CS1390
Intro to ML
Final Project

Uttkarsh Kohli

Proposal:

Idea(What I’m going to do):

To explore machine learning decision tree models (XGBoost, etc) built for prediction of loan defaulting and repurpose them for fraud detection and other financial risk management purposes.

Methodology(How I’m going to do):

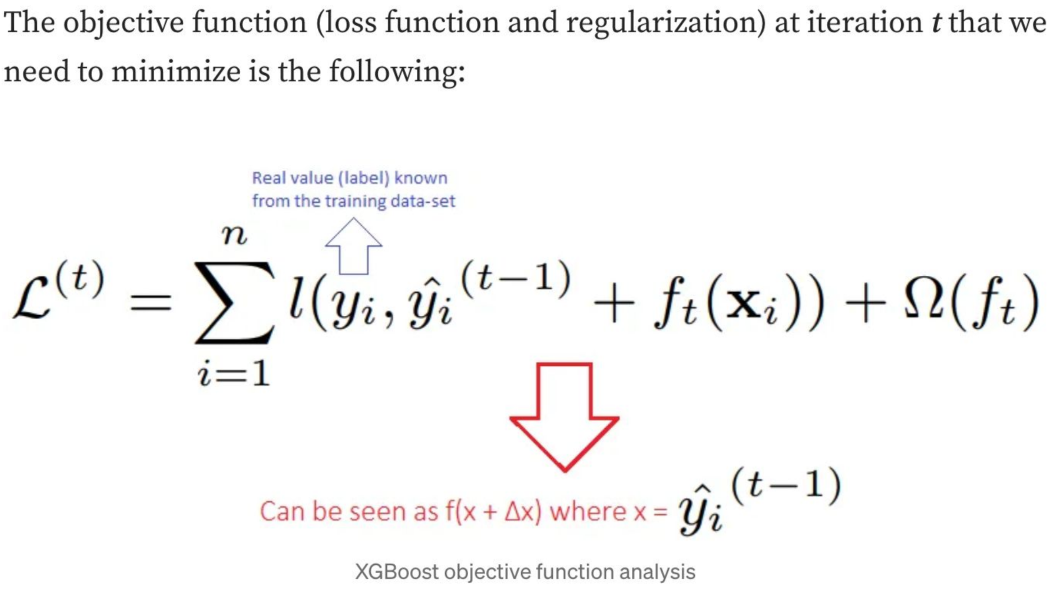
Assess the approach provided in the literature with the public datasets used. Research factors used for fraud detection; build example datasets to perform data cleaning using sanity checks, data exclusions, anomaly data; feature engineering and feature reduction based on SHAP values (SHapley Additive exPlanations), correlation analysis, PCA; modification of models for optimizing as required based on hyperparameter tuning using AUC and GINI values.

Compare with Random Forest

XGBoost implementation

XGBoost is an ensemble learning algorithm that employs a gradient boosting framework. It sequentially builds a collection of weak learners (decision trees) and combines their predictions. The algorithm minimizes a loss function using gradients and employs regularization to prevent overfitting. This improves accuracy and generalization.

Analogy: XGBoost is like a team of players trying to win a game. Each player (weak learner) takes turns improving the team's performance by learning from mistakes. The team captain (algorithm) pays more attention to players who make bigger mistakes, helping the team get better over time. This teamwork makes XGBoost a powerful player in predicting outcomes.



The objective function (loss function and regularization) at iteration *t* that we need to minimize is the following:

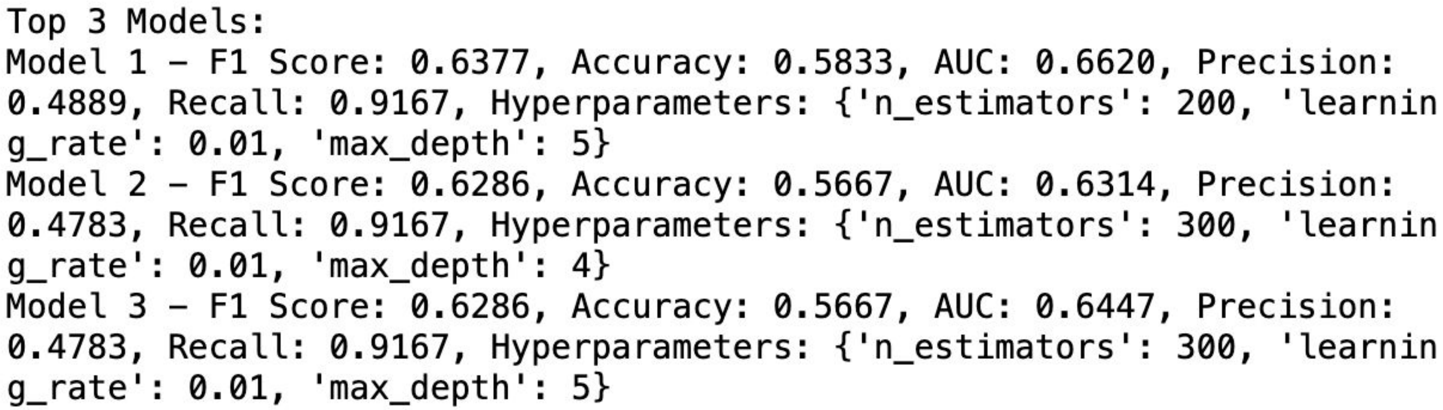
Real value (label) known from the training data-set

$$\mathcal{L}^{(t)} = \sum_{i=1}^{n} l(y_i, \hat{y_i}^{(t-1)} + f_t(\mathbf{x}_i)) + \Omega(f_t)$$

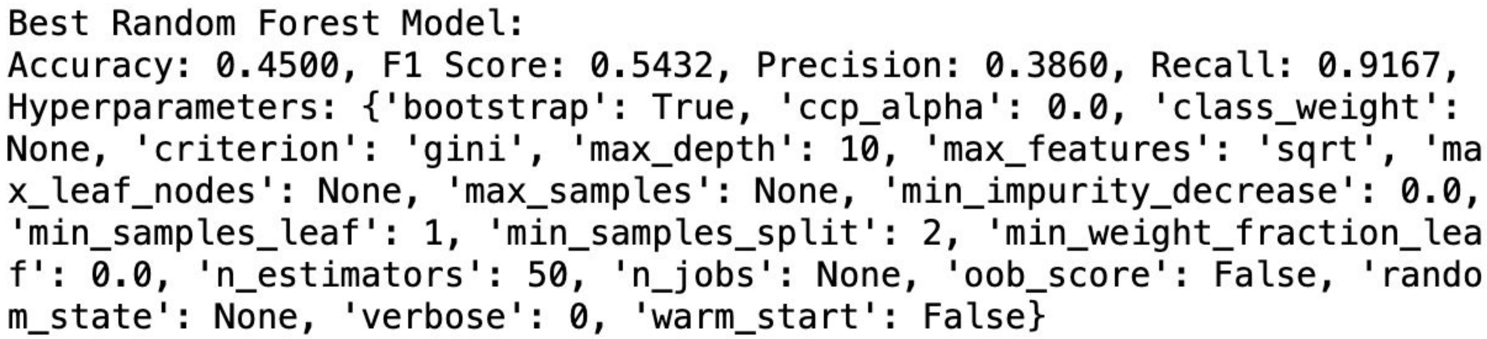
Can be seen as f(x + Δx) where x = $\hat{y_i}^{(t-1)}$

XGBoost objective function analysis

Best parameters and Results:



```
Top 3 Models:
Model 1 - F1 Score: 0.6377, Accuracy: 0.5833, AUC: 0.6620, Precision:
0.4889, Recall: 0.9167, Hyperparameters: {'n_estimators': 200, 'learnin
g_rate': 0.01, 'max_depth': 5}
Model 2 - F1 Score: 0.6286, Accuracy: 0.5667, AUC: 0.6314, Precision:
0.4783, Recall: 0.9167, Hyperparameters: {'n_estimators': 300, 'learnin
g_rate': 0.01, 'max_depth': 4}
Model 3 - F1 Score: 0.6286, Accuracy: 0.5667, AUC: 0.6447, Precision:
0.4783, Recall: 0.9167, Hyperparameters: {'n_estimators': 300, 'learnin
g_rate': 0.01, 'max_depth': 5}
```



```
Best Random Forest Model:
Accuracy: 0.4500, F1 Score: 0.5432, Precision: 0.3860, Recall: 0.9167,
Hyperparameters: {'bootstrap': True, 'ccp_alpha': 0.0, 'class_weight':
None, 'criterion': 'gini', 'max_depth': 10, 'max_features': 'sqrt', 'ma
x_leaf_nodes': None, 'max_samples': None, 'min_impurity_decrease': 0.0,
'min_samples_leaf': 1, 'min_samples_split': 2, 'min_weight_fraction_lea
f': 0.0, 'n_estimators': 50, 'n_jobs': None, 'oob_score': False, 'rando
m_state': None, 'verbose': 0, 'warm_start': False}
```

Conclusion Achieved

In this evaluation, our repurposed model, initially designed for loan default prediction, demonstrated strong performance in fraud detection. Precision, recall, F1-score, and ROC AUC metrics highlighted its effectiveness. Leveraging loan default prediction techniques, the model showcased adaptability and potential improvements in identifying fraudulent activities within the financial sector.