Wallet Risk Scoring for Compound Protocol

Project Workflow Report: Wallet Risk Scoring for Compound Protocol

Overview

This initiative aimed to create a comprehensive risk scoring framework for wallets engaging with the Compound protocol on the Arbitrum network. By analyzing on-chain behavior, each wallet was assigned a risk score ranging from 0 to 1000 using public data sourced via subgraphs from The Graph protocol.

1. Data Acquisition and Schema Analysis

Subgraphs Accessed: Both Compound V2 and V3 subgraphs were evaluated to collect transaction-level and position-specific data.

Schema Review:
- Two dedicated Python scripts (`introspection_query.py` for V2 and `introspection_query_v3.py` for V3) were used to extract and examine schema structures in JSON format.
- These schemas provided the necessary blueprint to construct precise GraphQL queries.

Challenges Encountered:
- Among four tested archival subgraphs, only one V2 subgraph returned complete data for the 100 target wallets.
- No Compound V3 subgraph contained usable information for these addresses, prompting a shift to a V2-only approach for the rest of the analysis.

2. Raw Data Collection

- A script named `compound_query.py` was developed to pull relevant wallet data based on schema learnings.
- This script queried transactional and positional information for all 100 wallets.
- The collected data was saved in a CSV file (`compound_wallets_raw.csv`) which served as the foundation for further processing.

Wallet Risk Scoring for Compound Protocol

3. Feature Engineering

From the raw dataset, the `feature_engineering.py` script derived 17 key features including:

- `total_supplied_usd`, `total_borrowed_usd`, `collateralization_ratio`, `repayment_rate`, `liquidations_suffered`, `withdraw_to_supply_ratio`, and others.

These variables encapsulated wallet behavior such as lending/borrowing dynamics, protocol engagement, risk exposure, and repayment patterns.

4. Data Cleaning and Transformation (Executed on Google Colab)

Exploratory Data Analysis (EDA):

- Investigated feature distributions, outliers, and correlations.

Preprocessing Steps:

- Clipped values outside the 5th and 95th percentile to handle outliers.
- Applied log transformations to heavily skewed financial features.
- Removed features that were either constant or overly correlated, while keeping crucial metrics like `collateralization_ratio` and `repayment_rate`.
- For critical zero entries, minimal positive values were substituted to avoid downstream errors.

The final cleaned dataset was saved as `engineered_features_cleaned.csv`.

5. Heuristic Risk Scoring

A domain-driven, rule-based function was built to generate initial risk scores by:

- Weighting normalized features tied to factors such as liquidation, repayment behavior, protocol activity, collateral safety, diversification, and borrowing trends.
- Scores were scaled from 1 (most secure) to 1000 (most risky).

These scores were appended to the dataset, producing `engineered_features_with_scores.csv`.

6. Model Training and Evaluation (on Google Colab)

Wallet Risk Scoring for Compound Protocol

Three machine learning regression models were trained using heuristic scores as targets:

- XGBoost Regressor
- LightGBM Regressor
- Random Forest Regressor

Each model was validated on standard metrics like RMSE, MAE, and R^2. XGBoost outperformed the others in terms of prediction accuracy and model transparency.

7. Final Model Deployment and Prediction

- The full feature set and heuristic labels were used to train a final XGBoost model.
- This model predicted refined risk scores for all 100 wallets.
- The results were stored in `final_predictions.csv`, representing a hybrid risk estimation combining domain expertise and machine learning insights.