© Volume 2, Issue 10, October 2026 | JATIR
e-ISSN: 2839-5786, p-ISSN: 3139-8464

Machine Learning-Assisted Validation of an Adaptive True Renko Trading Strategy an Empirical Study on crypto

[1]Venkata Maheswara Rao Atchula, [2]Dr. Pawan Kumar Pareek
[1,2]Nirwan University, Jaipur, Rajasthan

***Abstract*—This paper presents an empirical evaluation of a proprietary adaptive True Renko trading strategy in which machine learning is used directly as a signal-validation component. The True Renko engine first constructs directional bricks using an ATR-based brick mechanism and applies a moving-average filter to generate candidate trade signals. The candidate signal is then represented through Renko, trend, momentum, volatility, market-regime and risk/reward features and evaluated by an XGBoost-based supervised validator. The ML model does not generate an independent trade signal and is not a Meta ML layer; it only determines whether a True Renko candidate is accepted for execution. The study uses a seven-year BTCUSDT historical dataset, with four years used for development/training and three years reserved as unseen evaluation data. The empirical design evaluates ATR-based True Renko configurations with moving-average filtering and six brick sizes from 100 to 600. The unseen-test stage is subsequently evaluated with ML confidence thresholds of 0.50, 0.53, 0.55, 0.57, 0.60, 0.65 and 0.70. The supplied registry contains 52 experiment runs: six development runs, six no-ML unseen-test runs and forty ML-assisted unseen-test runs. Across the ML threshold experiments, increasing the confidence threshold consistently reduces the number of accepted trades, while profitability and win percentage do not change monotonically. The results demonstrate that ML validation can materially change trade selection and exposure while preserving the deterministic True Renko signal-generation architecture. The findings support further forward testing of the combined True Renko and direct ML-validation framework, while also highlighting the need for transaction-cost modelling, statistical testing and independent forward validation before live deployment.**

***Index Terms*—True Renko, ATR Renko, Machine Learning, XGBoost, Signal Validation, Algorithmic Trading, BTCUSDT, Feature Engineering, Out-of-Sample Testing.**

I. INTRODUCTION

Algorithmic trading systems commonly combine deterministic technical rules with statistical or machine-learning components. A deterministic strategy can provide transparency and

reproducibility, but its signals may still contain low-quality opportunities under changing volatility and market regimes. A practical architecture is therefore to retain the original signal generator and introduce machine learning as a second-stage validator.

Renko representations are particularly relevant because they emphasize directional price movement rather than the regular passage of time. The True Renko strategy studied here extends this concept through adaptive ATR-based brick construction. A moving-average filter is applied before a candidate signal is passed to the ML validator. This creates a layered decision process in which the deterministic strategy remains responsible for identifying the trading opportunity.

The central methodological distinction of this study is that the ML component is not a separate trading strategy and is not a Meta ML layer. It is a model-specific validation mechanism embedded in the True Renko strategy. A candidate BUY or SELL signal is generated by True Renko and its filters; the ML component returns a validation score; only a candidate satisfying the configured threshold is permitted to proceed to trade management.

The study evaluates three sequential stages: (1) True Renko with moving-average filtering on development data, (2) the same True Renko architecture on unseen test data without ML, and (3) the same unseen-test architecture with direct ML validation at multiple confidence thresholds. This design makes it possible to examine the incremental role of the ML validator without replacing the underlying True Renko decision mechanism.

1.1 Research Questions

- RQ1. What performance characteristics does the adaptive True Renko strategy exhibit when evaluated with the moving-average filter and without ML validation?
- RQ2. How does the True Renko strategy behave on the unseen evaluation period before adding ML validation?
- RQ3. How does direct ML validation change the number of accepted trades, win percentage, profitability and drawdown?
- RQ4. How sensitive are the observed outcomes to ML confidence thresholds from 0.50 to 0.70?
- RQ5. Does ML filtering preserve positive realized performance across the tested True Renko brick-size configurations?

1.2 Contributions

- The paper evaluates a proprietary adaptive True Renko architecture as an independent strategy rather than comparing it with another Renko implementation.
- Machine learning is explicitly positioned as a direct signal-validation component inside the True Renko decision pipeline.
- The study separates development/training, unseen no-ML testing and unseen ML-assisted testing.
- The paper reports threshold sensitivity across seven ML confidence levels and multiple ATR-True-Renko brick sizes.

- The complete supplied 52-run empirical registry is reproduced in the appendix for traceability.

II. TRUE RENKO STRATEGY ARCHITECTURE

The evaluated strategy is an adaptive True Renko system. The implementation supports fixed and ATR-based Renko construction; the experiments reported in this paper use the ATR configuration. The ATR engine uses a period of 14 and an ATR multiplier of 2.0 for the principal experimental configuration. Brick sizes are evaluated at 100, 200, 300, 400, 500 and 600.

2.1 Signal Generation

The strategy maintains a directional Renko history and identifies current upward or downward brick conditions. A moving-average engine provides a short/long trend filter. In the supplied implementation, the MA engine is configured with a short length and long length, and a candidate signal is permitted only when the MA filter allows the corresponding direction. Additional range, cooldown and re-entry controls are part of the strategy state.

2.2 Adaptive ATR Renko Construction

For ATR-based construction, the implementation initializes an ATR engine with an ATR period of 14 and an ATR Renko builder using an ATR multiplier of 2.0. The ATR mechanism allows the Renko representation to adapt to prevailing volatility while the experimental brick-size configurations provide a controlled sensitivity dimension.

2.3 Direct ML Validation

The ML validator is invoked only after a True Renko candidate signal has been generated and the MA filter has passed. The feature vector combines strategy identity, trade direction and context with Renko-derived information and market features. The implementation records the ML score, model, prediction, validation reason and validation latency.

Stage	Function
1. Price / Market Data	Receives the historical market sequence.
2. ATR True Renko	Builds directional adaptive Renko structure.
3. MA Filter	Confirms directional eligibility.
4. Candidate Signal	Creates BUY or SELL candidate.
5. Feature Engineering	Builds Renko and contextual feature vector.
6. XGBoost Validation	Produces a validation probability/score.
7. Threshold Decision	Accepts or rejects the candidate.

8. Trade Management	Manages accepted positions and exits.

2.4 Feature Representation

Chart-data note: the supplied experiment registry contains trade-level summary metrics but not the complete timestamped Renko-brick series required to reconstruct an empirical BTCUSDT True Renko chart. Therefore, Figure 1 is explicitly labelled illustrative rather than being presented as a measured market-period chart. The quantitative figures use only the supplied experiment registry.

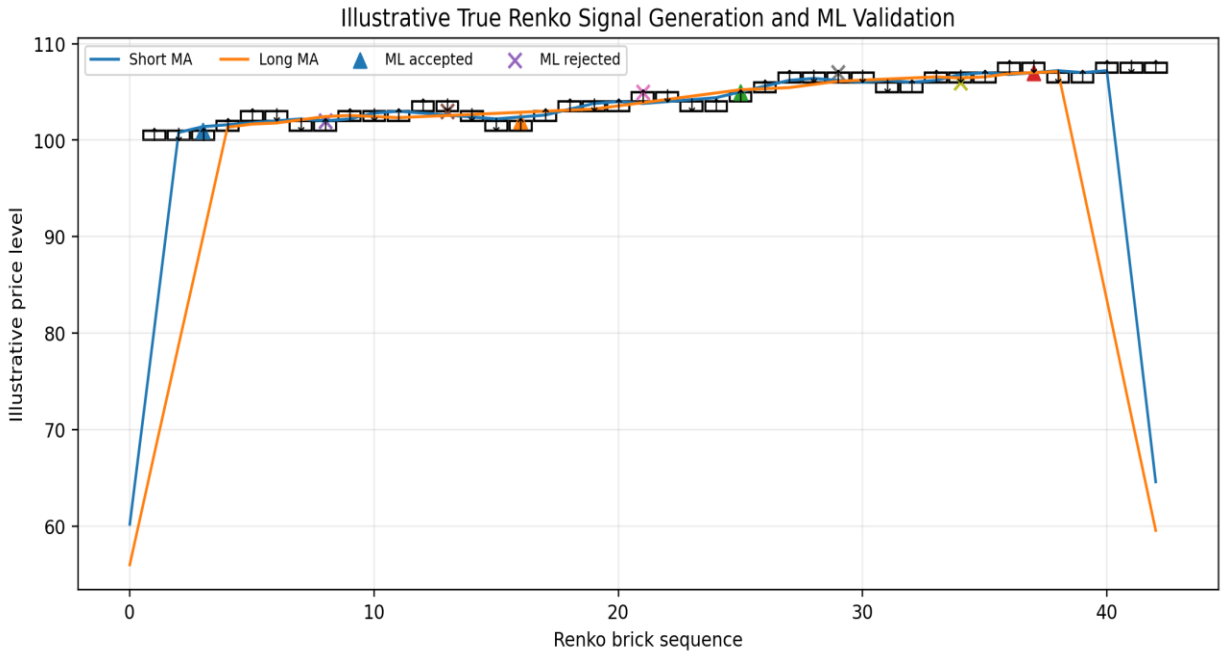

Figure 1. Illustrative True Renko brick representation with MA filtering and ML acceptance/rejection. The figure is conceptual and is not used to calculate the reported numerical results.

Explicit True Renko features include Renko direction, Renko brick size and consecutive-brick information. The broader feature builder also exposes moving-average information such as short and long MA values, slope, bias and crossover state, together with momentum, volatility, market-regime, signal-quality and risk/reward features. The resulting representation allows the validator to evaluate a candidate in both structural and contextual terms.

III. EXPERIMENTAL METHODOLOGY

The experimental methodology follows a development-to-unseen-test design. Seven years of historical BTCUSDT data are partitioned into four years of development/training and three years of unseen evaluation. The same True Renko strategy architecture is retained throughout the experiment. ML is introduced only as a validation stage in the later test runs.

3.1 Experimental Stages

Stage	Configuration	Purpose
Stage A	True Renko + MA; development/training; no ML	Establish development behaviour and baseline configuration characteristics.
Stage B	True Renko + MA; unseen test; no ML	Evaluate the deterministic strategy on unseen data before ML filtering.
Stage C	True Renko + MA + direct ML validation; unseen test	Measure the effect of ML acceptance thresholds on the same True Renko architecture.

3.2 Experimental Dimensions

ATR period: 14. ATR multiplier: 2.0. Brick sizes: 100, 200, 300, 400, 500 and 600. ML thresholds: 0.50, 0.53, 0.55, 0.57, 0.60, 0.65 and 0.70.

The supplied experiment registry contains 52 rows: 6 development runs, 6 unseen-test runs without ML, and 40 unseen-test runs with ML validation. Thresholds 0.50 and 0.53 are represented by five brick-size runs in the supplied registry, while the remaining thresholds are represented by six brick-size runs.

3.3 Evaluation Metrics

Metric	Definition
Total Orders	Number of recorded orders in the experiment run.
Wins / Losses	Counts reported by the supplied experiment registry.
Win %	Reported winning-trade percentage.
Total Profit	Aggregate profitable trade amount reported by the registry.
Total Loss	Aggregate losing trade amount reported by the registry.
Net Realized	Total Profit minus Total Loss.
Maximum Drawdown	Largest reported drawdown for the run.

Row-wise aggregation is descriptive only. Because different rows represent separate configurations over overlapping market periods, summed net realized values are not interpreted as the performance of a single pooled portfolio.

IV. RESULTS: TRUE RENKO WITHOUT ML

4.1 Development / Training Results

Brick	Orders	Wins	Losses	Win %	Profit	Loss	Net	Max DD
100	10,378	3,355	7,023	32.33%	727,573.04	610,166.71	117,406.33	10,786.88
200	9,790	3,325	6,464	33.96%	837,658.42	732,986.12	104,672.30	11,857.15
300	9,505	3,324	6,180	34.97%	869,831.95	770,484.48	99,347.47	15,247.48

400	9,346	3,329	6,016	35.62%	891,673.38	782,744.27	108,929.11	10,779.47
500	9,273	3,337	5,935	35.99%	880,431.17	787,903.92	92,527.25	22,442.68
600	9,229	3,331	5,897	36.09%	882,591.40	786,222.68	96,368.72	22,606.91

Across the six development configurations, the registry records 57,521 orders, a weighted win percentage of 34.77%, aggregate reported profit of 5,089,759.36, aggregate reported loss of 4,470,508.18, and descriptive row-wise net realized of 619,251.18. The largest reported drawdown is 22,606.91.

4.2 Unseen Test Results Without ML

Brick	Orders	Wins	Losses	Win %	Profit	Loss	Net	Max DD
100	22,794	6,864	15,929	30.11%	2,242,817.14	2,393,549.25	-150,732.11	465,784.03
200	14,105	4,942	9,163	35.04%	1,797,135.04	1,413,606.76	383,528.28	8,491.80
300	13,496	4,917	8,579	36.43%	1,880,376.53	1,475,804.00	404,572.53	10,597.58
400	13,168	4,890	8,278	37.14%	1,866,243.57	1,488,001.65	378,241.92	15,651.98
500	13,027	4,894	8,133	37.57%	1,914,392.16	1,484,831.69	429,560.47	17,402.14
600	12,949	4,879	8,070	37.68%	1,915,371.14	1,498,615.57	416,755.57	18,313.12

The six unseen no-ML configurations record 89,539 orders in total. The weighted win percentage is 35.05%. The row-wise descriptive net is 1,861,926.66, with the largest reported drawdown equal to 465,784.03. The 100-brick configuration is materially different from the other test rows, with negative reported net and a much larger drawdown; it is retained rather than removed because the purpose of the study is to report the supplied experiment registry without selective exclusion.

V. RESULTS: DIRECT ML VALIDATION OF TRUE RENKO SIGNALS

ML validation was applied to candidate True Renko signals on the unseen evaluation period. The validator did not replace the True Renko signal generator. Instead, it filtered candidate signals according to the configured confidence threshold. The threshold analysis therefore measures the change in accepted trading activity and performance within the same True Renko architecture.

ML Threshold	Runs	Orders	Weighted Win %	Profit	Loss	Net	Max Row DD
0.50	5	48,552	36.57%	6,123,427.09	4,943,398.41	1,180,028.68	15,393.80
0.53	5	43,341	36.46%	5,416,189.10	4,372,676.44	1,043,512.66	18,980.25
0.55	6	42,970	36.22%	5,312,744.47	4,244,023.44	1,068,721.03	17,576.59
0.57	6	37,503	36.09%	4,651,830.80	3,747,065.17	904,765.63	19,281.06
0.60	6	29,559	36.11%	3,709,281.81	2,979,600.75	729,681.06	19,212.37
0.65	6	18,192	36.44%	2,321,372.14	1,877,214.68	444,157.46	15,250.41
0.70	6	9,053	36.33%	1,243,550.31	1,001,345.07	242,205.24	16,311.33

5.1 Threshold Sensitivity

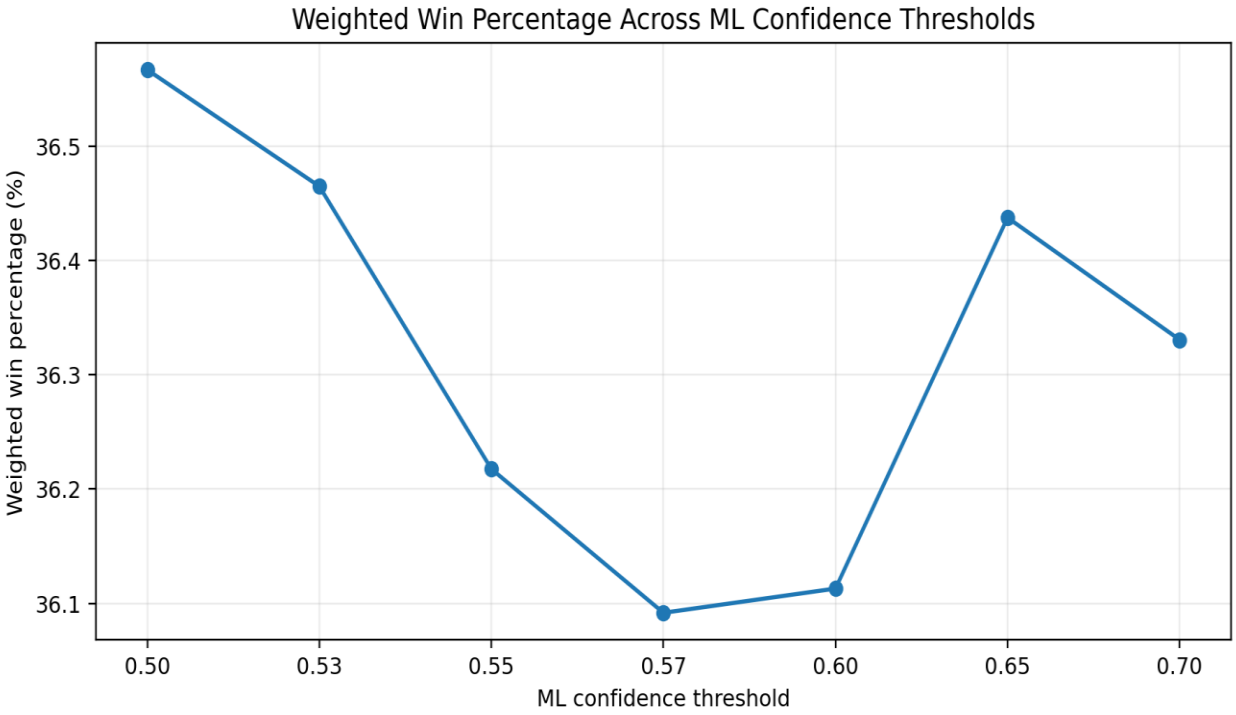

Figure 4. Weighted win percentage across the supplied ML-assisted experiment runs by confidence threshold.

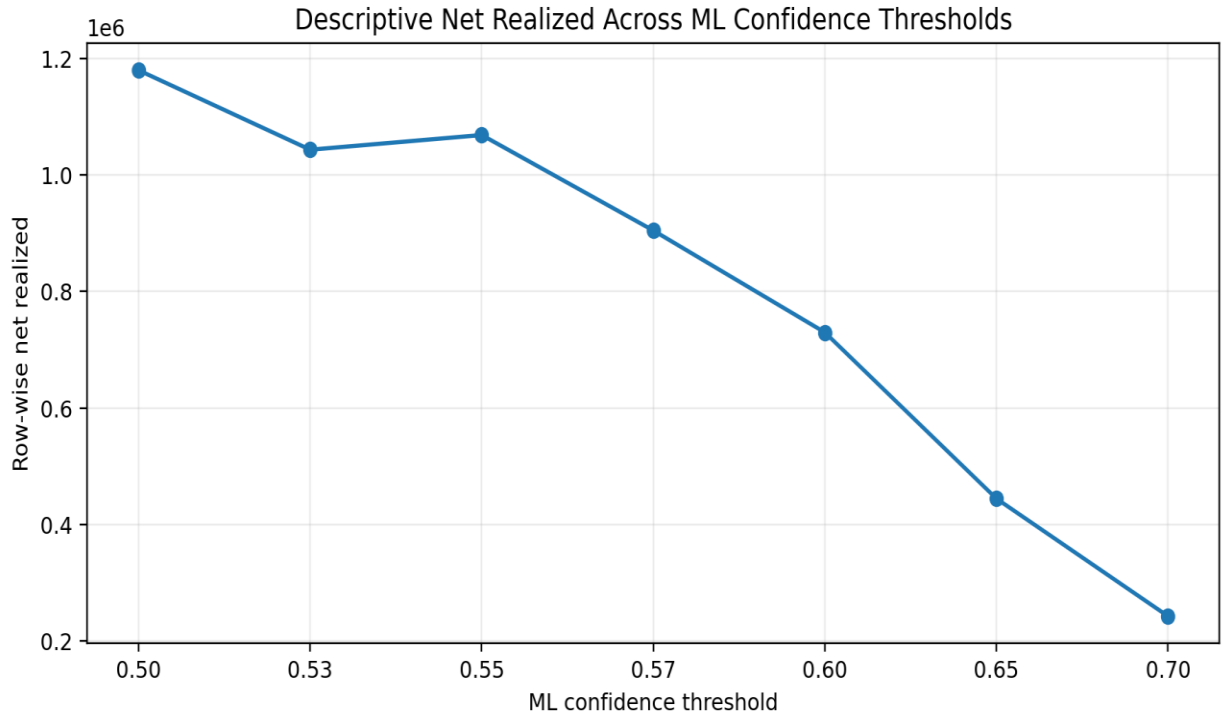

Figure 3. Descriptive row-wise net realized across the supplied ML-assisted experiment runs by confidence threshold.

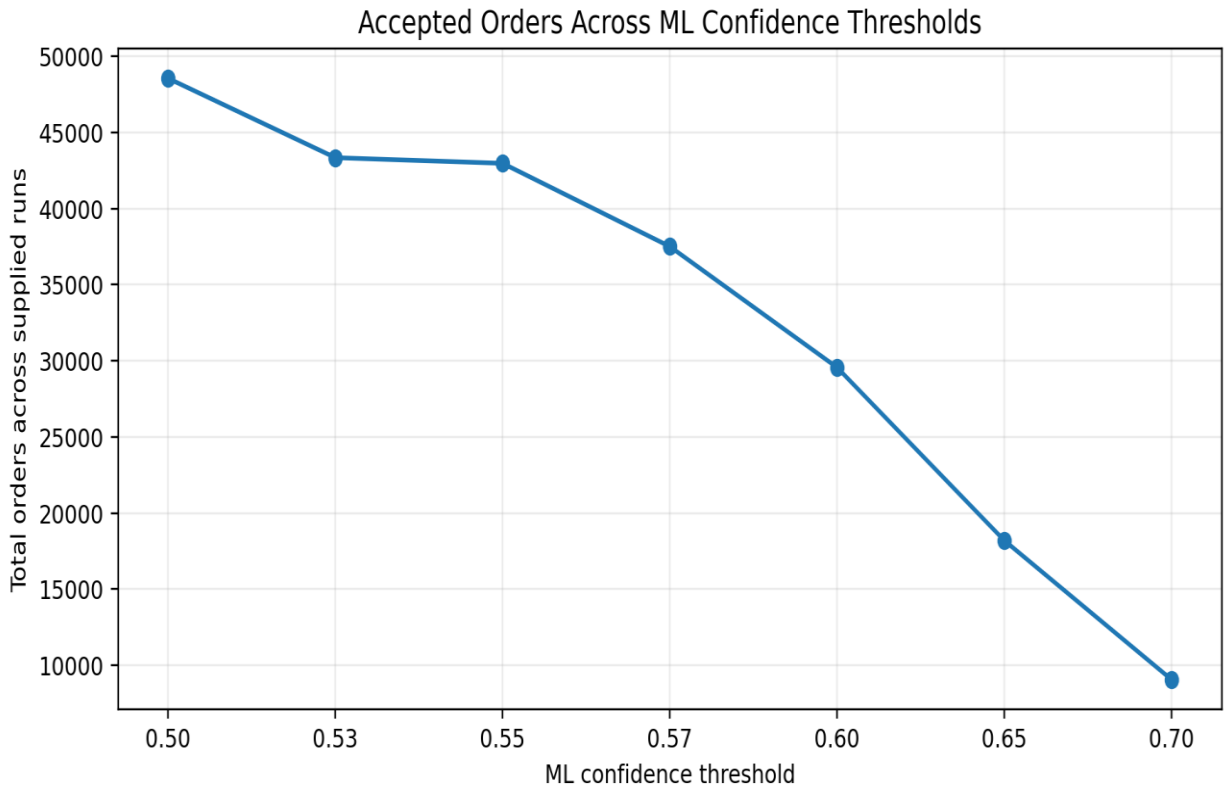

Figure 2. Total recorded orders across the supplied ML-assisted experiment runs as the validation threshold increases.

The registry shows a clear reduction in accepted trade activity as the ML threshold increases. The descriptive average orders per run fall from approximately 9,710 at threshold 0.50 to approximately 1,509 at threshold 0.70. This is consistent with the intended role of the ML component as a selective gate: a higher acceptance threshold allows fewer candidate signals to reach the trade-management stage.

Win percentage does not increase monotonically with the threshold. The weighted win percentage across the supplied ML runs is approximately 36.57% at 0.50, 36.46% at 0.53, 36.33% at 0.55, 36.09% at 0.57, 36.11% at 0.60, 36.44% at 0.65 and 36.33% at 0.70. Thus, the primary effect of increasing the threshold in this registry is stronger signal selection and lower activity rather than a simple monotonic improvement in win rate.

Because the thresholds are evaluated on the same unseen test period, the paper treats these values as a sensitivity analysis rather than declaring a single threshold to be universally optimal. Selecting a threshold for future deployment should be based on a development/validation period or an independent forward-test period rather than on the reported test-period result alone.

5.2 ML Validation by Brick Size

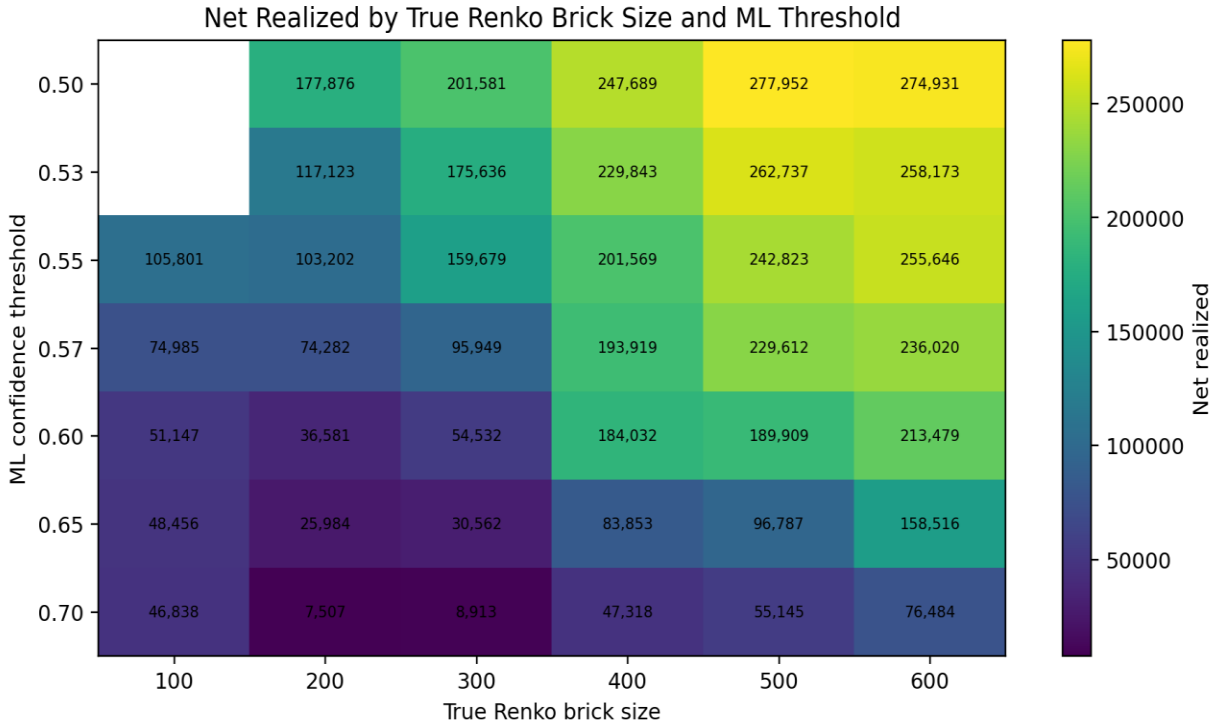

Figure 5. Net realized by True Renko brick size and ML confidence threshold.

Threshold	Brick	Orders	Win %	Net	Max DD
0.50	200	8,151	34.83%	177,876.37	10,421.53
0.50	300	9,140	35.86%	201,580.60	12,094.83
0.50	400	10,233	36.90%	247,688.76	10,692.63
0.50	500	10,507	37.44%	277,951.87	11,408.28
0.50	600	10,521	37.33%	274,931.08	15,393.80
0.53	200	6,503	35.00%	117,122.56	18,980.25
0.53	300	7,804	35.83%	175,636.07	13,106.63
0.53	400	9,380	36.73%	229,843.32	9,830.58
0.53	500	9,819	37.17%	262,737.43	12,070.47
0.53	600	9,835	36.98%	258,173.28	12,184.00
0.55	100	3,671	33.64%	105,801.42	5,884.05
0.55	300	6,787	35.77%	159,678.64	10,707.30
0.55	400	8,718	36.44%	201,569.24	11,724.29
0.55	500	9,262	36.99%	242,823.06	12,154.78
0.55	600	9,263	37.05%	255,646.34	11,454.84
0.57	100	2,635	33.93%	74,984.98	5,178.80
0.57	200	4,047	35.31%	74,281.50	19,281.06
0.57	300	5,631	34.95%	95,948.57	9,771.58
0.57	400	7,947	36.24%	193,919.02	12,804.70
0.57	500	8,637	36.88%	229,611.73	10,393.68
0.57	600	8,606	36.94%	236,019.83	11,559.85

0.60	100	1,626	32.84%	51,147.03	4,914.41
0.60	200	2,587	34.91%	36,581.18	16,016.20
0.60	300	3,919	34.37%	54,532.03	19,212.37
0.60	400	6,600	36.61%	184,032.13	12,838.16
0.60	500	7,366	36.75%	189,909.35	15,073.42
0.60	600	7,461	37.09%	213,479.34	13,012.92
0.65	100	910	34.73%	48,455.57	3,336.41
0.65	200	1,140	35.18%	25,984.22	5,304.69
0.65	300	1,755	35.21%	30,562.43	12,130.86
0.65	400	4,091	36.52%	83,852.85	15,250.41
0.65	500	5,064	36.35%	96,786.79	11,730.66
0.65	600	5,232	37.44%	158,515.60	13,778.62
0.70	100	612	36.44%	46,838.36	3,252.50
0.70	200	417	34.29%	7,506.81	3,073.32
0.70	300	575	35.13%	8,912.85	5,625.94
0.70	400	1,922	36.37%	47,318.10	12,101.89
0.70	500	2,719	36.04%	55,144.86	10,631.46
0.70	600	2,808	37.11%	76,484.26	16,311.33

5.3 Comparison Within the Same True Renko Architecture

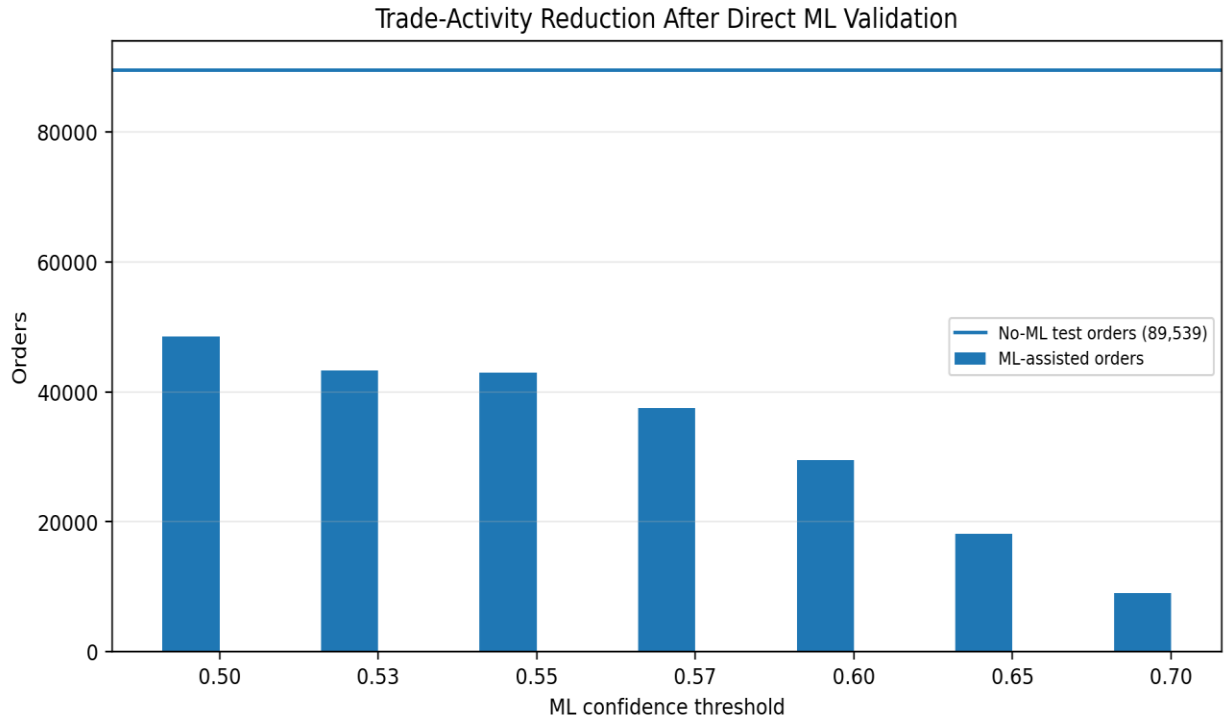

Figure 6. Reduction in trade activity after direct ML validation relative to the no-ML True Renko test reference.

The appropriate comparison is not between unrelated Renko systems; it is between the same True Renko + MA architecture before and after direct ML validation. The no-ML test contains six brick-

size configurations, while the ML runs apply the validator to candidate signals from the same strategy family. The ML stage substantially reduces order counts across all thresholds and preserves positive row-level net realized results in every supplied ML configuration.

All 39 of the 39 supplied ML-assisted runs report positive net realized results. This is an empirical property of the supplied registry and should not be interpreted as proof of future profitability.

VI. DISCUSSION

6.1 Role of ML in the True Renko Architecture

The results support the conceptual separation between signal generation and signal validation. True Renko remains the deterministic source of candidate opportunities, while the ML component acts as a selective acceptance mechanism. This separation is important for reproducibility because the effect of ML can be studied without changing the basic logic responsible for creating candidate signals.

6.2 Trade-Frequency Control

The strongest and most consistent threshold effect is on trade frequency. Increasing the threshold removes progressively more candidate signals. This provides a controllable trade-off between opportunity coverage and selectivity. A lower threshold permits more signals and therefore greater market exposure; a higher threshold produces fewer trades and stronger filtering.

6.3 Profitability and Drawdown

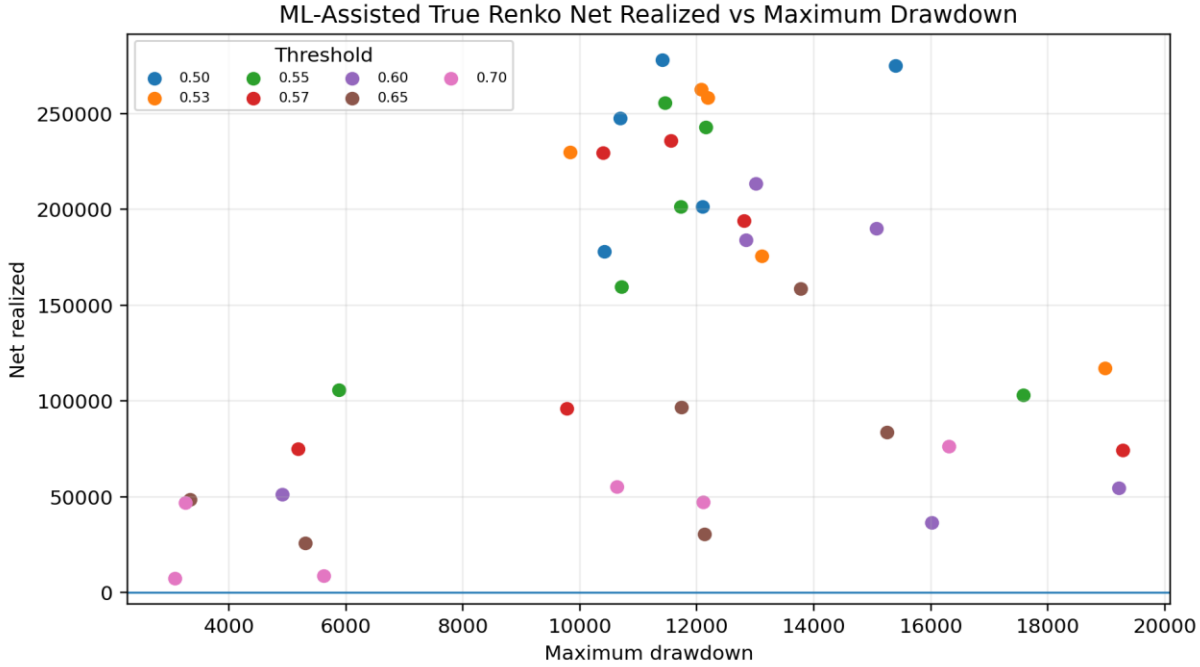

Figure 7. Net realized versus maximum drawdown for the supplied ML-assisted True Renko experiment runs.

Aggregate descriptive net realized decreases as the threshold becomes stricter, although the number of runs represented at 0.50 and 0.53 is smaller than at the higher thresholds. Maximum drawdown is not monotonic. These observations indicate that threshold selection is a multi-objective problem involving activity, profitability, win rate and risk rather than a single-score optimization problem.

6.4 Interpretation of the 100-Brick No-ML Result

The 100-brick no-ML test configuration reports negative net realized and substantially larger drawdown than the other no-ML test rows. It is not removed from the analysis. Retaining it is important because it demonstrates that the adaptive True Renko architecture has parameter sensitivity and that ML validation should be evaluated as a filtering mechanism rather than presented as a substitute for proper strategy configuration.

VII. REPRODUCIBILITY AND EXPERIMENTAL CONTROLS

- The strategy code used for the evaluation is the supplied True Renko implementation, including ATR Renko construction, MA filtering, feature generation and ML validation hooks.
- The ML layer is activated only after the True Renko candidate signal and MA filter have passed.
- The ML validator returns an acceptance decision and score; rejected candidates do not proceed to order submission.
- The experiment uses separated development/training and unseen evaluation periods.
- The supplied registry is reproduced in the appendix without selective removal of poor-performing configurations.
- Aggregate results across configurations are explicitly treated as descriptive and not as one portfolio equity curve.

VIII. THREATS TO VALIDITY AND LIMITATIONS

- Historical back testing does not establish live-market performance.
- The supplied results do not fully quantify exchange slippage, market impact, latency or order-book effects.
- The empirical asset is BTCUSDT; generalization to equities, commodities, foreign exchange or other cryptocurrencies is not established.
- Multiple brick sizes and ML thresholds create a multiple-comparison and selection risk.
- Threshold sensitivity is evaluated on the supplied test period; a deployment threshold should be frozen using development/validation data and then evaluated on a fresh forward period.
- The reported ML score is used as a thresholder validation confidence; calibration quality of the probability is not established by the registry alone.

- The study reports aggregate trade metrics but does not include a statistical significance test, confidence interval or bootstrap analysis.
- Row-wise sums across independent configurations are descriptive and should not be interpreted as a combined trading account.

IX. FUTURE RESEARCH

- Use a dedicated validation period to select the ML threshold before touching an independent final test set.
- Perform bootstrap confidence intervals and statistical tests for differences in trade outcomes.
- Evaluate probability calibration of the XGBoost validator.
- Introduce transaction costs, slippage and execution-latency modelling.
- Perform walk-forward validation and forward paper trading.
- Evaluate the strategy across additional crypto assets and market regimes.
- Study whether threshold adaptation based on volatility or market regime improves stability.
- Report trade-level confusion matrices and precision/recall for accepted versus rejected candidate signals.

X. CONCLUSION

This study presents an independent empirical evaluation of a proprietary adaptive True Renko trading strategy with direct machine-learning signal validation. The architecture preserves True Renko as the candidate signal generator and uses an XGBoost-based ML component only to determine whether a candidate signal is sufficiently validated for execution. The experimental design separates development/training, unseen no-ML testing and unseen ML-assisted testing.

The supplied 52-run registry demonstrates that ML validation materially reduces trade frequency as the confidence threshold increases. Across thresholds from 0.50 to 0.70, the weighted win rate remains in a relatively narrow range, while the number of accepted trades falls substantially. All supplied ML-assisted runs are profitable on a row-level net-realized basis, although this observation must be interpreted cautiously because the experiments are historical, multiple configurations are tested, and the same test period is reused for sensitivity analysis.

The principal contribution is therefore methodological: the work demonstrates a reproducible way to add supervised ML validation to a deterministic True Renko strategy without turning the ML model into an independent signal generator. The next stage should be independent threshold selection, walk-forward validation and forward paper trading with realistic execution assumptions.

REFERENCES

[1]. Lo, A. W., Mamaysky, H., & Wang, J. (2000). Foundations of Technical Analysis: Computational Algorithms, Statistical Inference, and Empirical Implementation. The Journal of Finance, 55(4), 1705–1765. https://doi.org/10.1111/0022-1082.00265

[2]. Drozda, M., Čavojský, M., & Sebes, P. (2024). On Suitability of Renko Charts for Algorithmic Trading. In Advances in Industrial Internet of Things and Engineering Management, 69–81. Springer. https://doi.org/10.1007/978-3-031-81261-3_6
[3]. Kumbure, M. M., Lohrmann, C., Luukka, P., & Porras, J. (2022). Machine learning techniques and data for stock market forecasting: A literature review. Expert Systems with Applications, 197, 116659.
[4]. Nazareth, N., & Reddy, Y. V. R. (2023). Financial applications of machine learning: A literature review. Expert Systems with Applications, 219, 119640.
[5]. Ludkovski, M. (2023). Statistical Machine Learning for Quantitative Finance. Annual Review of Statistics and Its Application, 10, 271–295.
[6]. Kelly, B. T., & Xiu, D. (2023). Financial Machine Learning. Foundations and Trends in Finance, 13(3–4), 205–363.
[7]. Sullivan, R., Timmermann, A., & White, H. (1999). Data-Snooping, Technical Trading Rule Performance, and the Bootstrap. The Journal of Finance, 54(5), 1647–1691. https://doi.org/10.1111/0022-1082.00163
[8]. Bailey, D. H., Borwein, J. M., López de Prado, M., & Zhu, Q. J. (2015). The Probability of Backtest Overfitting. Journal of Computational Finance.

List of Figures

Figure 1. Illustrative True Renko brick representation with MA filtering and ML acceptance/rejection.
Figure 2. Total recorded orders across ML confidence thresholds.
Figure 3. Descriptive net realized across ML confidence thresholds.
Figure 4. Weighted win percentage across ML confidence thresholds.
Figure 5. Net realized by True Renko brick size and ML confidence threshold.
Figure 6. Reduction in trade activity after direct ML validation.
Figure 7. Net realized versus maximum drawdown for ML-assisted True Renko runs.

Appendix A. Complete Supplied True Renko Experiment Registry

The following table reproduces the complete 52-run numerical registry used for this paper. Values are reported as supplied. Rows are ordered by the source registry sequence. No Base Renko strategy data are included.

#	Group	Brick	Orders	Wins	Losses	Win %	Profit	Loss	Net	Max DD
1	TEST-MA	500	13,027	4,894	8,133	37.57%	1,914,392.16	1,484,831.69	429,560.47	17,402.14
2	TEST-MA	600	12,949	4,879	8,070	37.68%	1,915,371.14	1,498,615.57	416,755.57	18,313.12
3	TEST-MA	300	13,496	4,917	8,579	36.43%	1,880,376.53	1,475,804.00	404,572.53	10,597.58

4	TEST-MA	200	14,105	4,942	9,163	35.04%	1,797,135.04	1,413,606.76	383,528.28	8,491.80
5	TEST-MA	400	13,168	4,890	8,278	37.14%	1,866,243.57	1,488,001.65	378,241.92	15,651.98
6	ML-TEST-0.50	500	10,507	3,934	6,573	37.44%	1,397,648.04	1,119,696.17	277,951.87	11,408.28
7	ML-TEST-0.50	600	10,521	3,928	6,593	37.33%	1,393,040.49	1,118,109.41	274,931.08	15,393.80
8	ML-TEST-0.53	500	9,819	3,650	6,169	37.17%	1,302,164.67	1,039,427.24	262,737.43	12,070.47
9	ML-TEST-0.53	600	9,835	3,637	6,198	36.98%	1,300,369.58	1,042,196.30	258,173.28	12,184.00
10	ML-TEST-0.55	600	9,263	3,432	5,831	37.05%	1,237,909.37	982,263.03	255,646.34	11,454.84
11	ML-TEST-0.50	400	10,233	3,776	6,457	36.90%	1,309,670.16	1,061,981.40	247,688.76	10,692.63
12	ML-TEST-0.55	500	9,262	3,426	5,836	36.99%	1,223,231.07	980,408.01	242,823.06	12,154.78
13	ML-TEST-0.57	600	8,606	3,179	5,427	36.94%	1,156,199.77	920,179.94	236,019.83	11,559.85
14	ML-TEST-0.53	400	9,380	3,445	5,935	36.73%	1,184,042.07	954,198.75	229,843.32	9,830.58
15	ML-TEST-0.57	500	8,637	3,185	5,452	36.88%	1,155,791.66	926,179.93	229,611.73	10,393.68
16	ML-TEST-0.60	600	7,461	2,767	4,694	37.09%	1,023,393.84	809,914.50	213,479.34	13,012.92
17	ML-TEST-0.50	300	9,140	3,278	5,862	35.86%	1,114,237.52	912,656.92	201,580.60	12,094.83
18	ML-TEST-0.55	400	8,718	3,177	5,541	36.44%	1,097,558.06	895,988.82	201,569.24	11,724.29

19	ML-TEST-0.57	400	7,947	2,880	5,067	36.24%	1,011,259.51	817,340.49	193,919.02	12,804.70
20	ML-TEST-0.60	500	7,366	2,707	4,659	36.75%	976,562.83	786,653.48	189,909.35	15,073.42
21	ML-TEST-0.60	400	6,600	2,416	4,184	36.61%	864,168.93	680,136.80	184,032.13	12,838.16
22	ML-TEST-0.50	200	8,151	2,839	5,312	34.83%	908,830.88	730,954.51	177,876.37	10,421.53
23	ML-TEST-0.53	300	7,804	2,796	5,008	35.83%	934,613.43	758,977.36	175,636.07	13,106.63
24	ML-TEST-0.55	300	6,787	2,428	4,359	35.77%	814,695.81	655,017.17	159,678.64	10,707.30
25	ML-TEST-0.65	600	5,232	1,959	3,273	37.44%	757,827.05	599,311.45	158,515.60	13,778.62
26	TRAIN-MA	100	10,378	3,355	7,023	32.33%	727,573.04	610,166.71	117,406.33	10,786.88
27	ML-TEST-0.53	200	6,503	2,276	4,227	35.00%	694,999.35	577,876.79	117,122.56	18,980.25
28	TRAIN-MA	400	9,346	3,329	6,016	35.62%	891,673.38	782,744.27	108,929.11	10,779.47
29	ML-TEST-0.55	100	3,671	1,235	2,436	33.64%	387,396.71	281,595.29	105,801.42	5,884.05
30	TRAIN-MA	200	9,790	3,325	6,464	33.96%	837,658.42	732,986.12	104,672.30	11,857.15
31	ML-TEST-0.55	200	5,269	1,865	3,404	35.40%	551,953.45	448,751.12	103,202.33	17,576.59
32	ML-TEST-0.65	500	5,064	1,841	3,223	36.35%	659,658.45	562,871.66	96,786.79	11,730.66
33	TRAIN-MA	300	9,505	3,324	6,180	34.97%	869,831.95	770,484.48	99,347.47	15,247.48

34	ML-TEST-0.57	300	5,631	1,968	3,663	34.95%	642,047.36	546,098.79	95,948.57	9,771.58
35	TRAIN-MA	600	9,229	3,331	5,897	36.09%	882,591.40	786,222.68	96,368.72	22,606.91
36	TRAIN-MA	500	9,273	3,337	5,935	35.99%	880,431.17	787,903.92	92,527.25	22,442.68
37	ML-TEST-0.65	400	4,091	1,494	2,597	36.52%	487,360.49	403,507.64	83,852.85	15,250.41
38	ML-TEST-0.70	600	2,808	1,042	1,766	37.11%	433,433.22	356,948.96	76,484.26	16,311.33
39	ML-TEST-0.57	100	2,635	894	1,741	33.93%	283,551.53	208,566.55	74,984.98	5,178.80
40	ML-TEST-0.57	200	4,047	1,429	2,618	35.31%	402,980.97	328,699.47	74,281.50	19,281.06
41	ML-TEST-0.70	500	2,719	980	1,739	36.04%	367,291.34	312,146.48	55,144.86	10,631.46
42	ML-TEST-0.60	300	3,919	1,347	2,572	34.37%	423,737.25	369,205.22	54,532.03	19,212.37
43	ML-TEST-0.60	100	1,626	534	1,092	32.84%	194,967.26	143,820.23	51,147.03	4,914.41
44	ML-TEST-0.65	100	910	316	594	34.73%	149,204.45	100,748.88	48,455.57	3,336.41
45	ML-TEST-0.70	400	1,922	699	1,223	36.37%	231,689.33	184,371.23	47,318.10	12,101.89
46	ML-TEST-0.70	100	612	223	389	36.44%	129,586.83	82,748.47	46,838.36	3,252.50
47	ML-TEST-0.60	200	2,587	903	1,684	34.91%	226,451.70	189,870.52	36,581.18	16,016.20
48	ML-TEST-0.65	300	1,755	618	1,137	35.21%	177,276.93	146,714.50	30,562.43	12,130.86

49	ML-TEST-0.65	200	1,140	401	739	35.18%	90,044.77	64,060.55	25,984.22	5,304.69
50	ML-TEST-0.70	300	575	202	373	35.13%	52,802.65	43,889.80	8,912.85	5,625.94
51	ML-TEST-0.70	200	417	143	274	34.29%	28,746.94	21,240.13	7,506.81	3,073.32
52	TEST-MA	100	22,794	6,864	15,929	30.11%	2,242,817.14	2,393,549.25	-150,732.11	465,784.03

Appendix B. Experimental Configuration Summary

Parameter	Setting
Asset	BTCUSDT
Historical span	Seven years
Development / training	Four years
Unseen evaluation	Three years
Renko type used in reported ML experiments	ATR-based True Renko
ATR period	14
ATR multiplier	2.0
Brick sizes	100, 200, 300, 400, 500, 600
MA filter	Enabled
ML model role	Direct True Renko signal validation
ML thresholds	0.50, 0.53, 0.55, 0.57, 0.60, 0.65, 0.70
ML model family	XGBoost supervised classification
Primary metrics	Orders, wins, losses, win %, profit, loss, net realized, max drawdown