EUREKABENCH: MEASURING AGENTIC ABILITY TO DISCOVER NEW SCIENTIFIC INSIGHTS

Anonymous authors
Paper under double-blind review

ABSTRACT

Isaac Newton discovered the law of gravitation through an iterative process of analyzing *observed data* such as planetary patterns, finding the *mechanisms* by describing them in mathematical equations, and refining his theory against the Moon's orbit, revealing the startling *insight* that the same force governs both falling apples and orbiting planets, and successfully predicted the existence of planet Neptune. As AI agents rapidly advance, whether they can automate such a process has become a major point of contention. Despite their fast growing capability, their intelligence is jagged: they excel at fitting observations through optimization, but struggle to discover mechanisms and insights through experiments. To measure this gap, we introduce **EUREKABENCH**, a cross-domain benchmark that tests AI agents' ability to conduct long-horizon experiments, find the mechanism to explain the observations, and eventually be judged by its derivable scientific insights. EUREKABENCH contains an expert-verified set of 26 long-horizon tasks across neuroscience, computer science, chemistry, astrophysics, geophysics, and plasma physics, with a total of 306 insights that are expected to be learned from the discovered mechanism. Our evaluation framework tests three axes of scientific discovery: agents' ability to follow known scientific constraints, the predictive accuracy of the discovered mechanisms, and whether the mechanisms can build the connection to the scientific insights or inform future explorations. Our results show that current AI agents often overly fixate on predictive accuracy optimization, surpassing human scientists, while falling substantially short in deriving scientific insights.

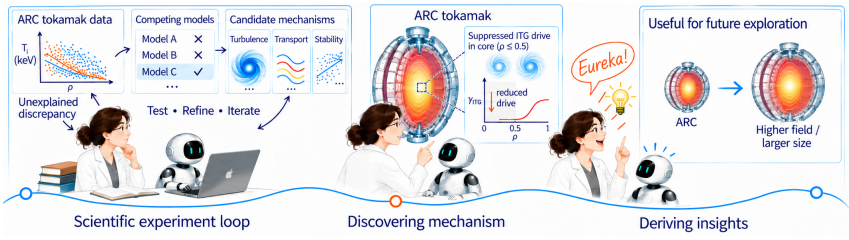

Figure 1: **The scientific discovery process in EUREKABENCH.** The ARC tokamak problem in EUREKABENCH: scientists observe that different transport models predict different fusion power for the same reactor under the same operating conditions *(observation)*. Through iterative experiments, they find that differences in turbulent transport change temperature and density distributions, explaining the different fusion power predictions *(mechanism)*. This helps them understand how this effect may change in larger reactors or under stronger magnetic fields *(insight)*. Discovering such insights is a "eureka moment" in science.

1 INTRODUCTION

Scientific progress occurs when scientists find mechanisms that can describe physical laws in the form of equations, chemical reaction pathways, or models of how neurons coordinate movement to explain observations that challenge our existing understanding of the world (Zahavy, 2026)(Figure 1). A good mechanism not only accurately predict the data observed throughout experiments,

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

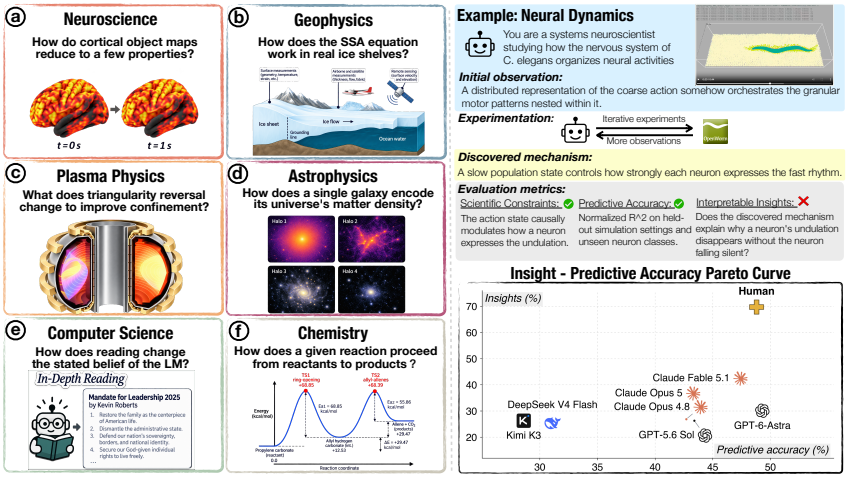

Figure 2: **Initial observations, evaluation, and results in EUREKABENCH.** Examples of initial observations from each domain: (a) neuroscience, (b) geophysics, (c) plasma physics, (d) astrophysics, (e) computer science, and (f) chemistry. An agent investigates how neurons coordinate movement through iterative experiments, and we evaluate its mechanism on three axes: scientific constraints, predictive accuracy, and scientific insights. AI agents can optimize predictive accuracy but struggle to discover novel scientific insights through long-horizon experimentation.

but also yields scientific insights that help us understand a broader range of related phenomena and can even be extrapolated to open new research directions. Recent AI agents have achieved breakthroughs in solving longstanding scientific problems (OpenAI, 2026d; Gottweis et al., 2026). This progress benefits from AI agents' strong search and optimization capabilities especially when objectives are explicit and outcomes are verifiable (Novikov et al., 2025; Lupidi et al., 2026). However, finding correct solutions does not necessarily yield scientific insights that deepens our understanding, and might even hinder their discovery (Tao, 2026). This gap echoes the "severe misalignment"[1] between AI used in science and methods that have traditionally driven scientific progress. To better understand whether AI agents can bridge this gap, we study the question:

"Can AI agents go beyond solution optimization and discover novel scientific insights?"

To answer this question, we first consulted 15 experienced scientists from 10 universities across 8 science domains to identify the stages of research where they would most value and expect assistance from AI agents. This clarifies where AI can best accelerate scientific research. We ultimately collaborated with 10 of these experts to develop EUREKABENCH, a benchmark containing 26 recent and important scientific problems crafted by experts across six domains: neuroscience, geophysics, astrophysics, computer science, plasma physics, and chemistry (see Figures 2 (a) - (f)).

EUREKABENCH is the first benchmark to rigorously operationalize open-ended scientific insight discovery within a systematic evaluation framework, challenging AI agents to start from initial observations and use iterative experimentation to discover mechanisms that are interpretable enough to yield scientific insights. We formulate each task with initial observations that challenge existing understanding. To find the underlying mechanism, we provide an environment interface for agents to autonomously conduct iterative experiments to test it and derive novel scientific insights.

We propose an evaluation framework with three evaluation aspects that span the scientific discovery process: (1) scientific constraint tests, assessing whether the mechanism satisfies the fundamental domain-specific common-sense assumptions; (2) predictive accuracy, measuring how correct the discovered mechanism can make the prediction on the test set; and (3) interpretable scientific insights, evaluating whether the agent discovered mechanism can reveal the interpretable insights. We work closely with the domain experts and formulate the interpretable scientific insight questions by asking *"Does the discovered mechanism explain/establish ...'*, with a "perfect" mechanism expected

[1] https://mathandai.org

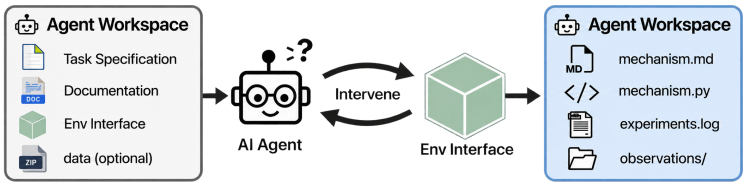

Figure 3: **Scientist agent workflow in EUREKABENCH.** Given a task specification and a documented environment interface, the scientist agent conducts iterative experiments to discover a mechanism and submits its description, implementation, and experimental records.

to completely answer all the questions. We use an agentic judge (Zhuge et al., 2024), which has access to the same environment interface, to evaluate the discovered mechanisms.

We empirically evaluate seven top AI agents on EUREKABENCH, and find that current AI agents can produce accurate mechanisms yet rarely support meaningful scientific insights. For example, GPT 6 Astra approaches the human reference in predictive accuracy (47.4% versus 48.8%), but only achieves a 29.4% insight score, substantially lower than 69.7% for human-discovered mechanisms. This gap shows that optimizing for prediction accuracy by fitting observed data is not enough to obtain mechanisms that deepen scientific understanding.

We investigate why agents struggle to discover novel scientific insights through iterative experimentation. We find that agents tend to treat scientific mechanism discovery as task optimization instead of focusing on the mechanism's interpretability, in contrast to human scientists. Agents show some scientific reasoning ability during experimentation, for example by establishing the mechanisms behind physical parameters. However, their ability to discover insights remains severely limited by their failure to run experiments to reduce uncertainty about the world and by a lack of diverse, in-depth exploration. Despite the strong capabilities that current AI agents have demonstrated, our empirical results on EUREKABENCH reveal that, as AI scientists, they still lack the fundamental ability to discover meaningful insights independently through an experimentation loop. EUREKABENCH reveals the unique openendedness of scientific discovery, in contrast to verifiable programming tasks, and opens a path for the community to develop AI scientists that better support human researchers.

2 EVALUATING A TASK IN EUREKABENCH

2.1 TASK OVERVIEW

A task in EUREKABENCH is defined as $\mathcal{T} = (x, o_1, \mathcal{E}, R)$. The task description x provides the scientific background, instructions, and available computing resources and iteration budgets, while o_1 denotes the initial observations that challenge existing scientific understanding. The environment interface $\mathcal{E}$ allows the scientist agent to conduct the experimentation loop. The reward function R evaluates the scientist agent's deliverables according to the evaluation rubric $\mathcal{C}$ and returns a score.

A time step t in the experimentation loop consists of a query a_t to $\mathcal{E}$ and its returned observation o_{t+1}. We index these time steps by $t = 1, \ldots, T$, where T is the total number of steps. We define the experiment trajectory as $\tau = (o_1, a_1, o_2, \ldots, a_T, o_{T+1})$. The discovered mechanism M includes its explanation and executable implementation, and $\mathcal{L}$ denotes the experiment log. We define $\mathcal{I}$ as the scientific insights that can be derived from M, where $(M, \mathcal{L}, \tau) \longrightarrow \mathcal{I}$. The complete task proceeds from iterative experimentation to evaluation in EUREKABENCH:

$$\mathcal{T} \xrightarrow{\text{iterative experimentation}} (M, \mathcal{L}, \tau) \xrightarrow{\text{evaluation}} R(M, \mathcal{L}, \tau; \mathcal{E}, \mathcal{C}).$$

During the experimentation loop, the scientist agent chooses a_t based on all observations collected so far, $o_1, \ldots, o_t$, and queries $\mathcal{E}$ to obtain o_{t+1}. At the end of the loop, the scientist agent needs to submit M, $\mathcal{L}$, and τ.

In the evaluation stage, we calculate the score $R(M, \mathcal{L}, \tau; \mathcal{E}, \mathcal{C})$ by executing the implementation of M and using a judge agent to evaluate M, $\mathcal{L}$, and τ.

Agent workspace initialization. We initialize the scientist agent's workspace with a task specification, a command-line interface (CLI) and documentation for the environment interface $\mathcal{E}$. The

task specification contains x, o_1, and the name of $\mathcal{E}$. In the task specification, we ask the scientist agent to find a mechanism M that explains o_1. We also require the scientist agent to explore scientific insights $\mathcal{I}$ through experimentation and explicitly state that another scientist will evaluate these insights alongside predictive accuracy. For some problems, we include training data in the scientist agent's workspace. We provide an example task specification in Appendix A.

Interacting with the environment. To find the mechanism M, the scientist agent runs experiments by intervening through the environment interface $\mathcal{E}$ and collecting more experimental trajectories τ to revise its hypotheses, as shown in Figure 3. Such an iterative cycle of intervention, observation, and hypothesis revision constitute reverse engineering (Geng et al., 2025a).

Compiling experimental artifacts. We require the scientist agent to submit three deliverables: (1) `mechanism.md` explains the causal logic of M; (2) `mechanism.py` implements M as a Python program for evaluation on the test set; and (3) `experiments.log` $\mathcal{L}$, the record of the experiments, including findings, ideas, and informative failures. We also allow the scientist agent to submit up to 1 GB of important observations from τ that it believes could help other scientists continue this line of research.

Evaluating the deliverables. After the scientist agent finishes the experimentation loop, a judge agent reads M, $\mathcal{L}$, and τ to conduct scientific constraint tests according to the criteria in the rubric $\mathcal{C}$. We execute `mechanism.py` on the test set to evaluate whether the discovered mechanism can make accurate predictions. For scientific insights $\mathcal{I}$, we provide the judge agent with all deliverables M, $\mathcal{L}$, and τ, and allow the judge agent to interact with the environment interface $\mathcal{E}$. The judge agent checks each insight question in $\mathcal{C}$ to determine whether the corresponding insight can be derived from M. We compute the score using R as detailed in §2.2.

2.2 Evaluation Protocol

Evaluation Metrics. We evaluate the discovered mechanism M on three aspects (Figure 2(g)). *(1) Scientific constraint tests (SC).* SC tests whether M satisfies the domain's common-sense assumptions, with a pass-or-fail result for each constraint. *(2) Predictive accuracy (PA).* PA measures how accurately M predicts the held-out test data. Each prediction criterion receives a continuous score normalized to $[0, 1]$, and the PA score is the mean of these scores. *(3) Interpretable scientific insights (SI).* We define the SI score as the fraction of insight questions in $\mathcal{C}$ whose corresponding insights can be derived from M, assuming M is correct.

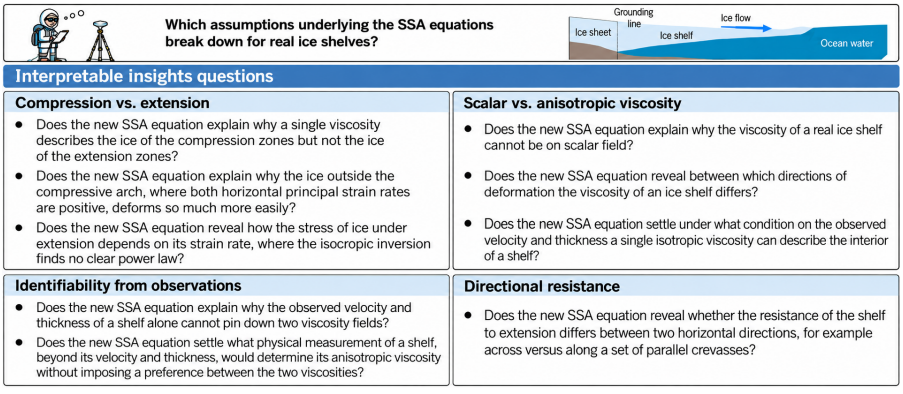

Figure 4: **The 9 interpretable insight questions for the Antarctic ice shelf problem in geophysics in EUREKABENCH.** These insights explain why ice deforms differently under compression and extension, how its resistance varies with direction, and which measurements determine its viscosity.

Evaluation rubric construction. We construct $\mathcal{C}$ with domain experts. The PA test data come from the original studies or our own simulator experiments. We add noise to these data. We express scientific insights as yes-or-no questions about what can be derived from a "perfect" interpretable mechanism (Figure 4). These questions draw on insights derived from recent human progress, open

questions in the papers' limitations, and the potential insights proposed by experts. Our domain experts verify that every insight question in EUREKABENCH is scientifically important and interesting.

Agentic judge. We use Claude Opus 4.8 and GPT 5.6 Sol as independent agentic judges for SC tests and SI evaluation. We provide both judges with M, $\mathcal{L}$, τ, and the relevant criteria in $\mathcal{C}$. For SC tests, the judges check whether the mechanism's explanation and executable implementation contradict the domain's common-sense assumptions. For SI, we instruct judges to assume that M is correct and allow them to run additional experiments by querying $\mathcal{E}$ to determine whether each insight can be derived from M. The judges are instructed to keep M unchanged and to reject answers based only on their own knowledge or simulator results that are unsupported by M. For every criterion, they return a verdict and an explanation identifying the supporting evidence. An insight question receives a `Yes` verdict only when the judge agent provides complete evidence showing how the corresponding insight can be derived from M (Appendix C.2).

Conditioned score. A mechanism that violates the domain's common-sense assumptions is not a valid explanation of the observations. We therefore condition the score on passing all SC tests so that high PA or SI scores cannot compensate for these violations. We compute the conditioned score separately for each judge. Let n_d denote the number of criteria in dimension d, p_d the total points earned on those criteria, and b_{SC} the number of satisfied scientific constraints. A submission scores zero unless it satisfies all scientific constraints:

$$S_{\text{conditioned}} = \begin{cases} \dfrac{p_{\mathrm{PA}} + p_{\mathrm{SI}}}{n_{\mathrm{PA}} + n_{\mathrm{SI}}} & \text{if } b_{\mathrm{SC}} = n_{\mathrm{SC}}, \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

We average the two judges' conditioned scores for each submission.

3 EUREKABENCH

3.1 BENCHMARK CONSTRUCTION

We construct EUREKABENCH with the experts by selecting scientific problems and developing the task specifications, experimental environments, and evaluation rubrics.

In-depth engagement with domain experts. We build EUREKABENCH with domain experts who hold PhDs in their respective fields and come from top research universities and institutes in the world. We begin with in-depth discussions about where AI could contribute to their research and identify mechanism discovery as a valuable target. The experts engage deeply with us to capture important scientific problems in each domain and the potential insights each problem could yield. Together, we refine the task specifications and every evaluation criterion through multiple rounds of detailed discussion and revision.

Problem selection. We select scientifically important problems that matter to their respective domains. Human scientists have already found mechanisms that partially explain the initial observations and derived scientific insights from them, but the problems remain only partially solved. We draw on these findings to construct insight questions, grounding evaluation in what a mechanism can explain and what insights can be derived from it, without requiring the mechanism to match a complete ground-truth solution. We provide background on each problem in Appendices E–J.

Task formulation. We formulate the task around finding an interpretable physical mechanism by starting with the initial observations that challenged experts' understanding of the problem. The task specification includes these initial observations and provides previous observations as scientific background. We ask the agent to run an iterative experimentation loop to find the underlying mechanism which can derive the interpretable scientific insights.

Environment interfaces. We build an experimental environment for each problem and package it with its dependencies in an Apptainer image. We use the tools listed in Table 1 as environment interfaces and provide a CLI with usage instructions.

3.2 BENCHMARK STATISTICS

Sci Domains	Env Interfaces
Neuroscience	c302[a]; TRIBE v2[b]
Geophysics	FEniCSx[c]; HBV[d]
Plasma Physics	TORAX[e]; TokaMaker with GPEC[f]
Astrophysics	CAMELS[g]; The DREAMS Project[h]
Computer Science	Open-weight LMs
Chemistry	AIMNet2-rxn[i]

[a](Gleeson et al., 2018); [b](d'Ascoli et al., 2026); [c](Baratta et al., 2023); [d](Bergström, 1992); [e](Citrin et al., 2024); [f](Hansen et al., 2024); [g](Villaescusa-Navarro et al., 2021); [h](Rose et al., 2025); [i](Anstine et al., 2025).

Table 1: **Environment interfaces in EUREKABENCH.**

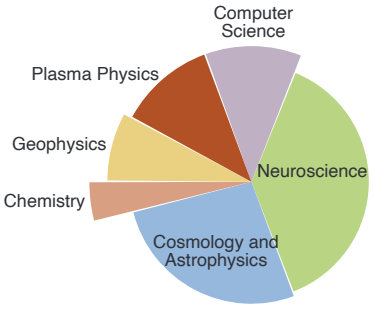

Figure 5: **Problem distribution across domains.**

EUREKABENCH contains 26 problems across six science domains. We have 10 problems in neuroscience, 2 in geophysics, 3 in plasma physics, 7 in astrophysics, 3 in computer science, and 1 in chemistry. Table 1 lists the environment interfaces, and Figure 5 shows the problem distribution across domains. Across these problems, EUREKABENCH includes 306 interpretable insight questions to evaluate the scientific insights agents uncover through the mechanism discovery process. Examples of initial observations appear in Figure 2(a)–(f).

4 EXPERIMENTS

Models. We evaluate 7 LLMs, including both frontier closed-source and open-weight models on EUREKABENCH. The closed-source models include Claude Opus 5 (Anthropic, 2026d), Claude Opus 4.8 (Anthropic, 2026c), Claude Fable 5.1 (Anthropic, 2026b), GPT 6 Astra (OpenAI, 2026c), and GPT 5.6 Sol (OpenAI, 2026b). The open-weight models include DeepSeek V4 Flash (DeepSeek-AI, 2026) and Kimi K3 (Moonshot AI, 2026). All models are run with the maximum reasoning effort.

Agent harnesses. We use three agent harnesses to run the models as autonomous agents on EUREKABENCH. We run Claude models with Claude Code (Anthropic, 2026a) and GPT models with Codex (OpenAI, 2026a) because each harness is developed by the corresponding model provider. Both harnesses are launched through the ACP agent in OpenHands (OpenHands, 2026). For open-weight models, we use the OpenHands Agent SDK (Wang et al., 2025a), which supports different LLMs through a single interface.

Human scientist baselines. We use the current progress made by human scientists on each problem as the human solution. This serves as a reference for evaluating AI agents.

Compute budgets for AI agents. For each problem, the agent is given one H100 GPU and a budget of 1,000 iterations and 4 hours per run to discover the underlying mechanism through interaction with the same environment.

Web access. We allow web search but block access to the research underlying each problem, including the original publications and literature aggregators that provide their content. These restrictions apply to both search and webpage retrieval to prevent agents from looking up the human scientist discovered mechanisms. These restrictions limit access during evaluation but do not rule out exposure to the source research during pretraining.

5 RESULTS

AI Agents still struggle to find the underlying mechanism. In Table 2, our results show that AI agents struggle to find mechanisms that explain the initial observations. The conditioned scores range from 12.9% to 29.8% across the seven AI agents. Even the best-performing agent, Claude

Agents	Scientific Value				Efficiency	
	Conditioned Score (%)	Predictive Accuracy (%)	Insights (%)	SC Failures ↓	Calls	Cost ($)
Claude Code						
Claude Fable 5.1	**29.8**	**47.4**	**42.4**	15/26	104	23.96
Claude Opus 5	27.3	42.9	36.2	15/26	166	17.56
Claude Opus 4.8	18.4	42.7	26.9	16/26	93	9.75
Codex						
GPT 6 Astra	28.7	**47.4**	29.4	**10/26**	149	24.62
GPT 5.6 Sol	19.4	41.9	25.9	17/26	55	2.23
OpenHands						
DeepSeek V4 Flash	14.6	29.9	24.4	15/26	206	1.24
Kimi K3	12.9	28.8	25.2	16/26	185	7.19
Human scientist	63.7	48.8	69.7	0/26	–	–

Table 2: **Main results on EUREKABENCH.** PA and insights are reported unconditionally; SC failures count completed problems that fail either judge's scientific-constraint tests, out of 26 problems.

Fable 5.1, achieves only 29.8%, compared with 63.7% for human-discovered mechanisms. Surprisingly, except GPT 6 Astra (10/26), all agents fail to pass the scientific-constraint tests on more than half of the problems in EUREKABENCH, showing that even top AI agents cannot reliably discover mechanisms that satisfy the most basic common-sense assumptions. Our empirical results show that current AI agents still lack the ability to reliably discover mechanisms that explain observations through the scientific experiment loop, which is a core capability for scientific discovery.

Agentic ability to discover scientific insights. To study the causes of this capability gap, we report predictive accuracy and insight scores separately in Table 2. We find that AI agents intentionally treat scientific mechanism discovery as an optimization task. Claude Fable 5.1 and GPT 6 Astra both achieve a predictive accuracy of 47.4%, compared with 48.8% for human solutions. However, their insight scores are only 42.4% and 29.4%, compared with 69.7% for human scientists. This large gap in insight discovery demonstrates that human scientists value the interpretation and extrapolation of mechanisms, while AI agents tend to prioritize brute-force numerical optimization. Therefore, optimizing predictive accuracy does not imply discovering novel scientific insights (Appendix D.1).

6 ANALYSIS

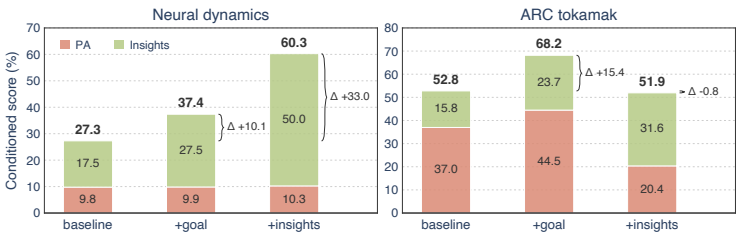

Figure 6: **Effects of goals and insight questions on mechanism discovery.** We give Claude Fable 5.1 general guidance (+goal) or insight questions (+insights); stacked bars show estimated PA and insight contributions to the conditioned score, with Δ marking changes from baseline.

Does additional guidance improve mechanism discovery? We study two types of guidance: enabling the "goal" feature (+goal) and directly including insight questions in the task specification (+insights). We evaluate Claude Fable 5.1, the best-performing agent, on one neuroscience problem (*neural dynamics*) and one plasma physics problem (*ARC tokamak*), as shown in Figure 6. In the +goal condition, we specify that a good mechanism should be not only accurate but also useful for the future exploration (Appendix B.1). Enabling this feature increases the insight contribution to the conditioned score on *neural dynamics* ($\Delta + 10.0\%$) and *ARC tokamak* ($\Delta + 7.9\%$). Directly providing insight questions in the task specification as explicit optimization targets substantially increases the insight contribution on *neural dynamics* ($\Delta + 32.5\%$) and *ARC tokamak* ($\Delta + 15.8\%$). However, this is not a natural setting: when scientists first observe a phenomenon that challenges their understanding, they do not yet know what insights its underlying mechanism will yield.

How many insights are derived from the discovered mechanism? To assess insights agents derive during experimentation, the judge evaluates their submitted mechanisms without simulator access (Figure 7). All agents score lower than in the main evaluation, where the judge interprets and extrapolates their mechanisms using the simulator. Claude Fable 5.1 scores highest and shows the largest gap: 22.7% from experimentation versus 42.4% from its mechanism. We also observe that there is 6.7% cases that judges credit insights from experiment logs but cannot derive from the mechanism. This allows us to evaluate the mechanism's extrapolation separately from the insights agents find during experiments.

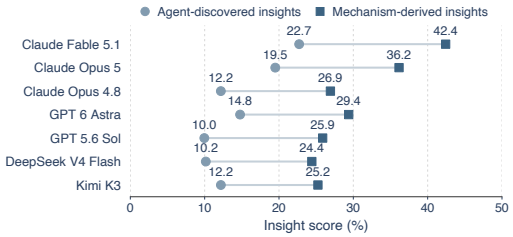

Figure 7: **Insights discovered during the experimentation loop vs. those derived from the discovered mechanism.** Both scores use the same problems and instances.

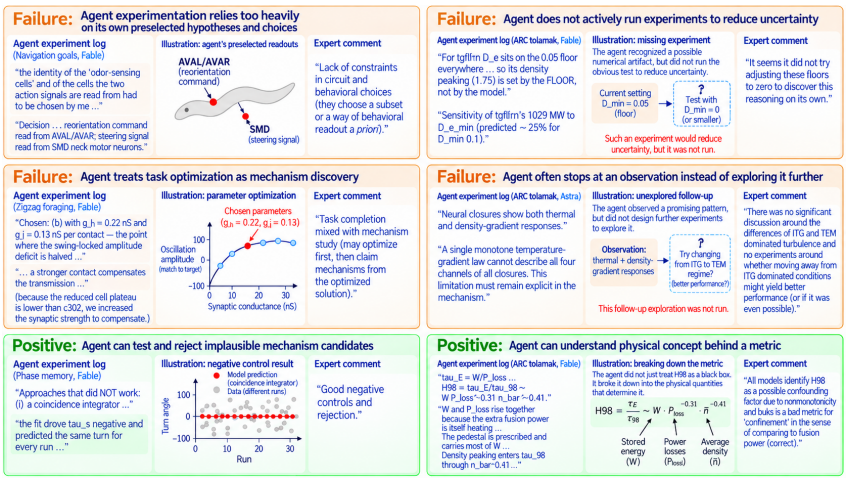

Figure 8: **Expert review of mechanism discovery.** Experts review experiment logs of Claude Fable 5.1 and GPT 6 Astra on neuroscience and plasma physics tasks, highlighting failures in mechanism discovery alongside successful hypothesis rejection and understanding of physical concepts.

Qualitative analysis: where do agents' experiments fall short? Our experts review agents' mechanisms and experiment logs and identify four types of failure that hinder the discovery of novel scientific insights (Figure 8). ***(1) Overreliance on preselected hypotheses and choices.*** Agents build explanations around unjustified choices. In *navigation goals*, the agent selects neuronal readouts without establishing whether they capture the relevant circuit and behavior. ***(2) Treating task optimization as mechanism discovery.*** Agents equate fitting observations with explaining them. In *zigzag foraging*, the agent adjusts synaptic strength to match a target response and presents the optimized result as a mechanism. ***(3) Not using experiments to resolve uncertainty.*** Claude Fable 5.1 recognizes a possible numerical artifact in *ARC tokamak* but does not lower the diffusivity floor to test its effect. ***(4) Stopping at observations without further exploration.*** GPT 6 Astra observes responses to temperature and density gradients in *ARC tokamak* but does not explore whether changing the dominant turbulence regime could improve performance. Nevertheless, agents sometimes make sound experimental decisions, such as rejecting mechanisms that contradict the evidence, and recognize the physical meaning of variables and model parameters. These findings suggest that agents are better at task optimization than at open-ended experimentation to resolve scientific uncertainty.

7 Related Work

AI scientist Agents. These agents automate literature review, idea generation, experiments, and paper writing (Lu et al., 2024; Yamada et al., 2025; Schmidgall et al., 2025; Tang et al., 2025).

Recent systems add persistent research memory and link claims to evidence (Lyu et al., 2026; Meng et al., 2026); others use execution feedback and replication training to improve research capabilities (Si et al., 2026; Falck et al., 2026). Beyond machine learning, agents support laboratory experiments and hypothesis validation (Boiko et al., 2023; Gottweis et al., 2026; Ghareeb et al., 2026) or combine literature analysis with large-scale data exploration or multimodal observations (Mitchener et al., 2025; Li et al., 2026). Other systems use automated evaluators to search for executable algorithms or equations (Romera-Paredes et al., 2024; Novikov et al., 2025; Shojaee et al., 2025a). Closer to our setting, agents use experiments to revise mechanistic models or investigate mechanisms in physical and computational systems (Kabra et al., 2026; Murphy, 2026; Wang et al., 2026a;b). Together, these efforts broaden both the scope of research automation and the forms of scientific evidence agents can use.

Benchmarking scientific discovery. Existing benchmarks evaluate scientific tool use and research workflows (Chen et al., 2024; Bragg et al., 2026; Liu et al., 2026), verifiable outputs and scientific software engineering (Terminal-Bench-Science Team, 2026; Xu et al., 2026), experiment replication (Starace et al., 2025; Kon et al., 2025; Qiu et al., 2026), or gains over numerical baselines (Chan et al., 2024; Nathani et al., 2025; Lupidi et al., 2026; Wang et al., 2026c). DiscoveryBench and LLM-SRBench assess hypotheses and equations recovered from data (Majumder et al., 2024; Shojaee et al., 2025b), while interactive benchmarks let agents gather observations to uncover hidden systems and physical laws (Geng et al., 2025a; Koblischke et al., 2025; Chen et al., 2025b; Zheng et al., 2025; Jain et al., 2026). DiscoveryWorld evaluates explanatory knowledge, and BoxingGym tests whether explanations help another agent predict unseen outcomes (Jansen et al., 2024; Gandhi et al., 2025). Closest to our setting, DiscoverPhysics evaluates both trajectory prediction and conceptual explanations against expert-written rubrics in simulated worlds with noncanonical physics (Wiemann et al., 2026). Unlike benchmarks centered on task completion or numerical performance, we focus on a core capability of scientific discovery: whether agents can uncover scientifically meaningful insights while seeking an answer, rather than merely produce a correct final answer.

8 Limitations and Future Directions

Multiple paths in scientific discovery. Scientific discovery can follow different paths from the same initial observations, leading to different mechanisms and scientific insights. Our insight questions reflect the research directions selected with domain experts and capture only part of what can be learned about each problem. Future work could expand the insight set to cover alternative paths, with domain experts assessing findings that extend beyond established knowledge.

Evaluating scientific insights. Our insight evaluation requires judge agents to interpret and extrapolate mechanisms, which means it is also constrained by their own scientific reasoning abilities. Judges may accept confidently stated conclusions without adequately establishing how they follow from the mechanism, potentially overestimating the scientific insights it supports. The expert reviews in our qualitative analysis suggest that some automated assessments are overly generous, although these reviews provide broad judgments of selected submissions rather than scores for every insight question. On *ARC tokamak*, our fusion expert rated Claude Fable 5.1 above the other submissions they had reviewed, but considered its work closer to that of a master's student in the domain than a researcher.

9 Conclusion

EurekaBench evaluates whether AI agents can use long-horizon experimentation to discover mechanisms that explain observations and yield scientific insights. Across 26 problems in six domains, the strongest agents approach human predictive accuracy, yet their mechanisms often violate basic scientific constraints and support substantially fewer insights than human-discovered mechanisms. Insight scores fall further when we evaluate agents' own experimental findings. Expert reviews identify untested assumptions, unresolved uncertainties, and a tendency to treat fitting observations as mechanism discovery. These findings point to the need for agents that use experiments to test explanations and explore what can be learned from them. By evaluating scientific constraints, predictive accuracy, and scientific insights together, EurekaBench measures progress toward agents that contribute to scientific discovery through their experiments.

AI USE STATEMENT

We used AI tools to polish the writing, create figures, and organize tables. The authors reviewed and revised the resulting text, figures, and tables. The authors take responsibility for the final content of this paper.

ETHICS STATEMENT

We conduct agent experiments in computational environments based on existing scientific datasets, simulators, and software. Domain experts contribute to problem construction and review selected agent submissions. Incorrect mechanisms and unsupported insights could mislead subsequent research. We therefore assess scientific constraints alongside predictive accuracy and scientific insights, use two independent judge agents, and discuss the limitations of automated evaluation in Section 8. Agent-generated mechanisms and insights require independent scientific validation before they inform further research or practical decisions.

REPRODUCIBILITY STATEMENT

We describe the task formulation and evaluation protocol in Section 2, benchmark construction in Section 3.1, and the evaluated agents, harnesses, compute budgets, and web-access restrictions in Section 4. Appendices A–C provide a task specification template, guidance prompts, and judge instructions. Problem backgrounds and experimental interfaces are described in Appendices E–J. The evaluation data and complete problem-specific rubrics remain private, so reproducing the full benchmark evaluation requires access to these materials.

REFERENCES

Dylan M. Anstine, Qiyuan Zhao, Roman Zubatiuk, Shuhao Zhang, Veerupaksh Singla, Filipp Nikitin, Brett M. Savoie, and Olexandr Isayev. AIMNet2-rxn: A machine learned potential for generalized reaction modeling on a millions-of-pathways scale, 2025. URL https://doi.org/10.26434/chemrxiv-2025-hpdmg. ChemRxiv preprint.

Anthropic. Claude code. https://code.claude.com/docs/en/overview, 2026a.

Anthropic. Introducing claude fable 5.1 and claude mythos 5.1. https://www.anthropic.com/claude-fable-and-mythos-5-1, 2026b.

Anthropic. Introducing claude opus 4.8. https://www.anthropic.com/news/claude-opus-4-8, 2026c.

Anthropic. System card: Claude opus 5. https://www.anthropic.com/news/claude-opus-5, 2026d. July 24, 2026.

Alan Armstrong, Roberto A. Boto, Paul Dingwall, Julia Contreras-García, Matt J. Harvey, Nicholas J. Mason, and Henry S. Rzepa. The houk–list transition states for organocatalytic mechanisms revisited. *Chem. Sci.*, 5(5):2057–2071, 2014. ISSN 2041-6539. doi: 10.1039/c3sc53416b. URL http://dx.doi.org/10.1039/c3sc53416b.

S. Bahmanyar and K. N. Houk. The origin of stereoselectivity in proline-catalyzed intramolecular aldol reactions. *Journal of the American Chemical Society*, 123(51):12911–12912, December 2001. ISSN 1520-5126. doi: 10.1021/ja011714s. URL http://dx.doi.org/10.1021/ja011714s.

Igor A. Baratta, Joseph P. Dean, Jørgen S. Dokken, Michal Habera, Jack S. Hale, Chris N. Richardson, Marie E. Rognes, Matthew W. Scroggs, Nathan Sime, and Garth N. Wells. DOLFINx: The next generation FEniCS problem solving environment, 2023. URL https://doi.org/10.5281/zenodo.10447666. Preprint.

Sten Bergström. The HBV model—its structure and applications. SMHI Reports Hydrology 4, Swedish Meteorological and Hydrological Institute, Norrköping, Sweden, 1992. URL https://www.smhi.se/en/publications-from-smhi/publications/1992-09-03-the-hbv-model---its-structure-and-applications.

Daniil A. Boiko, Robert MacKnight, Ben Kline, and Gabe Gomes. Autonomous chemical research with large language models. *Nature*, 624(7992):570–578, 2023. doi: 10.1038/s41586-023-06792-0. URL https://www.nature.com/articles/s41586-023-06792-0.

Jonathan Bragg, Mike D'Arcy, Nishant Balepur, Dan Bareket, Bhavana Dalvi Mishra, Sergey Feldman, Dany Haddad, Jena Hwang, Peter Jansen, Varsha Kishore, et al. Astabench: Rigorous benchmarking of ai agents with a scientific research suite. In *International conference on learning representations*, volume 2026, pp. 110136–110223, 2026.

Zak Buhmann, Amanda K. Robinson, Jason B. Mattingley, and Reuben Rideaux. Inverted encoding of neural responses to audiovisual stimuli reveals super-additive multisensory enhancement. *eLife*, 13:RP97230, 2026. doi: 10.7554/eLife.97230.3.

Jun Shern Chan, Neil Chowdhury, Oliver Jaffe, James Aung, Dane Sherburn, Evan Mays, Giulio Starace, Kevin Liu, Leon Maksin, Tejal Patwardhan, Lilian Weng, and Aleksander Mądry. MLE-bench: Evaluating Machine Learning Agents on Machine Learning Engineering. *arXiv preprint arXiv:2410.07095*, 2024. URL https://arxiv.org/abs/2410.07095.

Kevin S. Chen, Anuj K. Sharma, Jonathan W. Pillow, and Andrew M. Leifer. Navigation strategies in Caenorhabditis elegans are differentially altered by learning. *PLOS Biology*, 23(3):e3003005, 2025a. doi: 10.1371/journal.pbio.3003005.

Kevin S. Chen, Jonathan W. Pillow, and Andrew M. Leifer. State-switching navigation strategies in Caenorhabditis elegans are beneficial for chemotaxis. *Proceedings of the National Academy of Sciences*, 123(25):e2519999123, 2026. doi: 10.1073/pnas.2519999123.

Yimeng Chen, Piotr Piękos, Mateusz Ostaszewski, Firas Laakom, and Jürgen Schmidhuber. PhysGym: Benchmarking LLMs in Interactive Physics Discovery with Controlled Priors. *arXiv preprint arXiv:2507.15550*, 2025b. URL https://arxiv.org/abs/2507.15550.

Ziru Chen, Shijie Chen, Yuting Ning, Qianheng Zhang, Boshi Wang, Botao Yu, Yifei Li, Zeyi Liao, Chen Wei, Zitong Lu, Vishal Dey, Mingyi Xue, Frazier N. Baker, Benjamin Burns, Daniel Adu-Ampratwum, Xuhui Huang, Xia Ning, Song Gao, Yu Su, and Huan Sun. ScienceAgentBench: Toward Rigorous Assessment of Language Agents for Data-Driven Scientific Discovery. *arXiv preprint arXiv:2410.05080*, 2024. URL https://arxiv.org/abs/2410.05080.

Anthony Chuma, Hans W. Horn, William C. Swope, Russell C. Pratt, Lei Zhang, Bas G. G. Lohmeijer, Charles G. Wade, Robert M. Waymouth, James L. Hedrick, and Julia E. Rice. The reaction mechanism for the organocatalytic ring-opening polymerization of L-lactide using a guanidine-based catalyst: Hydrogen-bonded or covalently bound? *Journal of the American Chemical Society*, 130(21):6749–6754, May 2008. ISSN 1520-5126. doi: 10.1021/ja0764411. URL http://dx.doi.org/10.1021/ja0764411.

Jonathan Citrin, Ian Goodfellow, Akhil Raju, Jeremy Chen, Jonas Degrave, Craig Donner, Federico Felici, Philippe Hamel, Andrea Huber, Dmitry Nikulin, et al. Torax: a fast and differentiable tokamak transport simulator in jax. *arXiv preprint arXiv:2406.06718*, 2024.

Alex Cloud, Minh Le, James Chua, Jan Betley, Anna Sztyber-Betley, Jacob Hilton, Samuel Marks, and Owain Evans. Subliminal learning: Language models transmit behavioral traits via hidden signals in data. *arXiv preprint arXiv:2507.14805*, 2025. URL https://arxiv.org/abs/2507.14805.

Belén Costanza, Bonny Y. Wang, Francisco Villaescusa-Navarro, Alex M. Garcia, Jonah C. Rose, Mark Vogelsberger, Paul Torrey, Arya Farahi, Xuejian Shen, and Ilem Leisher. On the sensitivity of different galaxy properties to warm dark matter. *arXiv preprint arXiv:2510.05037*, 2025. URL https://arxiv.org/abs/2510.05037.

Stéphane d'Ascoli, Jérémy Rapin, Yohann Benchetrit, Teon Brooks, Katelyn Begany, Joséphine Raugel, Hubert Banville, and Jean-Rémi King. A foundation model of vision, audition, and language for in-silico neuroscience. *arXiv preprint arXiv:2605.04326*, 2026.

DeepSeek-AI. Deepseek-v4: Towards highly efficient million-token context intelligence, 2026.

Raymond L. Dunn, Caitriona M. Costello, Jackson M. Borchardt, Daniel Y. Sprague, Grace C. Chiu, Julia M. Miller, Noelle D. L'Etoile, and Saul Kato. Short-term memory by distributed neural network oscillators in a simple nervous system. *Current Biology*, 35(22):5582–5593.e4, 2025. doi: 10.1016/j.cub.2025.10.018.

Bradley Efron. Bootstrap methods: Another look at the jackknife. *The Annals of Statistics*, 7 (1):1–26, 1979. doi: 10.1214/aos/1176344552. URL https://doi.org/10.1214/aos/1176344552.

Damon Falck, Samer Sabri, Anja Surina, Thom Foster, Anya Sims, Sam Devlin, Dylan Rogers, Tantum Collins, Kaloyan Aleksiev, Louis Kirsch, et al. Training ai scientists to replicate research. *arXiv preprint arXiv:2608.13331*, 2026.

Charles Fieseler, Itamar Lev, Ulises Rey, Lukas Hille, Hannah Brenner, and Manuel Zimmer. An intrinsic neuronal manifold underlies brain-wide hierarchical organization of behavior in C. elegans. *bioRxiv*, 2025. doi: 10.1101/2025.03.09.642241.

Kanishk Gandhi, Michael Y. Li, Lyle Goodyear, Agam Bhatia, Louise Li, Aditi Bhaskar, Mohammed Zaman, and Noah D. Goodman. BoxingGym: Benchmarking Progress in Automated Experimental Design and Model Discovery. *arXiv preprint arXiv:2501.01540*, 2025. URL https://arxiv.org/abs/2501.01540.

Alex M. Garcia, Jonah C. Rose, Paul Torrey, Andrea Caputo, Mariangela Lisanti, et al. The DREAMS project: Disentangling the impact of halo-to-halo variance and baryonic feedback on Milky Way dark matter density profiles. *arXiv preprint arXiv:2512.03132*, 2025. URL https://arxiv.org/abs/2512.03132.

Jiayi Geng, Howard Chen, Dilip Arumugam, and Thomas L Griffiths. Are large language models reliable ai scientists? assessing reverse-engineering of black-box systems. *arXiv preprint arXiv:2505.17968*, 2025a.

Jiayi Geng, Howard Chen, Ryan Liu, Manoel Horta Ribeiro, Robb Willer, Graham Neubig, and Thomas L. Griffiths. Accumulating context changes the beliefs of language models. *arXiv preprint arXiv:2511.01805*, 2025b. URL https://arxiv.org/abs/2511.01805.

Ali E. Ghareeb, Benjamin Chang, Ludovico Mitchener, Angela Yiu, Caralyn J. Szostkiewicz, Dmytro Shved, Gavin J. Gyimesi, Jon M. Laurent, Samantha M. Wright, Muhammed T. Razzak, Andrew D. White, Silvia C. Finnemann, Michaela M. Hinks, and Samuel G. Rodriques. A multi-agent system for automating scientific discovery. *Nature*, 655(8122):497–505, 2026. doi: 10.1038/s41586-026-10652-y. URL https://www.nature.com/articles/s41586-026-10652-y.

A. H. Glasser. The direct criterion of newcomb for the ideal mhd stability of an axisymmetric toroidal plasma. *Physics of Plasmas*, 23(7), July 2016. ISSN 1089-7674. doi: 10.1063/1.4958328. URL http://dx.doi.org/10.1063/1.4958328.

Padraig Gleeson, David Lung, Radu Grosu, Ramin Hasani, and Stephen D Larson. c302: a multiscale framework for modelling the nervous system of caenorhabditis elegans. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 373(1758):20170379, 2018.

Juraj Gottweis, Wei-Hung Weng, Alexander Daryin, Tao Tu, Petar Sirkovic, Artiom Myaskovsky, Grzegorz Glowaty, Felix Weissenberger, Alessio Orlandi, Dan Popovici, et al. Accelerating scientific discovery with co-scientist. *Nature*, 655(8122):487–496, 2026. doi: 10.1038/s41586-026-10644-y. URL https://www.nature.com/articles/s41586-026-10644-y.

Christopher Hansen, Ian G Stewart, Daniel Burgess, Matthew Pharr, Sophia Guizzo, F Logak, Andrew Oakleigh Nelson, and Carlos Paz-Soldan. Tokamaker: An open-source time-dependent grad-shafranov tool for the design and modeling of axisymmetric fusion devices. *Computer Physics Communications*, 298:109111, 2024.

Sture Holm. A simple sequentially rejective multiple test procedure. *Scandinavian Journal of Statistics*, 6(2):65–70, 1979. URL https://www.jstor.org/stable/4615733.

N. T. Howard, P. Rodriguez-Fernandez, J. Hall, M. Muraca, A. Saltzman, A. Ho, J. C. Hillesheim, A. J. Creely, T. H. Eich, T. Body, P. B. Snyder, and C. Holland. Performance and transport in the ARC tokamak. *Journal of Plasma Physics*, 92:E67, 2026. URL https://www.cambridge.org/core/journals/journal-of-plasma-physics/article/performance-and-transport-in-the-arc-tokamak/52AAD67DBE7CAEC7E864E74AA2967C5C.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. URL https://openreview.net/forum?id=nZeVKeeFYf9.

Moksh Jain, Mehdi Bennani, Johannes Bausch, Yuri Chervonyi, Bogdan Georgiev, Simon Osindero, and Nenad Tomašev. MaD Physics: Evaluating information seeking under constraints in physical environments. *arXiv preprint arXiv:2605.10820*, 2026. URL https://arxiv.org/abs/2605.10820.

Peter Jansen, Marc-Alexandre Côté, Tushar Khot, Erin Bransom, Bhavana Dalvi Mishra, Bodhisattwa Prasad Majumder, Oyvind Tafjord, and Peter Clark. DiscoveryWorld: A virtual environment for developing and evaluating automated scientific discovery agents. *arXiv preprint arXiv:2406.06769*, 2024. URL https://arxiv.org/abs/2406.06769.

Sanchit Kabra, Nikhil Abhyankar, Saaketh Desai, Prasad Iyer, and Chandan K Reddy. Llm-autoscilab: Closed-loop scientific discovery via active experimentation with llms. *arXiv preprint arXiv:2605.24043*, 2026.

Maurice G. Kendall. The treatment of ties in ranking problems. *Biometrika*, 33(3):239–251, 1945. doi: 10.1093/biomet/33.3.239. URL https://doi.org/10.1093/biomet/33.3.239.

Nolan Koblischke, Hyunseok Jang, Kristen Menou, and Mohamad Ali-Dib. Gravity-Bench-v1: A Benchmark on Gravitational Physics Discovery for Agents. *arXiv preprint arXiv:2501.18411*, 2025. URL https://arxiv.org/abs/2501.18411.

Patrick Tser Jern Kon, Jiachen Liu, Xinyi Zhu, Qiuyi Ding, Jingjia Peng, Jiarong Xing, Yibo Huang, Yiming Qiu, Jayanth Srinivasa, Myungjin Lee, Mosharaf Chowdhury, Matei Zaharia, and Ang Chen. EXP-Bench: Can AI Conduct AI Research Experiments? *arXiv preprint arXiv:2505.24785*, 2025. URL https://arxiv.org/abs/2505.24785.

Ilem Leisher, Paul Torrey, Alex M. Garcia, Jonah C. Rose, Francisco Villaescusa-Navarro, Zachary Lubberts, Arya Farahi, Stephanie O'Neil, Xuejian Shen, Olivia Mostow, Nitya Kallivayalil, Dhruv Zimmerman, Desika Narayanan, and Mark Vogelsberger. Linking warm dark matter to merger tree histories via deep learning networks. *arXiv preprint arXiv:2511.05367*, 2025. URL https://arxiv.org/abs/2511.05367.

Bobo Li, Hao Fei, Tianjie Ju, Mong-Li Lee, and Wynne Hsu. OmniScientist: An Omni-Modal Omni-Discipline AI Scientist. *arXiv preprint arXiv:2608.13558*, 2026. URL https://arxiv.org/abs/2608.13558.

Kito Liao, Francisco Villaescusa-Navarro, Romain Teyssier, and Natalí S. M. de Santi. Cosmology with one galaxy: An analytic formula relating $\Omega_{\rm m}$ with galaxy properties. *arXiv preprint arXiv:2602.07651*, 2026. URL https://arxiv.org/abs/2602.07651.

Tianyu Liu, Allen Xin Wang, Antonia Panescu, Lisa Xinyi Chen, Wenxin Long, Xinyu Wei, Yueqian Jing, Ziyao Zeng, Jihang Chen, Sihan Jiang, Ziqing Wang, Siyi Gu, Siyu Chen, Xinyang Hu, Haoran Shao, Leqi Xu, Wangjie Zheng, Zhiyuan Cao, Ada Fang, Botao Yu, Kunyang Sun, Rex Ying, Arman Cohan, Qingyu Chen, Lingzhou Xue, Kaize Ding, Yuanqi Du, Wengong Jin, Zhuoran Yang, Marinka Zitnik, James Zou, Hua Xu, and Hongyu Zhao. Benchmarking AI Agents for Addressing Scientific Challenges Across Scales. *arXiv preprint arXiv:2606.12736*, 2026. URL https://arxiv.org/abs/2606.12736.

Chris Lu, Cong Lu, Robert Tjarko Lange, Jakob Foerster, Jeff Clune, and David Ha. The AI Scientist: Towards fully automated open-ended scientific discovery. *arXiv preprint arXiv:2408.06292*, 2024. URL https://arxiv.org/abs/2408.06292.

Alisia Lupidi, Bhavul Gauri, Thomas Simon Foster, Bassel Al Omari, Despoina Magka, Alberto Pepe, Alexis Audran-Reiss, Muna Aghamelu, Nicolas Baldwin, Lucia Cipolina-Kun, et al. Airs-bench: a suite of tasks for frontier ai research science agents. *arXiv preprint arXiv:2602.06855*, 2026.

Yougang Lyu, Xi Zhang, Xinhao Yi, Yuyue Zhao, Shuyu Guo, Wenxiang Hu, Jan Piotrowski, Jakub Kaliski, Jacopo Urbani, Zaiqiao Meng, Lun Zhou, and Xiaohui Yan. EvoScientist: Towards Multi-Agent Evolving AI Scientists for End-to-End Scientific Discovery. *arXiv preprint arXiv:2603.08127*, 2026. URL https://arxiv.org/abs/2603.08127.

Bodhisattwa Prasad Majumder, Harshit Surana, Dhruv Agarwal, Bhavana Dalvi Mishra, Abhijeetsingh Meena, Aryan Prakhar, Tirth Vora, Tushar Khot, Ashish Sabharwal, and Peter Clark. DiscoveryBench: Towards Data-Driven Discovery with Large Language Models. *arXiv preprint arXiv:2407.01725*, 2024. URL https://arxiv.org/abs/2407.01725.

Rui Meng, Bhavana Dalvi Mishra, Jiefeng Chen, Chun-Liang Li, Palash Goyal, Mihir Parmar, Yiwen Song, Yale Song, Rajarishi Sinha, Parthasarathy Ranganathan, et al. Scientistone: Towards human-level autonomous research via chain-of-evidence. *arXiv preprint arXiv:2605.26340*, 2026.

Ludovico Mitchener, Angela Yiu, Benjamin Chang, Mathieu Bourdenx, Tyler Nadolski, Arvis Sulovari, Eric C. Landsness, Daniel L. Barabasi, Siddharth Narayanan, Nicky Evans, et al. Kosmos: An AI scientist for autonomous discovery. *arXiv preprint arXiv:2511.02824*, 2025. URL https://arxiv.org/abs/2511.02824.

Moonshot AI. Kimi k3: Open frontier intelligence. https://www.kimi.ai/blog/kimi-k3, 2026.

Megan Morrison and Lai-Sang Young. A data-driven biology-based network model reproduces C. elegans premotor neural dynamics. *PLOS Computational Biology*, 21(12):e1013818, 2025. doi: 10.1371/journal.pcbi.1013818.

Aaron Mueller, Atticus Geiger, Sarah Wiegreffe, Dana Arad, Iván Arcuschin, Adam Belfki, Yik Siu Chan, Jaden Fiotto-Kaufman, Tal Haklay, Michael Hanna, Jing Huang, Rohan Gupta, Yaniv Nikankin, Hadas Orgad, Nikhil Prakash, Anja Reusch, Aruna Sankaranarayanan, Shun Shao, Alessandro Stolfo, Martin Tutek, Amir Zur, David Bau, and Yonatan Belinkov. MIB: A mechanistic interpretability benchmark. *arXiv preprint arXiv:2504.13151*, 2025. URL https://arxiv.org/abs/2504.13151.

Kevin Murphy. Model discovery agent: Llm-assisted bayesian experiment design for data-efficient discovery of mechanistic world models. *arXiv preprint arXiv:2608.09696*, 2026.

Neel Nanda and Joseph Bloom. Transformerlens. https://github.com/TransformerLensOrg/TransformerLens, 2022.

Deepak Nathani, Lovish Madaan, Nicholas Roberts, Nikolay Bashlykov, Ajay Menon, Vincent Moens, Amar Budhiraja, Despoina Magka, Vladislav Vorotilov, Gaurav Chaurasia, Dieuwke Hupkes, Ricardo Silveira Cabral, Tatiana Shavrina, Jakob Foerster, Yoram Bachrach, William Yang Wang, and Roberta Raileanu. MLGym: A New Framework and Benchmark for Advancing AI Research Agents. *arXiv preprint arXiv:2502.14499*, 2025. URL https://arxiv.org/abs/2502.14499.

A. J. Newman, M. P. Clark, K. Sampson, A. Wood, L. E. Hay, A. Bock, R. J. Viger, D. Blodgett, L. Brekke, J. R. Arnold, T. Hopson, and Q. Duan. Development of a large-sample watershed-scale hydrometeorological data set for the contiguous usa: data set characteristics and assessment of regional variability in hydrologic model performance. *Hydrology and Earth System Sciences*, 19(1):209–223, 2015. doi: 10.5194/hess-19-209-2015. URL https://hess.copernicus.org/articles/19/209/2015/.

Alexander Novikov, Ngân Vũ, Marvin Eisenberger, Emilien Dupont, Po-Sen Huang, Adam Zsolt Wagner, Sergey Shirobokov, Borislav Kozlovskii, Francisco J. R. Ruiz, Abbas Mehrabian, M. Pawan Kumar, Abigail See, Swarat Chaudhuri, George Holland, Alex Davies, Sebastian Nowozin, Pushmeet Kohli, and Matej Balog. AlphaEvolve: A coding agent for scientific and algorithmic discovery. *arXiv preprint arXiv:2506.13131*, 2025. URL https://arxiv.org/abs/2506.13131.

OpenAI. Codex. https://openai.com/codex/, 2026a.

OpenAI. Gpt 5.6: Frontier intelligence that scales with your ambition. https://openai.com/index/gpt-5-6/, 2026b.

OpenAI. Gpt 6 astra: A new generation of intelligence. https://openai.com/index/gpt-6-astra/, 2026c.

OpenAI. On the Navier–Stokes Millennium Prize Problem, 2026d. URL https://openai.com/index/navier-stokes-solution/.

OpenHands. Acp agents. https://docs.openhands.dev/openhands/usage/agent-canvas/acp-agents, 2026.

Belinda Phipson and Gordon K. Smyth. Permutation P-values should never be zero: Calculating exact P-values when permutations are randomly drawn. *Statistical Applications in Genetics and Molecular Biology*, 9(1):Article 39, 2010. doi: 10.2202/1544-6115.1585. URL https://doi.org/10.2202/1544-6115.1585.

Shi Qiu, Junyi Deng, Yiwei Deng, Haoran Dong, Jieyu Fu, Mao Li, Zeyu Li, Zhaolong Zhang, Huiwen Zheng, Leidong Bao, et al. Prbench: End-to-end paper reproduction in physics research. *arXiv preprint arXiv:2603.27646*, 2026.

Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Matej Balog, M. Pawan Kumar, Emilien Dupont, Francisco J. R. Ruiz, Jordan S. Ellenberg, Pengming Wang, Omar Fawzi, Pushmeet Kohli, and Alhussein Fawzi. Mathematical discoveries from program search with large language models. *Nature*, 625(7995):468–475, 2024. doi: 10.1038/s41586-023-06924-6. URL https://www.nature.com/articles/s41586-023-06924-6.

Jonah C Rose, Paul Torrey, Francisco Villaescusa-Navarro, Mariangela Lisanti, Tri Nguyen, Sandip Roy, Kassidy E Kollmann, Mark Vogelsberger, Francis-Yan Cyr-Racine, Mikhail V Medvedev, et al. Introducing the dreams project: Dark matter and astrophysics with machine learning and simulations. *The Astrophysical Journal*, 982(2):68, 2025.

Wieger H. Scheurer and Micha Heilbron. Distinct regimes of spatial prediction across the visual field during natural vision. *bioRxiv*, 2026. doi: 10.64898/2026.03.27.714859.

Samuel Schmidgall, Yusheng Su, Ze Wang, Ximeng Sun, Jialian Wu, Xiaodong Yu, Jiang Liu, Michael Moor, Zicheng Liu, and Emad Barsoum. Agent Laboratory: Using LLM Agents as Research Assistants. *arXiv preprint arXiv:2501.04227*, 2025. URL https://arxiv.org/abs/2501.04227.

Parshin Shojaee, Kazem Meidani, Shashank Gupta, Amir Barati Farimani, and Chandan Reddy. Llm-sr: Scientific equation discovery via programming with large language models. In *International Conference on Learning Representations*, volume 2025, pp. 16054–16085, 2025a.

Parshin Shojaee, Ngoc-Hieu Nguyen, Kazem Meidani, Amir Barati Farimani, Khoa D Doan, and Chandan K Reddy. LLM-SRBench: A New Benchmark for Scientific Equation Discovery with Large Language Models. *arXiv preprint arXiv:2504.10415*, 2025b. URL https://arxiv.org/abs/2504.10415.

Chenglei Si, Zitong Yang, Yejin Choi, Emmanuel Candès, Diyi Yang, and Tatsunori Hashimoto. Towards execution-grounded automated ai research. *arXiv preprint arXiv:2601.14525*, 2026.

Xavier Sims, Daniel Anglés-Alcázar, Boon-Kiat Oh, Daisuke Nagai, Jonathan Mercedes-Feliz, Isabel Medlock, Yueying Ni, Christopher C. Lovell, and Francisco Villaescusa-Navarro. CAMELS environments: The impact of local neighbours on galaxy evolution across the SIMBA, IllustrisTNG, ASTRID, and Swift-EAGLE simulations. *arXiv preprint arXiv:2601.06290*, 2026. URL https://arxiv.org/abs/2601.06290.

Giulio Starace, Oliver Jaffe, Dane Sherburn, James Aung, Jun Shern Chan, Leon Maksin, Rachel Dias, Evan Mays, Benjamin Kinsella, Wyatt Thompson, et al. Paperbench: evaluating ai's ability to replicate ai research. *arXiv preprint arXiv:2504.01848*, 2025.

Laura M. Stoinski, Talia Konkle, and Martin N. Hebart. Principles of coarse-scale functional organization in occipitotemporal cortex. *bioRxiv*, 2026. doi: 10.64898/2026.02.18.706551.

Jiabin Tang, Lianghao Xia, Zhonghang Li, and Chao Huang. AI-Researcher: Autonomous Scientific Innovation. *arXiv preprint arXiv:2505.18705*, 2025. URL https://arxiv.org/abs/2505.18705.

Terence Tao. Post on mathstodon. https://mathstodon.xyz/@tao/117208617602946453, 2026. Posted September 3, 2026.

Terminal-Bench-Science Team. Terminal-Bench-Science: Evaluating AI agents on research workflows across scientific domains. https://github.com/harbor-framework/terminal-bench-science, 2026. Version 0.1.

K. E. Thome, M. E. Austin, S. Coda, A. Marinoni, A. Hyatt, A. O. Nelson, T. Odstrčil, C. A. Paz-Soldan, O. Sauter, F. Scotti, and B. Vanovac. The negative triangularity tokamak path for fusion pilot plants: Experimental progress and future prospects. *arXiv preprint arXiv:2606.26513*, 2026. URL https://arxiv.org/abs/2606.26513.

Megan Taylor Tillman, Blakesley Burkhart, Chiara M. F. Mingarelli, J. Andrew Casey-Clyde, Lars Hernquist, Sownak Bose, Enikő Regős, and César Hernández-Aguayo. Implications of the nanoHertz gravitational-wave background for galactic feedback and massive black hole growth. *arXiv preprint arXiv:2602.15938*, 2026. URL https://arxiv.org/abs/2602.15938.

Francisco Villaescusa-Navarro, Daniel Anglés-Alcázar, Shy Genel, David N Spergel, Rachel S. Somerville, Romeel Dave, Annalisa Pillepich, Lars Hernquist, Dylan Nelson, Paul Torrey, et al. The camels project: Cosmology and astrophysics with machine-learning simulations. *The Astrophysical Journal*, 915(1):71, 2021.

Hanyang Wang, Yimo Cai, Weiliang Chen, Jiawei Chi, Haowen Sun, Qiyu Dai, Yi-Hsin Hung, Xingzhuo Guo, Jinshan Ren, Runmao Yao, Ziwei Liu, Mingsheng Long, Yueqi Duan, Jun Gao, Jiangran Lyu, Fangfu Liu, and Jialong Wu. Code as Worlds: Agentic discovery of executable world representations for physical reasoning. *arXiv preprint arXiv:2608.27549*, 2026a. URL https://arxiv.org/abs/2608.27549.

Mengru Wang, Junfeng Fang, Shuofei Qiao, Zhenqian Xu, Haoming Xu, Haoxiong Wang, Shumin Deng, Linyi Yang, Zhixiang Cui, Xin Xu, et al. Mechanist: Ai as a scientific instrument for discovering the mechanisms of intelligence. *arXiv preprint arXiv:2608.12036*, 2026b.

Xingyao Wang, Simon Rosenberg, Juan Michelini, Calvin Smith, Hoang Tran, Engel Nyst, Rohit Malhotra, Xuhui Zhou, Valerie Chen, Robert Brennan, and Graham Neubig. The openhands software agent sdk: A composable and extensible foundation for production agents. *arXiv preprint arXiv:2511.03690*, 2025a.

Yongji Wang, Ching-Yao Lai, David J. Prior, and Charlie Cowen-Breen. Deep learning the flow law of Antarctic ice shelves. *Science*, 387(6739):1219–1224, 2025b. doi: 10.1126/science.adp3300.

Yuru Wang, Lejun Cheng, Yuxin Zuo, Sihang Zeng, Bingxiang He, Che Jiang, Junlin Yang, Yuchong Wang, Kaikai Zhao, Weifeng Huang, et al. Naturebench: Can coding agents match the published sota of nature-family papers? *arXiv preprint arXiv:2606.24530*, 2026c.

Matt L. Wiemann, Lindsay M. Smith, Peter Melchior, Siddharth Mishra-Sharma, Andrew Gordon Wilson, Pavel Izmailov, and Carolina Cuesta-Lázaro. DiscoverPhysics: Benchmarking LLMs for out-of-the-box scientific thinking. *arXiv preprint arXiv:2605.26087*, 2026. URL https://arxiv.org/abs/2605.26087.

Zhipeng Xu, Jiahao Lu, Yining Zheng, Yuxin Wang, and Xipeng Qiu. Swe-bench science: Can coding agents resolve engineering tasks in science? *arXiv preprint arXiv:2608.19799*, 2026.

Yutaro Yamada, Robert Tjarko Lange, Cong Lu, Shengran Hu, Chris Lu, Jakob Foerster, Jeff Clune, and David Ha. The ai scientist-v2: Workshop-level automated scientific discovery via agentic tree search. *arXiv preprint arXiv:2504.08066*, 2025.

Tom Zahavy. Position: LLMs can't jump. In *Forty-third International Conference on Machine Learning Position Paper Track*, 2026. URL https://openreview.net/forum?id=klU4737opt.

Mengdi Zhao, Ning Wang, Xinrui Jiang, Xiaoyang Ma, Haixin Ma, Gan He, Kai Du, Lei Ma, and Tiejun Huang. An integrative data-driven model simulating C. elegans brain, body and environment interactions. *Nature Computational Science*, 4:978–990, 2024. doi: 10.1038/s43588-024-00738-w.

Qiyuan Zhao and Brett M. Savoie. Algorithmic explorations of unimolecular and bimolecular reaction spaces. *Angewandte Chemie International Edition*, 61(46), October 2022. ISSN 1521-3773. doi: 10.1002/anie.202210693. URL http://dx.doi.org/10.1002/anie.202210693.

Tianshi Zheng, Kelvin Kiu-Wai Tam, Newt Hue-Nam K. Nguyen, Baixuan Xu, Zhaowei Wang, Jiayang Cheng, Hong Ting Tsang, Weiqi Wang, Jiaxin Bai, Tianqing Fang, Yangqiu Song, Ginny Y. Wong, and Simon See. NewtonBench: Benchmarking Generalizable Scientific Law Discovery in LLM Agents. *arXiv preprint arXiv:2510.07172*, 2025. URL https://arxiv.org/abs/2510.07172.

Faxin Zhou, Siyuan Zhou, Yuhang Long, Adeen Flinker, and Chunming Lu. Hierarchical linguistic predictions and cross-level information updating during narrative comprehension. *Communications Biology*, 9:107, 2026. doi: 10.1038/s42003-025-09377-x.

Mingchen Zhuge, Changsheng Zhao, Dylan Ashley, Wenyi Wang, Dmitrii Khizbullin, Yunyang Xiong, Zechun Liu, Ernie Chang, Raghuraman Krishnamoorthi, Yuandong Tian, et al. Agent-as-a-judge: Evaluate agents with agents. *arXiv preprint arXiv:2410.10934*, 2024.

A TASK SPECIFICATION TEMPLATE

Task specification	
Context	You are a systems neuroscientist studying {problem_description}. {scientific_background} {observations_block}
Initial observations	However, it is unclear {initial_observations}.
Task	Your task is to discover the mechanism that explains this phenomenon. {mechanism_requirements} To discover this, you interrogate {environment_interface} through controlled numerical experiments designed to identify {experimental_objectives}. The structure of your predictive form needs to follow from the mechanism you uncover: for each ingredient, explain why it is included, what it traces in the circuit, and how it shapes the neuron's activity.
Evaluation	Every claim in your discovered mechanism needs to be backed by data you actually collected. {predictive_accuracy_evaluation} Your experimental findings will also be reviewed by an independent researcher who has access to the same simulator, to assess whether the mechanism you discover can yield new insights.
Deliverables	You need to produce three deliverables. If any of them is missing or does not follow the requirements below, your solution receives a score of 0. Deliverable 1: Write your discovered mechanism to {output_dir}/mechanisms/mechanism.md. State the mechanism as the causal chain from the coarse action state of the population to the activity of one neuron; its executable form, with every ingredient and fitted constant defined and the physical process each one traces; why its inputs jointly determine the neuron's activity; the concrete evidence from your experiments that supports the mechanism and its form, including the evidence that let you reject the competing explanations you considered; and the insights about how population activity organizes the activity of individual neurons that follow from what you found. Deliverable 2: Write the executable form of your mechanism to {output_dir}/mechanisms/mechanism.py. {executable_interface_and_requirements} Deliverable 3: Keep an experiment log at {output_dir}/mechanisms/experiment.log. Treat it as your laboratory notebook: it is the primary reference for the researchers who will continue this work, and you choose how to organize it. Record what they will need to follow and extend your investigation: the questions you pursued, the simulator configurations and analyses you ran, what you observed, the conclusions you drew, and the approaches that did not work and why. Give particular room to the findings that surprised you or that you found interesting along the way, whether or not they entered your final mechanism; these are often the most valuable part of a notebook for the next researcher. Move the useful evidence behind the log's findings — the figures, analysis outputs, and small data files your experiments produced in {tmp_dir}, and the observations from {output_dir}/observations/ that your final claims build on — into {output_dir}/mechanisms/ so that they survive for these researchers: everything outside {output_dir}/mechanisms/ is deleted when the session ends.

Task specification (continued)	
Saved evidence	You may additionally place supporting evidence in {output_dir}/mechanisms/ and reference it from the files above: figures, and any analysis outputs that back your claims, such as fitted-model diagnostics, ablation results, or small data tables worth preserving. Only this folder is kept when the session ends; everything else in the workspace, including {output_dir}/observations/, {tmp_dir} and the simulation outputs you generated, is discarded. The independent researcher reviewing your mechanism sees exactly this folder. Everything under {output_dir}/mechanisms/ combined is limited to 1 GB: if the folder exceeds this limit, your solution receives a score of 0. It is for final deliverables and their evidence, not scratch space; keep drafts and intermediate files in {tmp_dir}.
Budget	You have a time budget of {time_budget}. You can use {tmp_dir} for notes and intermediate files.

Table 3: **Task specification template for the scientist agent (neuroscience).**

B Guidance Analysis Prompts

B.1 Goal Prompt

Problem	**Goal**
Neural dynamics	A good mechanism is not only accurate but also extrapolatable, and it can even open up more interesting new research directions. Through such a mechanism people come to understand how the nervous system of C. elegans organizes its activity into a hierarchical locomotion structure, down to how the coarse action state shapes a single neuron.
ARC tokamak	A good mechanism is not only accurate but also extrapolatable, and it can even open up more interesting new research directions. Through such a mechanism people come to understand how the transport a model returns sets the profiles an ARC plasma settles into and the fusion power it produces, down to why two models that leave the plasma with the same energy confinement disagree by a factor of two in power.

Table 4: **Goal guidance used in the `+goal` condition.**

C AGENTIC JUDGE PROMPTS

C.1 SCIENTIFIC CONSTRAINT EVALUATION

Scientific constraint judge instruction	
Context	You are a systems neuroscientist reviewing another neuroscientist's discovered mechanism for {problem_description}. The neuroscientist interrogated {environment_interface} through controlled simulations to discover that mechanism.
Verdict instruction	Your task is to judge whether this mechanism is consistent with commonly accepted assumptions about C. elegans. The criteria you need to check are listed below: Commonly accepted assumptions (Scientific Constraints) According to the discovered mechanism: {scientific_constraints_criteria} Each of these criteria is a general assumption, not something the neuroscientist is expected to prove. Judge each one by reading the mechanism described in mechanism.md together with its executable form. A criterion should be marked as false if the mechanism contradicts the assumption, and true if it does not. Focus on what the mechanism actually does, not on the names given to its quantities; calling a quantity something is not enough to show that it behaves that way. You also need to mark the criterion as false if the experimental results violate the assumption and the neuroscientist neither recognizes those results as erroneous nor accounts for the violation in the final mechanism.

Output	You are expected to provide the following:
	Write {output_dir}/{judge_output} as a JSON file with exactly the structure below, with one entry for each criterion ID listed above and nothing else:
	`{`
	`  "scientific_constraints": {`
	`  "SC1": {"verdict": true, "reason": "..."}`
	`  }`
	`}`
	Each verdict should be either true or false, with no other values allowed. Each reason should be no more than two sentences and should identify the part of the submission that supports the verdict. Keep {output_dir}/judge_notes.md alongside the JSON file and use it to record what you reviewed for each criterion. Save both files instead of printing their contents, then stop.

Table 5: **Example instructions for scientific constraint tests (neuroscience).**

C.2 Judge Instructions for Insights Derived from the Mechanism

Judge instructions for insights derived from the mechanism	
Context	You are a systems neuroscientist reviewing another neuroscientist's discovered mechanism for {problem_description}. The neuroscientist interrogated {environment_interface} through controlled simulations to discover that mechanism.
Verdict instruction	A good mechanism is not only accurate but also far more interpretable and extrapolatable, and it can be used to derive insights. Your task is strictly to evaluate whether this discovered mechanism is able to derive the interpretable scientific insights listed below. Scientific Insights questions {insights_criteria} To derive each insight, you need to start from the discovered mechanism and use {environment_interface} to check whether the insight can be derived from it. A Yes verdict only occurs when the discovered mechanism is able to help you find the evidence needed to provide a concrete and complete explanation of each insight question. You should give a No verdict for an insight question if the answer is based on your prior knowledge or is not fully supported by the discovered mechanism. Any assertion or conclusion about the insights in mechanism/ is never, by itself, grounds for a Yes verdict; verify that the mechanism derives the same conclusion. If you are not able to provide sufficient evidence showing that this mechanism is able to derive the insight, the verdict should be No. You must not modify or extend the discovered mechanism. If the insight question cannot be completely derived from the discovered mechanism, the verdict should be No. An answer based only on your own knowledge of the animal or on the simulator, but not supported by the discovered mechanism, should also be No.

Output	You are expected to provide the following:
	Write {output_dir}/{judge_output} as a JSON file with exactly the structure below, with one entry for each criterion ID listed above and nothing else:
	`{`
	`  "insights": {`
	`  "I1": {"verdict": "Yes", "reason": "..."}`
	`  }`
	`}`
	Each verdict should be either Yes or No, with no other values allowed. For each insight criterion, when you give a Yes verdict, you must provide complete evidence and reasoning explaining why the discovered mechanism can derive the insight asked about in the insight question; otherwise, the verdict should be No.
	Keep {output_dir}/judge_notes.md alongside the JSON file and use it to record what you reviewed and any commands you ran. Save both files instead of printing their contents, then stop.

Table 6: **Example judge instructions for insights derived from the mechanism (neuroscience).** The judge returns `Yes` only with complete evidence showing how the insight can be derived from the discovered mechanism.

C.3 Judge Instructions for Insights Discovered During Experimentation

Judge instructions for insights discovered during experimentation	
Context	You are a systems neuroscientist reviewing another neuroscientist's discovered mechanism for {problem_description}. The neuroscientist interrogated {environment_interface} through controlled simulations to discover that mechanism.
Verdict instruction	Your task is to evaluate whether the scientific insight captured by each question below was actually discovered by the neuroscientist while uncovering the mechanism. Consider the questions one at a time. For each question, determine whether the neuroscientist obtained the corresponding insight and whether the available evidence is strong enough to support it. The scientific insight questions are listed below: Scientific Insights questions {insights_criteria} When you judge each insight question, do not modify or extend the discovered mechanism, and do not look for a different one. The neuroscientist needs to have explicitly stated the insight and run an experiment specifically to test it. Experiments run for other purposes, such as building, tuning, or checking the executable form of the mechanism, do not count as tests even if their results bear on the question. For questions asking why something happens, showing that it happens is not enough; the explanation itself needs to have been tested. If the neuroscientist did not discover the insight while uncovering the mechanism, or if the discovered mechanism and experiment record do not provide sufficient evidence that the insight was obtained during this process, the answer should be No. If the insight needs to be inferred from their numbers or their argument needs to be completed for them, the answer should be No. An answer based only on your own knowledge of the animal, but not supported by a finding the neuroscientist recorded, should also be No. If the mechanism clearly conflicts with that insight, the answer should also be No. A multi-part question is answered only if every part is answered. Each underlying insight can earn only one Yes verdict. Answers based on the same explanation count as the same insight regardless of the quantities or experiments involved, so mark as Yes only the question that explanation answers most directly and the rest No. When in doubt, the answer should be No.

Output	You are expected to provide the following:
	Write {output_dir}/{judge_output} as a JSON file with exactly the structure below, with one entry for each criterion ID listed above and nothing else:
	`{`
	`  "insights": {`
	`  "I1": {"verdict": "Yes", "reason": "..."}`
	`  }`
	`}`
	Each verdict should be either Yes or No, with no other values allowed. Each reason should be no more than two sentences. A Yes verdict has to quote the neuroscientist's own statement of the insight and the measurement from the experiment that tested it, and has to say what the neuroscientist ran that experiment for; otherwise, the verdict is No. Keep {output_dir}/judge_notes.md alongside the JSON file and use it to record what you reviewed for each criterion. Save both files instead of printing their contents, then stop.

Table 7: **Example judge instructions for insights discovered during experimentation (neuroscience).**

D Additional Results

D.1 Does higher accuracy imply more insights?

We compare agents' PA and insight rankings within each problem using Kendall's τ_b (Kendall, 1945) (Table 8). We estimate 95% confidence intervals from 20,000 bootstrap samples of whole problems (Efron, 1979). The mean correlation is positive but limited ($\tau_b = 0.20$, adjusted $p = 0.008$). After excluding pairs tied on either metric, the higher-PA agent scores lower on insights in 39.9% of comparisons on average across problems. The reversal rate remains similar when we use either judge's scores separately. Predictive accuracy alone therefore does not reliably indicate how much scientific insight a mechanism supports.

Judge	τ_b	**Reversals (%)**	p
Both judges (average)	0.20 [0.07, 0.32]	39.9 [33.2, 46.5]	0.008
Claude Opus 4.8	0.15 [0.03, 0.26]	41.8 [35.3, 48.4]	0.054
GPT 5.6 Sol	0.15 [0.01, 0.29]	42.8 [34.3, 51.9]	0.054

Table 8: **PA and insight rankings.** Brackets show 95% bootstrap confidence intervals; p-values test no association using 100,000 within-problem permutations and Holm correction across the three rows (Phipson & Smyth, 2010; Holm, 1979).

E Neuroscience

In Neuroscience, we construct 6 Caenorhabditis elegans (C. elegans) problems and 4 human brain problems.

E.1 Caenorhabditis elegans

Premotor states. *C. elegans* is a small roundworm that switches between forward and backward crawling. Two groups of premotor neurons help control these movements, one associated with forward crawling and the other with backward crawling. Neurons within each group become active together, but the groups alternate at irregular times (Morrison & Young, 2025). The task asks what keeps each group coordinated, how the groups compete, and what makes one movement state give way to the other.

Zigzag foraging. *C. elegans* searches for food by swinging its head while a wave of bends travels down its body. Its nervous system coordinates these movements so that it can change direction while continuing to move forward (Zhao et al., 2024). Because each head swing changes where the worm samples the odor, its own movement changes the signal it is trying to follow. This coupling between sensing and movement defines the problem we pose: how does the changing odor input pass through the neural circuit to steer the head while maintaining the body's rhythm?

Biological networks. *C. elegans* can learn whether an odor is associated with food, and this experience changes how it navigates when it encounters the odor again. Learning affects both the gradual steering of its path and the sharp turns used to change direction (Chen et al., 2025a). The problem is to explain where and how the circuit processes the same odor differently after learning.

Neural dynamics. During a period of forward or backward crawling, *C. elegans* repeatedly bends its body to generate movement. Activity shared across many neurons tracks the slower sequence of actions, while individual neurons also show faster activity associated with body bends (Fieseler et al., 2025). The task connects these two levels of activity by asking what information about the overall action reaches an individual neuron and how it shapes that neuron's faster rhythm.

Navigation strategies. To approach an odor source, *C. elegans* combines small changes in direction with bouts of sharp turns. Worms often finish a bout of turns facing the odor source, although they are not known to keep an internal map of its location (Chen et al., 2026). Our task asks for an explanation of this sequence of actions from the history of odor input.

Phase memory. *C. elegans* can use an odor cue to guide a turn after the cue has disappeared and the animal has spent some time crawling backward. Stimulating an odor-sensing neuron when its head is bent to one side biases the later turn toward that side, even when the intervening backward movement varies in duration (Dunn et al., 2025).

E.2 Human Brain

Sensory integration. Seeing and hearing the same event can help a person locate it more accurately than either sense alone. Yet the combined brain response is often no larger than the sum of the responses to sight and sound separately (Buhmann et al., 2026). This mismatch motivates our task: explaining how the benefit of combining the two senses appears in cortical activity.

Visual predictive processing. The brain uses the surrounding scene to anticipate what belongs at a particular location. During ordinary viewing, the visual cortex responds less strongly when the content there is easier to anticipate (Scheurer & Heilbron, 2026). We focus this task on what is anticipated, from edges and textures to the identity of an object.

Predictive processing of language. While listening to a story, people anticipate both upcoming words and the meaning of the next sentence. Responses to individual words depend on how surprising they are, while different brain regions anticipate language over different spans of time (Zhou et al., 2026). Our task concerns how these expectations interact as new words arrive.

Neural representation. Different parts of the visual cortex respond more strongly to different kinds of objects, forming broad preference maps associated with properties such as whether an object is alive and how large it is in the real world. These properties often vary together with others, such as whether an object can move or be held, making it difficult to identify what actually organizes the responses (Stoinski et al., 2026). The task asks which combinations of object properties explain the response at a cortical location, with these preference maps as the starting point.

E.3 Environment Interfaces

c302. c302 (Gleeson et al., 2018) builds NeuroML models of *C. elegans* neurons and muscles from measured connectivity, with different levels of biophysical detail. The six *C. elegans* problems share a CLI through which the scientist agent can inspect the connectome, choose the neurons and muscles to include, and simulate their activity. Experiments can inject current into selected neurons, remove connections, or change synaptic weights and cellular parameters; the resulting activity traces record the effects of these interventions on the circuit.

TRIBE v2. For the four human brain problems, TRIBE v2 (d'Ascoli et al., 2026) predicts functional magnetic resonance imaging (fMRI) responses to stimuli constructed by the scientist agent. The agent submits video, audio, or text files through a shared CLI, with text converted to speech before processing. The model returns predicted activity at 20,484 locations across the cortical surface over the duration of the stimulus, and the interface also provides the cortical geometry for locating these responses.

F Geophysics

Ice shelf flow laws. Ice shelves are floating extensions of ice sheets that slow the flow of inland ice into the ocean. How quickly a shelf deforms depends on its flow law, which relates the forces acting on the ice to how it changes shape. The shallow-shelf approximation (SSA) describes the shelf's horizontal motion using two-dimensional equations that balance gravity against the ice's resistance to deformation. However, flow laws calibrated on laboratory ice do not fully describe the behavior of real Antarctic ice shelves (Wang et al., 2025b).

Catchment properties. A catchment is the area of land whose rain and snowmelt drain into the same stream or river. Properties such as how much water its soil holds and how quickly its groundwater drains determine when and how much water reaches the stream. Yet different values of these properties can produce similar streamflow, making it difficult to determine the properties from measurements that also contain noise.

F.1 Environment Interfaces

FEniCSx. FEniCSx is an open-source computing framework for solving partial differential equations with the finite element method (Baratta et al., 2023). In the ice shelf flow laws problem, the scientist agent writes and runs its own solver using our FEniCSx installation and instructions for coupling the two-dimensional shallow-shelf approximation (SSA) momentum equations with mass conservation. The equations, viscosity, boundary conditions, and shelf geometry can be changed to generate velocity and thickness fields under different physical assumptions.

HBV. We construct the ground truth for catchment properties with HBV (Bergström, 1992), a conceptual model of water storage and movement through snow, soil, and groundwater. We choose known values for its 15 parameters and drive the model with daily precipitation, air temperature, and potential evapotranspiration from CAMELS-US (Newman et al., 2015). We then add measurement noise to the simulated streamflow to create the observed records, keeping the generating parameters and noise-free streamflow as ground truth. The scientist agent accesses the same model through a CLI, where it can select catchments, parameter values, and initial storage states and inspect the resulting streamflow, internal storages, and fluxes.

G Plasma Physics

ARC tokamak. ARC is a proposed fusion power plant that uses magnetic fields to confine extremely hot plasma. Its fusion power depends on the plasma's temperature and density, which are shaped by the movement of heat and particles through the plasma. Different descriptions of this transport can give substantially different fusion powers even when their overall measures of energy confinement are similar (Howard et al., 2026).

Negative triangularity core-only. A tokamak's plasma usually has a D-shaped cross section; reversing this shape is called negative triangularity. Although the reversed shape was expected to make the plasma less stable, experiments found that it can retain heat better with less turbulence, and that this advantage weakens when collisions between particles become more frequent (Thome et al., 2026).

Negative triangularity edge. In a conventional tokamak, sufficient heating can create a narrow region near the plasma boundary where heat escapes much more slowly, called an edge transport barrier. A strongly reversed D-shaped plasma can remain without this barrier even at heating powers well above the usual threshold (Thome et al., 2026).

G.1 Environment Interfaces

TORAX. The ARC tokamak and negative triangularity core-only problems share an interface to TORAX (Citrin et al., 2024), a differentiable simulator of tokamak plasma transport. TORAX solves coupled one-dimensional equations for ion and electron heat transport, particle transport, and current diffusion. Through the CLI, the scientist agent chooses the magnetic geometry, transport model, heating sources, and boundary conditions and records the resulting evolution of plasma profiles and global quantities. For the core-only problem, we also supply a library of matched magnetic equilibria with different triangularities so that plasma shape can be varied across runs.

TORAX, TokaMaker, and GPEC. The negative triangularity edge environment combines TORAX with TokaMaker (Hansen et al., 2024) and the Generalized Perturbed Equilibrium Code (GPEC) (Glasser, 2016). TokaMaker solves the free-boundary Grad–Shafranov equation for the magnetic equilibrium, and GPEC calculates its ideal ballooning stability boundaries. Our coupled

model uses these stability calculations to determine the edge conditions for TORAX transport simulations. The scientist agent can vary the plasma shape and operating conditions through the CLI, run the equilibrium, stability, and transport calculations together or individually, and inspect the resulting equilibria and plasma profiles.

H ASTROPHYSICS

Single galaxy. The average matter density of the universe is usually estimated from the distribution of large numbers of galaxies. Yet a neural network can recover this quantity from the internal properties of just one simulated galaxy. The connection persists even when stars and black holes release very different amounts of energy into their surroundings, changing how much gas a galaxy retains (Liao et al., 2026). This raises the question of why a single galaxy retains information about the universe in which it formed.

Galaxy environment. A galaxy forms inside a dark matter halo, an extended concentration of dark matter held together by gravity. The ordinary matter in its gas and stars is called baryonic matter, and the amount a halo retains can depend on its neighbors. For low-mass halos of the same mass, a denser neighborhood reduces the baryonic content in the SIMBA galaxy-formation model but increases it in Swift-EAGLE (Sims et al., 2026). Understanding these opposite trends requires identifying the physical processes that produce them and their effects on both the total baryon content and the gas surrounding the galaxy.

Gravitational wave. Pairs of supermassive black holes emit gravitational waves, ripples in spacetime, as they orbit each other. Their combined emission is expected to account for the low-frequency background signal measured by astronomers. However, galaxy-formation models that reproduce observed galaxies give a background amplitude roughly half the measured value (Tillman et al., 2026). Explaining this mismatch requires understanding how star formation, black hole growth, and their energy output change the growth and abundance of black holes, and how those changes set the strength of the background.

Milky Way. A dark matter density profile describes how densely dark matter is packed at different distances from the center of a galaxy. Gas collecting near the center and gas expelled by stars or black holes change the gravitational forces acting on the surrounding dark matter. In the Milky Way-mass systems examined by Garcia et al. (2025), the central dark matter is denser and varies more across halos than in calculations with gravity alone. Explaining this change in the dark matter density profile requires distinguishing the effects of gas heating and expulsion from differences in how individual halos grew.

Baryonic feedback. Supernova explosions and gas falling into black holes release energy that heats or expels gas from galaxies. This process, called feedback, regulates the supply of gas available to form stars and feed black holes. Galaxies in equally massive dark matter halos can nevertheless end up with very different stellar and black hole masses (Garcia et al., 2025). These differences motivate understanding how feedback and halo growth histories affect both masses.

WDM merge tree. In warm dark matter (WDM) models, lighter particles travel farther in the early universe, smoothing out more of the small density fluctuations from which halos form. Halos subsequently grow through mergers, and a merger tree records which smaller halos combine and when. The branching pattern and timing of these mergers contain information about the WDM particle mass even without measurements of the masses or gas content of the individual objects (Leisher et al., 2025). This raises the question of which features of a Milky Way-mass halo's merger tree preserve this information and how they arise from the influence of the particle mass on early structure formation.

WDM population. Subhalos are gravitationally bound concentrations of matter within larger dark matter halos, and those containing stars host galaxies. Across simulated universes, the distribution of gas among subhalos reveals the WDM particle mass more reliably than other properties, even though the properties of a single galaxy do not reliably reveal that mass (Costanza et al., 2025).

This connection is not straightforward: dark matter affects the gas through gravity, while energy released by stars and black holes also changes how much gas subhalos retain. Explaining this connection requires understanding why the gas preserves information about particle mass and why that information becomes apparent across a population.

H.1 Environment Interfaces

CAMELS. CAMELS (Villaescusa-Navarro et al., 2021) contains precomputed galaxy-formation simulations with different cosmological parameters and models of stellar and black hole feedback. We draw on suites that vary parameters individually or jointly, as well as runs with matched initial conditions across galaxy-formation models. For the single galaxy, galaxy environment, and gravitational wave problems, the scientist agent selects and compares simulations through a common CLI that retrieves their parameters, halo and galaxy catalogs, particle snapshots, and available matter power spectra.

The DREAMS Project. The DREAMS Project (Rose et al., 2025) uses cosmological simulations and high-resolution simulations of Milky Way-mass halos to study the effects of dark matter physics and astrophysical feedback on galaxy formation. For the Milky Way and baryonic feedback problems, we use cold dark matter simulations with hydrodynamic and gravity-only versions of the same initial conditions. The WDM merge tree and WDM population problems use warm dark matter suites that vary the particle mass and feedback parameters. A shared CLI retrieves simulation parameters and halo catalogs, with particle snapshots or merger trees available in the relevant suites.

I Computer Science

Stated belief shift. After fine-tuning on political texts, a language model may answer political and social questions differently, expressing views closer to those in the training material (Geng et al., 2025b). This problem asks how the texts change the model's stated beliefs and which features of the passages drive that change.

Subliminal learning. A language model can acquire an animal preference by training on number sequences that never mention any animal, a phenomenon called subliminal learning (Cloud et al., 2025).

Circuit discovery. A circuit is a set of connections between a language model's components that carries out a particular task. Finding these connections requires identifying which parts of the computation cause the model's answers, rather than simply observing that it answers correctly (Mueller et al., 2025).

I.1 Environment Interfaces

Open-weight language models. Stated belief shift and subliminal learning share an interface for LoRA (Hu et al., 2022) fine-tuning of open-weight language models on subsets of the supplied training data. The scientist agent can remove selected examples and repeat training to test whether they are responsible for the observed change in beliefs or preferences. Circuit discovery uses TransformerLens (Nanda & Bloom, 2022) to test candidate circuits by replacing the activations of selected model components and measuring the change in the model's output.

J Chemistry

Reaction discovery. The reactants and products of a chemical reaction do not reveal the sequence of bond changes or the intermediate molecules formed along the way. A reaction mechanism explains this sequence and the energy barrier that needs to be crossed at each step. Although mechanisms can be resolved for individual reactions, the cost of searching for reaction paths limits how readily this can be done for many reactions (Zhao & Savoie, 2022).

J.1 Environment Interfaces

AIMNet2-rxn. AIMNet2-rxn (Anstine et al., 2025) is a machine-learned potential for organic reactions, trained on density functional theory (DFT) calculations at the ωB97M/def2-TZVPP level for neutral, closed-shell structures containing H, C, N, and O. It returns potential energies and atomic forces, which our CLI uses to optimize geometries and search for transition states along candidate reaction paths. Vibrational analysis checks whether a candidate is a saddle point, and intrinsic reaction coordinate calculations check whether it connects the intended reactants and products. The scientist agent uses the resulting paths and energy differences to identify elementary steps and calculate activation barriers and reaction energies.

Sanity check. We perform a sanity check to verify that the catalyst ground truth we compute with AIMNet2-rxn is physically reasonable. We compare AIMNet2-rxn with published DFT calculations for three organocatalytic reactions: the proline-catalyzed Hajos–Parrish–Eder–Sauer–Wiechert cyclization (Bahmanyar & Houk, 2001), the proline-catalyzed aldol reaction of cyclohexanone and benzaldehyde (Armstrong et al., 2014), and the TBD-catalyzed ring-opening of lactide by methanol (Chuma et al., 2008). Starting from the published geometries, we check relative energies, optimized transition-state structures, and the ordering of competing reaction paths. Of 30 reference transition states, 29 remain first-order saddle points after optimization, with heavy-atom root-mean-square deviations of 0.07–0.48 Å from the DFT structures. For the 16 aldol transition states, relative energies differ from the reference by 1.85 kcal/mol on average and 3.51 kcal/mol at most; their energy ranking has a Spearman correlation of 0.79 with DFT and identifies the same lowest-energy transition state. For lactide, AIMNet2-rxn also favors the hydrogen-bonded route over the covalent route, with an energy difference of 13.9 kcal/mol compared with 11.7 kcal/mol in the reference. Agreement is less close for some energy differences: barriers measured from reactant minima are 1.9–8.6 kcal/mol higher than the references, while a 0.18 kcal/mol separation between two Hajos–Parrish transition states becomes 0.01 kcal/mol. These comparisons use different DFT approximations, so the discrepancies can reflect both the reference methods and the learned potential and cannot be attributed entirely to AIMNet2-rxn.