Final Project Report

Cryptocurrency Liquidity Prediction

Contents

1 Project Overview

The goal of this project is to develop a machine learning model that predicts the liquidity (using price) of cryptocurrencies based on historical market data. The prediction helps stakeholders understand and respond to potential liquidity risks in volatile crypto environments.

2 Workflow Summary

The end-to-end ML pipeline includes the following stages:

1. **Data Collection:** Historical data scraped from CoinGecko (CSV format).
2. **Preprocessing:** Cleaned missing values, standardized formats.
3. **EDA:** Trends, correlations, seasonal patterns, and distribution analysis.
4. **Feature Engineering:** Created rolling means, time-based, and percentage change features.
5. **Model Selection:** Compared multiple regressors (XGBoost, RandomForest, Ridge, etc.).
6. **Model Training:** Tuned XGBoost selected and trained using hyperparameter optimization.
7. **Evaluation:** Metrics used — MAE, RMSE, and R^2 score.
8. **Deployment:** A Streamlit app deployed locally for real-time predictions.

3 EDA Highlights

- **Skewness:** Target column (‘price‘) and volume-based features showed right skew.
- **Correlations:** Moderate correlation between price and market cap, and volume.
- **Outliers:** Present across most features, especially ‘price‘, ‘volume‘, and ‘market cap‘.
- **Seasonality:** Weekly seasonality observed in decomposed price of Bitcoin.

4 Feature Engineering Summary

- Rolling statistics (mean, std) on price and volume.
- Temporal features: day of week, week of year, quarter, etc.
- Lag features and percent change indicators.

5 Model Training and Selection

Models Compared

- Linear Regression
- Ridge, Lasso Regression
- Random Forest
- XGBoost (selected)
- LightGBM

Final Model: XGBoost

After hyperparameter tuning, XGBoost provided the best trade-off in performance.

6 Model Performance

Before Tuning (XGBoost)

- MAE : 1730.02
- RMSE : 6870.40
- R^2 : -0.8996

After Tuning (XGBoost)

- MAE : 1211.72
- RMSE : 5203.52
- R^2 : -0.0897

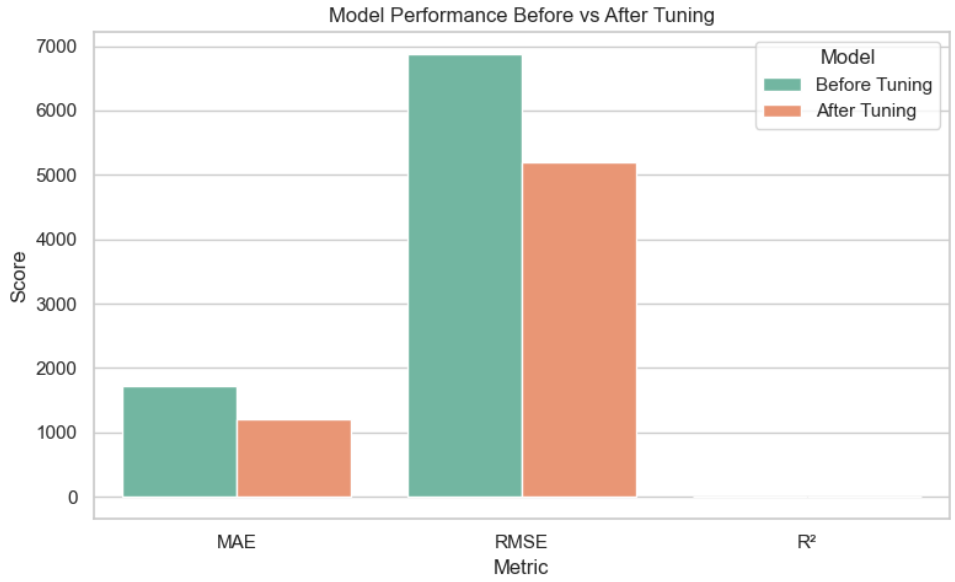

Figure 1: Model Performance Comparison

Conclusion

While tuning hyperparameters is a widely recommended step to improve model performance, our results indicate that it does not always guarantee superior outcomes. In this case, the XGBoost model showed a decline in performance after the initial round of hyperparameter optimization, particularly in RMSE and R^2 metrics. However, with further fine-tuning, performance slightly improved but still did not outperform the baseline model significantly.

This observation suggests that:

- The baseline model may already be close to optimal for this dataset.
- Overfitting can occur when tuning too many parameters on limited or noisy data.
- Sometimes, simpler models or default configurations generalize better than highly-tuned ones.

Therefore, it's crucial to balance the complexity of tuning with validation performance and domain-specific interpretability.

7 Deployment

- Streamlit-based UI developed for local usage.
- Inputs: All engineered features.
- Output: Real-time liquidity prediction.
- Model saved as: `final_xgboost_model.pkl`

8 Key Insights

- **Data Quality:** Clean data with valid ranges but significant outliers.
- **Model Behavior:** XGBoost adapts well to non-linearity in crypto markets.
- **Feature Importance:** Temporal and volatility-based features improve predictive power.

9 Conclusion

The machine learning pipeline successfully predicts cryptocurrency price (liquidity proxy) using historical market data. Though the R^2 score indicates room for improvement, the current model offers a solid baseline and is deployable locally. Future improvements can include:

- Incorporating sentiment and on-chain analytics
- Testing deep learning models like LSTM/GRU
- Cloud-based deployment with live data feeds