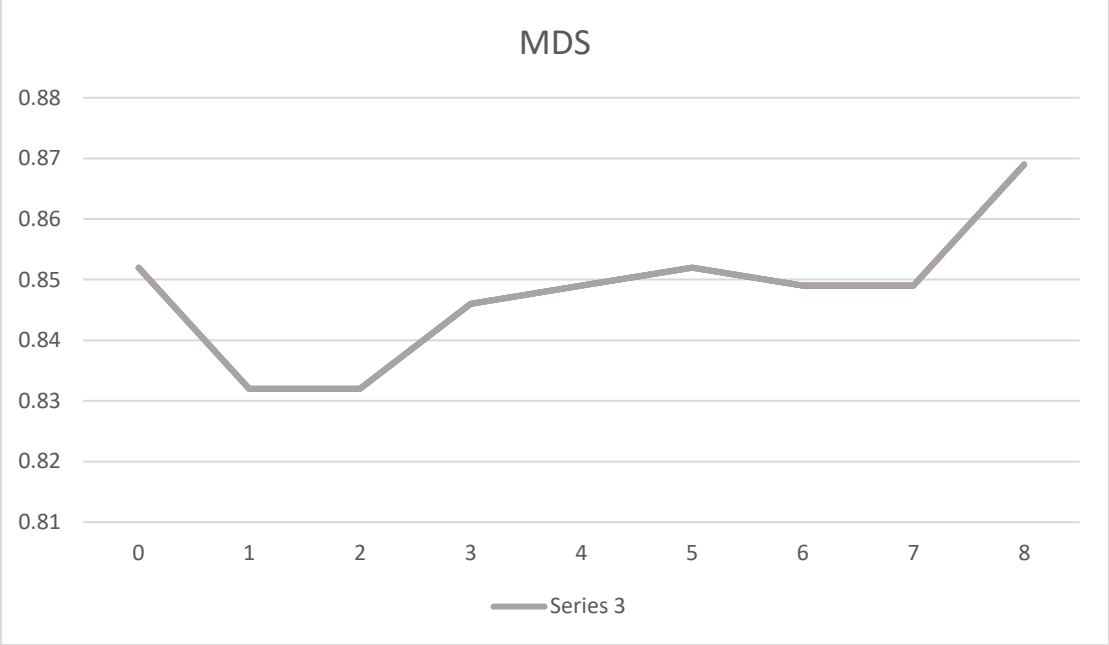


On the right are the features after the PCA transformation and on the left before this transformation.

The value of CA with MDS is 0.852.

Model	Train time [s]	Test time [s]	AUC	CA
Neural Network	4.552	0.128	0.885	0.852

Before the transformation the CA was 0.852.



This is the graph that represents the changes of value of Classification Accuracy I got after decreasing 0, 1, 2, 3, 4, 5, 6, 7 features from the dataset with the smallest eigen value after the PCA transformation.

t-SNE

The value of the CA = 0.849 with t-SNE.

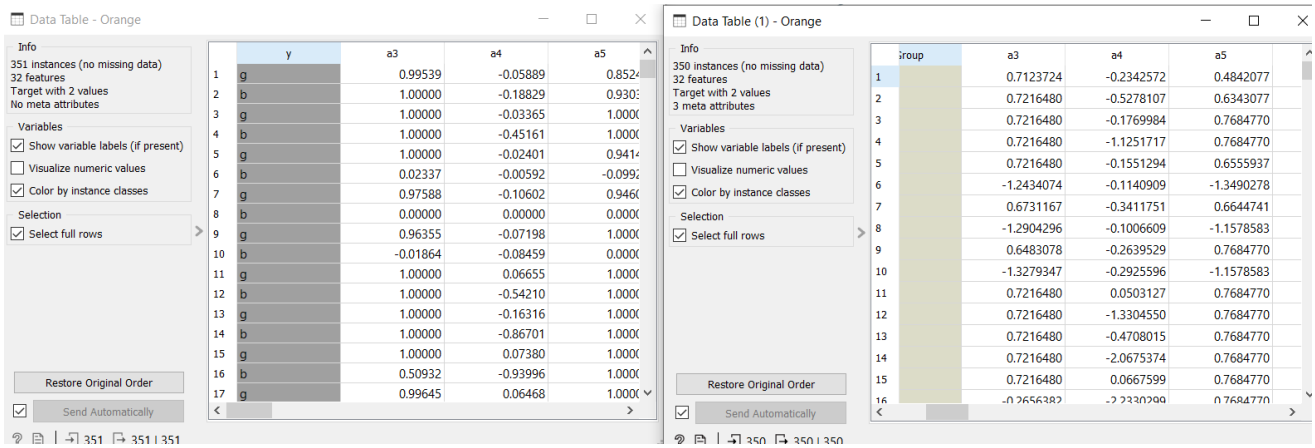
Model	Train time [s]	Test time [s]	AUC	CA
Neural Network	4.257	0.127	0.885	0.849

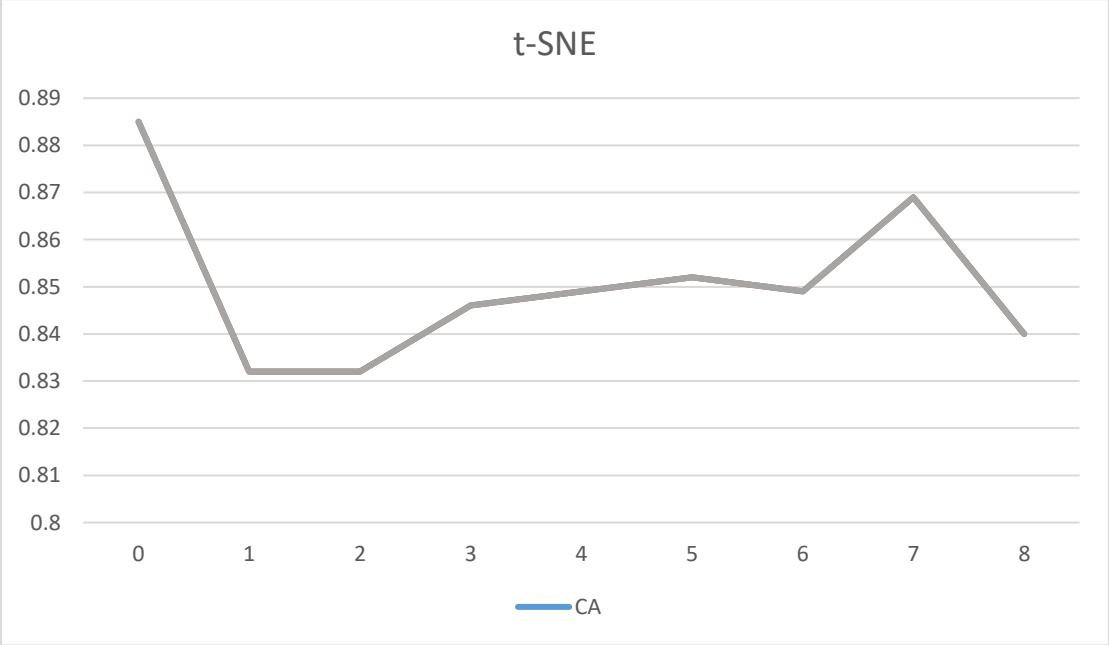
While this is the one before this transformation.

Model	Train time [s]	Test time [s]	AUC	CA
Neural Network	4.295	0.121	0.885	0.852

We can notice that after the transformation the CA has diminished a little bit.

As far as the features are concerned we can notice that there are a lot of changes.





This is the graph I got after decreasing 0, 1, 2, 3, 4, 5, 6, 7 features from the dataset with the smallest eigen value after the t-SNE transformation.