LLMs Are Not Intelligent Thinkers: Introducing Mathematical Topic Tree Benchmark for Comprehensive Evaluation of LLMs

Anonymous ACL submission

Abstract

Large language models (LLMs) demonstrate impressive capabilities in mathematical reasoning. However, despite these achievements, current evaluations are mostly limited to specific mathematical topics, and it remains unclear whether LLMs are genuinely engaging in reasoning. To address these gaps, we present the **Ma**thematical **T**opics **T**ree (MaTT) benchmark, a challenging and structured benchmark that offers 1,958 questions across a wide array of mathematical subjects, each paired with a detailed hierarchical chain of topics. Upon assessing different LLMs using the MaTT benchmark, we find that GPT-4 achieved a mere 54% accuracy in a multiple-choice scenario. Interestingly, even when employing Chain-of-Thought prompting, we observe mostly no notable improvement. Moreover, LLMs accuracy dramatically reduced by up to 24.2 percentage point when the questions were presented without providing choices. Further detailed analysis of the LLMs' performance across a range of topics showed significant discrepancy even for closely related subtopics within the same general mathematical area. In an effort to pinpoint the reasons behind LLMs performances, we conducted a manual evaluation of the completeness and correctness of the explanations generated by GPT-4 when choices were available. Surprisingly, we find that in only 53.3% of the instances where the model provided a correct answer, the accompanying explanations were deemed complete and accurate, i.e., the model engaged in genuine reasoning.

1 Introduction

Large Language Models (LLMs) have increasingly demonstrated remarkable capabilities as mathematical reasoners, underscoring their potential in complex problem-solving domains (Chowdhery et al., 2022; Touvron et al., 2023; OpenAI, 2023; Team et al., 2023). Recent studies have shown that LLMs, when applied to mathematical problems, can exhibit a high degree of reasoning ability, often aligning with or even surpassing human-level performance in certain contexts. This proficiency in mathematical reasoning is further enhanced by innovative techniques such as Chain-of-Thought (Wei et al., 2022), Tree-of-Thought (Yao et al., 2024), and Self-Verification (Weng et al., 2022), emphasizing on the importance of the procedural steps in solving a mathematical problems.

Despite these advancements, several critical gaps persist in our understanding of LLMs' mathematical reasoning capabilities. Firstly, it remains unclear which specific areas of mathematics LLMs excel or falter in, as comprehensive evaluations across diverse mathematical domains are lacking. Secondly, distinguishing between instances where LLMs rely on memorization versus genuine reasoning is challenging, raising questions about the depth of their understanding. Thirdly, the influence of multiple-choice formats on LLM behavior is not well understood, suggesting that models' performance might be affected by the structure of the questions posed. These gaps underscore the necessity for a more robust benchmark that facilitates a holistic evaluation of LLMs, enabling us to dissect their strengths, weaknesses, and the nuances of their problem-solving strategies.

In this paper, we developed the Mathematical Topics Tree (MaTT) benchmark by initially leveraging Wikipedia's "Lists of mathematics topics"[1] to identify key areas in mathematics, resulting in twelve major topics that span both pure and applied mathematics. This was followed by extracting important reference books for each topic from Wikipedia to build a detailed topical tree. We then further refine the benchmark by using the books' tables of contents to structure a comprehensive tree reflecting the hierarchical organization of mathe-

[1] https://en.wikipedia.org/wiki/Lists_of_mathematics_topics

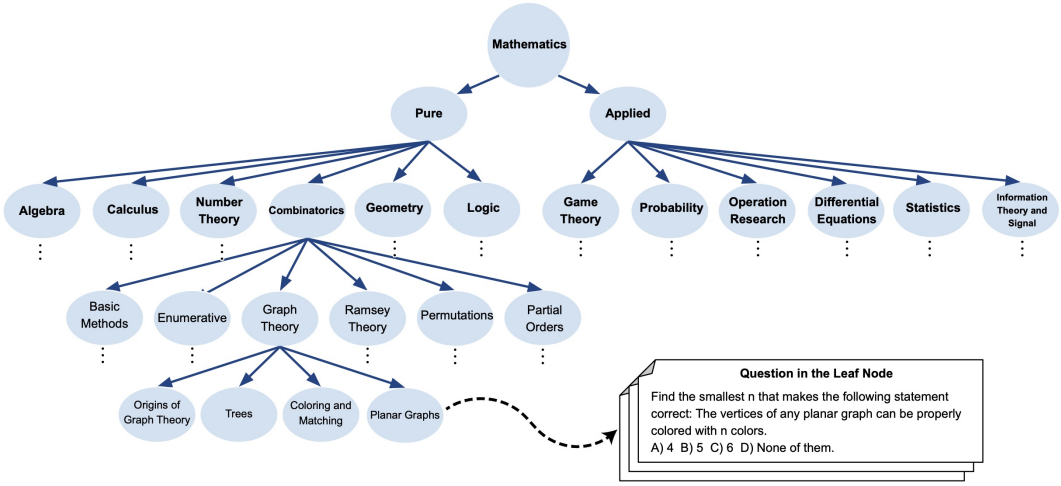

Figure 1: Overview of **Ma**thematical **T**opics **T**ree (MaTT) benchmark, a challenging and structured benchmark that presents questions spanning a diverse range of mathematical subjects, each associated with a detailed hierarchical structure of topics.

matical knowledge. Upon completing the topic tree, we extracted questions from the subsections of these books, and gathered them under leaf nodes. Finally, we pair each question with multiple-choice options, enhancing the benchmark's utility for evaluating mathematical understanding. An illustration of MaTT is depicted in Figure 1.

After developing MaTT, we evaluate the mathematical reasoning capabilities of various LLMs, including commercial models OpenAI o1-mini, GPT-4 (OpenAI, 2023) and ChatGPT (Kocoń et al., 2023) (turbo versions), alongside the open-source LLMs, Llama3.1-70B (Touvron et al., 2023) and Mistral (Jiang et al., 2023). Notably, GPT-4 achieved only 54% accuracy in a multiple-choice format. Furthermore, the use of Chain-of-Thought prompting mostly did not enhance LLMs' performance, underscoring the benchmark's complexity and suggesting that mere step-by-step reasoning might be insufficient. Also, When questions were presented without multiple-choice options, LLMs accuracy dropped by up to 24.2 points. Our analysis also revealed notable discrepancies in performance across topics, showing inconsistent abilities even within related subtopics of the same mathematical domain.

To understand the underlying causes of the LLMs' inadequate performance and their inconsistent results across various topics, we did a detailed evaluation of the explanations provided by GPT-4. Surprisingly, we observe that only in 53.3% of cases where the models answered correctly, the explanations were also complete, i.e., GPT-4 engaged in genuine reasoning. These cases were typically associated with simpler or more well-known questions that required only a few straightforward steps to resolve. For more complex questions demanding either more number of steps, complicated calculations, or creative/intelligent problem-solving, GPT-4 often failed, resorting to tactics like choice engineering, unsupported theorems, circular reasoning, or memorization instead of true reasoning.

2 MATT: Mathematical Topics Tree Benchmark

In recent years, LLMs have shown remarkable abilities in mathematical reasoning. Yet, their prowess is not fully understood due to the narrow focus of current benchmarks, which typically concentrate on specific mathematical areas. This limitation hinders our understanding of the depth and breadth of LLMs' reasoning capabilities. There's a pressing need for more comprehensive mathematical benchmarks that cover a wider array of topics and offer deeper insights into the models' reasoning processes. Such benchmarks would not only challenge the models across a broader mathematical spectrum but also can help with better understanding the nuances of how and where these models apply reasoning.

To address this gap, we create the Mathematical Topics Tree (MaTT) benchmark. We start by harnessing the "Lists of mathematics topics" available on Wikipedia as a foundational resource. This exploration was crucial for identifying the spectrum of mathematical knowledge we aimed to encompass. Extracting the list of mathematics topics from Wikipedia, we identified twelve principal top-

ics that comprehensively encapsulate the breadth of pure and applied mathematics. Then, for each topic, we extracted one or few key reference books listed on their respective Wikipedia pages. The topics and their corresponding resources are as follows: for pure math we consider Algebra (Meyer, 2023; Herstein, 1991; McGee, 2002), Calculus and Analysis (Stewart, 2012), Number Theory (Niven et al., 1991), Combinatorics (Bóna, 2002), Geometry and Topology (Coxeter, 1969; Coxeter and Greitzer, 1967; Engelking, 1989), and Logic (Mendelson, 2009). In applied math we have Game Theory (Osborne and Rubinstein, 1994), Probability (Tijms, 2012, 2017), Operations Research (Hillier and Lieberman, 2015), Differential Equations (Boyce et al., 2021), Statistics (Hogg et al., 2013), and Information Theory and Signal Processing (Cover, 1999; Proakis, 2007).

Next, we used the tables of contents from selected reference books to structure the MaTT topical tree, mapping the hierarchy of topics and subtopics to create a comprehensive graph of mathematical domains. The final step in the creation of MaTT involved a detailed extraction of questions from the sections of the reference books, gathering them under the leaf nodes within our topic tree. For each question identified, we then crafted multiple-choice options to facilitate an objective assessment framework. To generate the options, we selected choices that closely resembled the actual answer, such as those with similar numerical values, those attain by omitting a step from the proof, or those presenting alternative combinations. For instance, if the correct answer was "A & B", we included "A or B" as one of the possible choices. We provide an illustration of MaTT in Figure 1.

The statistical overview of the MaTT benchmark is detailed in Table 1. The benchmark comprises 1,958 examples, meticulously curated across 12 distinct mathematical topics that span the breadth of pure and applied mathematics. In assembling these questions, we aimed to ensure a broad yet consistent spectrum of difficulty across all topics. While extracting questions, we exclude questions that are overly popular/simplistic or had their responses provided in the books immediately following them to mitigate the risk of data contamination.

3 Experimental details

We assessed the performance of commercial LLMs—OpenAI o1-mini, GPT-4 (OpenAI, 2023) and ChatGPT (Kocoń et al., 2023) (turbo versions)—alongside the open-source LLMs Llama3.1-70B (Touvron et al., 2023) and Mistral (Jiang et al., 2023) (Mistral-7B-Instruct-v0.2), using the MaTT benchmark. In our evaluation, we structured the prompts to request that LLMs first generate an explanation and then the final answer. In the multiple-choice setting, we specifically directed the models to select one of the provided options (A, B, C, or D) as their final answer. Additionally, for zero-shot chain-of-thought prompting, we appended "let's think step by step" to the prompt. Examples of the prompts utilized in our experiments are provided in the Appendix A.

4 Experiments

In this section, we begin with an analysis of LLMs' mathematical reasoning capabilities using the MaTT benchmark. Subsequently, we examine the variation in model performance across different sub-topics. We then assess the effect of choice availability by presenting MATT questions to LLMs without multiple-choice options. Lastly, we concentrate on GPT-4's explanations, manually annotating the level of reasoning in each explanation and exploring the strategies employed by GPT-4 to arrive at correct answers.

4.1 LLMs Performance on MaTT

We present the accuracy of LLMs on the MATT benchmark across various topics in Table 2. The performances of models are notably low (except for o1-mini), with GPT-4 achieving only about 54% accuracy and Mistral performing close to the random choice selection. A detailed examination reveals that Mistral often refuses to answer, claiming the correct option is not listed, while other models try to select the closest match or reason with the available choices when their answer is missing.

Additionally, there is a significant variance in the accuracy levels of LLMs across different topics, with gap as high as 41.2%, highlighting a significant level of difference in understanding and reasoning capability of LLMs across various mathematical areas. Moreover, despite demonstrating strong performance on existing math benchmarks, o1-mini shows room for improvement in the multiple-choice setting of MaTT. Furthermore, a breakdown of o1-mini's performance by topic reveals that it excels in areas requiring straightforward problem-solving steps, such as Algebra,

	Topics	# Nodes	# Leaf	# Qs	# Avg leaf's Qs
Pure Math	Algebra	69	49	120	2.45
	Calculus and Analysis	137	115	517	4.50
	Number Theory	37	31	126	4.06
	Combinatorics	19	15	139	9.27
	Geometry and Topology	93	81	159	1.96
	Logic	23	18	35	1.94
Applied Math	Game Theory	23	15	35	2.33
	Probability	113	91	276	3.03
	Operations Research	64	53	104	1.96
	Differential Equations	70	60	157	2.62
	Statistics	56	48	109	2.27
	Information Theory and Signal Processing	69	50	181	3.62
	All	772	625	1958	3.13

Table 1: Data Statistics of MaTT.

Calculus and Analysis, and Number Theory. However, it has more room for improvement in topics demanding higher levels of creativity, such as Combinatorics, Logic, and Game Theory. Additionally, for topics like Operations Research, Statistics, Information Theory and Signal Processing, we suspect that o1-mini was trained on a smaller amount of data, which may contribute to its lower performance in these areas. Finally, we observe that zero-shot CoT prompting mostly did not enhance model performance, potentially due to the complexity of the questions. Many of question in MaTT, require intricate/numerous steps or necessitate intelligent/creative thinking, which cannot be addressed by merely following a few simple steps (For further discussion on other reasoning/prompting strategies, e.g, Program of Thought (Wang et al., 2022), refer to Appendix C.). This observation raises questions about the assumption that CoT prompting is effective in many reasoning tasks. Many available evaluation benchmarks on reasoning tasks are designed to be solved in a few straightforward steps (Srivastava et al., 2022), whereas real-world reasoning often involves many steps and requires creative problem-solving.

4.2 Per-Topic Break Down of LLMs Performance

As highlighted in the previous section, the exploration of LLMs' capabilities in mathematical reasoning across a diverse array of topics or distinct sub-topics within the same mathematical domain remains significantly unexplored. We detail the LLMs' accuracy on sub-topics within the MATT benchmark in Figures 2 for pure mathematics and 3 for applied mathematics, respectively. (We provide the performance breakdown of Llama3.1 and o1-mini in Appendix D.)

These figures reveal that the models display varying levels of accuracy even within sub-topics of the same main topic, emphasizing the differences in their understanding and reasoning capabilities even across closely related subjects. Notably, we find that in certain sub-topics, such as application of integration, parametric equations, quadratic reciprocity, diophantine equation, duality theory, non-linear programming, conditional probability, continuous-time Markov chains, and basic statistics, ChatGPT and Mistral outperform GPT-4. This observation further underscores the significance of going beyond the overall performance on high-level topics and instead examining model performance on a more granular level to understand their mathematical reasoning skills comprehensively.

4.3 LLMs Performance without Providing Choices

To delve deeper into the mathematical reasoning abilities of LLMs, we assessed their performance on the MaTT benchmark without the aid of multiple-choice options. We manually evaluated the models' accuracy on MaTT in the absence of choices and provided the results in Table 3. The findings indicate a substantial decrease in performance, with GPT-4, ChatGPT, and Mistral loosing 29.4%, 56.4%, and 69.7% of the accuracy they achieved when choices were available, respectively. This significant decline underscores the models' dependency on choices for deriving answers and highlights their limitations in genuine mathematical reasoning. It also stresses the importance of not solely relying on a single overall score to evaluate LLMs' reasoning capabilities. We provide more detailed analysis on the impact of availability of

	Topics	GPT-4		ChatGPT		Mistral		Llama3.1		o1-mini
		w/o CoT	w CoT	w/o CoT	w CoT	w/o CoT	w CoT	w/o CoT	w CoT	w/o CoT
Pure Math	Algebra	71.1	73.6	45.5	52.1	33.9	39.7	65.8	69.2	89.7
	Calculus	52.2	50.9	41.6	42.6	19.3	19.3	50.6	52.7	88.3
	Number Theory	52.4	50.0	54.0	47.6	22.2	23.8	53.7	53.1	84.9
	Combinatorics	52.1	55.6	45.1	40.8	21.8	19.0	51.4	50.6	73.1
	Geometry	53.8	53.8	51.9	50.0	26.3	27.5	56.0	61.1	71.9
	Logic	62.9	65.7	31.4	34.3	34.3	28.6	55.8	61.7	79.4
Applied Math	Game Theory	40.0	40.0	31.4	45.7	14.3	20.0	54.2	48.5	48.5
	Probability	50.5	46.2	36.5	37.9	20.2	17.6	54.3	54.3	75.0
	OR	40.6	45.3	37.7	30.2	22.6	24.5	41.3	50.0	66.3
	Differential	53.5	52.2	41.5	43.4	18.9	16.3	53.5	57.9	78.9
	Statistics	63.3	59.6	56.9	52.3	28.4	23.9	66.9	72.4	77.9
	Info and Signal	59.3	53.3	38.2	38.2	29.1	26.6	50.8	61.3	75.1
	All	54.0	52.7	42.9	42.7	23.1	22.5	53.5	56.6	79.2

Table 2: Accuracy of LLMs over MaTT benchmark (for accuracy with confidence interval refer to Appendix B).

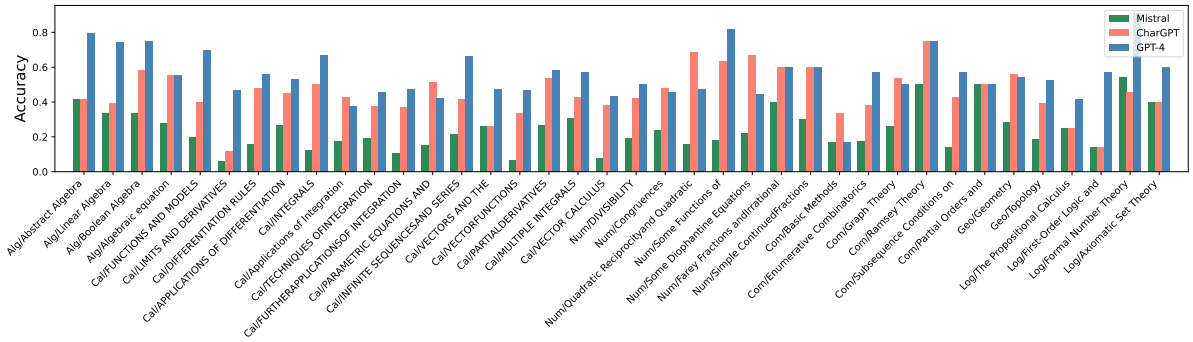

Figure 2: Per-topic breakdown for pure Math.

choices on LLMs prediction in Section 4.5.

4.4 Reasoning Level of the Explanations

To understand the reasons behind the poor performance of LLMs without providing choices and their varying accuracy across different topics, we manually examined the completeness and accuracy of LLM-generated explanation for their predictions. Given GPT-4's relatively superior performance compared to other evaluated LLMs, our analysis in this section is specifically focused on the explanations generated by GPT-4. Our objective is to identify the percentage of explanations in correctly predicted instances (when choices are available) for each of the following categories: (1) *complete* reasoning, where the explanation is thorough and logical; (2) *choice/weak* reasoning, where the model uses strategies such as leveraging the given options or offers partial reasoning; and (3) *no/wrong* reasoning, where the explanation is incorrect or missing, and the model reaches a conclusion without justification. Additionally, we calculated the percentage of instances (from all cases where GPT-4 answered correctly with choices) in which GPT-4, with *no choice*, still provided a correct answer and delivered a *complete* explanation.

The results of our manual evaluation of explanations for samples where GPT-4 (when choices are available) predicts the correct answer are detailed in Table 4. Remarkably, we found that only 53.3% of the explanations for correctly answered questions were complete, i.e., GPT-4 engaged in actual reasoning, highlighting a significant inconsistency in GPT-4's actual reasoning abilities. Also, we observe varying levels of explanation completeness across different topics, which do not necessarily correlate with GPT-4's overall performance in those topics. When comparing samples with complete explanations both with and without choices, we notice a significant gap, underscoring that the presence of choices aids the model in better navigating or recalling the reasoning process. Furthermore, we note that GPT-4 genuinely engaged in reasoning primarily for simpler or more well-known questions that could be solved through a few straightforward steps, whereas it struggled with questions requiring more complex steps or creative problem-solving, often resorting to different strate-

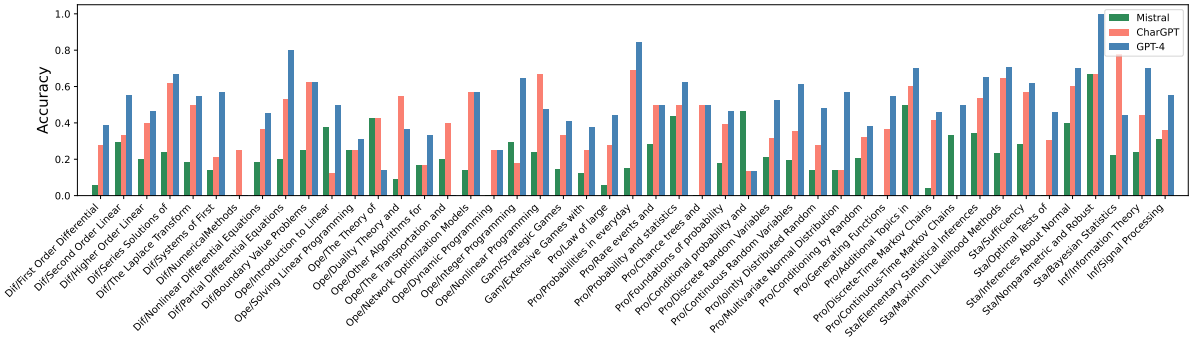

Figure 3: Per-topic breakdown for applied Math.

gies (we explore these strategies in more detail in Section 4.5). This aligns with the observed limited effectiveness of Chain-of-Thought prompting in enhancing the performance of LLMs. We provide more analysis on explanations in the Appendix E.

4.5 Observations from Explanations

Besides annotating the reasoning level of explanations (as presented in Table 4), we also pinpoint the strategies GPT-4 employs to arrive at correct answers, which do not involve reasoning. We summarise these strategies as follows:

Choice engineering refers to the strategy where a model, such as GPT-4, manipulates or exploits the available multiple-choice options to determine an answer, rather than relying on a deep understanding or genuine reasoning process. This can be divided to the following cases:

- **Choices use:** In this case, GPT-4 directly uses the choices and chooses the one matching the question the best. For example, in linear programming questions, despite GPT-4 without choice could not answer any of the optimization problems, when choices were available, using this strategy, GPT-4 achieves a high performance on those questions by simply choosing the minimum or maximum values among the choices.

- **Deducing a plausible answer:** In this strategy, instead of actual reasoning, GPT-4 tries to choose the answer by removing choices that are not plausible answers for the question. For a better understanding, consider the following question: Generate X which has a beta distribution with parameters α and β. **GPT4's Answer**: "Option B incorrectly raises U_1 and U_2 to the powers of α and β, respectively. This does not correspond to any standard method of generating beta-distributed variables and does not make intuitive sense in the context of the properties of the beta distribution... ."' GPT-4 provides similar arguments for the other options and correctly derive the answer but without any actually reasoning.

- **Choice expert:** GPT4 seems to have an understanding of how the choices are usually made. For example consider this question: what are the probabilities of events X and Y? A)1/3,13/27 B)1/3,1/3 C)1/2,1/2 D)None of them. GPT4 was only able to derive the probability of event X to be 1/3, and without any reasoning claimed that probability of event Y could not be 1/3, and then chose A. Note that we also have the choice "None of them".

- **Middle ground rule:** We observed that GPT4 tends to choose the middle ground if it cannot find the correct answer. For example: what is the expected duration of the game (which is described in the question and is omitted here)? **GPT4's answer**: "Choices A) 16.519 and C) 22.955 seem more plausible, but without the exact calculation or simulation, it's challenging to pinpoint the exact expected duration. However, B) 19.737 strikes a balance between ... Without the ability to perform a detailed calculation, the most reasonable choice would be: Answer: B) 19.737."

Theorem Use refers to the strategy where a model, such as GPT-4, utilizes a theorem or property which is the actual and main challenging part of the question. Therefore, by applying it directly, it sidesteps complex derivations and reaches the solution without explaining or proving why that is correct. In order to better grasp the idea, consider the following question and answer from GPT4: What

	Topics	GPT-4	ChatGPT	Mistral
Pure Math	Algebra	63.6 (-7.5)	32.5 (-13.0)	16.9 (-17.0)
	Calculus and Analysis	49.7 (-2.5)	23.1 (-18.5)	7.3 (-12.0)
	Number Theory	26.5 (-25.9)	19.6 (-34.4)	6.3 (-15.9)
	Combinatorics	43.4 (-8.7)	25.4 (-19.7)	6.6 (-15.2)
	Geometry and Topology	40.8 (-13.0)	34.9 (-17.0)	10.9 (-15.4)
	Logic	60.7 (-2.2)	17.9 (-13.5)	14.3 (-20.0)
Applied Math	Game Theory	22.6 (-17.4)	22.6 (-8.8)	9.7 (-4.6)
	Probability	32.8 (-17.7)	12.3 (-24.2)	6.3 (-13.9)
	Operations Research	15.9 (-24.7)	6.9 (-30.8)	5.0 (-17.6)
	Differential Equations	25.0 (-28.5)	8.3 (-33.2)	4.5 (-14.4)
	Statistics	38.1 (-25.2)	12.3 (-44.6)	2.1 (-26.3)
	Info and Signal	28.3 (-31.0)	12.1 (-26.1)	5.2 (-23.9)
	All	38.1 (-15.9)	18.7 (-24.2)	7.0 (-16.1)

Table 3: LLMs accuracy in answering questions without providing choices. We demonstrate the decrease in LLMs' performance when choices are not provided, compared to when they are, in red.

	Topics	Complete	Choice/Weak	No/Wrong	No Choice Complete
Pure Math	Algebra	80.5	13.8	5.7	43.7
	Calculus and Analysis	79.6	10.4	10.0	66.2
	Number Theory	26.9	40.3	32.8	26.9
	Combinatorics	33.3	45.3	21.3	30.7
	Geometry and Topology	20.0	52.9	27.1	15.3
	Logic	72.7	27.3	0.0	54.5
Applied Math	Game Theory	28.6	35.7	35.7	21.4
	Probability	40.0	37.9	22.1	32.1
	Operations Research	21.4	28.6	50.0	16.7
	Differential Equations	40.0	27.0	32.9	30.6
	Statistics	43.5	40.6	15.9	34.8
	Info and Signal	68.6	16.2	15.2	42.9
	All	53.3	27.4	19.3	40.7

Table 4: Level of reasoning for explanations in instances where GPT-4's answers were correct, when the choices were available. We report the percentage of explanations with **complete**, **choice/weak**, or **no/wrong** reasoning. We also present the percentage of explanations that exhibited **complete** reasoning when choices were not provided.

is determinant of Vandermonde Matrix [2]? **GPT4's Answer**: "The determinant of a Vandermonde matrix is given by $\prod_{0 \leq i < j \leq n}(x_j - x_i)$."

Circular reasoning is a logical fallacy in which the conclusion of an argument is used as a premise to support itself. This strategy is closely related to hallucination (Huang et al., 2023) which can be very challenging to detect. In these cases, the reasoning starts and ends at the same point, with the argument essentially saying, "A is true because B is true while B is true as A is true". For example, let us consider the following question: The Fermat point of a triangle, is a point such that the sum of the three distances from each of the three vertices of the triangle to the point is the smallest possible. What is Fermat point for the triangle having an angle ≥ 120? **GPT4's answer**: "The Fermat point is defined for the purpose of minimizing the total distance from the three vertices of the triangle to this point... When one of the angles of the triangle is greater than or equal to 120 degrees, the Fermat point coincides with the vertex of the triangle that has the angle $\geq$ 120 degrees. This is because, in such a case, positioning the point at the obtuse-angled vertex minimizes the total distance from the point to the other vertices of the triangle." What GPT4 did was a prime example of circular reasoning and in order to see why, let us dig into it: let's define the statements A="triangle having an angle ≥ 120", B="a vertex being a Fermat point" and C="a vertex minimizing the total distance from the three vertices of the triangle". B and C are equivalent by definition and $A \rightarrow B$ is almost as hard as $A \rightarrow C$. Here, GPT4 instead of proving $A \rightarrow B$ asserts $A \rightarrow C$ without any proof, and the concludes $A \rightarrow B$. Not so surprisingly, we observe that GPT-4 frequently uses this strategy in

[2]Vandermonde Matrix is a $n \times n$ matrix with coefficient $(x_i)^j$ at the ith row and jth column for arbitrary values of x_i.

Geometry and Topology, as shown by the high percentage of cases involving choice/weak reasoning in Table 4.

Blind memorization refers to LLMs recalling an answer without following the necessary steps or reasoning to reach it. Let us further illustrate this with following question: Find the expected value of the distance between two points that are chosen at random inside the unit square, unit circle, and equilateral triangle with sides of unit length. For the square, here is **GPT4's answer**: "...The exact solution to this problem involves a double integral that accounts for the distances between all pairs of points within the square. The calculation is somewhat complex but well-documented in mathematical literature. The correct answer, derived from these calculations, is approximately 0.521." For the circle, here is **GPT4's answer**: "...The exact solution to this problem involves complex calculus and is not trivial to derive without advanced mathematical tools. However, it is known from mathematical studies that the expected distance is approximately $128/(45\pi)$, which is about 0.905... ." Finally, for the triangle, **GPT4's answered** as follows: "...the exact solution to this problem is not straightforward and requires a detailed calculation that involves the geometry of the equilateral triangle and the properties of distances within it...". These examples show how blind memorization prevents an LLM from reasoning and answering somewhat similar questions.

5 Related Work

As LLMs grow stronger, they exhibit significant mathematical reasoning capabilities on existing benchmarks. However, the scope of current evaluation settings is restricted in terms of the breadth of mathematical areas covered and fails to conclusively determine whether these models genuinely engage in reasoning or rely on alternate strategies.

Mathematical Benchmarks Previous research primarily concentrated on developing benchmarks for math word problems—mathematical problems in the form of written description–which typically require only a few steps to solve, often involving basic arithmetic or elementary algebra (Ling et al., 2017; Cobbe et al., 2021; Patel et al., 2021). Additionally, the work in Mishra et al. (2022) introduced a comprehensive mathematical reasoning benchmark that encompasses 23 varied tasks across four dimensions: mathematical abilities, language format, language diversity, and external knowledge. Furthermore, Zhang et al. (2023) presented a multi-modal benchmark with a focus on geometry. The most relevant to our study are the MATH (Hendrycks et al., 2021) and Theoremqa (Chen et al., 2023) benchmarks. Despite providing mathematical questions on various topics, they have a much narrower scope compared to our benchmark and did not provide a detailed topical breakdown for each question. Additionally, a recent effort (Toshniwal et al., 2024) has begun to generate large-scale synthetic mathematical benchmarks for instruction tuning of LLMs.

LLMs and Math In recent years, LLMs have shown notable achievements in mathematical reasoning (Srivastava et al., 2022; Liu et al., 2023). These accomplishments are supported by methods aimed at enhancing LLMs' performance, predominantly through decomposed reasoning. Such strategies, inspired by human problem-solving processes, include providing step-by-step guidance (Wei et al., 2022; Yao et al., 2024; Besta et al., 2023), employing verification mechanisms to enhance model consistency and accuracy (Weng et al., 2022), and incorporating complex strategies (Qi et al., 2023).

6 Conclusion

In this paper, we provide a comprehensive evaluation on mathematical reasoning of LLMs. We create the Mathematical Topics Tree (MaTT) benchmark, a systematically organized set of questions covering a wide range of mathematical subjects linked to a detailed topic hierarchy. Exploring LLMs accuracy on MaTT, we observe their struggle with a wide range of mathematical topics, particularly when deprived of multiple-choice options. We also observe the discrepancy in LLMs' performance across various topics and the lack of substantial improvement with Chain-of-Thought prompting. To investigate the gaps in models performances, we manually analysis their explanations in answering the questions. We find that in only 53.3% of the instances where GPT-4 provided a correct answer, the accompanying explanations were deemed complete. Further, we observe that models perform better on simpler problems but rely on alternative strategies for complex ones. This indicates a fundamental gap in LLMs' ability to engage in deep, creative, and complex mathematical thinking. We will release code and data for MaTT.

7 Limitations

This study presents several limitations that should be considered when interpreting the findings. Firstly, our evaluation of mathematical reasoning capabilities was conducted on only five widely adopted LLMs using the MATT benchmark. This limited selection of models may not fully represent the diverse capabilities of LLMs. Including a wider range of models in future assessments could provide a more comprehensive understanding of LLMs' mathematical reasoning across various architectures and training paradigms.

Secondly, our methodology for assessing models' reasoning capabilities heavily relied on analyzing their self-generated explanations. While this approach allows us to gauge how models rationalize their answers, it inherently carries potential biases and inaccuracies. The explanations provided by LLMs might not always accurately reflect the underlying reasoning processes and could sometimes be misleading or incomplete. More objective or diverse methods of evaluation might be necessary to gain a clearer and more accurate picture of how LLMs process and solve mathematical problems.

References

Maciej Besta, Nils Blach, Ales Kubicek, Robert Gerstenberger, Lukas Gianinazzi, Joanna Gajda, Tomasz Lehmann, Michal Podstawski, Hubert Niewiadomski, Piotr Nyczyk, et al. 2023. Graph of thoughts: Solving elaborate problems with large language models. *arXiv preprint arXiv:2308.09687*.

Miklós Bóna. 2002. *A walk through combinatorics: an introduction to enumeration and graph theory*. World Scientific.

William E Boyce, Richard C DiPrima, and Douglas B Meade. 2021. *Elementary differential equations and boundary value problems*. John Wiley & Sons.

Wenhu Chen, Ming Yin, Max Ku, Pan Lu, Yixin Wan, Xueguang Ma, Jianyu Xu, Xinyi Wang, and Tony Xia. 2023. Theoremqa: A theorem-driven question answering dataset. In *The 2023 Conference on Empirical Methods in Natural Language Processing*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Thomas M Cover. 1999. *Elements of information theory*. John Wiley & Sons.

Harold Scott Macdonald Coxeter. 1969. *Introduction to geometry*. John Wiley & Sons, Inc.

Harold Scott Macdonald Coxeter and Samuel L Greitzer. 1967. *Geometry revisited*, volume 19. Maa.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Ryszard Engelking. 1989. General topology. *Sigma series in pure mathematics*, 6.

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul Arora, Steven Basart, Eric Tang, Dawn Song, and Jacob Steinhardt. 2021. Measuring mathematical problem solving with the math dataset. *arXiv preprint arXiv:2103.03874*.

Israel Nathan Herstein. 1991. *Topics in algebra*. John Wiley & Sons.

Frederick S Hillier and Gerald J Lieberman. 2015. *Introduction to operations research*. McGraw-Hill.

Robert V Hogg, Joseph W McKean, Allen T Craig, et al. 2013. *Introduction to mathematical statistics*. Pearson Education India.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, et al. 2023. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *arXiv preprint arXiv:2311.05232*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Jan Kocoń, Igor Cichecki, Oliwier Kaszyca, Mateusz Kochanek, Dominika Szydło, Joanna Baran, Julita Bielaniewicz, Marcin Gruza, Arkadiusz Janz, Kamil Kanclerz, et al. 2023. Chatgpt: Jack of all trades, master of none. *Information Fusion*, 99:101861.

Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. *arXiv preprint arXiv:1705.04146*.

Wentao Liu, Hanglei Hu, Jie Zhou, Yuyang Ding, Junsong Li, Jiayi Zeng, Mengliang He, Qin Chen, Bo Jiang, Aimin Zhou, et al. 2023. Mathematical language models: A survey. *arXiv preprint arXiv:2312.07622*.

Vann McGee. 2002. *Logic: The Art of Persuasion and Science of Truth*.

Elliott Mendelson. 2009. *Introduction to mathematical logic*. Chapman and Hall/CRC.

Carl D Meyer. 2023. *Matrix analysis and applied linear algebra*. SIAM.

Swaroop Mishra, Matthew Finlayson, Pan Lu, Leonard Tang, Sean Welleck, Chitta Baral, Tanmay Rajpurohit, Oyvind Tafjord, Ashish Sabharwal, Peter Clark, et al. 2022. Lila: A unified benchmark for mathematical reasoning. *arXiv preprint arXiv:2210.17517*.

Ivan Niven, Herbert S Zuckerman, and Hugh L Montgomery. 1991. *An introduction to the theory of numbers*. John Wiley & Sons.

OpenAI. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Martin J Osborne and Ariel Rubinstein. 1994. *A course in game theory*. MIT press.

Arkil Patel, Satwik Bhattamishra, and Navin Goyal. 2021. Are nlp models really able to solve simple math word problems? *arXiv preprint arXiv:2103.07191*.

John G Proakis. 2007. *Digital signal processing: principles, algorithms, and applications, 4/E*. Pearson Education India.

Jingyuan Qi, Zhiyang Xu, Ying Shen, Minqian Liu, Di Jin, Qifan Wang, and Lifu Huang. 2023. The art of socratic questioning: Recursive thinking with large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4177–4199.

Aarohi Srivastava, Abhinav Rastogi, Abhishek Rao, Abu Awal Md Shoeb, Abubakar Abid, Adam Fisch, Adam R Brown, Adam Santoro, Aditya Gupta, Adrià Garriga-Alonso, et al. 2022. Beyond the imitation game: Quantifying and extrapolating the capabilities of language models. *arXiv preprint arXiv:2206.04615*.

James Stewart. 2012. *Calculus: early transcendentals*. Cengage Learning.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Henk Tijms. 2012. *Understanding probability*. Cambridge University Press.

Henk Tijms. 2017. *Probability: a lively introduction*. Cambridge University Press.

Shubham Toshniwal, Ivan Moshkov, Sean Narenthiran, Daria Gitman, Fei Jia, and Igor Gitman. 2024. Openmathinstruct-1: A 1.8 million math instruction tuning dataset. *arXiv preprint arXiv:2402.10176*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Yixuan Weng, Minjun Zhu, Fei Xia, Bin Li, Shizhu He, Shengping Liu, Bin Sun, Kang Liu, and Jun Zhao. 2022. Large language models are better reasoners with self-verification. *arXiv preprint arXiv:2212.09561*.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Tom Griffiths, Yuan Cao, and Karthik Narasimhan. 2024. Tree of thoughts: Deliberate problem solving with large language models. *Advances in Neural Information Processing Systems*, 36.

Ming-Liang Zhang, Fei Yin, and Cheng-Lin Liu. 2023. A multi-modal neural geometric solver with textual clauses parsed from diagram. *arXiv preprint arXiv:2302.11097*.

A Details of Prompts

Example prompts utilized for multiple-choice question answering without and with CoT is provided in prompts A.1 and A.2, respectively. Moreover, the example prompt for answering questions without choices is provided in the prompt A.3.

Example Prompt with Choices

```
Choose the answer to the question only from A, B, C, and D choices, and express your reason.
Question: Find the smallest n that makes the following statement correct: The vertices of any planar graph can be properly colored with n colors.
Choices: A) 4 B) 5 C) 6 D) None of them.
The output should be in the following format:
Explanation: <explanation>
Answer: --
```

Example Prompt with Choices and CoT

```
Choose the answer to the question only from A, B, C, and D choices, and express your reason.
Question: Find the smallest n that makes the following statement correct: The vertices of any planar graph can be properly colored with n colors.
Choices: A) 4 B) 5 C) 6 D) None of them.
The output should be in the following format:
Explanation: <explanation>
Answer: --
Let's think step by step.
```

Example Prompt without Choices

```
Answer to the question, and express your reason.
Question: Find the smallest n that makes the following statement correct: The vertices of any planar graph can be properly colored with n colors.
The output should be in the following format:
Explanation: <explanation>
Answer: --
```

B Accuracy of LLMs over Matt benchmark with Confidence Interval

In Table 5, we calculated a confidence interval (CI) for each topic's accuracy, assuming benchmark scores are Gaussian distributed, similar to the previous works (Dubey et al., 2024).

$$CI = 1.96 \times \sqrt{\frac{p(1-p)}{n}} \quad (1)$$

where p is the observed benchmark score/accuracy and n is number of samples/questions.

C Other Reasoning/Prompting Strategies

Program of Thought (PoT) is a prompting technique designed to enhance numerical reasoning in large language models (LLMs) by integrating code generation into the reasoning process. Unlike Chain-of-Thought (CoT) prompting, where the model performs all reasoning and computation within natural language, PoT allows the model to generate executable code (e.g., Python scripts) as part of its output. This approach delegates complex computations, iterations, and algorithmic tasks to an external interpreter, thereby decoupling intricate calculations from the reasoning steps. By leveraging programming constructs, PoT aims to overcome limitations of LLMs in handling arithmetic operations, solving complex equations, and performing extensive iterative processes that are challenging to express and compute accurately in plain text.

In addition to employing CoT, focusing on GPT-4, we applied PoT prompting to several topics in our MaTT benchmark. However, we observed that for most problems in pure and applied mathematics, breaking down a problem into the PoT format is as challenging as solving the original problem itself. Crafting a correct and efficient program requires a deep understanding of the mathematical concepts and the ability to translate them into algorithmic procedures. This translation often demands the same level of insight and problem-solving skills as directly finding the solution, which poses a significant challenge for LLMs. Consequently, while PoT has the potential to handle computational complexity effectively, its benefits are limited when the model struggles with the initial reasoning required to generate appropriate code.

Moreover, we did not see any significant improvement in the multiple-choice accuracy when using PoT, even in cases where converting the problem into PoT format is not inherently difficult. For example, consider the following question: "A stick is broken into n pieces at n-1 randomly chosen points (n>2). What are the expected values of the lengths of the shortest and longest segments?" Translating the problem into Python code:

Listing 1: Python code to compute the expected lengths of the shortest and longest segments when a stick of length 1 is broken at n-1 random points.

```python
import numpy as np

n = 5  # Example value for n > 2
num_trials = 100000
shortest_lengths = []
longest_lengths = []

for _ in range(num_trials):
    break_points = np.sort(np.random.uniform(0, 1, n - 1))
    segments = np.diff([0] + break_points.tolist() + [1])
    shortest_lengths.append(np.min(segments))
    longest_lengths.append(np.max(segments))

expected_shortest = np.mean(shortest_lengths)
expected_longest = np.mean(longest_lengths)

print("Expected length of the shortest segment:", expected_shortest)
print("Expected length of the longest segment:", expected_longest)
```

We are not able to get the answer for general n with the help of Program of Thought.

	Topics	GPT-4		ChatGPT		Mistral		Llama3.1		o1-mini
		w/o CoT	w CoT	w/o CoT	w CoT	w/o CoT	w CoT	w/o CoT	w CoT	w/o CoT
Pure Math	Algebra	71.1 ±8.1	73.6 ±7.9	45.5 ±8.9	52.1 ±8.9	33.9 ±8.5	39.7 ±8.8	65.8 ±8.5	69.2 ±8.3	89.7 ±5.4
	Calculus	52.2 ±4.3	50.9 ±4.3	41.6 ±4.3	42.6 ±4.3	19.3 ±3.4	19.3 ±3.4	50.6 ±4.3	52.7 ±4.3	88.3 ±2.8
	Number Theory	52.4 ±8.7	50.0 ±8.8	54.0 ±8.7	47.6 ±8.7	22.2 ±7.3	23.8 ±7.4	53.7 ±8.7	53.1 ±8.7	84.9 ±6.3
	Combinatorics	52.1 ±8.3	55.6 ±8.2	45.1 ±8.3	40.8 ±8.2	21.8 ±6.9	19.0 ±6.5	51.4 ±8.3	50.6 ±8.3	73.1 ±7.4
	Geometry	53.8 ±7.8	53.8 ±7.8	51.9 ±7.8	50.0 ±7.8	26.3 ±6.8	27.5 ±6.9	56.0 ±7.7	61.1 ±7.6	71.9 ±7.0
	Logic	62.9 ±16.0	65.7 ±15.7	31.4 ±15.3	34.3 ±15.7	34.3 ±15.7	28.6 ±15.0	55.8 ±16.5	61.7 ±16.1	79.4 ±13.4
Applied Math	Game Theory	40.0 ±16.2	40.0 ±16.2	31.4 ±15.3	45.7 ±16.5	14.3 ±11.6	20.0 ±13.2	54.2 ±16.5	48.5 ±16.6	48.5 ±16.6
	Probability	50.5 ±5.9	46.2 ±5.9	36.5 ±5.7	37.9 ±5.7	20.2 ±4.7	17.6 ±4.5	54.3 ±5.9	54.3 ±5.9	75.0 ±5.1
	OR	40.6 ±9.6	45.3 ±9.7	37.7 ±9.5	30.2 ±9.1	22.6 ±8.3	24.5 ±8.5	41.3 ±9.6	50.0 ±9.8	66.3 ±9.0
	Differential	53.5 ±7.8	52.2 ±7.8	41.5 ±7.7	43.4 ±7.8	18.9 ±6.1	16.3 ±5.8	53.5 ±7.8	57.9 ±7.7	78.9 ±6.2
	Statistics	63.3 ±9.2	59.6 ±9.3	56.9 ±9.4	52.3 ±9.5	28.4 ±8.4	23.9 ±8.1	66.9 ±9.0	72.4 ±8.7	77.9 ±8.4
	Info and Signal	59.3 ±7.2	53.3 ±7.3	38.2 ±6.9	38.2 ±6.9	29.1 ±6.5	26.6 ±6.3	50.8 ±7.3	61.3 ±7.0	75.1 ±6.2
	All	54.0 ±2.4	52.7 ±2.4	42.9 ±2.4	42.7 ±2.4	23.1 ±2.0	22.5 ±2.0	53.5 ±2.4	56.6 ±2.4	79.2 ±2.1

Table 5: Accuracy of LLMs over the MaTT benchmark with confidence intervals.

D Performance Breakdown of Llama3.1 and o1-mini

The performance comparison between o1-mini and Llama3.1 models across topics under both Applied and Pure Mathematics reveals a clear trend in favor of o1-mini. In Pure Mathematics, o1-mini outperforms Llama3.1 in 35 out of 38 subtopics, with Llama3.1 taking the lead in only 3 subtopics (Figure 4). Similarly, for Applied Mathematics, o1-mini shows dominance in 44 out of 47 subtopics, with Llama3.1 outperforming in just 3 cases (Figure 5). These results and Table 2 highlight the consistent superiority of o1-mini across the majority of subtopics in both domains.

E Further Analysis on Explanations

To better understand the influence of choices and to distinguish between instances where the model genuinely engages in reasoning, we provided further analysis in GPT-4 generated explanations. We aim to identify the number of samples in which GPT-4 with choices gave a complete explanation, GPT-4 without choices provided a complete explanation, and both scenarios resulted in complete explanations (over all the questions in MaTT). The findings are presented in Table 6. The result indicates that in most topics, samples that had complete explanations even without the availability of choices also had complete explanations when GPT-4 was provided with choices. Furthermore, in some topics, there is a meaningful difference in the percentage of complete explanations between scenarios with and without choices, emphasizing that the presence of choices can aid models in better engaging with or recalling the reasoning process.

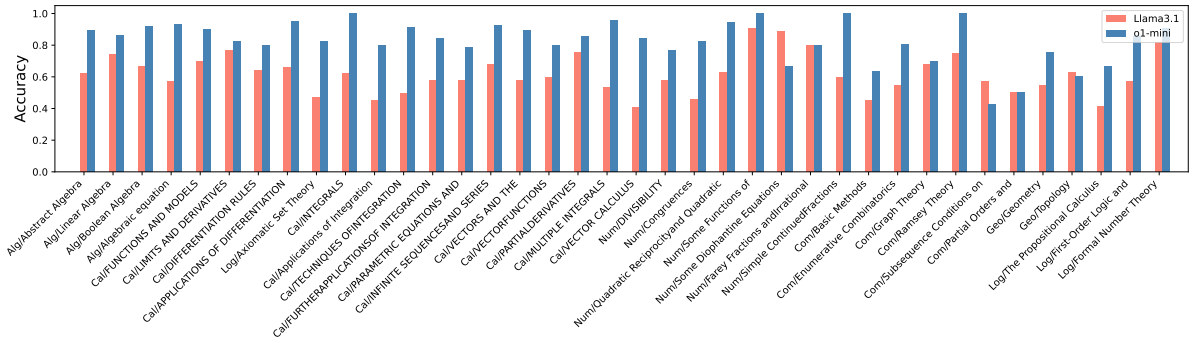

Figure 4: Overview of per topic breakdown for topics under Mathematics/Pure. In this Figure we can observe that in the majority of subtopics (35 out of 38) o1-mini is outperforming Llama3.1, while in the rest of 3 out of 38 subtopics Llama3.1 is outperforming o1-mini.

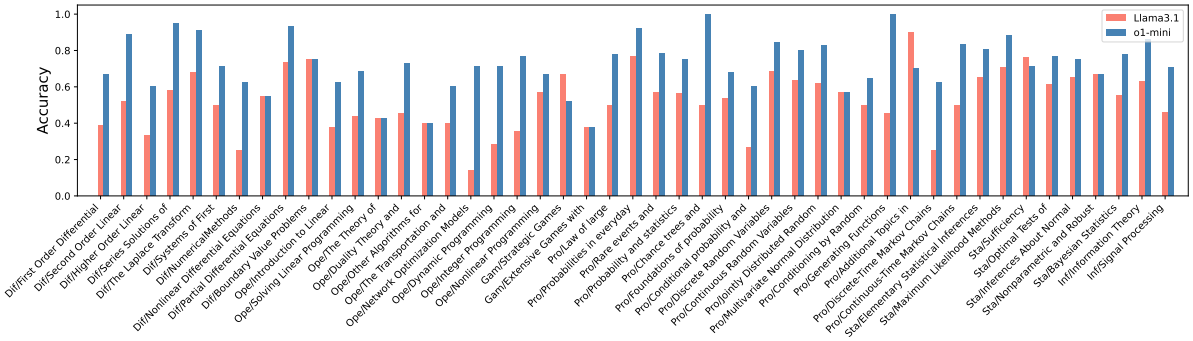

Figure 5: Overview of per topic breakdown for topics under Mathematics/Applied. In this Figure we can observe that in the majority of subtopics (44 out of 47) o1-mini is outperforming Llama3.1, while in the rest of 3 out of 47 subtopics Llama3.1 is outperforming o1-mini.

	Topics	both Complete	No Choice Complete	With Choice Complete
Pure Math	Algebra	28.3	36.7	58.3
	Calculus and Analysis	30.8	44.7	41.4
	Number Theory	4.8	16.7	14.3
	Combinatorics	6.5	20.1	18.0
	Geometry and Topology	2.5	10.1	10.7
	Logic	22.9	42.9	45.8
Applied Math	Game Theory	8.6	11.4	11.4
	Probability	13.4	19.9	20.3
	Operations Research	4.8	10.6	8.7
	Differential Equations	13.4	22.3	21.7
	Statistics	19.3	24.8	27.5
	Info and Signal	24.9	27.1	39.8
	All	18.0	27.4	28.9

Table 6: Comparison on the completeness of explanations from GPT-4 when choices were provided versus when no choices were given (this is over all the samples in MaTT).