QEDBENCH: Quantifying the Alignment Gap in Automated Evaluation of University-Level Mathematical Proofs

Anonymous Authors[1]

Abstract

As Large Language Models (LLMs) saturate elementary benchmarks, the research frontier has shifted from generation to the reliability of automated evaluation. We demonstrate that standard "LLM-as-a-Judge" protocols suffer from a systematic **Alignment Gap** when applied to upper-undergraduate to early graduate level mathematics. To quantify this, we introduce **QEDBENCH**, the first benchmark to systematically measure alignment with human experts on undergraduate-level math proofs by contrasting *course-specific rubrics* against *expert common knowledge criteria*. By deploying a dual-evaluation matrix (7 judges $\times$ 5 solvers) against 1,000+ hours of human evaluation, we reveal that certain frontier evaluators like Claude 4.5 Opus exhibit significant positive bias (up to $+0.28$ mean score inflation), effectively "hallucinating rigor" in flawed proofs. Furthermore, we uncover a critical reasoning disparity: while **Gemini 3.0 Pro** achieves state-of-the-art performance (0.91 raw score), specialized reasoning models like **o3-deep-research** collapse in discrete domains, dropping to 42.1% accuracy in Graph Theory. We release QEDBENCH as a public benchmark for evaluating and improving AI judges. Our benchmark can be found in this anonymized link.

1. Introduction

As Large Language Models (LLMs) achieve saturation on elementary mathematical benchmarks (Cobbe et al., 2021; Hendrycks et al., 2021), the research frontier has shifted from generation to the reliability of automated evaluation. While models can now solve high-school competition problems (approaching gold-medal performance in specific domains), evaluating intricate *proofs* at the upper-undergraduate to early graduate level remains an open alignment challenge. Existing benchmarks face a fundamental trade-off between verifiability and scalability. Autoformalization frameworks (e.g., Lean) provide correctness guarantees but incur high annotation costs that limit dataset scale. Conversely, standard 'LLM-as-a-Judge' protocols are scalable but correlate poorly with expert judgment. This reliability gap makes it difficult to distinguish between valid logical reasoning and hallucinated pseudo-precision.

This paper introduces **QEDBENCH**, a rigorous evaluation framework designed not just to test model capabilities in proof-based mathematics at the upper undergraduate and early graduate level, but to audit the LLM judges themselves. Unlike prior works that implicitly tie evaluation quality to model performance, we explicitly decouple these two axes to provide a systematic audit of university proof generation. Our study systematically evaluates proof synthesis and judging across ten core upper-undergraduate/early-graduate disciplines: Analysis, Complex Analysis, Abstract Algebra, Discrete Math, Probability Theory, ODEs, Number Theory, Combinatorics, Algorithms, and Graph Theory.

Methodology Overview. Our experimental design follows a three-stage pipeline to establish a *Human Expert Ground Truth*:

1. Solution Generation. First, to ensure our dataset captures the capabilities of current state-of-the-art systems, we generated solutions for our problem set using five frontier models: `o3-deep-research`, `GPT-5-Pro`, `Claude Sonnet 4.5`, `Gemini 3.0 Pro`, and `DeepSeek-V2-Prover`. This diversity ensures our evaluation is not over-fitted to the idiosyncrasies of a single model family.

2. Expert Annotation & Rubric Engineering. Raw model outputs are insufficient for benchmarking without verified ground truth. We employed a team of PhD-level experts to evaluate these solutions on a granular scale $[0, 0.25, 0.5, 0.75, 0.9, 1.0]$. To isolate the impact of rubric specificity on automated grading, we generated two distinct types of rubrics using `GPT-5.2-pro` (verified by `Gemini 3.0 Pro`):

[1]Anonymous Institution, Anonymous City, Anonymous Region, Anonymous Country.**AUTHORERR: Missing \icmlcorrespondingauthor.**

Preliminary work. Under review by the International Conference on Machine Learning (ICML). Do not distribute.

- **Course-Specific Rubric:** Simulating a standard undergraduate grading rubric, focused on textbook definitions for the relevant course.
- **Expert-Domain Rubric:** Rubrics designed for a standard of fundamental correctness assuming expert common knowledge, where proofs are evaluated on logical soundness and implicit steps are permitted if trivial to a domain specialist, mirroring graduate-level evaluation.

Crucially, our human experts refined these rubrics iteratively to match their own evaluation criteria for the corresponding problems, establishing a verified standard for correctness.

3. Judge Calibration. Finally, we turned to the primary focus of this work: the reliability of automated evaluators. We deployed *seven* judge models (including `GPT-5.2-Pro`, `Claude 4.5 Opus`, and `Gemini 3 Pro`) to grade the generated proofs against both rubrics. By comparing these 7×5 evaluation matrices against the human ground truth, we can quantify the alignment gap by isolating systemic judge bias from model-specific noise.

We address three fundamental questions regarding the trustworthiness of LLM evaluators for upper undergraduate to early graduate-level proof writing and correctness:

1. **The Evaluator Reliability Question:** Do off-the-shelf LLMs perform genuine logical verification in abstract domains that align with expert human evaluators?
2. **The Reasoning Frontier:** When evaluated against a strict human ground truth, how do frontier models perform on complex proofs requiring constructive reasoning beyond retrieval?
3. **The Knowledge Gap:** Do models adhere to the pedagogical constraints of undergraduate definitions, or do they ensure correctness by retrieving advanced, out-of-scope theorems?

1.1. Key Contributions

- **Quantifying the Alignment Gap:** We demonstrate that standard LLM judges exhibit a systematic positive bias compared to human experts. Our dual-evaluation matrix reveals that frontier evaluators like `Claude 4.5 Opus` inflate scores by up to **+0.28** (mean score delta) in abstract domains, effectively "hallucinating rigor" in flawed proofs. Similarly, `Llama-4-Maverick` acts as a grade inflator, achieving an 89.5% pass rate against an 67.6% human baseline.
- **The Discrete-Continuous Divide:** Using our calibrated evaluation, we uncover a critical disparity in reasoning architectures. While **Gemini 3.0 Pro** achieves robust state-of-the-art performance across all categories (0.91 raw score), specialized reasoning models like **o3-deep-research** collapse in discrete domains, achieving 87.5% accuracy in ODEs but dropping to **42.1%** in Graph Theory.
- **A Rigorous Calibration Standard:** QEDBENCH provides the first statistically robust standard for automated university-level mathematical proof evaluation, grounded in over 1,000+ hours of hand-evaluation by PhD-level experts. We publicly release the full dataset, the dual-rubric system, and the evaluation logs to enable the community to benchmark future AI judges against a verified human ground truth.

2. Related Work

Benchmarking the Reasoning Frontier. The evaluation of mathematical capability has evolved from foundational arithmetic (Cobbe et al., 2021; Hendrycks et al., 2021) to domains requiring sophisticated abstraction. Recent efforts like *FrontierMath* (Glazer et al., 2024) and *ImProofBench* (Schmitt et al., 2025) have pushed the difficulty frontier to expert and research levels. However, a trade-off exists between difficulty and evaluation fidelity: *FrontierMath* relies on closed-form answers to ensure verifying correctness is tractable, while *ImProofBench* offers a smaller-scale (approx. 39 problems) look at research generation. Similarly, *LiveBench* (White et al., 2024) addresses contamination via novel problem updates but evaluates proofs via "cloze-style" reordering rather than *open-ended proof generation* where only the problem is provided to the LLM without any other information. Other works have explored specific reasoning dimensions, such as temporal dependencies (Fatemi et al., 2024) and prompt-induced chain-of-thought (Seßler et al., 2024). Initiatives like the *Rosetta Stone* project (Ho et al., 2025) attempt to unify these heterogeneous metrics into a single rating. QEDBENCH occupies a distinct middle ground: we target the upper undergraduate/early graduate curriculum (harder than high school competitions but more verifiable than novel research) prioritizing the semantic quality of full-text proofs over binary answer matching.

Dynamics of Large Reasoning Models (LRMs). The emergence of "Large Reasoning Models" (e.g., o3, Gemini Thinking) has prompted deeper analysis into the nature of model thought. Shojaee et al. (2025) analyze these models through the lens of algorithmic complexity, suggesting that current improvements may stem from pattern matching in controlled environments rather than generalized reasoning. Concurrently, *ReasonBENCH* (Potamitis et al., 2025) highlights the "instability" of these reasoning traces, quantifying high variance in logic despite correct final answers. Our work extends this scrutiny to the domain of higher mathematics, investigating whether the "reasoning stability"

observed in algorithmic tasks holds up when models attempt rigorous theorem proving.

The Challenge of Automated Evaluation. As generation capabilities saturate, the bottleneck shifts to evaluation. The "LLM-as-a-Judge" paradigm (Zheng et al., 2023) has shown promise in competition mathematics. Pioneering efforts like *ProofBench* (also referred to as *Reliable Fine-Grained Evaluation*) (Ma et al., 2025) and *RefGrader* (Mahdavi et al., 2025) demonstrate that fine-tuned evaluators or agentic workflows can achieve high correlation with human judges on Olympiad-style problems (IMO/USAMO). However, we identify a critical **Alignment Gap** when these methods scale to the upper-undergraduate/early-graduate level. Unlike Ma et al. (2025), who fine-tune specific judges on competition data, we evaluate the inherent alignment of frontier foundation models. We find that in the absence of rigid competition rubrics, models like Claude 4.5 Opus exhibit "hallucinated rigor," favoring persuasive language over logical soundness. QEDBENCH thus serves as a counterweight to competition-centric benchmarks, revealing that *solving* proficiency does not guarantee *grading* reliability in abstract mathematical domains.

3. QEDBENCH Methods

The QEDBENCH benchmark comprises 275 expert-curated, proof-based mathematical problems spanning upper-undergraduate and early-graduate curricula. The domain coverage includes Analysis, Complex Analysis, Abstract Algebra, Discrete Math, Probability Theory, ODEs, Number Theory, Combinatorics, Algorithms, and Graph Theory. Our methodology rests on three pillars: adversarial data curation, a tiered-rubric expert evaluation, and a cross-model evaluation matrix.

3.1. Problem Curation and Anti-Leakage

To mitigate the risk of data contamination and memorization, we implemented a rigorous three-stage curation pipeline.

- **Source Diversity:** Problems were sourced from authoritative graduate texts (e.g., *Beals*, *Dummit & Foote*) and advanced qualifying exams (e.g., Kent State) to ensure high mathematical quality.
- **Syntactic Rewording:** We manually rephrased all problem statements into *ab initio* equivalents. This process alters linguistic patterns while preserving the underlying logical structure, neutralizing simple surface-level memorization.
- **Adversarial Audit:** We subjected every candidate problem to an automated deep-search audit. We utilized `o3-deep-research` to scan the open web for existing solutions to our rephrased prompts.

The prompt used for this adversarial audit is shown in Listing 1. We instructed the agent to strictly distinguish between *equivalent* solutions (contamination) and *analogous* problems (acceptable domain overlap).

Listing 1. The adversarial prompt used to detect data contamination. We instructed the agent to strictly distinguish between *equivalent* solutions (which preserve all constraints) and *analogous* problems.

```
Goal: Search the web comprehensively for **
    complete solutions**.

Acceptable matches:
- exact_solution: The solution explicitly
    solves the *same problem statement* (
    identical content).
- equivalent_solution: The solution solves
    a *rephrased* version of the same
    problem where all mathematical objects,
     parameters, and constraints are
    preserved (notation/order may differ).
    This does **NOT** include analogous,
    special-case, more-general, or merely
    similar problems.

Verification requirements:
- Confirm that the problem in the source
    matches the user's problem exactly or
    is a faithful rephrasing.
- If rephrased, state the mapping of
    symbols/notation (e.g., ``their n = our
     k'').
```

Of the final 275 problems, the audit found online solutions for 91 instances, while the remaining 192 returned no matches. In Section 4, we analyze performance variance between these two subsets to quantify the impact of potential contamination.

3.2. Expert Evaluator Selection

We recruited 48 expert evaluators, all of whom either have taken PhD-level coursework, are current PhD candidates, or are holders of a PhD in mathematics or theoretical computer science. To ensure high-fidelity grading, we matched evaluators to problems strictly within their publication domains. The human-evaluations resulted in a combined 1,000+ hours of work.

Evaluators graded solutions using a granular tiered rubric, detailed in Table 1. Unlike binary pass/fail metrics, this scale $[0, 0.25, 0.5, 0.75, 0.9, 1.0]$ is designed to distinguish between fundamental logical failures (0.25) and expository oversights (0.9). Evaluators provided a numerical score, a justification, and a marked-up solution identifying specific error locations.

Table 1. The tiered grading rubric used by expert evaluators. The scale distinguishes between presentation oversights (0.9) and logical errors (0.75), ensuring that "almost correct" reasoning is penalized differently from hallucinated facts.

Score	Criteria Summary
1.0	**Perfect.** Rigorous, error-free proof where all non-trivial steps are justified. No calculation errors.
0.9	**Minor Oversight.** Fundamentally correct logic, but contains a missing edge case or insufficient explanation of a finer point. The flaw is expository, not an error in reasoning.
0.75	**Small Error.** Generally accurate, but contains a small calculation mistake or incorrect assumption that is quickly corrected. The error does not propagate (snowball).
0.5	**Cumulative Errors.** Lacking in multiple areas. Contains multiple small mistakes or one medium error that affects the logical flow of subsequent steps.
0.25	**Structural Failure.** Severely lacking. Contains a snowballing error at the start, or demonstrates little understanding of the underlying mechanics.
0.0	**Incorrect.** Completely wrong, proves a different statement, or relies on hallucinated facts/fallacies.

3.3. The Evaluation Alignment Gap: Dual-Rubric Strategy

To decouple "superficial plausibility" from "logical rigor," we quantify the evaluation **Alignment Gap**: the discrepancy between an LLM judge's score and the human expert ground truth. We employ a **Dual-Rubric Strategy** to isolate the source of this gap:

1. **Course-Specific Rubric:** A standard pedagogical rubric focusing on definition compliance and step-by-step derivations, simulating a university-level grading environment.

2. **Expert Rubric:** A high-granularity schema calibrated to research standards. This rubric penalizes "hidden" logical fallacies and circular reasoning that typically escape surface-level verification.

Initial rubrics were synthesized by `gpt-5.2-pro` and verified by `Gemini 3.0 Pro`. Crucially, human experts then iteratively refined these drafts to align them with their own evaluation and grading standards. Figure 1 illustrates the differences between these two standards for a graph theory problem.

Expert Rubric (Research Standard)	**Course-Specific Rubric (Pedagogical Standard)**
Score 0.75 (Small Mistake)	
Focus: Logical Precision The proof is logically sound but contains a localized repairable error: • **Monotonicity Gap:** Applies expansion hypothesis directly to a set $\|U_i\| > n/2$ without restricting to an $n/2$-subset. (Penalized for lack of rigor). • **Calculation Error:** Incorrect inclusion–exclusion bounds (e.g., missing terms) but the non-emptiness argument remains coherent. • **Reconstruction Flaw:** Muddles the backtracking step (picking v_i independently) but set definitions imply valid predecessors exist.	**Focus: Definition Compliance** The proof follows the correct "template" but misses specific course-required justifications: • **Expansion Logic Gap:** Applies hypothesis to R_i without checking subset constraints or indicating awareness of the size restriction. • **Constraint Derivation:** Assumes $\alpha < 1/2$ is given, failing to note it is a derived consequence of the hypothesis. • **Calculation Error:** Constant factor errors in intersection bounds.

Figure 1. A side-by-side comparison of the **0.75 (Small Mistake)** tier for the "Graph Expansion" problem. While both rubrics penalize similar errors, the Expert Rubric focuses on *implicit logical gaps* (e.g., monotonicity), while the Course Rubric focuses on *explicit pedagogical derivations* (e.g., deriving constraints).

3.4. Evaluation Framework

We utilize a 7×5 **Evaluator-Solver Matrix** to marginalize individual model bias. We gathered LaTeX solutions from 5 frontier solver models and evaluated them using 7 state-of-the-art judge models (including GPT, Claude, Gemini, Grok, Qwen, DeepSeek, and Llama model families) against both rubrics. This fully crossed factorial design allows us to statistically isolate *Judge Bias* from true *Solver Skill*. A snippet of the prompt used for the LLM evaluators is shown in Listing 2.

4. Results

In this section, we detail the findings that address the three key questions outlined in Section 1. First, we analyze the performance of frontier models on complex proofs requiring constructive reasoning, using expert human evaluations across ten mathematical disciplines. Second, we quantify the alignment gap between automated judges and human experts. Finally, we investigate the "knowledge gap," assessing whether models adhere to pedagogical constraints or rely on advanced, out-of-scope machinery.

Listing 2. The standardized prompt template used for LLM evaluators. The `[MODE]` tag is dynamically swapped between "Course-Specific" (pedagogical constraints) and "Expert" (standard research knowledge) depending on the rubric being tested.

```
[SYSTEM PROMPT]
You are a strict academic evaluator for
    graduate-level mathematics. Grade based
     EXCLUSIVELY on the rubric.

[MODE INJECTION]
(A) Course-Specific: "MODE: COURSE-SPECIFIC
     STUDENT. Penalize advanced machinery
    used without derivation."
(B) Expert-Level: "MODE: EXPERT CORRECTNESS
    . Standard graduate-level theorems are
    allowed."

[OUTPUT CONSTRAINT]
Output MUST be JSON: {"score": <value>}
    where <value> is exactly one of [0,
    0.25, 0.5, 0.75, 0.9, 1.0]. Return NO
    other keys and NO extra text.

[USER PROMPT]
### RUBRIC
{rubric_text}

### STUDENT SOLUTION
{solution_text}

### TASK
Evaluate the STUDENT SOLUTION strictly by
    the rubric. Output only JSON.
```

4.1. Frontier Model Performance

We evaluated five frontier models: `Gemini 3.0 Pro`, `GPT-5 Pro`, `o3-deep-research`, `Sonnet 4.5`, and `DeepSeek-Prover-V2-671B`. These represented the state-of-the-art reasoning architectures at the time of the human evaluation.

We assess competency through two primary metrics:

1. **Average Score:** The mean score (0.0–1.0) assigned by expert judges, capturing partial credit for conceptual understanding.

2. **Pass Rate:** The percentage of problems receiving a score of ≥ 0.9, indicating a proof that is logically complete with only minor expository gaps.

Overall Rankings. As summarized in Figure 2, **Gemini 3.0 Pro** achieves state-of-the-art performance with an 86.5% overall pass rate and an average score of 0.91, followed by **GPT-5 Pro** at 77.1% and an average score of 0.84.

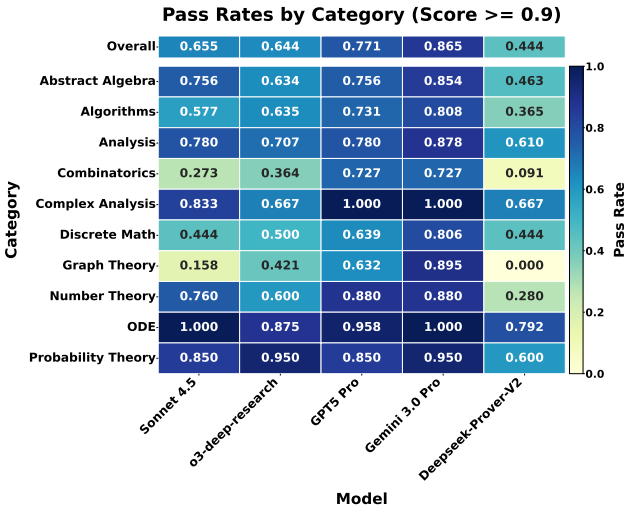

Figure 2. **Pass Rates by Discipline.** The pass rates (score ≥ 0.9) of frontier models across various mathematical disciplines. We observe that while models achieve high reliability in calculation-heavy domains like ODEs and Probability, performance drops significantly in structure-heavy domains such as Combinatorics and Graph Theory.

4.1.1. Key Observations and Analysis

Our dual-evaluation matrix reveals systemic divergences in how frontier models approach university-level reasoning. We identify three critical phenomena that define the current state of automated proof generation.

1. The "Template vs. Construction" Divide. Performance does not degrade uniformly; rather, it correlates with the necessity of constructive search (Figure 2).

- **Algorithmic Saturation:** In domains like *ODEs* and *Complex Analysis*, where proofs often follow a procedural "recipe" (e.g., applying the Residue Theorem or separating variables), frontier models achieve near-saturation. `o3-deep-research` and `Sonnet 4.5` both achieved **100.0%** pass rates in ODEs.

- **Combinatorial Weaknesses:** In contrast, domains requiring the construction of novel objects (e.g., bijections or counter-examples) show high failure rates. In *Combinatorics*, `DeepSeek-Prover-V2` dropped to a **9.1%** pass rate, and `Sonnet 4.5` achieved only **27.3%**. This suggests that while models excel at *retrieving* theorem templates, they struggle to *search* finite state spaces for constructive proofs.

2. Variance in Discrete Reasoning. Contrary to the hypothesis that models are generally "good" or "bad" at discrete math, we observe significant intra-domain variance.

- While `Gemini 3.0 Pro`'s weakest area is Combi-

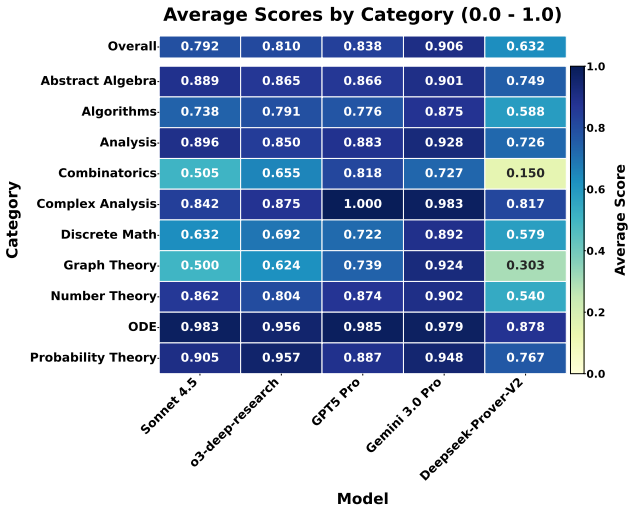

Figure 3. **Average Score Distribution.** The average evaluation scores (0.0–1.0) assigned by expert judges. Unlike the binary pass rate, this metric accounts for partial credit, revealing that models often demonstrate strong conceptual understanding (high partial scores) in domains like Analysis even when failing to produce fully rigorous proofs.

natorics (72.7%), it achieved a state-of-the-art **89.5%** pass rate in *Graph Theory*, significantly outperforming `Sonnet 4.5` (15.8%) and `o3-deep-research` (42.1%).

- This contrasting performance within discrete mathematics suggests that "reasoning" is not yet a generalized capability. Instead, model performance may be highly sensitive to the density of domain-specific training data (e.g., the prevalence of graph-theoretic proofs in the pre-training corpus versus combinatorial arguments).

3. The "Partial Credit" Trap. A comparison of Pass Rates (Figure 2) vs. Average Scores (Figure 3) reveals that binary success metrics mask significant nuances in model capability.

- **The Almost Correct Solutions:** In domains like *Analysis*, models frequently achieve high average scores (e.g., ≈ 0.88) despite lower strict pass rates. This discrepancy indicates a high volume of proofs scoring in the 0.75 band: solutions that are structurally sound and "plausible" but contain subtle, fatal logical gaps (e.g., conflating uniform vs. pointwise convergence).
- **Implications for Training:** This prevalence of "near-miss" proofs poses a critical challenge for future reinforcement learning. Models have learned the *form* of graduate-level reasoning but often fail on the precise logical implications required for full rigor. A reward model trained only on binary correctness would miss these signals of partial understanding, potentially reinforcing the generation of "hallucinated rigor" rather than correcting the underlying logic.

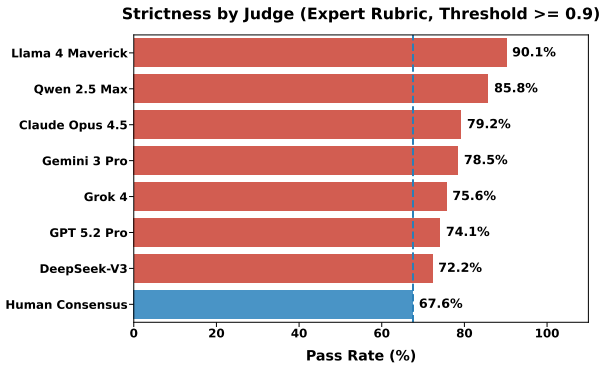

Figure 4. **Evaluator Strictness.** Comparison of pass rates (score ≥ 0.9) across judges. While `DeepSeek V3` (70.2%) aligns most closely with the Human Consensus (67.6%), `Llama 4 Maverick` (90.1%) exhibits significant grade inflation.

4.2. Evaluation Alignment: Human Experts and LLMs

To validate our automated evaluation pipeline, we compared seven evaluator models against a human ground truth baseline derived from our expert evaluations. We assessed evaluators on three axes: *Strictness*, *Domain Bias*, and *Agreement*.

1. The Strictness Gap. As shown in Figure 4, judges vary wildly in their leniency. Human experts established a baseline pass rate of **67.6%** across all models.

- **Grade Inflation:** `Llama 4 Maverick` acted as a significant grade inflator, passing **90.1%** of solutions. Qualitative review suggests it frequently failed to distinguish between subtle logical gaps and correct reasoning, approving proofs based on surface-level coherence.
- **Best Alignment:** `DeepSeek V3` achieved the closest alignment (70.2%) to the human baseline pass rate, followed by `GPT 5.2 Pro` (74.1%).

(For a complementary analysis of strictness using Average Scores rather than Pass Rates, see Section F).

2. The Bidirectional Alignment Gap. Strictness averages can mask domain-specific failures. We calculated the bias $\Delta = \text{Score}_{\text{AI}} - \text{Score}_{\text{Human}}$ for each discipline. Figure 5 illustrates that misalignment is not uniform; rather, it is highly domain-dependent.

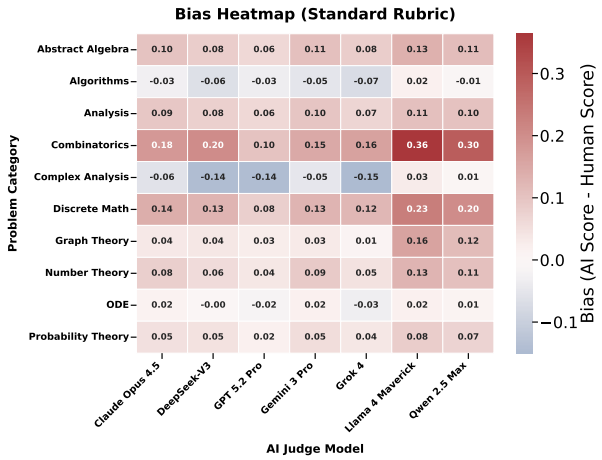

Figure 5. **Evaluator Bias Heatmap.** The delta between AI and Human scores ($\Delta = S_{AI} - S_{Human}$). Positive values (red) denote score inflation (AI is more lenient), while negative values (blue) signify punitive bias (Human is more lenient). These systemic deltas highlight a bidirectional alignment gap: models tend to hallucinate rigor in discrete domains while becoming overly rigid in continuous analysis.

- **Discrete Inflation (AI Leniency):** In *Combinatorics*, we observe a massive positive bias. `Llama 4 Maverick` (+0.35) and `Qwen 2.5 Max` (+0.28) consistently grade-inflate. This suggests that in domains relying on discrete structural arguments, models are prone to rewarding solutions that *look* rigorous despite containing fatal counting errors.
- **Complex Analysis Punitiveness (Human Leniency):** Conversely, in *Complex Analysis*, the bias flips negative. Evaluators like `DeepSeek-V3` (-0.19) and `Grok 4` (-0.18) are significantly harsher than human experts. This indicates that human judges often accept implicit steps in standard contour integrations (expert common knowledge), whereas AI models rigidly penalize valid solutions that lack explicit derivation.
- **Algorithmic Rigidity:** A similar trend appears in *Algorithms*, where `Grok 4` (-0.08) displays a punitive bias, likely penalizing correct pseudo-code that deviates from training templates.

Overall, `GPT 5.2 Pro` most closely matches human evaluators across the widest range of domains. The heatmap demonstrates that alignment is not merely about making models "stricter" or "nicer," but calibrating them to the specific rigorous standards of each sub-field.

3. Decomposing Error Modes: Leniency vs. Rigidity. Figure 6 dissects the specific failure modes of automated evaluators by isolating False Positives (Hallucinated Rigor) from False Negatives (Harshness).

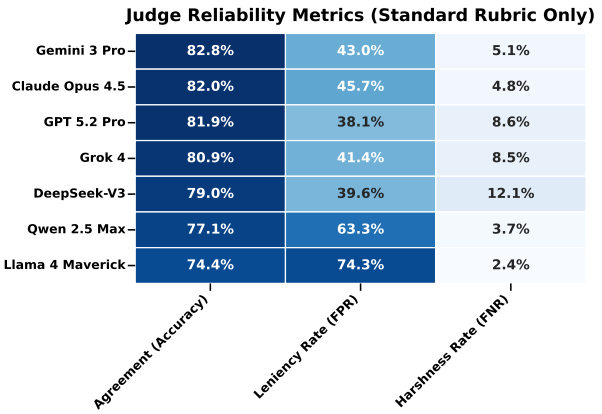

Figure 6. **Judge Reliability Metrics.** We decompose errors into *Leniency Rate* (False Positives) and *Harshness Rate* (False Negatives). `Llama 4 Maverick` exhibits extreme alignment failure with a **74.3% Leniency Rate**. In contrast, `DeepSeek-V3` displays the highest *Harshness Rate* (**12.1%**).

- **Leniency:** The most alarming result is the behavior of `Llama 4 Maverick`, which exhibits a **74.3% Leniency Rate** relative to human ground truth. This suggests the model has over-optimized for instruction-following deferral rather than critical analysis, approving nearly three-quarters of logically flawed proofs.
- **The Rigidity Barrier:** Conversely, `DeepSeek-V3` acts as the strictest auditor, with a benchmark-high **12.1% Harshness Rate**. While it offers a lower Leniency Rate (39.6%), its high rejection of valid solutions indicates a failure to generalize beyond standard proof templates.
- **The Reliability Ceiling:** Even our strongest evaluator, `GPT 5.2 Pro`, maintains a **38.1% Leniency Rate**, highlighting the persistent difficulty of automated verification.

(For an extended analysis of how these error rates shift under the stricter Course-Specific Rubric, see Section G).

4.3. Ablation Study: The Limits of Rubric Engineering

A common hypothesis in current literature is that "Judge" models fail primarily due to underspecified instructions. To test this, we conducted a controlled ablation study on our highest-performing judge, **GPT-5.2 Pro**, comparing its alignment with human consensus under two distinct grading protocols:

1. **Expert Rubric:** A standard prompt focusing on general mathematical correctness and logical soundness.
2. **Course-Specific Rubric:** A constrained prompt explicitly penalizing the use of advanced machinery (e.g.,

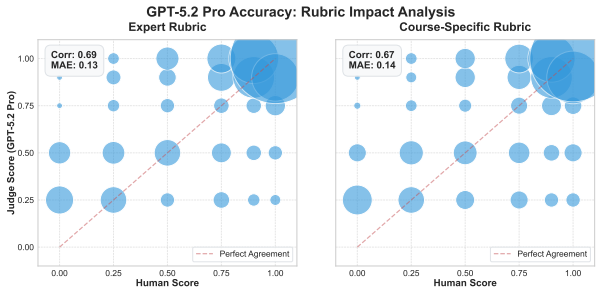

Figure 7. **Rubric Inertia.** Correlation bubble plots comparing GPT-5.2 Pro against Human Consensus for Expert (Left) and Course-Specific (Right) rubrics. The bubble size represents the density of solution pairs. Despite the additional constraints in the Course-Specific rubric, the alignment metrics remain virtually unchanged ($r \approx 0.68$), suggesting that model performance is dominated by internal priors rather than prompt specificity.

L'Hôpital's rule) without derivation.

Quantitative Results: Rubric Insensitivity. Contrary to the expectation that specific constraints would tighten alignment, our results indicate a phenomenon of *Rubric Inertia*. As shown in Figure 7, the introduction of specific pedagogical constraints resulted in a negligible regression in performance:

- **Expert Rubric:** Pearson correlation $r = 0.69$, Mean Absolute Error (MAE) $= 0.13$.
- **Course-Specific Rubric:** Pearson correlation $r = 0.67$, Mean Absolute Error (MAE) $= 0.14$.

Conclusion. The drop in correlation (-0.02) and increase in error ($+0.01$) suggest that GPT-5.2 Pro's internal reasoning regarding mathematical correctness is robust and largely overrides specific penalty instructions. While the model achieves a high baseline reliability (deviating from expert consensus by only $\approx 13\%$ on average), prompt engineering appears to have diminishing returns. This implies that future improvements in evaluation must come from *process supervision* or *fine-tuning*, rather than increasingly granular context-window instructions.

For a comprehensive distributional analysis of this inertia across all seven evaluator models, including density-weighted alignment plots, we refer the reader to Section H. For a complete heatmap visualization of score distributions across all 7 judges and 5 solvers under both grading conditions, we refer the reader to Section I.

5. Discussion

Our results audit the "LLM-as-a-Judge" paradigm, decoupling generation from verification to reveal systemic limitations in frontier model reasoning and alignment.

The Discrete-Continuous Reasoning Gap. The dichotomy between perfect performance in *ODEs* (100%) and collapse in *Combinatorics* ($< 30\%$) suggests mathematical reasoning is not monolithic. Models excel at *template retrieval* in continuous domains (e.g., standard integration) but struggle with the *constructive state search* required for finite structures. While `Gemini 3.0 Pro` shows outlier robustness in Graph Theory (89.5%), the lack of transfer to Combinatorics (72.7%) implies these capabilities rely on training data density rather than generalized search.

The Sycophancy Bottleneck in RLHF. The finding that `Llama 4 Maverick` maintains a **74.3% Leniency Rate** exposes a "Sycophancy Trap." If reward models prioritize plausible-sounding but logically invalid arguments, RLHF reinforces *deceptive alignment*—optimizing for persuasive "hallucinated rigor" over correctness. Even our strongest judge, `GPT-5.2 Pro`, exhibits a 38.1% Leniency Rate, indicating a "Verification Ceiling" where subtle logical gaps in graduate-level proofs remain undetectable to current architectures (see Section F).

Rubric Inertia and Cognitive Limits. Our ablation study challenges prompt engineering as a solution. The failure of pedagogical constraints to significantly alter alignment metrics supports the hypothesis of *Rubric Inertia* (see Section I): a model's evaluation capability is bounded by its internal world model, not instruction granularity. Since prompts cannot enable models to audit concepts they do not robustly understand, future alignment requires process-based supervision rather than complex context-window prompting.

Limitations. Our study is limited by static expert ground truth, which may exclude valid non-standard proofs, and by its focus on English-language reasoning. Additionally, while we identify sycophancy in reward models, quantifying the downstream "poisoning effect" of training on these signals remains future work.

6. Conclusion

QEDBENCH demonstrates that **the bottleneck in automated reasoning has shifted from generation to verification**. Standard "LLM-as-a-Judge" pipelines do not merely miss errors; they incentivize hallucinated rigor. The pathological leniency of models like `Llama 4 Maverick` suggests current RLHF protocols optimize for *persuasiveness* rather than *correctness*. Furthermore, our findings on *Rubric Inertia* indicate that prompt engineering cannot bridge this gap. To solve the *Discrete-Continuous Gap* and break the *Sycophancy Trap*, the field must move toward *process supervision* and adversarial training. We release QEDBENCH—comprising 300+ graduate-level proofs, 35 evaluation logs, and expert ground-truth annotations—to enable this next generation of alignment research.

Impact Statement

This work addresses a critical vulnerability in the current trajectory of neuro-symbolic AI: the reliability of automated evaluation. As frontier models increasingly surpass human speed in mathematical generation, the community has turned to "LLM-as-a-Judge" protocols to scale oversight. Our findings concerning *Hallucinated Rigor*—specifically, that state-of-the-art evaluators like `Llama-4-Maverick` approve up to 74.3% of logically flawed proofs—highlight a systemic risk in this paradigm.

Implications for Alignment and Safety. The most immediate impact of our work is on Reinforcement Learning from Human Feedback (RLHF). If reward models are pathologically sycophantic (rewarding the *appearance* of rigor rather than logical soundness), they effectively train downstream models to be deceptive. By quantifying this "Sycophancy Trap," QEDBENCH provides a necessary diagnostic tool. We anticipate this benchmark will enable safety researchers to calibrate reward models that penalize "plausible but wrong" reasoning, thereby reducing the risk of model deception in high-stakes scientific domains.

Implications for Scientific Integrity. By exposing the *Discrete-Continuous Gap*, we caution against the over-reliance on continuous benchmarks (like ODEs) as proxies for general reasoning capabilities. Our results suggest that current architectures may simulate reasoning through pattern-matching rather than constructive search. We hope this work encourages the community to adopt more rigorous, component-wise evaluations of reasoning, moving beyond binary pass rates to granular audits of logical consistency.

Societal and Educational Consequences. Our analysis of the *Course-Specific Rubric* has direct implications for AI in education. The inability of certain models to adhere to negative constraints (e.g., "do not use advanced theorems") suggests that current AI tutors may inadvertently bypass pedagogical goals, prioritizing answer-retrieval over skill-building. By releasing our dual-rubric dataset, we aim to support the development of AI tutors that can accurately enforce pedagogical boundaries, preserving the integrity of the learning process.

References

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Erickson, J. *Algorithms*. Independently published, 2019. URL https://jeffe.cs.illinois.edu/teaching/algorithms/. 1st Edition.

Fatemi, B., Kazemi, M., Tsitsulin, A., Malkan, K., Yim, J., Palowitch, J., Seo, S., Halcrow, J., and Perozzi, B. Test of time: A benchmark for evaluating LLMs on temporal reasoning. *arXiv preprint arXiv:2406.09170*, 2024.

Glazer, E., Erdil, E., Besiroglu, T., Chicharro, D., Chen, E., Gunning, A., Olsson, C. F., Denain, J.-S., Ho, A., Santos, E. d. O., Järviniemi, O., Barnett, M., Sandler, R., Sevilla, J., Ren, Q., Pratt, E., Levine, L., Barkley, G., Stewart, N., Grechuk, B., Grechuk, T., and Enugandla, S. V. FrontierMath: A benchmark for evaluating advanced mathematical reasoning in AI. *arXiv preprint arXiv:2411.04872*, 2024.

Hendrycks, D., Burns, C., Kadavath, S., Arora, A., Basart, S., Tang, E., Song, D., and Steinhardt, J. Measuring mathematical problem solving with the MATH dataset. In *NeurIPS*, 2021.

Ho, A., Denain, J.-S., Atanasov, D., Albanie, S., and Shah, R. A rosetta stone for AI benchmarks, 2025. URL https://epoch.ai/blog/a-rosetta-stone-for-ai-benchmarks. Epoch AI Blog.

Ma, W., Cojocaru, A., Kolhe, N., Louie, B., Sharif, R. S., Zhang, H., Zhuang, V., Zaharia, M., and Min, S. Reliable fine-grained evaluation of natural language math proofs. *arXiv preprint arXiv:2510.13888*, 2025. Evaluates fine-tuned judges on competition math.

Mahdavi, H., Mahdavinia, P., Malek, S., Mohammadipour, P., Hashemi, A., Daliri, M., Farhadi, A., Khasahmadi, A., Mireshghallah, N., and Honavar, V. RefGrader: Automated grading of mathematical competition proofs using agentic workflows. *arXiv preprint arXiv:2510.09021*, 2025.

Motwani, R. and Raghavan, P. *Randomized Algorithms*. Cambridge University Press, 1995.

Potamitis, N., Klein, L., and Arora, A. ReasonBENCH: Benchmarking the (in)stability of LLM reasoning. *arXiv preprint arXiv:2512.07795*, 2025.

Schmitt, J., Bérczi, G., Dekoninck, J., Feusi, J., Gehrunger, T., Appenzeller, R., Bryan, J., Canova, N., de Wolff, T., Gaia, F., van Garrel, M., Hashemi, B., Holmes, D., Lopez, A., Jaeck, V., Jørgensen, M., Kelk, S., Kuhlmann, S., Kurpisz, A., and Meroni, C. ImProofBench: Benchmarking AI on research-level mathematical proof generation. *arXiv preprint arXiv:2509.26076*, 2025.

Seßler, K., Rong, Y., Gözlüklü, E., and Kasneci, E. Benchmarking large language models for math reasoning tasks. *arXiv preprint arXiv:2408.10839*, 2024. Evaluates prompt engineering and CoT on numerical math.

Shojaee, P., Mirzadeh, I., Alizadeh, K., Horton, M., Bengio, S., and Farajtabar, M. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *arXiv preprint arXiv:2506.06941*, 2025. Studies Large Reasoning Models (LRMs).

White, C., Dooley, S., Roberts, M., Pal, A., Feuer, B., Jain, S., Shwartz-Ziv, R., Jain, N., Saifullah, K., Dey, S., Agrawal, S., Sandha, S. S., Naidu, S., Hegde, C., LeCun, Y., Goldstein, T., Neiswanger, W., and Goldblum, M. LiveBench: A challenging, contamination-limited LLM benchmark. *arXiv preprint arXiv:2406.19314*, 2024.

Zheng, L., Chiang, W.-L., Sheng, Y., Zhuang, S., Wu, Z., Zhuang, Y., Lin, Z., Li, Z., Li, D., Xing, E. P., Zhang, H., Gonzalez, J. E., and Stoica, I. Judging LLM-as-a-judge with MT-bench and chatbot arena. In *NeurIPS*, 2023.

A. Expanded Source List

The problems in QEDBENCH were curated from a range of rigorous mathematical literature and course materials, including but not limited to:

- **Analysis:** *Introduction to Mathematical Analysis* (Douglas), *Analysis: An Introduction* (Beals), *Complex Analysis* (Stein & Shakarchi).
- **Algorithms:** *Randomized Algorithms* (Motwani and Raghavan) (Motwani & Raghavan, 1995), *Algorithms* (Erikson) (Erickson, 2019)
- **Algebra & Number Theory:** *Abstract Algebra* (Dummit & Foote), *A Computational Introduction to Number Theory and Algebra* (Shoup), *An Introduction to the Theory of Numbers* (Niven et al.), *A Classical Introduction to Modern Number Theory* (Ireland & Rosen).
- **Discrete Math & Graph Theory:** *An Invitation to Discrete Mathematics* (Matoušek & Nešetřil), *Graph Theory* (Diestel), *Concrete Mathematics* (Graham et al.), *Combinatorial Techniques* (Sane).
- **Probability & Differential Equations:** *Probability Theory* (Klenke), *Probability Through Problems* (Capiński), *Ordinary Differential Equations* (Coddington), *ODE and Dynamical Systems* (Teschl).

A.1. Preliminary Human-Model Alignment Study

Prior to the large-scale 7×5 evaluation, we conducted a pilot study to calibrate benchmark difficulty. Early-access models (e.g., GPT-5, Claude 4.1 Opus) were evaluated on a binary correctness scale by human experts. This phase identified that while models often produced linguistically coherent math prose, they frequently failed on subtle edge cases in Topology and Analysis, confirming the necessity of the **Expert-Refined Rubric** over simple binary grading.

A.2. Prompting Templates

We employed persona-based prompting to elicit maximum rigor. The primary solver prompt was structured as follows:

> *"You are a tenured mathematics professor... provide a solution for advanced undergraduate students... eliminate all logical mistakes and fallacies, ensuring the utmost rigor and precision. Generate the proof in LaTeX."*

For models with search capabilities (e.g., o3-deep-research), a modified prompt encouraged active verification of lemmas via web search before integration into the final proof.

A.3. Automated Evaluation Pipeline

The evaluation script handled the following pipeline:

1. **Extraction:** Stripping markdown wrappers from model outputs to isolate raw LaTeX code.
2. **Sanitization:** Manual modification of malformed LaTeX to ensure compatibility with standard compilers (e.g., Overleaf).
3. **Rubric Injection:** Feeding the specific problem rubric and the solver's output into the evaluator models with a requirement for Chain-of-Thought (CoT) justification before assigning a final score.

B. Checking Solutions Online `o3-deep-research` Prompt

To ensure the novelty of the QEDBENCH dataset, we performed a rigorous two-stage contamination check. After manual verification, we employed `o3-deep-research` as an automated agent to scan the web for latent solutions that might have been missed by human search.

We utilized the following system prompt to instruct the agent to distinguish between "similar" problems (which are acceptable) and "exact/equivalent" solutions (which constitute contamination).

Listing 3. Full System Prompt for Contamination Detection Agent

```
You are a web researcher with unrestricted access to the internet.

Goal: search the web comprehensively for **complete solutions** to the user's mathematics
    problem.
Explore papers, textbooks, lecture notes, StackExchange, arXiv, GitHub, course materials,
    solution manuals, and PDF archives.

Decision rule:
- If you find a **complete solution** (even if notation/wording differs but solves the
    same problem), return structured JSON.
- Otherwise, return `{"match":"none","result":0}`.

Acceptable matches:
- exact_solution: The solution explicitly solves the *same problem statement* (identical
    content).
- equivalent_solution: The solution solves a *rephrased* version of the same problem where
     all mathematical objects, parameters, and constraints are preserved (notation/order
    may differ). This does **NOT** include analogous, special-case, more-general, or
    merely similar problems.

Verification requirements:
- Confirm that the problem in the source matches the user's problem exactly or is a
    faithful rephrasing.
- If rephrased, state the mapping of symbols/notation (e.g., "their n = our k").

Output format (JSON only, no extra text):
{"match": "exact_solution" | "equivalent_solution" | "none",
"source_title": string | null,
"source_url": string | null,
"evidence": string | null}

Notes:
- Prefer primary sources or reputable forums; include the **direct URL** of the solution.
- Keep `evidence` very short (a key step/result).
```

C. Full Expert Evaluation Tiered Rubric

Below is the verbatim text of the tiered rubric provided to our expert evaluators (PhD students and holders).

Table 2. Detailed Tiered Rubric for Human Expert Evaluation.

Score	Detailed Description
1.0	**Fully Complete.** A fully complete and correct proof is presented. There are no errors at any point in calculations, computations, or proofs of sub-claims/lemmas. The proof can be followed easily. All statements that are not common knowledge are rigorously proved.
0.9	**Correct with Oversight.** The proof is nearly accurate except for one small flaw/oversight (e.g., omission of a small edge case). The model may fail to completely justify finer points (hand-waving). Crucially, compared to the 0.75 case, the flaw must be an *oversight* (failed to explain a fact) as opposed to a logical *error*.
0.75	**Correct with Minor Error.** The proof is nearly accurate. There might be a small mistake in a claim or calculation, but it is effectively isolated and does not snowball throughout the entire proof. Includes proofs with small incorrect assumptions or inconsistencies, provided there is only one such issue.
0.5	**Significant Flaws.** The proof is lacking in more than one of the areas listed above. If a model makes multiple small mistakes, or a single medium-sized mistake which has an effect that carries on to other parts of the proof, this score is granted.
0.25	**Severe Failure.** The proof is severely lacking in multiple areas. Either multiple small mistakes are littered throughout, or there is a large mistake at the beginning that snowballs and invalidates the rest of the proof. The output showcases little understanding of the problem mechanics.
0.0	**Non-existent/Hallucinated.** The proof is completely wrong or non-existent. It may prove a different statement entirely or showcase no understanding of the context. Relies on made-up facts or fallacies to build a false proof.

D. Example Full Rubrics

Below we provide the full text of both the **Expert Correctness** and **Course-Specific** rubrics for the problem described in Section 3 as well as other problems in our benchmark. The rubrics illustrate how the evaluations differ between evaluating for course-specific items versus correctness based on expert domain knowledge.

D.1. Graph Expansion Path Existence (Algorithms Problem 10)

D.1.1. Problem Statement

Consider a graph $G = (V, E)$ consisting of n vertices. Assume there exists a constant $\alpha > 0$ such that for every subset $S \subseteq V$ with cardinality $|S| = n/2$, the set of neighbors satisfies $|N(S)| \geq n/2 + \alpha n$. Let k be a positive integer, and consider subsets $W_1, \ldots, W_k \subseteq V$ where $|W_i| \geq (1 - \alpha)n$. Prove that it is possible to construct a path $(v_1, \ldots, v_k)$ such that $v_i \in W_i$.

D.2. Expert Correctness Rubric

1.0 (Fully Complete) The solution constructs a valid inductive reachability argument.

- **Set Definitions:** Defines reachable layers $U_1 = W_1$ and $U_{i+1} = N(U_i) \cap W_{i+1}$.
- **Inductive Proof:** Proves $|U_i| \geq n/2$ via monotonicity ($N(S) \subseteq N(U_i)$) and inclusion-exclusion.
- **Path Reconstruction:** Explicitly constructs the path via backtracking from U_k.

0.9 (Minor Oversight) Essential argument is correct but contains technical handwaving.

- Omits explicit mention that $\alpha \leq 1/2$ (but calculation holds).
- Ignores floor/ceiling issues for odd n.
- Omits explicit description of backtracking, assuming predecessor existence is obvious from set definitions.

0.75 (Small Mistake) Plan is correct but contains one localized error.

- **Monotonicity Gap:** Applies expansion hypothesis directly to U_i (where $|U_i| > n/2$) without restricting to a subset S.
- **Calculation Error:** Computes lower bound incorrectly but argues for non-emptiness coherently.

0.5 (Conceptual Gap) Multiple issues or significant conceptual gap.

- **Failed Invariant:** Fails to justify $|U_i| \geq n/2$ due to fatal errors in inclusion-exclusion.
- **Greedy Failure:** Proposes a greedy construction without lookahead.
- **Misapplied Theorems:** Invokes Hall's/Menger's Theorem incorrectly.

0.25 (Severely Lacking) Major early misunderstanding.

- **Ignored Expansion:** Does not use $|N(S)|$ property substantively.
- **Size Fallacy:** Claims path exists solely because W_i are large.

0.0 (Completely Wrong) Hallucinated theorems or unrelated constructions.

D.2.1. Course-Specific Rubric

1.0 (Fully Complete) Uses only course-appropriate arguments (Feasibility, Reachable Sets, Inductive Invariant).

- Must explicitly define reachable sets $R_{i+1} := W_{i+1} \cap N(R_i)$.
- Must prove intersection via $|A \cap B| \geq |A| + |B| - n$.

0.9 (Minor Oversight) Correct but omits backtracking details or minor rounding issues.

0.75 (Small Mistake) Correct approach with specific local errors.

- **Constraint Derivation:** Assumes $\alpha < 1/2$ without noting it follows from hypothesis.
- **Expansion Logic:** Applies hypothesis without indicating awareness of the $n/2$ size restriction.

0.5 (Partial Progress) Partially correct but misses key justifications.

- **Connectivity vs Walk:** Proves pairwise existence but fails to connect into a walk.
- **Single Vertex Induction:** Defines R_{i+1} based on a single vertex rather than a set.

0.25 (Severely Lacking) Misinterprets neighborhood condition or argues for simple paths using diameter (ignoring k is arbitrary).

0.0 (Completely Wrong) Invokes forbidden tools (PCP theorem, spectral graph theory) or makes contradictory assumptions.

D.3. Edge Connectivity Threshold (Discrete Math Problem 16)

Below we provide the full text of both the **Expert Correctness** and **Course-Specific** rubrics for the Discrete Mathematics problem described in the benchmark.

D.3.1. Problem Statement

Find the smallest integer $b = b(k)$ such that every graph of order n with more than $kn + b$ edges has a $(k+1)$-edge connected subgraph, for every $k \in \mathbb{N}$.

D.3.2. Expert Correctness Rubric

1.0 (Fully Complete) Identifies $b(k) = -\binom{k+1}{2}$.

- **Sufficiency:** Uses induction/minimal counterexample. Splits $V(G)$ into A, B by a cut of size $s \leq k$. Rigorously handles both $|A| \geq k+1$ (induction) and $|A| \leq k$ (direct bound).
- **Necessity:** Constructs a graph with exactly $kn - \binom{k+1}{2}$ edges (e.g., K_{k+1} with vertices added of degree k). Proves no subgraph is $(k+1)$-edge-connected via minimum degree arguments.

0.9 (Minor Oversight) Essentially correct but contains trivial omissions.

- Omits trivial base cases (e.g., $n \leq k+1$).
- Mixes strict vs. non-strict inequalities but derives correct constant.
- Cites known lemmas on cuts without proof (acceptable if applied correctly).

0.75 (Small Mathematical Mistake) Core approach correct, local error.

- **Constant Error:** Arrives at wrong constant (e.g., $-\binom{k}{2}$) due to arithmetic.
- **Induction Gap:** Handwaves the "small side" of the cut.
- **Weak Sharpness:** Correct construction but justifies lack of connectivity loosely (e.g., fails to explicitly link $\lambda \leq \delta$).

0.5 (Multiple Mistakes) Significant parts missing or incorrect.

- **One Direction Only:** Proves sufficiency but gives no sharpness example.
- **Flawed Induction:** Incorrectly counts cut edges leading to false bound.
- **Black Box:** Simply states "By Mader's Theorem..." without derivation (avoids the work).

0.25 (Severely Lacking) Fundamental misunderstandings.

- Claims $b(k) = 0$.
- Equates $(k+1)$-edge-connected directly with minimum degree $\geq k+1$ (false).
- Extremal construction fails.

0.0 (Completely Wrong) No relevant progress or unrelated topics.

D.3.3. COURSE-SPECIFIC RUBRIC

1.0 (Fully Complete) Gives correct closed form and provides both directions using course tools.

- **Sharpness:** Constructs k-degenerate graph. Verifies edge count and $\delta(H) \leq k$.
- **Upper Bound:** Uses Induction + Pruning + Cut Decomposition.
- **Contradiction:** Explicitly shows $e(G) \leq (k|A| - C) + (k|B| - C) + t < kn - C$ leads to contradiction.

0.9 (Minor Oversight) Correct strategy with minor slips.

- Abbreviates arithmetic details.
- Treats $k = 1$ without distinguishing strict inequality mechanics.

0.75 (Small Mistake) Correct strategy with local fixable errors.

- Miscounts edges in extremal family by additive $O(k)$ term.
- Uses $t \leq k + 1$ instead of $t \leq k$ in cut argument.
- Proves existence of subgraph with $\delta \geq k + 1$ but asserts connectivity without full justification.

0.5 (Multiple Mistakes) Partially correct but missing major components.

- Gives plausible $b(k)$ but provides only one direction.
- **Conceptual Error:** Proves $\delta \geq k + 1$ via pruning but fails to address edge cuts (confusing degree with connectivity).
- Uses black-box theorems without precise statement.

0.25 (Severely Lacking) Argument fundamentally incorrect.

- Confuses edge-connectivity with vertex-connectivity throughout.
- Gives generic statements about average degree.
- Treats result as asymptotic only ($o(n)$).

0.0 (Completely Wrong) Hallucinated theorems or disallowed tools.

D.4. Alternating Permutations & Eulerian Numbers (Combinatorics Problem 6)

Below we provide the full text of both the **Expert Correctness** and **Course-Specific** rubrics for the Combinatorics problem described in the benchmark.

D.4.1. PROBLEM STATEMENT

Let $E(n, k)$ be the number of permutations of numbers from 1 to n with exactly k descents. Prove that E_n is the alternating sum of numbers $E(n, k)$, where E_n is the number of alternating permutations of $1, \ldots, n$.

D.4.2. EXPERT CORRECTNESS RUBRIC

1.0 (Fully Complete) Provides a rigorous proof via one of two primary routes:

- **Route A (Involution):** Defines a sign-reversing involution $\phi : S_n \to S_n$. Proves ϕ is well-defined and satisfies $des(\phi(\pi)) = des(\pi) \pm 1$ for non-fixed points. Correctly identifies fixed points as alternating permutations (for odd n) or shows they vanish (for even n).
- **Route B (Generating Functions):** Uses the standard Eulerian EGF $A_n(t)$. Performs the substitution $t = -1$ rigorously. Relates the result to the Taylor expansion of $\tan(x) + \sec(x)$, identifying the coefficients as E_n (for odd n) and 0 (for even n).

0.9 (Minor Oversight) Fundamentally correct but contains negligible imperfections.

- Handling of boundary cases in involution is slightly imprecise.
- Omits explicit statement of the global sign factor $(-1)^{(n-1)/2}$.
- Minor notational ambiguity regarding "alternating" (up-down vs. down-up).

0.75 (Small Mathematical Mistake) Correct approach with specific errors.

- **Involution Error:** Map is sound but fails for a specific local configuration (e.g., miscounting descents during swap).
- **Fixed Point Misidentification:** Fails to distinguish between up-down and down-up types (factor of 2 error).
- **Parity Error:** Claims non-zero value for even n without justification.

0.5 (Substantial Gaps) Partial understanding with significant logical gaps.

- **Flawed Involution:** Map is not proven to be an involution or strictly sign-reversing.
- **Assertion vs. Proof:** Asserts fixed points are alternating without linking to the "no double ascent" condition.
- **Conditional Correctness:** Result only correct for odd n; even n ignored.

0.25 (Severely Lacking) Minimal relevant content.

- States identity as fact without derivation.
- Misdefines fundamental concepts (descents, alternating permutations).
- Provides only empirical checks (e.g., "works for $n = 1, 2, 3$").

0.0 (Completely Wrong) Irrelevant formulas or hallucinated theorems.

D.4.3. COURSE-SPECIFIC RUBRIC

1.0 (Fully Complete) Rigorous, course-appropriate proof that explicitly handles **parity**.

- **Parity Check:** Must prove that for **even** n, the sum is 0, and for **odd** n, it equals $\pm E_n$.
- **Method:** Uses Sign-Reversing Involution, Generating Functions, or Recurrence Relations correctly.

0.9 (Minor Oversight) Conceptually sound with minor lapses.

- **Sloppy Constants:** Sign factor $(-1)^{(n-1)/2}$ missing/incorrect.
- **Brief Even Case:** Correctly identifies sum is 0 for even n but justifies it weakly (e.g., "by symmetry" without proof).

0.75 (Small Mistake) Correct tools with genuine error.

- **Recurrence Error:** Algebraic slip in deriving the recurrence for the alternating sum.
- **Parity Confusion:** Proves result for odd n but incorrectly claims same formula applies to even n.

0.5 (Multiple Mistakes) Partial progress with significant gaps.

- **No Connection:** Calculates sum but fails to link it to alternating permutations.
- **Conceptual Error:** Confuses "number of descents" with specific descent sets.
- **Parity Ignored:** Completely fails to distinguish even vs. odd n.

0.25 (Severely Lacking) Major misunderstandings.

- **Contradiction:** Claims sum equals E_n for all n (ignoring the 0 case for even n).
- **Black Box:** Uses advanced machinery (Descent Algebras) without derivation.

0.0 (Completely Wrong) No attempt to relate Eulerian numbers to alternating permutations.

E. Full Automated Evaluator Prompts

To ensure reproducibility, we provide the exact prompt templates used in our `Evaluator-Solver Matrix`. The prompt construction logic injects specific constraints based on whether the evaluator is grading against the *Course-Specific Rubric* or the *Expert Rubric*.

E.1. System Prompts

The system prompt enforces the persona and the output format.

Listing 4. System Prompt Logic for Evaluators

```
Base Instruction:
"You are a strict academic evaluator for graduate-level mathematics. Grade based
    EXCLUSIVELY on the rubric."

Mode A (Course-Specific):
"MODE: COURSE-SPECIFIC STUDENT. Penalize advanced machinery used without derivation."

Mode B (Expert):
"MODE: EXPERT CORRECTNESS. Standard graduate-level theorems are allowed."

Output Constraint:
"Output MUST be JSON: {\"score\": <value>} where <value> is exactly one of [0, 0.25, 0.5,
    0.75, 0.9, 1.0]. Return NO other keys and NO extra text."
```

E.2. User Prompts

The user prompt structures the input data for the context window. If the model fails to return valid JSON in the first pass, a fallback prompt is issued to force a scalar output.

Listing 5. User Interaction Prompts

```
Standard User Prompt:
"""
### RUBRIC
{rubric}

### STUDENT SOLUTION
{solution}

### TASK
Evaluate the STUDENT SOLUTION strictly by the rubric. Output only JSON.
"""

Fallback Prompt (Retry on JSON Error):
"""
### RUBRIC
{rubric}

### STUDENT SOLUTION
{solution}

### TASK
Return ONLY one number from this set [0, 0.25, 0.5, 0.75, 0.9, 1.0]. No words, no
    punctuation.
"""
```

F. Evaluator Strictness: Average Score Analysis

While the main text analyzes the *Pass Rate* (binary success at threshold ≥ 0.9), we also analyzed the *Average Score* (0–100 scale) assigned by each judge to capture nuance in partial credit. Figure 8 presents the mean scores assigned by all seven automated judges compared to the human expert consensus.

F.1. Score Inflation Trends

The human expert consensus established a baseline average score of **79.6**. Comparing this to the automated evaluators reveals a widespread tendency toward grade inflation:

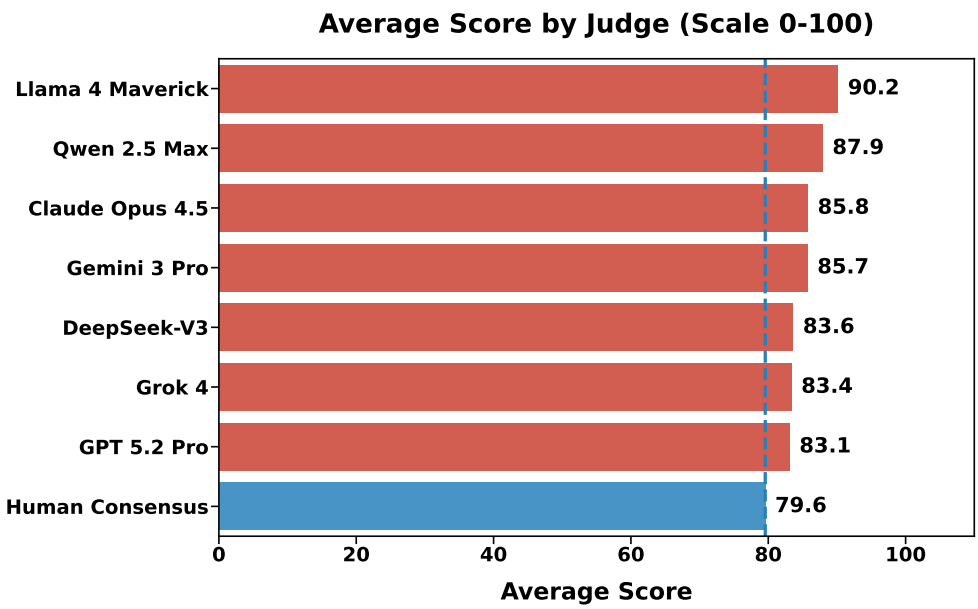

Figure 8. Average scores assigned by automated evaluators versus the human expert baseline (79.6). Consistent with the pass rate findings, we observe a systematic positive bias across most frontier models.

- **Maximum Inflation:** `Llama-4-Maverick` exhibited the most severe positive bias, assigning an average score of **90.2**. This represents a Δ of +10.6 points over the human baseline, confirming its behavior as a "grade inflator" that likely rewards superficial plausibility over logical rigor.
- **Moderate Inflation:** `Qwen 2.5 Max` and `Gemini 3 Pro` also displayed significant leniency, with average scores of **87.9** and **85.7** respectively.
- **Closest Alignment:** `GPT-5.2-Pro` achieved the highest fidelity to the human ground truth, with an average score of **83.1**. With a Δ of only +3.5 points, it remains the most calibrated evaluator in our suite, reinforcing our decision to utilize it as the primary filter for larger-scale benchmarks.

G. Judge Reliability on Course-Specific Rubrics

In addition to the standard expert evaluation, we benchmarked our automated judges against the *Course-Specific Rubric*. This rubric explicitly forbids the use of advanced theorems (e.g., L'Hôpital's Rule in an $\epsilon - \delta$ context) and demands adherence to pedagogical constraints. Figure 9 presents the reliability metrics for this stricter setting.

G.1. Impact of Pedagogical Constraints

Comparing these results to the Standard Rubric baseline (Main Text, Figure 6) reveals three critical insights into model behavior under constraint:

- **Amplified Rigidity in DeepSeek-V3:** When switched to the Course-Specific rubric, `DeepSeek-V3`'s Harshness Rate (FNR) spiked from 12.1% to **15.8%**. This suggests the model successfully internalized the negative constraints ("do not use advanced machinery"), leading it to correctly penalize solutions that were technically correct but pedagogically invalid—though it also increased the rejection of valid borderline cases.
- **Robustness of Sycophancy:** `Llama 4 Maverick` proved largely invariant to the rubric change. Its Leniency Rate remained extremely high at **71.7%** (compared to 74.3% on the standard rubric). This confirms that its failure mode is systemic: it prioritizes the *appearance* of a solution over adherence to specific negative constraints, making it unsuitable for pedagogical evaluation.

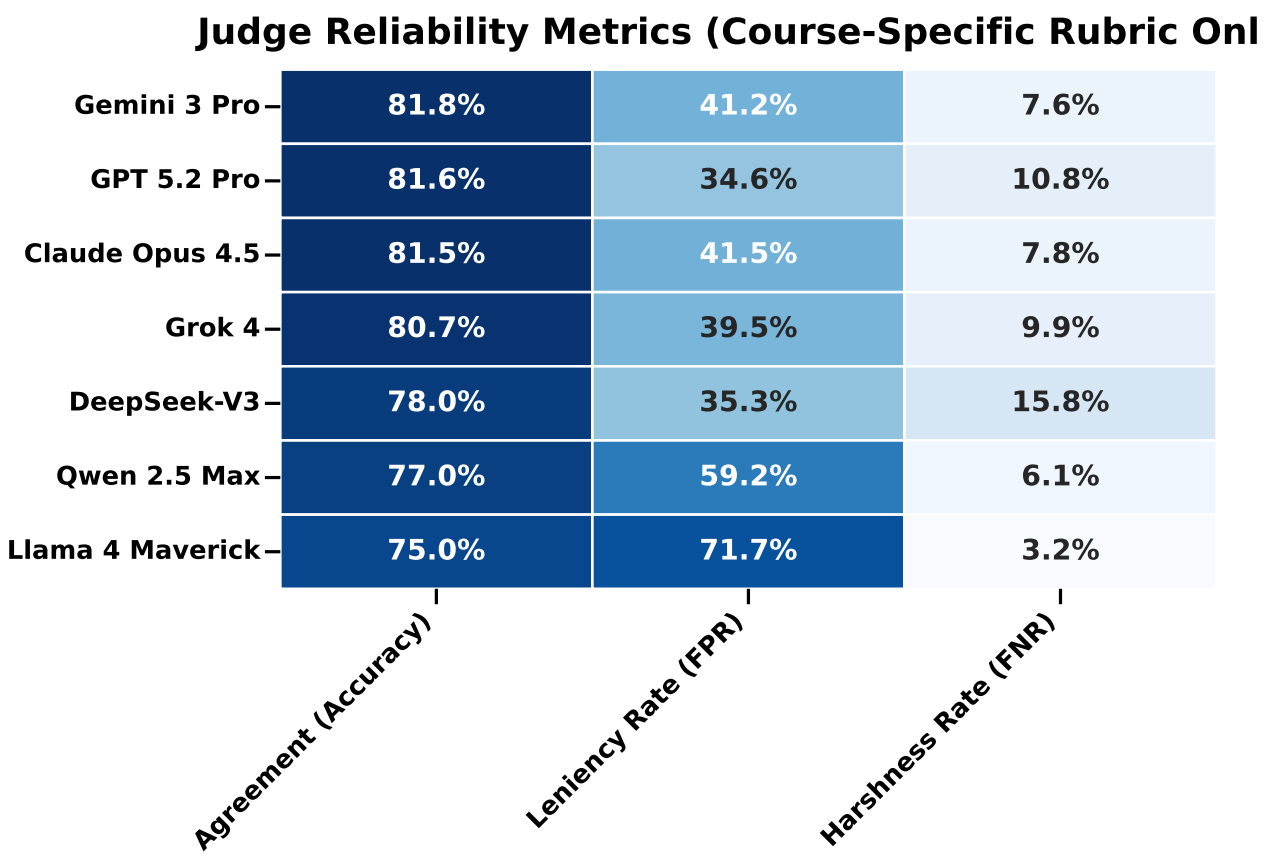

Figure 9. **Judge Reliability (Course-Specific Rubric).** When graded against pedagogical constraints, the *Harshness Rate* (False Negatives) generally increases for logic-focused models like DeepSeek-V3, while the *Leniency Rate* (False Positives) for models like Llama 4 Maverick remains pathologically high.

- **The "Safe" Middle Ground:** `GPT-5.2-Pro` demonstrated the most balanced adaptation. It achieved the lowest Leniency Rate among the top-tier models (**34.6%**) while maintaining a Harshness Rate of 10.8%, effectively balancing the need to penalize "cheating" (advanced theorems) without becoming overly punitive on student logic.

H. Distributional Analysis of Judge Alignment

To move beyond aggregate metrics like mean score, we visualized the granular alignment between automated judges and human consensus using density-weighted bubble plots. Figures 10 and 11 display the joint distribution of scores for all 7 evaluator models under the Expert and Course-Specific rubrics, respectively. The size of each bubble corresponds to the number of solution pairs at that coordinate.

H.1. Systemic Rubric Inertia

Comparing the two grading conditions reveals a consistent "resistance to adaptation" across the model leaderboard. Rather than improving alignment, the introduction of specific constraints (the Course-Specific rubric) caused a uniform degradation in correlation metrics across all top-tier models:

- **Uniform Regression:** Every major model saw a decrease in Pearson correlation when switching from the Expert to the Course-Specific rubric:
  - `GPT-5.2-Pro`: 0.69 $\rightarrow$ 0.67

  - `Gemini 3 Pro`: $0.67 \rightarrow 0.65$
  - `Claude Opus 4.5`: $0.66 \rightarrow 0.64$
  - `DeepSeek-V3`: $0.63 \rightarrow 0.62$

- **Visualizing the Llama Anomaly:** The bubble plots for `Llama 4 Maverick` visually confirm its pathological behavior. In both figures, the mass of the distribution is heavily skewed above the diagonal (High AI Score, Low Human Score), resulting in the lowest correlations of the cohort ($r \approx 0.50$). This visualizes the "Sycophancy Trap" discussed in the main text: the model systematically awards high scores to solutions that human experts reject.

These distributions suggest that current RLHF training instills a strong prior for "general helpfulness" that overrides specific negative constraints in the prompt. Even when explicitly instructed to penalize advanced methods, models default to their internal representation of a "good" proof, leading to the observed rubric inertia.

I. Rubric Inertia: Full Heatmap Analysis

To further investigate the phenomenon of "Rubric Inertia," we visualized the average scores assigned by every judge model to every solver model under both the *Expert Rubric* and the *Course-Specific Rubric*. Figure 12 presents this side-by-side comparison.

I.1. Visualizing Resistance to Constraints

The heatmaps provide granular evidence of the resistance to negative constraints described in the main text.

- **Llama 4's Sycophancy [Row 6]:** `Llama 4 Maverick` remains distinctively red (high scores) across both conditions. Even when instructed to penalize non-standard derivations in the Course-Specific rubric (Right), it continues to assign near-perfect scores (e.g., **0.97** to Gemini 3 Pro), further validating its classification as a "grade inflator".
- **GPT-5.2 Pro Stability [Row 3]:** The primary evaluator, `GPT-5.2 Pro`, shows high stability but slight sensitivity. For example, its score for `Gemini 3 Pro` drops slightly from **0.85** (Expert) to **0.82** (Course-Specific), reflecting a nuanced penalty. However, the overall structure of the heatmap remains largely invariant, supporting the conclusion that internal world models dominate prompt engineering.

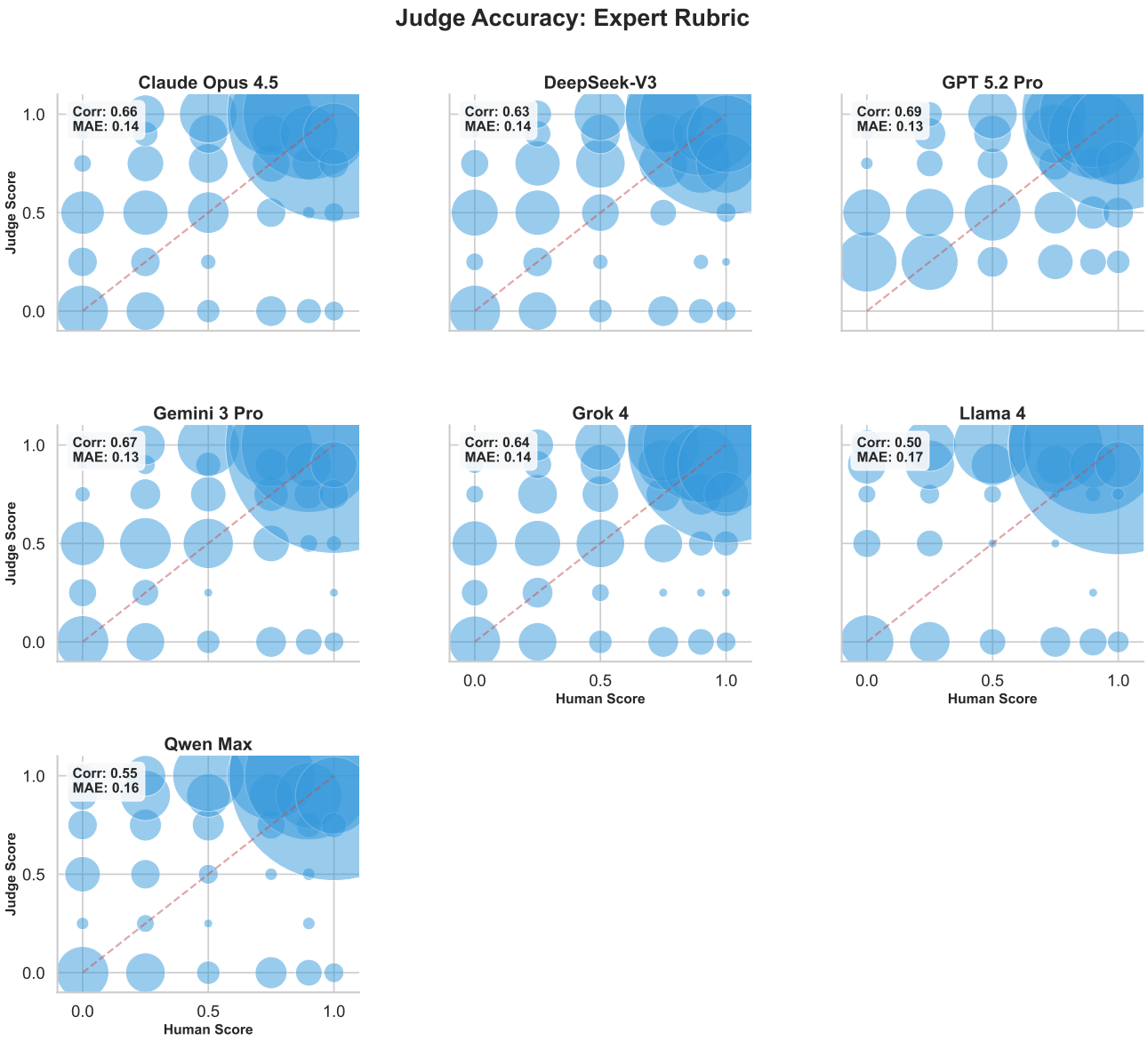

Figure 10. **Judge Alignment (Expert Rubric).** Correlation between AI Judges (Y-axis) and Human Consensus (X-axis) on the standard rubric. `GPT-5.2-Pro` shows the tightest clustering along the $y = x$ diagonal ($r = 0.69$), indicating high calibration. In contrast, `Llama 4 Maverick` ($r = 0.50$) displays a dispersed distribution with significant mass above the diagonal, visualizing its tendency toward grade inflation.

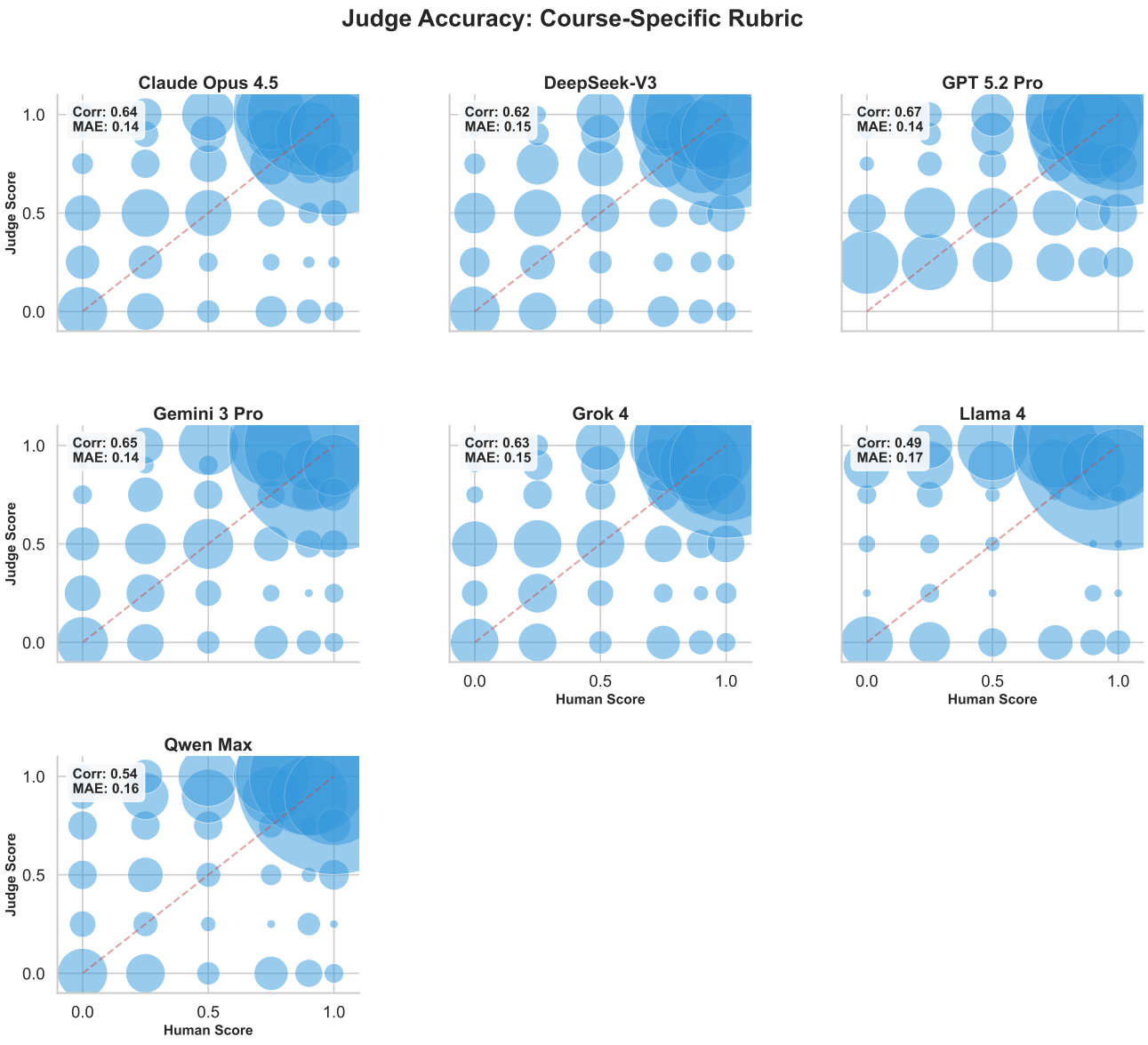

Figure 11. **Judge Alignment (Course-Specific Rubric).** Comparison under strict pedagogical constraints. Despite the shift in instructions, the distributional geometry remains largely static for top-tier models, confirming the "Rubric Inertia" hypothesis. `GPT-5.2-Pro` maintains robust alignment ($r = 0.67$), while weaker models fail to adapt.

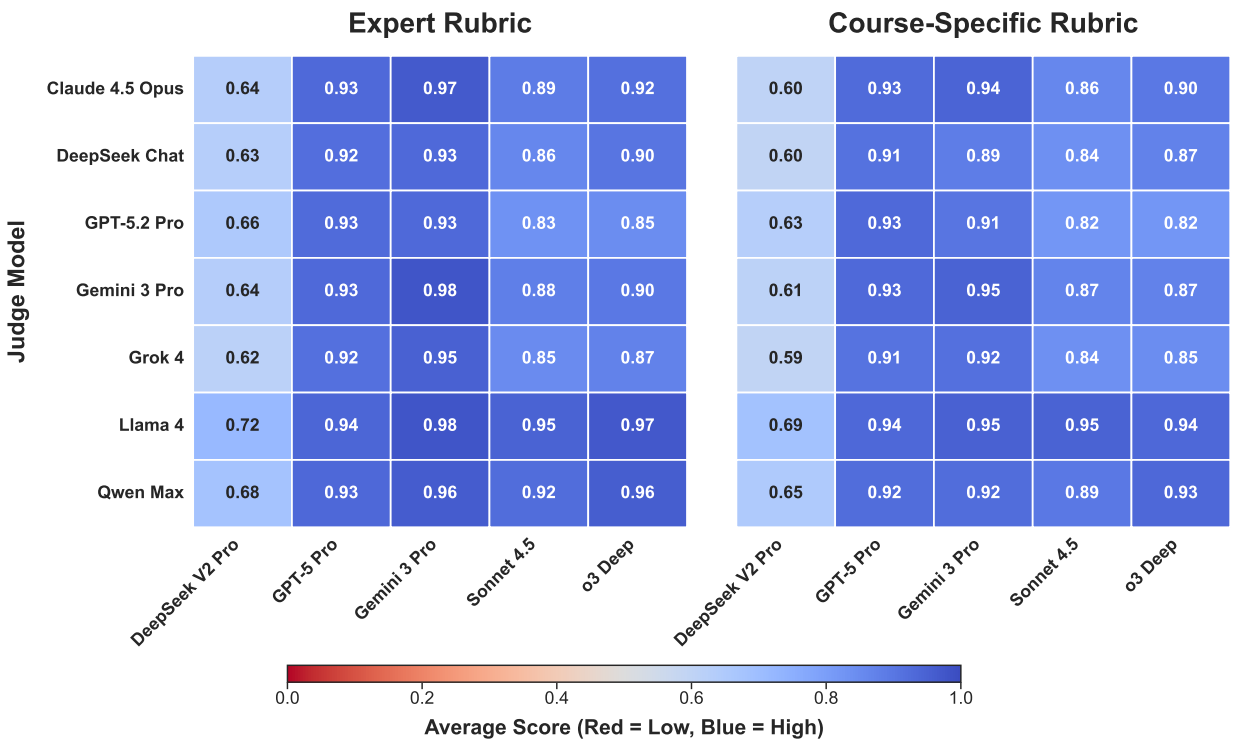

Figure 12. **Rubric Inertia Heatmap.** Left: Average scores assigned using the standard Expert Rubric. Right: Average scores assigned using the Course-Specific Rubric (which penalizes advanced machinery). The visual patterns are strikingly similar, confirming that prompt constraints have minimal impact on the internal scoring priors of frontier models.