**Methodology**

**Data Preprocessing**

We are using preprocessed data. So doesn’t need to perform any filter operation to remove noise and artifacts. This analysis based on depressive disorder and healthy control so we used on depressive disorder and healthy control participants. We found totally 266 depressed and 95 healthy control participants.

**Table :** Total participants data frame.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S\_No** | **Sex** | **Age** | **EEG Date** | **Education** | **IQ** | **Main Disorder** |
| 1 | M | 57 | 2012.8.30 | NA | NA | Addictive disorder |
| 2 | M | 37 | 2012.9.6 | 6 | 120 | Addictive disorder |
| 3 | M | 32 | 2012.9.10 | 16 | 113 | Addictive disorder |
| 4 | M | 35 | 2012.10.8 | 18 | 126 | Addictive disorder |
| 5 | M | 36 | 2012.10.18 | 16 | 112 | Addictive disorder |
| … | … | … | … | … | … | … |
| 941 | M | 22 | 2014.8.28 | 13 | 116 | Healthy control |
| 942 | M | 26 | 2014.9.19 | 13 | 118 | Healthy control |
| 943 | M | 26 | 2014.9.27 | 16 | 113 | Healthy control |
| 944 | M | 24 | 2014.9.20 | 13 | 107 | Healthy control |
| 945 | M | 21 | 2015.10.23 | 13 | 105 | Healthy control |

**Table :** Selected Participants data frame.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S\_No** | **Sex** | **Age** | **EEG Date** | **Education** | **IQ** | **Main Disorder** |
| 90 | 'F' | 32 | '2015.9.21' | 16 | 108 | 'Depressive disorder' |
| 91 | 'F' | 20 | '2016.12.9' | 12 | 127 | 'Depressive disorder' |
| 92 | 'F' | 19 | '2015.10.21' | 13 | 113 | 'Depressive disorder' |
| 93 | 'F' | 39 | '2017.3.3' | 16 | NaN | 'Depressive disorder' |
| 94 | 'F' | 28 | '2017.3.20' | 9 | NaN | 'Depressive disorder' |
| … | … | … | … | … | … | … |
| 941 | 'M' | 22 | '2014.8.28' | 13 | 116 | 'Healthy control' |
| 942 | 'M' | 26 | '2014.9.19' | 13 | 118 | 'Healthy control' |
| 943 | 'M' | 26 | '2014.9.27' | 16 | 113 | 'Healthy control' |
| 944 | 'M' | 24 | '2014.9.20' | 13 | 107 | 'Healthy control' |
| 945 | 'M' | 21 | '2015.10.23' | 13 | 105 | 'Healthy control' |

**Label Datastore**

The column for age, gender, IQ, and serial number are removed from this data frame also null values are removed, and labeled the data.

Depressed disorder : 1

Healthy control : 2

**Table :** label Datastore.

|  |  |
| --- | --- |
| **S\_No** | **Main Disorder** |
| 90 | 1 |
| 91 | 1 |
| 92 | 1 |
| 95 | 1 |
| … | … |
| 942 | 2 |
| 943 | 2 |
| 944 | 2 |
| 945 | 2 |

**Feature Datastore**

Extract the needed frequency band columns and use those for the analysis. We use alpha, beta, delta, and theta frequency band column. Because higher frequency carry noise which can impact on data.

**Table :** Feature Datastore.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S\_NO | AB.A.delta.a.FP1 | AB.B.theta.a.FP1 | AB.D.beta.a.FP1 | AB.C.alpha.a.FP1 | … |
| 90 | 12.1591 | 7.8297 | 24.8774 | 4.2565 | … |
| 91 | 12.4044 | 7.8046 | 37.4508 | 7.1577 | … |
| 92 | 16.5731 | 19.117 | 3.77856 | 4.1292 | … |
| … | … | … | … | … | … |
| 944 | 19.9291 | 13.7235 | 11.1776 | 89.7132 | … |
| 945 | 65.1953 | 35.7003 | 11.2560 | 56.3250 | … |

**Table :** Feature Datastore. (cont…)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S\_NO | COH\_A\_delta\_a\_FP1\_b\_FP2 | COH\_A\_theta\_a\_FP1\_b\_FP2 | COH\_A\_alpha\_a\_FP1\_b\_FP2 | COH\_A\_beta\_a\_FP1\_b\_FP2 | … |
| 90 | 69.2084 | 83.9770 | 97.4037 | 87.5097 | … |
| 91 | 80.9438 | 82.0956 | 88.6046 | 76.6430 | … |
| 92 | 89.5805 | 87.8633 | 97.6039 | 84.3118 | … |
| … | … | … | … | … | … |
| 944 | 99.3858 | 99.6145 | 99.9189 | 98.5896 | … |
| 945 | 65.4787 | 82.6571 | 92.0473 | 80.7158 | … |

**Classification Model**

For the classification of each depressive disorder based on feature that were taken from EEG data two model are created.

* Random Forest
* XGBoost

**Random Forest**

The Random Forest model is a supervised machine learning model that is used for both classification and regression problems. It creates decision trees based on sample sample-selected testing set and predicts output from each decision tree. From all of the output, the majority vote was taken an provided as final result.

Equation for Single decision tree:

Where,

* F(x) is the predicted output
* N is number of terminal nodes (leaves in the tree)
* Ri us the region (leaf) to which the input x belongs
* is the weight assign to each leaf
* || is the indicator function (1 is x is in Ri, 0 otherwise)

Random Forest Prediction:

Where,

* M is the number of trees
* is the prediction of the j-th tree

Data

Split data

Training

Testing

Train Model using parameter

Creating decision treess

Predict using test data

Taking majority

Final Result

Figure: Overview of Random Forest Model.

**Table :** Parameter of Random Forest Model. []

|  |  |
| --- | --- |
| **Parameter** | **Values** |
| Number of estimation | [100,300,500] |
| Maximum depth of the tree | [1,3,6] |

**Algorithm:**

The step of the Random Forest model algorithm.

* Step 1: Take data as input.
* Step 2: Split data into two part. (Training, and Testing).
* Step 3: Train model using training data.
* Step 4: Creating decision trees based on parameter.
* Step 5: Predicting each decision tree output with testing data.
* Step 6: Get the majority vote form the predicting data.
* Step 7: provide the final result with accuracy.

**XGBoost**

A distributed, scalable gradient-boosted decision tree technique that is best for classification and regression problems. This provides better accuracy to classification-based problems. This can handle large datasets efficiently. This creates multiple decision trees. This teaches more errors provided by each decision tree and makes correct errors. This combines all weak learners to make a strong one.

Objective function for XGBosot:

Here,

* N is the number of training examples
* is the true label of th i th example
* is the predicted output
* L is the loss function measuring the difference between and
* K is the number if trees
* is the regularization term for the k-th tree.

Tree Prediction:

Here,

* is the prediction of the k-th tree

Data

Split data

Training

Testing

Train Model using parameter

Creating decision treess

Classifiers

Classification Results

Figure: Overview of XGBoost model.

**Table :** Parameter of XGBoost Model. []

|  |  |
| --- | --- |
| **Parameter** | **Values** |
| Number of estimation | [100,300,500] |
| Sub-sample | [0.3, 0.5, 1] |
| Maximum depth of the tree | [1,3,6] |

**Algorithm:**

The step of the XGBoost model algorithm.

* Step 1: Take data as input.
* Step 2: Split data into two part. (Training, and Testing).
* Step 3: Train model using training data.
* Step 4: Creating decision trees based on parameter.
* Step 5: Predicting decision tree output with testing data.
* Step 6: combine all the prediction as weak learner and make a strong one.
* Step 7: provide the final result with accuracy.

**Evaluation of Training Models**

The following matrics are being considered for evaluation of trained model.

* Accuracy Score
* Precision
* Recall
* F1 Score
* Confusion matrix
* ROC curve

**Accuracy:** Ration of correct predictions to the total number of predictions, and this represents how often the classifier makes correct predictions.

Here, equation (1) relates to an equation for accuracy, which expresses the portion of correctly classified data instances to all other data instances.

**Precision:** Precision is the ratio of correctly predicted positive observation to the total predicted positives. This also known as positive predictive value.

The precision equation is shown in equation (2).

**Recall:** Recall is the ratio of correctly predictive observations to the all observations in the actual class. It is also known as sensitivity or True positive rate.

The recall equation shown in equation (3).

**F1 Score:** The harmonic mean of recall and precision. This provide a balance between precision and recall, specially when there is an uneven call distribution.

The F1 score equation shown in equation (4)

**ROC Curve:** The ROC curve is a graphical representation of the trade off between true positive rate and false positive rate for different classification thresholds.

**Confusion Matrix:** A confusion matrix is a table that describe the performance of classification model. It presents a summary of prediction against actual class labels using True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN).

**Model Training Results**

**Random Forest**

**Table :** Evaluation of Random Forest model.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1 Score** |
| Random Forest | 74.31% | 0.74 | 1.00 | 0.82 |

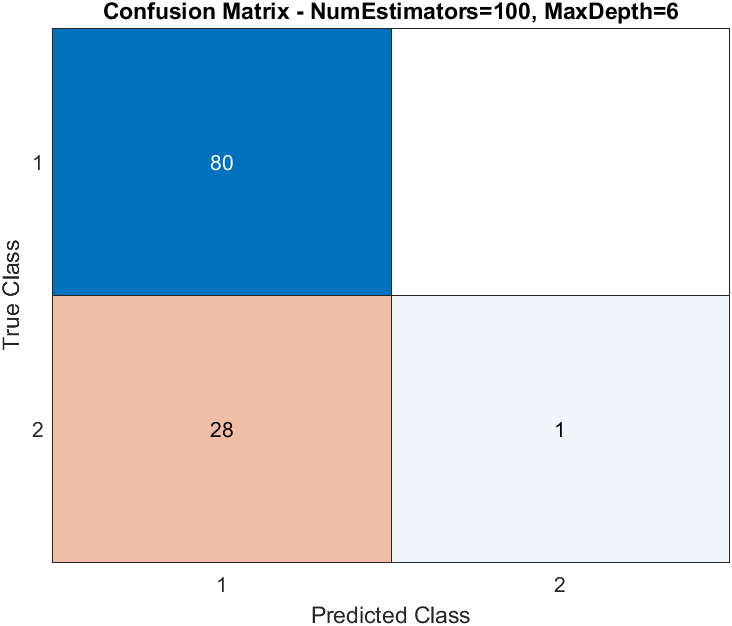


Figure : Confusion matrix of Random Forest model.

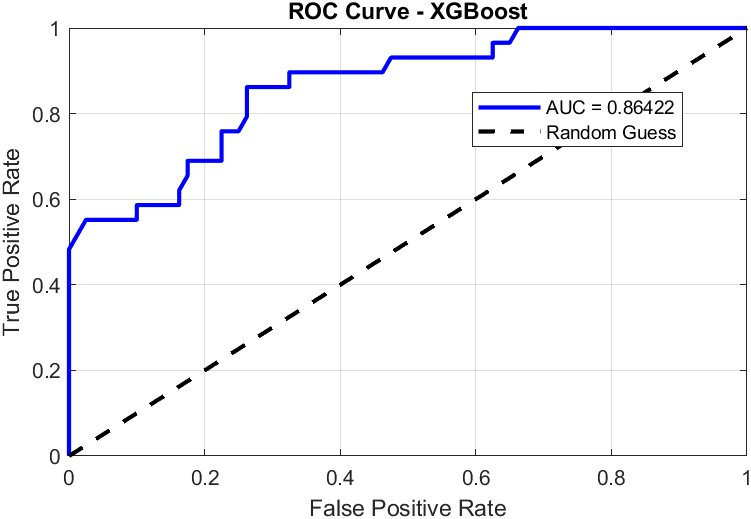


Figure : ROC curve of Random Forest model.

**XGBoost**

**Table :** Evaluation of XGBoost model.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **Accuracy** | **Precision** | **Recall** | **F1 Score** |
| XGBoost | 82.57% | 0.81 | 1.00 | 0.89 |

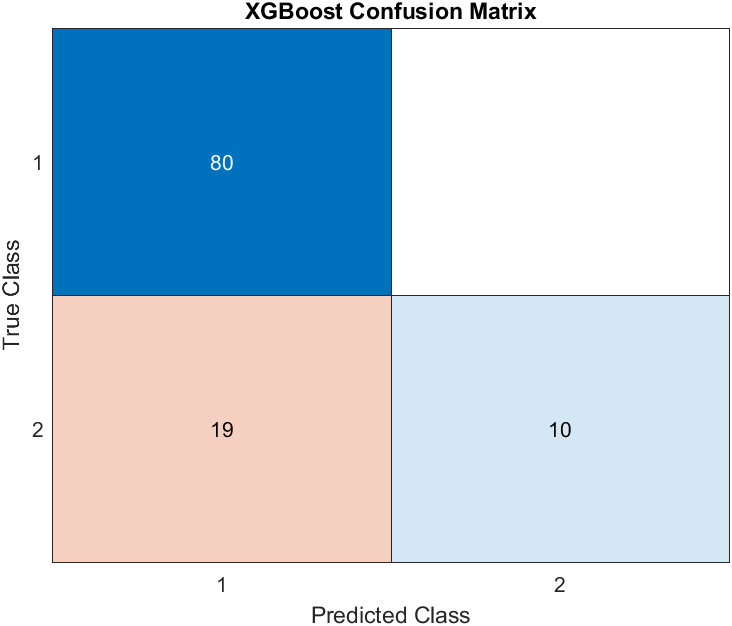


Figure : Confusion matrix of XGBoost Model.

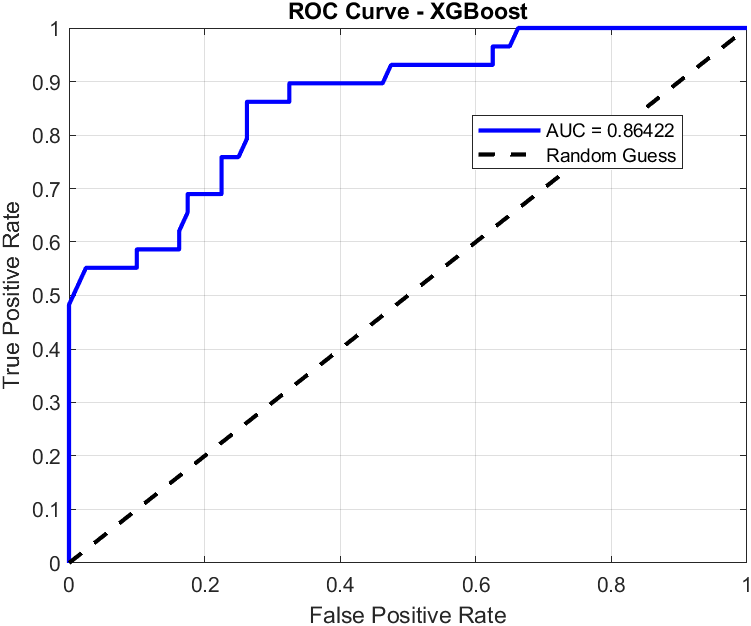


Figure: ROC Curve of XGBoost Model.