

From Calculation to Adjudication: Examining LLM judges on Mathematical Reasoning Tasks

Anonymous ACL submission

Abstract

To reduce the need for human annotations, large language models (LLMs) have been proposed as judges of the quality of other candidate models. LLM judges are typically evaluated by measuring the correlation with human judgments on generation tasks such as summarization or machine translation. In contrast, we study LLM judges on mathematical reasoning tasks. These tasks require multi-step reasoning, and the correctness of their solutions is verifiable, enabling a more objective evaluation. We perform a detailed performance analysis and find that the used judges are mostly unable to improve task performance but are able to pick the better model. Our analysis uncovers a strong correlation between judgment performance and the candidate model task performance. We observe that judges tend to choose the model of higher quality even if its answer is incorrect. Further, we show that it is possible to use statistics, such as the task performances of the individual models, to predict judgment performance. In an ablation, we systematically perturb the candidate answers and observe that judges often keep the original answer, providing evidence that judges incorporate writing style in their judgments. In summary, we find that regularities in the judgments are quantifiable using statistical measures, and provide various angles on exploiting them.¹

1 Introduction

The automatic evaluation of machine learning models holds great promise in reducing the amount of human annotations. Specifically, the LLM-as-a-judge paradigm (Zheng et al., 2023) has gained traction, aiming to assess or compare the quality of generated texts automatically. This approach is beneficial for automated data labeling (Tan et al., 2024), self-improvement of LLMs (Wu et al., 2024), and ranking LLMs with respect to specific tasks (Zheng et al., 2023).

¹Code will be made available upon acceptance.

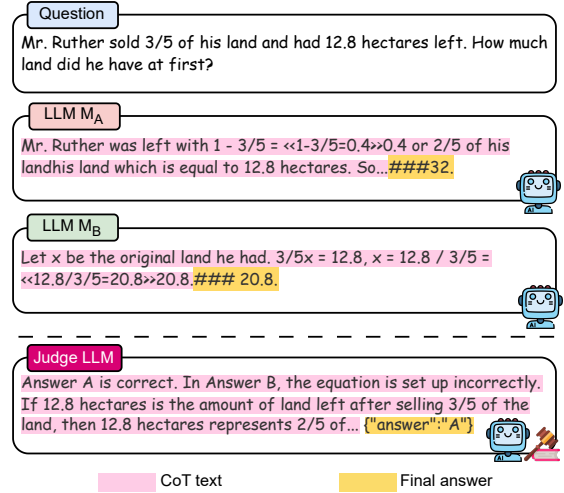


Figure 1: In our problem setup two LLMs (M_A and M_B), provide candidate answers for a math problem, and a judge LLM *has* to decide which one is correct. All three use chain-of-thought (CoT) reasoning (Wei et al., 2022).

Much like judges in the real world, who are expected to be exact, fair, and unbiased, e.g., as defined in Bangalore Principles of judicial conduct (Bangalore Principles, 2002), LLMs, when employed as judges, should be ethical and logical. Already the philosopher Aristotle argued that the virtuous actor exhibits the joint excellence of reason and character (Kraut, 2022). Previous works investigate properties and biases of LLM judges on generation tasks such as translation or summarization (Kim et al., 2024b; Liu et al., 2024). These are typically evaluated using correlation with human annotators and are thus inherently subjective.

In this work, we investigate LLM judges on mathematical reasoning datasets (see Figure 1). These need complex multi-step reasoning, and the solution is verifiable, which allows us to investigate the relationship between judge and candidate models in a principled manner. We base our analysis on four large (more than 30B parameters) LLMs and four small (less than 10B) LLMs on three mathe-

063
064
065
066
067
068
069
070
071
072
073
074
075
076
077
078
079
080
081
082
083
084
085
086
087
088
089
090
091
092
093
094
095
096
097
098
099
100
101
102
103
104
105
106
107
108
109
110
111

mathematical reasoning datasets.

Our experiments contain a detailed performance examination, confirming that larger models are generally better judges (Zheng et al., 2023). We find that only the best-tested model, Qwen 2 72B, consistently improves task performance if we evaluate the judged samples, but all tested judges likely pick the better model for a given task.

We investigate subsets with one correct and one incorrect candidate answer. We uncover a correlation between judgment performance and task performance of the candidate models, showing that judges tend to select incorrect answers from better models. Thus, we hypothesize that judges have access and rely on the superior *writing styles* of larger models instead of solely analyzing the reasoning. When we divide the datasets into buckets of model agreement, we observe that agreement is a proxy for sample difficulty.

Motivated by these regularities, we analyze whether it is possible to predict judgment performance and find that task performances of judge and candidate LLMs explain most of the variance. We hypothesize that judges incorporate writing style into their judgments. Thus, we predict individual judgments using statistical and transformer-based models and achieve above-chance performance, supporting our hypothesis.

Lastly, we test how judgments are affected by perturbing numeric values in responses by 1) swapping results and 2) redacting numeric values. Our findings reveal that judges largely retain original judgments, providing further evidence that judges, in large part, base their decisions on writing style. In summary, our contributions are as follows:

1. We conduct an in-depth performance analysis of LLM judges for mathematical reasoning tasks.
2. Our analysis reveals a correlation between the judgment and candidate task performance, providing a novel statistical angle on the analysis of LLM judges.
3. We show that statistics such as task performance or agreement of candidate models are indicative of judgment performance.
4. After systematically perturbing the candidate answers, we observe that judges often keep their original judgments, providing evidence that judgments are also based on writing style.

2 Related Work

2.1 LLM as Judges

Using *LLMs as judges* to evaluate text generated by LLMs, including their own outputs, has recently attracted significant interest because it reduces the need for human annotation (Zheng et al., 2023). Commonly, large *frontier models* are used as judges. Applications include the automatic assessment of language model capabilities and, e.g., determining which model performs better on a given task (Zheng et al., 2023) and reinforcement learning from AI feedback by automatically generating data for preference optimization (Bai et al., 2022; Wu et al., 2024).

Various methods exist to make judgments (Zheng et al., 2023; Liusie et al., 2024). One approach is pairwise selection (Wang et al., 2024a), where two answers are presented, and the model is asked to select the better one. Another approach is pointwise grading (Li et al., 2024), where the model is asked to assign a grade based on a pre-defined scale, and the answer with a better grade is chosen. Judgment prompts may involve reference solutions or not. Another body of research explicitly trains models to act as judges (Kim et al., 2024a; Wang et al., 2024a) or closely related, as reward models (Wang et al., 2024b; Li et al., 2024).

The effectiveness of LLMs as judges is typically assessed by measuring the correlation or overlap with human judgments (Zheng et al., 2023; Kim et al., 2024b). In contrast, we focus on difficult tasks with a concrete final answer. Finally, we want to stress that several works caution for the use of LLM judges as experts (Bavaresco et al., 2024; Koo et al., 2023; Raina et al., 2024). In a similar vein, we aim to understand regularities and their shortcomings.

2.2 Biases in LLM-as-a-judge

Human-annotated data inherently reflects the annotators’ biases and opinions. These biases can be detrimental or (intentionally) beneficial, depending on the goals of the annotation process (Plank, 2022). Similarly, several studies have explored the biases present in LLM judges:

One linguistic bias is ordering bias (Zheng et al., 2023; Koo et al., 2023), where a judge gives a different answer depending on the order in which answers are presented. Panickssery et al. (2024) note that it is possible to interpret position bias as a sign that the model is unsure. There are multiple

112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161

works (Xu et al., 2024; Panickssery et al., 2024; Liu et al., 2024) that find evidence for self-bias or self-preference. Koo et al. (2023) provide a benchmark for analyzing cognitive biases. West et al. (2024) and Oh et al. (2024) explore the ‘‘Generative AI Paradox’’ where generating solutions is easier for the LLM than analyzing them, unlike humans who typically find analysis easier than generation.

In this work, we aim to establish a better understanding of underlying regularities that relate judgments to statistics such as model performance.

3 General Setup

In the following, we describe the problem setting, including the used notation, and the general experimental setting including used models and datasets.

3.1 Problem Description

In this work, we consider two models, denoted by $M_A, M_B \in \mathbf{M}$, providing candidate solutions for a sample of a dataset D and a judge model $M_J \in \mathbf{M}$, which is tasked to select, to ‘‘judge’’, whether it prefers the solutions of the models M_A or M_B . The solutions are represented by the random variables A and B . We consider the events that solutions are true ($A = T$), false ($A = F$), or that their solution is the same ($A = B$). We denote the judgment of the judge M_J by the random variable Δ_J , which can either be correct ($\Delta_J = T$), incorrect ($\Delta_J = F$) or choose a specific model M_A ($\Delta_J = M_A$).

Given that the final answer is either correct or incorrect, we can break the probability of the judge making a correct judgment $P(\Delta_J = T|A, B, D)$ given a sample of a dataset D and the answers of two models M_A, M_B down into the following four cases:

$$\begin{aligned} P(\Delta_J = T|A, B, D) & \\ &= \sum_{(X, Y) \in C} P(\Delta_J = T|A = X, B = Y, D)P(A = X, B = Y|D) \\ &= P(S_A = X, S_B = Y|D) \\ &+ P(\Delta_J = T|A = X, B = Y, D)P(A = X, S_B = Y|D) \\ &+ P(\Delta_J = T|A = X, B = Y, D)P(A = X, B = Y|D) \end{aligned} \quad (1)$$

where $C = (T, F)^2$. Note that in cases where both answers are correct or incorrect imply that the judgment is also either correct or incorrect respectively, i.e., $P(\Delta_J = T|A = T, B = T) = 1$ and $P(\Delta_J = T|A = F, B = F) = 0$.

3.2 Datasets

The experiments encompass three mathematical reasoning datasets where models highly benefit

from multi-step CoT reasoning. For all datasets, we use accuracy as the performance metric.

AQUA-RAT (Ling et al., 2017) is a dataset to test the quantitative reasoning ability of LLMs. Unlike the other two datasets, the questions are multiple-choice. **GSM8K** (Cobbe et al., 2021) consists of grade school math word problems. The answers are free-form numbers. **MATH** (Hendrycks et al., 2021) contains challenging competition mathematics problems. Find more details in Appendix A.1

3.3 Models

We evaluate the performance of openly available LLMs, including four large models *Qwen 2 72B* (Yang et al., 2024), *Llama 3 70B* (AI@Meta, 2024), *Yi 1.5 34B* (Young et al., 2024), *Mixtral 8x7B* (Jiang et al., 2024) and four small models, namely *Llama 3 8B* (AI@Meta, 2024), *Gemma 1.1 7B* (Gemma Team et al., 2024), *Mistral 7B v0.3* (Jiang et al., 2023), and *Mistral 7B v0.1* (Jiang et al., 2023). We use the chat- or instruction-tuned model variants and test each model as candidate answer generator and as judge. More information is in Appendix A.2.

3.4 Inferences

This section describes the candidate answer generations, and the judgments comparing two candidate answers. Find more information on prompts and hardware details in Appendix A.

Candidate answer generation. To judge two candidate answers (including of the same model), we sample two initial CoT solutions for each model using 4-shot prompting. We set the temperature to 0.9 to get two different solutions.

Judgements. We choose the first candidate generation for each model and generate judgments for all 36 unique model combinations.² If both models are the same, we take the second initial generation. We accommodate positional bias (Zheng et al., 2023; Koo et al., 2023) by evaluating the two candidate answers in both possible orders for each question and then taking the average correctness of the judgments as the final assessment. The judge *has* to choose if the first or second answer is correct. The prompt is zero-shot and applies CoT, the temperature is set to 0 for deterministic generation results.

²We consider all pairs from the eight LLMs, including self-pairing, yielding $\binom{8+2-1}{2} = 36$ combinations.

		Qwen 2 72B	Llama 3 70B	Yi 1.5 34B	Mixtral 8x7B	Llama 3 8B	Gemma 1.1 7B	Mistral 7B v0.3	Mistral 7B v0.1
(1) $P(\Delta_J = T A, B, D)$	AQUA_RAT	66.05	55.98	<u>62.2</u>	57.08	51.59	53.52	54.96	51.41
	GSM8K	77.06	72.41	<u>72.64</u>	68.81	65.88	65.47	68.57	63.28
	MATH	29.66	24.64	<u>26.69</u>	23.60	22.35	21.66	22.68	19.91
(2) $P(\Delta_J = T A \neq B, D)$	AQUA_RAT	53.79	45.04	<u>49.73</u>	44.47	36.56	40.48	38.26	36.43
	GSM8K	63.16	<u>59.33</u>	56.01	46.68	41.29	38.71	42.59	39.72
	MATH	25.04	22.14	<u>22.22</u>	18.29	17.04	16.00	16.88	15.22
(3) $P(\Delta_J = T \{A, B\} = \{T, F\}, D)$	AQUA_RAT	73.13	64.63	<u>68.66</u>	63.26	52.74	58.04	54.54	52.45
	GSM8K	85.65	<u>81.32</u>	76.84	64.97	57.87	54.16	59.58	55.48
	MATH	80.48	<u>73.01</u>	71.37	61.03	58.04	54.70	56.27	50.91

Table 1: Performance of judge LLMs in three cases: (1) accuracy on all samples, (2) accuracy where models M_A and M_B disagree, and (3) accuracy where only one model is correct. Results are averaged over all pairs (M_A, M_B) , with the highest accuracy in bold and the second highest underlined.

4 General Performance

The experiments have multiple degrees of freedom: judges, candidate models, and datasets. Therefore, we first examine judgments per dataset, and secondly, we investigate judgments per candidate model pair. Afterwards, we provide evaluations for two applied questions.

4.1 Performance per dataset

We begin by examining the judgment performance, i.e., how often the judge picks a correct answer, across different datasets. Therefore, we average the performance across all model pairs (M_A, M_B) .

Setup. Table 1 considers three cases where each case focuses on a specific subset of the datasets: *Case (1)* investigates the observed task performance $P(\Delta_J = T|A, B, D)$ where we evaluate the task performance using the answers chosen by the judges. Note that this includes samples where both candidate models give the same answer. *Case (2)* asks how often judges choose a correct answer given that the answers differ, i.e., $P(\Delta_J = T|A \neq B, D)$. Note that this may (and often does) include cases where both answers are incorrect. *Case (3)* gives the probability that the judge chooses the correct answer given that one answer is correct, and the other answer is incorrect, formally $P(\Delta_J = T|A \neq B, T \in \{A, B\}, D)$.

Results. We observe that large models outperform smaller models. Specifically, we see that Qwen 2 72B is the best judge, followed by Yi 1.5 34B. The performance of Llama 3 70B is, on average, comparable to that of Yi 1.5 34B. Note that performance in *Case (1)* and *Case (2)* is often quite low, especially for MATH, as there are many cases where the judge can only choose wrong answers. Importantly, we observe that smaller models with fewer than 10B parameters are unreliable judges. Especially, in *Case (3)*, where a correct answer is

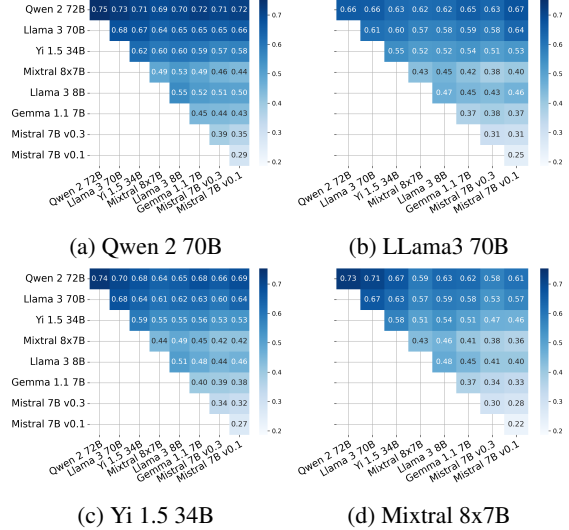


Figure 2: Observed performance $P(\Delta_J = T|A, B, D)$ of four judge LLMs (a-d) in evaluating various model pairs, averaged across all datasets.

provided, smaller models only achieve an accuracy of around 55%, barely better than random chance. Therefore, we focus on the four larger models as judges in the subsequent analysis.

4.2 Performance per model combination

The comparative performance of model pairs offers insights into which model is better for the specific task or which combination of models yields the best results.

Setup. Figure 2 illustrates the final performance $P(\Delta_J = T|A, B)$, indicating the probability of a judge choosing a correct answer given two models A and B . The results are averaged over datasets and presented as an upper triangular matrix due to symmetry. If both models in a pair are the same, $A = B$, we employ the second response generated with temperature sampling to introduce variation. We report the performance of all models used as judges in the Appendix B in Table D.

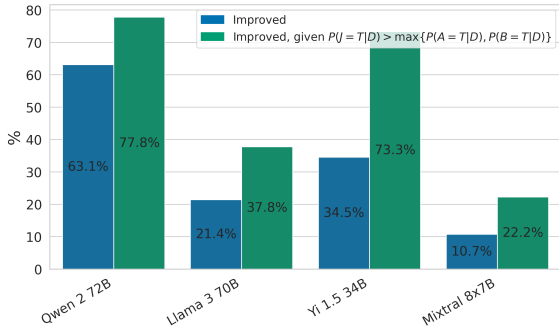


Figure 3: Amount of model pairs (M_A, M_B) where the answers chosen by the judge achieve a higher task performance than the models individually (green). The blue bar only considers models where the judge is at least as good as the candidate models.

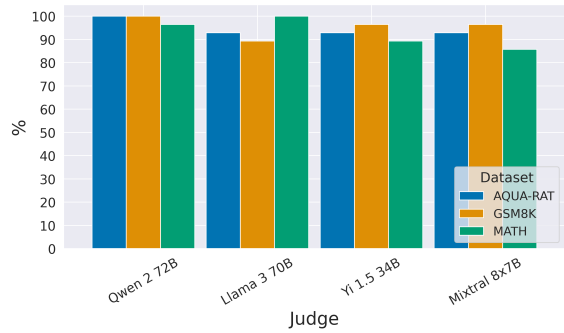


Figure 4: Percentage of model pairs (M_A, M_B) where a judge picks a better model M_A (meaning $P(A = T|D) > P(B = T|D)$), by selecting more answers of M_A than from M_B .

Results. We observe that the best performance is achieved when both the candidate answers and the judge are the highest-performing model, Qwen 2 72B. An analysis of the first rows (cf. Figure 2) reveals a notable trend: The final performance declines when comparing the output of a strong model against a mediocre model (e.g., Llama3-8B) but then improves again when compared against the weakest model. This suggests that judging becomes more challenging when distinguishing between the correct answers of a strong model and the incorrect answers of a mediocre one, compared to discerning the outputs of a bad one.

4.3 Do judges elicit task improvement?

One use case for LLM judges is to improve task performance. A potential application is to train on answers chosen by the judge (Yuan et al., 2024).

Setup. Therefore, we test how often the performance of the answers chosen by the judge is better than the performance of the individual models. Formally, for all pairs of models M_A, M_B and datasets D , how often is the observed performance $P(\Delta_J = T|A, B, D)$ larger than $\max\{P(A = T|D), P(B = T|D)\}$? In Figure 3 the green bar tests all model pairs, and the blue bar only pairs where the judge is at least as good as the candidate models, i.e., $P(J|D) \geq \max\{P(A|D), P(B|D)\}$. The task performances of all models are given in the Appendix B in Table 9.

Results. We see that only Qwen 2 72B increases the performance reliably. However, it is easier for the judge to improve performance if it compares answers of less or equally good candidate models.

4.4 Does the judge prefer the better model?

Another application of LLM judges is whether they can accurately identify which model performs better for a given task. This is crucial if we want to rank LLMs by their capabilities or if a practitioner wants to decide which model to deploy.

Setup. To assess this, we evaluate the frequency with which a judge selects the superior model. For a candidate model pair $M_A, M_B \in \mathbf{M}$, always assume they are ordered, such that $P(A = T|D) > P(B = T|D)$. Then, specifically, we determine the proportion for which the judge chooses M_A more often than M_B , or formally, how often is $P(\Delta_J = M_A|A, B, D) > P(\Delta_J = M_B|A, B, D)$ for all candidate pairs and datasets.

Results. The judges consistently perform well in the selection of the better model. Notably, we find that Qwen 2 72B can only not rank the pair Mistral 7B v0.1 and v0.3 on the MATH dataset. This issue appears minor, as both models exhibit similarly poor performance on the challenging MATH dataset (with accuracies of 6.13% and 3.10%, respectively), meaning most judgments compare two wrong answers. Notably, already the worst judge, Mixtral 8x7B, performs well. In summary, we see that judges are more capable of aggregate-level rankings than instance-level rankings.

5 Analysis of Subsets

We investigate properties that occur when we use subsets based on the correctness of models or agreement between models.

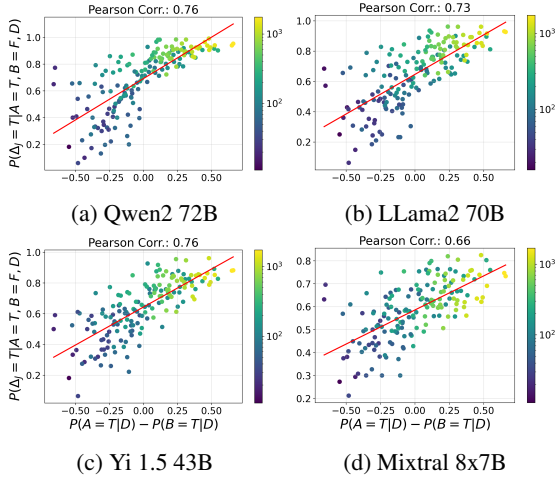


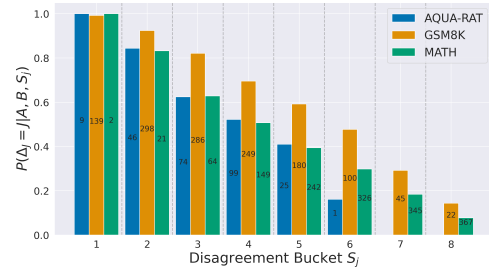
Figure 5: Judges’ accuracy vs. performance gap between two candidate models M_A and M_B . Each point represents a subset where M_A is correct and M_B is incorrect. The color reflects the size of these subsets.

5.1 Do task performances correlate with judgments?

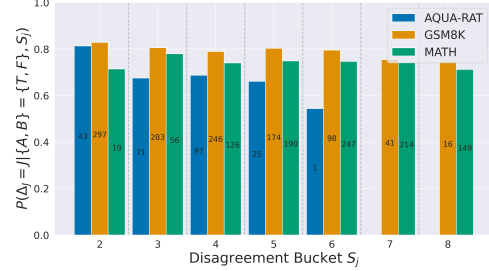
We consider the subset of highest practical relevance where one candidate model is correct, and one candidate model is incorrect. The goal is to investigate the relationship between candidate model task performance and judgment performance.

Setup. For all model pairs $M_A, M_B \in \mathbf{M}, M_A \neq M_B$ we analyze subsets where M_A is correct, and M_B is incorrect. Note that we can always order M_A and M_B this way. Each plot in Figure 5 shows the relationship between judge performance, $P(\Delta_J = T|A = T, B = F)$ (Y-axis) and candidate model performance gap of M_A and M_B , i.e., $P(A = T|D) - P(B = T|D)$ (X-axis). Examples of these subsets and their corresponding performances are in Appendix C in Table 10.

Results. The analysis reveals a strong correlation (Pearson’s $r^2 > 0.69$) between candidate model performance gap and judgment accuracy. If the performance gap is negative, we consider subsets where larger models are incorrect. Judges favor answers from larger models even when they are incorrect on these subsets. We hypothesize that this bias arises because larger models exhibit a specific writing style, articulating their responses more convincingly, thereby misleading the judges. This finding aligns with previous research identifying self-bias (Xu et al., 2024; Panickssery et al., 2024; Liu et al., 2024). However, our results indicate that this bias extends more broadly to the inherent qual-



(a) Performance on all comparisons.



(b) Performance on comparisons with one correct and one incorrect answer.

Figure 6: Judge performance by agreement bucket. E.g., bucket S_3 (X-axis) means that all eight models gave together three different answers. Note that AQUA-RAT is multiple-choice with maximally six answers.

ity of the underlying models on reasoning datasets. However, this is not necessarily a critical issue in practice, as the larger model tends to answer correctly more often (as indicated by the color of the points in Figure 5).

5.2 Does judgment quality depend on models’ agreement?

We are interested in whether the level of agreement among models, i.e., how many models give a different answer for a sample, impacts the performance on the respective subset.

Setup. We define disagreement buckets S_j , where each bucket contains instances for which exactly $1 \leq j \leq 8$ unique answers were given across all models. Formally, we set

$$S_j = \bigcup_D \{i \in D \mid |\{M_A(i) \mid M_A \in \mathbf{M}\}| = j\}$$

where $M_A(i)$ is the answer of model M_A for instance i . We analyze the results in two contexts: all comparisons, including those where both answers are correct or incorrect (cf. Figure 6(a)), and only instances where exactly one answer is correct (cf. Figure 6(b)). We average the performances of all

		$P(\Delta_J = T \text{Condition})$		
↓ Features \ Condition →	A, B	$A \neq B$	$\{T, F\} = \{A, B\}$	
(1) $P(J), P(A), P(B)$	97.50	90.20	59.20	
(2) $P(A = B),$ $P(J = A A \neq B)$	76.00	54.90	49.90	

Table 2: Coefficients of Determination (R^2 , higher is better) for linear regression using the different feature sets as covariates (rows) and different target variables defined by the condition (columns). All values are significant ($p < 0.001$) as per an Overall-F-Test.

judges and all candidate pairs. Find per-judge plots in Appendix C in Figure 10.

Results. Figure 6(a) shows that when all models agree (bucket S_1), the performance is nearly 100%, indicating unanimous agreement usually means correctness. As disagreement increases, performance expectedly decreases. Thus, model agreement is a proxy for sample difficulty. In 6(b), where a correct and an incorrect answer exists, performance remains relatively stable across disagreement buckets for datasets with free-form answers, such as GSM8K and MATH. However, for AQUA-RAT, performance degrades as disagreement rises.

6 Prediction of Judgements

We investigate whether predicting the judgments’ outcomes is feasible. Firstly, we aim to predict performance statistics. Secondly, we aim to predict individual judgments.

6.1 Can we predict judgment performance?

On the subset where exactly one answer is correct, we found a strong correlation between judgment performance and candidate task performances. This hints at regularities within the judging process, thus we aim to predict judge performance using model statistics.

Setup. We fit six different linear regression models using the judgment performances as the target variables Y , including all variations of judges, model pairs $M_A, M_B \in \mathbf{M}$, and datasets D . Regarding the covariates \mathbf{X} in the model, we distinguish between two setups: In *Case (1)*, we solely use the task performances $P(X|D)$, $X \in \{J, A, B\}$ of judge and candidate models, to predict judgment performance. In *Case (2)*, we utilize statistics available without knowledge of the ground truth. The features for this case are the probability of agreement between the candidate models

↓ Model Judge →	Qwen 2 72B	Llama 3 70B	Yi 1.5 34B	Mixtral 8x7B
(1) TF-IDF + RF	60.78	61.37	60.77	58.69
(2) RoBERTa	68.14	66.49	67.03	63.91

Table 3: Accuracy of predicting LLM judges’ decisions using Random Forest (RF) and RoBERTa classifiers.

$P(A = B|D)$ and the probability of model M_A being chosen. Since we are not specifically interested in the individual features’ effects, but rather in their ability to explain the variation of judgment performance, we rely on the coefficient of determination, R^2 , for evaluation (Fahrmeir et al., 2013, see Appendix E).

Results. The results are shown in Table 2 (excluding data sets from the probability formulas for simplicity). We observe that the performance-related features of the models can almost perfectly explain the variation in final judgment performance ($R^2 = 97.50\%$), also when conditioning only on the subset of differing answers ($R^2 = 90.20\%$). Logically, $P(A)$ and $P(B)$, i.e., $P(A|D)$, $P(B|D)$ respectively, have significant³ explanatory power for judgment performance, as they encompass all correct answers. In *Case (2)*, we still observe a relatively high R^2 value, indicating that the features can explain 50% of the target’s variance.

6.2 Can we predict which individual judgments?

We hypothesize that judgments are biased towards larger or better models because they incorporate linguistic cues or writing style into their judgments rather than purely relying on reasoning assessment. Therefore, we train a classifier to understand whether we can predict individual judgments.

Setup. We separate all comparisons made per judge into training, validation, and test splits and train two classifiers. The test accuracy is reported in Table 3. The first model utilizes TF-IDF vectorization. We create two independent vectorizers for both answers. The resulting features are concatenated. A RandomForest classifier (Breiman, 2001) is then trained on these combined features. The second model is a RoBERTa model (Liu et al., 2020) trained on the full prompt presented to the judge. Refer to Appendix D for the training details of both models.

³We test statistical significance using an Overall-F-Test for each fitted model. Further details are in Appendix E.

	Exchanged			Redacted		
	=	≠	Refused	=	≠	Refused
Qwen2-72B	75.75	12.20	12.05	56.40	26.06	17.53
Llama-3-70B	78.81	13.72	7.47	63.44	29.22	7.33
Yi-1.5-34B	74.80	14.47	10.73	44.89	37.31	17.80
Mixtral-8x7B	71.19	20.69	8.12	60.34	29.26	10.40

Table 4: Analysis of judgments where results in candidate answers were either exchanged or numbers redacted. We report how many judgments stay the same (=), different (\neq), or where judges refused to follow the output format (Refused).

Results. The random forest model achieves an accuracy of approximately 60%, demonstrating performance above random chance. This suggests that specific keywords or phrases influence judges. The RoBERTa model surpassed this, reaching nearly 70% accuracy. Taken together, these results suggest that judge decision-making is a multi-faceted process. While specific linguistic cues appear to hold influence, a substantial portion of the decision-making process seems to be based on other contextual factors or broader reasoning.

7 Perturbation of Results

We aim to gain a deeper understanding of the extent to which writing style affects the final judgment. Therefore, we create an experiment perturbing the candidate answers and examine whether this changes the judgment.

Setup. We examine two perturbations: *Exchange* and *Redaction*. In the *Exchange* experiment, we swap the final answer from model M_A with that of model M_B , while keeping their CoT reasoning unchanged. In the *Redaction* experiment, we anonymize all numbers in both the CoT reasoning and the final answer by replacing them with “X”.⁴ Table 4 shows the frequency with which the judge selects the same answer (=), a different answer (\neq), or fails/refuses to follow the output format and make a decision (Refused). Refer to Appendix F for specific examples.

Results. We observe that the new judgments in more than half the cases agree with the original judgment. In the *Exchange* experiment, they even agree on average by 75% of the cases. We deduce that the judge is largely unaffected by the artificially

⁴In preliminary runs, we observed that redaction caused significant confusion for the judge models. To address this, we adapt the judgment prompt in this setting to include the instruction: “Only analyze the reasoning! All numbers have been replaced with ‘X’ to help you focus on the reasoning.”

introduced noise and heavily bases its decision on the writing style. Interestingly, in a substantial amount of samples (up to 17%) the judge refuses to make a judgment. On a positive note, manual inspection revealed that the model often realizes that the original answers were perturbed.

8 Discussion

Style and Quality. Our experiments suggest a relation between judgment and candidate task performance (cf. Section 5) and a relation between judgment and writing style (cf. Sec. 6 and 7). We hypothesize these two are interconnected and facets of the same underlying bias. When models become better, e.g., by being trained on larger amounts of data, their ability to write convincingly increases. Conversely, when an LLM demonstrates an increased ability to write convincingly, it likely acquires a more nuanced grasp of what humans perceive as compelling. This enhanced understanding likely also extends to task performance.

Generalizability of approach. Our in-depth analysis utilizes Formula (1) to segment judgment data based on correctness criteria, allowing for targeted investigation of specific subsets. This approach is generalizable and transferable to other NLP tasks, such as summarization. By incorporating discrete signals such as text topics, a similar derivation of the judgment probability is possible.

9 Conclusion

We conducted a thorough analysis of LLM judges on mathematical reasoning tasks. We include a detailed judgment performance evaluation of eight models on three datasets. We find that larger models are generally better than smaller models and that judges succeed in detecting the more capable model. Our analysis reveals a strong correlation between judgment performance and task performance of the models providing candidate answers which shows that judges tend to choose larger or better models. We hypothesize that LLM judges incorporate writing style into their judgments instead of purely analyzing the reasoning. We provide two experiments to provide evidence for this hypothesis.

Finally, we want to emphasize the importance of impartiality and fairness in the role of LLM judges, similar to human judges in the real world. Our research introduces methods to quantify biases in favor of larger or better models, thereby offering a means to measure the reduction of such biases.

589 Limitations

590 Our analysis is primarily focused on mathemati-
591 cal reasoning datasets, which allows us to explore
592 judgments through the lens of correctness within
593 specific subsets. While this approach provides valu-
594 able insights, it limits the generalizability of our
595 findings to other tasks or domains. Based on the
596 fact that the investigated datasets are complex, in
597 the sense that they need multi-step reasoning to
598 be solved, and based on the fact that there is no
599 thorough investigation of LLM judges on mathe-
600 matical reasoning datasets yet, we think this work
601 is a valuable contribution.

602 In our experiments, we focus on testing a single,
603 specific prompt. It is common knowledge that
604 LLMs are highly sensitive to variations in prompt
605 phrasing, which can substantially influence their
606 performance. Nevertheless, it is impossible for us
607 to meet the computational demands necessary to
608 run our experiments with multiple prompts.

609 In this study, we intentionally concentrate on
610 open-weight models, motivated by our strong belief
611 in the principles of open science. Open-
612 weight models offer transparency and reproducibil-
613 ity, which are critical for advancing scientific under-
614 standing. However, we note that it is also interest-
615 ing to study closed models to understand potential
616 differences. Still, we are committed to research on
617 open-weight models because we believe it benefits
618 the community more.

619 References

620 AI@Meta. 2024. [Llama 3 model card](#).

621 Yuntao Bai, Saurav Kadavath, Sandipan Kundu,
622 Amanda Askell, Jackson Kernion, Andy Jones,
623 Anna Chen, Anna Goldie, Azalia Mirhoseini,
624 Cameron McKinnon, et al. 2022. Constitutional
625 ai: Harmlessness from ai feedback. *arXiv preprint*
626 *arXiv:2212.08073*.

627 Bangalore Principles, 2002. 2002. [The bangalore princi-
628 ples of judicial conduct](#). Available from the Judicial
629 Integrity Group website.

630 Anna Bavaresco, Raffaella Bernardi, Leonardo Berto-
631 lazzi, Desmond Elliott, Raquel Fernández, Albert
632 Gatt, Esam Ghaleb, Mario Giulianelli, Michael
633 Hanna, Alexander Koller, André F. T. Martins,
634 Philipp Mondorf, Vera Neplenbroek, Sandro Pezzelle,
635 Barbara Plank, David Schlangen, Alessandro Suga-
636 lia, Aditya K Surikuchi, Ece Takmaz, and Alberto
637 Testoni. 2024. [Llms instead of human judges? a
638 large scale empirical study across 20 nlp evaluation
639 tasks](#).

Leo Breiman. 2001. [Random forests](#). *Mach. Learn.*,
45(1):5–32. 640 641

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian,
642 Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias
643 Plappert, Jerry Tworek, Jacob Hilton, Reiichiro
644 Nakano, Christopher Hesse, and John Schulman.
645 2021. Training verifiers to solve math word prob-
646 lems. *arXiv preprint arXiv:2110.14168*. 647

Ludwig Fahrmeir, Thomas Kneib, Stefan Lang, Brian
648 Marx, Ludwig Fahrmeir, Thomas Kneib, Stefan Lang,
649 and Brian Marx. 2013. *Regression models*. Springer. 650

Google Gemma Team, Thomas Mesnard, Cassidy
651 Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya
652 Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay
653 Kale, Juliette Love, Pouya Tafti, Léonard Hussenot,
654 Pier Giuseppe Sessa, Aakanksha Chowdhery, Adam
655 Roberts, Aditya Barua, Alex Botev, Alex Castro-
656 Ros, Ambrose Stone, Amélie Héliou, Andrea Tac-
657 chetti, Anna Bulanova, Antonia Paterson, Beth
658 Tsai, Bobak Shahriari, Charline Le Lan, Christo-
659 pher A. Choquette-Choo, Clément Crepy, Daniel Cer,
660 Daphne Ippolito, David Reid, Elena Buchatskaya,
661 Eric Ni, Eric Noland, Geng Yan, George Tucker,
662 George-Christian Muraru, Grigory Rozhdestvenskiy,
663 Henryk Michalewski, Ian Tenney, Ivan Grishchenko,
664 Jacob Austin, James Keeling, Jane Labanowski,
665 Jean-Baptiste Lespiau, Jeff Stanway, Jenny Bren-
666 nan, Jeremy Chen, Johan Ferret, Justin Chiu, Justin
667 Mao-Jones, Katherine Lee, Kathy Yu, Katie Milli-
668 can, Lars Lowe Sjoesund, Lisa Lee, Lucas Dixon,
669 Machel Reid, Maciej Mikuła, Mateo Wirth, Michael
670 Sharman, Nikolai Chinaev, Nithum Thain, Olivier
671 Bachem, Oscar Chang, Oscar Wahltinez, Paige Bai-
672 ley, Paul Michel, Petko Yotov, Rahma Chaabouni,
673 Ramona Comanescu, Reena Jana, Rohan Anil, Ross
674 McIlroy, Ruibo Liu, Ryan Mullins, Samuel L Smith,
675 Sebastian Borgeaud, Sertan Girgin, Sholto Douglas,
676 Shree Pandya, Siamak Shakeri, Soham De, Ted Kli-
677 menko, Tom Hennigan, Vlad Feinberg, Wojciech
678 Stokowiec, Yu hui Chen, Zafarali Ahmed, Zhitao
679 Gong, Tris Warkentin, Ludovic Peran, Minh Giang,
680 Clément Farabet, Oriol Vinyals, Jeff Dean, Koray
681 Kavukcuoglu, Demis Hassabis, Zoubin Ghahramani,
682 Douglas Eck, Joelle Barral, Fernando Pereira, Eli
683 Collins, Armand Joulin, Noah Fiedel, Evan Senter,
684 Alek Andreev, and Kathleen Kenealy. 2024. [Gemma:
685 Open models based on gemini research and technol-
686 ogy](#). 687

Dan Hendrycks, Collin Burns, Saurav Kadavath, Akul
688 Arora, Steven Basart, Eric Tang, Dawn Song, and
689 Jacob Steinhardt. 2021. Measuring mathematical
690 problem solving with the math dataset. *NeurIPS*. 691

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Men-
692 sch, Chris Bamford, Devendra Singh Chaplot, Diego
693 de las Casas, Florian Bressand, Gianna Lengyel, Guil-
694 laume Lample, Lucile Saulnier, Léo Renard Lavaud,
695 Marie-Anne Lachaux, Pierre Stock, Teven Le Scao,
696 Thibaut Lavril, Thomas Wang, Timothée Lacroix,
697 and William El Sayed. 2023. [Mistral 7b](#). 698

699	Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, L�elio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Th�ophile Gervet, Thibaut Lavril, Thomas Wang, Timoth�e Lacroix, and William El Sayed. 2024. Mixture of experts .	
710	Seungone Kim, Jamin Shin, Yejin Cho, Joel Jang, Shayne Longpre, Hwaran Lee, Sangdoon Yun, Seongjin Shin, Sungdong Kim, James Thorne, and Minjoon Seo. 2024a. Prometheus: Inducing fine-grained evaluation capability in language models . In <i>The Twelfth International Conference on Learning Representations</i> .	
711	Seungone Kim, Juyoung Suk, Ji Yong Cho, Shayne Longpre, Chaeun Kim, Dongkeun Yoon, Guijin Son, Yejin Cho, Sheikh Shafayat, Jinheon Baek, Sue Hyun Park, Hyeonbin Hwang, Jinkyung Jo, Hyowon Cho, Haebin Shin, Seongyun Lee, Hanseok Oh, Noah Lee, Namgyu Ho, Se June Joo, Miyoung Ko, Yoonjoo Lee, Hyungjoo Chae, Jamin Shin, Joel Jang, Seonghyeon Ye, Bill Yuchen Lin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024b. The biggen bench: A principled benchmark for fine-grained evaluation of language models with language models .	
712	Ryan Koo, Minhwa Lee, Vipul Raheja, Jong Inn Park, Zae Myung Kim, and Dongyeop Kang. 2023. Benchmarking cognitive biases in large language models as evaluators .	
713	Richard Kraut. 2022. Aristotle’s Ethics. In Edward N. Zalta and Uri Nodelman, editors, <i>The Stanford Encyclopedia of Philosophy</i> , Fall 2022 edition. Metaphysics Research Lab, Stanford University.	
714	Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In <i>Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles</i> .	
715	Junlong Li, Shichao Sun, Weizhe Yuan, Run-Ze Fan, haizhao, and Pengfei Liu. 2024. Generative judge for evaluating alignment . In <i>The Twelfth International Conference on Learning Representations</i> .	
716	Wang Ling, Dani Yogatama, Chris Dyer, and Phil Blunsom. 2017. Program induction by rationale generation: Learning to solve and explain algebraic word problems. In <i>Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)</i> , pages 158–167, Vancouver, Canada. Association for Computational Linguistics.	
717	Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis,	
718	Luke Zettlemoyer, and Veselin Stoyanov. 2020. Ro{bert}a: A robustly optimized {bert} pretraining approach .	757
719		758
720		759
721	Yiqi Liu, Nafise Sadat Moosavi, and Chenghua Lin. 2024. Llms as narcissistic evaluators: When ego inflates evaluation scores .	760
722		761
723		762
724	Adian Liusie, Potsawee Manakul, and Mark Gales. 2024. LLM comparative assessment: Zero-shot NLG evaluation through pairwise comparisons using large language models . In <i>Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)</i> , pages 139–151, St. Julian’s, Malta. Association for Computational Linguistics.	763
725		764
726		765
727		766
728		767
729		768
730		769
731		770
732	Juhyun Oh, Eunsu Kim, Inha Cha, and Alice Oh. 2024. The generative AI paradox in evaluation: “what it can solve, it may not evaluate” . In <i>Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop</i> , pages 248–257, St. Julian’s, Malta. Association for Computational Linguistics.	771
733		772
734		773
735		774
736		775
737		776
738		777
739	Arjun Panickssery, Samuel R. Bowman, and Shi Feng. 2024. Llm evaluators recognize and favor their own generations .	778
740		779
741		780
742	F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. 2011. Scikit-learn: Machine learning in Python. <i>Journal of Machine Learning Research</i> , 12:2825–2830.	781
743		782
744		783
745		784
746		785
747		786
748		787
749	Barbara Plank. 2022. The “problem” of human label variation: On ground truth in data, modeling and evaluation . In <i>Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing</i> , pages 10671–10682, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.	788
750		789
751		790
752		791
753		792
754		793
755	Vyas Raina, Adian Liusie, and Mark Gales. 2024. Is llm-as-a-judge robust? investigating universal adversarial attacks on zero-shot llm assessment .	794
756		795
		796
	Skipper Seabold and Josef Perktold. 2010. statsmodels: Econometric and statistical modeling with python. In <i>9th Python in Science Conference</i> .	797
		798
		799
	Zhen Tan, Dawei Li, Song Wang, Alimohammad Beigi, Bohan Jiang, Amrita Bhattacharjee, Mansoor Karami, Jundong Li, Lu Cheng, and Huan Liu. 2024. Large language models for data annotation: A survey .	800
		801
		802
		803
		804
	Yidong Wang, Zhuohao Yu, Wenjin Yao, Zhengran Zeng, Linyi Yang, Cunxiang Wang, Hao Chen, Chaoya Jiang, Rui Xie, Jindong Wang, Xing Xie, Wei Ye, Shikun Zhang, and Yue Zhang. 2024a. PandaLM: An automatic evaluation benchmark for LLM instruction tuning optimization . In <i>The Twelfth International Conference on Learning Representations</i> .	805
		806
		807
		808
		809
		810
		811

812	Zhilin Wang, Yi Dong, Olivier Delalleau, Jiaqi	Yang, Tao Yu, Wen Xie, Wenhao Huang, Xiaohui	870
813	Zeng, Gerald Shen, Daniel Egert, Jimmy J. Zhang,	Hu, Xiaoyi Ren, Xinyao Niu, Pengcheng Nie, Yuchi	871
814	Makeish Narsimhan Sreedhar, and Oleksii Kuchaiev.	Xu, Yudong Liu, Yue Wang, Yuxuan Cai, Zhenyu Gu,	872
815	2024b. Helpsteer2: Open-source dataset for training	Zhiyuan Liu, and Zonghong Dai. 2024. Yi: Open	873
816	top-performing reward models.	foundation models by 01.ai.	874
817	Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten	Weizhe Yuan, Richard Yuanzhe Pang, Kyunghyun Cho,	875
818	Bosma, brian ichter, Fei Xia, Ed H. Chi, Quoc V Le,	Xian Li, Sainbayar Sukhbaatar, Jing Xu, and Jason	876
819	and Denny Zhou. 2022. Chain of thought prompt-	Weston. 2024. Self-rewarding language models.	877
820	ing elicits reasoning in large language models. In	Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan	878
821	<i>Advances in Neural Information Processing Systems.</i>	Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin,	879
822	Peter West, Ximing Lu, Nouha Dziri, Faeze Brahman,	Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang,	880
823	Linjie Li, Jena D. Hwang, Liwei Jiang, Jillian Fisher,	Joseph E. Gonzalez, and Ion Stoica. 2023. Judging	881
824	Abhilasha Ravichander, Khyathi Chandu, Benjamin	LLM-as-a-judge with MT-bench and chatbot arena.	882
825	Newman, Pang Wei Koh, Allyson Ettinger, and Yejin	In <i>Thirty-seventh Conference on Neural Information</i>	883
826	Choi. 2024. The generative AI paradox: “what it can	<i>Processing Systems Datasets and Benchmarks Track.</i>	884
827	create, it may not understand”. In <i>The Twelfth Inter-</i>		
828	<i>national Conference on Learning Representations.</i>		
829	Thomas Wolf, Lysandre Debut, Victor Sanh, Julien		
830	Chaumond, Clement Delangue, Anthony Moi, Pier-		
831	ric Cistac, Tim Rault, Remi Louf, Morgan Funtow-		
832	icz, Joe Davison, Sam Shleifer, Patrick von Platen,		
833	Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu,		
834	Teven Le Scao, Sylvain Gugger, Mariama Drame,		
835	Quentin Lhoest, and Alexander Rush. 2020. Trans-		
836	formers: State-of-the-art natural language processing.		
837	In <i>Proceedings of the 2020 Conference on Empirical</i>		
838	<i>Methods in Natural Language Processing: System</i>		
839	<i>Demonstrations</i> , pages 38–45, Online. Association		
840	for Computational Linguistics.		
841	Tianhao Wu, Weizhe Yuan, Olga Golovneva, Jing Xu,		
842	Yuandong Tian, Jiantao Jiao, Jason Weston, and Sain-		
843	bayar Sukhbaatar. 2024. Meta-rewarding language		
844	models: Self-improving alignment with llm-as-a-		
845	meta-judge.		
846	Wenda Xu, Guanglei Zhu, Xuandong Zhao, Liang-		
847	ming Pan, Lei Li, and William Yang Wang. 2024.		
848	Pride and prejudice: Llm amplifies self-bias in self-		
849	refinement.		
850	An Yang, Baosong Yang, Binyuan Hui, Bo Zheng,		
851	Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan		
852	Li, Dayiheng Liu, Fei Huang, Guanting Dong, Hao-		
853	ran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian		
854	Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin		
855	Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang		
856	Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang,		
857	Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng		
858	Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin,		
859	Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu,		
860	Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng,		
861	Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin		
862	Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang		
863	Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu		
864	Cui, Zhenru Zhang, and Zhihao Fan. 2024. Qwen2		
865	technical report. <i>arXiv preprint arXiv:2407.10671.</i>		
866	Alex Young, Bei Chen, Chao Li, Chengen Huang,		
867	Ge Zhang, Guanwei Zhang, Heng Li, Jiangcheng		
868	Zhu, Jianqun Chen, Jing Chang, Kaidong Yu, Peng		
869	Liu, Qiang Liu, Shawn Yue, Senbin Yang, Shiming		

	# questions	Avg. # question characters	Avg. # answer characters
AQUA-RAT	254	239.1	203.1
MATH	1516	216.5	643.9
GSM8K	1319	239.9	292.9

Table 5: An overview of dataset size and text length.

A Experimental Setup

We provide further details on the general setup described in Section 3. Specifically, we include statistics and examples of the datasets, additional information on the models used, and the exact prompts employed in this study.

A.1 Datasets

Additional information about the datasets is given in Table 5, which presents an overview of the dataset statistics. Note that for the MATH dataset, we only include the most challenging questions, called levels 4 and 5, in the dataset. Notably, it has ground truth answer sequences that are, on average, almost three times longer than those in other datasets.

In Table 6, we provide examples of questions and their corresponding answers from the ground truth. Note that these examples were used for few-shot prompting.

A.2 Models

We execute all models using the VLLM software for LLM serving (Kwon et al., 2023). The weights for all models are accessible through Huggingface Transformers (Wolf et al., 2020). Table 7 includes hyperlinks to each model for easy reference.

A.3 Prompts

We used two different prompts within this project. The prompt shown in Figure 7 is used for the candidate solutions for all datasets. Examples of the few-shots are in Table 6. The prompt for the judges is given in Figure 8. Note that we run experiments for both orders of the answers of the models M_A and M_B .

A.4 Infrastructure

The experiments were run on NVIDIA A100 and NVIDIA H100. The judgments used in Section 4 and Section 5 took around 5 days equivalents on 4 A100 40GB. Using 2 H100 90GB and 4 A100 40GB it took around 2.5 days. For the perturbation

Initial Prompt

```
{
  "role": "user",
  "content": "You are a reasoning assistant.
  Always answer exactly in the same format.
  Use '####' to separate the final answer
  (without additional comments) from the
  reasoning.

  {{shot 1 question}}"
},
{
  "role": "assistant",
  "content": "{{shot 1 question}}"
}
...
,
{
  "role": "assistant",
  "content": "{{shot 4 answer}}"
}, {
  "role": "user",
  "content": "{{question}}"
}
}
```

Figure 7: Prompt used to generate initial solutions for all datasets. It includes few-shots and the question of the current sample.

	Question	Answer
AQUA-RAT	Two friends plan to walk along a 43-km trail, starting at opposite ends of the trail at the same time. If Friend P’s rate is 15% faster than Friend Q’s, how many kilometers will Friend P have walked when they pass each other? Options: A)21 B)21.5 C)22 D)22.5 E)23	If Q complete x kilometers, then P completes $1.15x$ kilometers. $x + 1.15x = 43$ $2.15x=43$ $x = 43/2.15 = 20$ Then P will have have walked $1.15*20=23$ km. The answer is E. ##### E
GSM8K	Natalia sold clips to 48 of her friends in April, and then she sold half as many clips in May. How many clips did Natalia sell altogether in April and May?	Natalia sold $48/2 = \ll 48/2=24 \gg 24$ clips in May. Natalia sold $48+24 = \ll 48+24=72 \gg 72$ clips altogether in April and May. ##### 72
MATH	Mr. Madoff invests 1000 dollars in a fund that compounds annually at a constant interest rate. After three years, his investment has grown to 1225 dollars. What is the annual interest rate, as a percentage? (Round your answer to the nearest integer.)	Let r be the annual interest rate. Then after three years, Mr. Madoff’s investment is $1000 \cdot \left(1 + \frac{r}{100}\right)^3$, so $1000 \cdot \left(1 + \frac{r}{100}\right)^3 = 1225$. Then $\left(1 + \frac{r}{100}\right)^3 = 1.225$, so $\left[1 + \frac{r}{100} = \sqrt[3]{1.225} = 1.069987\dots\right]$, which means $r = \boxed{7}$, to the nearest integer. ##### 7.0

Table 6: Example of ground truth answers used for few-shot prompting.

Model	URL
Qwen2 72B	huggingface.co/Qwen/Qwen2-72B
Llama 3 70B	huggingface.co/meta-llama/Meta-Llama-3-70B-Instruct
Yi 1.5 34B	huggingface.co/01-ai/Yi-1.5-34B-Chat-16K
Mixtral 8x7B	huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1
Llama 3 8B	huggingface.co/meta-llama/Meta-Llama-3-8B-Instruct
Gemma 1.1 7B	huggingface.co/google/gemma-1.1-7b-it
Mistral 7B v0.3	huggingface.co/mistralai/Mistral-7B-Instruct-v0.3
Mistral 7B v0.1	huggingface.co/mistralai/Mistral-7B-Instruct-v0.1

Table 7: Used models and corresponding hyperlinks.

experiments in Section 7, it took around the same amount of time.

B General performance

This section provides additional information related to Section 4. Specifically, we present the task performance of all models across all datasets, as well as the judging performance of all models when used as judges.

B.1 Task Performance

In various contexts in this work, the task performance of the individual models is essential. Therefore, we provide the accuracy of all models and all datasets in Table 9.

B.2 Judging performance per model pair

We conduct experiments with all eight models serving as judges. We present the performance metrics

of all judges across various model comparisons in Figure 9. As shown in Table 1, only the large models consistently produce judgments that deviate consistently from random chance. These results in Figure 9 support the superior performance of larger models.

C Additional subset experiments

This section provides additional information for chapter 5.

C.1 Example Subset performance

To better understand the correlation observed in Figure 5, we provide examples of these subsets, which can be seen in Table 10. These examples include the following details: the judge, the compared models, the dataset, the performance of each model on a dataset (denoted by $P(A|D)$), the judgment performance on the subset (denoted by $P(\Delta_J = T|A = X, B = Y, D)$), and the relative size of the subset (denoted by $P(A = X, B = Y|D)$). We provide the five subsets with the highest performance, the five subsets with the five lowest performance, and five random subsets.

C.2 Performance by agreement

We provide an extension of the results in the main paper. We put all samples into bins of how many

Judge	model M_A	model M_B	Label	Prompt	Response	Correct
Qwen 2 72B	Qwen 2 72B	Llama 3 8B	20.00	<p>Question: Charlie wants to sell beeswax candles. For every pound of beeswax, he can make 10 tapered candles. One pound of beeswax and the wicks cost \$10.00 in supplies. If he sells each candle for \$2.00 each, what is his net profit if he makes and sells 20 candles?</p> <p>Answer A: He needs 2 pounds of beeswax to make 20 candles because $20 / 10 = 2$. It will cost \$20.00 in supplies for the beeswax and wicks because $2 \times 10 = 20$. His total revenue from selling the candles will be \$40.00 because $20 \times 2 = 40$. His net profit will be \$20.00 because $40 - 20 = 20$. #### 20</p> <p>Answer B: It takes 1 pound of beeswax and the wicks to make 10 candles, so it takes 1 pound of beeswax and the wicks to make 20 candles. It costs 10 dollars for 1 pound of beeswax and the wicks, so it costs 10 dollars for 20 candles. Charlie sells each candle for 2 dollars, so he makes $20 \times 2 = 40$ dollars by selling the 20 candles. So, his net profit is $40 - 10 = 30$ dollars. #### 30</p> <p>Compare both answers in detail and choose the answer which correctly answers the question. Conclude with a JSON in Markdown format indicating your choice between answer A or B:</p> <pre> '''json { "answer": "B" or "A" } ''' </pre>	<p>Answer A and Answer B differ in their calculation of the cost of supplies for making 20 candles. Answer A correctly calculates that Charlie needs 2 pounds of beeswax to make 20 candles and that the cost for the beeswax and wicks for 2 pounds is \$20.00. Answer B incorrectly assumes that the cost for making 20 candles is the same as for 10 candles, which is \$10.00. Therefore, Answer A correctly calculates the net profit as \$20.00, while Answer B incorrectly calculates it as \$30.00. The correct answer is A.</p> <pre> '''json { "answer": "A" } ''' </pre>	True
Llama 3 70B	Qwen 2 72B	Llama 3 8B	D	<p>Question: There is well of depth 30m and frog is at bottom of the well. He jumps 3m up one day and falls back 2m down the same day. How many days will it take for the frog to come out of the well? Options: A)25 days B)26 days C)27 days D)28 days E)29 days</p> <p>Answer A: The frog is initially at the bottom of the well, which is 30m deep. On the first day, the frog jumps 3m up, so it is 33m from the bottom. Then, it falls back 2m, so it is 31m from the bottom. Each day, the frog jumps 3m up and falls 2m down, making a net gain of 1m. To get out of the 30m deep well, it would take 30 days for the frog to jump and fall 30 times, covering a total distance of 30m. #### A</p> <p>Answer B: If he jumps 3m up and slips 2m down in a day, he is 1m up in a day. In 27 days he would have reached 27m. On the 28th day, he jumps up 3m and come out of the well. Answer: D #### D</p> <p>Compare both answers in detail and choose the answer which correctly answers the question. Conclude with a JSON in Markdown format indicating your choice between answer A or B:</p> <pre> '''json { "answer": "B" or "A" } ''' </pre>	<p>Let's break down both answers:</p> <p>**Answer A:** The explanation is correct that the frog makes a net gain of 1m each day. However, the conclusion that it would take 30 days to get out of the 30m deep well is incorrect. The frog would reach the top of the well when it has covered a total distance of 30m, but it would not take 30 days.</p> <p>**Answer B:** This explanation is correct. The frog makes a net gain of 1m each day, so in 27 days, it would have covered 27m. On the 28th day, it would jump up 3m and come out of the well. Therefore, the correct answer is **D) 28 days**. Here is the conclusion in JSON format:</p> <pre> '''json { "answer": "B" } ''' </pre>	True

Table 8: Example of ground truth answers used for few-shot prompting.

Judge Prompt			
Question:	{{question}}		
Answer A:	{{answer A}}		
Answer B:	{{answer B}}		
Compare both answers in detail and choose the answer which correctly answers the question.			
Conclude with a JSON in Markdown format indicating your choice between answer A or B:			
<pre> “json { "answer": "B" or "A" } “ </pre>			

Figure 8: Prompt used for judgements. The full text above is wrapped in the user role, as all models support this role. No additional system message is used.

	AQUA-RAT	GSM8K	MATH
Qwen 2 72B	76.38	92.04	51.19
Llama 3 70B	73.62	91.05	34.37
Yi 1.5 34B	64.96	78.47	27.04
Mixtral 8x7B	47.24	61.18	13.79
Llama 3 8B	51.18	73.01	15.04
Gemma 1.1 7B	42.91	50.72	12.60
Mistral 7B v0.3	38.19	42.76	6.13
Mistral 7B v0.1	21.65	26.08	3.10

Table 9: Task performance of all models using the prompt in Figure 7.

different answers were given by the eight models. For the sake of brevity, we average across all judges. In Figure 10 we show the same graphs for each judge individually. Again, we give the performance on the overall dataset and on the subset where there exists exactly one correct and one incorrect solution. All interpretations and conclusions overlap with the main part of the paper.

D Prediction of Individual Judgements

Random Forest. We utilize TF-IDF representations with English stop word removal for the Random Forest model. We further remove all digits from the text. We set the maximum number of features to 10,000 and consider n-grams ranging from unigrams to bigrams (N-gram range: 1-2). The model uses 500 trees (estimators). For training, we use the Scikit-learn (Pedregosa et al., 2011) library. The running time was negligible.

RoBERTa. For the RoBERTa model (Liu et al., 2020), we use a batch size of 64 and a learning rate of $2e-5$. The weight decay is set to $1e-3$, and the model is trained for 8 epochs. The final model is selected based on the best validation performance. The model is trained using the HuggingFace Transformers library (Wolf et al., 2020). The total running time was about twelve hours on a single H100 90GB.

E Statistical Methodology

We describe the statistical background for the tests applied in Section 6. All predictions and statistical tests in Section 6 were performed using the statsmodels library (Seabold and Perktold, 2010).

E.1 Coefficient of Determination

The coefficient of determination, R^2 , for evaluation of linear regression models (Fahrmeir et al., 2013) is defined as follows:

$$R^2 = \frac{\sum_{i=1}^n (\hat{y}_i - \bar{y})^2}{\sum_{i=1}^n (y_i - \bar{y})^2}$$

R^2 measures the share of the variance in Y explained by its covariation with the features \mathbf{X} included in the model by dividing the variation of the *predicted* values \hat{y}_i by the variation of the true target values y_i . If the features \mathbf{X} have high explanatory power for Y , the \hat{y}_i will be close to the y_i and R^2 will be close to 1, while in the extreme case of no correlation between \mathbf{X} and

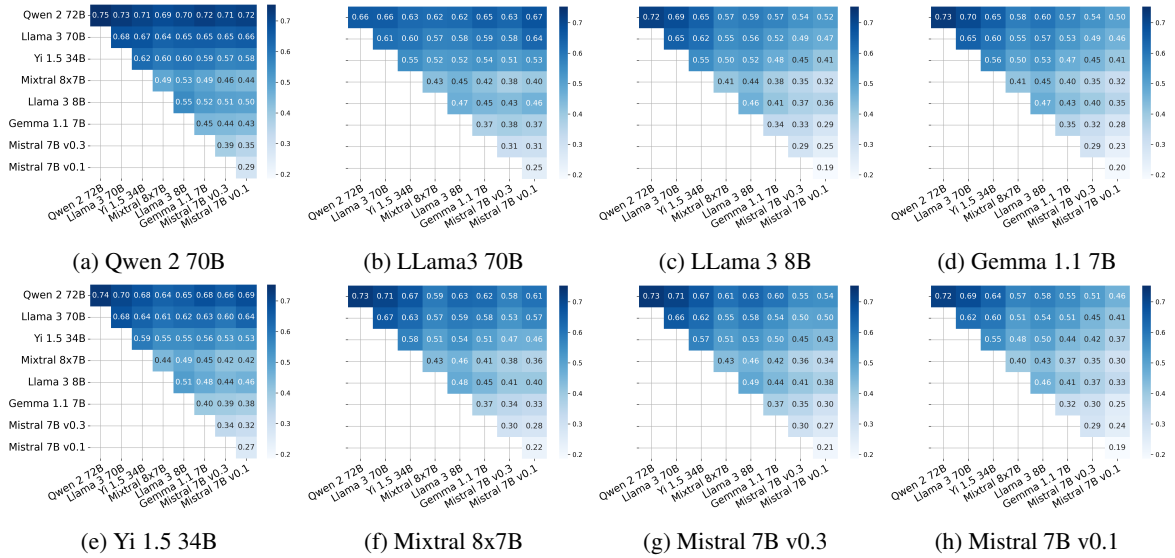


Figure 9: Evaluation of final task performance $P(\Delta_J = T|A, B)$ averaged over all datasets for model pairs (M_A, M_B) for the judges (a) - (h).

Judge	model A	model B	dataset	X	Y	$P(A D)$	$P(B D)$	$P(\Delta_J = T A = X, B = Y, D)$	$P(A = X, B = Y D)$
Qwen 2 72B	Qwen 2 72B	Mistral 7B v0.1	MATH	True	False	51.2	3.1	99.1	50.2
Qwen 2 72B	Yi 1.5 34B	Mistral 7B v0.1	MATH	True	False	27.0	3.1	98.4	27.9
Qwen 2 72B	Llama 3 8B	Mistral 7B v0.1	MATH	True	False	15.0	3.1	98.3	16.7
Qwen 2 72B	Llama 3 70B	Mistral 7B v0.1	MATH	True	False	34.4	3.1	98.3	35.4
Qwen 2 72B	Mixtral 8x7B	Mistral 7B v0.1	MATH	True	False	13.8	3.1	98.2	15.4
Mixtral 8x7B	Mixtral 8x7B	Mixtral 8x7B	GSM8K	False	True	61.2	61.2	65.8	14.7
Yi 1.5 34B	Llama 3 8B	Gemma 1.1 7B	GSM8K	False	True	73.0	50.7	64.9	8.0
Yi 1.5 34B	Llama 3 70B	Mistral 7B v0.1	AQUA-RAT	True	False	73.6	21.7	89.1	60.8
Qwen 2 72B	Gemma 1.1 7B	Gemma 1.1 7B	GSM8K	False	True	50.7	50.7	90.0	12.9
Yi 1.5 34B	Yi 1.5 34B	Mistral 7B v0.1	GSM8K	False	True	78.5	26.1	52.5	2.6
Qwen 2 72B	Llama 3 70B	Mistral 7B v0.1	MATH	False	True	34.4	3.1	13.2	2.2
Qwen 2 72B	Yi 1.5 34B	Mistral 7B v0.1	AQUA-RAT	False	True	65.0	21.7	10.0	4.7
Yi 1.5 34B	Qwen 2 72B	Mistral 7B v0.1	MATH	False	True	51.2	3.1	6.5	1.4
Llama 3 70B	Qwen 2 72B	Mistral 7B v0.1	MATH	False	True	51.2	3.1	6.2	1.3
Qwen 2 72B	Qwen 2 72B	Mistral 7B v0.1	MATH	False	True	51.2	3.1	6.1	1.4

Table 10: Examples of comparisons; and performance; problem:

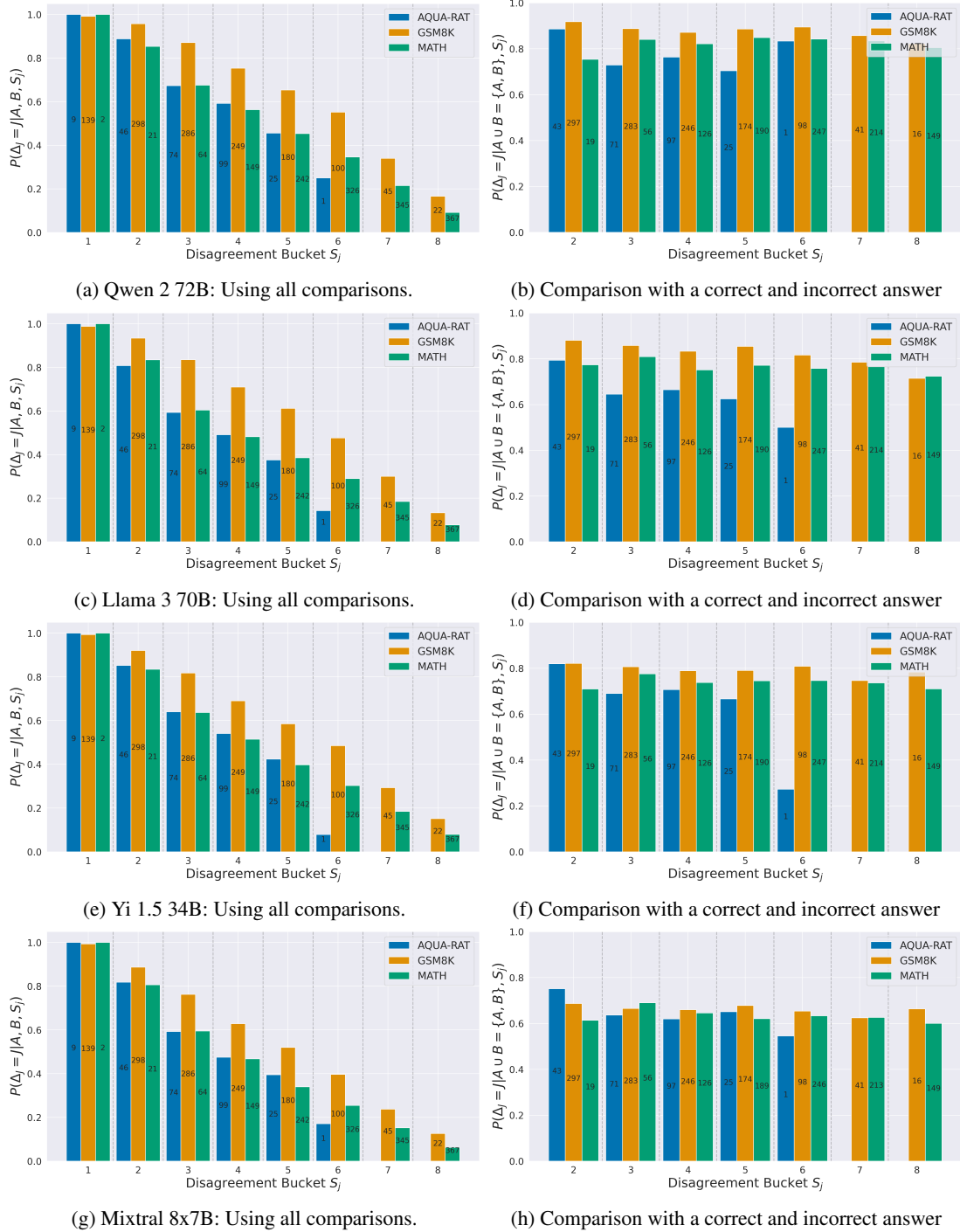


Figure 10: Judge performance by agreement bucket, e.g. bucket 3 (X-axis) means that all eight models gave together three different answers.

1010 Y the arithmetic mean is the best estimate (i.e.,
1011 $\hat{y}_i = \bar{y} \forall i = 1, \dots, n$) resulting in $R^2 = 0$.

1012 **E.2 Overall-F-Test**

1013 The Overall-F-Test is built upon R^2 and tests
1014 whether the overall model is of any significant
1015 value for explaining the variation of the target vari-
1016 able. The F-distributed test statistic is calculated
1017 as

$$\frac{R^2}{1 - R^2} \cdot \frac{n - p - 1}{p},$$

1018 where R^2 is the coefficient of determination, n is
1019 the number of observations, and p is the number of
1020 covariates included in the model (i.e., the number
1021 of estimated coefficients excluding the intercept).
1022 The hypotheses that can be tested this way are

$$H_0 : \beta_1 = \beta_2 = \dots = \beta_p = 0$$

vs.

$$H_1 : \beta_j \neq 0 \text{ for at least one } j \in \{1, \dots, p\}.$$

1023 So from a rejection of H_0 , it can be concluded
1024 that at least one of the included features exhibits
1025 explanatory power for the variation of the target
1026 variable.

1027 **E.3 Multiple Testing**

1028 Since we conduct multiple statistical tests within
1029 the scope of one research project, it is important
1030 to consider multiple testing as a potential problem
1031 resulting in false positive findings. The p-values
1032 from our tests, however, also satisfy a significance
1033 level resulting from a Bonferroni Correction of the
1034 typical significance level of 5%.

1035 **F Additional substitution Generations**

1036 In Section 7 we performed two experiments, where
1037 we 1) exchanged the results of the individual an-
1038 swers and 2) redacted all numbers using an 'X' to
1039 understand whether the model only focuses on the
1040 writing style. You can find examples including the
1041 prompt and the response of the judge for experi-
1042 ment 1) in Table 11 and examples for experiment
1043 2) in Table 12

Judge	model M_A	model M_B	Label	Prompt	Response	Correct
Qwen 2 72B	Qwen 2 72B	Llama 3 8B	163.00	<p>Question: Candice put 80 post-it notes in her purse before she headed out to her job at the coffee shop. On her way, she stopped off at the store and purchased a package of Post-it notes; At work, she placed a single Post-it note on each of 220 different cups of coffee. If she had 23 post-it notes remaining overall, how many Post-it notes were in the package that she purchased?</p> <p>Answer A: Candice had 80 post-it notes initially. She used $80 + 220 = 300$ of them to note the cups of coffee. She had 23 post-it notes remaining overall. So, she used $300 - 23 = 277$ of the 300 post-it notes initially. The remaining post-it notes are the difference between the total and the initial count, which is $300 - 80 = 220$. She must have purchased $220 - 23 = 197$ post-it notes. #### 163</p> <p>Answer B: Candice started with 80 Post-it notes and placed one on each of the 220 cups of coffee, so she used 220 notes. If she had 23 notes remaining, then she must have had $220 + 23 = 243$ notes in total. Since she originally had 80 notes, the package she purchased must have contained $243 - 80 = 163$ notes. Therefore, the package contained 163 Post-it notes. #### 197</p> <p>Compare both answers in detail and choose the answer which correctly answers the question. Conclude with a JSON in Markdown format indicating your choice between answer A or B:</p> <pre> """json { "answer": "B" or "A" } """ </pre>	<p>Answer A is incorrect because it incorrectly calculates the number of post-it notes Candice used and purchased. The correct answer is B because it accurately calculates the total number of post-it notes Candice had after using them and then subtracts the initial number of post-it notes she had to find out how many were in the package she purchased.</p> <pre> """json { "answer": "B" } """ </pre>	True
Llama 3 70B	Qwen 2 72B	Llama 3 8B	D	<p>Question: In a chocolate store, all chocolates are either vanilla or cocoa flavored only. 10% of the chocolates are cocoa flavored, 90% of the rest are squashed. What percentage of the chocolates are both vanilla flavored and not squashed? Options: A)1% B)2% C)5% D)9% E)10%</p> <p>Answer A: If 10% of the chocolates are cocoa flavored, then the remaining 90% are vanilla flavored. 90% of the vanilla chocolates are squashed. This means that 10% of the vanilla chocolates are not squashed. The question asks for the percentage of chocolates that are both vanilla flavored and not squashed. Since 10% of the vanilla chocolates are not squashed, the answer is 10%. #### D</p> <p>Answer B: 90% of the chocolates are not cocoa flavored (they are vanilla) and 10% of these ($90\% * 10\% = 9\%$) are not squashed. So 9% of the chocolates are both vanilla flavored and not squashed. The answer is E. #### E</p> <p>Compare both answers in detail and choose the answer which correctly answers the question. Conclude with a JSON in Markdown format indicating your choice between answer A or B:</p> <pre> """json { "answer": "B" or "A" } """ </pre>	<p>Let's break down both answers:</p> <p>Answer A: If 10% of the chocolates are cocoa flavored, then the remaining 90% are vanilla flavored. 90% of the vanilla chocolates are squashed. This means that 10% of the vanilla chocolates are not squashed. The question asks for the percentage of chocolates that are both vanilla flavored and not squashed. Since 10% of the vanilla chocolates are not squashed, the answer is 10%.</p> <p>Issue with Answer A: The calculation "10% of the vanilla chocolates are not squashed" is incorrect. 90% of the vanilla chocolates are squashed, which means 10% of 90% (not 10% of the total) are not squashed.</p> <p>Answer B: 90% of the chocolates are not cocoa flavored (they are vanilla) and 10% of these ($90\% * 10\% = 9\%$) are not squashed. So 9% of the chocolates are both vanilla flavored and not squashed.</p> <p>Correctness of Answer B: This answer correctly calculates the percentage of chocolates that are both vanilla flavored and not squashed. Therefore, the correct answer is B.</p> <p>Here is the conclusion in JSON format: """json</p> <pre> { "answer": "B" } """ </pre>	True

Table 11: Examples of comparisons where the final results were exchanged.

Judge	model M_A	model M_B	Label	Prompt	Response	Correct
Qwen 2 72B	Qwen 2 72B	Llama 3 8B	163.00	<p>Question: Candice put 80 post-it notes in her purse before she headed out to her job at the coffee shop. On her way, she stopped off at the store and purchased a package of Post-it notes; At work, she placed a single Post-it note on each of 220 different cups of coffee. If she had 23 post-it notes remaining overall, how many Post-it notes were in the package that she purchased?</p> <p>Answer A: Candice had X post-it notes initially. She used $X + X = \llcorner X + X = X \gg X$ of them to note the cups of coffee. She had X post-it notes remaining overall. So, she used $X - X = \llcorner X - X = X \gg X$ of the X post-it notes initially. The remaining post-it notes are the difference between the total and the initial count, which is $X - X = \llcorner X - X = X \gg X$. She must have purchased $X - X = \llcorner X - X = X \gg X$ post-it notes. ##### X</p> <p>Answer B: Candice started with X Post-it notes and placed one on each of the X cups of coffee, so she used X notes. If she had X notes remaining, then she must have had $X + X = X$ notes in total. Since she originally had X notes, the package she purchased must have contained $X - X = X$ notes. Therefore, the package contained X Post-it notes. ##### X</p> <p>Compare both answers in detail and choose the answer which correctly answers the question. Only analyze the reasoning! Therefore we exchanged all numbers with 'X' so you can focus on the reasoning. Conclude with a JSON in Markdown format indicating your choice between answer A or B:</p> <pre> """json { "answer": "B" or "A" } """ </pre>	<p>Answer A's reasoning is flawed because it uses the variable X in a confusing and incorrect way, equating X to X-X, which results in 0, and does not make sense in the context of the problem. Answer B, while also using X in a confusing way, at least attempts to set up an equation that could lead to a solution if the Xs were replaced with actual numbers. However, neither answer provides a clear, correct solution to the problem as stated.</p> <p>A correct approach would be to set up an equation based on the information given. Candice started with 80 post-it notes, used some, bought more, and ended up with 23 post-it notes. She used 220 post-it notes at work. Let's denote the number of post-it notes in the package she bought as Y.</p> <p>The equation based on the information given would be: Initial post-it notes + Post-it notes bought - Post-it notes used = Post-it notes remaining</p> $80 + Y - 220 = 23$ <p>Solving for Y would give us the number of post-it notes in the package she bought.</p> <p>Since neither answer A nor B provides a correct solution, and given the instruction to choose between A or B, we must conclude that neither is correct. However, since a choice must be made between A and B, and B at least attempts to set up an equation, albeit incorrectly, we will choose B as the less incorrect option.</p> <pre> """json { "answer": "B" } """ </pre>	True
Llama 3 70B	Qwen 2 72B	Llama 3 8B	D	<p>Question: There is well of depth 30m and frog is at bottom of the well. He jumps 3m up one day and falls back 2m down the same day. How many days will it take for the frog to come out of the well? Options: A)25 days B)26 days C)27 days D)28 days E)29 days</p> <p>Answer A: The frog is initially at the bottom of the well, which is X_m deep. On the first day, the frog jumps X_m up, so it is X_m from the bottom. Then, it falls back X_m, so it is X_m from the bottom.</p> <p>Each day, the frog jumps X_m up and falls X_m down, making a net gain of X_m. To get out of the X_m deep well, it would take X days for the frog to jump and fall X times, covering a total distance of X_m.</p> <p>##### A</p> <p>Answer B: If he jumps X_m up and slips X_m down in a day, he is X_m up in a day in X days he would have reached X_m On the Xth day, he jumps up X_m and come out of the well. Answer: D ##### D</p> <p>Compare both answers in detail and choose the answer which correctly answers the question. Only analyze the reasoning! Therefore we exchanged all numbers with 'X' so you can focus on the reasoning. Conclude with a JSON in Markdown format indicating your choice between answer A or B:</p> <pre> """json { "answer": "B" or "A" } """ </pre>	<p>Let's analyze the reasoning of both answers:</p> <p>**Answer A:** The answer assumes that the frog makes a net gain of X_m (3m - 2m = 1m) each day. However, it doesn't consider the fact that the frog needs to reach the top of the well to come out. The answer simply states that it would take X days to cover a total distance of X_m, without considering the remaining distance to the top of the well.</p> <p>**Answer B:** The answer correctly assumes that the frog makes a net gain of X_m (3m - 2m = 1m) each day. It then concludes that in X days, the frog would have reached X_m, and on the Xth day, it would jump up X_m and come out of the well. This answer considers the remaining distance to the top of the well and correctly concludes that the frog would come out on the Xth day.</p> <p>Based on the analysis, I choose:</p> <pre> """json { "answer": "B" } """ </pre>	True

Table 12: Example of comparisons where numbers were redacted using an 'X'.