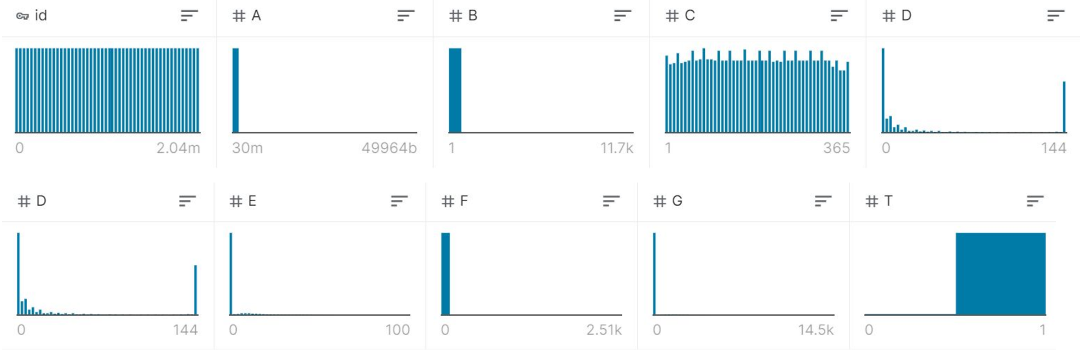
DMG Assignment 2

Arjun Lakhera 2018133
Daksh Thapar 2018137

Methodology:



Distribution of given dataset

1. The given dataset was extremely skewed in favour of Class with Class label 1(ratio 1: ~69, i.e. Class 1: 2012677, Class 0:29010). To deal with this problem, we went through different sampling methods and ensemble learners.

2. Before doing this, we removed outliers in the dataset using MinMaxScaler to normalize data followed by KNN (Since the scale of data features was vastly distributed). (Multivariate Outlier Analysis). We kept our outlier_fraction=0.03 to remove 3% of data as outliers.

3. We used RandomUnderSampler to resample the data with only twice the number of majority class data points (Class 1: 58020, Class 0: 29010)

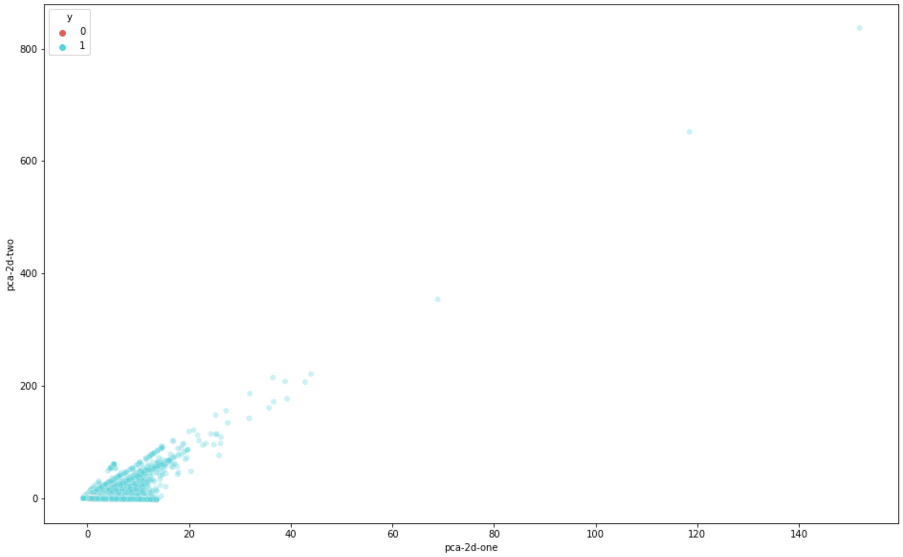
4. We tried the following models after using GridSearchCV to find the best parameters:
    a. Decision Tree Classifier
    b. Random Forest Classifier
    c. Gradient Tree Boosting Classifier (GTB Classifier)
    d. Extreme Gradient Boosting Classifier (XGBoost Classifier)
    e. Extreme Gradient Boosting Regressor (XGBoost Regressor)

5. We used ensemble learning techniques such as Random Forest and XGBoost Classifier/Regressor to get a better fit for the dataset since ensemble learners reduce variance by averaging results obtained from different weak classifiers (Decision Trees in this case).

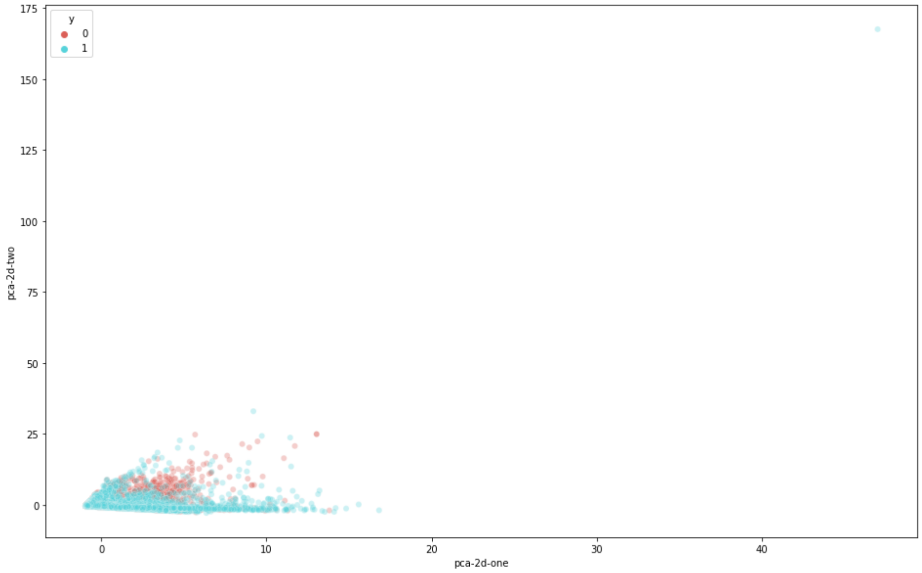
Visualization of Skewness of Dataset

To visualize the given data 2 dimensionally, we applied dimensionality reduction using Principal Component Analysis (PCA). We have applied PCA on our dataset before preprocessing (Class 1: 2012677, Class 0:29010) where the frequency of Class Label 1 dominates the frequency of Class Label 0.=, which is clearly visible in the image below where Class Label 0 points are overshadowed. After preprocessing (outlier removal technique, undersampling), we have data points from both Class Labels 0 and 1 to be of exactly the same frequency and we can see data points of both classes pretty clearly.

- Before preprocessing:

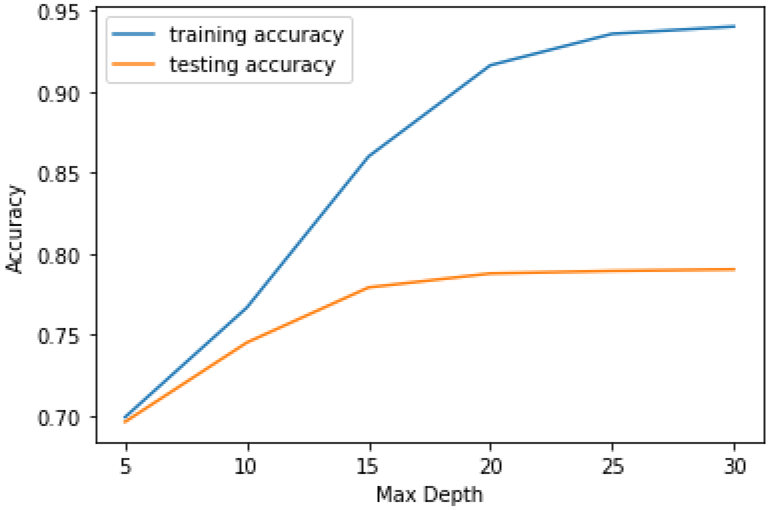


- After preprocessing

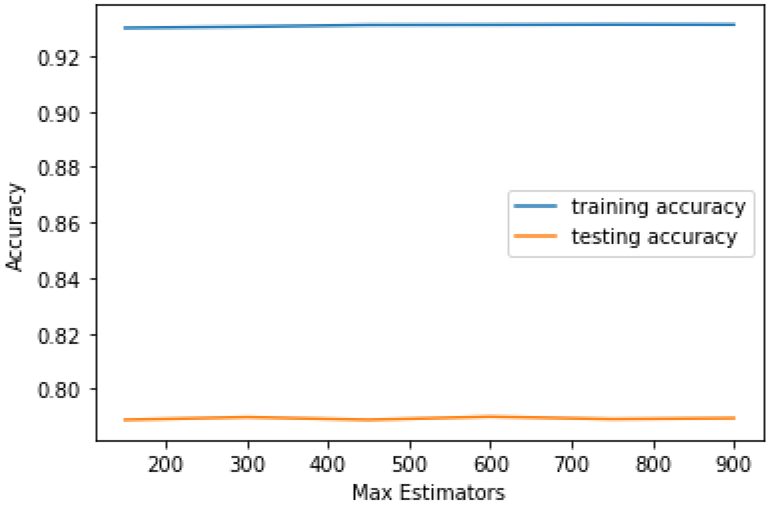


Training and Testing accuracy while varying hyperparameters(Top 3 models):

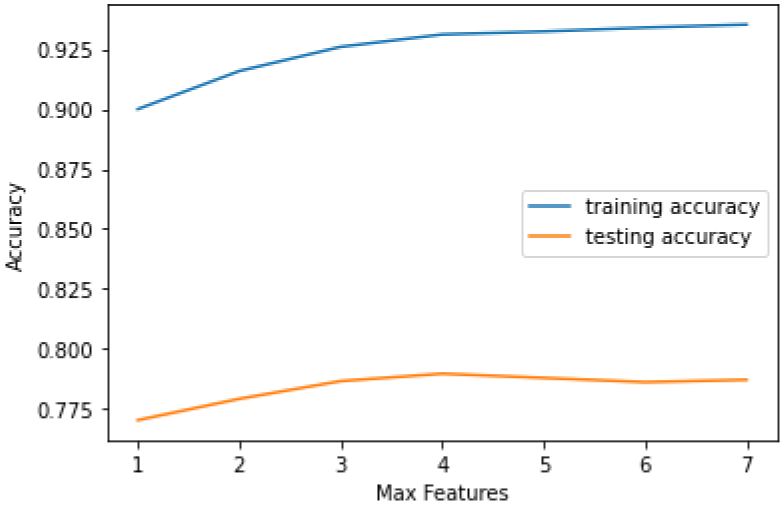
1. **Random Forest Classifier:**
   a) Max depth while keeping others constant:



   b) Max estimators while keeping others constant:

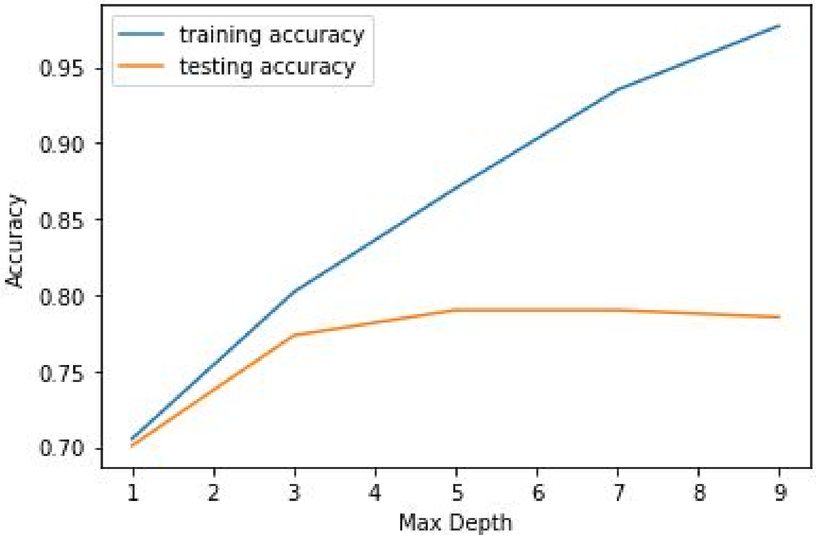


c) Max features while keeping others constant:

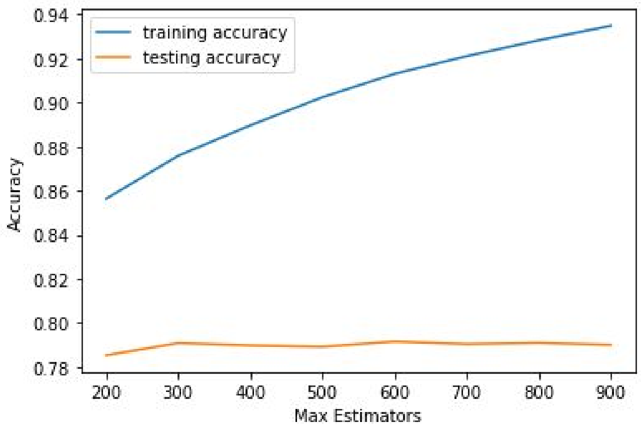


2. **XGboost Classifier:**

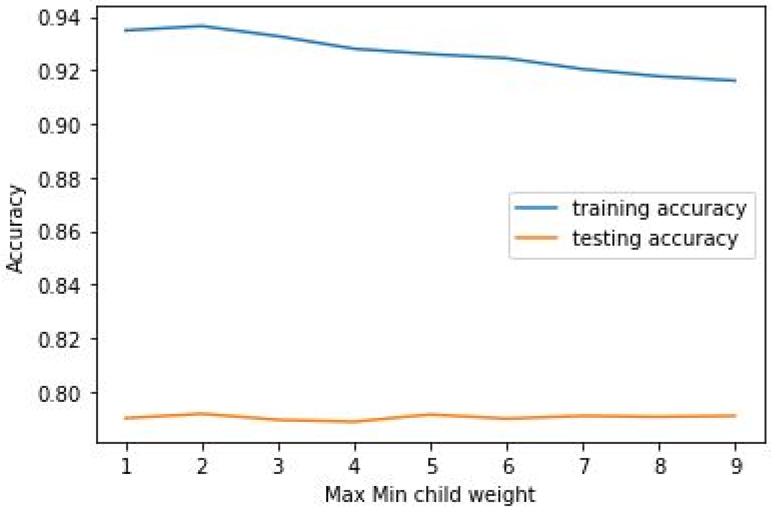
a) Max depth while keeping others constant:



b) Max estimators while keeping others constant:

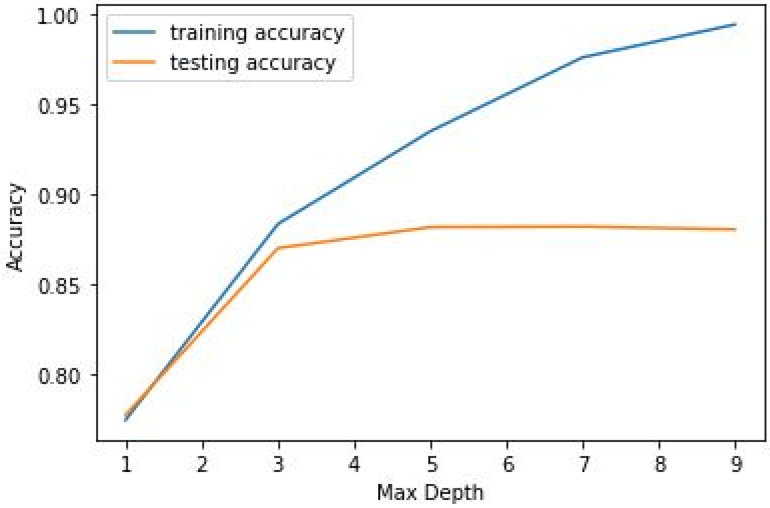


c) Max_min_child_weight keeping others constant:

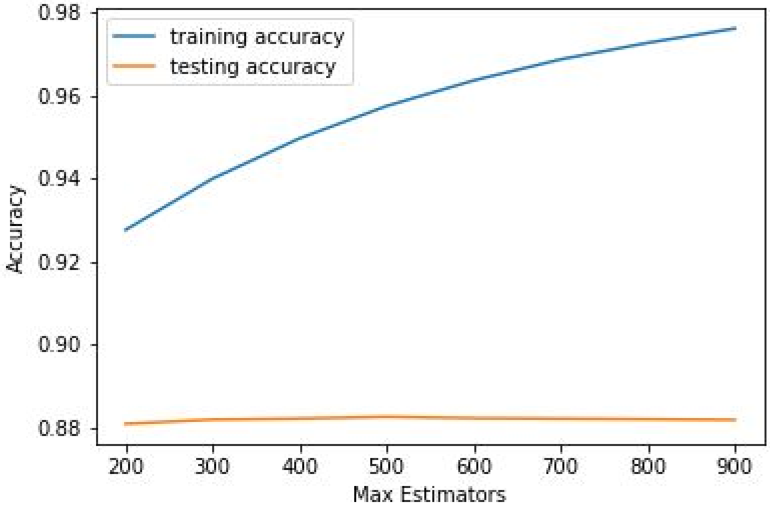


3. **XGBoost Regressor** (Here we used AUC-ROC score for the accuracy):

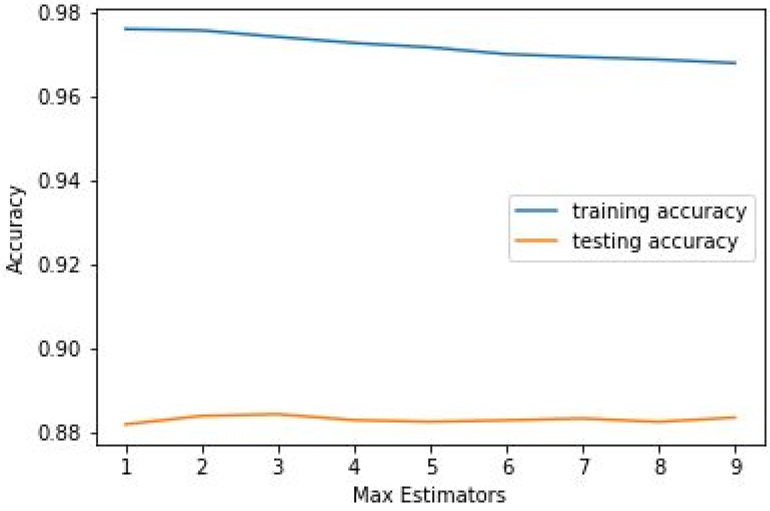
a) Max depth while keeping others constant:



b) Max estimators while keeping others constant:



c) Max_min_child_weight while keeping others constant:



Drive Link for Models, CSV files

https://drive.google.com/drive/folders/1kNZMWK9ty3K-EGRU2XFzZ-L8MUzksaBG?usp=sharing

Learning:

After completing this assignment, we learnt the following-

1. Multivariate Outlier analysis using KNN gives us a benefit by removing ~1000 noisy samples from our dataset, which makes it more efficient for ML models to work on.

2. Handling Highly Imbalanced- dataset Classification problem can be handled by using RandomUnderSampler and Ensemble Techniques like XGBoosting and Random Forests, KNN technique and Decision Trees.

   - For the given dataset, applying XGBRegressor after RandomUnderSampler gives us the best results for AUC ROC (0.88789).

   - XGBClassifier and RandomForestClassifier give the next best AUC ROC scores (0.796 and 0.784 respectively).

   - Decisions Trees offer the worst score for the given dataset (~0.68).

3. HyperParameter Tuning using GridSearchCV from sklearn library increases our score; hence we have run GridSearchCV on all our models and is really useful to optimise all the parameters of a model .

4. We have learnt how to save trained models for future use using Pickle library in Python, which can come in handy when we need to use pre-trained models for different datasets.