Assignment Report: Crypto Trader Performance Predictor Based on Market Sentiment

Objective

The goal of this project is to analyze the relationship between Bitcoin market sentiment and individual trader performance, and to build a predictive machine learning system integrated into a Streamlit dashboard. The system aims to identify whether a trade will be profitable based on input parameters such as trade size, leverage, market sentiment, and more.

Dataset Overview

a) Market Sentiment Dataset

- **Source**: Fear & Greed Index (CSV format)
- **Columns**: date, sentiment (values: Fear, Greed)

b) Trader Data (Hyperliquid Platform)

- **Source**: Historical trade logs
- **Columns** (after preprocessing):
  account, coin, execution_price, size_tokens, size_usd, side, timestamp_ist, start_position, direction, closed_pnl, transaction_hash, order_id, crossed, fee, trade_id, timestamp, leverage, date, sentiment, profitable

Data Preprocessing

- **Leverage Calculation**:
  Calculated as:

$$\text{leverage} = \frac{\text{execution\_price} \times \text{size\_usd}}{|\text{start\_position}|}$$

  (If start_position == 0, leverage is set to 0)
- **Target Variable**:
  profitable: A binary classification label (1 if closed_pnl > 0, else 0)
- **Encoding Categorical Variables**: One-hot encoding applied to features like side, coin, and direction.
- **Feature Selection**: Removed high cardinality and non-predictive fields (account, transaction_hash, order_id, etc.)
- **Missing Values**: Filled or removed based on relevance to prediction.

Exploratory Data Analysis (EDA)

- Plotted distribution of sentiments and profit outcomes.
- Analyzed correlation between features such as leverage, size, and profit.
- Observed that side, leverage, and sentiment had visible influence on trade outcomes.

1. PnL Distribution by Sentiment: This plot shows the distribution of closed PnL (Profit and Loss) for trades under different market sentiment conditions—namely *Fear* and *Greed*. It helps us understand whether traders are generally more profitable during one sentiment over the other. A right-skewed distribution for "Greed" might indicate more profits, whereas a left-skew for "Fear" could suggest losses are more common in fearful markets.

2. Estimated Leverage vs Closed PnL: This scatter plot displays the relationship between the estimated leverage used in trades and the resulting closed PnL. High leverage often implies high risk and high reward. By visualizing this, we can detect if traders using higher leverage tend to gain or lose more, and whether there's a visible pattern or threshold where risk becomes unprofitable.

3. Daily Average PnL Trend : This line plot tracks the average PnL per day over time. It's useful for identifying broader profitability trends in the dataset—whether trader outcomes are improving or declining with time. This can also reflect changing market conditions or evolving trader behaviour.

4. Sentiment Distribution Over Time: This time-series area or bar plot tracks how "Fear" and "Greed" sentiments fluctuate across different dates. It helps to understand the dominant market mood during specific periods and allows correlation with profit trends. For example, if a spike in “Greed” coincides with higher average PnLs, it might validate sentiment's impact on trader behaviour.

5. Top Coins by Trade Volume: This bar chart ranks the most actively traded coins based on cumulative USD volume. It gives insights into trader preferences and where most of the activity is concentrated. For instance, a dominance of BTC might indicate centralized trading, while diversity could imply a broader portfolio strategy among traders.

6. Win vs Loss Rate by Sentiment: This grouped bar plot shows the percentage of profitable and non-profitable trades under each sentiment condition. It directly compares success rates in “Fear” vs. “Greed” markets, helping validate whether positive sentiment actually leads to better trading outcomes. A higher win rate during “Greed” might support risk-on behaviour from traders during bullish conditions.

Machine Learning Model

For this assignment, a Random Forest Classifier was chosen as the core prediction model to determine whether a given trade would result in a profit or a loss, based on historical trader behaviour and prevailing market sentiment.

Random Forest is an ensemble learning technique that builds multiple decision trees and aggregates their outputs to make final predictions. Its robustness lies in its ability to reduce overfitting (common with individual decision trees) while maintaining high predictive accuracy.

In our implementation, the Random Forest model achieved an accuracy of approximately 92% on the full dataset. This high score can be attributed to several factors:

- Handling of Non-linearity: Random Forest is well-suited for complex datasets where relationships between variables may not be linear. In this case, the relationship between sentiment, leverage, and PnL outcomes is likely non-linear and interdependent.
- Feature Importance Awareness: The model internally evaluates which features contribute most to prediction. This aligns well with our dataset, where attributes like market sentiment, leverage, side (buy/sell), and asset type significantly influence trade outcomes.
- Low Variance and High Stability: By averaging across many trees, Random Forest reduces the effect of noise and outliers, producing more consistent results.
- Good Performance on Imbalanced Data: Even with uneven class distributions (e.g., more losses than wins), Random Forest can be tuned (via class weights or sampling) to provide stable accuracy.

While cross-validation accuracy was around 61%, the model performed exceptionally well on the entire dataset, indicating it learned important patterns in trader behavior and sentiment correlations. For deployment in the dashboard, it was selected for its strong empirical performance, ease of interpretability, and rapid training time.

Streamlit Dashboard Features

- **Overview Page**: View historical sentiment data and trader behavior visualizations.
- **Trade Predictor Page**: Input form to enter trade details like coin, side, size, execution price, direction, and sentiment.Displays prediction (Profitable or Not) using trained Random Forest model.
- **Visualization Section**: Displays leverage distribution, coin-wise sentiment stats, and profit trends.

Tools and Technologies

- **Languages**: Python
- **Libraries**:
  pandas, numpy, scikit-learn, joblib, streamlit, matplotlib, seaborn, Plotly

Project Outcomes

- Successfully demonstrated that market sentiment, leverage, and trade parameters can be used to predict trade profitability.
- Built an interactive and visual dashboard for real-time exploration and prediction.
- Deployed a practical machine learning model into a user-friendly interface.

Future Additions

- Support for multiple coins (ETH, SOL, etc.)
- Add a model training tab inside dashboard
- Real-time data integration via APIs
- A bot for instant predictions