Digital Health
Report for the Statistical Analysis

Nikolaos Theodorakopoulos[*]

July 14, 2020

[*] School of Mathematics, National and Kapodistrian University of Athens, Athens, Greece (nikostheodorakopoulos1990@hotmail.com).

Abstract

Our dataset contain 430 patients and 1197 blood metabolites for each patient and our goal is to find key biological metabolites that could explain part of the variability in diabetes. In our opinion the biggest problem of this project was the huge amount of NaN values that we had in our initial dataset. At first, we cleaned our data from NaN columns and outliers. Next we followed this process:

- Cluster Analysis: We want to see how many clusters can be created for our data. Moreover, we need to know which of the metabolites belong to each cluster, i.e how the clustering was made.
- Classification/Regression analysis: We created some models to find the important features (blood metabolites) that were used in order to classify if a person has diabetes or not.

In the end, we find that the most important features are `CC_48153` and `CC_48152`. Also other blood metabolites have an important role.

1 Data Preparation

We cleaned the data with the following way: First of all, we removed the columns containing more than 100 NaN values. Next, we consider all values more than 4 standard deviations outliers, and remove them from the dataset. In fact we replace the outliers with NaN values. Next, we replace the remaining NaN values with Knn-Imputer with 100 neighbours. The remaining dataset has 403 rows and 904 columns and this is the final dataset that we will work for this project. Finally, we will scale our data set and begin the cluster and classification analysis.

2 Cluster Analysis

We will do PCA to reduce the dimensionality of our data. Applying the PCA method we will get the following figure:

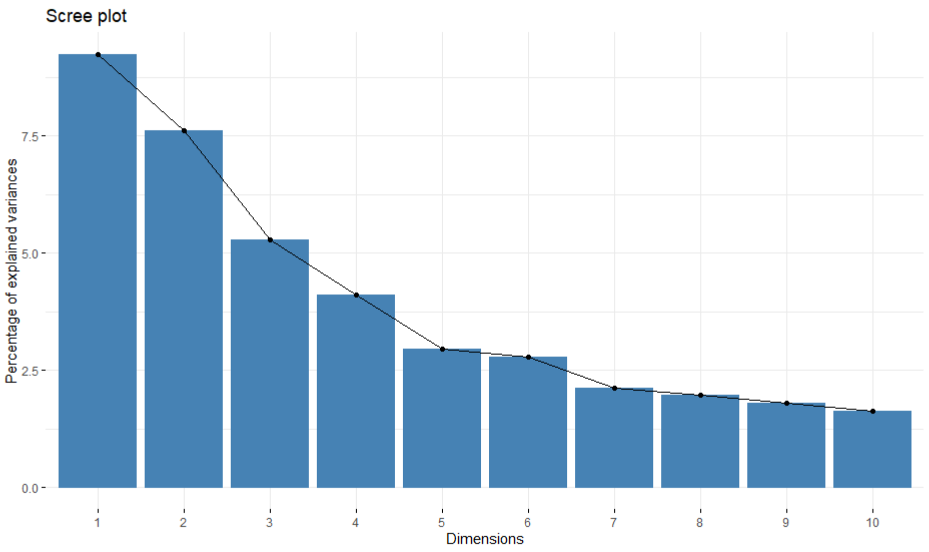

Figure 1: The PCA Method

It is clear that the PCA dimensions have a small percentage of explained variance. A closer look suggest, that taking 100 PCA columns we will describe the 83.5% of our data variance. So we will take 100 PCA columns. Next we need to find the number of clusters for the PCA columns. We will use the elbow method for this:

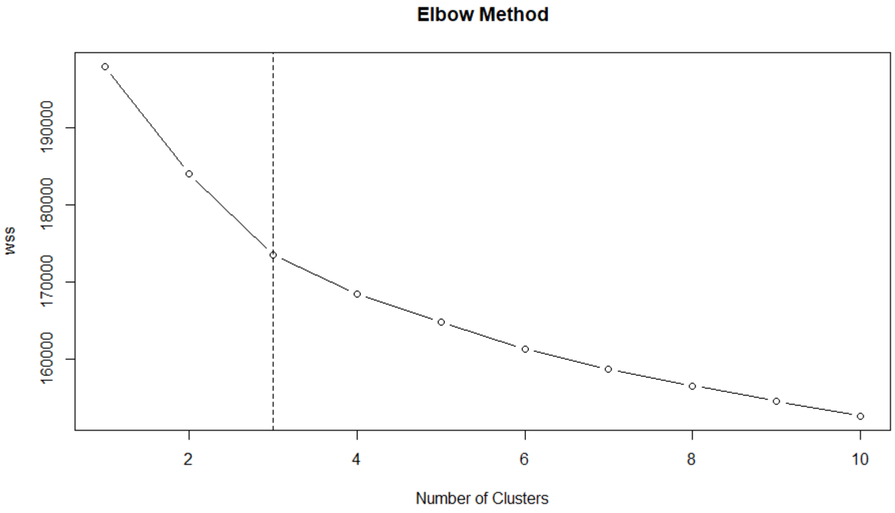

Figure 2: The Elbow Method suggest 3 Clusters

Now we want to see which of the PCA columns play important role in clustering .We know that the clustering is impacted by the random initialization. Thus it is usually recommended to run the clustering alogrithm several times with different seeds. For our project we repeat the clustering and feature importance calculation 20 times and we get the following figure:

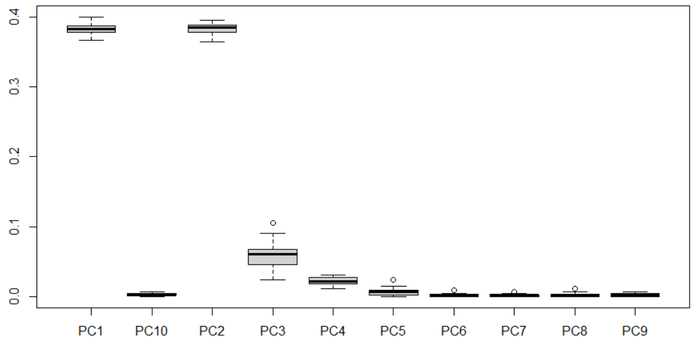

Figure 3: The importance of the first 10 columns of PCA, for clustering

From the figure above we see that the first and second dimensions of PCA have the biggest impact on clustering. Moreover, for each of the three clusters we have the following:

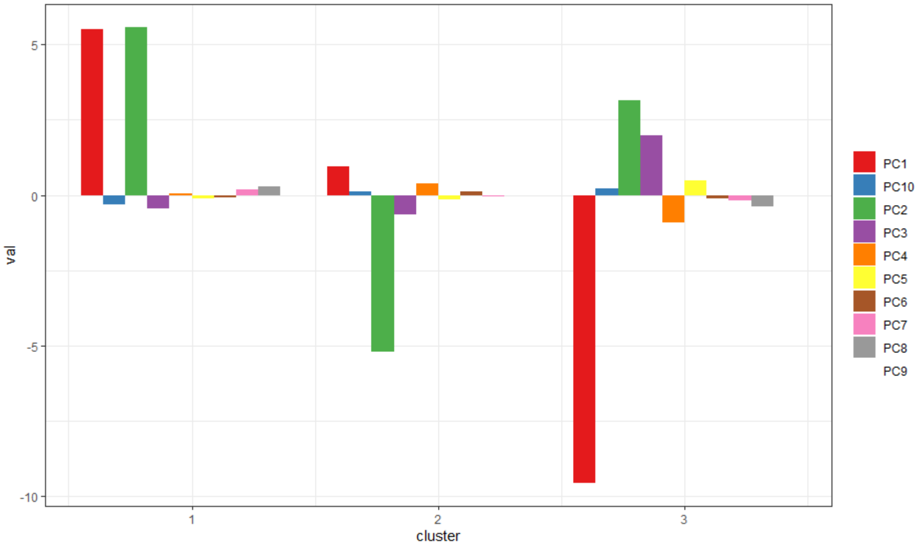

Figure 4: The role of PCA columns for each cluster

We observe that in cluster 1 the greatest impact was from the positive values of PC1 and PC2, while in cluster 2 the greatest impact was from the negative values of PC2. Finally in cluster 3 the greatest impact was from the negative values of PC1 and the positive values of PC2 and PC3. Another way to visualize this is the following:

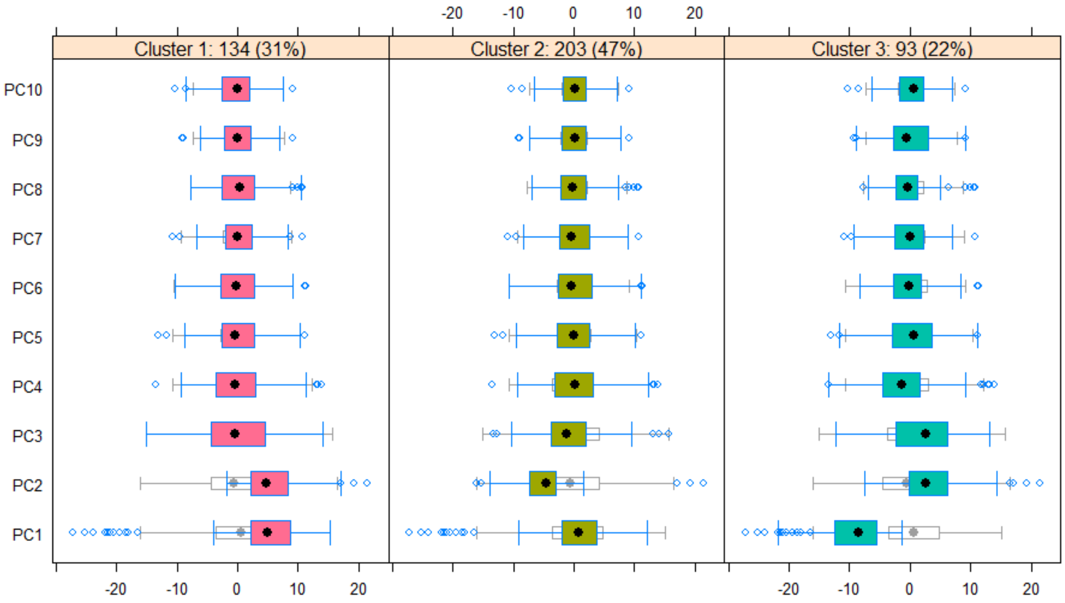

Figure 5: The role of PCA columns for each cluster

All that remains, is to see the quality of representation of blood metabolites (columns) in each PCA dimension. Now we present the top 30 blood metabolites for each of the first three PCA dimensions:

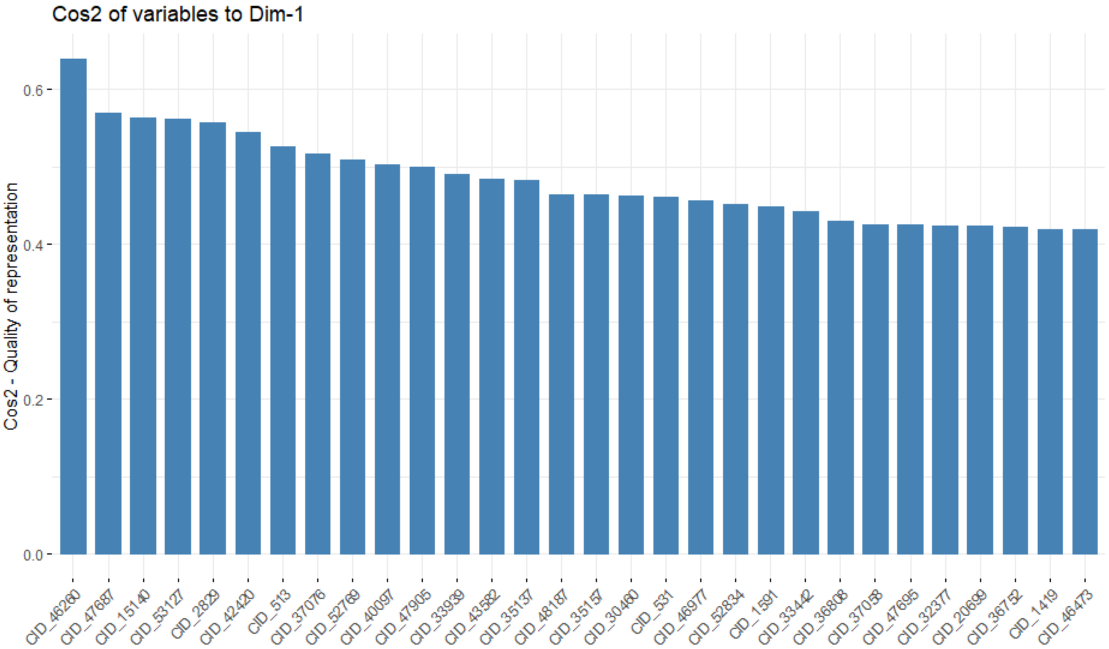

Figure 6: The top 30 blood metabolites in PC1

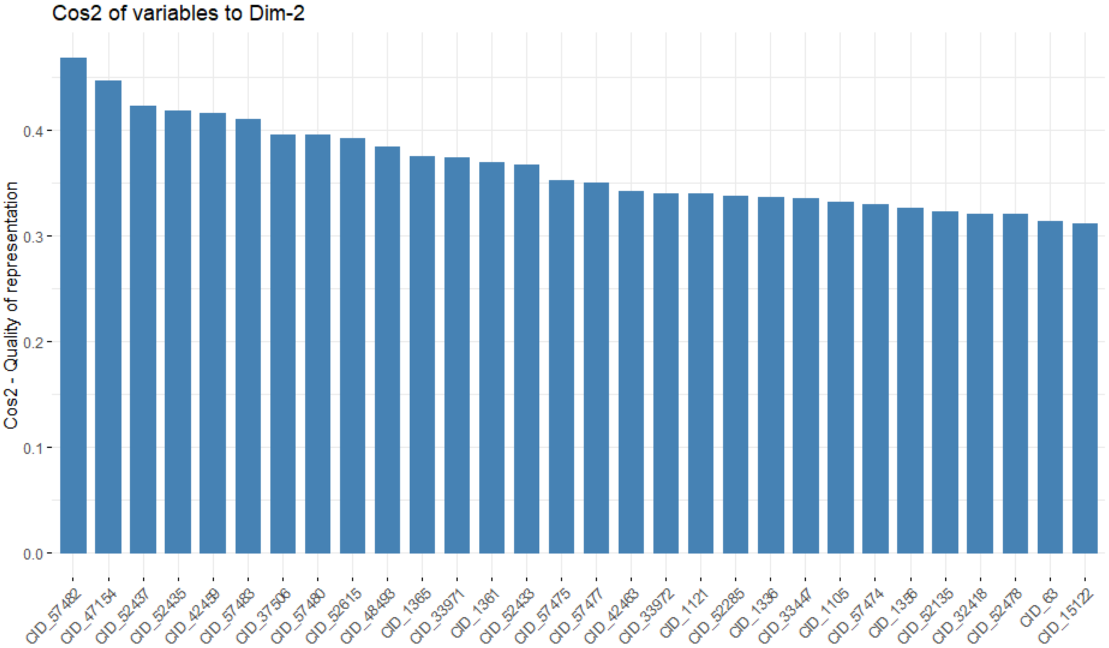

Figure 7: The top 30 blood metabolites in PC2

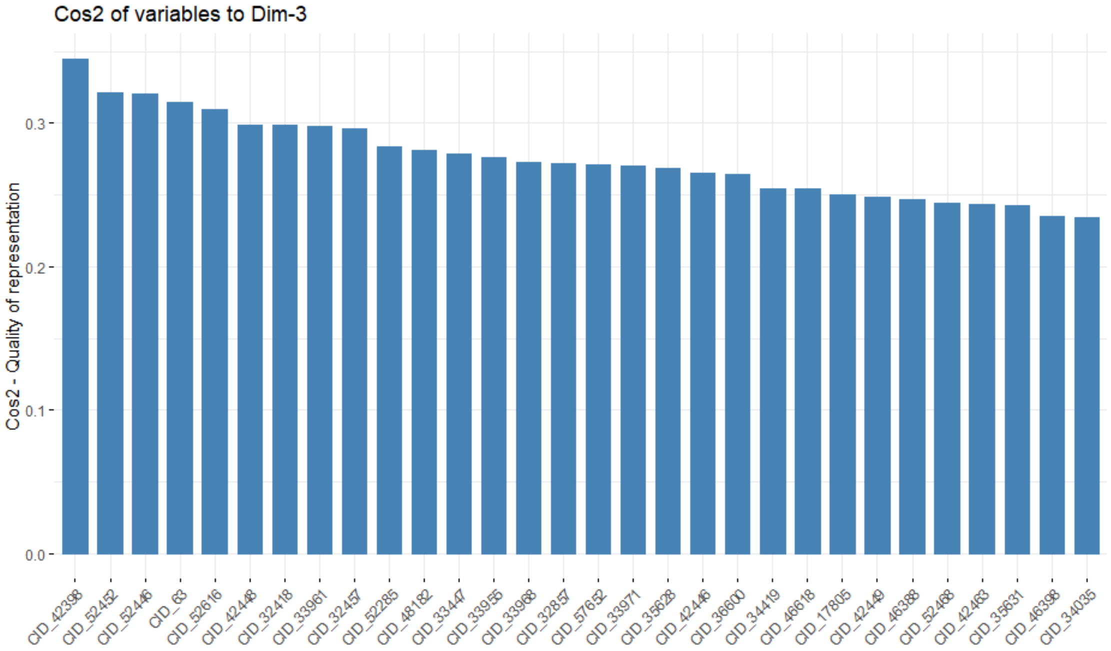

Figure 8: The top 30 blood metabolites in PC3

Now we can say that the above blood metabolites in PC1, PC2 or PC3 play a key role for the grouping/clustering of our data and thus we know the grouping relationship between the clusters.

3 Regression/Classification Analysis

In this section we will use Random Forest, xGboost and Logistic Regression as classifiers. We will train and tune these models on a training set and then evaluate them on the test set to measure their performance. The train stage will give us an inside vision about what features are used in order to fit the training data i.e which are the important features(blood metabolites) to classify a person as diabetic or healthy.

3.1 Splitting the data into train and test set

We will use the 75% of the initial data as training set and the rest will be our test set, i.e the train set will have 322 rows and the test set will have 108 rows.

3.2 Random Forest Classifier

We will tune our model on the train set and the we will make predictions on the test set. So we have:

- Accuracy on Train set: 67% and 95% CI = (0.6165, 0.7219)
- Accuracy on Test set: 80% and 95% CI = (0.7183, 0.8754)

From the 95% CI results we can say that our model predicts better the diabetes class (right) than the healthy class(left) both in training and test set. Also, the 80% accuracy on the test set is satisfying. Perhaps our model is under fitting a bit, but this could be fixed with a better tuning on the training data. Now in the figure below, we present the important features that our model used to make the classifications.

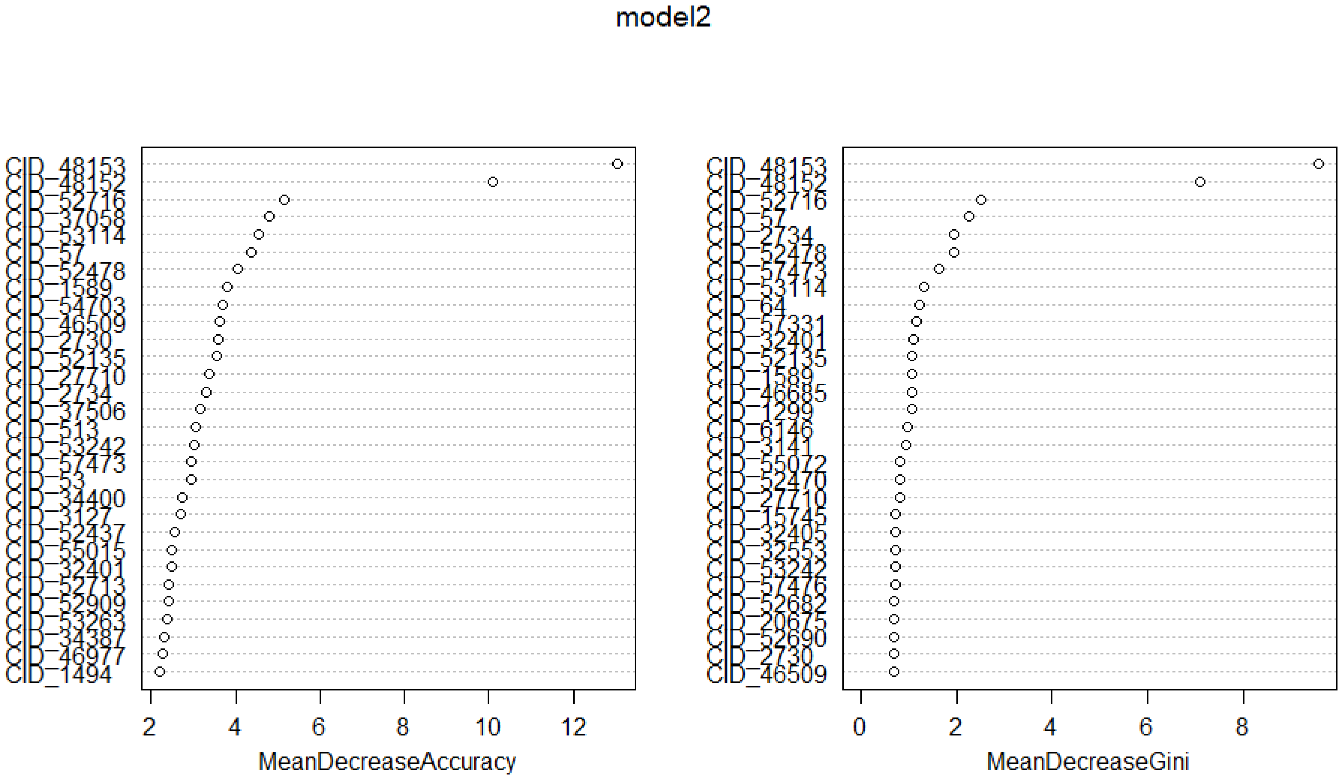

Figure 9: The Important Features of Random Forest Classifier

Metabolite	Importance
CID_48153	100.00
CID_48152	80.69
CID_37058	55.89
CID_52716	55.86
CID_1589	53.86
CID_52470	49.01
CID_57	48.48
CID_3141	48.35
CID_1299	47.80
CID_2342	46.52
CID_46509	45.44
CID_2730	45.35
CID_53114	43.68
CID_54979	43.13
CID_55015	42.14
CID_38168	42.13
CID_32401	41.43
CID_57473	41.20
CID_57564	40.67
CID_57331	40.52

Table 1: The important Features of RF

3.3 xGboost Classifier

We will follow the same process here. So we have:

- Accuracy on Train set: 83 % and 95% CI = (0.7937, 0.877)
- Accuracy on Test set: 74 % and 95% CI = (0.6475, 0.8203)

This time our model do not under-fit. Still our model does not fit very well with the healthy class. We now present the important features of xGboost Classifier.

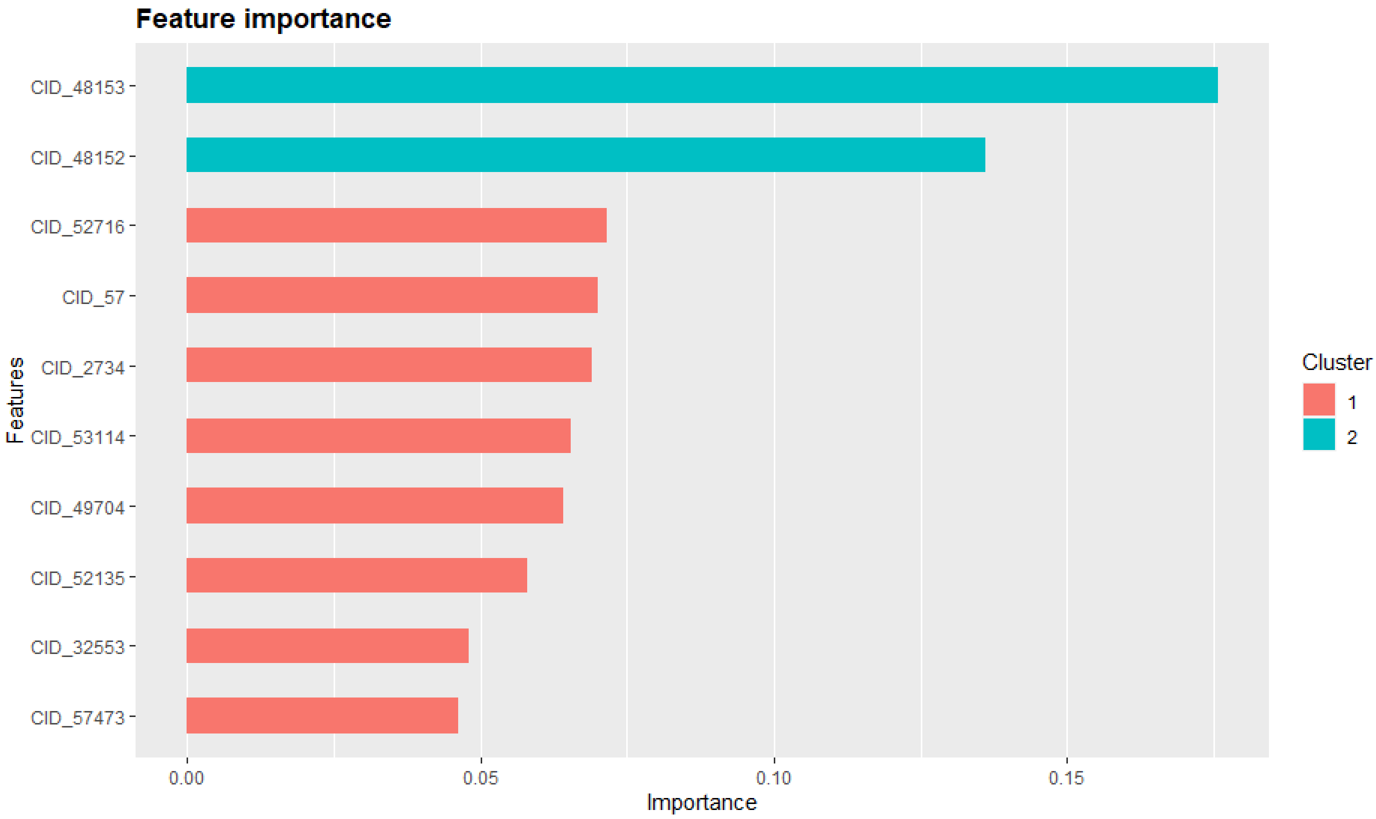

Figure 10: The Important Features of xGboost Classifier

We observe that the important features are similar with the random forest classifier. Again the `CC_48153` and `CC_48152` are on top of importance.

3.4 Logistic Regression

For the logistic regression if we try to fit the data, as we did with the previous models we will have good results. So we will do a small trick. We will take the important features of the xGboost and we will use them as the formula for the Logistic Regression. So we have:

- Accuracy on Train set: 70 % and 95% CI = (0.6519, 0.7542)
- Accuracy on Test set: 75 % and 95% CI = (0.6675, 0.8363)

We now plot the train and test fitting:

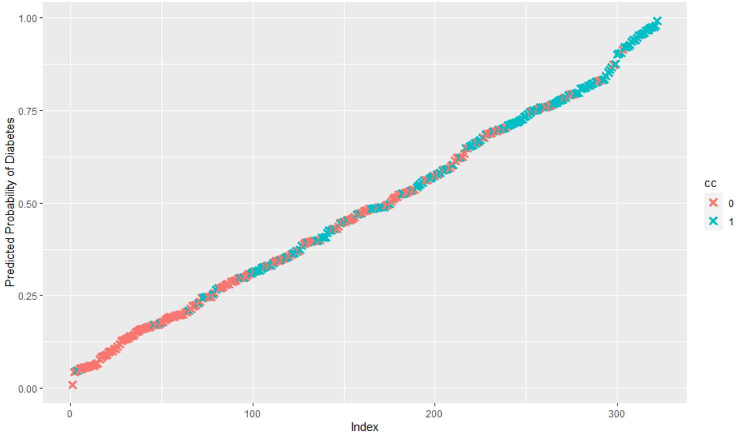

Figure 11: Logistic Regression fitting the Train Set

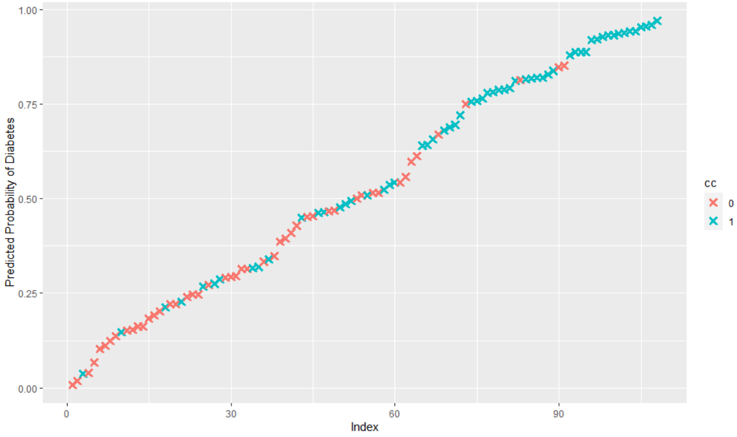

Figure 12: Logistic Regression fitting the Test Set

4 Results and Future Improvement

For the clustering we found a grouping correlation between the blood metabolites while in regression/classification analysis we found that the most important features that our models used to classify diabetes and non-diabetes were `CC_48153`, `CC_48152`, `CC_52716`, `CC_57`, `CC_37058`,`CC_53114` and others. We also propose the following for future improvement:

- Have a better dataset with not so many NaN values.
- Deeper understanding about the true meaning of outliers in blood metabolites. Perhaps we could ask a doctor or a biologist about extreme values in blood metabolites.
- Better tuning of the regression models.