Pre-registered, refuted, kept

How we test our own claims — including the one this page is about

Every vendor in the AI market publishes its wins. Nearly all such claims are self-attested: the company grading its own homework, after the fact. You cannot diligence a retrospective claim directly. What you *can* diligence is a **prospective method** — predictions written down before the experiment runs, under rules that make a good result impossible to manufacture and a bad one impossible to bury.

Here is ours, from one research cycle in August 2026, on our own core thesis (that a governed "operations floor" under an AI agent improves its work). Every item below is in a permanent record available for inspection in a diligence session.

1. We wrote the predictions before any run — under a no-edit rule. The spec pre-registered five predictions, including a falsifiable point prediction for how much our un-generalized floor would help, and a failure-shape prediction stating explicitly that one class of failure — wrong values, rather than missing coverage — would *"outrank the score"* if it appeared. The file's own rule: *"do not edit predictions after results exist; append outcomes."*

2. The method was overfit-proofed by construction, including against ourselves. Schema-derived views only; no task-conditioned logic; and an excluded-knowledge clause barring any ground-truth values our own team had computed while contributing to the upstream benchmark — a contamination we were positioned to commit and chose to make impossible instead.

3. The first result refuted our own point prediction — and we appended it to the same file. We predicted the un-generalized floor would score 75–88%. It scored **36%, against a 43% baseline** — below the thing it was supposed to beat. The pre-registered failure clause materialized exactly as written: a wrong-value correctness cluster that, by our own rule, mattered more than the score. The predictions were not edited. The refutation sits in the record above the recovery.

4. The refutation, not the ambition, produced the fix. The failure analysis pointed at one structural gap (a data-grain mismatch our views inherited). Correcting it under the same rules, later runs improved at both tiers — a cheap-model arm from 27% to 53% against its own baseline, and a frontier-model arm from a 71% baseline to 86% with the floor, at 0.6× the baseline's cost.

Method caveats, stated because they are the point: frontier-tier results are n=1; scoring used our own judge configuration, so **no figure here is leaderboard-comparable and none is published as a benchmark claim** — the valid quantity is the floor-versus-baseline delta under identical conditions, and the valid artifact is the method. **We have made no public leaderboard submission.**

Why this matters to you: a prediction registered before the run is expensive to fake. A refutation kept in the same file cannot be backfilled. That is the same property we sell in operations — a governed system whose work records its own verdicts as it runs — applied to our own research. In a market of self-attested wins, a kept refutation is the only kind of claim that arrives with its own evidence.

Companion pages: "How We're Wrong on Purpose" (jiegou.ai/downloads/wrong-on-purpose-v1.pdf) and the Diligence Trust Brief (jiegou.ai/security).

Current as of 24 August 2026.