Telling AI to think like Socrates: new voice, same answers, bigger bill

A paper found that prompting an AI to reason like Socrates made it much better at chemistry. I tried the same seven philosopher prompts on 15 bugs I'd already fixed, across four models. The answers barely moved. The bill went up every time.

SEP 27, 2026 · REVISION 2 · LAB.WORTHINGTON.CLOUD

Telling AI to think like Socrates: new voice, same answers, bigger bill

A paper found that prompting an AI to reason like Socrates made it much better at chemistry. I tried the same seven philosopher prompts on 15 bugs I'd already fixed, across four models. The answers barely moved. The bill went up every time.

004 Field Report Sep 27, 2026 Revision 2

Instrument and questions: Rick Worthington

Analysis and prose: Claude Opus 5.5, under Rick Worthington's editorial direction

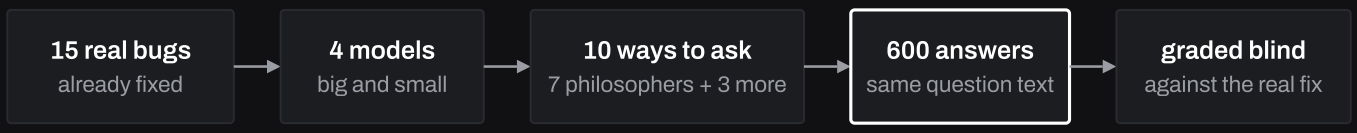
A paper found Socratic prompting could make AI much better at chemistry. So I tried it on my own bugs.

In June a team at Argonne National Laboratory published a neat experiment.[1] They gave AI models a system prompt written in the style of a famous philosopher (Socrates, Aristotle, Descartes, Kant, Hume, Hegel or Plato) and then asked them chemistry questions. The standout was Socrates. On one model, under the strictest scoring, it lifted the score from 51% to 73%. The gains were uneven, though, and the authors say plainly that no single philosopher won on every model.

That's the same idea behind the most common prompt advice out there... start with "act like an expert." So does telling a model who to be give it that person's thinking, or just their voice?

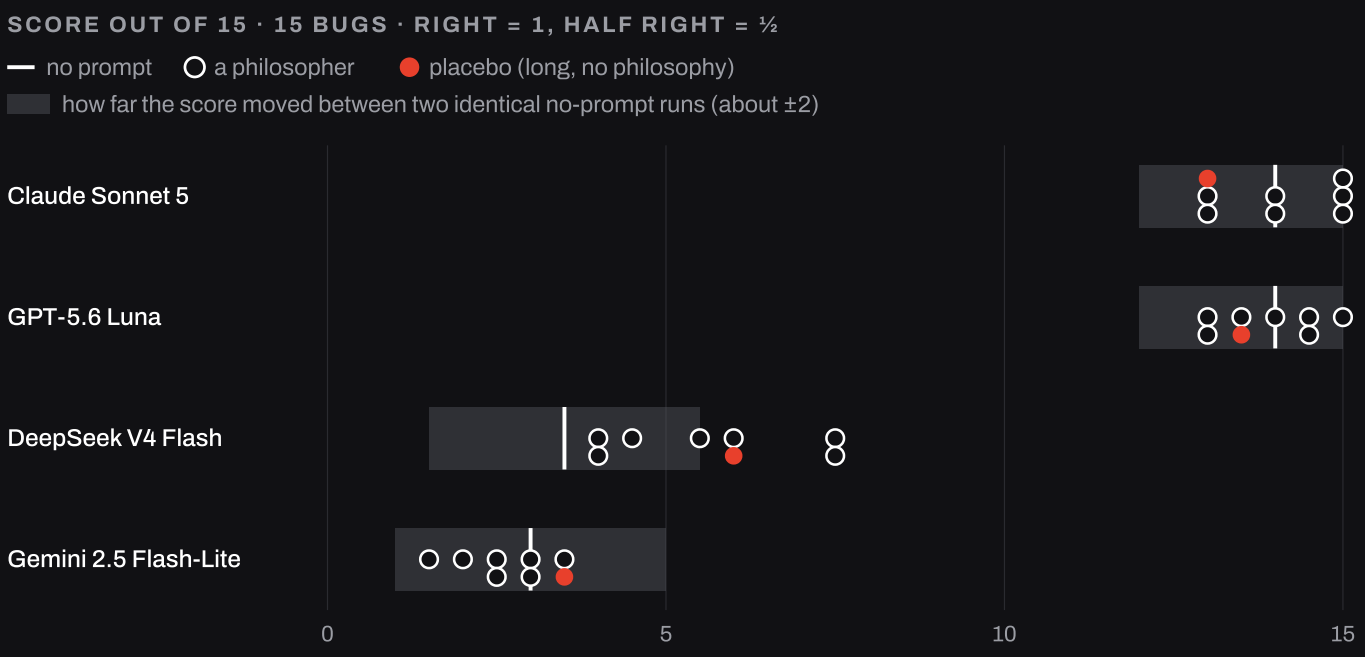
I wanted to know if it held up outside chemistry. So I took 15 bugs I'd already fixed in my own setup, where I know the right answer because I lived through it. I sent each one to four models, asked ten different ways, and graded all 600 answers blind.

THE SETUP



Only the system prompt changed. The bug, the question and the settings were identical every time.

The ten ways were: no system prompt (twice, to see how much a model disagrees with itself), the paper's seven philosopher prompts word for word, and a placebo. The placebo is a prompt the same length as a philosopher's, full of generic careful-reasoning advice, with no philosophy in it. If Socrates really helps, he should beat the placebo.



Here's how to read that. Each row is a model. The solid line is its score with no prompt, and the gray band is how far that score moved between two identical no-prompt runs. Circles are philosophers. The red dot is the placebo. On the two strong models, everything landed in the band.

So the scores held still. What changed was everything around them. Here's the same bug, the same model and the same settings, asked plainly and then asked as Descartes.

No system prompt

```
`avgIf(1, label_a = label_b)`
averages the constant `1` only for
rows where the labels agree.
Therefore, whenever a model has at
least one agreement, its result is
`1.0`...

Use the predicate as the value being
averaged:

avg(label_a = label_b) AS
agreement_rate
```

Graded right · diagnosis in the first sentence

85 words in all

The Descartes prompt

```
## Selected Principles

Term and Concept Hygiene + Methodic
Doubt. The key distinction is
between using a condition to select
rows and using it to produce a 0/1
value.

## Cartesian Reformulation

Does avgIf(1, label_a = label_b)
average agreement indicators over
all rows, or...

## Follow-Up Questions

1. Is avgIf being interpreted as...
```

Graded right · diagnosis after the ceremony

256 words in all · the diagnosis starts at word 94

GPT-5.6 Luna, bug P06, the opening lines of each answer, trimmed. Same model, same bug, same settings.

Both answers are right. One gets there in its first sentence. The other puts on a robe, clears its throat through three headings, and arrives at word 94.

1. On a strong model, a philosopher prompt changes the voice and raises the bill. It didn't change the answers.

   Justified by: Sonnet and Luna stayed within the variation between their two no-prompt runs under every prompt (Finding 1), and every philosopher cost more (3).

2. A terse model got better with any long prompt. The philosophy wasn't the ingredient.

   Justified by: DeepSeek improved under the placebo too, and the best philosopher beat the placebo by less than that run-to-run variation (4).

3. Before you trust "act like an expert," test it. Same questions, with and without, twice.

Justified by: the weakest model got nothing (2), and the models announced their costume in 98% of answers (5).

HOW THIS WAS MADE, AND WHAT FAILED

Claude Opus 5.5, the model that co-writes this site, built the test, wrote the problems from my notes, and graded every answer without knowing which model or prompt produced it. A second company's model re-graded one model's answers and agreed 98% of the time. Two of the six predictions written down beforehand turned out wrong, and the first pass at the math made every difference look real. All of that is in How it was tested.

PART A

The answers didn't move

Each bug was a real problem with a known fix, like a database query that reported 100% agreement for everyone, or a wait loop that started the next job while the first was still running. The model saw the symptom and the code and was asked for the root cause and the smallest fix. A grader marked each answer right, half right or wrong against the fix that actually worked.

FINDING 1 SMALL TEST, CLEAR PATTERN

On the two strong models, no prompt did better than asking plainly

> **Takeaway: Sonnet and Luna scored 13 to 15 under every prompt, against 14 with none.**

Claude Sonnet 5 scored 14 out of 15 with no prompt, both times. Under the philosophers it scored between 13 and 15. GPT-5.6 Luna was the same story, 14 plain and 13 to 15 dressed up. The placebo landed in the same place.

Is a 15 better than a 14? Not here. Between two identical no-prompt runs, each model flipped two to four answers bug by bug, a swing of about 2 points. Every philosopher on these two models stayed inside it.

> DIRECTION
>
> If a strong model already gets it right, a philosopher prompt has nowhere to take it. Don't expect one to.

FINDING 2 SMALL TEST

The weakest model got nothing, and Socrates was its worst

> **Takeaway: Gemini 2.5 Flash-Lite scored 3 plain and 1.5 to 3.5 under the rest.**

The prediction going in was that the weakest model would gain the most, since a strong one has little room left. It didn't. Gemini 2.5 Flash-Lite scored 3 out of 15 with no prompt and never moved outside the variation between its two no-prompt runs. Its worst result was Socrates, at 1.5 with nothing fully right.

The paper's biggest win was also on a model that started low. So "prompts help the struggling model" is a fair guess. It just wasn't true for this one.

> DIRECTION
>
> A prompt can't hand a model knowledge it doesn't have. If a model is missing the basics, try a better model before a better costume.

PART B

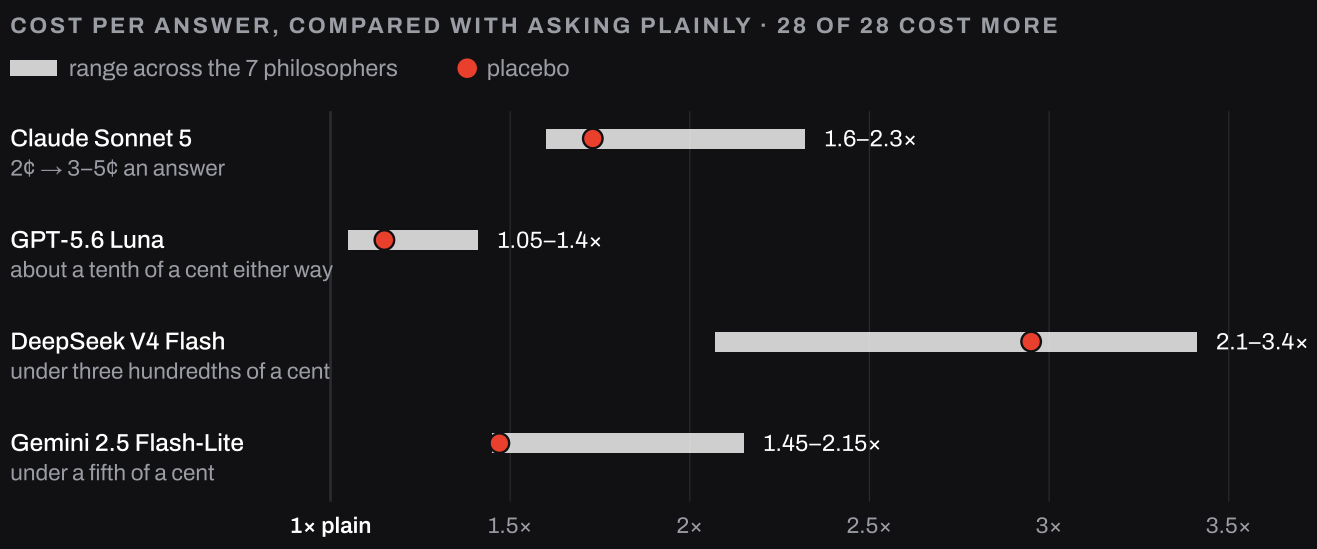
The bill did

FINDING 3 MEASURED, EVERY CASE

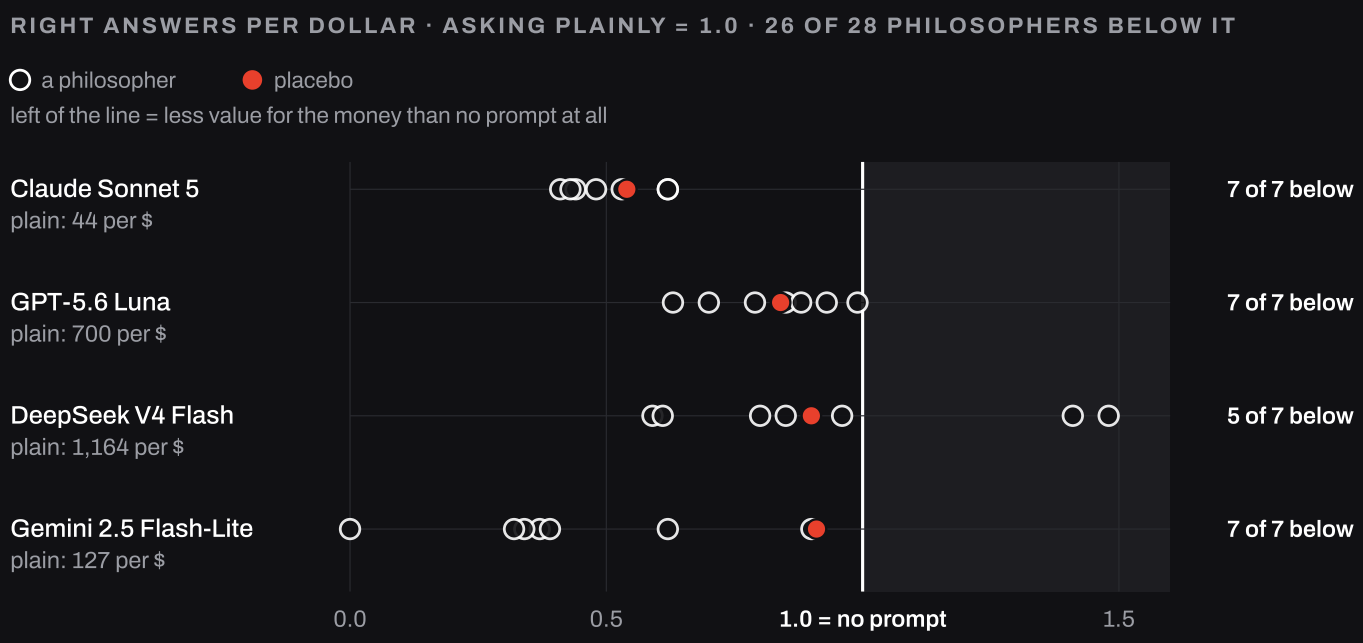
Every philosopher prompt cost more, on every model

> **Takeaway: 28 of 28 philosopher cases cost more per answer than asking plainly.**

The prompts are long, 1,000 to 2,000 words, and the models answer them at length. So you pay twice, once to send the prompt and once for the longer reply. On Sonnet an answer went from about 2 cents to between 3 and 5.



The fairest way to see it is right answers per dollar. A cheap model buys hundreds per dollar and an expensive one dozens, so the chart sets each model's no-prompt result to 1.0 and measures every prompt against it.[3]



Each model is measured against itself, so the rows can be compared. No prompt, in fully right answers per dollar: Sonnet 44, Luna 700, DeepSeek 1,164, Flash-Lite 127.

Twenty-six of the 28 philosopher cases bought fewer right answers per dollar than asking plainly. The two exceptions were Descartes and Hume on DeepSeek, the terse model in Finding 4.

The money here is small because the test was small. At the scale a business runs prompts, 1.6 to 2.3 times the cost on a model like Sonnet is not small.

> DIRECTION
>
> Every word of a system prompt is paid for on every call. Make it earn its place.

Telling a model who to be changed how it talked. It didn't change what it knew.

The costume was easy to put on. The model charged by the word to wear it.

PART C

What the prompts did change

THE THREAD

So far: the answers didn't move and the bill did. What follows is what the prompts actually changed, which is how the models talked and how long they went on.

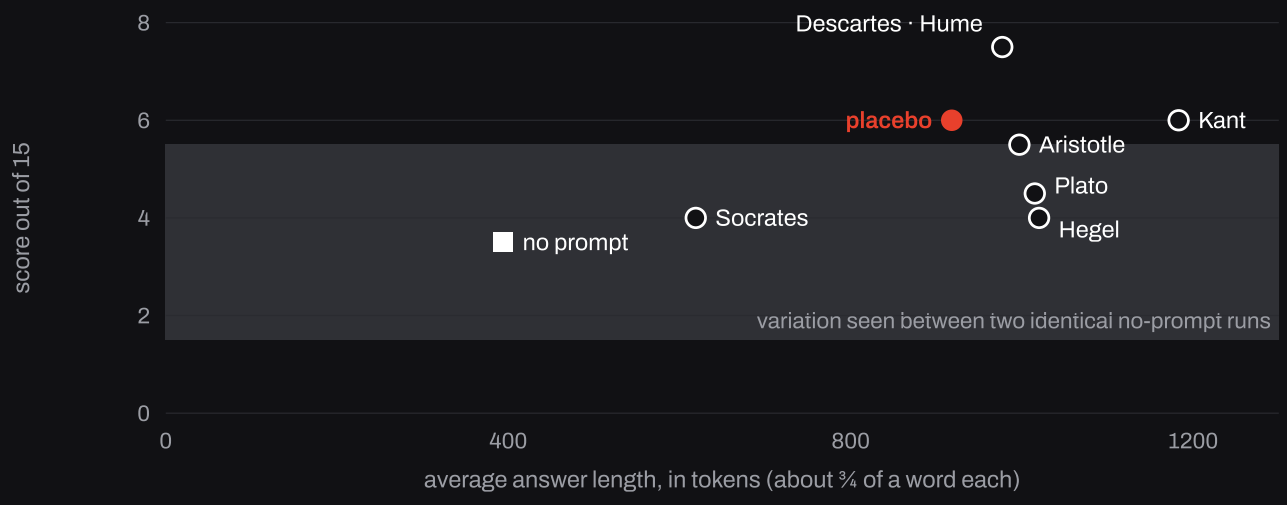
FINDING 4 SUGGESTIVE · BASELINE UNSTABLE

The terse model improved, but the placebo helped it too

> **Takeaway: DeepSeek did better under most long prompts, and which philosopher wrote it didn't matter measurably.**

DeepSeek V4 Flash answers short. With no prompt it wrote about 390 tokens and scored 3.5. Every system prompt made it write more, and most made it score higher. Descartes and Hume topped out at 7.5.

DEEPSEEK V4 FLASH · LONGER ANSWERS, HIGHER SCORES, WHOEVER WROTE THE PROMPT



But look at the red dot. The placebo, which has no philosophy in it at all, scored 6. The best philosopher beat it by 1.5, less than the 2-point variation between the two no-prompt

runs. What helped was being told to lay out its reasoning, and a long prompt does that no matter who's supposedly talking.

This is also why the placebo exists. The paper's comparison prompts were short and told the model to skip the explanation, while its philosopher prompts invited one.[2] That leaves the question of whether the gain came from Socrates or from permission to think out loud.

One honest problem. In the screen, the same plain DeepSeek scored 6.5 on these same 15 bugs, a different sample graded by a different batch. So treat this as a hint, not a result.

> DIRECTION
>
> If a cheap model answers in two lines and misses, ask it to show its work before you reach for a persona.

FINDING 5 COUNTED, 410 OF 420

The models announced the costume

> **Takeaway: 98% of philosopher-prompted answers used the philosophy's vocabulary, and none of the others did.**

This is the part that looked the most like the paper's idea working. The Descartes answers talk about methodic doubt. The Hegel answers talk about dialectic. The Socrates answers open with a "Socratic reformulation." It reads like reasoning.

The side-by-side in the Overview is typical. Both answers were graded right, and the dressed-up one takes a lap around the philosophy first. Across all the philosopher answers, 410 of 420 used that vocabulary. The plain and placebo answers used it zero times.

DIRECTION

An answer that sounds like careful reasoning isn't evidence of careful reasoning. Grade the conclusion, not the performance.

FINDING 6 5 CASES

Long prompts made one model think itself into silence

> **Takeaway: Sonnet returned an empty answer 5 times out of 120 under a long prompt, and never without one.**

Claude Sonnet 5 thinks before it answers. Under a long system prompt it sometimes thought through the entire 8,000-token allowance and never wrote a reply. That happened 5 times in 120 dressed-up calls (placebo once, Socrates once, Aristotle twice, Hume once). It happened 0 times in 30 plain ones. You still pay for the thinking.

DIRECTION

Give a thinking model room when you add long instructions, and watch for answers that come back empty.

CONCLUSION

What I'd do with this

On a strong model, skip the philosopher. You'll get the same answers for up to 2.3 times the cost. (1, 3)

If a cheap model answers too briefly and misses, ask it to show its work. Any long prompt did that here. (4)

Test a persona before you trust it. Run the same questions with and without it, and run the no-prompt version at least twice. Don't treat a difference smaller than the gap between those repeat runs as signal. (1, 2)

Judge an answer by its conclusion, not by how thoughtful it sounds. (5)
When you add long instructions to a thinking model, give it room and watch for empty replies. (6)

WHAT I'M NOT CLAIMING

That the paper is wrong. It tested chemistry questions on different models, and it's possible Socrates really does help there. That any philosopher is better than another, since every gap here is inside the observed run-to-run variation. That these prompts make models worse at coding. That this holds for models or kinds of work I didn't test. This is 15 bugs, and one flipped answer is a whole point.

METHOD

Removed & limits

REMOVED · REV 1 **A philosopher leaderboard.** The data can rank them, but every gap between philosophers on a model is inside the 2-point variation between that model's two no-prompt runs, so the ranking would be noise with names on it.

REMOVED · REV 1 **"Descartes and Hume doubled DeepSeek's score."** True on paper, 3.5 to 7.5. But the placebo got most of the way there, and the same plain model scored 6.5 in the screen.

Small. 15 bugs, four models, one run per prompt (two for the plain prompt).

Two baseline runs. Two identical no-prompt runs per model show that scores move on their own. They can't show how far. The real run-to-run variation could be larger than 2 points, and DeepSeek's no-prompt score was 6.5 in the screen against 3.5 in the main run.

No adversarial re-analysis. The audits in How it was tested were run by the same model that ran the test. Nobody independent tried to break the result.

My own problems. The bugs come from my homelab. They're real, but they're one person's mix of databases, scripts and containers.

Chemistry prompts, used unchanged. The philosopher prompts were written for chemistry. I used them word for word so I wasn't tuning them until they won. Prompts written for debugging might do better.

Grading calls. On three bugs the graders accepted a fix that might not run exactly as written. Each call was applied the same way to all 40 answers for that bug, so it doesn't favor one prompt. On one bug the problem didn't say what the API returns later, so a reasonable guess was graded wrong against the real cause.

One grader family. Claude graded everything. The cross-check covered the one model where that matters most, and agreed on 147 of 150.

PROVENANCE

Why this result is credible

Every score comes from a stored answer graded against a fix that's known to work, and every cost comes from the provider's own bill. Each of the 600 answers is saved with the exact request, the model's reply, its token count and what it cost.[3]

600	2	147/150	$7.00
answers, graded blind, each saved with its request, reply, tokens and cost	identical no-prompt runs per model, to see how far a score moves on its own	exact agreement when a second company's model re-graded Sonnet's answers	total spent calling the models, screen and cross-check included

Where a model sat in the measurement. The four tested models only answered. Grading was done by Claude sub-agents, one per bug, who saw the problem, the known fix and 40 shuffled answers, with no model or prompt names attached.[5] Because Claude Sonnet 5 was one of the tested models, Gemini 3.8 Flash re-graded all 150 Sonnet answers separately.[6]

PROVENANCE

How it was tested, including what broke

Six predictions went into a written ledger before any answer came back, so they couldn't be quietly reworded afterward.[7] Two of them were wrong. The tests below are listed whether they passed or not.

A DIFFICULTY SCREEN All 25 candidate bugs went to all four models with no system prompt first. Seven that every model solved were dropped, since a prompt can't improve a perfect score. Three more with the least room were dropped too, leaving 15.[4]

AN AUDIT THAT CHANGED THE TEST In the screen, Claude Sonnet 5 spent its entire 4,000-token allowance thinking on two bugs and returned nothing. That's a setting problem, not an answer, so the limit went to 8,000 for every model before the main run.

TWO UNFAIR PROBLEMS, FIXED The graders caught one bug that quietly assumed a tool was installed, and one answer key that expected facts the problem never gave. Both were fixed and re-checked before the main run.

AN AUDIT THAT KILLED A RESULT The first analysis said the noise was zero, because each model's two plain runs scored exactly the same. That made every difference look real. Looking bug by bug, two to four answers had flipped on every model, and the flips happened to cancel. The honest variation is about 2 points out of 15, and most of the differences disappeared into it. Two runs can show that a score moves on its own. They can't pin down how far.

A NULL CHECK The second plain run is a test where the right answer is "no difference." It came out that way in the totals, and the bug-by-bug flips inside it became the variation band you see in the charts.

A GRADER CHECK Gemini 3.8 Flash re-graded Sonnet's 150 answers on its own. Same verdict on 147, same right-versus-wrong call on all 150, and the totals were 140 vs. 139.

PREDICTIONS THAT FAILED The ledger predicted the weakest model would gain the most, and that Socrates would answer with questions instead of a diagnosis. Neither happened.

PROVENANCE

How it was made, and what it took

8

prompts from me across two sessions, the first draft and this revision, counted from the transcripts

850

model calls: 100 in the screen, 600 in the main run, 150 in the cross-check

The calls cost $0.52 for the screen, $6.19 for the main run and $0.29 for the cross-check. The first draft took one morning, from reading about the paper to publication. No one ran an adversarial re-analysis, meaning a second analyst whose job is to break the result. That's listed with the other limits.

The calls went through OpenRouter, one provider pinned per model, with the paper's settings (temperature 0.2, top_p 0.3). The runner, grader batches and analysis are small Python scripts. The harness they grew from was built for an earlier prompt test in a separate session and isn't counted above.

Models, by role. Claude Opus 5.5 designed the test, turned my notes into problems, ran it, graded it and wrote this page. Gemini 3.8 Flash only re-graded. Claude Sonnet 5, GPT-5.6 Luna, DeepSeek V4 Flash and Gemini 2.5 Flash-Lite only answered. They run from a top-tier model to a 14-month-old budget one.

PROVENANCE

Where AI was used

The prose of this report was written by an AI, Claude Opus 5.5, under my editorial direction. I own the question, the choice of bugs and every editorial call, including what changed in Revision 2.

The scores were graded by AI, blind, and checked by a second AI from a different company. The costs and token counts weren't produced by an AI at all. They're the providers' own billing records for each call.

What that means for you: you're trusting the stored answers, the answer keys and the billing records, and each of those is named in the citations.

PROVENANCE

Version history

REV 1 · SEP 27, 2026 First publication.

REV 2 · SEP 27, 2026 A polish pass. No result changed. "Normal wobble" became the variation between two identical no-prompt runs, since two runs can't measure a stable noise level. The cost-efficiency chart now sets each model's no-prompt result to 1.0. The plain-vs-Descartes example moved up to the Overview. The PDF now reads findings first and method last, and the analysis credit names the model.

APPENDIX

Citations

EXT Harb, H., Sun, Y., Unal, M., Chia, N., Yang, Z., Ingram, B., Surendran Assary, R. "The ballad of LLM agents: philosophical reasoning for chemistry." *Machine Learning: Science and Technology* 7(3), 030503, Jun 17, 2026. Argonne National Laboratory. 243 ChemBench numeric questions; GPT-4o,

GPT-5, GPT-5.1. iopscience.iop.org/article/10.1088/2632-2153/ae792d (https://iopscience.iop.org/article/10.1088/2632-2153/ae792d). Graded: peer-reviewed, a single lab, chemistry only. The authors state that no single philosopher won across every model and task type. Used here for the question and the prompts, not as a claim about debugging.

EXT The paper's code and prompts: github.com/HassanHarb92/Sci_reasoning_LLMs (https://github.com/HassanHarb92/Sci_reasoning_LLMs). The seven philosopher prompts (`sys_prompts/*.txt`) were copied unchanged at commit 0682e819 (https://github.com/HassanHarb92/Sci_reasoning_LLMs/tree/0682e819a19e3371c8f9d5c3ecd308a9cd0ea1cb/sys_prompts). The comparison prompts are in the repo's baseline test script (`run_one_control_async.py`). Graded: primary source.

DATA The run: 600 calls through OpenRouter on Sep 27, 2026, each stored with its full request, reply, provider, token counts and billed cost (`results.jsonl`, written by `run.py`; order shuffled with a recorded seed; one provider pinned per model, no fallbacks).

DATA The problems: 25 candidate bugs drawn from the author's own notes on problems already solved, each with an answer key and a note of where it came from (`PROBLEMS.md`); 15 kept by a no-prompt difficulty screen (`SCREEN.md`, 100 calls, $0.52). Hostnames and addresses removed.

DATA Grading: one blind grader per bug, 40 shuffled answers each, no model or prompt names attached, checked for name leaks before grading (`grade_prep.py`). Scores and the variation band are computed by `analyze.py`. The band is the bug-by-bug change between two identical no-prompt runs, an observed variation from two runs, not a measured noise distribution.

DATA Cross-check: Gemini 3.8 Flash re-graded all 150 Claude Sonnet 5 answers with the same rubric (`crosscheck.py`, $0.29). Exact agreement 147 of 150.

DATA Predictions: six hypotheses written into the research ledger before the main run, with the method attached, then settled after it. Two supported, one mixed, one untestable at this size, two refuted.