

# InelidaMarketScan — Scientific Analysis of Financial Markets with Machine Learning

Quantitative Finance Research: ICT, Ichimoku, XGBoost & Stochastic Calculus By Didier Vally / Reuniware Systems Version 1.0.6 — 2026-07-19 □  
PyPI: <https://pypi.org/project/inelida-marketscan> | GitHub: <https://github.com/reuniware/InelidaMarketScanner>

---

## Abstract

This document presents the results of 6 months of quantitative finance research applied to algorithmic trading. The InelidaMarketScan framework combines ICT (Inner Circle Trader) technical analysis and multi-timeframe Ichimoku with an XGBoost Machine Learning pipeline (123 features) and stochastic calculus functions (Ornstein-Uhlenbeck, Hurst, GARCH, Monte Carlo, Kelly). Model v18 achieves 64.0% cross-validation accuracy on 1,210 historical trades. Analysis of 27 successive models reveals that Ichimoku features (Kijun/Tenkan distances, T/K crosses) are more predictive than ICT patterns (FVG, Order Blocks). Capital simulation with  $ML\% \geq 50\%$  filtering shows a profit factor of 2.27 on in-sample data, while out-of-sample validation identifies a ceiling at 1.173. This document serves as a comprehensive scientific reference on the intersection of ICT, Ichimoku, and Machine Learning.

---

## 1. Introduction

### 1.1 Context and Problem Statement

Algorithmic trading relies on identifying recurring patterns in financial time series. ICT (Inner Circle Trader) concepts developed by Michael Huddleston describe institutional market behavior: liquidity sweeps, Fair Value Gaps, Order Blocks, Market Structure Shifts. In parallel, the Ichimoku Kinko Hyo indicator provides a multi-timeframe reading of market equilibrium. This project explores the fusion of these two paradigms through an automated scanner coupled with an XGBoost model.

### 1.2 Research Objectives

1. Develop an automated scanner combining ICT and Ichimoku across 8 timeframes (M5 to MN).
2. Build an ML pipeline with 123 named features extracted from the scanner.
3. Validate predictive power via historical backtest (2025-2026) and out-of-sample testing.
4. Integrate quantitative finance functions (OU, Hurst, GARCH, Monte Carlo, Kelly).
5. Achieve a profit factor  $> 1.5$  under real conditions.

## 2. ICT Theoretical Foundations

ICT concepts model the behavior of financial institutions (banks, hedge funds) that manipulate price levels to accumulate positions. The following 8 concepts are automatically detected by the scanner:

| ICT Concept                      | Description and Detection                                                                                                                 |
|----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------|
| Liquidity Sweeps                 | Brief break of a key level (Asian High/Low, London High/Low) to trigger stops before reversal. Detected on H1 with reversal confirmation. |
| Fair Value Gaps (FVG)            | Gap between 3 consecutive candles where price did not trade. Detected on 6 TFs (M5 to D1) with mitigation percentage.                     |
| Order Blocks (OB)                | Last candle before a directional impulse — zone where institutions placed orders. Detected via ATR(14) on 6 TFs.                          |
| Breaker Blocks (BRK)             | Broken Order Block that reverses into opposite support/resistance. Derived from existing OBs, zero additional MT5 calls.                  |
| Market Structure Shift (MSS/BOS) | Change in market structure: Break of Structure and Change of Character on 6 TFs.                                                          |
| Volume HVN/LVN                   | High Volume Node and Low Volume Node based on MT5 tick volume. Detected on 6 TFs.                                                         |
| Reversal Pipeline                | Complete sweep → reversal → confirmation chain with post-sweep bar counter.                                                               |
| Fibonacci Extensions             | +1.618 to +4.0 extensions based on Asian range for TP targets.                                                                            |

### 3. Multi-TF Ichimoku Architecture

The Diamond scanner analyzes 8 timeframes simultaneously. For each TF, 5 Ichimoku components are computed:

| Timeframe | Tenkan                   | Kijun                    | Kumo                     | T/K Cross                | Flat History             |
|-----------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| M5        | <input type="checkbox"/> | <input type="checkbox"/> | —                        | <input type="checkbox"/> | —                        |
| M15       | <input type="checkbox"/> | <input type="checkbox"/> | —                        | <input type="checkbox"/> | —                        |
| M30       | <input type="checkbox"/> | <input type="checkbox"/> | —                        | <input type="checkbox"/> | —                        |
| H1        | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| H4        | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| D1        | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| W1        | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |
| MN        | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> | <input type="checkbox"/> |

### 4. Diamond Scanner Architecture

The Diamond Scanner aggregates 109 scoring criteria across 10 categories:

| Category               | Criteria | Examples                                                       |
|------------------------|----------|----------------------------------------------------------------|
| Ichimoku HI-D1         | 24       | Kijun/Tenkan distances, T/K crosses, Kumo positions, Flat bars |
| Ichimoku W1-MN         | 8        | Kijun W1/MN, Kumo W1/MN, Cloud color, Double memory            |
| FVG (6 TFs)            | 12       | FVG bear/bull M5 to D1, mitigation %, time priority            |
| Order Blocks (6 TFs)   | 12       | OB bear/bull M5 to D1, position ABOVE/BELOW/AT                 |
| MSS/BOS (6 TFs)        | 18       | Regime, BOS, CHoCH on M5 to D1                                 |
| Volume HVN/LVN (6 TFs) | 12       | Volume spike, HVN, LVN on M5 to D1                             |
| Breaker Blocks (6 TFs) | 12       | BRK bear/bull M5 to D1                                         |
| Asian/London Sweeps    | 4        | AH/AL sweep, London H/L sweep                                  |
| Reversal Pipeline      | 5        | Sweep→reversal→confirmation, post-sweep FVG, retest            |
| M30/M15/M5 Kijun       | 2        | Kijun M30/M15/M5 directional cross                             |

**Maximum score:** 109 criteria (107 without MN)

## 5. Machine Learning Pipeline

The ML pipeline transforms scanner results into win probability through 3 stages:

| Stage | Tool            | Details                                                                                                                                     |
|-------|-----------------|---------------------------------------------------------------------------------------------------------------------------------------------|
| 1     | ml_labeler.py   | Feature extraction from Diamond scan (DB / manual CSV / backtest). Generates labeled CSV with outcome column (1=won, 0=lost).               |
| 2     | ml_trainer.py   | XGBoost training with 5-fold cross-validation, scale_pos_weight for imbalanced classes, hyperparameter grid search.                         |
| 3     | ml_predictor.py | Real-time inference: extract_features() → predict_proba() → ML% displayed in live scan. Simplified MLPredictor: 1 joblib.load instead of 5. |

## 6. XGBoost Model v18 — Configuration

Model v18 is the active production model. Trained on 1,210 historical trades with 123 named features (including GARCH and stochastic).

| Hyperparameter   | Value                          |
|------------------|--------------------------------|
| max_depth        | 6                              |
| learning_rate    | 0.03                           |
| n_estimators     | 200                            |
| subsample        | 0.8                            |
| min_child_weight | 3                              |
| gamma            | 0.1                            |
| reg_alpha        | 0.5 (L1)                       |
| reg_lambda       | 1.0 (L2)                       |
| scale_pos_weight | auto (based on win/loss ratio) |

**123 named features** include: scores/bias (6), Kijun 6 TFs (18), Tenkan 6 TFs (18), T/K crosses (6), Kumo (6), M5/M15/M30 crosses (12), Flat bars (2), Active FVGs (1), Order Blocks (6), Sweeps (5), Volume (6), MSS/BOS (6), Breaker Blocks (6), Compression (1), Stochastic (8: OU, Hurst, Monte Carlo, GARCH), Temporal (3), Asset type (3).

## 7. Model History

27 XGBoost models have been trained and evaluated since the project’s inception. Here is the history of major versions:

| Model | Data           | Features      | Trades | CV     | OOS PF   |
|-------|----------------|---------------|--------|--------|----------|
| v1    | 6 majors       | 36            | 852    | 61.0%  | —        |
| v4    | 13 symbols     | 111           | 1,210  | 62.9%  | 1.26*    |
| v7    | 13 symbols     | 25 best       | 2,923  | 60.1%  | 1.173    |
| v13   | 4 models/asset | 25            | 2,923  | 53-64% | 4.000    |
| v15   | 2025-2026      | 25 + OU/Hurst | 5,009  | 59.8%  | —        |
| v18   | 13 symbols     | 123 (GARCH)   | 1,210  | 64.0%  | 0.503 F1 |

## 8. Feature Importance (v4)

Feature importance analysis reveals which criteria carry the most weight in model decisions:

| Rank | Feature       | Gain | Insight                                                         |
|------|---------------|------|-----------------------------------------------------------------|
| 1    | is_dxy        | 14.2 | The dollar follows a radically different logic from forex pairs |
| 2    | rr            | 10.0 | Theoretical risk/reward ratio predicts actual outcome           |
| 3    | tkx_h1        | 9.2  | H1 Tenkan/Kijun cross is the single best predictor              |
| 4    | kj_h1         | 6.9  | Distance to H1 Kijun captures short-term equilibrium            |
| 5    | s1            | 6.4  | Stop-loss level is as important as take-profit                  |
| 6    | day_wednesday | 6.4  | Wednesdays (FOMC) show statistically distinct behavior          |
| 7    | d_kj_h4       | 6.3  | Distance to H4 Kijun indicates underlying trend                 |
| 8    | d_kj_d1       | 6.0  | Distance to D1 Kijun anchors macro context                      |
| 9    | tkx_h4        | 6.0  | H4 T/K cross confirms session trend                             |
| 10   | is_forex      | 5.9  | FX pairs have distinct behavior vs indices/metals               |

## 9. Out-of-Sample Validation

OOS validation (training January-June 2026, testing July 2026) is the decisive test. Model v4 shows an in-sample PF of 2.27 at  $ML\% \geq 50\%$  threshold, but this includes data leakage (the model saw 80% of test data during training). The true OOS PF of v7 is 1.173 — a statistically significant but modest edge. Model v13 (per-asset) achieves PF=4.000 on 15 trades, but the sample is too small to be reliable. Conclusion: ML provides a real edge ( $PF > 1.0$ ), but the current ceiling is  $\sim 1.17$  with a unified architecture. Improvement will come from more data and potentially asset-specialized models.

## 10. Capital Simulation — ML% Filtering

Simulation on 1,210 historical trades with initial capital of €10,000 and 1% risk per trade:

| ML% Threshold    | Trades | Win Rate | PF (RR) | P&L (€10k) |
|------------------|--------|----------|---------|------------|
| No filter        | 1,210  | 37.6%    | 1.02    | +€1,442    |
| $ML\% \geq 45\%$ | 661    | 54.3%    | 1.91 □  | +€27,542   |
| $ML\% \geq 50\%$ | 504    | 59.7%    | 2.27 □□ | +€25,842   |
| $ML\% \geq 55\%$ | 327    | 67.3%    | 2.80    | +€19,242   |
| $ML\% \geq 60\%$ | 170    | 78.8%    | 3.59    | +€9,340    |
| $ML\% \geq 70\%$ | 71     | 88.7%    | 1.64    | +€514      |

## 11. Quantitative Finance & Stochastic Calculus

6 quantitative finance functions have been integrated into the scanner and ML pipeline. These functions compute risk, volatility, and market regime metrics:

| Function                  | Purpose                                                                                          | ML Feature        | v18 Rank   |
|---------------------------|--------------------------------------------------------------------------------------------------|-------------------|------------|
| <b>Ornstein-Uhlenbeck</b> | Detects price overextension from long-term mean. Score 0-100%: >80% = likely reversal.           | ou_score          | Not tested |
| <b>Hurst Exponent</b>     | Classifies market regime: $H < 0.45$ = RANGE (mean-reversion), $H > 0.55$ = TREND (momentum).    | hurst_h           | Not tested |
| <b>GARCH(1,1)</b>         | Models conditional volatility. 4 features: persistence, long_run_vol, half_life, forecast_vol_1. | garch_persistence | #14        |
| <b>Monte Carlo</b>        | Simulates 10,000 price trajectories. Computes P(SL hit) and P(TP hit).                           | mc_p_sl, mc_p_tp  | Not tested |
| <b>Kelly Criterion</b>    | Optimal position sizing by asset type. Gain=0.00 (redundant with already-optimized SL/TP).       | kelly_full        | #124       |
| <b>FTMO Ruin</b>          | Ruin probability for a prop firm account with 10% drawdown over 100 simulated trades.            | ftmo_ruin_prob    | Not tested |

## 12. Kelly Criterion by Asset Type

The Kelly Criterion uses real win rates by asset type (from backtest) rather than a uniform win\_rate=0.40. Results:

| Asset Type     | Win Rate | Kelly Full | Kelly Half |
|----------------|----------|------------|------------|
| <b>DXY</b>     | 73%      | 46.0%      | 23.0%      |
| <b>XAUUSD</b>  | 42%      | 13.0%      | 6.5%       |
| <b>Forex</b>   | 31%      | 3.5%       | 1.8%       |
| <b>Indices</b> | 38%      | 7.0%       | 3.5%       |
| <b>Crypto</b>  | 35%      | 2.5%       | 1.3%       |

## 13. Simplified MLPredictor (2026-07-19)

The MLPredictor previously loaded the same model 5 times (1× per prediction method).

The simplification of 2026-07-19 reduced this to a single `joblib.load()`. Result: 30 lines of code removed, loading time divided by 4, and the redundant `_ASSET_MODEL_MAP` eliminated. Model v18 is now the single unified model for all asset types.

## 14. Bug Fixes — 2026-07-19

A thorough code review on 2026-07-19 identified and fixed 4 bugs:

| # | Severity | File                    | Description                                                                                                                                                  |
|---|----------|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | CRITICAL | diamond_scanner.py      | Dead code after return in <code>ml_available</code> — symbols were never activated. <code>initialize()</code> did not set <code>_initialized = True</code> . |
| 2 | HIGH     | stochastic_analytics.py | GARCH: if no valid $(\alpha, \beta)$ pair found, silent fallback with invented parameters. Added <code>best_ll &lt;= -1e100</code> check.                    |
| 3 | MEDIUM   | stochastic_analytics.py | Monte Carlo: log-return volatility fallback was computed but never used when OU sigma was zero.                                                              |
| 4 | LOW      | stochastic_analytics.py | FTMO: <code>pnl_avg/roi_avg</code> outside try block — risk of <code>NameError</code> if code added between computation and return.                          |

## 15. Final Decision Matrix

The decision matrix combines Diamond score, ML%, safety guards, and macro context:

| Condition                                      | Action                             | Rationale                                              |
|------------------------------------------------|------------------------------------|--------------------------------------------------------|
| ML% $\geq$ 60% + TKx HI aligned + no HIGH NEWS | <input type="checkbox"/> GO        | Triple confirmation: ML, Ichimoku, macro               |
| ML% $\geq$ 60% + TKx HI conflict               | <input type="checkbox"/> WAIT      | ML is contradicted by T/K HI cross                     |
| ML% 50-60% + TKx HI aligned                    | <input type="checkbox"/> CAUTION   | Moderate ML edge, needs additional confirmation        |
| ML% < 50%                                      | <input type="checkbox"/> NO GO     | Model has no statistical edge                          |
| HIGH NEWS DAY + BULL                           | <input type="checkbox"/> WAIT      | Macro days ( $\geq$ 2 major events) break correlations |
| Wednesday (FOMC day) + Forex                   | <input type="checkbox"/> High Risk | Wednesdays show statistically distinct behavior        |

## 16. Conclusion and Perspectives

The InelidaMarketScan project demonstrates that an ML pipeline trained on Ichimoku features can generate a statistically significant edge (OOS PF = 1.173). The 4 critical bugs fixed on 2026-07-19 strengthen live scanner reliability. The 27 successive models revealed key insights: Ichimoku features (Kijun/Tenkan distances) are more predictive than ICT patterns, asset type (is\_dxy, is\_forex) is a major predictor, and stochastic calculus functions (GARCH, OU, Hurst) provide complementary volatility signal. The next performance leap will require a larger dataset (2024-2026, >5,000 trades/asset) and potentially deep learning architectures (LSTM, Transformer) to capture temporal dependencies that XGBoost does not model.

## References

- Huddleston, M. — Inner Circle Trader (ICT) Concepts, 2022-2024
- Chen, T. & Guestrin, C. — XGBoost: A Scalable Tree Boosting System, KDD 2016
- Bollerslev, T. — Generalized Autoregressive Conditional Heteroskedasticity, Journal of Econometrics, 1986
- Hurst, H.E. — Long-term Storage Capacity of Reservoirs, Trans. American Society of Civil Engineers, 1951
- Uhlenbeck, G.E. & Ornstein, L.S. — On the Theory of Brownian Motion, Physical Review, 1930
- Kelly, J.L. — A New Interpretation of Information Rate, Bell System Technical Journal, 1956
- Hosoda, G. — Ichimoku Kinko Hyo, 1968

Quantitative finance research work. Reproduction permitted with citation.

### ☐ Community

Join the **Trading Pro** Discord server to connect with the community:  
<https://discord.gg/u9JApXqhXy>

Didier Vally / Reuniware Systems — InelidaMarketScan v1.0.6 2026-07-19