IITM-CS5691 : Pattern Recognition and Machine Learning  Release Date: 22 Oct, 2021
New Method Writeup  Due Date: 10 Nov, 2021

Roll No: CS21M028, CS21M037  Names: Karthikeyan S, N Kausik

- Dear Student, You may have tried different methods for predicting each of the clinical descriptors in the data contest. Submit a write-up of the methods chosen for the data contest in the template provided below. **You will have to add the details in your own words and submit it as a team in gradescope**.
- We will run plagiarism checks on codes/write-up, and any detected plagiarism in writing/code will be strictly penalized.

1. ( points) [Name of the Method chosen for each descriptor, and its Paradigm: paradigm could be linear/non-linear models, etc.)]:

**Solution:**

For all endpoints, we have used the **AdaBoost** classifier with varied hyperparameters. Paradigm of the AdaBoost classifier is **Ensemble** model.

2. ( points) [Brief description on the dataset: could show graphs illustrating data distribution or brief any additional analysis/data augmentation/data exploration performed]

**Solution:**

There are two datasets provided.

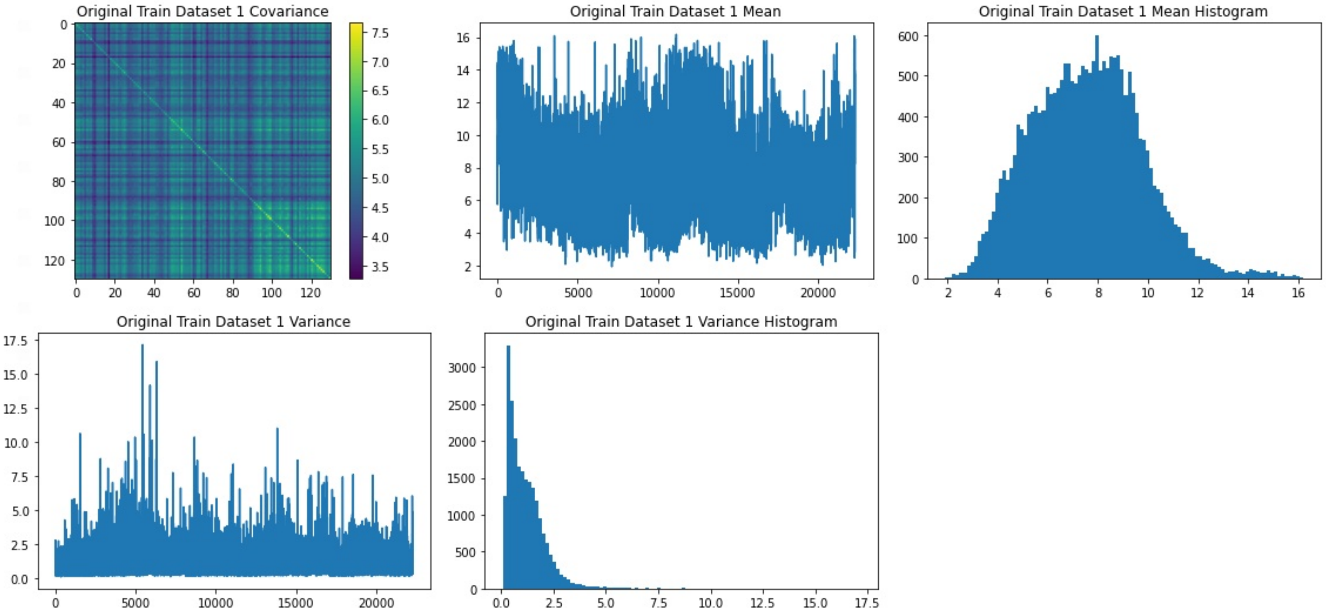
**Train Dataset** 1

Number of points = 130

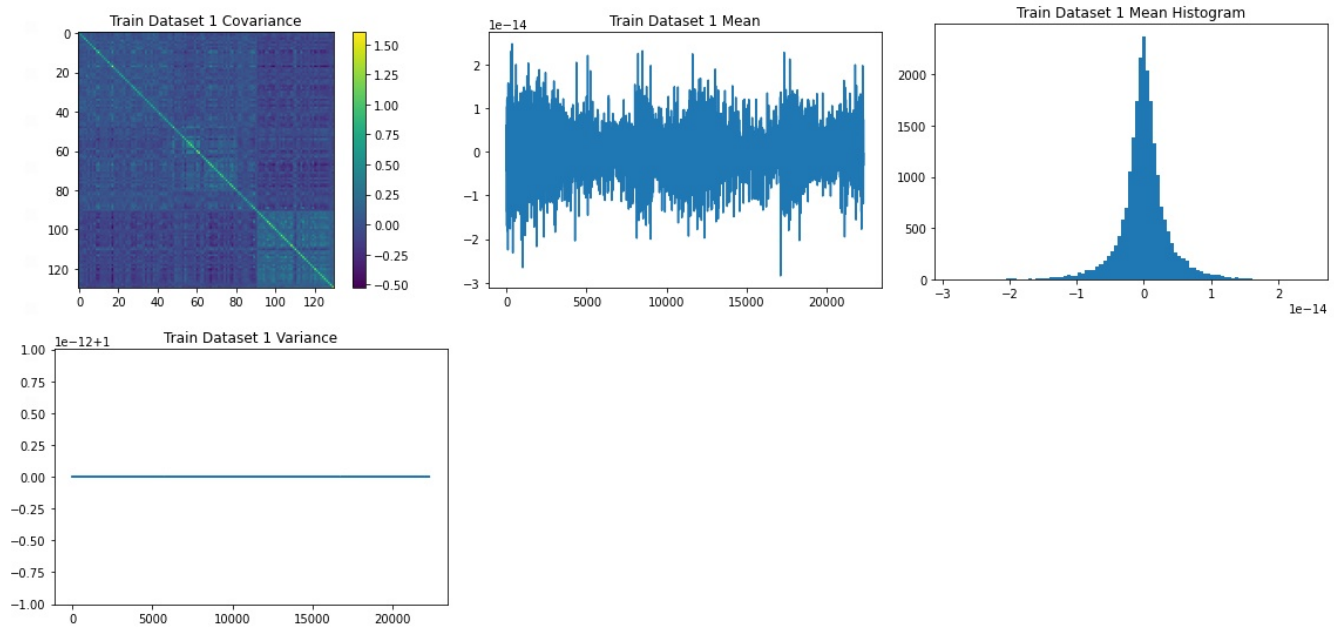
Number of genes/features = 22283

Plots:

Original Train Dataset 1:



Scaled Train Dataset 1:



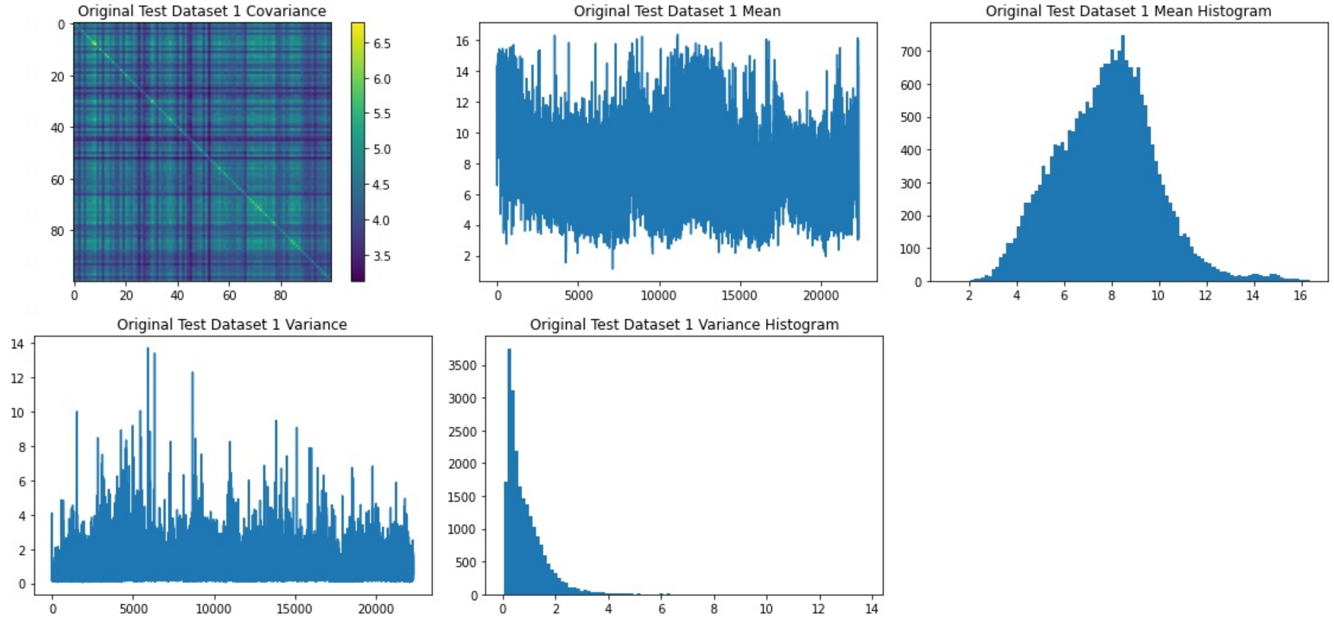
**Test Dataset** 1

Number of points = 100

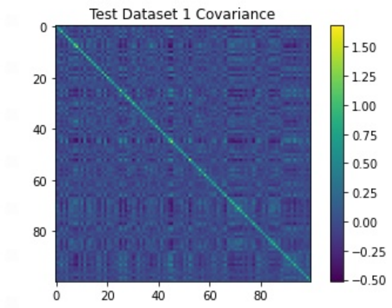
Number of genes/features = 22283

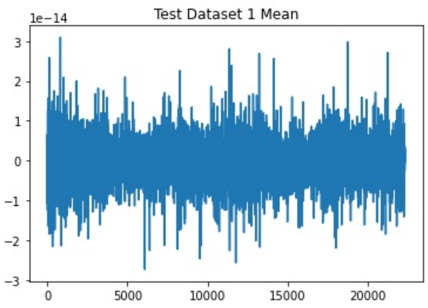
Plots:

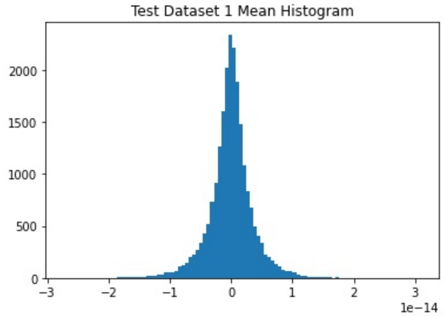
Original Test Dataset 1:

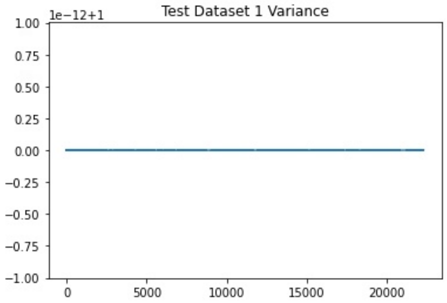


Scaled Test Dataset 1:









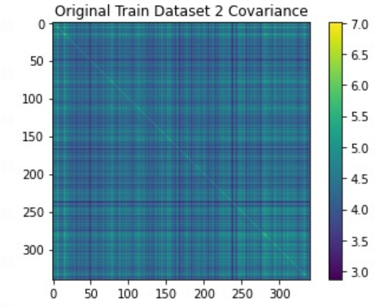
**Train Dataset** 2

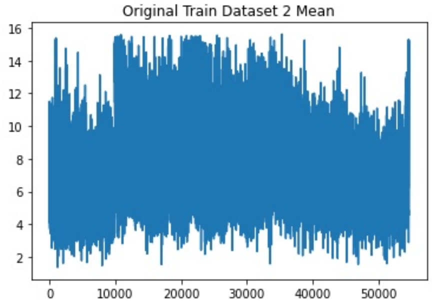
Number of points = 340

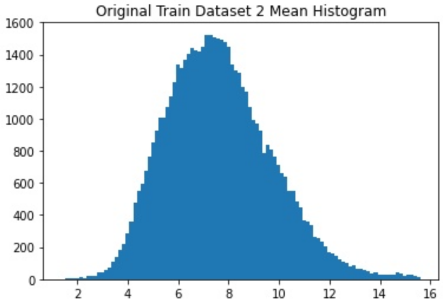
Number of genes/features = 54675

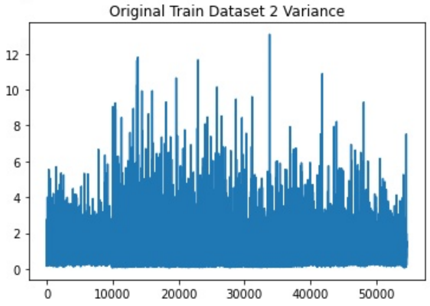
Plots:

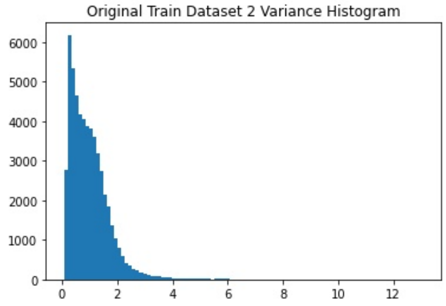
Original Train Dataset 2:



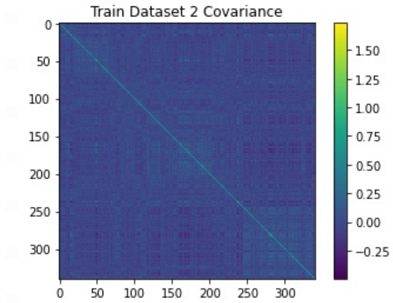


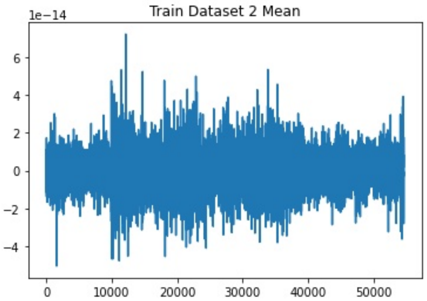


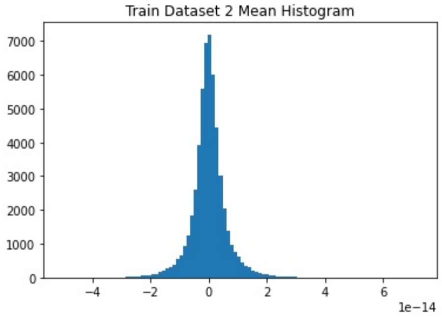


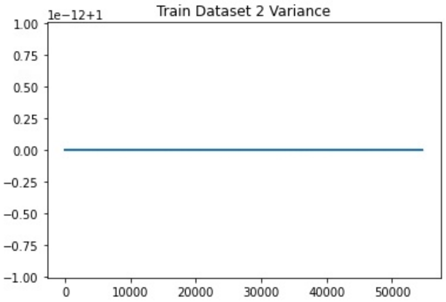


Scaled Train Dataset 2:









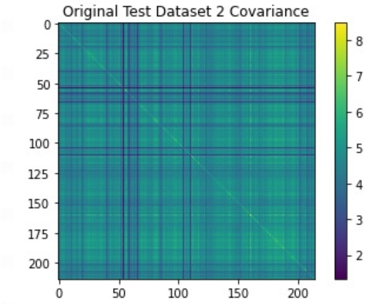
**Test Dataset** 2

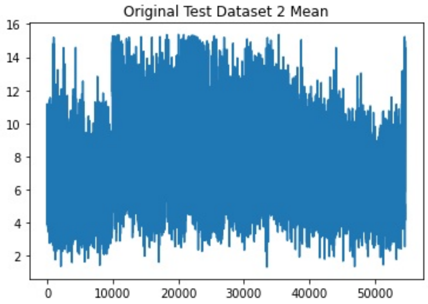
Number of points = 214

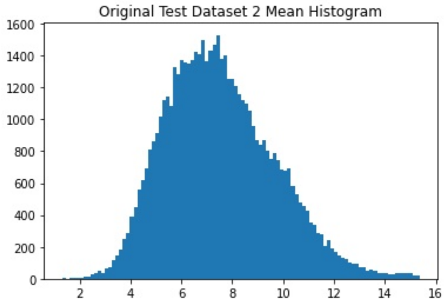
Number of genes/features = 54675

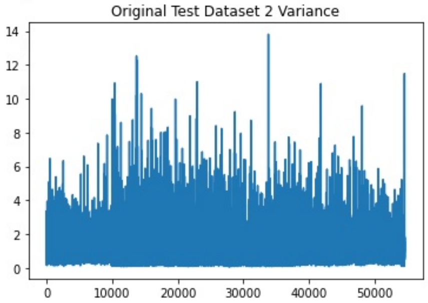
Plots:

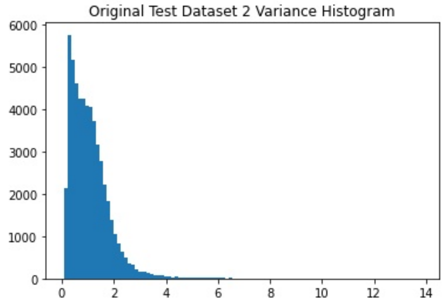
Original Test Dataset 2:



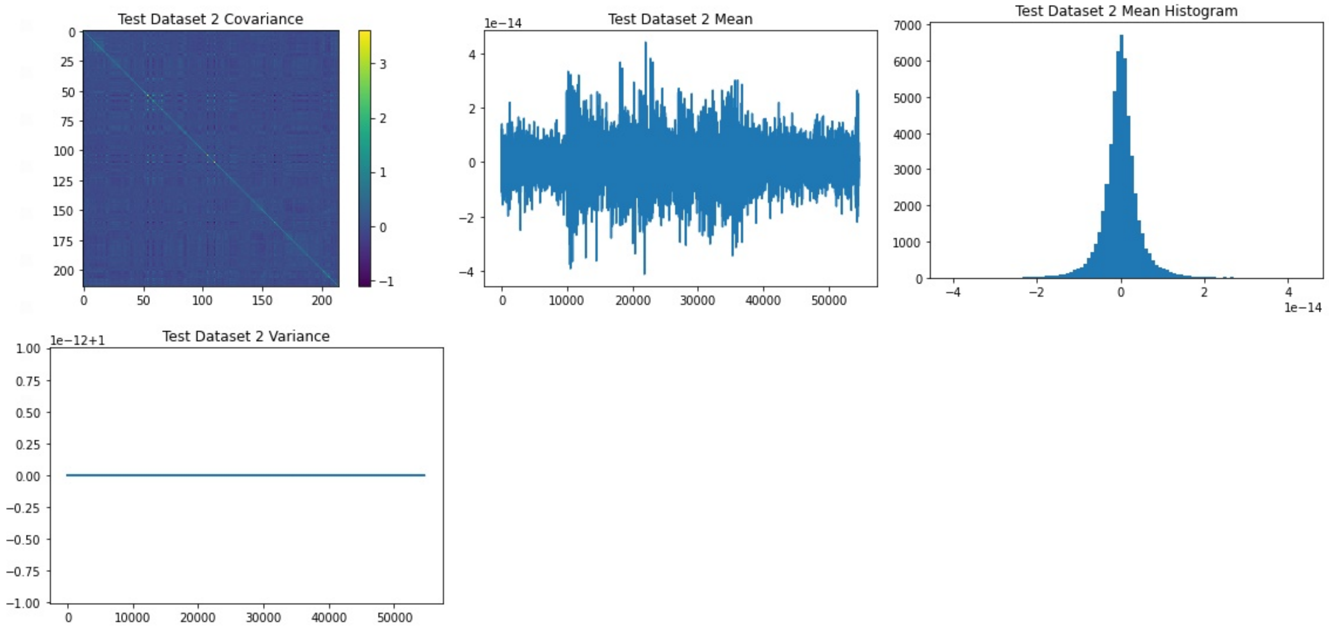




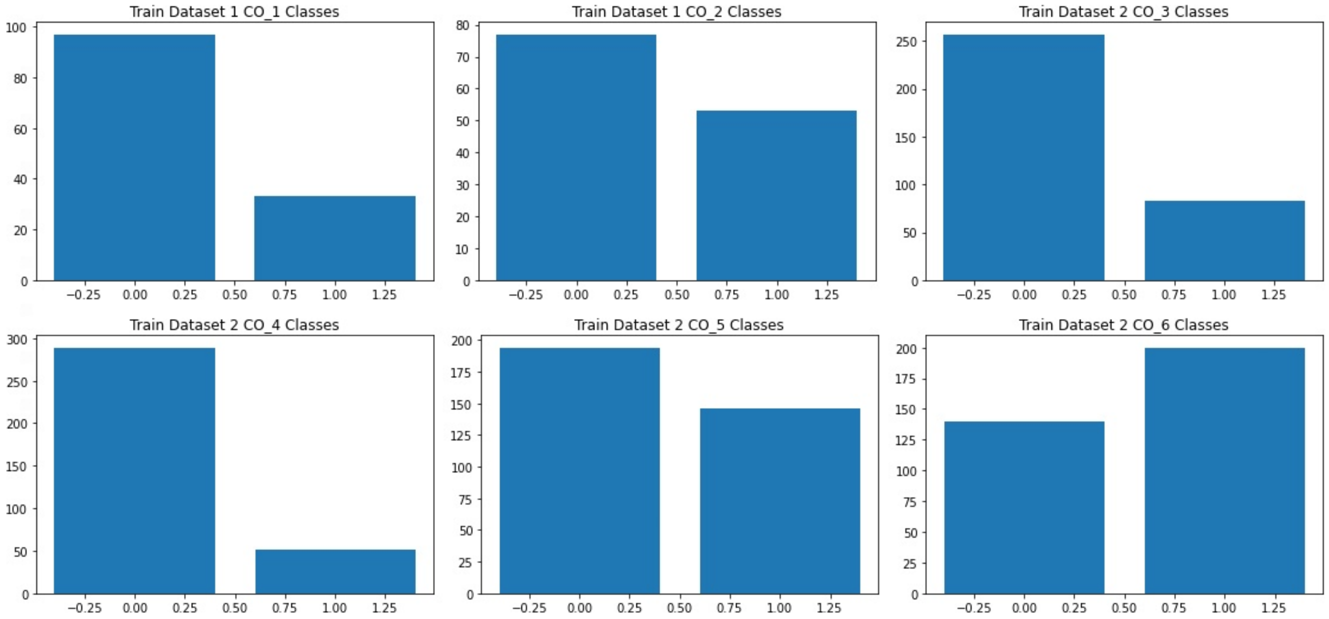




Scaled Test Dataset 2:



Class Imbalance Plots:



3. ( points) [Brief introduction/motivation: Describe briefly the reason behind choosing a specific method for predicting each of the descriptors (could show plots or tables summarizing the scores

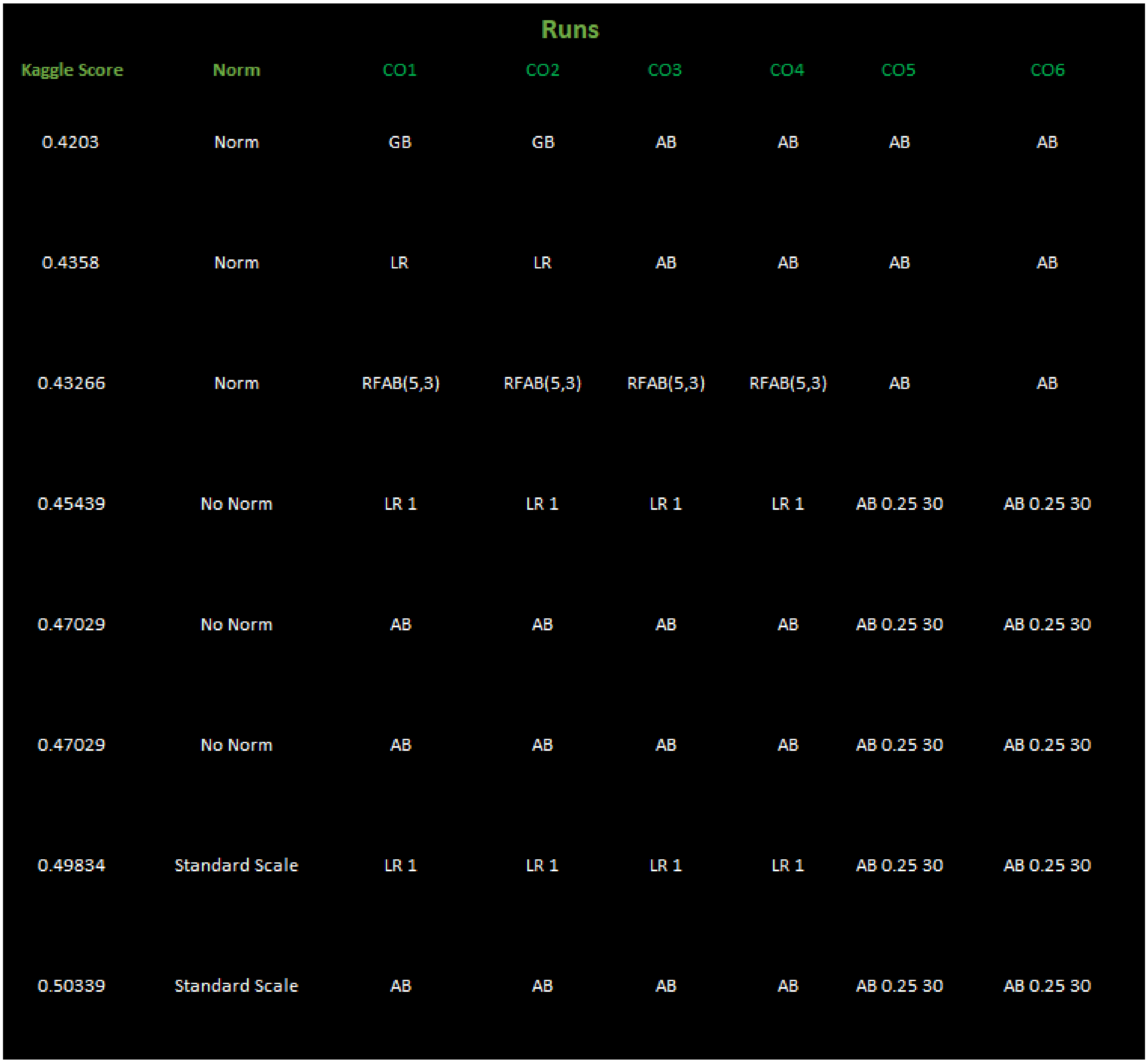
of training different model)]

**Solution:**

We used AdaBoost for all the descriptors with varying hyperparameters after running various algorithms.

Several algorithms like SVM and Bayes were tried out initially but they performed very poor on validation. We started testing with more algorithms like Logistic Regression, Gradient Boosting and AdaBoost with Decision Tree as base estimator and AdaBoost with Random Forest as base estimator.

The results are as given below.



| | | | | Runs | | | |
|---|---|---|---|---|---|---|---|
| Kaggle Score | Norm | CO1 | CO2 | CO3 | CO4 | CO5 | CO6 |
| 0.4203 | Norm | GB | GB | AB | AB | AB | AB |
| 0.4358 | Norm | LR | LR | AB | AB | AB | AB |
| 0.43266 | Norm | RFAB(5,3) | RFAB(5,3) | RFAB(5,3) | RFAB(5,3) | AB | AB |
| 0.45439 | No Norm | LR 1 | LR 1 | LR 1 | LR 1 | AB 0.25 30 | AB 0.25 30 |
| 0.47029 | No Norm | AB | AB | AB | AB | AB 0.25 30 | AB 0.25 30 |
| 0.47029 | No Norm | AB | AB | AB | AB | AB 0.25 30 | AB 0.25 30 |
| 0.49834 | Standard Scale | LR 1 | LR 1 | LR 1 | LR 1 | AB 0.25 30 | AB 0.25 30 |
| 0.50339 | Standard Scale | AB | AB | AB | AB | AB 0.25 30 | AB 0.25 30 |

Here,

$\Longrightarrow$ GB - Gradient Boosting

$\Longrightarrow$ LR - Logistic Regression with L2 penalty and C = 1

$\Longrightarrow$ RFAB(n estimators, seed) - AdaBoost (200 estimators, max depth = 1, learning rate = 0.2) with Random Forest (5 estimators) as base estimator

$\Longrightarrow$ AB - AdaBoost (200 estimators, max depth = 1, learning rate = 0.2) with Decision Tree (max depth = 1) as base estimator

$\Longrightarrow$ AB 0.25 30 - AdaBoost (30 estimators, max depth = 1, learning rate = 0.25) with Decision Tree (max depth = 1) as base estimator

After deciding AdaBoost as the method, we also split the training data into train and validation data and trained for several hyperparameters. Finally we arrived at the best values.

$\Longrightarrow$ $CO_1$ - AdaBoost with Decision Tree base estimator, 0.2 Learning Rate, 1 Max Depth, 200 Estimators, Seed 0

$\Longrightarrow$ $CO_2$ - AdaBoost with Decision Tree base estimator, 0.2 Learning Rate, 1 Max Depth, 200 Estimators, Seed 0

$\Longrightarrow$ $CO_3$ - AdaBoost with Decision Tree base estimator, 0.2 Learning Rate, 1 Max Depth, 200 Estimators, Seed 0

$\Longrightarrow$ $CO_4$ - AdaBoost with Decision Tree base estimator, 0.2 Learning Rate, 1 Max Depth, 200 Estimators, Seed 0

$\Longrightarrow$ $CO_5$ - AdaBoost with Decision Tree base estimator, 0.25 Learning Rate, 1 Max Depth, 30 Estimators, Seed 0

$\Longrightarrow$ $CO_6$ - AdaBoost with Decision Tree base estimator, 0.25 Learning Rate, 1 Max Depth, 30 Estimators, Seed 0

4. ( points) [As the data comes from a clinical setting, model interpretation is also an important task along with prediction. Describe briefly your preferred choice of methods if you are asked to determine the significant genes behind the regulation of the endpoints or restrict the number of features/genes.]:

**Solution:**

For the purpose of regulation of endpoints by identifying significant genes can be done using **PCA**, but here we did not use it as the number of datatpoints available for training are very less when compared to the number of features of a datapoint. Hence, we could not apply PCA since number of data points is less than number of features.

5. ( points) [Share your thoughts on each of the endpoints: whether easy/difficult to predict]:

**Solution:**

**Endpoint 1: ( CO1):**

As the number of datapoints available for training were low, this endpoint seemed to be difficult for predicting at initial stages but soon we found the the hyperparamters for the model which made the model to perform well on this endpoint.

**Endpoint 2: ( CO2):**

Once the relevant hyperparamters for endpoint 1 were found, the task of finding the relevant hyperparamters for other endpoints became quite easy. The hyperparamters which were found for endpoint 1, performed well for this endpoint too.

**Endpoint 3: ( CO3):**

A similar approach which was used in finding the hyperparamters for ealier endpoints, was followed for this endpoint too and those hyperparamters found earlier, generalized and performed well on this endpoint also.

**Endpoint 4: ( CO4):**

The hyperparamters found for endpoint 1 was performing well over endpoint 4 too and thus the same set of hyperparamters were used to predict endpoint 4.

**Endpoint 5: ( CO5):**

Endpoint 5 and 6 showed similar behaviour and were tricky to find well performing hyperparamters. Then we preprocessed the data by standardizing the features for this endpoint using Standard Scaler module of sklearn, which behaved very satisfying than before and thus we arrived at a set of better performing hyperparamters.

**Endpoint 6: ( CO6):**

Predicting the endpoint 6 was really challenging, as this endpoint was very susceptible to even minor changes than any other endpoint. So predicting this endpoint was most difficult. So we followed the same preprocessing methods applied on endpoint 5 and finally we observed the improvement in overall result.

6. ( points) [Add any additional information: like challenges faced or some details that would help us to better understand the strategies that you have utilized for model development]:

**Solution:**

There have been many difficulties faced during the process as follows:

a) Since we are given with unlabelled test data, we found it difficult to observe what is happening behind as the model behaves very different than it did with train data. Even we tried finding optimal hyperparameters by doing a train-test split on the train dataset given and found some parameters but those behaved very differently when trained over the whole dataset. This may be due to the low number of training data points given for each dataset (Only 100 and 340 points).

b) The data is imbalanced with respect to the target values and the number of data points available is much lesser than the dimensions of the data.

c) We were allowed to use Ensemble methods but we were not allowed to use XGBoost algorithm which has been behaving like "state-of-the-art" model and thus we had to try more parameters to increase our MCC score.

d) Even after trying different models for each endpoints, we aren't much sure about the behaviour of the model as even a little change in the hyperparameters varied the result in a greater aspect.