Trader Behavior & Market Sentiment Analysis

■ Objective

This project analyzes the relationship between market sentiment (Fear/Greed Index) and trader performance using historical trading data from Hyperliquid. The aim is to uncover patterns, correlations, and actionable insights to improve trading strategies.

■ Datasets

1. Bitcoin Market Sentiment Dataset: Includes timestamp, value, classification, and date. Classifications include Extreme Fear, Fear, Neutral, Greed, and Extreme Greed.
2. Historical Trader Data (Hyperliquid): Includes trader account info, trade details, execution price, size, side, timestamp, leverage, PnL, and other trading metrics.

■■ Methodology

- Data Cleaning: Checked for missing values and duplicates, standardized column names, converted timestamps to datetime, aligned time zones, and extracted dates.
- Data Merging: Combined sentiment data with trading data on the date field.
- Exploratory Data Analysis: Assessed PnL trends, leverage patterns, win rates, and trading frequency across different sentiment regimes.
- Correlation Analysis: Encoded sentiment numerically and explored its relationship with PnL, leverage, and trade size.
- Metric Calculation: Calculated average PnL, win rate, leverage, trade size, and created trader rankings per sentiment.

■ Key Findings

- Contrarian Potential: 'Fear' and 'Extreme Fear' phases sometimes had higher win rates than 'Greed' periods.
- Risk Behavior: Leverage tends to spike during 'Greed' phases, amplifying both profit and loss potential.
- Trader Specialization: Some traders consistently outperform in specific sentiment regimes, indicating adaptive or contrarian strategies.
- Correlation Insights: Sentiment moderately correlates with leverage but has a weaker direct link to PnL.

■ Recommendations

- Explore contrarian strategies in Fear-dominated markets rather than following Greed-driven trends.
- Adjust leverage dynamically based on market sentiment to manage risk more effectively.
- Track and learn from top-performing traders in each sentiment regime.
- Combine sentiment analysis with technical and on-chain data for stronger predictions.

■ How to Run

1. Clone the repository or download the notebook.
2. Install dependencies: pip install pandas numpy matplotlib seaborn plotly scikit-learn
3. Open the Jupyter Notebook and run all cells to reproduce the analysis.