Technical Description: Risk-Managed PDT-Compliant Trading Algorithm

1. Overview

This algorithm is designed for retail traders with accounts under $25,000, complying with Pattern Day Trader (PDT) rules while implementing sophisticated risk management techniques. It utilizes the Alpaca Trading API for data retrieval and trade execution, operating on a minute-by-minute timeframe with updates every 5 minutes.

2. Key Components

2.1 Symbol Selection

- Dynamically screens all available tradable assets on Alpaca.
- Applies bullish criteria: price above 5-day and 10-day SMAs, above-average volume, price between $5 and $100.
- Selects up to 5 symbols (configurable) meeting these criteria.

2.2 Data Management

- Fetches 1-minute bar data for the past 30 minutes for selected symbols.
- Calculates key metrics including price, volume, VWAP, momentum, volatility, and moving averages.

2.3 Signal Generation

- Computes a trading signal score based on price relative to VWAP, momentum, SMA crossover, and volatility.
- Scores range from negative to positive, with higher positive scores indicating stronger buy signals.

2.4 Position Management

- Enters new positions when the signal score exceeds 20 and no current position exists.
- Exits positions when the signal score falls below -20.
- Implements risk parity by limiting each position to a maximum of 10% of the portfolio value.

2.5 Trade Execution

- Uses limit orders for entries to minimize slippage.
- Immediately places stop-loss and take-profit orders upon entering a position.
- Stop-loss set at 1% below entry price, take-profit at 2% above (configurable).

2.6 Risk Management

- Drawdown Limit: Ceases trading if daily drawdown exceeds 2% (configurable).

- PDT Compliance: Tracks day trades, ensuring no more than 3 in a rolling 5-day period.
- Position Sizing: Limits each position to 10% of the portfolio value.
- Stop-Loss and Take-Profit: Automatically applied to each trade.

2.7 Performance Monitoring

- Tracks and logs account equity, buying power, number of positions, and overall performance.
- Calculates and reports drawdown relative to the initial balance.

3. Algorithmic Flow

1. **Initialization**
    - Set up API connection and initialize parameters.
    - Determine initial account balance.
2. **Main Loop (Executes every 5 minutes)** a. Symbol Selection
    - Screen for bullish symbols.
    - Update the list of traded symbols.

b. Data Fetching

- Retrieve recent 1-minute bar data for selected symbols.

c. Metric Calculation

- Compute technical indicators and metrics for each symbol.

d. Signal Generation

- Calculate trading signals based on computed metrics.

e. Position Management

- Check drawdown limit.
- If drawdown is acceptable:
    - For each symbol, decide to enter, exit, or hold based on signals and current positions.
    - Ensure compliance with PDT rules and position size limits.

f. Trade Execution

- Submit buy/sell orders as determined by position management.
- For buy orders, simultaneously submit stop-loss and take-profit orders.

g. Performance Monitoring

    - Log current account status, positions, and performance metrics.

3. **Error Handling and Logging**
    - Comprehensive error catching and logging throughout the process.
    - Detailed logs for debugging and performance analysis.

4. Risk Management Details

4.1 Drawdown Control

- Tracks daily drawdown relative to the initial daily balance.
- If drawdown exceeds the threshold (default 2%):
    - Stops opening new positions.
    - Logs a warning message.
    - Continues monitoring existing positions.

4.2 PDT Rule Compliance

- Maintains a list of day trades with timestamps.
- Before each potential buy, checks if executing would exceed 3 day trades in 5 days.
- If limit would be exceeded, prevents the buy order.

4.3 Position Sizing

- Calculates maximum position size as 10% of current account equity.
- Limits buy orders to respect this maximum size.

4.4 Per-Trade Risk Management

- Stop-Loss: Sell order at 1% below entry price.
- Take-Profit: Sell order at 2% above entry price.
- Both orders submitted immediately upon position entry.

5. Technical Considerations

5.1 API Usage

- Utilizes Alpaca Trading API for all market data and order execution.
- Implements rate limiting and error handling to manage API constraints.

5.2 Data Processing

- Uses pandas for efficient data manipulation and calculation of technical indicators.
- Numpy employed for numerical computations.

5.3 Asynchronous Operation

- Leverages Python's asyncio for non-blocking operations, enabling efficient API calls and data processing.

5.4 Scalability

- Designed to handle multiple symbols concurrently.
- Modular structure allows for easy addition of new features or modification of existing components.

6. Limitations and Considerations

- Limited to long positions only; no short selling.
- Relies on technical indicators and does not incorporate fundamental analysis.
- Performance is heavily dependent on the effectiveness of the symbol selection criteria and signal generation logic.
- May be sensitive to market volatility and sudden price movements.
- Requires consistent internet connection and API availability.

7. Future Enhancements

- Implementation of machine learning for improved signal generation and risk management.
- Incorporation of sector rotation and broader market trend analysis.
- Addition of sentiment analysis from news and social media sources.
- Development of a user interface for real-time monitoring and parameter adjustment.
- Expansion to include options trading for additional strategies and hedging.