

Every Headline Number, and How to Reproduce It

The research figures across all six projects, each with its source file and the command that regenerates it. Dated, because data moves.

Daniel Lichtenberger · October 2026
danny-397.github.io · [Source on GitHub](#)

Every headline number on this site, where it comes from, and the command that reproduces it. Figures are dated because data sources and package versions move; a rerun on a later date can legitimately differ, and the source files say how. Test counts are not listed here: they change with every commit, so the site reads them from each repository's CI instead.

Compiled 2026-10-28.

How RL agents fool you (RL-Trader)

Claim	Figure	Source	Reproduce
A single favourable run was not an edge	crypto agent +56.8% [+16.3%, +106.3%] across 10 seeds vs buy-and-hold +33.5%, permutation p = 0.79	RESULTS.md §5	<code>python tools/real_significance.py --seeds 10 --timesteps 150000</code>
The stock agent is reliably worse than holding	-19.1% [-26.7%, -11.2%] vs +239.5%, p = 0.0021	RESULTS.md §5	same command
Memorisation, and the fix	held-out return single-asset → across tickers: stocks -25% → +31% , crypto -44% → +187% (10 seeds, all four intervals exclude zero)	RESULTS.md §1	<code>python tools/ablation_multiseed.py --seeds 42 43 44 45 46 47 48 49 50 51 --timesteps 60000</code>
Pre-registered replication	predictions committed (commit <code>d344b4e</code>) before running seeds 200–209; all four held (stock p = 0.0021, crypto p = 0.88). Not replicated: the crypto mean's interval now includes zero, +26.7% [-13.8%, +70.4%]	docs/preregistration/	command in the registration file; input data archived by SHA-256

QuantumSafe Scan

Claim	Figure	Source	Reproduce
Precision on the project's labeled corpus	100% precision and recall on 26 labeled findings (18 files, 9 languages); naive line-regex 49.1%	benchmark/RESULTS.md	<code>python benchmark/evaluate.py</code>
Recall under mutation	50/50 seeded cases found, 0 false positives on 50 negative mutations	benchmark/RESULTS-seeded.md	<code>python benchmark/seeded.py</code>
Precision on real code nobody chose	93% (95% CI 86–97%) on 100 random findings from 37 PyPI packages, each hand-labeled with a reason; measured before the TLS rule fix it exposed	benchmark/RESULTS-precision.md	<code>python benchmark/precision_sample.py --score</code>
Scale	5,609 findings (4,236 HIGH) across 11,275 files in 37 packages, 2026-10-28, scanner 0.3.1	benchmark/RESULTS-realworld.md	<code>python benchmark/realworld.py</code>
Versus general security tools	quantum-relevant findings caught on the labeled corpus: QuantumSafe 26/26, Semgrep (<code>p/r2c-security-audit</code>) 4/26, Bandit 2/26	benchmark/COMPARISON.md	<code>python benchmark/comparison.py</code>

AlphaGlyph

Claim	Figure	Source	Reproduce
The best backtest in a search is probably overfit	probability of backtest overfitting 72% over a 42-variant grid on SPY (CSCV); median out-of-sample Sharpe 0.65, ~52% of in-sample	technical report §7	<code>python tools/pbo_snapshot.py</code> (pinned SPY snapshot, SHA-256 recorded)
The ML forecaster cannot predict direction	test AUC $\approx$ 0.51	technical report §7	model metadata, <code>ml_model_meta.json</code>
One conditional positive, and how fragile it is	Dip Buyer on AAPL/MSFT, 2021-01-01 → 2024-01-01: +21.6% vs SPY +5.6% in ranging markets; one more year shrinks the edge to +3.6 points	README, technical report §7	backtester with those dates; <code>regime_conditional</code> in the result

AlphaZero-style chess engine

Claim	Figure	Source	Reproduce
The plateau was a training-signal problem, not compute	Elo vs a random player: baseline -21 , 10× self-play -7 , material signal +28	docs/experiment-results.md	<code>python run_experiment.py --device cuda</code> (9.8 h on 2× T4)
Strength of the classical control engine	Easy (depth 2) $\approx$ 1390 [1284, 1512] on Stockfish's engine scale: +14 =8 -8 vs Stockfish 19 at UCI_Elo 1320	docs/data/stockfish_rating_d2_e1320.json	<code>python tools/stockfish_rating.py --stockfish <path> --depth 2 --games 30</code>

Tradeski and Neural Canvas

These two make engineering claims rather than research ones: Tradeski's indicators are hand-coded on NumPy and covered by its CI (Python and JavaScript suites); Neural Canvas runs MediaPipe's 21-landmark hand model on-device at roughly 60 frames a second. Both are live and can be checked directly.

Accessibility

All seven sites report **zero** violations under axe-core 4.10 with the WCAG 2.1 A/AA and best-practice rule sets (main pages and views, 2026-10-28).