Quanta Mini Strategy Analysis Project

Volume Breakout Scanner

Streamlit URL: https://quant-mini-project.streamlit.app/
GitHub Repo: https://github.com/shreyan241/quant-mini-project
Time Taken to build and deploy: ~ 8 hours.

Project Setup:

The project was structured into four main components: data handling, analysis, visualization, and the web dashboard.

For data, we used YFinance to fetch daily stock prices and volume information. The data pipeline includes validation checks to handle missing data, duplicates, and ensure data quality. I also used a comprehensive CSV file containing all US-listed stocks with their company names and tickers to build a searchable dropdown in the UI.

The core analysis tracks when a stock's trading volume spikes above its 20-day average (calculated with a one-day shift to avoid look-ahead bias) by a user-defined threshold (e.g., 2x or 200% above average), while also requiring a minimum price increase (e.g., 2% daily change). When these conditions are met, the system identifies a breakout signal and calculates forward returns over a specified holding period (1-30 trading days).

The visualization centers around an interactive chart built using Plotly. The main chart displays price action through color coded candlesticks with a volume bar chart below showing daily trading volume. The 20-day volume moving average is overlaid on the volume bars for easy comparison. Breakout signals are clearly marked on both charts, with markers indicating entry points where volume and price conditions are met. The chart is fully interactive and users can zoom, pan, and hover over data points to see detailed information.

The dashboard, built with Streamlit, offers three main views:

1. Charts - Interactive price and volume visualization with breakout signals.
2. Report - Detailed breakdown of each trading signal with downloadable report.
3. Summary - Overall strategy performance metrics and statistics.

The interface allows users to select any US-listed stock, adjust strategy parameters (volume threshold, price change requirement, holding period), and analyze results across different time periods. All calculations and visualizations update in real-time as users modify these parameters.

Roadblocks/Challenges

- Data Issues:
  - YFinance API sometimes gave incomplete or missing data.
  - Built checks to handle missing values and duplicate dates.
  - YFinance integration handled:
    - Stock splits
    - Adjusted prices
    - Market holidays in data

- Metrics and Strategy parameters:
  - Made sure to use only past data in the 20-day volume average by shifting calculations back one day and avoid lookahead bias.
  - Used closing prices for all calculations to ensure consistency.
  - Added 1 to volume breakout threshold (e.g., 2.0 means volume must be 3x the average, as its 200% greater than average).

- Dashboard Performance:
  - Initially, switching tabs caused all calculations to run again unnecessarily.
  - Implemented efficient state management - Stored processed data in session state to avoid recalculations.
  - Cached stock data and analysis results.
  - Only updated necessary components when parameters changed.
  - Used containers to update only the necessary parts of the UI.
  - Separated data processing from UI rendering for better performance.
  - Added an "Apply" button for stock selection to prevent constant data reloading.

- Date Handling:
  - Implemented essential date validations:
    - End date must be after start date.
    - End date must have enough future days for the holding period.
    - Minimum 30 trading days required for analysis (for moving averages).
  - Used NYSE calendar to:
    - Count actual trading days between dates.
    - Validate sufficient trading days exist for the holding period.
    - Added buffer of 5 days to holding period when fetching data to ensure enough future data.

Suggestions

- Intraday Price Consideration:
    - Current Implementation: Uses end-of-day prices, which means we often miss the majority of the breakout movement
    - Observation: In high-growth tech stocks (e.g., NVDA, AAPL), significant price moves occur during the trading day
    - Improvement: Consider using mid-day prices uniformly throughout the system (e.g., prices at noon) to:
        - Capture more of the breakout movement
        - Enter positions earlier in the trend
        - Potentially improve overall returns

- Benchmark Comparison:
    - Add comparison of forward returns against market benchmarks (SPY or QQQ).
    - Calculate excess returns over the same holding period. This would provide:
        - True measure of strategy effectiveness
        - Risk-adjusted performance metrics
        - Better understanding of whether the signals generate alpha

- Market Context:
    - Add sector-relative volume analysis
        - Get sector ETFs for each major sector.
        - Calculate average volume for sector ETFs using same 20-day window.
        - Compare stock's volume spike to its sector's volume spike.
        - Signal is stronger if both stock and sector show high volume.
    - Consider market sentiment indicators
        - Calculate market breadth.
        - Could use options data for sentiment.
        - Filter signals based on overall market trend.