Design and Performance Analysis of a Kalman-Filter-Based Strategy in Cryptocurrency Markets: Regime-Aware Pair Trading with HMM

Ryu Chan Hyeong

June 28, 2025

Abstract

This study proposes a dynamic pair trading strategy designed to address the high volatility and structural inefficiencies of cryptocurrency markets. We estimate time-varying hedge ratios via a Kalman Filter, infer hidden market regimes with a Hidden Markov Model (HMM), and robustly forecast the innovation variance using the DDIVF (Data-Driven Innovation Volatility Forecasting) technique. Regime-specific entry/exit thresholds are optimized using Bayesian Optimization with objective criteria such as Sharpe ratio, hit rate, average win, average loss, and average return. The proposed strategy is empirically tested on the ETH–SOL, BTC–SOL, and BTC–XRP pairs; notably, the Average Win and Ensemble variants maintain high profitability and stability even in the test period. This work demonstrates a path beyond static OLS-based pair trading, toward an adaptive, regime-aware trading system.

1 Introduction

Cryptocurrency markets exhibit structural characteristics that differ from traditional financial assets. While equities and bonds trade during fixed hours under regulatory oversight with relatively stable liquidity, cryptocurrencies trade 24/7 with high price volatility and a comparatively light regulatory environment. Even after the 2020s saw increased institutional participation, the market still features retail-driven speculative activity and pronounced structural inefficiencies. These features make the crypto space a favorable testbed for statistical arbitrage strategies.

Inefficiency manifests when prices fail to converge to fundamental values or when structural co-movements between assets regularly break down despite their presence. This creates profit opportunities for statistical arbitrage, among which pair trading is widely used for its simplicity and effectiveness. The strategy enters and exits positions when the price spread between two historically related assets widens beyond certain bounds and subsequently mean-reverts. Given observable time-series linkages among large-cap cryptocurrencies such as BTC, ETH, SOL, and XRP, pair trading is a suitable framework in this setting.

However, conventional pair trading often relies on OLS-based regressions to obtain a fixed hedge ratio, which then anchors spread calculations. Because crypto markets are highly nonstationary and asset relationships shift with changing participant composition, liquidity, and exogenous shocks, static coefficients estimated from the past cannot be assumed valid in the future. This mismatch can lead to erroneous signals, excessive turnover and costs, and ultimately performance deterioration.

To overcome these limitations, we propose a dynamic pair trading strategy combining a Kalman Filter (KF) and an HMM. The Kalman Filter denoises observations and performs real-time Bayesian estimation of the time-varying coefficient β_t. In noisy environments such as crypto, time-varying coefficients estimated by the Kalman Filter can be more reliable than fixed regression coefficients, allowing entries to react to structural changes and reducing false starts.

In tandem, we use an HMM to infer latent market regimes from observables such as returns or regression innovations. The HMM identifies whether the market is stable, volatile, or trending. This regime awareness enables regime-dependent entry and exit thresholds, improving precision and adaptability. For example, lower thresholds can be used in stable regimes to capture signals more sensitively, whereas higher thresholds in volatile regimes help suppress false signals.

Regime thresholds are not set heuristically but determined with Bayesian Optimization. Using a Gaussian Process surrogate and an acquisition function that balances exploration and exploitation, the optimizer searches the hyperparameter space efficiently and yields regime-wise thresholds that maximize performance metrics such as Sharpe and Calmar ratios.

In summary, we propose a dynamic, regime-aware pair trading strategy that combines Kalman Filter estimation with HMM-based regime inference and DDIVF-based innovation variance forecasting, replacing static spread rules with a system that adapts to market states. Empirical analysis on BTC, ETH, SOL, and XRP demonstrates feasibility and profit potential.

2 Literature Review

Pair trading is a canonical form of statistical arbitrage grounded in the cointegration theory of Engle and Granger. It seeks pairs with a long-run equilibrium relationship, enters positions when the spread widens sufficiently, and exits upon mean reversion to harvest profits.[1] In theory, the spread of cointegrated pairs is mean-reverting, which provides a clear mathematical basis that has been validated empirically over time.

Recent research extends this static approach dynamically. Johnson–Skinner et al. (2021) propose a trading framework that couples a Kalman Filter with an HMM.[2] By applying the HMM to Kalman innovations, they adapt strategy parameters across regimes. While their experiments center on traditional assets including the S&P 500 rather than crypto, their

[1] Engle, R. F., & Granger, C. W. J. (1987). Co-integration and error correction: Representation, estimation, and testing. *Econometrica*, 55(2), 251–276.

[2] E. Johnson-Skinner, Y. Liang, N. Yu and A. Morariu, “A Novel Algorithmic Trading Strategy using Hidden Markov Model for Kalman Filtering Innovations,” *2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*, Madrid, Spain, pp. 1766–1771.

insights are directly relevant here.

First, the Kalman Filter effectively reduces noise and estimates the time-varying regression coefficient β_t in real time, improving flexibility and accuracy relative to OLS in environments where relationships drift. This simultaneously enhances sensitivity and precision of signals.

Second, the HMM captures the notion that markets transition across distinct regimes, enabling regime-specific strategy design. Prior work (e.g., Singh and Srivastava) shows that regime-dependent thresholds help curb over-trading and reduce failure rates by adapting to volatility levels.

Third, the accuracy of signals derived from Kalman filtering can be further improved by forecasting the volatility of innovations more carefully. DDIVF (Data-Driven Innovation Volatility Forecasting) offers a robust alternative to the standard innovation variance, which may be overly reactive or sensitive to initial conditions. DDIVF incorporates directional and dispersion information of past innovations to produce a stable variance forecast, yielding more reliable z-scores and, ultimately, better trading performance and consistency.

Fourth, Johnson–Skinner et al. implement policy switching across regimes rather than a single static policy. For instance, lower entry thresholds in stable regimes and higher ones during turbulence improved Sharpe ratios and risk-adjusted returns.

Building on this framework, we extend the application to cryptocurrency markets—an environment of higher volatility, pronounced nonstationarity, thinner liquidity, and frequent trading interruptions. We also introduce Bayesian Optimization to determine regime-specific thresholds in a data-driven manner, simultaneously improving performance and reducing manual tuning.

3 Methodology

We design a dynamic pair trading strategy for cryptocurrency markets by combining a Kalman Filter (KF), DDIVF, an HMM, and Bayesian Optimization. This section details the role of each component and their integration.

3.1 Time-Varying Hedge Ratio via Kalman Filter

Let the hedge coefficient β_t evolve over time. A state-space model is specified as

$$\beta_t = \beta_{t-1} + v_t, \quad v_t \sim \mathcal{N}(0, \Sigma_v) \tag{1}$$

$$y_t = A_t\beta_t + \varepsilon_t, \quad \varepsilon_t \sim \mathcal{N}(0, \sigma_\varepsilon^2) \tag{2}$$

where y_t is the dependent variable (e.g., log(ETH)), A_t is the regressor vector (e.g., log(SOL)), β_t is the time-varying state, and ε_t is measurement noise. The Kalman Filter computes

$$\hat{\beta}_{t|t} = \hat{\beta}_{t-1|t-1} + (P_{t-1|t-1} + \Sigma_v)A_t^\top Q_t^{-1}(y_t - A_t\hat{\beta}_{t-1|t-1}), \tag{3}$$

where $Q_t = A_t(P_{t-1|t-1} + \Sigma_v)A_t^\top + \sigma_\varepsilon^2$ is the innovation variance and $\nu_t = y_t - A_t\hat{\beta}_{t-1|t-1}$ is the innovation.

3.2 DDIVF: Robust Forecasting of Innovation Variance

The conventional use of $\sqrt{Q_t}$ can be sensitive to initial values and distributional assumptions. To address this, Thavaneswaran et al. (2019) propose a DD-EWMA-style estimator (DDIVF).[3]

$$\hat{\sigma}_t = (1-\alpha)\hat{\sigma}_{t-1} + \alpha \frac{|\nu_{t-1} - \bar{\nu}|}{\hat{\rho}_\nu} \tag{4}$$

Here, $\hat{\rho}_\nu$ denotes the sign-correlation estimate and α is a smoothing parameter chosen to minimize one-step-ahead FESS. This better captures time-varying dispersion in innovations and stabilizes the z-score:

$$z_t = \frac{\nu_t}{\hat{\sigma}_t^{DD}}. \tag{5}$$

3.3 HMM: Regime Inference

Innovations ν_t may be generated by regime-specific distributions. We infer hidden states $S_t \in \{1, \ldots, K\}$ via an HMM with transition matrix

$$P = \{p_{ij}\}, \quad p_{ij} = P(S_{t+1} = j \mid S_t = i). \tag{6}$$

Using EM, we estimate P and emission parameters and compute filtered posteriors $\gamma_{i,t} = P(S_t = i \mid \nu_{1:t})$, assigning a regime to each t. We then apply regime-specific entry/exit thresholds $p_{\text{opt}}^{(j)}$—higher thresholds under volatile regimes to suppress noise, lower under stable regimes to capture signals.

3.4 Bayesian Optimization for Thresholds

Let x collect regime thresholds. We solve

$$\max_{x \in \mathcal{X}} f(x), \quad \mathcal{X} \subset \mathbb{R}^d, \quad d \leq 20, \tag{7}$$

where f is a performance functional (e.g., Sharpe). Treating f as a black box, we use a GP prior and an acquisition function (Expected Improvement) to choose the next query and obtain $p_{\text{opt}}^{(1)}, \ldots, p_{\text{opt}}^{(K)}$.

3.5 Trading Simulation and Constraints

We incorporate realistic frictions and constraints:

- $Q_{i,n}$: capital allocated to asset i at the start of period n
- $R_{i,n}$: period return of asset i
- r_n: risk-free rate (13-week T-bill)

[3] A. Thavaneswaran, A. Paseka, and J. Frank, "Generalized Value at Risk Forecasting," *Communications in Statistics - Theory and Methods*, pp. 1–8, 2019.

- $r \pm \delta r$: lending/borrowing rates for long/short
- ξ: transaction costs including impact, slippage, commission
- E_n: equity at the start of period n
- $\Lambda_{\max}$: maximum leverage ratio

PnL equation:

$$E_{n+1} - E_n = r_n \Delta t E_n + \sum_{i=1}^{N} Q_{i,n} R_{i,n} - r_n \Delta t \sum_{i=1}^{N} Q_{i,n} - \delta r \Delta t \sum_{i=1}^{N} |Q_{i,n}| - \xi \sum_{i=1}^{N} |Q_{i,n+1} - Q_{i,n}| \tag{8}$$

Leverage constraint:

$$\sum_{i=1}^{N} |Q_{i,n}| \leq \Lambda_{\max} E_n$$

Baseline assumptions:

- Initial equity: $E_0 = 100{,}000$
- Transaction cost: $\xi = 0.0005$
- Long/short funding spread: $\delta r = 0$
- Maximum leverage: $\Lambda_{\max} = 2$

3.6 Signal Generation Pipeline

Signals are generated as follows:

1. Estimate $\hat{\beta}_{t|t}$ and innovation ν_t via the Kalman Filter.
2. Forecast $\hat{\sigma}_t^{DD}$ via DDIVF.
3. Infer hidden regime S_t via the HMM.
4. Compute $z_t = \nu_t / \hat{\sigma}_t^{DD}$.
5. Trigger trades based on:
   - Buy: v_t crosses from $< -p_{\text{opt}}(S_t)\hat{\sigma}_t$ to $> -p_{\text{opt}}(S_t)\hat{\sigma}_t$
   - Sell: v_t crosses from $> +p_{\text{opt}}(S_t)\hat{\sigma}_t$ to $< +p_{\text{opt}}(S_t)\hat{\sigma}_t$

This integrated KF–DDIVF–HMM–BO architecture enables robust statistical arbitrage in highly volatile, nonstationary crypto markets.

4 Empirical Results

4.1 Data

We use one-minute price data for BTC, ETH, SOL, and XRP from June 25, 2024 to June 25, 2025 (525,525 observations). The training period spans June 25, 2024–March 1, 2025; the test period spans March 2, 2025–June 25, 2025. Prices are sourced from Deribit. The risk-free rate is the U.S. 13-week T-bill obtained via the `yfinance` API.

4.2 Cross-Asset Correlations

To assess linear relations, we compute the correlation matrix on training data using log returns (Figure 1).

We find high correlations for BTC–XRP (0.92), BTC–SOL (0.87), and SOL–XRP (0.70). ETH shows comparatively lower correlations with others, suggesting more care in pair selection involving ETH.

We test for cointegration using the Engle–Granger two-step procedure and retain pairs with p-values below 0.4 as trading candidates (Table 1).

Accordingly, we select **ETH–SOL** as the representative pair for model design and strategy evaluation, and we also apply the same framework to BTC–SOL and BTC–XRP for robustness checks.

4.3 Choosing the Number of HMM States

We apply an HMM to DDIVF-based innovations and determine the number of states using AIC and BIC (Figure 2).

Both criteria reach a minimum at three states; to avoid overfitting, we set the HMM to **three regimes**.

4.4 DDIVF-Based Signals

For ETH–SOL, we estimate the hedge ratio $\hat{\beta}_t$ via the Kalman Filter, compute innovations ν_t, forecast $\hat{\sigma}_t^{DD}$, and generate signals from $z_t = \nu_t / \hat{\sigma}_t^{DD}$ (Figure 3).

Relative to simple moving averages, DDIVF more responsively captures abrupt changes in innovations and improves entry/exit handling around spikes.

Using the optimized three-state HMM, we classify innovations by regime (Figure 4).

State 0 clusters tightly around the mean (stable), State 1 shows modest dispersion, and State 2 captures extreme movements (high risk). We set regime-wise thresholds via Bayesian Optimization to build a regime-aware dynamic entry/exit policy.

4.5 ETH–SOL Performance

Table 2 reports regime thresholds $p(S_0)$, $p(S_1)$, $p(S_2)$ optimized under different criteria (Sharpe, Hit Rate, etc.) for the training set.

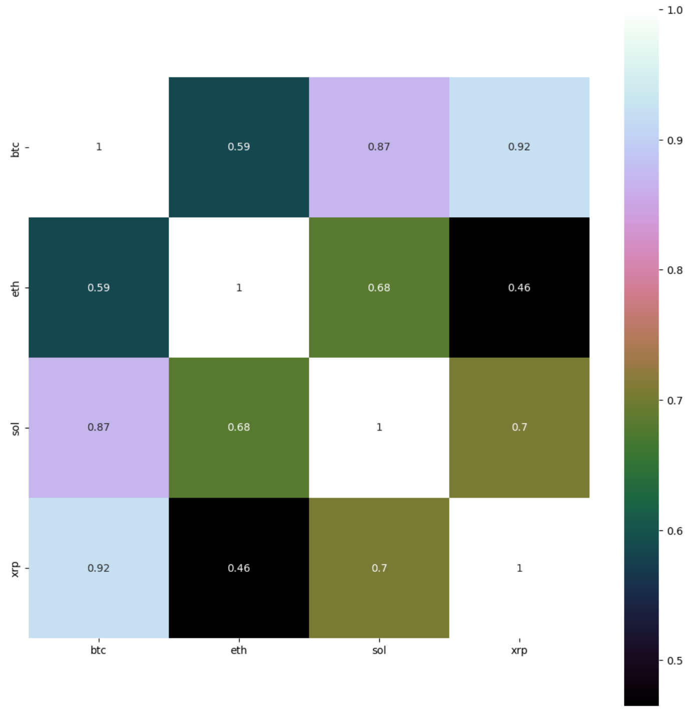

Figure 1: Correlation Matrix (Train Set)

Table 1: Engle–Granger Cointegration Test (p-value)

Pair	p-**value**
(btc, sol)	0.365
(btc, xrp)	0.350
(eth, sol)	0.075

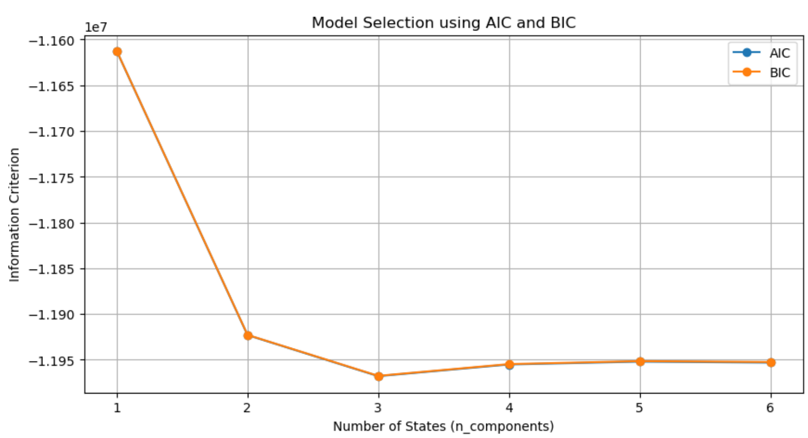

Figure 2: HMM State-Count Selection via AIC and BIC

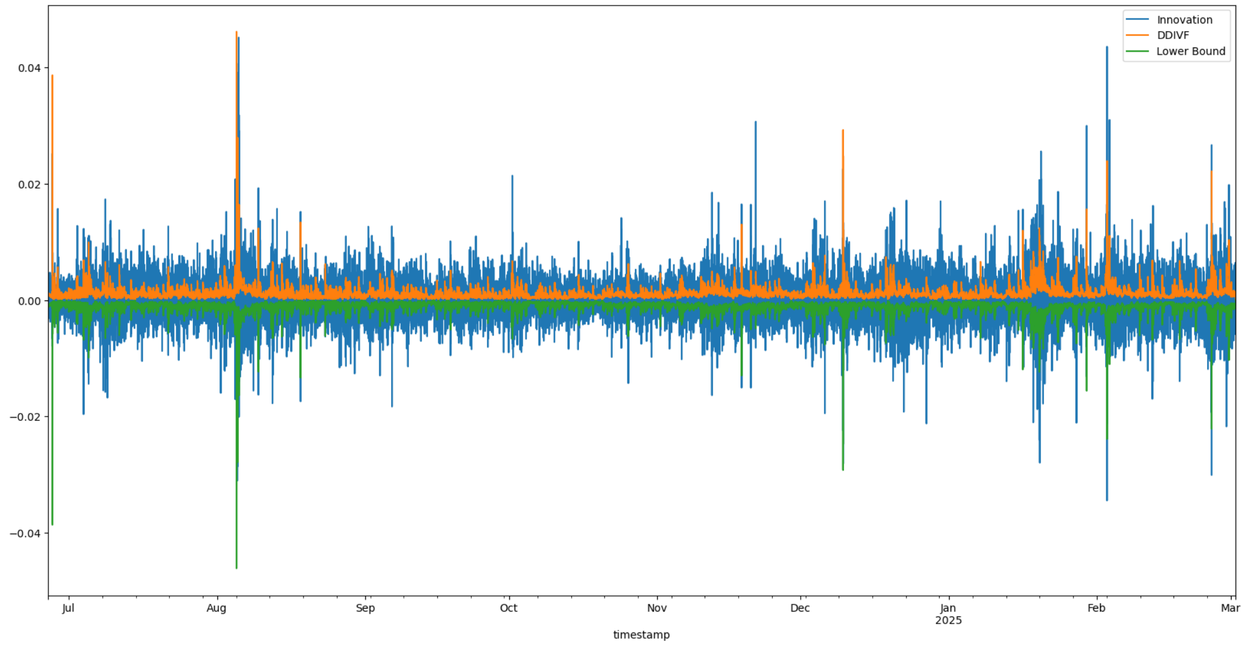

Figure 3: Innovation and DDIVF-Based Signal (Lower Bound)

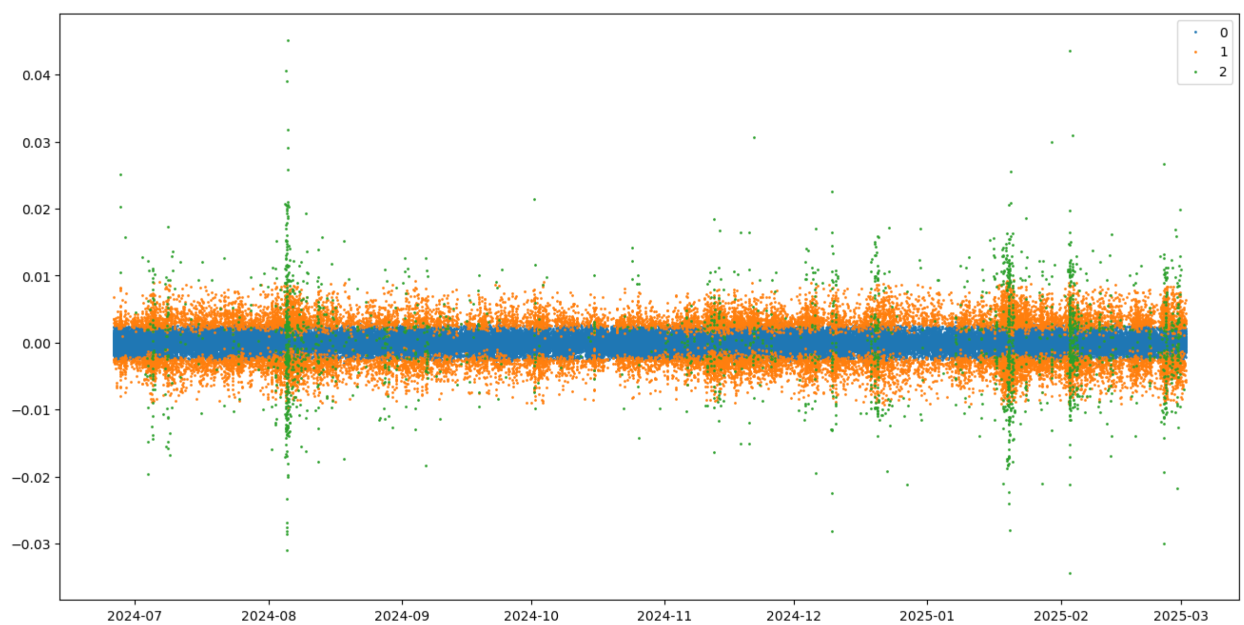

Figure 4: HMM Regime-Specific Innovations (0: Stable, 1: Moderate, 2: High Volatility)

On the training set, the *Average Win* criterion achieves the best Sharpe (11.20%) and Calmar (1.42%), while *Hit Rate* and *Average Loss* variants are conservative, with fewer trades and strong loss avoidance.

Applying the same thresholds to the test set yields Table 4. On average, the *Average Win* and *Ensemble* strategies remain strong out-of-sample, whereas *Sharpe* and *Average Return* degrade.

4.6 BTC–SOL Performance

We repeat the exercise for BTC–SOL (Tables 5, 6, Figure 7).

4.7 BTC–XRP Performance

We also test BTC–XRP (Tables 7, 8, Figure 8).

5 Conclusion

We integrate a Kalman Filter, HMM, DDIVF, and Bayesian Optimization to design a pair trading strategy tailored to high-volatility crypto markets, and validate it on ETH–SOL, BTC–SOL, and BTC–XRP.

Core to the design is **regime awareness**: the Kalman Filter yields time-varying β_t, DDIVF robustly forecasts innovation variance, and the HMM clusters the market into three regimes. We then set regime-specific thresholds and optimize them via Bayesian Optimization.

Key takeaways:

- In the **test set**, certain variants (Average Win, Ensemble) remained profitable and stable, whereas Sharpe- and Average-Return-optimized variants degraded due to overly sensitive thresholds.
- Sharpe/Average Return criteria tended to choose **low thresholds**, leading to **many trades**, vulnerability to false signals, higher costs, and worse out-of-sample performance.
- The **Average Win** variant used **higher thresholds**, reduced noise entries, and achieved high Sharpe with low MDD via fewer, higher-quality trades.
- The **Ensemble** variant averaged thresholds across five criteria, dampening extremes and delivering consistently solid results.

Overall, the integrated approach overcomes limitations of static OLS-based pairs by combining time-varying estimation, regime recognition, robust variance forecasting, and data-driven threshold optimization. The importance of regime-specific thresholds and the trade-off between trade count and threshold stringency are confirmed empirically, suggesting that filtering for higher-quality signals yields better stability and returns in practice. Future work may explore deep-learning-based regime models, multi-pair joint optimization, and extensions to options/derivatives.

Table 2: ETH–SOL: Regime-Wise Optimal Thresholds and Metrics (Train Set)

Criterion	$p(S_0)$	$p(S_1)$	$p(S_2)$	Key Metric
Sharpe Optimized	27.20	23.30	30.79	Sharpe = 7.68%
Hit-Rate Optimized	39.77	44.74	33.06	Hit Rate = 100%
Average Win Optimized	28.28	45.00	31.72	Avg. Win = 22,072
Average Loss Optimized	43.83	45.00	45.00	Avg. Loss = 0
Average Return Optimized	27.20	23.30	30.79	Return = 0.33%

Table 3: ETH–SOL: Strategy Performance Comparison on Train Set (with Benchmarks)

Strategy	#Trades	Hit Rate	Avg. Return	Sharpe	MDD	Calmar
Sharpe Optimized	52	53.85%	0.33%	7.68%	-36.60%	0.88%
Hit Rate Optimized	5	100%	0.33%	7.17%	-54.23%	0.59%
Average Win Optimized	9	66.67%	0.45%	**11.20%**	-30.70%	**1.42%**
Average Loss Optimized	3	100%	0.23%	4.65%	-46.77%	0.46%
Average Return Optimized	52	53.85%	0.33%	7.68%	-36.60%	0.88%
BH(ETH)	–	–	-0.09%	-1.09%	-79.44%	-0.13%
BH(SOL)	–	–	0.31%	3.95%	-66.11%	0.45%

[1] All statistics are computed from **daily returns** (Daily PnL).

Figure 5: ETH–SOL: Cumulative Return Simulation (Train Set)

Table 4: ETH–SOL: Strategy Performance Comparison on Test Set (with Benchmarks)

Strategy	#Trades	Hit Rate	Avg. Return	Sharpe	MDD	Calmar
Sharpe Optimized	95	45.26%	-0.13%	-2.70%	-47.55%	-0.30%
Hit Rate Optimized	28	46.43%	0.32%	7.33%	-29.11%	1.05%
Average Win Optimized	31	51.61%	**0.59%**	**14.86%**	**-20.80%**	**2.77%**
Average Loss Optimized	22	59.09%	0.53%	10.37%	-36.82%	1.40%
Average Return Optimized	95	45.26%	-0.13%	-2.70%	-47.55%	-0.30%
Ensemble Optimized	43	53.49%	0.45%	10.25%	-28.72%	1.51%
BH(ETH)	–	–	0.46%	8.52%	-35.48%	1.26%
BH(SOL)	–	–	0.58%	5.15%	-69.28%	0.81%

Figure 6: Cumulative Return Simulation (Test Set)

Table 5: BTC–SOL: Regime-Wise Optimal Thresholds and Metrics

Criterion	$p(S_0)$	$p(S_1)$	$p(S_2)$	Key Metric
Sharpe Optimized	26.80	21.95	30.84	Sharpe = 17.04%
Hit Rate Optimized	27.45	29.22	45.00	Hit Rate = 63.33%
Average Win Optimized	25.41	44.58	42.28	Avg. Win = 15,579
Average Loss Optimized	5.00	5.00	5.00	Avg. Loss = 272.72
Average Return Optimized	26.80	21.95	30.84	Return = 0.54%

Table 6: BTC–SOL: Strategy Performance Comparison (with Benchmarks)

Strategy	#Trades	Hit Rate	Avg. Return	Sharpe	MDD	Calmar
Sharpe Optimized	90	47.78%	0.07%	1.27%	-27.23%	0.21%
Hit Rate Optimized	52	51.92%	0.39%	10.27%	-23.08%	1.64%
Average Win Optimized	21	42.86%	**0.48%**	**13.99%**	**-20.04%**	**2.32%**
Average Loss Optimized	2614	48.32%	-0.53%	-7.95%	-73.92%	-0.73%
Average Return Optimized	90	47.78%	0.07%	1.27%	-27.23%	0.21%
Ensemble Optimized	79	48.10%	0.16%	3.57%	-26.76%	0.55%
BH(BTC)	–	–	0.46%	8.62%	-35.35%	1.28%
BH(SOL)	–	–	0.59%	5.26%	-68.83%	0.84%

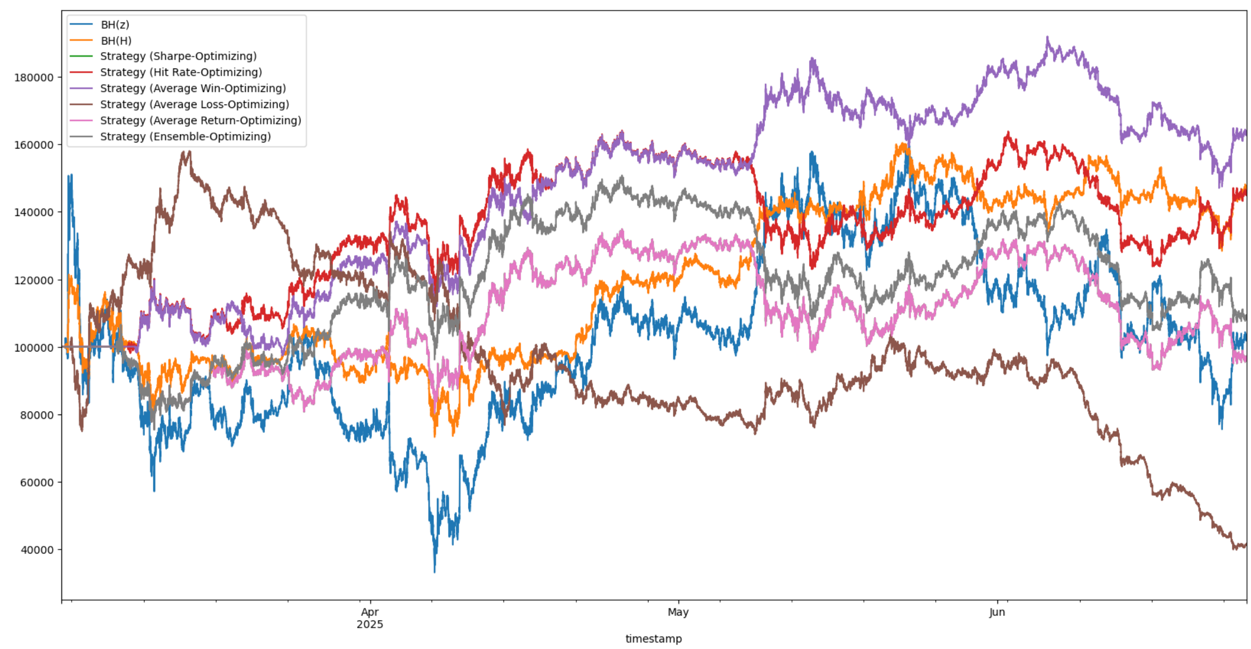

Figure 7: BTC–SOL: Cumulative Return Comparison

Table 7: BTC–XRP: Regime-Wise Optimal Thresholds and Metrics

Criterion	$p(S_0)$	$p(S_1)$	$p(S_2)$	Key Metric
Sharpe Optimized	26.80	21.95	30.84	Sharpe = 16.73%
Hit Rate Optimized	22.50	40.67	43.55	Hit Rate = 56.89%
Average Win Optimized	26.80	21.95	30.84	Avg. Win = 14,586
Average Loss Optimized	32.15	23.69	5.00	Avg. Loss = 1,503
Average Return Optimized	9.12	43.68	43.31	Return = 10.55%

Table 8: BTC–XRP: Strategy Performance Comparison (with Benchmarks)

Strategy	#Trades	Hit Rate	Avg. Return	Sharpe	MDD	Calmar
Sharpe Optimized	23	60.87%	0.29%	9.91%	-12.40%	2.27%
Hit Rate Optimized	16	**68.75%**	0.35%	**11.91%**	-12.87%	2.65%
Average Win Optimized	23	60.87%	0.29%	9.91%	-12.40%	2.27%
Average Loss Optimized	57	59.65%	0.22%	5.39%	-30.42%	0.69%
Average Return Optimized	165	52.73%	-0.11%	-3.16%	-32.42%	-0.38%
Ensemble Optimized	15	73.33%	**0.34%**	11.51%	**-12.87%**	**2.55%**
BH(BTC)	–	–	0.46%	8.62%	-35.35%	1.28%
BH(XRP)	–	–	0.38%	3.64%	-63.60%	0.58%

Figure 8: BTC–XRP: Cumulative Return Comparison

References

[1] Engle, R. F., and Granger, C. W. J. (1987). "Co-integration and error correction: Representation, estimation, and testing." *Econometrica*, 55(2), 251–276.

[2] Johnson-Skinner, E., Liang, Y., Yu, N., and Morariu, A. (2021). "A Novel Algorithmic Trading Strategy using Hidden Markov Model for Kalman Filtering Innovations." *2021 IEEE 45th Annual Computers, Software, and Applications Conference (COMPSAC)*, Madrid, Spain, pp. 1766–1771.

[3] A. Thavaneswaran, A. Paseka, and J. Frank, "Generalized Value at Risk Forecasting," *Communications in Statistics - Theory and Methods*, pp. 1–8, 2019.

[4] G. Xuan, W. Zhang, and P. Chai, "EM algorithms of Gaussian mixture model and hidden Markov model," *Proceedings 2001 International Conference on Image Processing (Cat. No.01CH37205)*, vol. 1, pp. 145–148, 2001.

[5] Bergstra, J., and Bengio, Y. (2011). "Random Search for Hyper-Parameter Optimization." *Journal of Machine Learning Research*, 13(2), 281–305.

[6] Deribit API Documentation. Available at: `https://docs.deribit.com`

[7] YFinance API Documentation. Available at: `https://pypi.org/project/yfinance/`