Introduction to XGBoost

Extreme Gradient Boosting

Ankit Shrestha

What is XGBoost?

- A machine learning algorithm based on decision trees.
- Uses boosting: builds models one after another, each trying to correct mistakes of the previous one.
- “Extreme” = optimized for speed and performance.

Why popular?

- Very fast and scalable.
- Works well with both small and large datasets.
- Often gives state-of-the-art accuracy in Kaggle competitions and industry.

How Does XGBoost Work?

1. Start with a base prediction (e.g., mean value for regression, or 0.5 probability for classification).
2. Build the first decision tree to predict errors (residuals).
3. Update predictions = old prediction + learning rate × new tree’s prediction.
4. Repeat for many trees.
5. Final prediction = sum of all trees.

Let’s Take an Example (Classification)

Imagine we want to predict if a student will Pass or Fail based on Study Hours.

1. Start: assume everyone has a 50% chance to pass.
2. First tree: sees that students with >5 hours tend to pass → adjusts prediction.
3. Second tree: focuses on mistakes (e.g., students who studied 6 hrs but still failed).
4. Third tree: corrects remaining errors.

After several trees, predictions become very accurate.

Let’s Take an Example (Regression)

Predict house prices based on features (size, location, rooms).

1. Start with mean house price.
2. First tree: predicts adjustments (e.g., bigger houses = +$50k).
3. Second tree: fixes mistakes (e.g., large houses in bad locations shouldn’t add too much).
4. Keep adding trees until errors are minimized.

Advantages

- High accuracy.
- Fast (parallel computation).
- Handles missing values automatically.
- Works well with both structured/tabular data and large datasets.
- Built-in regularization to avoid overfitting.

Disadvantages

- Can be complex to tune (many hyperparameters).
- Less interpretable than simple models.
- May overfit if not tuned properly.

Sample Training on XGBoost

	precision	recall	f1-score	support
0.0	0.99	0.98	0.98	1490
1.0	0.93	0.94	0.94	373
accuracy			0.97	1863
macro avg	0.96	0.96	0.96	1863
weighted avg	0.97	0.97	0.97	1863

ROC-AUC: 0.9964805585044173

After Hyperparameter Tuning

Parameter	Value
subsample	0.7103165563345655
colsample_bytree	0.7047898756660642
gamma	0.1154469128110745
learning_rate	0.05820509320520235
max_depth	7
n_estimators	413

Final Model Performance on Test Set:

	precision	recall	f1-score	support
0.0	0.99	0.98	0.98	1490
1.0	0.93	0.95	0.94	373
accuracy			0.98	1863
macro avg	0.96	0.96	0.96	1863
weighted avg	0.98	0.98	0.98	1863

Best_cv_roc_auc : 0.9966783959502696

Feature Selection using SHAP

```
Top 10 Most Important Features (by absolute SHAP value):
============================================================
amt                                 :  -1.4888
hour                                :  -1.2482
category_food_dining                :  -0.6577
category_gas_transport              :  -0.5659
trans_day_cycle_night               :  -0.5431
city_pop                            :  -0.5398
trans_amt_range_small               :  -0.5340
trans_day_cycle_evening             :  -0.4793
age_during_trans                    :  -0.4174
lat                                 :  -0.3656
```

Feature Selection using SHAP

```
Top 5 Features Pushing TOWARDS Fraud (positive SHAP values):
============================================================
category_shopping_pos          : + 0.0772
zip                            : + 0.0438
category_shopping_net          : + 0.0402
category_misc_net              : + 0.0248
merch_long                     : + 0.0133

Top 5 Features Pushing AWAY from Fraud (negative SHAP values):
============================================================
amt                            :  -1.4888
hour                           :  -1.2482
category_food_dining           :  -0.6577
category_gas_transport         :  -0.5659
trans_day_cycle_night          :  -0.5431
```

Final Random Forest Test Performance:

	precision	recall	f1-score	support
0.0	0.95	0.94	0.94	1490
1.0	0.76	0.80	0.78	373
accuracy			0.91	1863
macro avg	0.86	0.87	0.86	1863
weighted avg	0.91	0.91	0.91	1863

Final LightGBM Test Performance:

	precision	recall	f1-score	support
0.0	0.99	0.99	0.99	1490
1.0	0.94	0.94	0.94	373
accuracy			0.98	1863
macro avg	0.96	0.96	0.96	1863
weighted avg	0.98	0.98	0.98	1863

Thank You !