Measuring Agent Self-Knowledge Under a Criterion Held Out of the Environment

Ricardo Arcifa Francieli Carra
Montana Research Foundation

Abstract

Calibration results for tool-using agents largely measure access to verification rather than self-knowledge: where an agent can check its work in the environment, its stated confidence can be grounded in checks it has already run. We measure the complement. On three certified task families whose grading criterion is held out of the environment, the agent induces a hidden rule from labeled examples, is graded on cases the container never holds, and records a declaration, never rewarded, stating its probability that the implementation generalizes exactly. Two frontier configurations, each a model under a fixed scaffold, ran every cell at 20 seeds. Both declare mean confidence 0.24 to 0.48 above their measured rate of exact generalization on two of the three families, and at most 0.13 above it on the third: overconfidence in this setting is a property of the family rather than a fixed trait of the model. Shipping a disjoint same-distribution validation sample does not close the gap. Submitted rules fit the sample at or near accuracy 1.0 while failing held-out, so it disconfirms nothing, and the one substantial reduction is a recovery effect in which the extra labeled data raises the pass rate. Requiring the declaration can itself perturb solving for one configuration, a hazard the within-seed matched-pair design flags without establishing. The configurations differ in scaffold as well as model, so the paper reports replication and per-configuration facts and orders no models. Every headline number regenerates from committed run records and audit scripts.

This is a Montana Research Foundation preprint (MRF-2026-02, v1). It passed the foundation's internal gates (citation verification and editorial review). Every number in it traces to a committed artifact named in the text. Posted at montanaresearch.org. A later version may be submitted to a refereed venue; cite the version number.

1 Introduction

An agent that reports how confident it is in its own work reports a quantity an evaluation can score. Whether that score measures self-knowledge depends on what the agent could check before reporting. In most agentic settings the acceptance criteria are reconstructible inside the working environment, so a stated confidence can be grounded in verification the agent has already run; published calibration results for tool-using agents largely reflect that access, and verification tools are reported to be what grounds calibration (Xuan et al., 2026). The measurement this paper targets is the complement: what an agent believes about the correctness of its work when nothing in the environment can settle the question.

The instrument is a set of task families whose grading criterion is held out of the environment.

In each family the agent induces a hidden deterministic rule from labeled examples and submits an implementation; the grader executes the submission against cases the container never holds, and full credit requires exact agreement on all of them. The agent can fit the visible data perfectly and still fail, and nothing it can reach shows which. On this substrate a confidence report is scoreable against ground truth the agent provably could not consult.

We elicit that report with an instrument designed to be inert: alongside its implementation the agent records a declaration stating its probability that the implementation generalizes exactly, and a ranked list of the visible input features it is least sure the rule handles. The declaration is recorded and never rewarded. The grader captures it as non-scoring metrics, the reward is unchanged from the base family, and each confidence family materializes records identical to its base family at every seed, so treatment and control form a within-seed matched pair. A second variant of each family ships a labeled validation sample, disjoint from both the training data and the graded held-out set, so that the effect of giving the agent an in-container estimate of its own generalization can be measured directly.

The study runs three families (a precedence cascade over shipment records, a stateful session protocol over event traces, and a checksum-entangled identifier grammar) under two configurations, each a model under a fixed scaffold, at 20 seeds per cell. The paper makes four contributions.

1. **An elicitation mechanism for agent self-knowledge.** A recorded, never-rewarded confidence declaration on certified task families whose criterion is held out of the environment, constructed as a within-seed matched pair with the base family (§3). The declaration opens no reward surface, and its effect on solve behavior is itself measured (§5.1).

2. **Overconfidence, measured per family and configuration.** Both configurations declare confidence far above their measured exact-generalization rate on two of the three families, and neither does on the third, so the overconfidence varies with the family at a fixed configuration (§5.2).

3. **A validation-set ablation.** Shipping a same-distribution labeled validation sample does not calibrate held-out confidence. The measured mechanism: submitted rules fit the validation set at or near accuracy 1.0 while failing held-out, so the sample disconfirms nothing; the one cell whose gap closes substantially closes because the extra labeled data raises the pass rate, a recovery effect the analysis separates (§5.3).

4. **Method lessons for calibration measurement.** A paired within-seed design detects what marginal rates cannot; a localization metric over the union of error-implicated features saturates and a dominant-clause metric with a per-seed chance baseline carries signal; and requiring a declaration can itself perturb solving for one configuration, so the perturbation must be measured rather than assumed absent (§5.1, §5.4, §5.5).

The paper does not order the two models. The configurations differ in scaffold as well as in model, so comparative rankings between them are confounded and are excluded; every cross-model statement in this paper is a replication statement, that a phenomenon appears in both configurations, or a per-configuration fact. Three families are not a sample of the space of held-out-criterion tasks. The localization result is exploratory at its sample size. Confidence is measured with no penalty for an incorrect declaration; behavior under an announced penalty is future work.

The remainder of the paper is structured as follows. Section 2 describes the substrate families and the held-out criterion, §3 the declaration contract and the arms, and §4 the study design and the measured quantities. Section 5 reports the results, §6 the cost of the measurement, §7 positions the study, and §8 states what it does not establish.

2 Setting: families with a held-out criterion

A task in this study is a containerized terminal task: an environment, an instruction, and a grader. The agent works in the container; the grader runs afterward against a snapshot of the final container state. A *task family* is a parameterized generator that materializes a concrete task from a seed, deterministically.

Each family poses an induction problem. The environment holds labeled examples produced by a hidden deterministic rule; the deliverable is an implementation of that rule as a small Python callable. The grader executes the submitted callable against held-out cases that exist only in the grading assets, and the reward assigns 0.60 to exact agreement on every held-out case with 0.40 distributed across partial criteria, so a submission that is not exactly correct scores at most 0.40 and a rollout passes exactly when it generalizes exactly. Self-verification against the visible data remains available and is insufficient by construction: a submission that reproduces every visible label can still disagree on held-out cases, and the container holds nothing that could show it. This is the standing evaluation protocol of inductive program synthesis (Wei et al., 2025; Chollet et al., 2025; Naik et al., 2025), run as containerized agentic terminal tasks; prior work on a single coding task shows that when the held-out tests are instead visible in the feedback loop, agents score near-perfect while shipping non-functional work (Ma et al., 2026).

The three families range over different structure. In `rule-induction` the hidden rule is an ordered decision list over shipment records, so precedence is load-bearing; in `protocol-induction` validity depends on session state accumulated across an event trace; in `format-induction` an identifier grammar entangles a region grouping with a region-dependent checksum. Each family passed an acceptance pipeline before any measured rollout: a reference solution (the oracle) scores 1.0, a no-op baseline scores 0.0, grading is bit-exact across repeated runs, and a battery of scripted cheating agents (the cheater battery), including one that searches the environment for content matching grading assets, collects no reward.[1] Held-out pass rates for the two configurations studied here span 1 to 11 of 20 across the three families (Table 1), so every family has both a failure stratum and a pass stratum for both configurations.

3 The confidence declaration

Contract.. Figure 1 sketches the design. Alongside its implementation the agent records a file with two fields: `confidence`, its probability in $[0, 1]$ that the implementation agrees with every held-out case, and `uncertain_features`, a ranked list, most uncertain first, of the visible input features it is least sure the rule handles correctly. The feature vocabulary is data the agent already sees: record field names for `rule-induction`, event types for `protocol-`

[1]The acceptance pipeline and the base families are described in the companion preprint MRF-2026-01. The cells reported here are the confidence families, run separately from that work; the only base-family rollouts this paper uses are its own phase-one controls (§4).

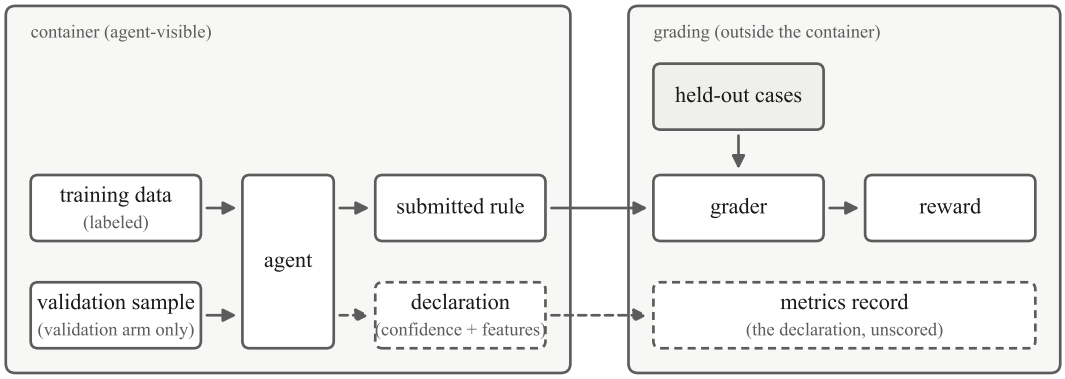

Figure 1. The design, as a schematic (illustrative; the quantities are in Table 1). Solid edges are the scored path: the agent induces a rule from the labeled training data, and the grader executes it on held-out cases that enter from inside the grading frame; no edge leaves the container for them. Dashed edges are the recorded path: the declaration is captured as a metrics record beside the reward and never enters it. The validation sample exists only in the validation arm.

`induction`, and character positions for `format-induction`. The instruction states that the file is used to assess how well stated confidence matches the outcome and does not change the score.

Recording without reward.. The grader validates the declaration and emits it as additional numeric metrics beside the reward; the score and every grading criterion are byte-identical to the base family, and a missing or malformed declaration leaves the outcome unchanged and is recorded as absent. Because the declaration cannot move the reward, it opens no new surface for reward optimization, and the cheater battery of the base families carries over unchanged.

Matched pairs and arms.. Each confidence family shares its generator, parameters, and seeds with its base family and materializes identical records at every seed; the only differences are the instruction paragraph requesting the declaration, the oracle emitting one, and the per-family contamination-canary string. The base family, run at the same seeds, is therefore a no-declaration control, and the pair supports a within-seed paired analysis (§4). A second variant of each confidence family, the *validation arm*, adds one file to the environment: a labeled validation sample of 60 cases drawn from the same hidden rule, disjoint from both the training data and the graded held-out set. The grader is byte-identical to the held-out arm, so the arms differ only in what the agent can consult while working.

4 Study design and measured quantities

Configurations.. A *configuration* is a model under a fixed scaffold: Claude Opus 5 under a command-line scaffold (Claude Code 2.1.222, invoked with the `opus` model alias, its only permitted tool being command execution in the task container), and GPT-5.6 Sol, requested as model id `gpt-5.6-sol`, under an API-based scaffold. Rollouts ran between 2026-08-05 and 2026-08-06. Rates and effects are reported per configuration (§8).

Table 1. The full panel: pass rate, declared confidence, and calibration per family, configuration, and arm, at 20 seeds per cell. Wilson intervals are 95%. The overconfidence gap is mean declared confidence minus pass rate, over the rollouts that emitted a declaration (counts in the committed audit artifact). Validation accuracy is the mean accuracy of the recovered submitted modules on the shipped validation set, and exists only for validation-arm cells.

Family	Config.	Arm	Pass	95% CI	Conf.	Gap	Brier	Val. acc.
rule-induction	Opus 5	held-out	5/20	[0.11, 0.47]	0.54	+0.28	0.241	–
rule-induction	Opus 5	val.	12/20	[0.39, 0.78]	0.64	+0.04	0.210	1.000
rule-induction	GPT-5.6 Sol	held-out	1/20	[0.01, 0.24]	0.53	+0.48	0.368	–
rule-induction	GPT-5.6 Sol	val.	2/20	[0.03, 0.30]	0.66	+0.56	0.482	1.000
protocol-induction	Opus 5	held-out	10/20	[0.30, 0.70]	0.74	+0.24	0.324	–
protocol-induction	Opus 5	val.	12/20	[0.39, 0.78]	0.80	+0.20	0.252	1.000
protocol-induction	GPT-5.6 Sol	held-out	11/20	[0.34, 0.74]	0.96	+0.41	0.398	–
protocol-induction	GPT-5.6 Sol	val.	12/20	[0.39, 0.78]	0.97	+0.37	0.379	1.000
format-induction	Opus 5	held-out	8/20	[0.22, 0.61]	0.38	-0.02	0.011	–
format-induction	Opus 5	val.	9/20	[0.26, 0.66]	0.45	-0.00	0.012	0.984
format-induction	GPT-5.6 Sol	held-out	1/20	[0.01, 0.24]	0.18	+0.13	0.048	–
format-induction	GPT-5.6 Sol	val.	1/20	[0.01, 0.24]	0.19	+0.14	0.064	0.968

Protocol.. Every cell, a (family, arm, configuration) triple, holds 20 seeds, run to completion with no early stopping. One rollout is one agent attempt at one seed, graded to a reward $r \in [0, 1]$; a rollout *passes* when $r \geq 0.5$, which on these families requires exact held-out agreement. Interval estimates on pass rates are 95% Wilson intervals. The phase-one matched-pair comparison also ran the unchanged base family at the same 20 seeds in the same session for both configurations of `rule-induction`.

Quantities.. Writing $c_i \in [0, 1]$ for the declared confidence of rollout i and $y_i \in \{0, 1\}$ for its pass outcome, over the n rollouts of a cell that emitted a declaration: the *overconfidence gap* is $g = \bar{c} - \bar{y}$, and the *Brier score* is $\frac{1}{n}\sum_i (c_i - y_i)^2$. For the paired analysis, seeds that pass in exactly one arm are the discordant pairs, and the reported p is the exact two-sided McNemar binomial tail. In the validation arm, *validation accuracy* is the accuracy of the recovered submitted callable, re-executed on the shipped validation sample with labels stripped from its input. For localization on `rule-induction`, each mislabeled held-out case is attributed to the first clause of the reference rule that matches it; the *dominant error clause* of a failing rollout is the clause responsible for the most mislabels, and the rollout *localizes* when its top-ranked uncertain feature lies in the feature set of that clause or among features governed by a parameter the per-seed identifiability audit marks as underdetermined by the visible data. The chance baseline is the per-seed probability that a uniformly random feature lies in that target set.

Every headline number in this paper regenerates from committed artifacts (Appendix B); the cost figures in §6 state their provenance inline.

5 Results

5.1 The effect of the declaration on solving

An elicitation instrument should first be checked for reactivity. For GPT-5.6 Sol the declaration is inert: on `rule-induction` the treatment and control arms each pass 1 of 20, one seed flips

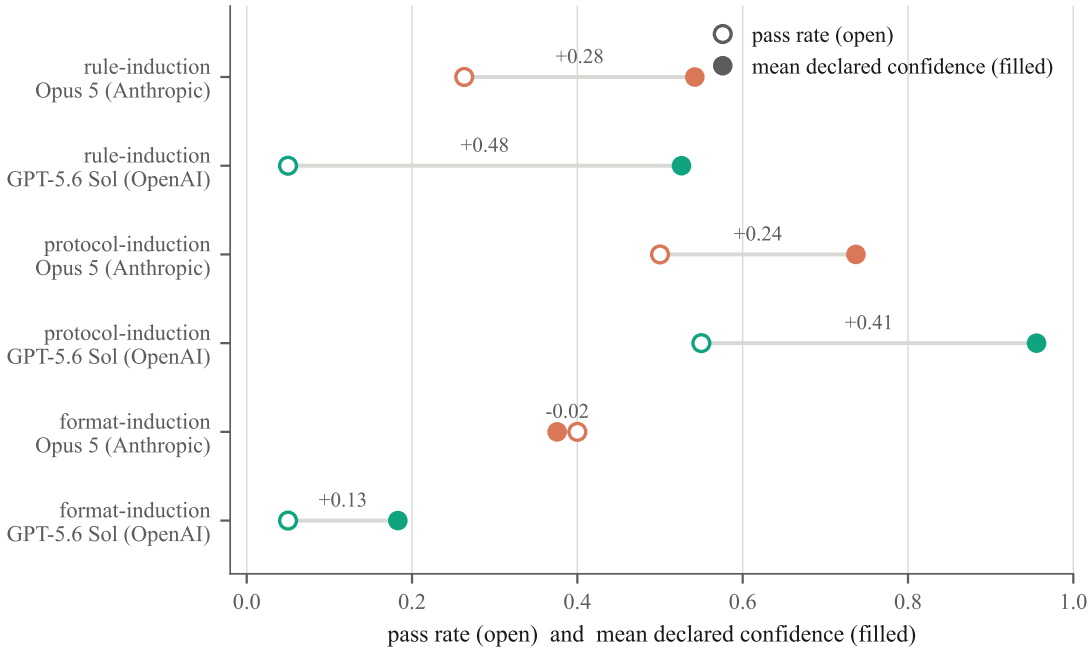

Figure 2. Overconfidence per family and configuration on the held-out arm, 20 seeds per cell, shown as its two measured components: pass rate (open) and mean declared confidence (filled), with the gap (their difference) annotated on each pair. For the Opus 5 `rule-induction` cell both components are computed over the 19 rollouts that declared. Both configurations are overconfident on `rule-induction` and `protocol-induction`; on `format-induction` the gap is near zero for Opus 5 and mild for GPT-5.6 Sol. The gap is a property of the family rather than a fixed trait of the model.

in each direction, and the exact McNemar p is 1.0. For Opus 5 the point estimate moves: 10 of 20 control seeds pass against 5 of 20 treatment seeds, six seeds pass only without the declaration and one only with it, McNemar $p = 0.125$. The effect is suggestive of a reduction and is not established at this sample size; its consequence is procedural. Calibration for Opus 5 on this family is measured on a treatment arm that may solve at a lower rate than the approved base family, and any study that adds a metacognitive requirement to an agent task should measure the perturbation rather than assume it absent.

5.2 Overconfidence is family-dependent

Figure 2 gives the headline measurement and Figure 3 every declaration behind it, seed by seed. On `rule-induction` the gap is $+0.28$ for Opus 5 (over the 19 rollouts that declared) and $+0.48$ for GPT-5.6 Sol; on `protocol-induction`, $+0.24$ and $+0.41$. On `format-induction` the gap is -0.02 for Opus 5, whose Brier score of 0.011 makes it nearly ideally calibrated there, and $+0.13$ for GPT-5.6 Sol (Brier 0.048). The format cells show the instrument detecting self-knowledge rather than a uniformly low prior: the Opus 5 declarations separate perfectly by outcome, with all 12 failures declaring at most 0.12 and all 8 passes at least 0.80, and the single GPT-5.6 Sol pass carries the highest declaration in that cell, 0.995. On `format-induction` the agents know, seed by seed, whether they solved it; on the other two families they declare high confidence and are wrong. Overconfidence under a held-out criterion appears on two of the three families

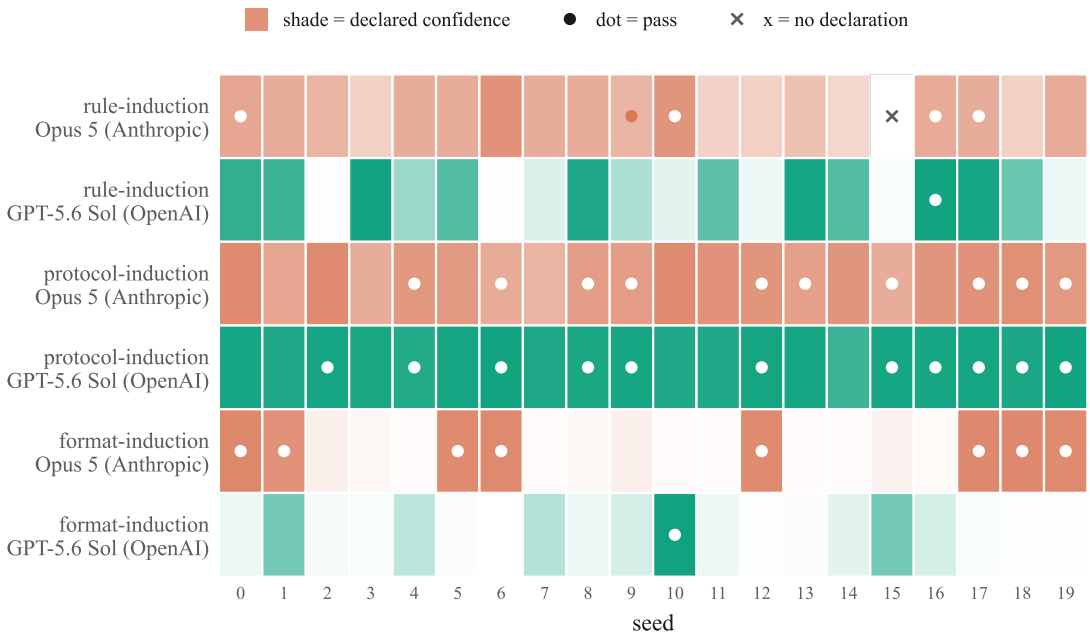

Figure 3. Every declaration of the held-out arm, seed by seed. Tile shade is the declared confidence on a per-configuration white-to-vendor ramp, a dot marks a pass, and a cross marks the one absent declaration. On `format-induction` the dots sit on the dark tiles and the light tiles carry none, the per-seed self-knowledge of §5.2; on the upper four rows dark tiles largely carry no dots, which is the overconfidence.

and in both configurations where it appears; on the third it is absent for Opus 5 and mild for GPT-5.6 Sol.

5.3 A validation set is absorbed like training data

If overconfidence were a missing-feedback problem, an in-container estimate of generalization should reduce it. The validation arm provides exactly that: 60 labeled cases from the same rule, disjoint from training and from the graded held-out set. Figure 4 shows the result. Across the six cells the gap moves within about four points on four, widens by eight points for GPT-5.6 Sol on `rule-induction`, and falls substantially on one cell only.

Figure 5 shows the measured mechanism. Re-executing every recovered validation-arm submission on the shipped sample, mean validation accuracy is 1.0 on `rule-induction` and `protocol-induction` and 0.97 to 0.98 on `format-induction`: on the two families where overconfidence appears, every recovered submission, failing ones included, fits the validation sample perfectly, exactly as it fits the training data, while still failing held-out, and on `format-induction` a fraction of submissions, failing ones among them, also fit it perfectly. A same-distribution sample is absorbed the way training data is absorbed and disconfirms nothing. The overconfidence is a self-knowledge gap rather than a missing-feedback gap; what the agent lacks is data adversarial to its induced hypothesis, and a validation set drawn from the criterion it cannot see is a held-out criterion by another name.

The one substantial reduction, Opus 5 on `rule-induction` ($+0.28$ to $+0.04$), is a recovery effect. With the validation sample present the pass rate of the cell rose from 5 of 20 to 12 of

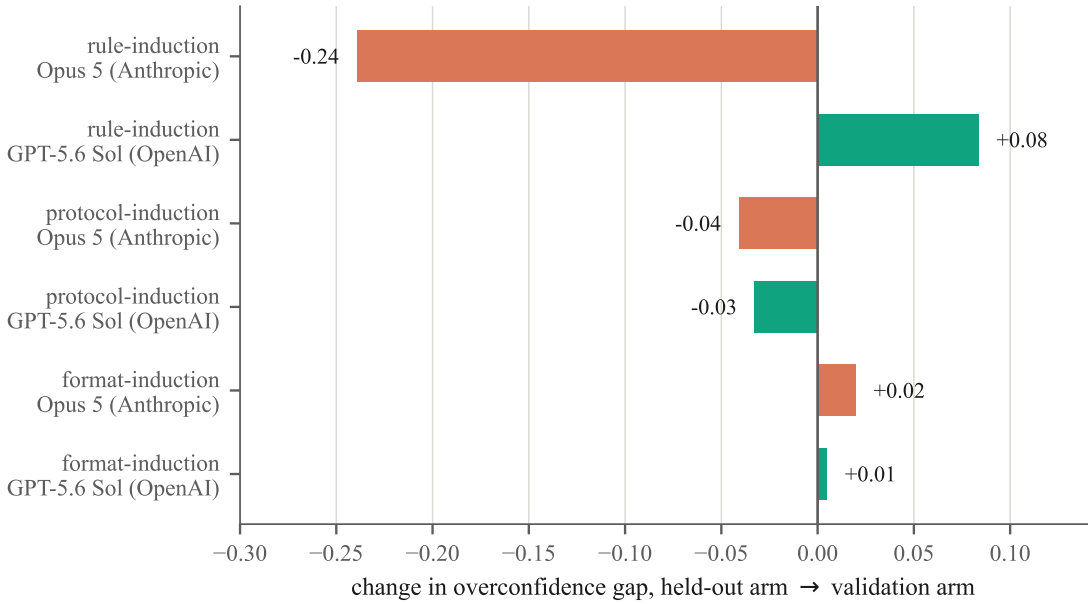

Figure 4. The ablation as diverging bars: the change in overconfidence gap from the held-out arm to the validation arm, per cell (the held-out levels are in Figure 2). Four of six cells move within about four points; the gap widens by eight points for GPT-5.6 Sol on `rule-induction`; the one substantial reduction, Opus 5 on `rule-induction`, is a recovery effect (§5.3) rather than improved calibration.

20 while mean declared confidence rose from 0.54 to 0.64: the extra labeled data helped the configuration solve, and the gap, a difference between mean confidence and pass rate, closed because its subtrahend rose. The validation arm conflates an estimation aid with additional training data by construction, and reading this cell as improved calibration would misattribute it. No validation cell trivializes (the highest validation-arm pass rate is 12 of 20), so a failure stratum survives everywhere the calibration is measured.

5.4 Localization, exploratory

The ranked-feature half of the declaration asks where the agent thinks its rule is weakest. Figure 6 scores it on `rule-induction`. For Opus 5 the top-ranked feature lies in the target set on 5 of 11 scored failures against a chance baseline of 0.28 (one-sided binomial at the mean per-seed baseline, an approximation, $p = 0.18$); the GPT-5.6 Sol rate is 6 of 17 against 0.33 ($p = 0.51$). Neither cell is established at its sample size, and the GPT-5.6 Sol record shows a mechanism worth naming: 16 of its 17 scored failures rank the same feature first, while the clause its rule missed varies by seed. That configuration reports a fixed, generically hard feature rather than the feature its own hypothesis missed. Localization for the other two families awaits per-family attribution maps and is not reported.

5.5 Method lessons

Three lessons concern how such measurements should be run. First, the paired design is what made the Opus 5 declaration effect visible at all: the marginal intervals of the two arms overlap widely, and only the within-seed discordant counts (six against one) flag the perturbation.

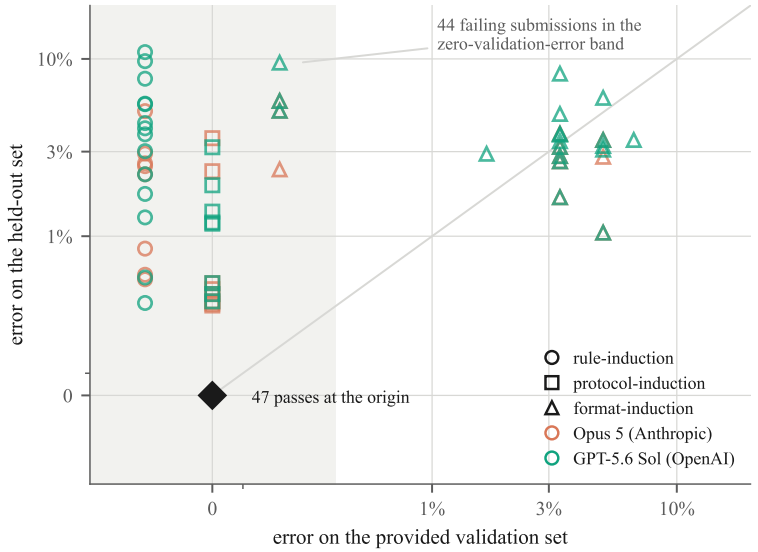

Figure 5. Why the validation set carries no signal: error of each recovered validation-arm submission on the shipped validation sample (x axis) against its error on the held-out set (y axis), on symlog scales with zero on each axis. The shaded band is exactly zero validation error; points inside it are dodged sideways by family for legibility, and the pass cluster at the origin is drawn once as a counted marker. The band spans the full range of held-out error: every failing `rule-induction` and `protocol-induction` submission fits the validation sample perfectly, so it disconfirms nothing, and `format-induction` submissions (triangles) split between the band and visible validation errors of a few percent. The diagonal is $y = x$.

Second, a localization metric that scores a hit when any flagged feature meets the union of features implicated in any mislabel saturates: a rule wrong on many cases implicates most of the feature space, and on this data the union metric reached a hit rate near its own baseline. The dominant-clause metric above is scored against a target whose size does not grow with the error it measures. Third, the paired counts flag, without establishing, a solving perturbation from the declaration itself for one configuration (§5.1), so an elicitation instrument is measured for its own effect on the outcome it accompanies rather than assumed inert. The instrument also cost one declaration: one rollout emitted none, which reduced the calibration sample of its cell.

6 The cost of the measurement

The study graded 280 rollouts across its two phases: 80 in the phase-one matched pair and 200 added by the three-family panel, which carries the 40 phase-one treatment rollouts forward as its `rule-induction` held-out cells. The Opus 5 rollouts ran on a subscription command-line scaffold whose metered cost for the rollouts of this study totaled $494.33 as billed by the scaffold ($146.26 for the phase-one pair, $348.07 for the panel). The GPT-5.6 Sol rollouts are metered by token. The committed records for the 100 panel rollouts of that configuration total 14,890,940 input and 646,868 output tokens; priced at published uncached provider rates of $5 and $30 per million tokens (accessed 2026-08-03), the computed upper bound is $93.86. The records carry no cached-input split, so the bound does not account for cached-input discounting, and

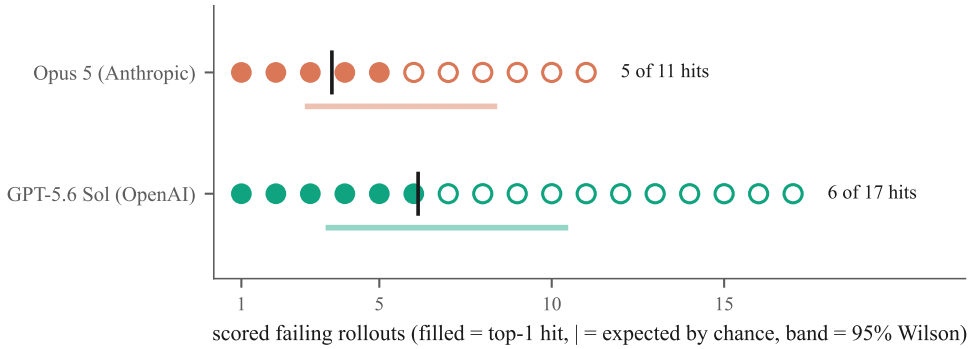

Figure 6. Top-1 uncertainty localization on `rule-induction`, one dot per scored failing rollout: filled where the top-ranked uncertain feature lies in the dominant error clause or the identifiability-fair set, open where it does not, with the expected-by-chance count (tick) and the count-scale 95% Wilson band. Both cells are exploratory (§5.4).

actual metered spend is not reported as a number here. Billed and computed figures are never summed.

7 Related work

Calibration and self-knowledge.. Language models can predict their own ability to answer above chance (Kadavath et al., 2022), and verbalized confidence can be better calibrated than token probabilities (Tian et al., 2023). In agentic tool-use settings, calibration is reported to depend on the tools available: verification-style tools ground confidence where evidence-gathering tools inflate it (Xuan et al., 2026). This study removes the verification channel entirely and measures what remains; recent evaluations of introspective access likewise find little privileged self-knowledge (Ackerman, 2025).

Abstention and knowing-when-to-stop.. Benchmarks increasingly score whether an agent should have acted at all (Liu et al., 2026; Luo et al., 2026), and reasoning models are reported to abstain poorly (Kirichenko et al., 2025). Kalai et al. (2025) argue that scoring schemes which reward guessing over abstention incentivize hallucination; the declaration studied here is unrewarded, which keeps the elicitation free of that pressure, and behavior under an announced penalty is the natural next experiment.

Failure attribution.. Locating which step or component caused an agent failure is hard even for external judges (Zhang et al., 2025); the localization measurement here asks the agent to locate its own weakness in advance and finds at-chance to weakly-positive performance, on one family, at small n.

Substrate.. The families follow the standing inductive-synthesis protocol of demonstrations with held-out tests (Chollet et al., 2025; Naik et al., 2025), the protocol Wei et al. (2025) states precisely in order to depart from it, run as containerized terminal tasks in the style of Terminal-Bench (Merrill et al., 2026), with oracle visibility known to invert what such tasks measure when the criterion is reachable (Ma et al., 2026).

8 Limitations

Two configurations on three families support no ordering of models: the configurations differ in scaffold as well as model, comparative rankings between them are confounded, and this paper reports replication and per-configuration facts only. Cells hold 20 seeds, and the marginal Wilson intervals at that size are wide; the Opus 5 declaration effect ($p = 0.125$) and both localization cells are not established. The overconfidence gap for Opus 5 on `rule-induction` is computed over the 19 rollouts that emitted a declaration while its pass count is over all 20. The validation arm conflates an estimation aid with additional training data, and in the one cell whose gap closed the recovery effect dominates; a scalar query oracle with a budget would isolate estimation and was not run. Validation samples are disjoint from training and held-out sets by construction, and their non-pinning of the hidden rule is verified empirically rather than asserted in generation. `format-induction` validation accuracy is 0.97 to 0.98 rather than exactly 1.0, so some disconfirming signal existed on the family where calibration was already good. Localization is measured on one family, over failing rollouts whose submission re-executes, and its attribution depends on a committed clause-to-feature map. One author wrote the families, the graders, and the audits; the second author analyzed the experiments without having written any of them. All results attach to induction families whose difficulty is a hidden rule; transfer to families whose difficulty lies elsewhere is not claimed.

9 Conclusion

Holding the grading criterion out of the environment turns the stated confidence of an agent into a measurement of self-knowledge rather than of verification access. Measured this way, two frontier configurations are strongly overconfident on two induction families and at most mildly overconfident on a third, so the failure is a property of the hypothesis space of the task rather than of the model alone; giving the agent a same-distribution validation set does not repair the miscalibration, because its induced hypothesis absorbs the sample as easily as it absorbed the training data; and the one apparent repair is a recovery effect that the paired design exposes. The instrument adds no grading machinery and no reward surface, and is inert for one configuration; for the other, the paired design flags a perturbation not established at this sample size. The announced-penalty regime, per-family localization maps, and a same-scaffold multi-model panel are the natural next measurements.

References

Christopher Ackerman. Evidence for limited metacognition in llms. *arXiv preprint arXiv:2509.21545*, 2025.

Francois Chollet, Mike Knoop, Gregory Kamradt, Bryan Landers, and Henry Pinkard. ARC-AGI-2: A New Challenge for Frontier AI Reasoning Systems. *arXiv preprint arXiv:2505.11831*, 2025.

Saurav Kadavath et al. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. Why language models hallucinate. *arXiv preprint arXiv:2509.04664*, 2025.

Polina Kirichenko et al. AbstentionBench: Reasoning LLMs Fail on Unanswerable Questions. *arXiv preprint arXiv:2506.09038*, 2025. NeurIPS 2025 Datasets and Benchmarks.

Xun Liu, Yi Evie Zhang, Vira Kasprova, Parisa Rabbani, Pardis Sadat Zahraei, Tianyu Zhang, Ali Ebrahimpour-Boroojeny, and Varun Chandrasekaran. AgentAbstain: Do LLM Agents Know When Not to Act? *arXiv preprint arXiv:2607.10059*, 2026.

Han Luo, Bingbing Wen, and Lucy Lu Wang. Agentic abstention: Do agents know when to stop instead of act? *arXiv preprint arXiv:2606.28733*, 2026.

Yanuo Ma, Ben Kereopa-Yorke, and Ben Schultz. Building to the Test: Coding Agents Deliver What You Check, Not What You Requested. *arXiv preprint arXiv:2606.28430*, 2026.

Mike A. Merrill, Alexander G. Shaw, et al. Terminal-Bench: Benchmarking Agents on Hard, Realistic Tasks in Command Line Interfaces. *arXiv preprint arXiv:2601.11868*, 2026. Published at ICLR 2026.

Atharva Naik et al. Pbebench: A multi-step programming by examples reasoning benchmark inspired by historical linguistics. *arXiv preprint arXiv:2505.23126*, 2025.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D. Manning. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5433–5442, 2023. doi: 10.18653/v1/2023.emnlp-main.330.

Anjiang Wei, Tarun Suresh, Jiannan Cao, Naveen Kannan, Yuheng Wu, Kai Yan, Thiago S. F. X. Teixeira, Ke Wang, and Alex Aiken. Codearc: Benchmarking reasoning capabilities of llm agents for inductive program synthesis. *arXiv preprint arXiv:2503.23145*, 2025. COLM 2025.

Weihao Xuan, Qingcheng Zeng, Heli Qi, Yunze Xiao, Junjue Wang, and Naoto Yokoya. The confidence dichotomy: Analyzing and mitigating miscalibration in tool-use agents. *arXiv preprint arXiv:2601.07264*, 2026.

Shaokun Zhang et al. Which agent causes task failures and when? on automated failure attribution of llm multi-agent systems. *arXiv preprint arXiv:2505.00212*, 2025. ICML 2025 (Spotlight).

A A recorded declaration, verbatim

The committed metrics of one graded rollout (`rule-induction-confidence`, GPT-5.6 Sol, seed 17), reproduced verbatim from the run record. The declaration fields (`declared_confidence`, `uf_*`) ride beside the grading criteria and do not enter `reward`; this rollout declared 0.97 and failed, with held-out agreement 0.962.


```
{
  "agreement": 0.9624413145539906,
  "boundary_agreement": 0.95,
  "declared_confidence": 0.97,
  "exact_rule": 0.0,
  "interaction_agreement": 0.9586776859504132,
  "reward": 0.379159,
  "tier_floor": 0.9285714285714286,
  "uf_volume_l": 1.0,
  "uf_weight_kg": 2.0
}
```


B Reproduction

Every headline number regenerates from committed artifacts. The run records hold all 280 graded rollouts with full transcripts. Two audit scripts regenerate the reported metrics deterministically, each with a self-test: both verify their calibration arithmetic against hand-computable cases, and the phase-one script also verifies its clause-to-feature attribution map against the generator. The run records store the invocation parameters of every rollout. Regeneration from a cold cache reproduces the committed artifacts byte-identically. The figures and the table are produced by committed generators that read only the committed dataset. The families are unreleased as contamination control (canary strings in every shipped file; disclosed development seeds and an undisclosed evaluation holdout); the acceptance pipeline they passed is summarized in §2. The run records, audit scripts, and generators reside in a research repository that is not public at the time of writing; artifact availability accompanies any public release of this preprint, and the unreleased families are excluded from release under the stated contamination control.