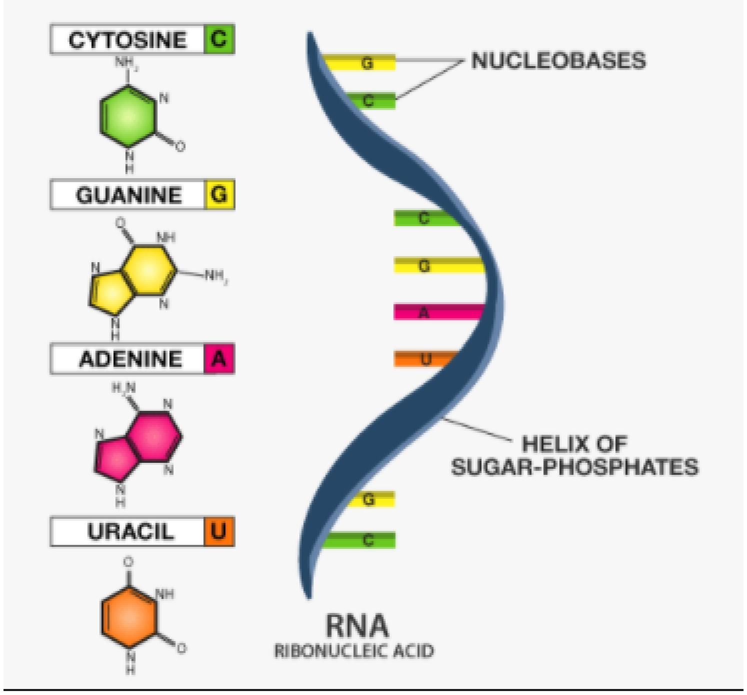
BDMH ASSIGNMENT : 02

CLASSIFICATION OF INTERACTING AND NON-INTERACTING RNA SEQUENCES



Group No. 42
TEAM MEMBERS

Akanksha Pandey MT20048

Shailja Upadhyay MT20095

Gaurav MT20112

1. Dataset Description:

Training data for classification of Interacting, Non-Interacting RNA contained 330862 rows with two columns, Sequence and Label. Sequence contained twenty types of Amino Acids. Each Sequence is concatenated with a character ‘X’ to make it of length 17. The given dataset is highly imbalanced between classes. Testing data contained a total 6276 rows.

2. Methodology:

a. Libraries Used :

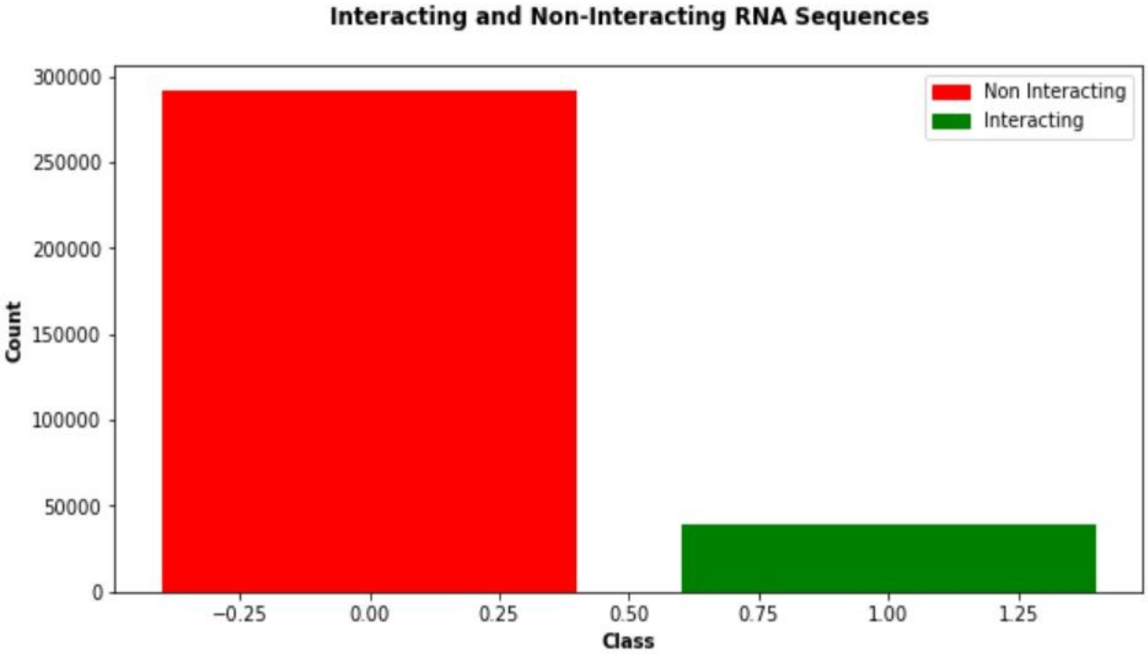
```
import pandas as pd
import imblearn
from imblearn.over_sampling import RandomOverSampler
from sklearn.ensemble import RandomForestClassifier
from xgboost import XGBClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.ensemble import StackingClassifier
from sklearn.pipeline import make_pipeline
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, matthews_corrcoef, confusion_matrix
from sklearn.metrics import roc_curve
from sklearn.metrics import precision_recall_curve
```

b. Preprocessing and Feature Generation:

Since the given data was in String Form and we can not apply a Machine Learning Model on it. To apply Machine Learning model we have to convert the data to numeric datatype. For converting it to numeric datatype we map all the 21 characters to some number.

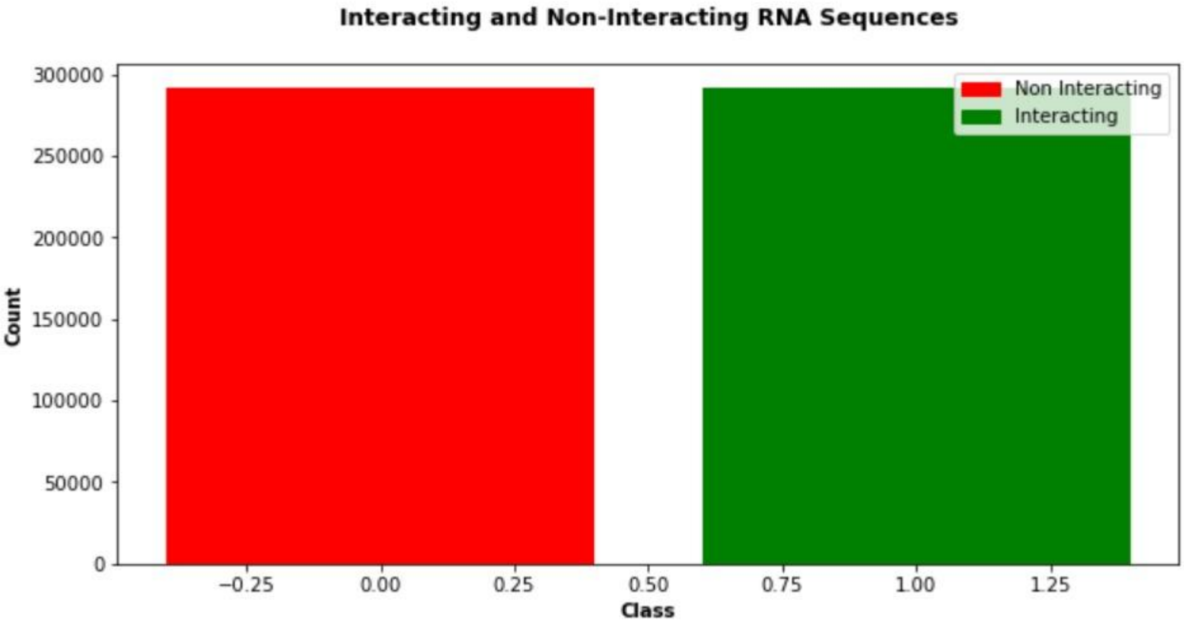
The data set was highly imbalanced, we used Random Oversampling technique on the minority to sample our data, as Imbalanced dataset can give biased results for testing data.

Original Dataset:



Number of columns with class zero : 291963
Number of columns with class one : 38899

After handling data imbalance problem:



Number of columns with class zero : 291963
Number of columns with class one : 291963

c. Models:

For training our model we used 100% training data.

1. Random Forest Classifier:

Random Forest Classifier is an ensemble learning method with bagging technique which combines the result of multiple predictions. Multiple decision trees are combined to form a Forest and each decision tree gives votes to find the best decision on the most popular class. The final result of a particular sequence is given to the class which got maximum votes by averaging all the votes.

2. **XGBoost Classifier:**

   XGBoost or eXtreme Gradient Boosting is also an ensemble learning method which implements gradient boosting machines. Some of the advantages of using XGBoost can be that it is Highly flexible and it implements Tree Pruning Methodology.

3. **Hybrid Model :**

   Here we used a pipelining methodology or Stacking Classifier with two different Classifiers, Random Forest and XG Boost. Stacking Classifier takes the predictions of each individual model and predicts the final result. The main advantage of using a Stacking Classifier is Stacking different types of models increases their strength.

3. <u>Learning Outcome:</u>

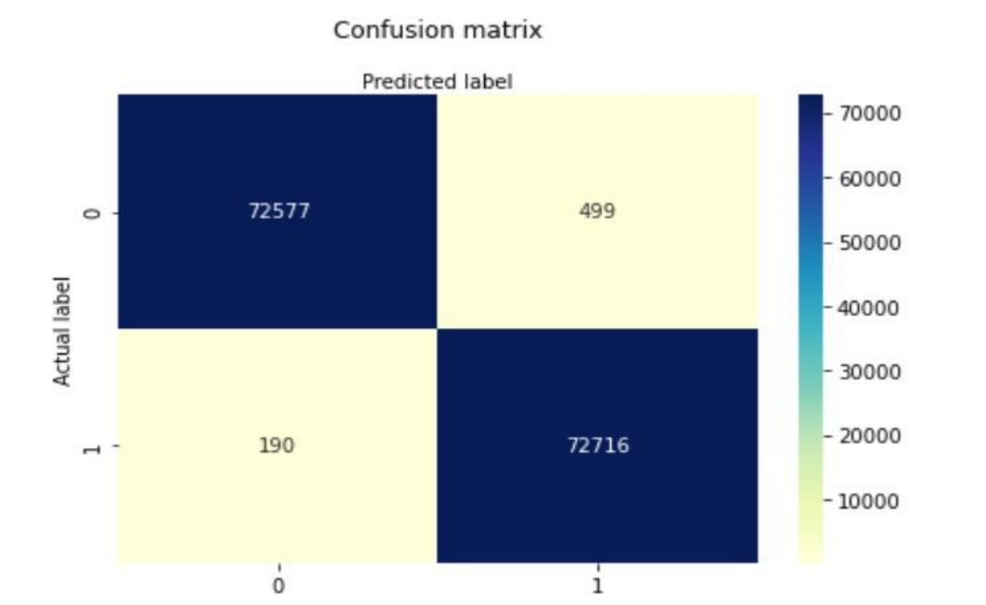
1. Encoding Nominal data to numeric data.
2. Handling imbalanced data.
3. Use of Hybrid Models (Pipeline to combine different models).
4. Team Work.

4. <u>Result:</u>

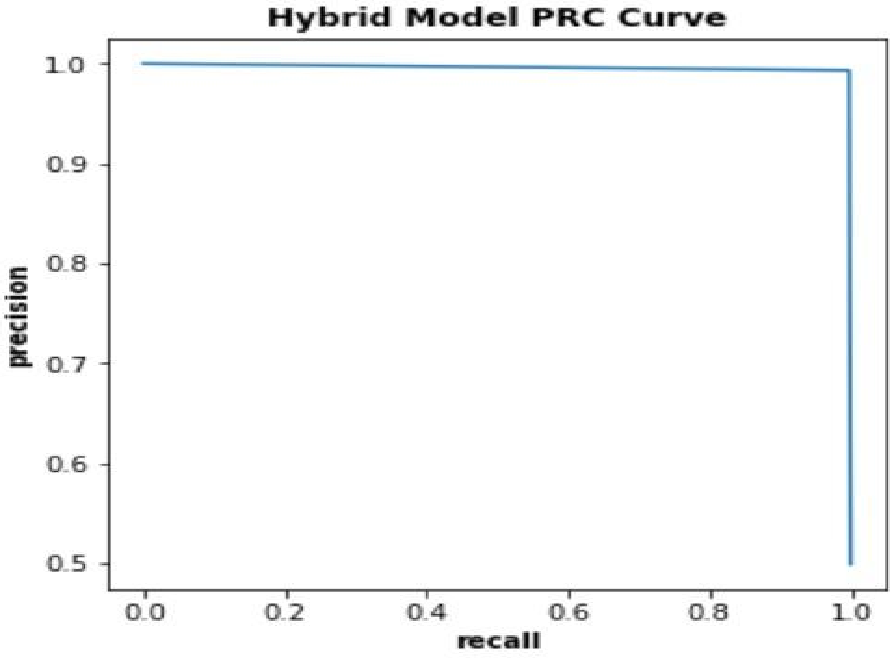
<u>Split Training Data into X_train (75%) and X_test (25%)</u>

Model	**MCC**	**Accuracy**
Random Forest	0.9866425309902978	0.993300543902673
XGBoost	0.49665043885262095	0.7483251359756682
Hybrid (XGBoost + Random Forest)	0.9905693900961009	0.9952802400295927

<u>Confusion Matrix for Hybrid Model (75% training data and 25% testing data)</u>

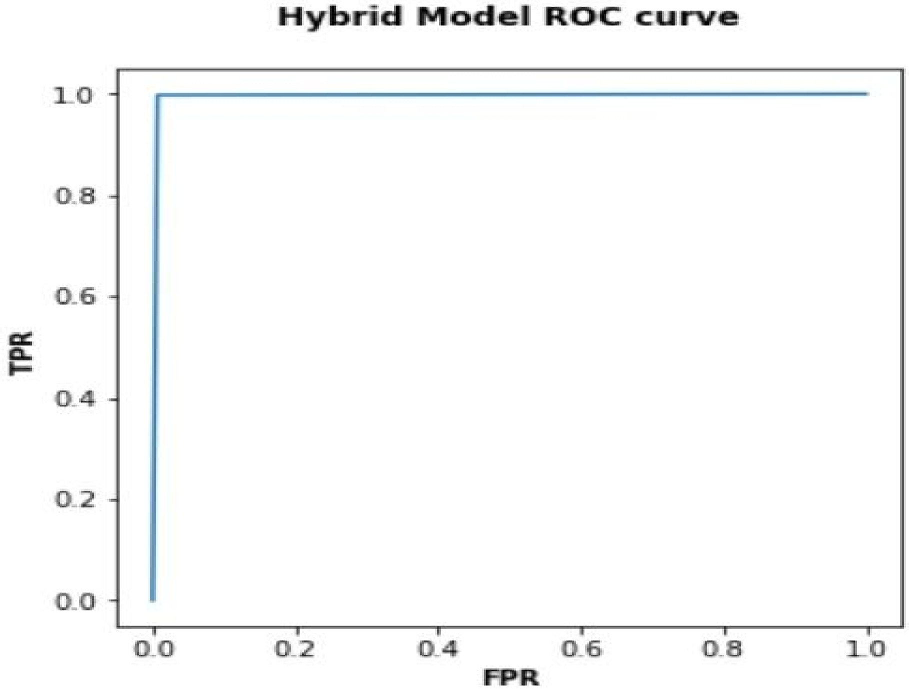


<u>Precision Recall Curve for Hybrid Model (75% training data and 25% testing data)</u> :



-

ROC Curve for Hybrid Model (75% training data and 25% testing data) :



Final Result (On Training Data):

Model	**MCC**
Random Forest	84.787
Hybrid (XGBoost + Random Forest)	85.380

Maximum MCC Score was noted for Hybrid Model, therefore we used it for our final Kaggle submission.