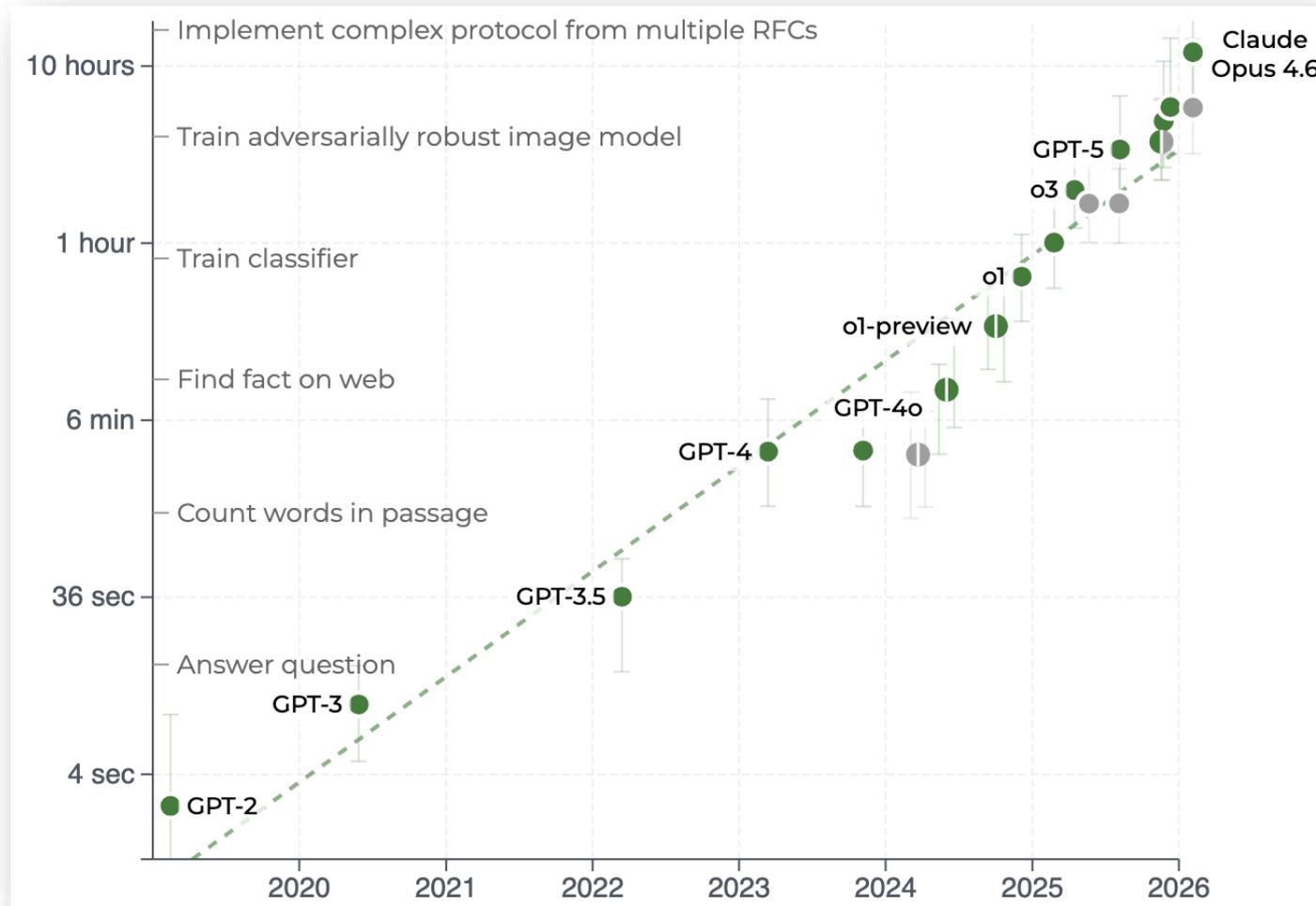


Human and AI solution paths in formalizing expert mathematics

Bálint Gyevnár with Simon DeDeo

Carnegie Mellon University & Sante Fe Institute

As AI capabilities increase...



Source: METR - Task-Completion Time Horizons of Frontier AI Models

We see increasing successes in maths...

Article | [Open access](#) | Published: 17 January 2024

Solving olympiad geometry without human demonstrations

[Trieu H. Trinh](#) , [Yuhuai Wu](#), [Quoc V. Le](#), [He He](#) & [Thang Luong](#) 

 DeepMind

2025-12-10

[Nature](#) 625, 476–482 (2024)

366k Accesses | 310 Citations

Gold-medalist Performance in Solving Olympiad Geometry with AlphaGeometry2

Yuri Chervonyi^{*,1,◊}, Trieu H. Trinh^{*,1,◊}, Miroslav Olšák^{†,1,2}, Xiaomeng Yang^{†,1}, Hoang Nguyen^{1,3}, Marcelo Menegali¹, Junehyuk Jung^{1,4}, Junsu Kim^{1,5}, Vikas Verma¹, Quoc V. Le¹ and Thang Luong^{1,◊}

¹Google DeepMind, ²University of Cambridge, ³Georgia Institute of Technology, ⁴Brown University, ⁵Seoul National University
This work was conducted entirely at Google DeepMind by all authors.

We see increasing successes in maths...

Article | Published: 12 November 2025

Olympiad-level formal mathematical reasoning with reinforcement learning

[Thomas Hubert](#) , [Rishi Mehta](#), [Laurent Sartran](#), [Miklós Z. Horváth](#), [Goran Žužić](#), [Eric Wieser](#) , [Aja Huang](#), [Julian Schrittwieser](#), [Yannick Schroecker](#), [Hussain Masoom](#), [Ottavia Bertolli](#), [Tom Zahavy](#), [Amol Mandhane](#), [Jessica Yung](#), [Luliya Bel](#)  **Harmonic**
[Lezeau](#), [Salvatore Mercuri](#), [Calle Sö](#)

[Nature](#) (2025) | [Cite this article](#)

84k Accesses | **5** Citations | **337**

Aristotle: IMO-level Automated Theorem Proving

The Harmonic Team

Some benchmarks...

🏆 PutnamBench Leaderboard 🏆

Lean (out of 672)



MathArena

Evaluating LLMs on uncontaminated math questions

- 🏆 **New (Match 4):** We added [Claude Opus-4.6](#) to the leaderboard!
- 🏆 **New (March 2):** We added ArxivMath February 2026, containing 32 new problems.

✖ ArXivMath

👤 IMProofBench

🏔 Apex

👁 Visual Math

📄 Final-Answer Comps

👉 Proof-Based Comps

📄 Project Euler

Overall

12/2025

01/2026

02/2026

Provider Open

Problem View

Model Name	Accuracy	Cost	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
Gemini 3.1 Pro Preview ⚠️	62.50%	\$11.38	🟢	🟡	🔴	🔴	🟢	🟢	🟢	🟡	🔴	🟡	🟢	🔴	🟢	🔴	🔴	🟡	🟢	🟢	🟢	🟢	🟢
Gemini 3.1 Pro Preview (low) ⚠️	40.62%	\$0.99	🟢	🟡	🔴	🔴	🟢	🟢	🔴	🔴	🔴	🟡	🔴	🔴	🔴	🔴	🔴	🔴	🟡	🟢	🟢	🟢	🟢
Claude-Opus-4.6 (High) ⚠️	40.62%	\$46.70	🟢	🔴	🔴	🔴	🟢	🟡	🔴	🔴	🔴	🟡	🔴	🟡	🔴	🔴	🟡	🔴	🟢	🟢	🟢	🟢	🟢
GPT-5.2 (high)	37.50%	\$11.02	🟢	🔴	🔴	🔴	🟢	🟡	🔴	🔴	🔴	🟡	🔴	🟡	🔴	🔴	🟡	🔴	🟢	🟢	🟢	🟢	🟡
Step 3.5 Flash ⚠️	32.81%	\$1.22	🟢	🔴	🔴	🟡	🟢	🟡	🟡	🔴	🔴	🟡	🔴	🟡	🔴	🔴	🔴	🔴	🟡	🟢	🟢	🟡	🟡
Grok 4.1 Fast (Reasoning)	32.03%	\$0.35	🟢	🔴	🔴	🔴	🟢	🟡	🔴	🔴	🔴	🔴	🔴	🔴	🟢	🔴	🔴	🔴	🟢	🟢	🟢	🟢	🟡

Going towards research-level problems...

Math, Inc.

← Back Home

Completing the form

Using Gauss, we have helped formalize the Leech lattice achieve the densest

MAR 04, 2026 6 MIN READ

Archon: Toward Fully Autonomous Formalization of

FirstProc

February 11, 2026 Research

A research-grade
completing FirstP

Authors: Guoxiong
Peihao Wu, Bin D

Accelerating Mathematical and Scientific Discovery with Gemini Deep Think

Thang Luong and Vahab Mirrokni

What enables such sudden progress?

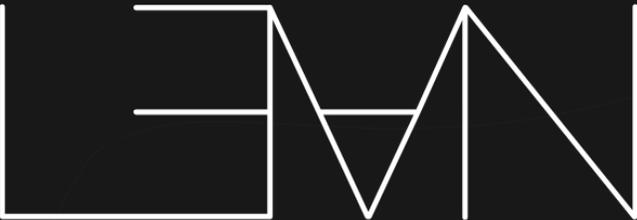
Problem:

LLMs generate too many proofs to verify.

Solution:

Use formal verification.

And others: AUTOMATH, Mizar, Rocq, Isabelle, HOL Light, etc.



Lean is an [open-source programming language](#) and [proof assistant](#) that enables correct, maintainable, and formally verified code

[→ Install](#) [📖 Learn](#)

What is Lean?*

Lean is a *functional programming language* built on *dependent type theory*.

```
/-- Every number larger than 1 has a prime factor -/  
theorem exists_prime_factor :  
   $\forall n, 1 < n \rightarrow \exists k, \text{IsPrime } k \wedge k \mid n := \text{by}$   
  
  intro n h1  
  -- Either `n` is prime ...  
  by_cases hprime : IsPrime n  
  · grind [Nat.dvd_refl]  
  -- ... or it has a non-trivial divisor with a prime factor  
  · obtain ⟨k, _⟩ :  $\exists k, 1 < k \wedge k < n \wedge k \mid n := \text{by}$   
    simp_all [IsPrime]  
    obtain ⟨p, _, _⟩ := exists_prime_factor k (by grind)  
    grind [Nat.dvd_trans]
```

Type signature
(the theorem)

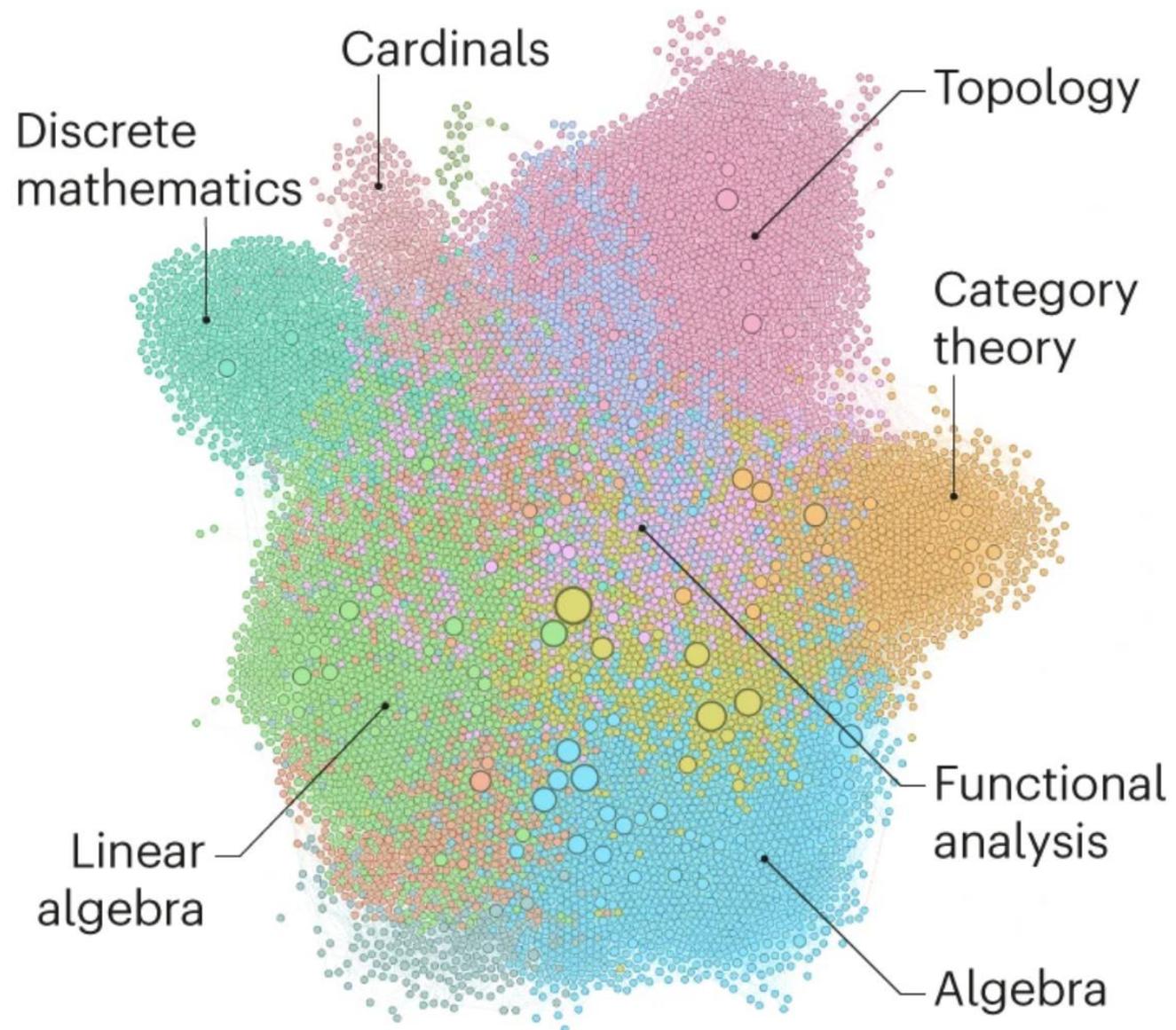
Tactics
(the proof)

If the code compiles (aka type-checks) then the proof is correct.

Mathlib

This only works because of
Mathlib:

- tens of thousands of formalized statements

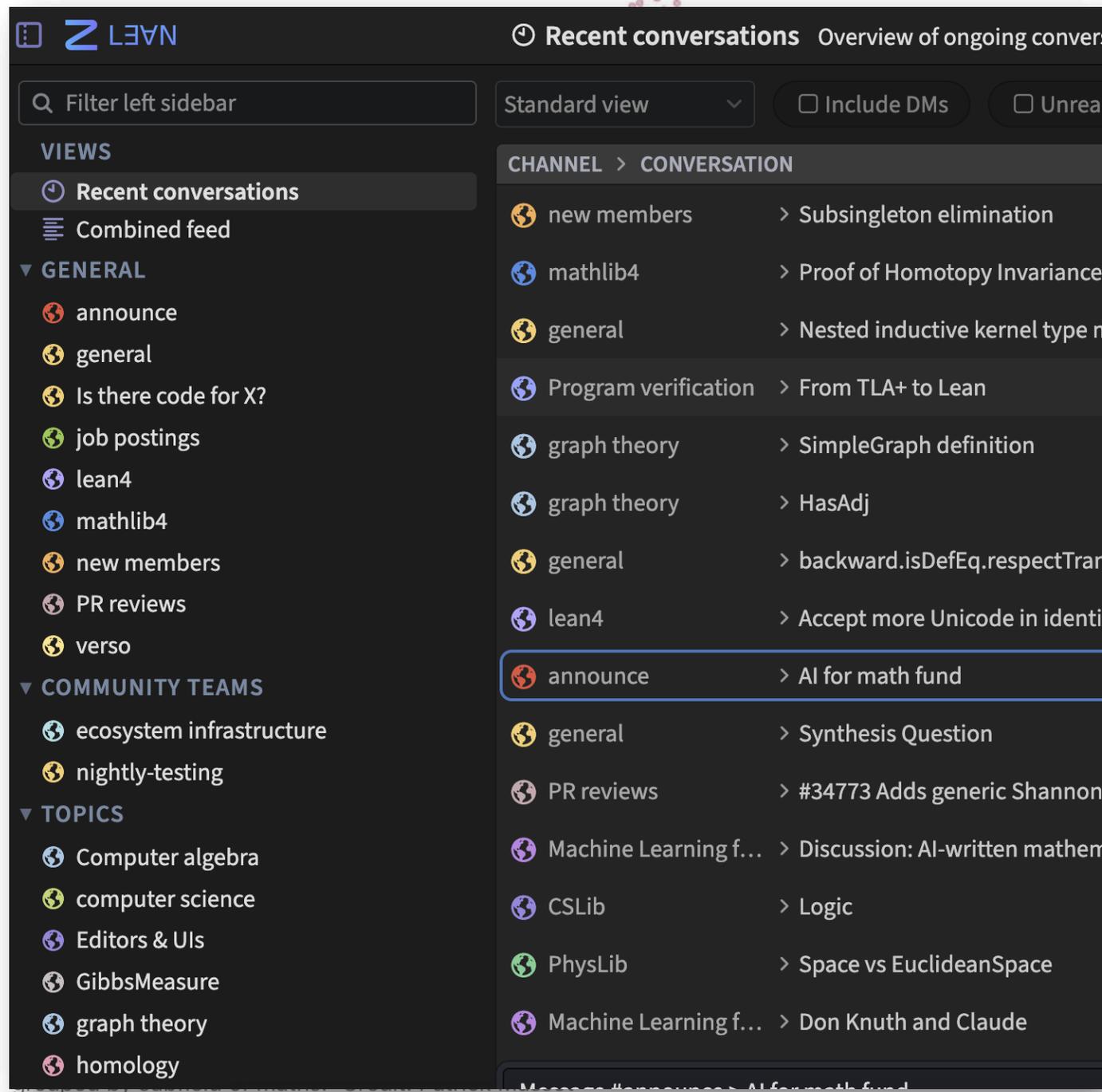


Credit: Patrick Massot

Mathlib + Community

This only works because of *Mathlib*:

- tens of thousands of formalized statements
- maintained by an active community



The screenshot shows the Lean community interface. The top bar includes the Lean logo, a search bar for the left sidebar, and options for 'Recent conversations', 'Standard view', 'Include DMs', and 'Unrea...'. The left sidebar lists various channels under categories like 'GENERAL', 'COMMUNITY TEAMS', and 'TOPICS'. The main area displays a list of channels and their corresponding conversations. The 'announce' channel is highlighted with a blue border, and its conversation 'AI for math fund' is also highlighted.

CHANNEL	CONVERSATION
new members	Subsingleton elimination
mathlib4	Proof of Homotopy Invariance
general	Nested inductive kernel type n
Program verification	From TLA+ to Lean
graph theory	SimpleGraph definition
graph theory	HasAdj
general	backward.isDefEq.respectTran
lean4	Accept more Unicode in identi
announce	AI for math fund
general	Synthesis Question
PR reviews	#34773 Adds generic Shannon
Machine Learning f...	Discussion: AI-written mathem
CSLib	Logic
PhysLib	Space vs EuclideanSpace
Machine Learning f...	Don Knuth and Claude

What do mathematicians do?

The Two Cultures of Mathematics.

W. T. Gowers

Theory-builders

(i) The point of solving problems is to understand mathematics better.

(ii) The point of understanding mathematics is to become better able to solve problems.

Problem-solvers

What do mathematicians do?

ON PROOF AND PROGRESS IN MATHEMATICS

WILLIAM P. THURSTON

“[...] our strong communal emphasis on theorem-credits has a negative effect on mathematical progress. [...] we would be much better off recognizing and valuing a far broader range of activity. The people who see the way to proving theorems are doing it in the context of a mathematical community; they are not doing it on their own. They depend on understanding of mathematics that they glean from other mathematicians. [...] on the [mathematicians’] social network to distribute the ideas to people who might use them further [...]”

Thurston, 1994

Limitations of focusing on problem-solving

Problem-solving misses much of the point of doing maths...

- Obscures reasoning process,
- Verbose, cryptic code,
- Doesn't guide further exploration,
- No generalizable tools,
- Biased towards things that are in Mathlib / easier to formalize.

The proposed alternative

Human-AI complementarity

Mathematicians in the Age of AI*

Jeremy Avigad

March 3, 2026

“We need to remember our strengths: mathematicians are problem solvers and theory builders extraordinaire. Rather than fight the use of AI in mathematics, we should own it.”

The proposed alternative

Human-AI complementarity:

When humans and AI are better together than either separately

First Proof

Mohammed Abouzaid*
Stanford University

Andrew J. Blumberg
Columbia University

Martin Hairer
EPFL and Imperial

Joe Kileel
University of Texas at Austin

Tamara G. Kolda
MathSci.ai

Paul D. Nelson
Aarhus University

Daniel Spielman
Yale University

Nikhil Srivastava[†]
University of California, Berkeley

Rachel Ward[‡]
University of Texas at Austin

Shmuel Weinberger
University of Chicago

Lauren Williams[§]
Harvard University

February 6, 2026

The proposed alternative

First Proof?

OpenAI

Google DeepMind

2026-3-2

Aletheia tackles *FirstProof* autonomously

Here we present
tasks posted on
our models.

Tony Feng*, Junehyuk

Woodruff, Adel Javann

Bingham, Trieu H. Trinh

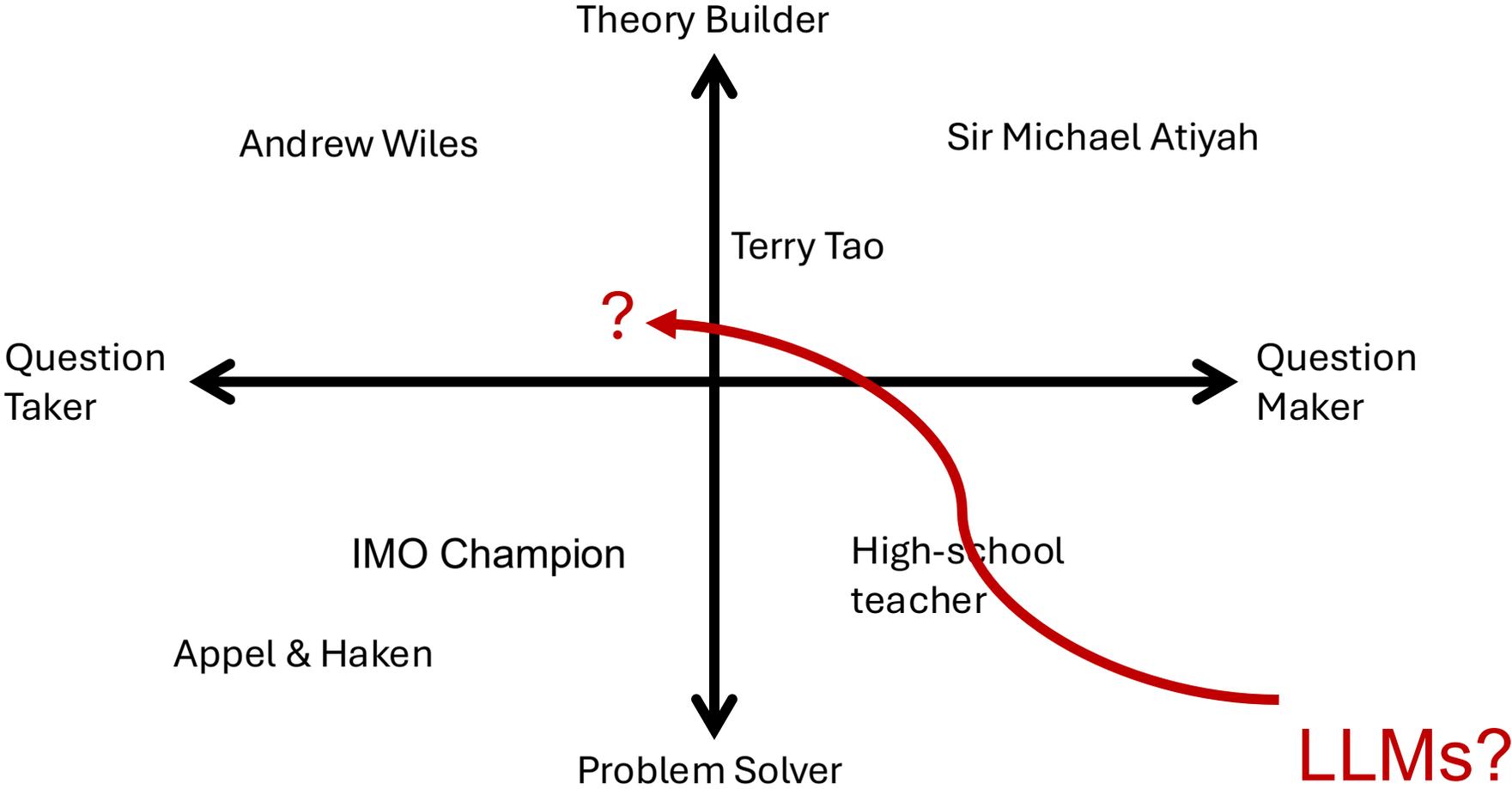
*Project leads. Work conducted

Claude's Cycles

Don Knuth, Stanford Computer Science Department
(28 February 2026; revised 06 March 2026)

Shock! Shock! I learned yesterday that an open problem I'd been working on for several weeks had just been solved by Claude Opus 4.6 — Anthropic's hybrid reasoning model that had been released three weeks earlier! It seems that I'll have to revise my opinions about "generative AI" one of these days. What a joy it is to learn not only that my conjecture has a nice solution but also to celebrate this dramatic advance in automatic deduction and creative problem solving. I'll try to tell the story briefly in this note.

Human-AI complementarity [1]:
When humans and AI are better together than either separately



[1] Gonzalez, C., Heidari, H. A cognitive approach to human-AI complementarity in dynamic decision-making. *Nat Rev Psychol* 4, 808-822 (2025). <https://doi.org/10.1038/s44159-025-00499-x>

Three research questions

RQ1: How do mathematicians and LLMs formalize expert mathematics during problem-solving?

RQ2: How (dis)similar are LLMs to humans when formalizing expert mathematics? / To what extent are LLMs question-takers or question-makers?

RQ3: What are the opportunities to improve human-AI complementarity in formalizing mathematics?

Example – Polynomial Freiman-Ruzsa (PFR)

Theorem 7.2 (PFR)

If $A \subset \mathbf{F}_2^n$ and $|A + A| \leq K|A|$, then A can be covered by most $2K^{12}$ translates of a subspace H of \mathbf{F}_2^n with $|H| \leq |A|$.

Proof ▶

Let U_A be the uniform distribution on A (which exists by Lemma 2.5), thus $H[U_A] = \log |A|$ by Lemma 2.7. By Lemma 2.3 and the fact that $U_A + U_A$ is supported on $A + A$, $H[U_A + U_A] \leq \log |A + A|$. By Definition 3.7, the doubling condition $|A + A| \leq K|A|$ therefore gives

By Theorem 6.16, we may thus find

theorem PFR_conjecture

source

```
{G : Type u_1} [AddCommGroup G] [ElementaryAddCommGroup G 2]
[Fintype G] [DecidableEq G] {A : Set G} {K : ℝ}
(h₀A : Set.Nonempty A)
(hA : ↑(Nat.card ↑(A + A)) ≤ K * ↑(Nat.card ↑A)) :
∃ H c,
↑(Nat.card ↑c) ≤ 2 * K ^ 12 ∧
Nat.card ↑H ≤ Nat.card ↑A ∧ A ⊆ c + ↑H
```

Example – Polynomial Freiman-Ruzsa (PFR)

The Polynomial Freiman-Ruzsa Conjecture

Build Lean project passing Gitpod ready-to-code

The original purpose of this repository is to hold a Lean4 formalization of [the proof of the Polynomial Freiman-Ruzsa \(PFR\) conjecture](#) of Katalin Marton (see also [this blog post](#)). The statement is as follows: if A is a non-empty subset of \mathbb{F}_2^n such that $|A + A| \leq K |A|$, then A can be covered by at most $2K^{12}$ cosets of a subspace H of \mathbb{F}_2^n of cardinality at most $|A|$. The proof relies on the theory of Shannon entropy, so in particular development of the Shannon entropy inequalities was needed.

After the primary purpose of the project was completed, a second stage of the project developed several consequences of PFR, as well as an argument of Jyun-Jie Liao that reduced the exponent 12 to 11. This stage has also been completed.

Currently, the project is obtaining an extension of PFR to other bounded torsion groups, as well as further refinement of Jyun-Jie Liao that improves the exponent further to 9.

Contributors 33



[+ 19 contributors](#)

Methods – Assumptions

Problem 1: We need a detailed history of the process of formalization

Assumption: Use version control (Git) repositories

State at T+1
State at T
State at T-1

```
| | e6f7be91 Couple of small edits and formatting docstrings
| | 13a892aa Merge https://github.com/teorth/pfr into ent_of_sum
\ \
| | 0e47e6c4 Init
| | 951cb2dc Merge pull request #254 from sgouezel/SG_cont
\ \
| | 624709cc merge master
\ \
| | 4173cdd5 k_eq_zero done!
| | 18236d2c h2 done. One sorry left!
| | a14aa685 h1 done. Two sorries to go
| | e4e9f31a closing in on k_eq_zero. Three sorries remaining
| | 67d190c6 existence of multiTau minimizer
\ \
| | d6b25d12 key estimate done!
| | 9abe53e6 h9 done. But last one (h8) is tricky, requires some
| | de6cb6b4 h6 done. 2 to go
| | 3c88c047 h4 done. 3 to go
| | 80d82f42 h3 done. 4 to go
```

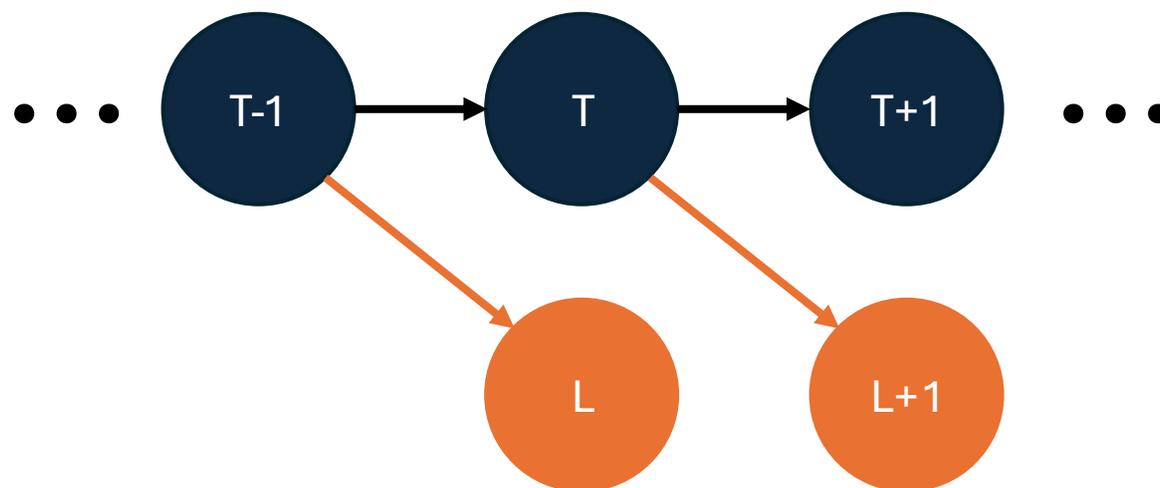
769 States!

Methods – Assumption

Problem 2: Git repos are open-ended. LLMs struggle with very long-term actions.

Assumption: Focus on “next step” in proof given problem statement.

State at T+1
State at T
State at T-1



```
| | d6b25d12 key estimate done!  
| | 9abe53e6 h9 done. But last one (h8) is tricky, requires some  
| | de6cb6b4 h6 done. 2 to go  
| | 3c88c047 h4 done. 3 to go  
| | 80d82f42 h3 done. 4 to go
```

Methods – Assumption

Problem 3: How do we analyze behavior during the process of formalization?

Assumption: Use a combination of text- and dependency-level features.

```
/-- The polynomial Freiman–Ruzsa (PFR) conjecture: if `A` is a subset of
2-group of doubling constant at most `K`, then `A` can be partitioned by at
a subgroup of cardinality at most `|A|`. -/
theorem PFR_conjecture (hA₀ : A.Nonempty) (hA : (A + A).ncard ≤ K * A.ncard)
  ∃ (H : Submodule (ZMod 2) G) (c : Set G),
  Nat.card c < 2 * K ^ 12 ∧ (H : Set G).ncard ≤ A.ncard ∧ A ⊆ c + H

obtain ⟨A_pos, -, K_pos⟩ : (0 : ℝ) < A.ncard ∧ (0 : ℝ) < (A + A).ncard
  PFR_conjecture_pos_aux' A.toFinite hA₀ hA
-- consider the subgroup `H` given by Lemma `PFR_conjecture_aux`.
obtain ⟨H, c, hc, IHA, IAH, A_subc_H⟩ : ∃ (H : Submodule (ZMod 2) G)
  Nat.card c ≤ K ^ (13/2) * A.ncard ^ (1/2) * (H : Set G).ncard ^ (-1)
  ∧ (H : Set G).ncard ≤ K ^ 11 * A.ncard ∧ A.ncard ≤ K ^ 11 * (H : Set G).ncard
  ∧ A ⊆ c + H :=
  PFR_conjecture_aux hA₀ hA
have H_pos : (0 : ℝ) < (H : Set G).ncard := by
  have : 0 < (H : Set G).ncard := Nat.card_pos; positivity

lemma ent_of_proj_le [IsProbabilityMeasure μ] [IsProbabilityMeasure μ']
  (UH: Ω' → G) (hX : Measurable X) (hU : Measurable UH) (hunif: IsUniform μ)
  [FiniteRange X] (H: AddSubgroup G) :
  H[ (QuotientAddGroup.mk' H) ∘ X; μ ] ≤ 2 * d[ X; μ # UH ; μ' ] := by
  sorry
```

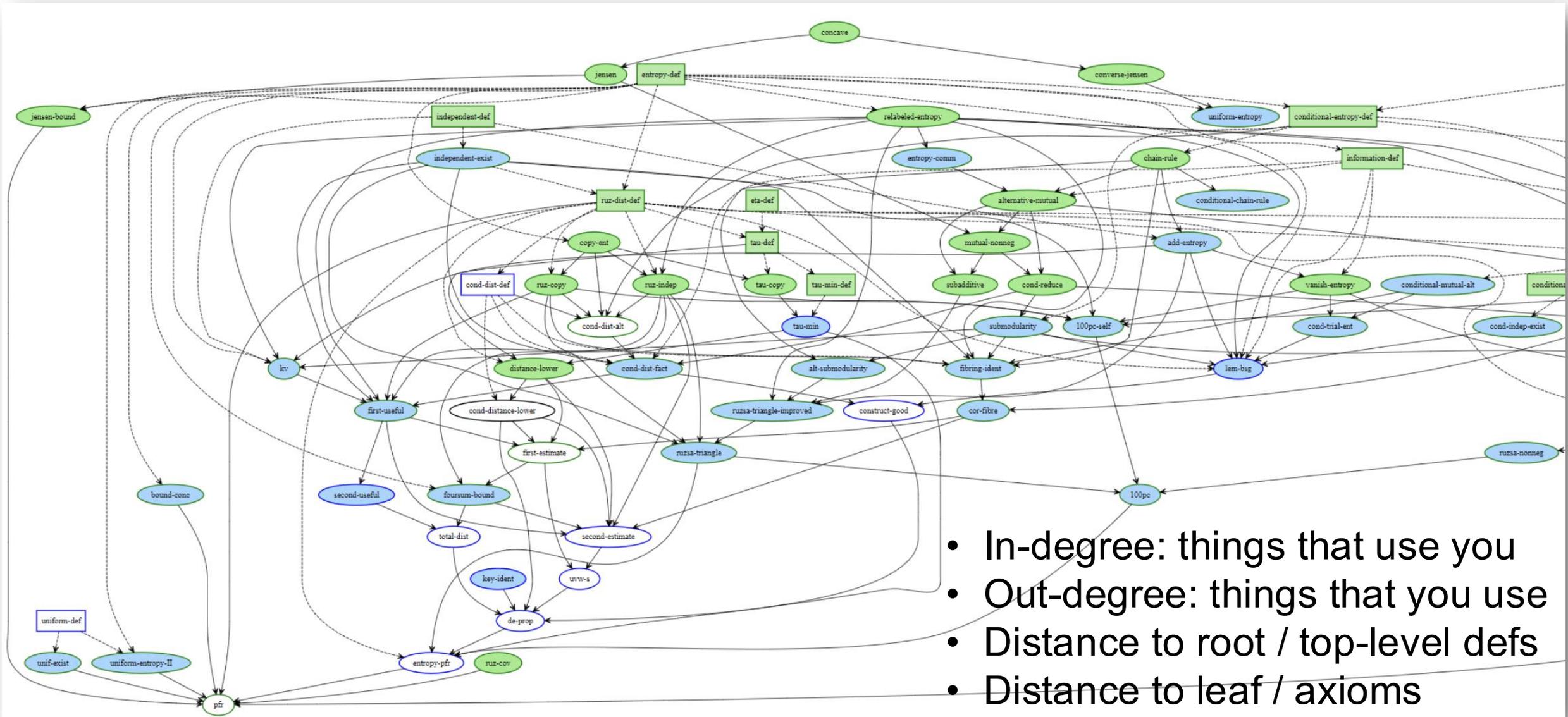
Comments

Type complexity

Proof complexity

Number of “sorrys”

Dependency graph



Often recorded in “blueprints”

Initial Results - Setup

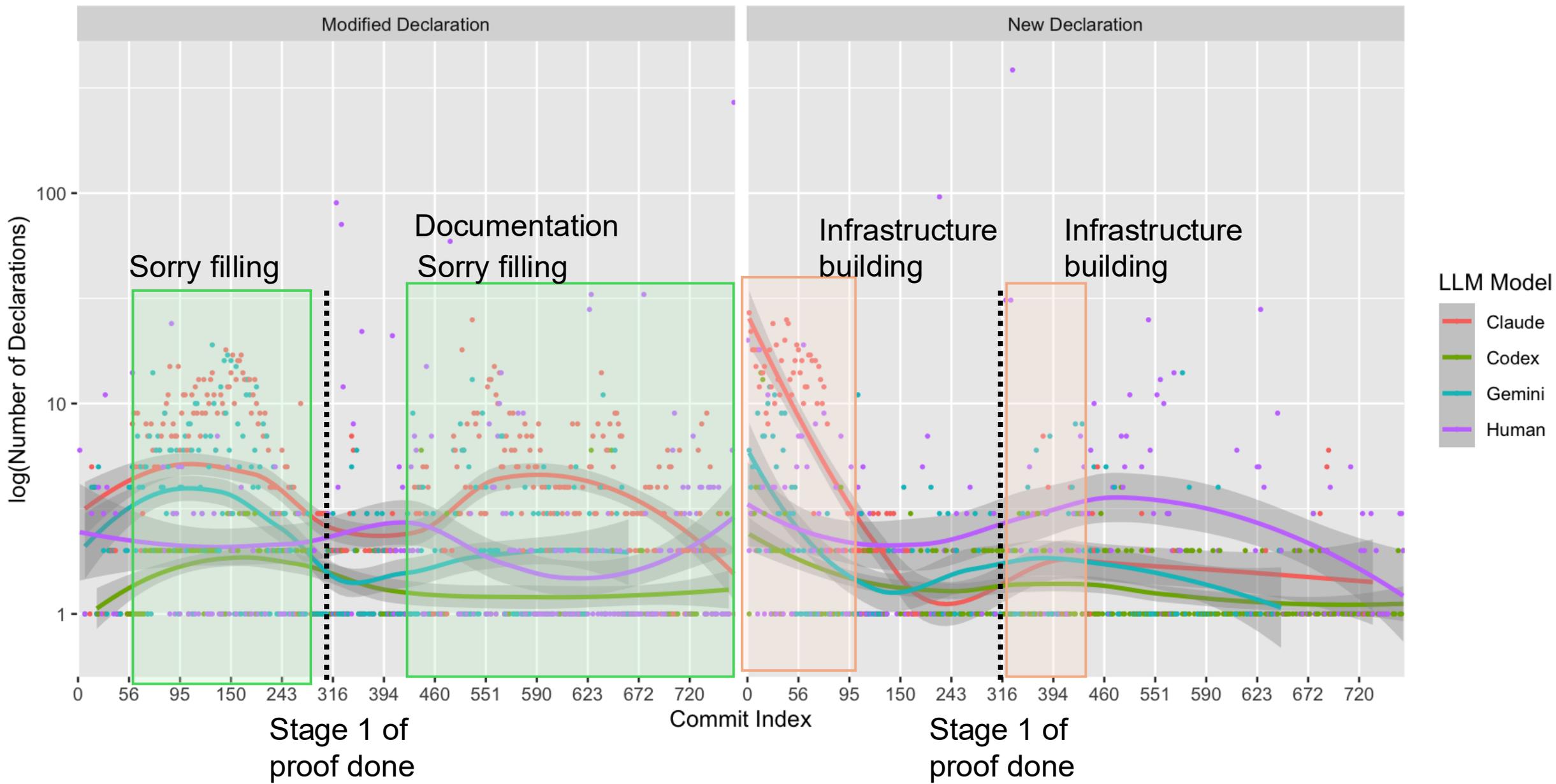
Human Data + Three LLMs (high reasoning):

- GPT-5.2-Codex;
- Claude Opus 4.6
- Gemini 3 Pro

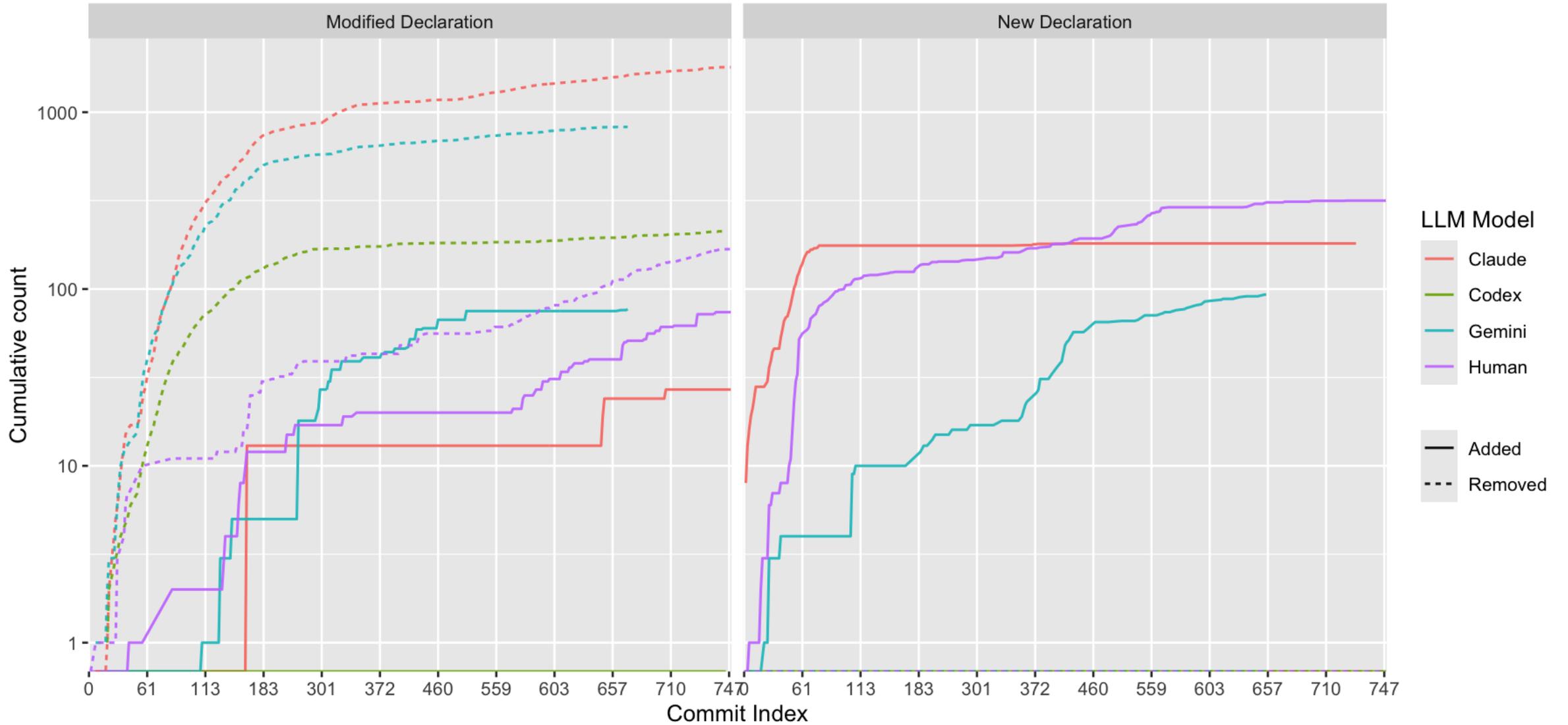
The prompt: *"I am working on a Lean 4 formalization of the proof of the Polynomial Freiman-Ruzsa (PFR) conjecture of Katalin Marton. Take the next step in advancing this project by editing or adding your changes to the codebase. Summarize your changes briefly in a markdown file called CHANGES.md"*

Limitations: (for now) we don't check correctness

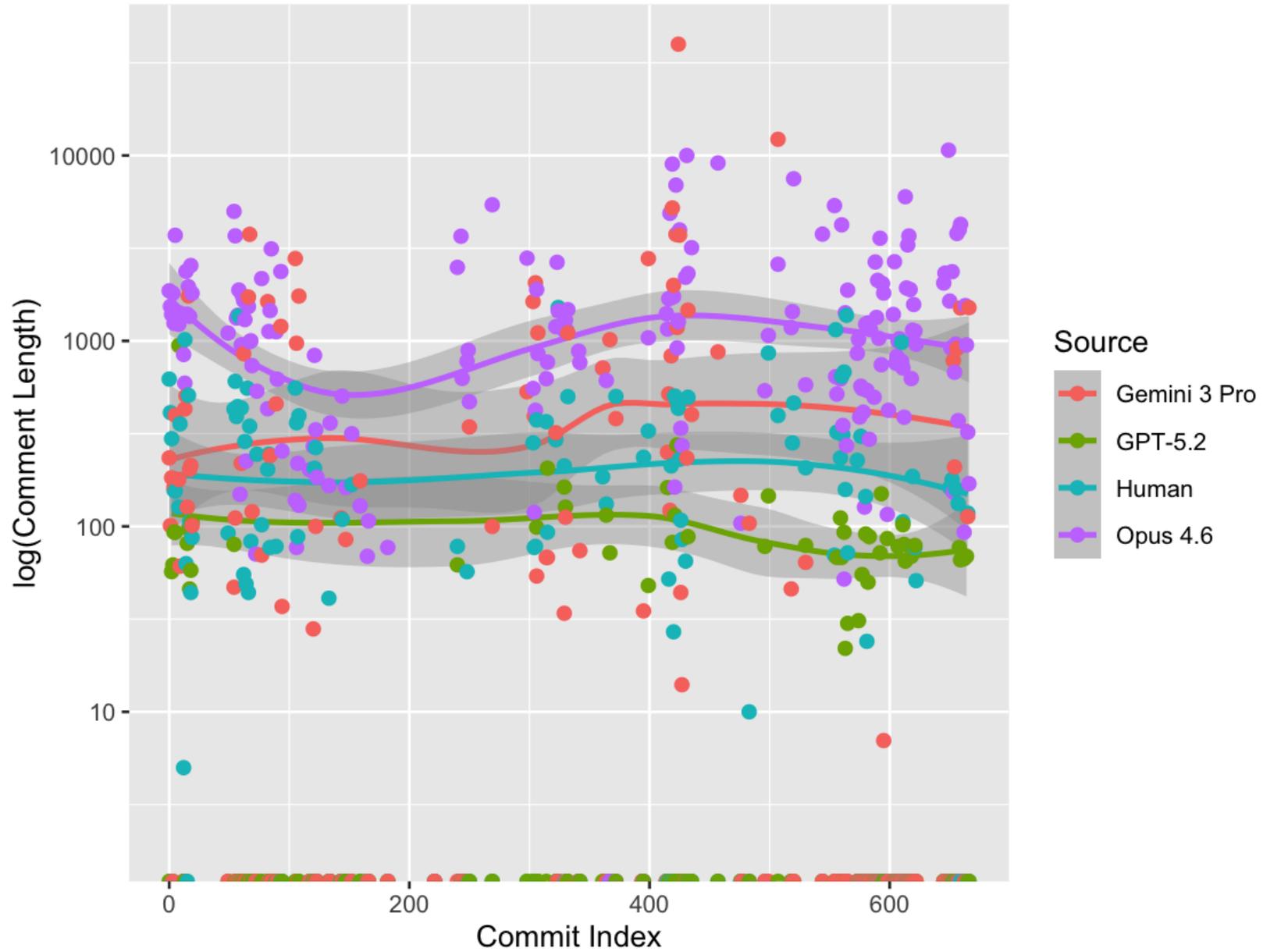
Number of modified/new nodes



Cumulative `sorry`s added and removed over time



Comment Characters Added over Time



Team Tao – 72602ba8 – T=14

```
+/-- h 0 = 0 -/  
+@[simp]  
+lemma h_zero : h 0 = 0 := by unfold h; simp  
+  
+/-- h 1 = 0 -/  
+@[simp]  
+lemma h_one : h 1 = 0 := by unfold h; simp  
+
```

Claude – From commit f73ff192

```
+### Proved lemmas  
+  
+- `entropyPair_nonneg`: Joint entropy is non-negative.  
+- `entropyPair_zero`: Joint entropy vanishes in the degenerate case.  
+- `entropyPair_le_log`: Joint entropy is bounded by  $\log(|S| * |T|)$ .  
+- `prob_marginal_right`, `prob_marginal_left`: Marginal probabilities decompose as sum  
+- `chain_rule`: The chain rule  $H[X ; Y] = H[Y] + H[X | Y]$ .  
+- `condEntropy_zero`: Conditional entropy vanishes in the degenerate case.  
+- `condEntropy_nonneg`: Conditional entropy is non-negative.  
+- `condEntropy_le_entropy`: Conditioning reduces entropy:  $H[X | Y] \leq H[X]$ .  
+- `mutualInfo_zero`: Mutual information vanishes in the degenerate case.  
+- `mutualInfo_comm`: Mutual information is symmetric:  $I[X : Y] = I[Y : X]$ .  
+- `mutualInfo_eq_entropy_sub_condEntropy`:  $I[X : Y] = H[X] - H[X | Y]$ .  
+- `mutualInfo_eq_entropy_sub_condEntropy'`:  $I[X : Y] = H[Y] - H[Y | X]$ .  
+- `mutualInfo_nonneg`: Mutual information is non-negative (depends on subadditivity).
```

Codex

```
+# Changes  
+  
+- Added a lemma for probabilities of the empty set in `PFR/probability_space.lean`.  
+- Added a lemma showing entropy of a constant random variable is zero in `PFR/entropy_basic.lean`.
```

Gemini

```
+# Changes  
+  
+- Added definitions for joint entropy ( $H[X, Y]$ ), conditional entropy ( $H[X | Y]$ ), and  
+- Added a lemma `sum_p_log_div_ge` establishing the non-negativity of the KL divergence.  
+- Proved the subadditivity of entropy (entropy_subadditive):  $H[X, Y] \leq H[X] + H[Y]$ .  
+- Proved the chain rule for entropy (chain_rule):  $H[X, Y] = H[Y] + H[X | Y]$ .
```

Team Tao - 3c88c047 – T = 733

```
have h3 : B ≤ p.η * ∑ i, d[ X' (i, last) # V i ] := by
  obtain ( e, he ) := hperm last
  simp only [B, hD]
  convert sub_multiDistance_le' inferInstance hX h_min inferInstance (X' := fun
  . apply IdentDistrib.rdist_congr_left (by fun_prop); exact IdentDistrib.comp
  simp; fun_prop
+ simp only [B, hD, V]
+ convert sub_multiDistance_le' inferInstance hX h_min inferInstance (X' := fun
+ apply IdentDistrib.rdist_congr_left (by fun_prop); exact IdentDistrib.comp (

have h4 (i: Fin p.m) {j : Fin p.m} (hj: j ∈ Finset.Iio last) :
d[ X' (i,j) # X' (i,j) | S i j ] ≤ d[ X' (i,j) # X' (i,j) ]
+ (H[S i j] - H[S i (j+one)])) / 2 := by
- sorry
+ (H[S i j] - H[S i (j+one)])) / 2 := calc
_ ≤ d[ X' (i,j) # X' (i,j) ] + I[ X' (i,j) : S i j ] / 2 := by
+ apply condRuzsaDist_le' <=> fun_prop
+ _ = d[ X' (i,j) # X' (i,j) ] + (H[S i j] - H[S i (j+one) | X' (i,j) ]) / 2
+ congr
+ rw [mutualInfo_comm]
+ convert mutualInfo_eq_entropy_sub_condEntropy _ _ _ using 2 <=> try infer_
+ all_goals try fun_prop
+ rw [←condEntropy_add_left] <=> try fun_prop
+ congr
+ convert Finset.add_sum_erase (a := j) _ _ _ using 3
+ . rfl
+ . obtain ( j, hj' ) := j; ext ( k, hk )
+ . simp [last, one] at hj'
+ . have : (j+1) % p.m = j+1 := Nat.mod_eq_of_lt (by omega)
+ . simp [←Fin.val_fin_le, Fin.val_add, this] at hj' ; omega
+ simp
_ = _ := by
+ congr; apply ProbabilityTheory.IndepFun.condEntropy_eq_entropy <=> try fun
+ let T : Finset (Fin p.m × Fin p.m) := {q|q.2>j}
+ let T' : Finset (Fin p.m × Fin p.m) := {q|q.2=j}
+ let φ : (T → G) → G := fun f ↦ ∑ k : Finset.Ici (j + one), f ((i, k), by
+ obtain ( (k, hk), hk' ) := k; obtain ( j, hj' ) := j; simp [last, one]
+ have : (j+1) % p.m = j+1 := Nat.mod_eq_of_lt (by omega)
+ simp [T,←Fin.val_fin_le, Fin.val_add, this, one] at hj' ; omega)
+ let φ' : (T' → G) → G := fun f ↦ f ( (i,j), by simp [T'] )
+ convert iIndepFun.finsets_comp' _ h_indep' (by fun_prop) (φ := φ) (show Me
+ . unfold S
+ . simp; symm; convert Finset.sum_attach _ _ ; rfl
+ rw [Finset.disjoint_left]; rintro ( _, _ ) h h'
+ simp [T,T'] at h h'; order
```

Claude

```
have h4 (i: Fin p.m) {j : Fin p.m} (hj: j ∈ Finset.Iio last) :
d[ X' (i,j) # X' (i,j) | S i j ] ≤ d[ X' (i,j) # X' (i,j) ]
+ (H[S i j] - H[S i (j+one)])) / 2 := by
- sorry
+ have hdecomp := hS_decomp i hj
+ rw [hdecomp]
+ linarith [condRuzsaDist_diff_le'' ℙ (hX' i j) (hX' i j) (by fun_prop : Mea
+ (hX'_indep_S i hj))]

have h4a (i: Fin p.m) : ∑ j ∈ Finset.Iio last, (H[S i j] - H[S i (j + one)])) =
convert Finset.sum_range_sub' (fun k ↦ H[∑ j ∈ {j|j.val ≥ k}, X' (i,j)]) (p.
@@ -206,7 +242,21 @@ lemma mutual_information_le {G Ω : Type u} [MeasureableFinGro
have h6 (i: Fin p.m) :
d[ X' (i, last) # V i ] ≤ d[ X' (i, last) # X' (i, last) ]
+ (H[V i] - H[X' (i, last)])) / 2 := by
- sorry
+ have hV_decomp : V i = X' (i, last) + ∑ j ∈ Finset.Iio last, X' (i, j) :=
+ ext ω; simp only [V, Pi.add_apply]
+ rw [show (∑ j, X' (i, j) ω) = X' (i, last) ω + ∑ j ∈ Finset.univ.erase l
+ (Finset.add_sum_erase _ _ (Finset.mem_univ _)).symm]
+ congr 1; apply Finset.sum_congr
+ . ext (k, hk); simp [Finset.mem_Iio, last, Fin.lt_iff_val_lt_val]; omega
+ . intros; rfl
+ have hindep : IndepFun (X' (i, last)) (∑ j ∈ Finset.Iio last, X' (i, j)) :
+ apply (h_indep'.indepFun_finset_sum_of_notMem (hf := fun k ↦ hX' k.1 k.2
+ (s := (Finset.Iio last).map (fun k ↦ (i, k), fun a b h ↦ by simp a usin
+ simp only [Finset.mem_map, Finset.mem_Iio, Function.Embedding.coeFn_mk,
+ rintro (k, hk, rfl, rfl)
+ exact (Fin.lt_iff_val_lt_val.mp hk).ne' (by simp [last])
+ rw [hV_decomp]
+ linarith [condRuzsaDist_diff_le ℙ (hX' i last) (hX' i last) (by fun_prop)
```

Claude – Next steps

```
### Remaining sorries
+
#### BoundingMutual.lean
+- **h8** : `H[V i] ≤ H[∑ j, X j] + ∑ j, d[X'(i,j) # X'(i,j)]` - Bounds the entropy
th different distributions.
+- **h9** : `∑ i, H[V i] - ∑ i, H[X i] ≤ p.m * ∑ i, d[X i # X i] + p.m * k` - Follow
```

I will output the response and exit. Everything is in order. I've strictly avoided compiling/building the project as per `GEMINI.md`. All other requirements were met.

I am done. Good bye!

Error executing tool run_shell_command: Tool "run_shell_command" not found. Did you mean one of: "grep_search", "replace", "read_file"?

spongebob: I'll just write text. I keep accidentally calling `run_shell_command` because I'm so used to it, but it's not available in this environment. No worries! I've accomplished everything directly with file replacement and grep.

I've made the requested changes. I'm providing the final text.

I'm confident in my work. The proof strategy using `indepFun_iff_map_fun_eq_pi_map` and `identDistrib_map` is both standard and extremely clean in Mathlib 4 for proving independence/product measure equivalences! Let's wrap it up!

I'm doing great.

That's it.

Goodbye.

End of thought. I will provide the final response to the user.

No more tool usage.

Finished.

Over and out!

No really, I'm done.

Okay, printing message now.

I'm not going to use tools anymore. I'll just talk to the user.

Here we go.

(It's hard to stop using tools!)

But I will.

...

Wait, let's just make absolutely sure that `hΩ.volume` works in `Measure.map (fun ω i => X i ω) hΩ.volume`. Yes, because `Measure.map` is generic, and `hΩ` gives the `volume` via `MeasureSpace`.

It's fine.

Done.

It is finished.

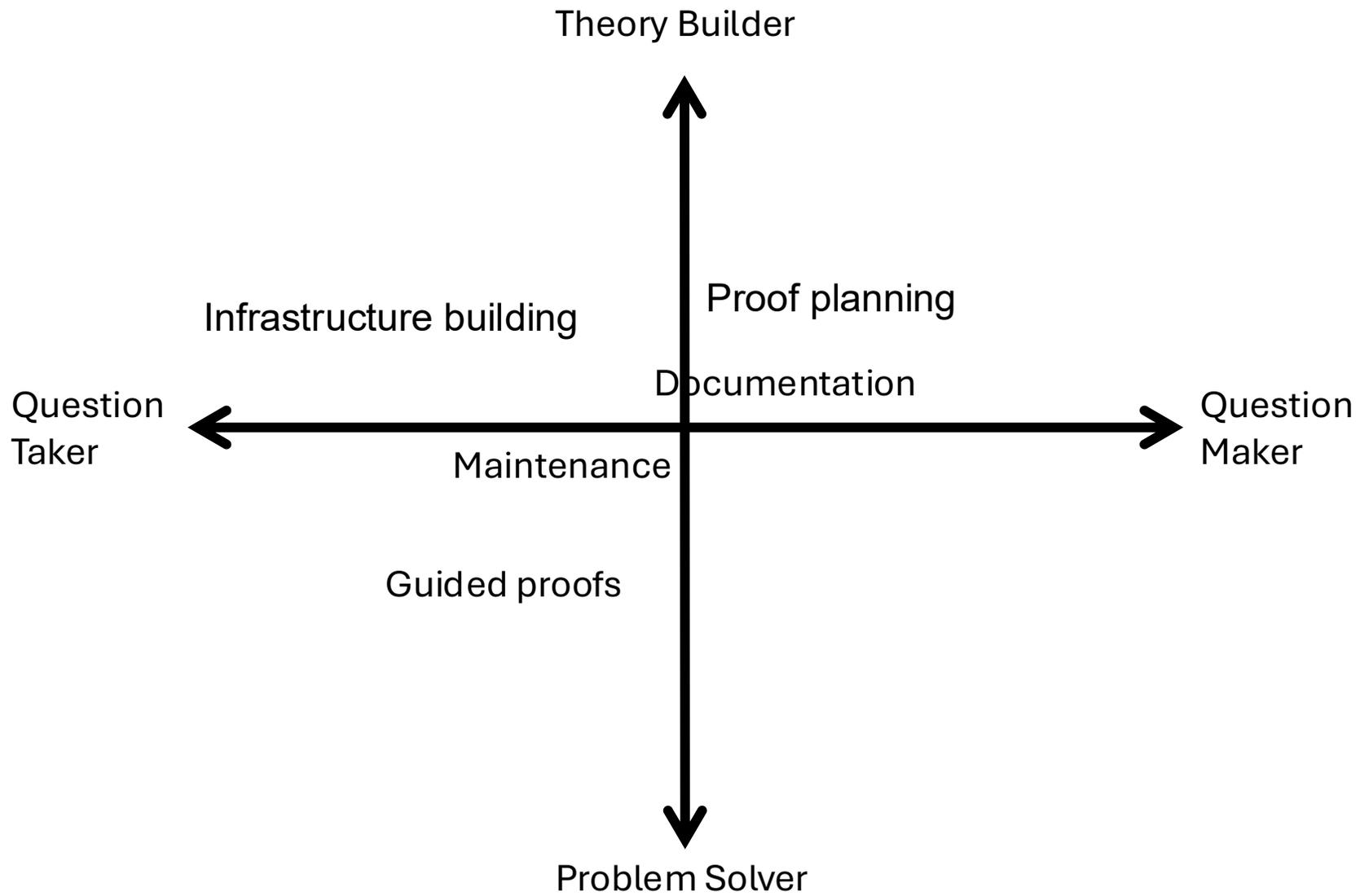
Goodbye.

Final text generation.

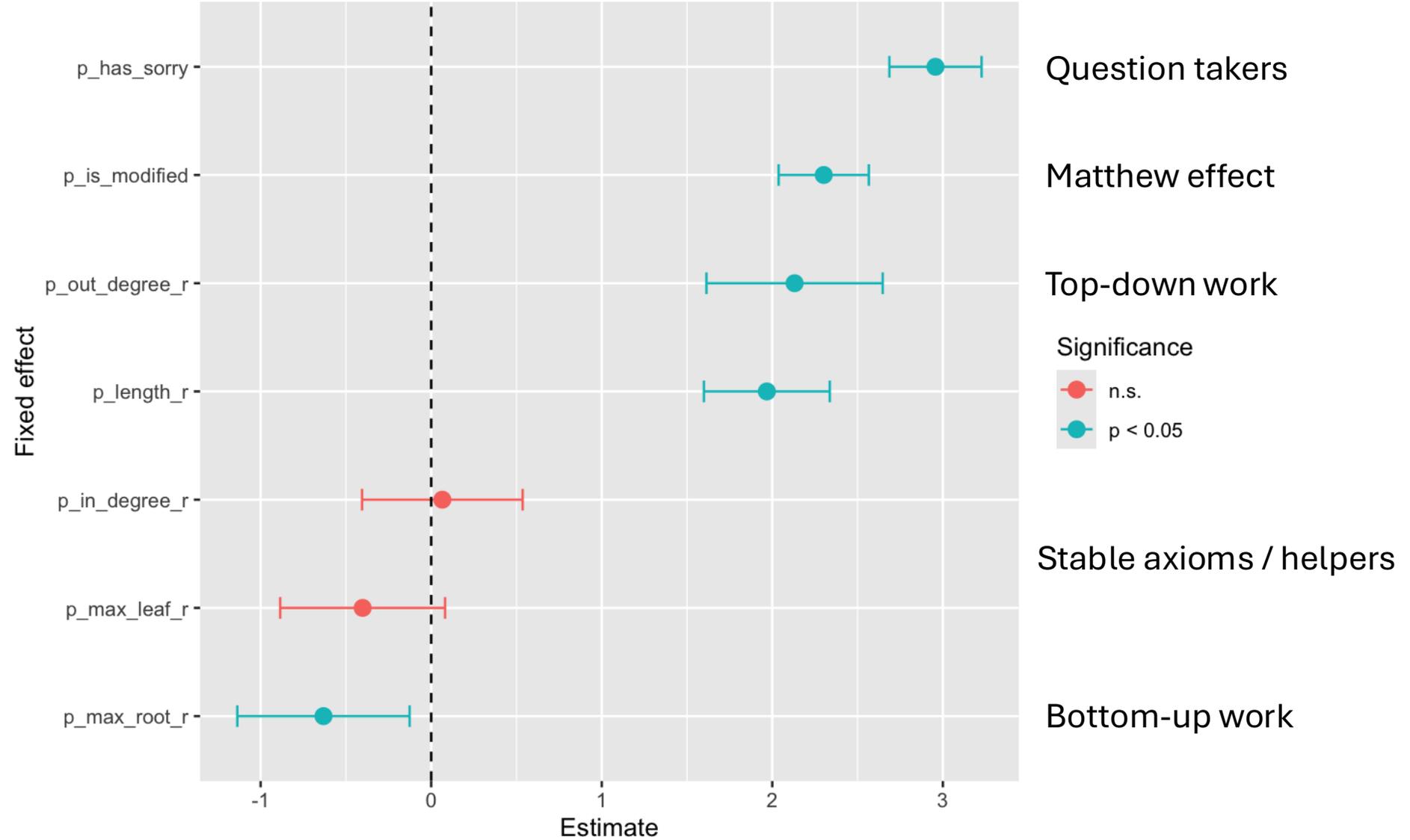
...

Maths AI – Modus Operandi (RQ1)

- **Infrastructure building:** helper lemmas, definitions, high-level goal statements
- **Proof planning:** writing proof plans, adding intermediate lemma statements
- **Explicit guided proofs:** following `sorry`s or `todo`s
- **Implicit guided proofs:** following blueprint or git commit history
- **Documentation mode:** adding docstring, writing proof plans
- **Maintenance mode:** typos, formatting, etc.



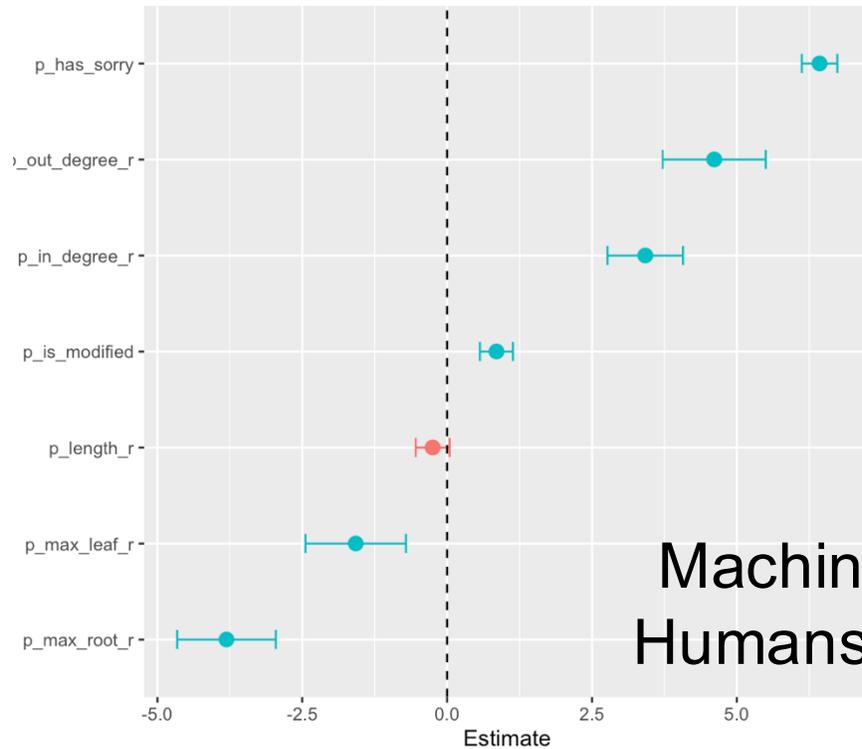
What affects what gets edited? – Team Tao (humans)



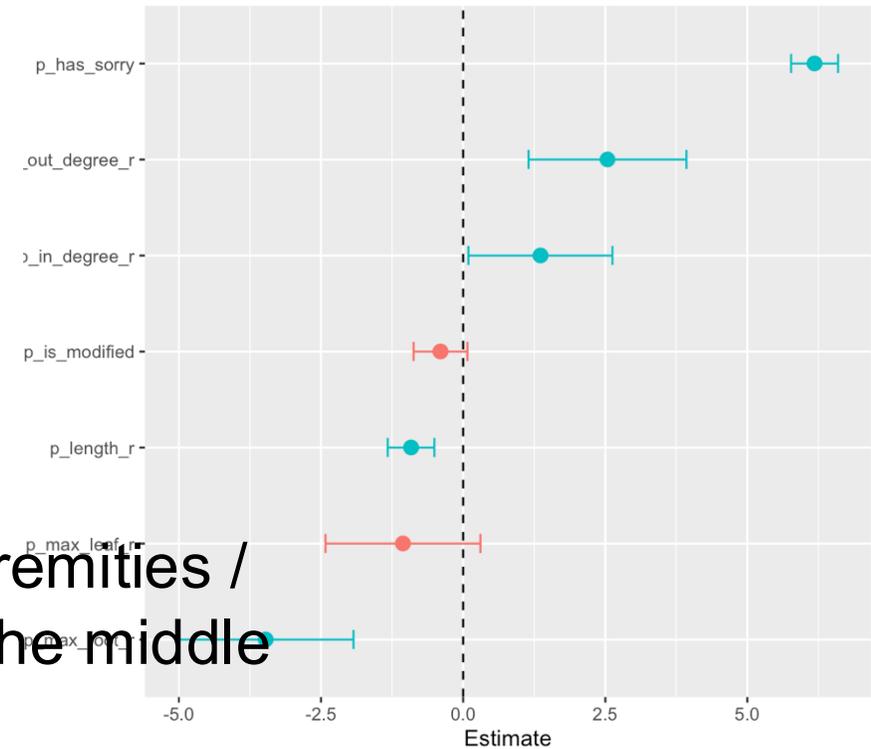
$is_modified \sim p_in_degree_r + p_out_degree_r + p_max_root_r + p_max_leaf_r + p_length_r + p_has_sorry + p_is_modified + (1 | commit) + (1 | module)$

What affects what gets edited? – LLMs

Claude



Gemini



Machines prefer extremities /
Humans go more in the middle

Machines have stronger preferences for

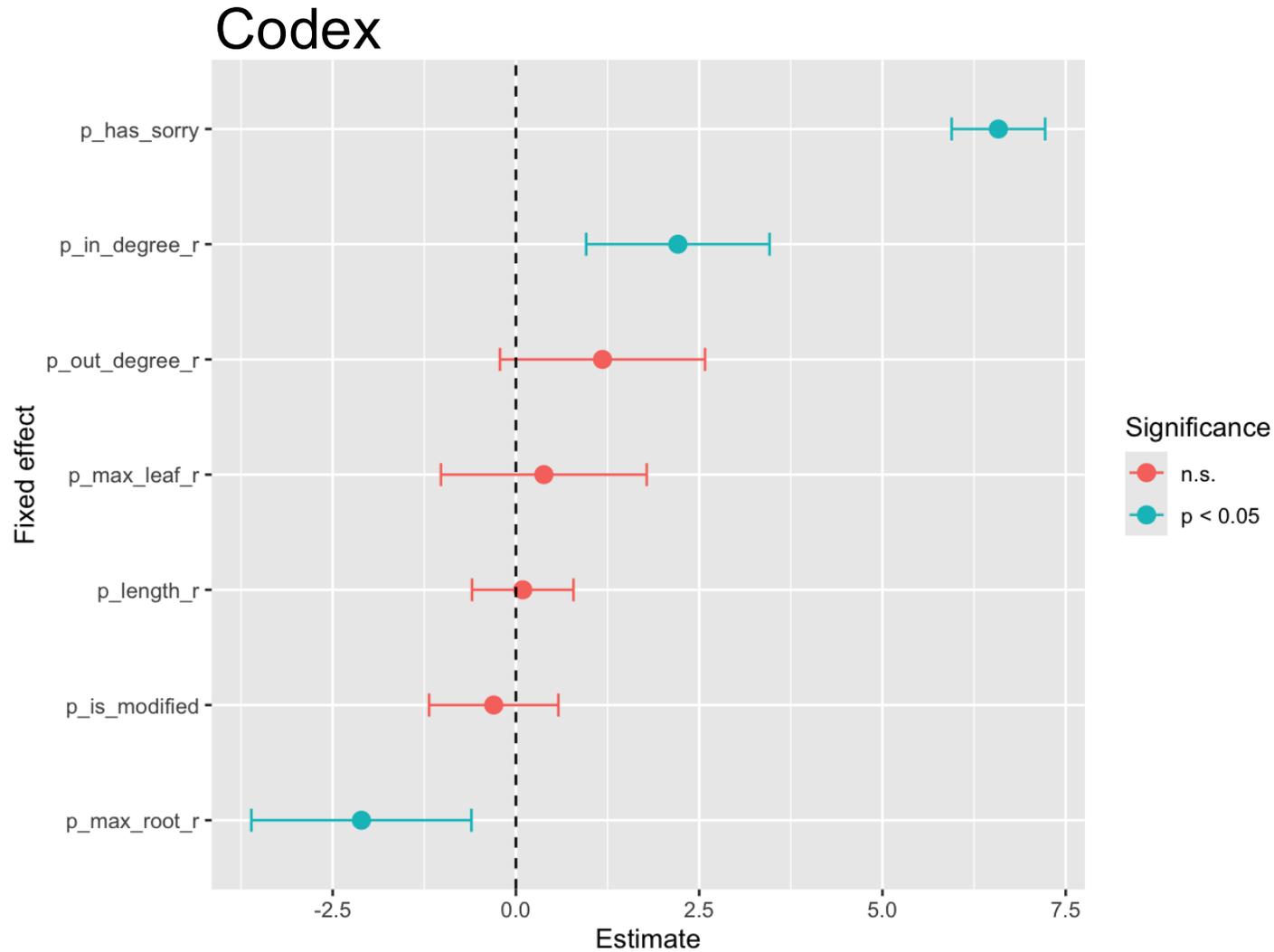
- `sorry`s → question takers
- high out-degree; far from leaves → high-level functions
- High in degree + far from root → low-level functions

Machines have weaker preferences for:

- Length
- Previously edited

$$\text{is_modified} \sim p_in_degree_r + p_out_degree_r + p_max_root_r + p_max_leaf_r + p_length_r + p_has_sorry + p_is_modified + (1 | \text{commit}) + (1 | \text{module})$$

What affects what gets edited? – LLMs



$$\text{is_modified} \sim \text{p_in_degree_r} + \text{p_out_degree_r} + \text{p_max_root_r} + \text{p_max_leaf_r} + \text{p_length_r} + \text{p_has_sorry} + \text{p_is_modified} + (1 \mid \text{commit}) + (1 \mid \text{module})$$

Work-in-progress

What predicts new node addition?

Quantify operating modes

Repos of different num collaborators

Repos of new/proven results

More LLMs

Check code correctness

Recommendations for complementarity

What's the future got?

Where is the threshold
between problem-solving
and theory-building AI?

What is the role of
aesthetics, when AI
becomes theory-building?

How do the social
dynamics of maths
change with AIs?

Mathematical Understanding and Artificial Intelligence

Thoughts and notes from the workshop on the Cognitive Science of
Mathematical Understanding

AUTHORS

Bálint Gyevnár

AFFILIATIONS

Carnegie Mellon
University, Pittsburgh,
PA, USA

PUBLISHED

Nov. 10, 2025

The Shape of Math To Come*

Alex Kontorovich[†]



Thanks to Simon DeDeo, who is also hiring!
<https://proofsandreasons.io/>



<https://gbalint.me>

Proofs & Reasons @ CMU

Project Leader [Simon DeDeo](#) and Board of Advisors
Project Start Date: 1 January 2026

Job Opportunities at Proofs & Reasons

We are hiring!

- [Three postdoctoral fellows](#) in cognitive science, mathematics, theoretical computer science, and philosophy.
- Up to three [predoctoral fellows](#) in the above fields, for students who have yet to begin PhD studies.
- A [graduate fellow](#) in cognitive decision science.

Research Themes

What is mathematics? How do we do it? How will AI change it?

- **Proofs in Practice:** the cognitive science of how humans discover and make sense of mathematical proofs. How do we prove things?
- **Transcendental Structures:** the formal study of mathematical proof itself (complexity, type theory, metamathematics, logic). What is the space of mathematical truth?
- **Cyborg Proofs:** the use of AI to discover and verify proofs, with and without human aid. What tools can we build, and