QUANTITATIVE TRADING STRATEGY RESEARCH

Non-IID Analysis, Dual Kalman Filter Strategy & Feature Engineering

Ngo Thanh An
Ethereum Algorithmic Trading

January 5, 2026

Contents

1 Part 1: The IID Problem in Financial Data

1.1 Violations of the IID Assumption

Financial markets are complex adaptive systems; therefore, time-series data (OHLCV) rarely satisfy the Independent and Identically Distributed (IID) assumption required by many standard statistical models. The primary causes include:

- **Independence Violation:**
  - **Volatility Clustering:** Price volatility is not random but tends to cluster. Large shocks are often followed by large changes, and small shocks by small changes (ARCH/GARCH effects).
  - **Serial Correlation:** The current value is influenced by past positions and market microstructure. For example, the price at $t+1$ is dependent on the momentum and order flow at t, unlike independent coin flips.
- **Identical Distribution Violation:**
  - **Non-Stationarity:** Statistical parameters such as the mean (μ) and variance (σ^2) change over time due to macroeconomic events.
  - **Structural Breaks:** The market undergoes Regime Shifts (e.g., from a Low Volatility regime to a Crisis regime). Old data may become obsolete or introduce noise (Concept Drift).

  In Machine Learning models, the IID assumption is foundational. Applying models directly to raw non-IID time-series data often leads to overfitting or poor generalization in live trading.

1.2 Mitigation Strategies

To apply Machine Learning effectively, I propose the following methods to address Non-IID issues while retaining the market's "memory":

1.2.1 Method 1: Fractional Differentiation

Instead of using Log-Returns (which erase memory) or Raw Prices (which are non-stationary), we use Fractional Differentiation to find a balance between stationarity and memory preservation. The mathematical formulation of the fractional difference operator is:

$$(1-L)^d X_t = \sum_{k=0}^{\infty} \frac{(-1)^k \prod_{i=0}^{k-1}(d-i)}{k!} L^k X_t \tag{1}$$

Where L is the lag operator and d is the order of differentiation (e.g., $d = 0.4$). This method renders the series stationary while maintaining the correlation with past data necessary for ML models to learn trends.

1.2.2 Method 2: Triple Barrier Method (Labeling)

To solve the noise issue associated with fixed-time horizon labeling, I utilize the Triple Barrier Method (inspired by Marcos Lopez de Prado). The label Y_i is not determined by time $t+\Delta t$, but by the first event to touch one of three barriers:

1. **Upper Barrier:** Take Profit threshold (based on volatility σ).
2. **Lower Barrier:** Stop Loss threshold.
3. **Vertical Barrier:** Holding period limit.

This method ensures that labels reflect true Dynamic Volatility rather than being arbitrarily defined by static timeframes.

2 Part 2: Trading Strategy Development Report

Note: Detailed source code, hypothesis testing steps, and full backtest results are available in the repository file `ETHUSDT_Strategy.ipynb`*. This section summarizes the research workflow and performance.*

2.1 Research Workflow & Data Processing

The strategy development process follows a quantitative research framework:

1. **Exploratory Data Analysis (EDA):** Before modeling, I analyzed Price Action and Volume on the Weekly timeframe to understand the long-term market structure.

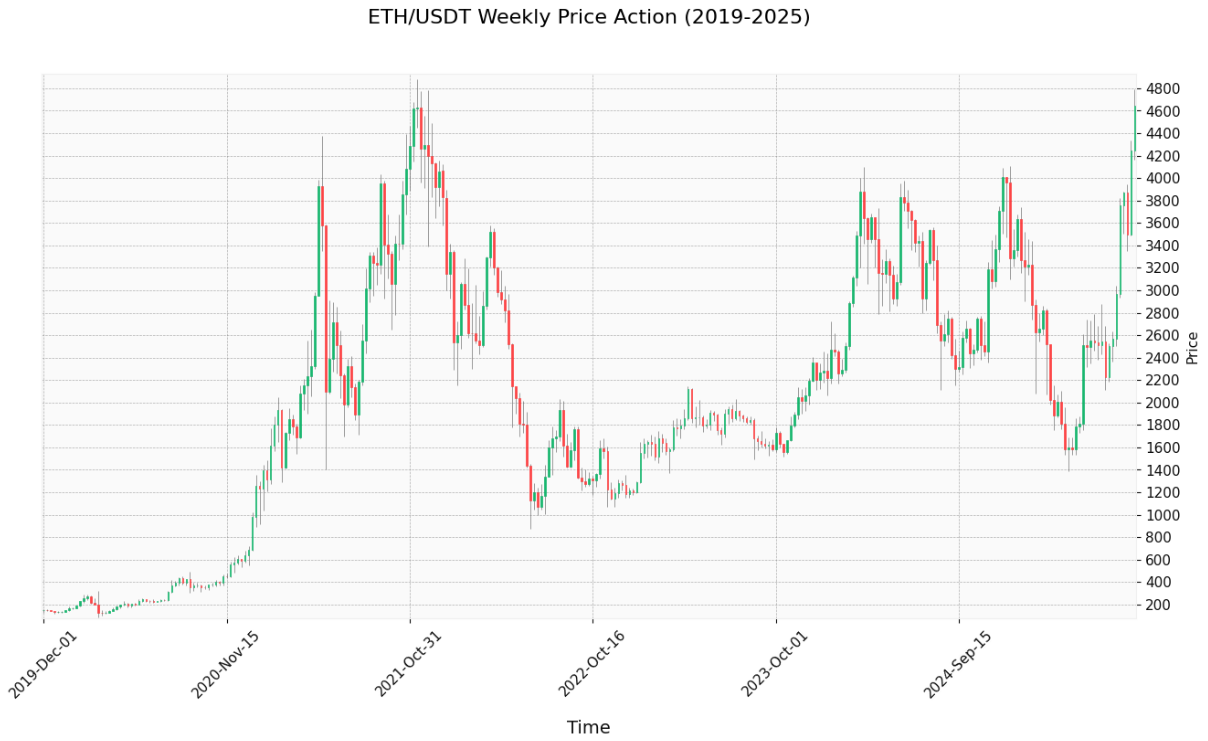

Figure 1: ETH/USDT Weekly Chart (2019-2025). The data covers full market cycles: Accumulation, Bull Run (2021), Bear Market (2022), and Recovery (2023-2024), ensuring representative backtesting conditions.

2. **Data Preprocessing:** ETHUSDT data (2019-2025) was cleaned and resampled to the **H4 (4-Hour)** timeframe. The H4 timeframe was selected to reduce micro-structure noise common in M30/H1 data while retaining sufficient granularity for Swing Trading.

3. **Hypothesis Testing (Volume Correlation):** Before selecting indicators, I tested the correlation between *Volume* and *Price Volatility*. The results showed an average correlation coefficient of **0.64** (> 0.3). This confirms that large price movements in ETH are supported by real liquidity flow, validating the application of a **Trend Following** strategy.

   [Image of Correlation Plot]

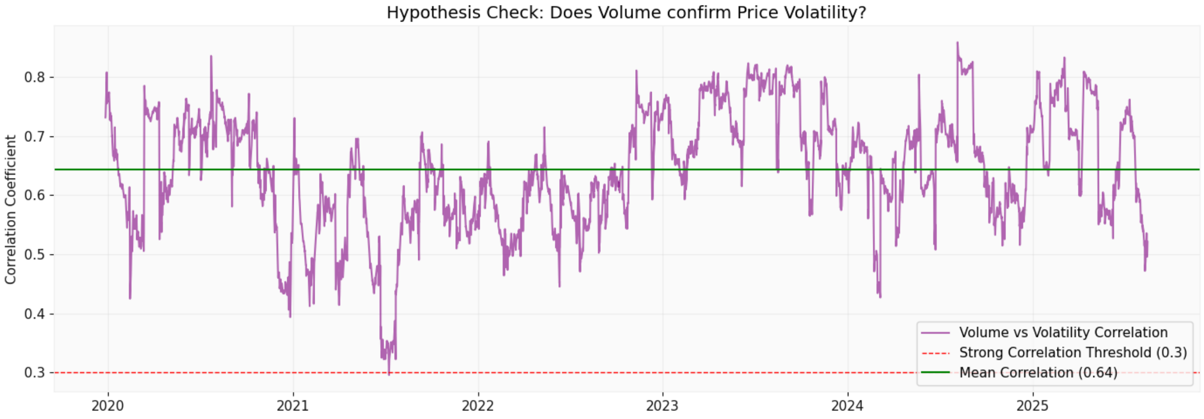

Figure 2: Rolling Correlation between Volume and Price Volatility (H4, 30-Day Window). The purple line fluctuates around 0.64, indicating a strong relationship between flow and price trends.

2.2 Model Selection: Why Kalman Filter?

This is the core decision of the strategy. Instead of traditional technical indicators like SMA or EMA, I selected the **Dual Kalman Filter**.

Addressing the Non-IID Problem:

- As analyzed in Part 1, financial data is *Non-Stationary* with frequent Regime Shifts. Moving Averages (MA) with fixed weights often lag significantly during structural shocks.
- The **Kalman Filter** is an *Adaptive State-Space* model. It treats the "true price" as a hidden state obscured by noise. The algorithm continuously updates its estimate based on the latest prediction error, allowing the model to adapt dynamically to market changes without assuming IID data.

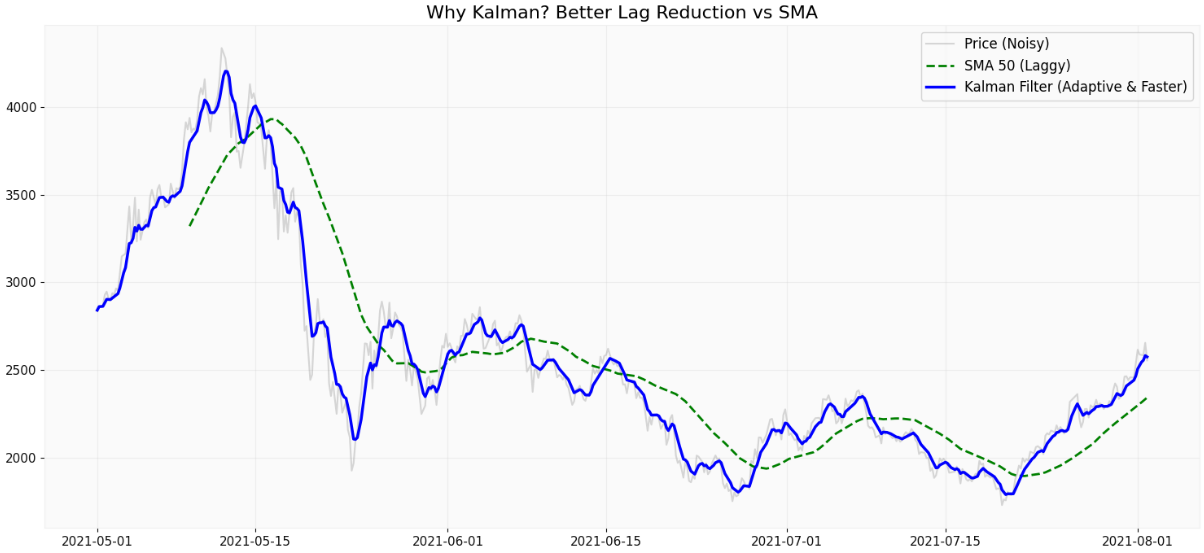

Figure 3: Lag Visualization: Comparison between Kalman Filter (Blue) and SMA 50 (Green) during high volatility (2021). The Kalman line reverses and tracks price much faster, helping preserve capital during crashes.

Empirical Evidence: Visual comparison (Figure 3) demonstrates that the Kalman Filter has significantly lower lag compared to an SMA of the same period, facilitating earlier Entry signals and faster Exits during reversals.

2.3 Strategy Logic & Leverage Mechanism (Base Strategy)

The Base Strategy (*Fixed Leverage*) is designed as follows:

- **Structure:** Utilizes two Kalman lines: Fast ($Q_{fast} = 0.003$) and Slow ($Q_{slow} = 0.0001$). These parameters were optimized via Grid Search.
- **Signal:** Long when $Kalman_{Fast} > Kalman_{Slow}$. Cash (Neutral) otherwise (Long-only strategy).
- **Capital Management (Volatility Targeting):** While the Kalman Filter effectively filters noise, the absolute return is often lower than Buy & Hold during strong Uptrends. To compensate, I applied a fixed leverage of **1.5x**. This level targets market outperformance without exposing the account to excessive liquidation risk (unlike 2x or 3x).

2.4 Base Strategy Performance (Fixed Leverage 1.5x)

The table below summarizes performance on the Training Set (In-Sample) and Testing Set (Out-of-Sample).

Table 1: Performance Summary: Train (2019-2022) vs Test (2023-Present)

Metric	In-Sample (Train)	Out-of-Sample (Test)
Ann. Return	174.92%	**56.58%**
Sharpe Ratio	1.54	**1.02**
Max Drawdown	-77.85%	-73.32%
Win Rate	31.80%	30.10%
Profit Factor	1.90	1.58
Avg Win	16.00%	10.72%
Avg Loss	-3.92%	-2.92%
Total Trades	216	195

Performance Analysis:

- **Return & Risk:** On the Test set (2023-Present), the strategy achieved an annualized return of **56.58%** with a Sharpe Ratio of **1.02**. Although there is some decay compared to the Train set, these metrics remain robust.
- **Trading Characteristics:** The Win Rate remained stable around **30-31%**, typical for Trend Following strategies. However, the *Avg Win / Avg Loss* ratio on the Test set reached **3.67** (10.72/2.92), indicating excellent ability to let profits run.
- **Warning:** The Max Drawdown on the Test set was **-73.32%**, reflecting high risk when using fixed leverage during adverse market conditions. This motivated the development of the Machine Learning component (Section 2.5).

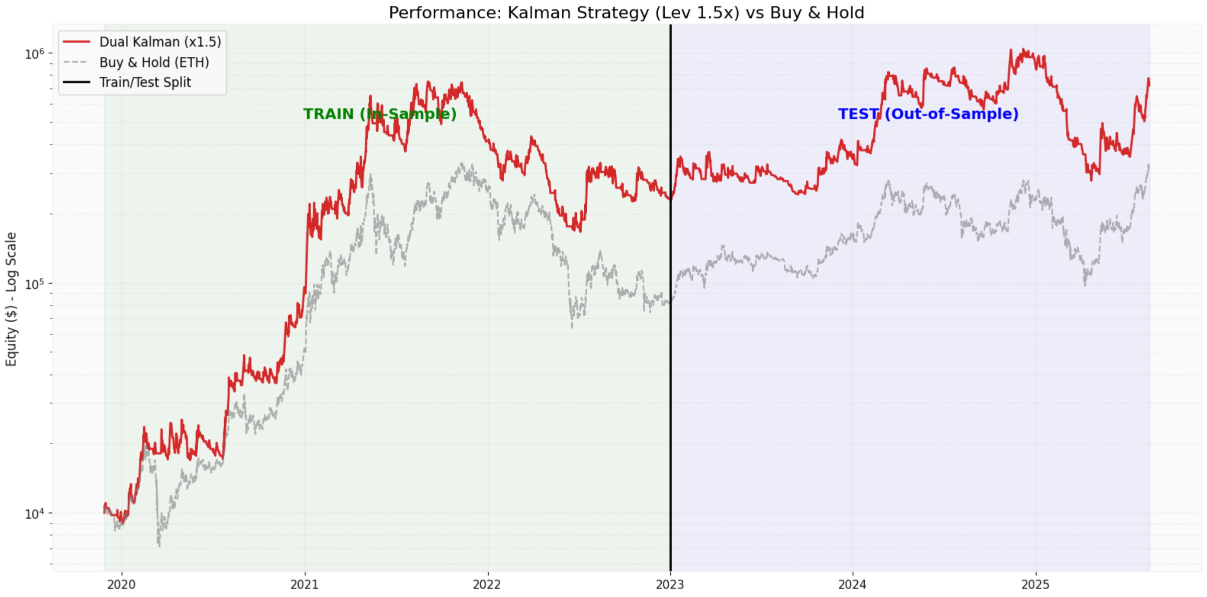

Figure 4: Equity Curve Comparison: Dual Kalman Strategy (1.5x Leverage) vs ETH Buy & Hold. The strategy (Red) maintains a significant advantage over passive holding (Gray) in the long run.

Current Issue: As observed in Figure 4, despite good returns and Sharpe Ratio, the **Max Drawdown (-73.32%)** is excessive, particularly during the 2022 Downtrend. This highlights the risk of fixed leverage in unclear trends, necessitating a smarter risk management mechanism.

2.5 Advanced Optimization: Machine Learning Dynamic Leverage

To address the Drawdown issue without sacrificing too much return, I implemented **Meta-Labeling** using Machine Learning.

- **Concept:** Use a **Random Forest** classifier to predict the "Probability of Profit" for each Kalman signal, replacing fixed leverage.
- **Features:** Spread (Kalman distance), Volatility, RSI, Volume Trend.
- **Dynamic Leverage Mechanism:**
  - Probability $>$ 55%: Increase Leverage to **2.0x** (Aggressive).
  - Probability $>$ 50%: Maintain Leverage **1.0x** (Neutral).
  - Low Probability: Reduce Leverage to **0.5x** (Defensive).

Optimization Results (Out-of-Sample):

Table 2: Detailed Comparison: Fixed Leverage vs ML Dynamic Leverage (Test Set 2023-Present)

Metric	**Fixed (1.5x)**	**ML Dynamic (0.5x-2x)**
Ann. Return	56.41%	33.98%
Sharpe Ratio	1.02	**1.03**
Max Drawdown	-73.32%	**-52.56%**
Win Rate	30.10%	29.59%
Profit Factor	1.58	1.54
Avg Win	10.71%	5.48%
Avg Loss	-2.92%	-1.49%
Total Trades	195	195

Impact Analysis:

- **Risk Management:** The ML model reduced the *Max Drawdown* from a critical **-73.32%** to a manageable **-52.56%** (an improvement of over 20%).
- **Capital Efficiency:** Although *Ann. Return* decreased from 56.41% to 33.98%, the **Sharpe Ratio** increased slightly to **1.03**. This proves that the profit reduction was due to deliberate de-leveraging to preserve capital, not strategy degradation.
- **Model Behavior:** Both *Avg Win* and *Avg Loss* for the ML model decreased by half compared to Fixed. This confirms the model's defensive nature, operating at a lower average leverage (approx. 0.7x - 0.8x) to prioritize safety.

Conclusion: Integrating Machine Learning acts as an automated "Risk Manager," significantly reducing Drawdown (by over 20

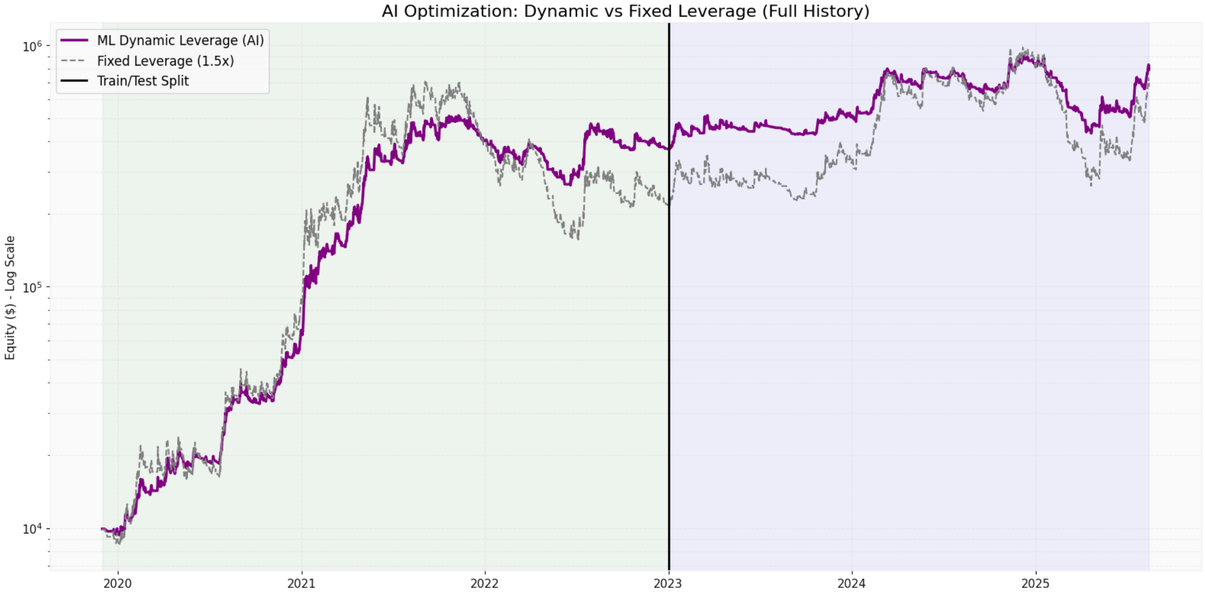

Figure 5: Equity Curve Comparison: ML Dynamic Leverage (Purple) vs Fixed Leverage (Gray). The purple line demonstrates smoother growth and significantly shallower drawdowns.

3 Part 3: Feature Engineering Proposal

In the Backtest section, I utilized 4 core features (Spread, Vol, RSI, VolTrend) for the Meta-Labeling model. Below is an expanded list of 8 Advanced Features designed to capture Flow, Momentum, and Volatility for future model iterations.

3.1 Momentum & Trend Features

1. **Feature 1: Kalman Filter Slope**
   - *Description:* Utilizes the Kalman Filter to smooth the price series and calculates the first derivative (velocity) of the trend.
   - *Significance:* Removes noise better than traditional Moving Averages, enabling early detection of true trend shifts.
2. **Feature 2: RSI Divergence Strength**
   - *Description:* Calculates the divergence between price and the RSI indicator over a Rolling Window.
   - *Significance:* Predicts potential reversals when momentum weakens despite price continuing to rise (or fall).

3.2 Volatility Features

3. **Feature 3: Parkinson Volatility Estimator**
   - *Description:* Volatility estimation formula based on High and Low prices: $\sigma_{Parkinson} = \sqrt{\frac{1}{4n\ln 2}\sum(\ln\frac{H_t}{L_t})^2}$
   - *Significance:* Utilizes intraday range information rather than just closing prices, providing a more accurate measure of market "temperature."
4. **Feature 4: Volatility Risk Premium (VRP) Proxy**
   - *Description:* The difference between short-term and long-term Historical Volatility.
   - *Significance:* Measures market fear. A spike in short-term volatility relative to long-term often signals a local bottom.

3.3 Volume Flow Features

5. **Feature 5: Volume-Weighted Average Price (VWAP) Deviation**
   - *Description:* Percentage distance between the current price and the daily VWAP.
   - *Significance:* Identifies if the price is "expensive" or "cheap" relative to the average price at which the majority of the market has traded. Useful for Mean Reversion.
6. **Feature 6: Force Index**
   - *Description:* $FI = Volume \times (Close_t - Close_{t-1})$.
   - *Significance:* Combines price and volume to determine the true strength of bulls or bears.

3.4 Microstructure / Statistical Features

7. **Feature 7: Autocorrelation of Returns (Lag 1)**

   - *Description:* Autocorrelation of the return series over a 14-period window.
   - *Significance:* Measures the short-term "Momentum" (positive) or "Mean Reversion" (negative) effect.

8. **Feature 8: Fractal Dimension (Hurst Exponent)**

   - *Description:* Estimates the Hurst exponent (H) of the time series.
   - *Significance:*
     - $H > 0.5$: Trending Market.
     - $H < 0.5$: Mean Reverting Market.
     - Helps the model weigh between Breakout and Swing trading strategies.

3.5 Feature Interaction

These features do not operate in isolation but complement each other:

- **Hurst Exponent + Kalman Filter:** The Hurst Exponent acts as a "Gatekeeper." If $H < 0.5$ (Mean Reversion), signals from the Kalman Filter (Trend) are down-weighted.

- **Volatility + VWAP Deviation:** When Volatility is low, the VWAP bands contract, making Breakout signals from VWAP more reliable (Squeeze setup).