Concept Note: <u>Predicting Uganda Treasury Bill Auction Yields</u>

1. Background

The Bank of Uganda (BoU) conducts weekly Treasury Bill (T-bill) auctions for 91-day, 182-day, and 364-day instruments. These are key benchmarks for Uganda's fixed income market, impacting both government borrowing costs and investor strategies.

Currently, auction outcomes (cut-off yields, bid-to-cover ratios, total bids) are only available post-auction. For portfolio managers, traders, and analysts, the ability to forecast auction yields before the event would provide a competitive edge in market participation, asset allocation, and liquidity management.

This project seeks to build a predictive model for Uganda T-bill auction yields, using historical auction data combined with secondary market indicators and macro variables.

2. Objectives

Primary Objective:

Develop and validate predictive models for auction cut-off yields (in basis points) for 91d, 182d, and 364d T-bills.

Secondary Objectives:

01. Test multiple model families (linear, tree-based, etc.) for predictive power.
02. Evaluate performance via backtests (expanding-window cross-validation).
03. Provide interpretable outputs (feature importance, sensitivity to bid-to-cover, macro drivers).
04. Generate scenario forecasts for upcoming auctions (e.g., given different offer sizes or demand assumptions).

3. Data Sources

Two core datasets will be prepared and maintained:

01. Auction Results Dataset (historical, per tenor per auction date):

- Auction date
- Tenor (91, 182, 364 days)
- Amount offered (UGX bn)
- Total bids received (UGX bn)
- Bid-to-cover ratio
- Cut-off yield (bp)

- Weighted average yield (bp)

02. **Secondary Market + Macro Dataset (daily or weekly, rolled to auction dates):**

- Secondary market indicative yields (91d, 182d, 364d, bp)
- Secondary market turnover (7-day cumulative, UGX bn)
- Interbank 7-day rate (bp)
- BoU policy rate (bp)
- Inflation (CPI YoY, %)
- FX (USD/UGX 30-day % change)
- Optional: T-bond yields, fiscal issuance calendar, liquidity operations, global rates

4. Methodology

Step 1 – Data Engineering

- Clean and align auction + secondary datasets.
- Forward-fill secondary data to auction dates.
- Create lagged and rolling features:
- Past cut-off yields, bid-to-cover ratios, and weighted average yields.
- Rolling averages (3–5 auction windows).
- Curve slope indicators (91 vs 364, 182 vs 364).
- Liquidity proxies (e.g., turnover z-score).
- Calendar dummies (month, quarter, year-end).

Step 2 – Model Development

- Baseline Models: Ridge regression, Lasso/Elastic Net.
- Tree-based Models: Gradient Boosting, Random Forest, XGBoost/LightGBM.
- Benchmark: Simple lag model (last auction yield = prediction).

Step 3 – Validation & Backtesting

- Expanding-window cross-validation (per tenor).
- Error metrics: Mean Absolute Error (MAE, in bp) and directional accuracy.
- Compare performance across models.

Step 4 – Forecasting & Scenarios

- Fit on full historical dataset.
- Produce next-auction forecasts per tenor.
- Allow scenario input (e.g., change in offer amount, total bids, BTC) to see elasticity of predicted yields.

5. Deliverables

01. Cleaned Historical Datasets: Auction + secondary market panels.

02. Modeling Pipeline (Python/Notebooks):

- Data ingestion & feature engineering.
- Model training, CV backtests, charts.
- Next-auction forecast generator.

03. Backtest Results: CSVs + plots showing actual vs predicted yields.
04. Scenario Simulator: Function/script to test how changes in bids/offer affect yields.
05. Documentation & Handover: Explanation of methodology, feature set, and usage guide.

6. Expected Benefits

- Improved pre-auction planning for investors and traders.
- Insights into key drivers of auction yields (demand, liquidity, macro).
- Foundation for expanding into T-bond yield predictions and curve modeling.
- A reproducible data science framework for ongoing updates.

7. Next Steps

01. Populate auction + secondary data templates with historical data.
02. Run initial model training on at least 2–3 years of data.
03. Compare baseline vs tree-based models on backtests.
04. Deploy model outputs into reporting dashboards (e.g., Impala Market blog + data platform).