

InelidaMarketScan — Análise Científica dos Mercados Financeiros com Machine Learning

Pesquisa em Finanças Quantitativas: ICT, Ichimoku, XGBoost e Cálculo Estocástico Por **Didier Vally** / Reuniware Systems Versão 1.0.6 — 2026-07-19
□ PyPI: <https://pypi.org/project/inelida-marketscan> | GitHub: <https://github.com/reuniware/InelidaMarketScanner>

Resumo

Este documento apresenta os resultados de 6 meses de pesquisa em finanças quantitativas aplicadas ao trading algorítmico. O framework InelidaMarketScan combina análise técnica ICT (Inner Circle Trader) e Ichimoku multi-timeframe com um pipeline de Machine Learning XGBoost (123 features) e funções de cálculo estocástico (Ornstein-Uhlenbeck, Hurst, GARCH, Monte Carlo, Kelly). O modelo v18 atinge 64,0% de precisão em validação cruzada em 1.210 trades históricos. A análise de 27 modelos sucessivos revela que as features Ichimoku (distâncias Kijun/Tenkan, cruzamentos T/K) são mais preditivas que os padrões ICT (FVG, Order Blocks). A simulação de capital com filtragem $ML\% \geq 50\%$ mostra um fator de lucro de 2,27 em dados in-sample, enquanto a validação out-of-sample identifica um teto em 1,173. Este documento constitui uma referência científica completa sobre a interseção entre ICT, Ichimoku e Machine Learning.

1. Introdução

1.1 Contexto e Problemática

O trading algorítmico baseia-se na identificação de padrões recorrentes nas séries temporais financeiras. Os conceitos ICT (Inner Circle Trader) desenvolvidos por Michael Huddleston descrevem o comportamento institucional dos mercados: varreduras de liquidez, Fair Value Gaps, Order Blocks, Market Structure Shifts. Paralelamente, o indicador Ichimoku Kinko Hyo fornece uma leitura multi-timeframe do equilíbrio do mercado. Este projeto explora a fusão destes dois paradigmas através de um scanner automatizado acoplado a um modelo XGBoost.

1.2 Objetivos de Pesquisa

- Desenvolver um scanner automático combinando ICT e Ichimoku em 8 timeframes (M5 a MN).
- Construir um pipeline ML com 123 features nomeadas extraídas do scanner.
- Validar o poder preditivo via backtest histórico (2025-2026) e teste out-of-sample.
- Integrar funções de finanças quantitativas (OU, Hurst, GARCH, Monte Carlo, Kelly).
- Atingir um fator de lucro $> 1,5$ em condições reais.

2. Fundamentos Teóricos ICT

Os conceitos ICT modelam o comportamento das instituições financeiras (bancos, hedge funds) que manipulam os níveis de preço para acumular posições. Os 8 conceitos seguintes são detectados automaticamente pelo scanner:

Conceito ICT	Descrição e Detecção
Varreduras de Liquidez	Ruptura breve de um nível-chave (Asian High/Low, London High/Low) para acionar stops antes da reversão. Detectado em H1 com confirmação de reversão.
Fair Value Gaps (FVG)	Lacuna entre 3 velas consecutivas onde o preço não negociou. Detectado em 6 TFs (M5 a D1) com porcentagem de mitigação.
Order Blocks (OB)	Última vela antes de um impulso direcional — zona onde as instituições colocaram ordens. Detectado via ATR(14) em 6 TFs.
Breaker Blocks (BRK)	Order Block quebrado que se inverte em suporte/resistência oposto. Derivado dos OBs existentes.
Market Structure Shift (MSS/BOS)	Mudança na estrutura do mercado: Break of Structure e Change of Character em 6 TFs.
Volume HVN/LVN	High Volume Node e Low Volume Node baseados no tick volume MT5. Detectado em 6 TFs.
Reversal Pipeline	Cadeia completa varredura → reversão → confirmação com contador de barras pós-varredura.
Extensões de Fibonacci	Extensões +1,618 a +4,0 baseadas no range asiático para alvos de TP.

3. Arquitetura Ichimoku Multi-TF

O scanner Diamond analisa 8 timeframes simultaneamente. Para cada TF, 5 componentes Ichimoku são calculados:

Timeframe	Tenkan	Kijun	Kumo	T/K Cross	Flat History
M5	☐	☐	—	☐	—
M15	☐	☐	—	☐	—
M30	☐	☐	—	☐	—
H1	☐	☐	☐	☐	☐
H4	☐	☐	☐	☐	☐
D1	☐	☐	☐	☐	☐
W1	☐	☐	☐	☐	☐
MN	☐	☐	☐	☐	☐

4. Arquitetura do Scanner Diamond

O Diamond Scanner agrega 109 critérios de pontuação em 10 categorias:

Categoria	Critérios	Exemplos
Ichimoku HI-D1	24	Distâncias Kijun/Tenkan, cruzamentos T/K, posições Kumo, Flat bars
Ichimoku W1-MN	8	Kijun W1/MN, Kumo W1/MN, cor da nuvem, Dupla memória
FVG (6 TFs)	12	FVG bear/bull M5 a D1, % de mitigação, prioridade temporal
Order Blocks (6 TFs)	12	OB bear/bull M5 a D1, posição ACIMA/ABAIXO/EM
MSS/BOS (6 TFs)	18	Regime, BOS, CHoCH em M5 a D1
Volume HVN/LVN (6 TFs)	12	Pico de volume, HVN, LVN em M5 a D1
Breaker Blocks (6 TFs)	12	BRK bear/bull M5 a D1
Varreduras Asian/London	4	Varredura AH/AL, varredura London H/L
Reversal Pipeline	5	Varredura→reversão→confirmação, FVG pós-varredura, reteste
Kijun M30/M15/M5	2	Cruzamento direcional Kijun M30/M15/M5

Pontuação máxima: 109 critérios (107 sem MN)

5. Pipeline de Machine Learning

O pipeline ML transforma os resultados do scanner em probabilidade de ganho em 3 etapas:

Etapas	Ferramenta	Detalhes
1	ml_labeler.py	Extração de features do scan Diamond (BD / CSV manual / backtest). Gera CSV rotulado com coluna outcome (1=ganho, 0=perda).
2	ml_trainer.py	Treinamento XGBoost com validação cruzada 5-fold, scale_pos_weight para classes desequilibradas, grid search de hiperparâmetros.
3	ml_predictor.py	Inferência em tempo real: extract_features() → predict_proba() → ML% exibido no scan ao vivo. MLPredictor simplificado: 1 joblib.load em vez de 5.

6. Modelo XGBoost v18 — Configuração

O modelo v18 é o modelo de produção ativo. Treinado em 1.210 trades históricos com 123 features nomeadas (incluindo GARCH e estocástico).

Hiperparâmetro	Valor
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 (baseado na razão win/loss)

123 features nomeadas incluem: pontuações/viés (6), Kijun 6 TFs (18), Tenkan 6 TFs (18), cruzamentos T/K (6), Kumo (6), cruzamentos M5/M15/M30 (12), Flat bars (2), FVGs ativos (1), Order Blocks (6), Varreduras (5), Volume (6), MSS/BOS (6), Breaker Blocks (6), Compressão (1), Estocástico (8: OU, Hurst, Monte Carlo, GARCH), Temporal (3), Tipo de ativo (3).

7. Histórico de Modelos

27 modelos XGBoost foram treinados e avaliados desde o início do projeto. Aqui está o histórico das versões principais:

Modelo	Dados	Features	Trades	CV	OOS PF
v1	6 principais	36	852	61.0%	—
v4	13 símbolos	111	1.210	62.9%	1.26*
v7	13 símbolos	25 melhores	2.923	60.1%	1.173
v13	4 modelos/ativo	25	2.923	53-64%	4.000
v15	2025-2026	25 + OU/Hurst	5.009	59.8%	—
v18	13 símbolos	123 (GARCH)	1.210	64.0%	0.503 F1

8. Importância das Features (v4)

A análise de importância das features revela quais critérios pesam mais nas decisões do modelo:

Rank	Feature	Ganho	Insight
1	is_dxy	14.2	O dólar segue uma lógica radicalmente diferente dos pares forex
2	rr	10.0	A razão risk/reward teórica prevê o resultado real
3	tkx_h1	9.2	O cruzamento Tenkan/Kijun H1 é o melhor preditor individual
4	kj_h1	6.9	A distância à Kijun H1 captura o equilíbrio de curto prazo
5	s1	6.4	O nível de stop-loss é tão importante quanto o take-profit
6	day_wednesday	6.4	Quartas-feiras (FOMC) mostram comportamento estatisticamente distinto
7	d_kj_h4	6.3	A distância à Kijun H4 indica a tendência subjacente
8	d_kj_d1	6.0	A distância à Kijun D1 ancora o contexto macro
9	tkx_h4	6.0	O cruzamento T/K H4 confirma a tendência de sessão
10	is_forex	5.9	Os pares FX têm comportamento próprio vs índices/metals

9. Validação Out-of-Sample

A validação OOS (treinamento Janeiro-Junho 2026, teste Julho 2026) é o teste decisivo. O modelo v4 mostra um PF in-sample de 2,27 no limiar $ML\% \geq 50\%$, mas isso inclui data leakage (o modelo viu 80% dos dados de teste durante o treinamento). O verdadeiro PF OOS do v7 é 1,173 — uma vantagem estatisticamente significativa mas modesta. O modelo v13 (por ativo) atinge PF=4,000 em 15 trades, mas a amostra é muito pequena para ser confiável. Conclusão: o ML fornece uma vantagem real ($PF > 1,0$), mas o teto atual é ~1,17 com arquitetura unificada.

10. Simulação de Capital — Filtragem ML%

Simulação em 1.210 trades históricos com capital inicial de 10.000 € e risco de 1% por trade:

Limiar ML%	Trades	Taxa de Acerto	PF (RR)	L&P (10k€)
Sem filtro	1.210	37.6%	1.02	+1.442 €
ML% $\geq$ 45%	661	54.3%	1.91 $\square$	+27.542 €
ML% $\geq$ 50%	504	59.7%	2.27 $\square\square$	+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. Finanças Quantitativas e Cálculo Estocástico

6 funções de finanças quantitativas foram integradas ao scanner e ao pipeline ML. Estas funções calculam métricas de risco, volatilidade e regime de mercado:

Função	Objetivo	Feature ML	Rank v18
Ornstein-Uhlenbeck	Detecta a sobreextensão do preço em relação à sua média de longo prazo. Score 0-100%: >80% = reversão provável.	ou_score	Não testado
Expoente de Hurst	Classifica o regime de mercado: $H < 0.45$ = RANGE, $H > 0.55$ = TENDÊNCIA.	hurst_h	Não testado
GARCH(1,1)	Modela a volatilidade condicional. 4 features: persistência, vol_longo_prazo, meia-vida, vol_previsão.	garch_persistence	#14
Monte Carlo	Simula 10.000 trajetórias de preço. Calcula P(SL atingido) e P(TP atingido).	mc_p_sl, mc_p_tp	Não testado
CrITÉrio de Kelly	Dimensionamento ótimo de posições por tipo de ativo. Ganho=0.00 (redundante com SL/TP já otimizados).	kelly_full	#124
Ruína FTMO	Probabilidade de ruína de uma conta prop firm com drawdown de 10% em 100 trades simulados.	ftmo_ruin_prob	Não testado

12. Critério de Kelly por Tipo de Ativo

O Critério de Kelly usa taxas de acerto reais por tipo de ativo (do backtest) em vez de um

win_rate=0.40 uniforme. Resultados:

Tipo de Ativo	Taxa de Acerto	Kelly Cheio	Kelly Meio
DXY	73%	46.0%	23.0%
XAUUSD	42%	13.0%	6.5%
Forex	31%	3.5%	1.8%
Índices	38%	7.0%	3.5%
Cripto	35%	2.5%	1.3%

13. MLPredictor Simplificado (19/07/2026)

O MLPredictor carregava anteriormente o mesmo modelo 5 vezes (1× por método de predição). A simplificação de 19/07/2026 reduziu isso a um único `joblib.load()`. Resultado: 30 linhas de código removidas, tempo de carregamento dividido por 4, e o `_ASSET_MODEL_MAP` redundante eliminado. O modelo v18 é agora o modelo unificado único para todos os tipos de ativos.

14. Correção de Bugs — 19/07/2026

Uma revisão de código minuciosa em 19/07/2026 identificou e corrigiu 4 bugs:

#	Gravidade	Arquivo	Descrição
1	CRÍTICO	diamond_scanner.py	Código morto após <code>return</code> em <code>ml_available</code> — os símbolos nunca eram ativados. <code>initialize()</code> não definia <code>_initialized = True</code> .
2	ALTO	stochastic_analytics.py	GARCH: se nenhum par (α, β) válido encontrado, fallback silencioso com parâmetros inventados. Adicionada verificação <code>best_ll <= -1e100</code> .
3	MÉDIO	stochastic_analytics.py	Monte Carlo: o fallback de volatilidade (log-returns) era calculado mas nunca usado quando o sigma OU era zero.
4	BAIXO	stochastic_analytics.py	FTMO: <code>pnl_avg/roi_avg</code> fora do bloco <code>try</code> — risco de <code>NameError</code> se código adicionado entre cálculo e <code>return</code> .

15. Matriz de Decisão Final

A matriz de decisão combina pontuação Diamond, ML%, guardrails e contexto macro:

Condição	Ação	Justificativa
ML% $\geq$ 60% + TKx HI alinhado + sem HIGH NEWS	☐ GO	Tripla confirmação: ML, Ichimoku, macro
ML% $\geq$ 60% + TKx HI conflito	☐ AGUARDAR	O ML é contradito pelo cruzamento T/K HI
ML% 50-60% + TKx HI alinhado	☐ CUIDADO	Vantagem ML moderada, necessita confirmação adicional
ML% < 50%	☐ NO GO	O modelo não tem vantagem estatística
HIGH NEWS DAY + BULL	☐ AGUARDAR	Dias macro (≥ 2 eventos maiores) quebram correlações
Quarta-feira (dia FOMC) + Forex	☐ Risco Alto	Quartas-feiras mostram comportamento estatisticamente distinto

16. Conclusão e Perspectivas

O projeto InelidaMarketScan demonstra que um pipeline ML treinado com features Ichimoku pode gerar uma vantagem estatística significativa (PF OOS = 1,173). Os 4 bugs críticos corrigidos em 19/07/2026 reforçam a confiabilidade do scanner ao vivo. Os 27 modelos sucessivos revelaram insights-chave: as features Ichimoku são mais preditivas que os padrões ICT, o tipo de ativo (is_dxy, is_forex) é um preditor principal, e as funções de cálculo estocástico fornecem sinal complementar de volatilidade. O próximo salto de desempenho exigirá um dataset maior (2024-2026, >5.000 trades/ativo) e potencialmente arquiteturas de deep learning.

Referências

- 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. ASCE, 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

Obra de pesquisa em finanças quantitativas. Reprodução permitida com citação.

☐ Comunidade

Junte-se ao servidor Discord **Trading Pro** para se conectar com a comunidade:
<https://discord.gg/u9JApXqhXy>