MLBA
Assignment 1

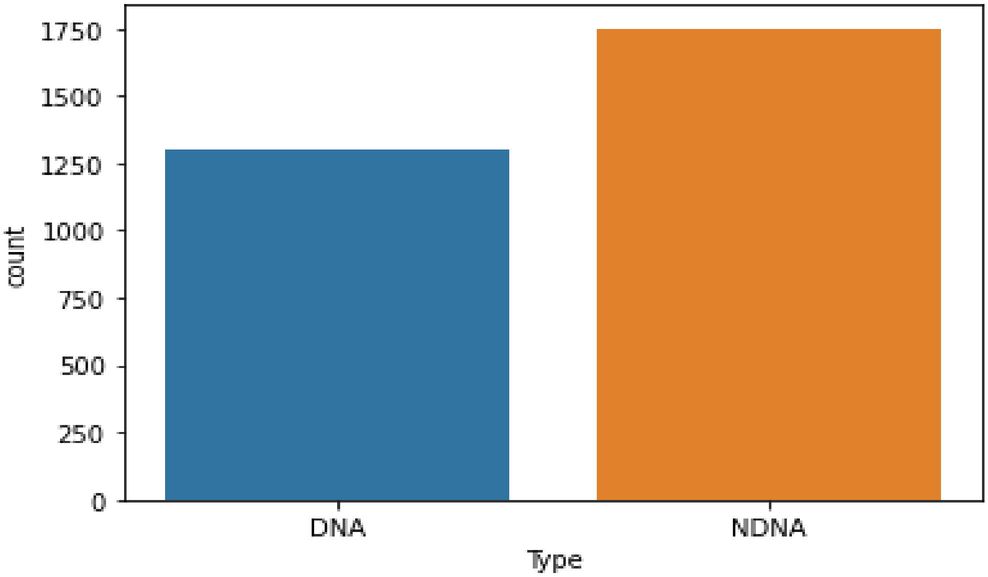
Submitted By: Gaurav Lodhi
MT19063

Prediction of DNA Binders:

To Predict whether a Binding is DNA or NDNA my approach was the following :

1. **Preprocessing:-** By using the Pandas library I read the given data files. I separated the train labels as Y_train and remove the id field. I found the DNA sequences and NDNA sequences as the following count.

   'NDNA': 1750, 'DNA': 1299



The preprocessing step majorly involves the following :

(i). Feature Extraction: I separated all the sequences in data_train. And used different feature techniques for feature extraction as following:-

A. AAC (Amino Acid Composition)
B. Dipeptide with a different order(j) Composition
C. Tripeptide composition

For my Features, I took some hybrid approach as follow

A. Traditional dipeptide composition Alone
B. Amino Acid Composition + Traditional Dipeptide composition
C. Amino Acid Composition + Dipeptide Composition with different order(0-5)
D. Amino Acid Composition +Tripeptide Composition

(ii). Feature Selection: Different techniques to select the features from the extracted features as following:

A. PCA: I was used for feature reduction to speed up the process.
B. SelectKBest: This technique is used to select the features.

(iii). OverSampling:- From the above analysis and I found out that we have a class imbalance problem in the dataset because we have 1750 NDNA sequences but the number of DNA sequences is only 1299. So to overcome this imbalance I used Sklearns's RandomOverSampling and oversample the minority class and make a balanced dataset with 1750 NDNA and 1750 DNA sequences.

2. Methodology:- After getting the preprocessed dataset I used the following steps to proceed:-

(i) Model Selection: This is a classification dataset so I tried many different models to those that give good performance on classification. Some of those are mentioned below:

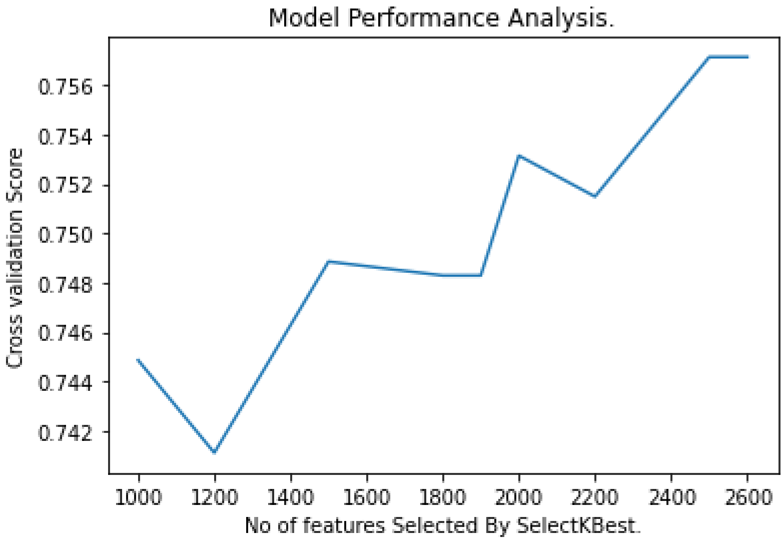
a. Random Forest Classifier
b. XGBoost Classifier
c. SVC

(ii) Model evaluation: I used accuracy measure for performance and use 5-fold cross-validation. I found Random Forest and XGBoost are performing similar and got the cross-validation score near .71 which gives a .70280 score on the public leaderboard. That was the best performance of these models. So I tried the different models as SVC classifiers.

SVC classifier performs better than the above classifiers. Here I am taking a cross-validation score as a metric to rank all the models

(iii). Hyperparameter Tuning:- To tune the best Hyperparameters I used GridSearchCV with cv=5 and tune the best parameters for models. I found the best parameters for SVC as, C= 10, gamma= 1, kernel= 'rbf'

On these parameters, I plotted the following plot, and on the basis of it, I tunned all the parameters.



3. Result: After accessing all the models and tunning their parameters I found the following configurations most accurate on the basis of their cross-validation scores.

A. SVC:
   Parameters: C= 10, gamma= 1, kernel= 'rbf'

   SelectKBest Parameters: random State:40
   Cross Validation accuracy:.7585
   Public leaderboard: 0.73084

B. SVC:
   Parameters: C= 10, gamma= 1, kernel= 'rbf'

   SelectKBest Parameters: random State:48
   Cross-Validation accuracy:.7585
   Public leaderboard: 0.71962

Note: There may have some problems in running the file because of local libraries. This is my humble request either update the libraries or use the code on google colab.