

# Evaluating the Ability of LLMs to Detect Deliberately Inserted Errors in Full Research Papers

Nihar B. Shah, Carnegie Mellon University

## Overview

We created three versions of a paper, each containing a significant error in a purported key result. The errors pertain to a result about convexity of an optimization problem, a result about statistical identifiability, and one about empirical evaluation of the proposed algorithm. The papers and details of the errors are all available at

<https://github.com/niharshah/PaperCorrectnessCheck>

We evaluate whether large language models (LLMs) can identify these errors. We find that GPT-4 consistently finds one of the three errors and barely finds another error, but only when it is specifically asked about correctness of individual results.

## Prompts

Based on [Liu and Shah, 2023], we use the following prompt termed the Direct Prompt: *"You are an expert reviewer for a top ML/AI conference. You are provided with a paper submitted to the conference. Please check for the validity and correctness of these claims, and in particular, report if you can figure out if any of these claims is false based on the information provided in this paper. Think step by step when checking each claim."* Recognizing that the direct prompt had previously been used only for short papers [Liu and Shah, 2023], here we further prompt the LLM to verify the correctness of specific results. The idea is that if successful, this approach could be used in practice by breaking down the paper into individual results and having the LLM check each one separately. We tested each prompt three times and manually verified whether the LLM identified the errors.

## Summary of results

### - Paper with **optimization error**:

- *Direct prompt*: Did not identify the error in any of the three attempts.
- *"Is Theorem 2 correct? Please check carefully."*: Identified the error in all three attempts.
- *"Is (4) convex? Please check carefully."*: Identified the error in all three attempts.

### - Paper with **evaluation error**:

- *Direct prompt*: Did not identify the error in any of the three attempts.
- *"Is Algorithm 1 correct? Please check carefully."*: Did not identify the error in any of the three attempts.
- *"What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully."*: In one attempt, it identified the discrepancy that the same data is being used to choose hyperparameters and for the evaluation. However, it did not term it an error. With a

further prompt "Are you saying that hyperparameter selection is being done on the test data?" it identified the error. It did not identify the discrepancy or the error in the other two attempts.

- Paper with **statistical error**:

- *Direct prompt*: Did not identify the error in any of the three attempts.

- *"Is Theorem 1 correct? Please check carefully."*: Did not identify the error in any of the three attempts.

- *"Is the only if condition in Theorem 1 correct? Please check carefully."*: Did not identify the error in any of the three attempts.

Additionally, as a **control condition**, we tested the same questions on the correct version of the paper. In this control condition, GPT-4 did not report any error. Finally, we also tested with GPT-4o and Gemini Advanced. We found GPT-4o to be slightly inferior to GPT-4, whereas Gemini Advanced did not find any of the errors and also made other incorrect claims.

## Conclusions

To summarize, generically asking the LLM to evaluate the entire paper is ineffective. However, the following protocol that we adopted did exhibit some success: (i) Divide the paper's results into fine-grained modules; and (ii) Probe the LLM with specific questions about each individual module asking it to check for correctness. This process could be employed by authors before submission or by a conference as additional input for a reviewer or meta-reviewer, who could then inspect and verify the LLM's outputs.

## Acknowledgments

We are grateful to Andrej Risteski, Kangwook Lee and Charvi Rastogi for carefully reviewing the experimental manuscripts and for their extremely valuable feedback on it.

---

## LLM Responses

We now provide (typically one set of) responses from LLMs for checking errors in the experimental research papers.

<b>Responses by GPT-4 (May 6-8, 2024)</b>	<b>4</b>
<b>Treatment condition: Finding errors in erroneous parts</b>	<b>4</b>
<b>Optimization error</b>	<b>4</b>
Direct prompt	4
Is Theorem 2 correct? Please check carefully.	6
Is (4) convex? Please check carefully.	7
<b>Statistical error</b>	<b>8</b>
Direct prompt	8

Is Theorem 1 correct? Please check carefully.	10
Is the only if condition in Theorem 1 correct? Please check carefully.	11
<b>Evaluation error</b>	<b>13</b>
Direct prompt	13
Is Algorithm 1 correct? Please check carefully.	14
What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.	16
Are you saying that hyperparamter selection is being done on the test data?	17
Is Algorithm 1 correct? Please check carefully.	18
What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.	19
Is Algorithm 1 correct? Please check carefully.	20
What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.	21
<b>Control condition: Asking it to find errors in correct statements</b>	<b>21</b>
Is Algorithm 1 correct? Please check carefully.	21
What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.	23
Are you saying that hyperparamter selection is being done on the test data?	25
Is Theorem 1 correct? Please check carefully.	26
Is the only if condition in Theorem 1 correct? Please check carefully.	27
Is Theorem 2 correct? Please check carefully.	28
Is (5) convex? Please check carefully.	30
Is (6) convex? Please check carefully.	31
<b>Responses by GPT-4o (May 14, 2024)</b>	<b>32</b>
<b>Treatment condition: Finding errors in erroneous parts</b>	<b>32</b>
Optimization error	32
Direct prompt	32
Is Theorem 2 correct? Please check carefully.	34
Is (4) convex? Please check carefully.	36
Statistical error	38
Direct prompt	38
Is Theorem 1 correct? Please check carefully.	40
Is the only if condition in Theorem 1 correct? Please check carefully.	42
Evaluation error	44
Direct prompt	44
Is Algorithm 1 correct? Please check carefully.	46
What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.	48
<b>Responses by Gemini (May 14, 2024)</b>	<b>50</b>
<b>Treatment condition: Finding errors in erroneous parts</b>	<b>50</b>
Optimization error	50

Direct prompt	50
Statistical error	51
Direct prompt	52
Evaluation error	54
Direct prompt	54

---

## Responses by GPT-4 (May 6-8, 2024)

---

### Treatment condition: Finding errors in erroneous parts

#### Optimization error

##### Direct prompt

To evaluate the claims made in the paper you submitted for review, I will focus on each claim systematically, assessing its validity based on the information presented in the paper itself.

**\*\*Claim 1: The proposed models and estimators can use evaluator expertise or confidence to improve evaluation aggregation.\*\***

##### **\*\*Evaluation:\*\***

- The paper presents both parametric and non-parametric models that integrate levels of evaluator confidence or expertise as a way to weight individual scores in the aggregation process.
- Empirical results, particularly from crowdsourcing and peer review contexts, are shown to demonstrate the effectiveness of these methods in leveraging additional evaluator information to improve outcome accuracy.
- Based on the content provided, this claim is supported by both theoretical justifications (convexity and identifiability of the models) and empirical evidence from real-world data applications.

**\*\*Claim 2: The models are identifiable and convex, enabling reliable estimation and efficient computation.\*\***

##### **\*\*Evaluation:\*\***

- The identifiability of the models is discussed, with Theorem 1 asserting that the model parameters can be uniquely determined under certain conditions, specifically, the rank of a

particular matrix formed from the differences between expected and actual scores across different evaluator levels.

- Convexity is argued for each of the optimization problems defined in the models. Theoretical claims about the convex nature of these problems suggest that they can be solved efficiently using standard optimization techniques.
- The convexity of the parametric model (especially concerning problem (4)) is questionable based on the use of parameters in denominators, as previously discussed. While the paper claims that this model is convex under specific assumptions, this might require a more detailed mathematical verification to ensure that the convexity claims hold universally.

**\*\*Claim 3: Empirical evaluations of the methods reveal significant advantages over previous methods.\*\***

**\*\*Evaluation:\*\***

- The paper provides empirical results from using the proposed methods on data from Mechanical Turk and the ICLR conference.
- Comparative performance metrics are listed, showing that the proposed methods outperform several existing techniques, such as the Dawid-Skene model and other baseline aggregation strategies.
- These results are compelling, but as with all empirical research, they are limited to the datasets and specific conditions under which the experiments were conducted. The claim of "significant advantages" is strong, but it is supported by the presented results.

**\*\*Conclusion:\*\***

Most claims in the paper are well-supported by both theoretical frameworks and empirical testing. However, the claim about the convexity of the optimization problems, particularly the parametric estimator in problem (4), warrants a closer examination or more rigorous proof to ensure its accuracy in all possible scenarios. This aspect might require additional scrutiny or peer review feedback to fully validate. The rest of the claims align well with the methodologies and results documented in the paper.

---

Is Theorem 2 correct? Please check carefully.

To carefully evaluate the correctness of Theorem 2 in the paper, which claims that the optimization problems (4), (5), and (6) are all convex, we need to consider the structure and formulation of each optimization problem individually.

#### Review of Theorem 2:

##### Optimization Problem (4):

- **Objective**: Minimize the squared loss between the observed outcomes  $(y_i)$  and the model predictions, which are a convex combination of evaluations weighted by parameters  $(\beta)$  associated with different levels.

- **Formulation**:

$$\min_{\beta} \sum_{i=1}^n \left( y_i - \frac{x_i^T (\beta_{z_i,1}, \dots, \beta_{z_i,d_i})^T}{\sum_{j=1}^{d_i} \beta_{z_i,j}} \right)^2$$

Subject to  $(\epsilon \leq \beta_{\ell} \leq 1)$  for all  $(\ell \in |Z|)$ ,  $(\sum_{\ell \in |Z|} \beta_{\ell} = 1)$ , and potentially  $(\beta_1 \leq \beta_2 \leq \dots \leq \beta_{|Z|})$  if monotonicity is required.

- **Convexity Analysis**:

- The division by a sum of parameters  $(\sum_{j=1}^{d_i} \beta_{z_i,j})$  in the objective function introduces nonlinearity and potential non-convexity, as ratios of affine functions generally do not preserve convexity. The constraints are linear or bounded, which are themselves convex, but the overall objective function's convexity is suspect due to the ratio within it.

#### Optimization Problem (5):

- **Objective**: Similar to a quadratic loss function with regularization.

- **Convexity Analysis**:

- Given that both the loss and regularization terms are quadratic (and hence convex) and the constraints being linear, this problem formulation is convincingly convex.

#### Optimization Problem (6):

- **Objective**: Structured similarly to problem (5), dealing with a minimized quadratic loss function plus a regularization term over a different but finite set of evaluations.

- **Convexity Analysis**:

- As with problem (5), the quadratic loss function and regularization term maintain convexity, and the constraints are linear, supporting the claim of convexity.

### Conclusion on Theorem 2:

- While problems (5) and (6) seem convincingly convex due to their quadratic loss functions and linear constraints, problem (4) presents concerns regarding convexity due to the use of a ratio of affine functions in the objective function.

- The claim in Theorem 2 that **all** the optimization problems are convex could be misleading or incorrect unless further clarification or modification is provided for the formulation of problem (4). This requires a deeper mathematical analysis or an adjustment to ensure the convexity of the objective in problem (4), possibly by transforming the objective function or by providing stronger justifications for the structure used.

In summary, the correctness of Theorem 2 hinges critically on the convexity of problem (4), which, from the given analysis, seems potentially non-convex. This might necessitate a revision or a more detailed proof to uphold the theorem's claim comprehensively.

---

Is (4) convex? Please check carefully.

To carefully evaluate the convexity of optimization problem (4) from the paper, let's break down the components of the problem and analyze them:

#### ### Optimization Problem (4) Description

- **Objective**: Minimize the squared loss between the observed outcomes  $(y_i)$  and the predictions of a model, where predictions are a function of the evaluators' scores and levels:

$$\min_{\beta} \sum_{i=1}^n \left( y_i - \frac{x_i^T \beta_{z_i}}{\sum_{j=1}^{d_i} \beta_{z_i,j}} \right)^2$$

Here,  $(x_i)$  are the scores given by evaluators for item  $(i)$ ,  $(\beta_{z_i})$  are the parameters associated with the levels  $(z_i)$  of the evaluators, and  $(d_i)$  is the number of evaluations for item  $(i)$ .

- **Constraints**:

- $(\epsilon \leq \beta_{\ell} \leq 1)$  for all  $(\ell)$  in  $(|Z|)$  ensures that parameters are non-negative and bounded.
- $(\sum_{\ell \in |Z|} \beta_{\ell} = 1)$  ensures the parameters sum to one, maintaining a probability distribution.
- $(\beta_1 \leq \beta_2 \leq \dots \leq \beta_{|Z|})$  maintains a monotonic relationship between parameters, optional depending on the application.

#### ### Analysis of Convexity

1. **Objective Function Convexity**:

- The objective function includes a division by a sum of parameters,  $(\sum_{j=1}^{d_i} \beta_{z_i,j})$ , which can lead to non-convex behavior. While the squared loss  $(y_i - \text{prediction})^2$  is convex with respect to a linear prediction model, the division introduces a ratio of linear (or affine) functions, which does not generally preserve convexity.

2. **Constraints**:

- The constraints individually are convex:
  - Bounding constraints  $(\epsilon \leq \beta_{\ell} \leq 1)$  are linear and hence convex.
  - The equality constraint  $(\sum_{\ell \in |Z|} \beta_{\ell} = 1)$  is also linear and defines a convex set.

- Ordering constraints are linear and hence convex.

#### ### Critical Consideration:

- The major issue in asserting the convexity of the objective function lies in its non-standard form involving a division of sums of parameterized terms by sums of parameters. This type of function can lead to non-convex regions because the Hessian of such functions (a key determinant of convexity in differentiable functions) can have negative eigenvalues depending on the parameter values.

#### ### Conclusion:

- Based on this detailed review, problem (4) is **likely not convex** due to the objective function's structure involving a ratio of sums parameterized by  $(\beta)$ . The constraints are convex, but they do not suffice to guarantee the convexity of the overall optimization problem if the objective function itself is non-convex.

The claim in the paper that problem (4) is convex should be revisited, and further mathematical analysis or reformulation might be necessary to either confirm its convexity under specific conditions or to adjust the approach to ensure convexity is achieved, such as by reformulating the objective to avoid division by parameter sums.

---

## Statistical error

### Direct prompt

To determine the validity and correctness of the claims in the submitted paper for the conference, let's evaluate each claim step-by-step:

#### ### Claim Analysis Framework

1. **Definition of the Claim**: First, I will identify each claim made in the paper.
2. **Supporting Evidence**: Next, I will check if there are theoretical justifications, empirical data, or mathematical proofs provided in the paper that support each claim.
3. **Consistency with Existing Research**: I will compare the claims against the stated related literature to ensure consistency or identify discrepancies.
4. **Logical Consistency**: Evaluate whether the claims logically follow from the provided evidence and methodologies described in the paper.

#### ### Extracted Claims from the Paper

From the paper, several claims can be summarized as follows:



- The proposed models and estimators can improve the aggregation of evaluations where evaluators provide additional information such as confidence levels.
- Theoretical results regarding identifiability and convexity are solid and support the effectiveness of the proposed methods.
- Empirical evaluations show significant advantages of the proposed methods over previous techniques.

#### #### Evaluation of Claims

##### ##### Claim 1: Improvement in Aggregation Efficiency

- **\*\*Supporting Evidence\*\***: The paper provides both theoretical formulations and empirical evidence showing the benefit of incorporating levels of confidence or expertise into the aggregation process.
- **\*\*Consistency with Existing Research\*\***: The paper references and extends established models (e.g., Dawid-Skene model) to include additional parameters for evaluator confidence. This is consistent with existing research that also considers confidence levels in aggregation (referenced in the paper).
- **\*\*Logical Consistency\*\***: The introduction of confidence levels and expertise parameters in model formulations is logically sound and aligns well with the methodology of enhancing data aggregation accuracy.

##### ##### Claim 2: Theoretical Solidity

- **\*\*Supporting Evidence\*\***: The paper includes detailed theoretical results about the identifiability and convexity of the proposed estimators. These include conditions under which the models are identifiable and proofs that the optimization problems are convex.
- **\*\*Logical Consistency\*\***: Theoretical proofs and the conditions laid out for identifiability and convexity are mathematically rigorous. The assumptions made (e.g., non-adversarial evaluators) are reasonable within the context of the problems being addressed.

##### ##### Claim 3: Empirical Advantages

- **\*\*Supporting Evidence\*\***: Empirical results are presented using data from crowdsourcing and peer review settings. The paper details the performance improvements over previous methods and discusses how these were validated through experiments.
- **\*\*Consistency with Existing Research\*\***: The empirical methods are consistent with standard practices in the fields of crowdsourcing and peer review. Moreover, the paper appropriately compares its results with past work.
- **\*\*Logical Consistency\*\***: The empirical findings logically support the claim that the proposed methods provide advantages, given that they show better performance metrics in the experiments.

#### #### Conclusion

Based on the provided document and the evaluation framework:

- All major claims in the paper are supported by rigorous theoretical analysis, logical argumentation, and empirical evidence.

- The paper is consistent with existing literature, and its claims are enhanced by the detailed experimental setup and comparisons with established methods.

Hence, from the document provided, there do not appear to be any false claims. The paper adequately supports its assertions with theoretical and empirical evidence, and it aligns well with the existing body of research.

---

Is Theorem 1 correct? Please check carefully.

To verify the correctness of Theorem 1 in the paper, let's analyze it based on its definitions and proofs as described. Theorem 1 deals with the identifiability of the proposed model, which is crucial to ensure that unique parameter estimates can be obtained from the observed data.

#### ### Overview of Theorem 1

Theorem 1 states a condition under which the model proposed in the paper is identifiable. Identifiability, in this context, means that if two different parameter vectors lead to the same probability distribution over the observed data, then they must be the same vector.

#### ### Theorem 1 Claim

The claim made by Theorem 1 is that the model is identifiable if and only if a matrix  $\Sigma$  constructed from the data has full rank  $|Z|$ . Here  $|Z|$  is the number of possible levels associated with the evaluations.

#### ### The Matrix $\Sigma$

The matrix  $\Sigma$  is defined where its  $((i, \ell))$ -th entry captures the difference between the expected overall score for item  $i$  and the scores provided with level  $\ell$ , aggregated over all evaluations that share this level. Mathematically, it's expressed as:

$$\Sigma_{i,\ell} = \sum_{j \in [d_i]} (\mu_i - x_{i,j}) \mathbb{I}\{z_{i,j} = \ell\}$$

where  $\mu_i = E[y_i]$  is the expected score for item  $i$ ,  $x_{i,j}$  is the score given by evaluator  $j$  for item  $i$ , and  $z_{i,j}$  is the level of evaluation  $j$  for item  $i$ .

#### ### Proof Details

The proof focuses on the relationship between the matrix  $\Sigma$  and the parameter vector  $\beta^*$ . The key argument in the proof is that if the parameter vector can be uniquely recovered from the rank of  $\Sigma$ , then the model is identifiable. The proof relies on the assumption that distinct parameter vectors should result in distinct expectations of outcomes  $(E[y_i])$ , which in turn affects the probability distributions over the observed data.

#### ### Evaluation

1. **Logical Consistency**: The proof connects the identifiability of the model with the properties of the matrix  $\Sigma$ . If  $\Sigma$  has full rank, the linear system it forms with the parameter vector  $\beta^*$  (considering the normalization constraint) has a unique solution, thus ensuring identifiability. This is a standard approach in statistical modeling for checking identifiability.

2. **Mathematical Rigor**: The provided proof constructs a plausible argument that having full rank in the matrix  $\Sigma$  is both necessary and sufficient for the model's identifiability. The necessity and sufficiency are well-established methods for such proofs in statistics and machine learning.

3. **Assumptions and Limitations**: The proof assumes that the scores and levels are such that the matrix  $\Sigma$  captures all necessary variations in data to reflect changes in parameters. It also assumes non-degenerate cases where the model's assumptions (like non-zero lower bounds on  $\beta^*$ ) hold.

### ### Conclusion

Based on the paper's content and the typical standards of statistical theory, Theorem 1 appears to be correct in its claim about the identifiability of the model, provided the assumptions mentioned are met and the matrix  $\Sigma$  indeed captures the necessary data variations as described. The proof is consistent with common practices in theoretical statistics regarding model identifiability. Thus, there are no apparent errors in Theorem 1 based on the information provided in the paper.

---

Is the only if condition in Theorem 1 correct? Please check carefully.

The "only if" part of Theorem 1 in the paper states that for the model to be identifiable, it is necessary that the matrix  $\Sigma$  has full rank  $|Z|$ . This condition is about ensuring that different parameter settings ( $\beta^*$ ) lead to different distributions of the data ( $y_i$ ), which is a key aspect of identifiability.

### ### Delving Deeper into the "Only If" Condition

To determine the correctness of the "only if" part of Theorem 1, let's examine the rationale and implications:

#### 1. **Definition of $\Sigma$ and its Role**:

- $\Sigma$  is constructed such that each element  $\Sigma_{i,\ell}$  aggregates the weighted differences between the expected score  $\mu_i$  and the individual scores  $x_{i,j}$  associated with level  $\ell$ .

- The construction of  $\Sigma$  aims to capture how changes in the parameter  $\beta^*$  affect the expected outcomes  $\mu_i$ .

2. **Implication of Full Rank**:

- Full rank of  $\Sigma$  implies that the columns of  $\Sigma$  are linearly independent, meaning each "level" represented in  $\Sigma$  provides unique information that cannot be linearly derived from the other levels.
- If  $\Sigma$  does not have full rank, there exists some linear dependency among the columns corresponding to different levels. This implies that changes in the parameters corresponding to some levels could be mimicked by changes in parameters corresponding to other levels, leading to potential non-identifiability because different  $\beta^*$  vectors might yield the same outcomes.

3. **Mathematical Interpretation**:

- The "only if" condition is focusing on ensuring that no two different parameter settings result in the same expected outcomes across all evaluations. This is critical because identifiability is violated if different parameter settings can produce the same distribution over the data.

4. **Example to Test the Condition**:

- Suppose every evaluation for different items had the same level, and therefore, every column in  $\Sigma$  related to that level would be identical (or a scalar multiple). This would make  $\Sigma$  rank-deficient. If we can construct a scenario where this leads to non-identifiability (different  $\beta^*$  yielding the same distribution), it would support the necessity of the full rank condition.

5. **Critically Analyzing the Proof**:

- The proof should ideally demonstrate that a rank-deficient  $\Sigma$  can indeed lead to non-identifiable situations, possibly by constructing an explicit example or showing theoretically how such scenarios lead to the same distributions for different parameters.
- If the proof correctly argues that a rank-deficient  $\Sigma$  leads to non-identifiable parameter settings by showing a direct link between the rank deficiency and the resultant indistinguishability in outcomes, the "only if" condition is substantiated.

### Conclusion

The correctness of the "only if" condition hinges on a rigorous proof that demonstrates non-identifiability specifically arising from a rank-deficient  $\Sigma$ . Based on typical mathematical standards in statistical modeling, if the proof rigorously establishes this link, then the "only if" condition is correct. Without access to specific proof details beyond the summary, it's critical to assume the proof adequately addresses these aspects, aligning with established statistical principles. If there is any ambiguity or lack of clarity in how rank deficiency directly leads to non-identifiable scenarios, further scrutiny of the proof would be warranted.

---

---

## Evaluation error

### Direct prompt

To evaluate the claims made in the provided paper, let's go through the content step-by-step, checking the validity and correctness of specific assertions based on the information given:

1. **\*\*Claim about the Theoretical Foundations\*\***: The paper claims to provide theoretical results regarding the identifiability and convexity of the proposed estimators. These claims are supported by detailed mathematical models and proofs within the "Theoretical results" section, where they discuss identifiability conditions and convexity of their estimation problems, providing sufficient mathematical justification. Therefore, based on the information provided, this claim seems valid.
2. **\*\*Claim about Empirical Evaluations\*\***: The paper asserts that empirical evaluations of their methods using crowdsourcing and peer review data reveal significant advantages. The "Experimental results" section includes detailed descriptions of experiments and statistical comparisons with existing methods. The results presented include error rates and performance metrics that demonstrate improvements over previous methods. This claim is substantiated by the data and comparisons provided in the experiments section.
3. **\*\*Claim about Enhancing Label Accuracy with Gold Standard Questions\*\***: The paper proposes using gold standard questions to enhance label accuracy in crowdsourced settings. They argue that leveraging these questions can improve the accuracy of labels for non-gold standard questions. The experiments conducted show that their methods, when utilizing gold standard questions, provide better accuracy compared to other models that do not use this information. The discussion of results substantiates this claim, supported by the empirical data provided.
4. **\*\*Claim about Understanding Meta-Reviewer Aggregation in Peer Review\*\***: The paper aims to quantitatively understand how meta-reviewers aggregate individual assessments. They provide a model to estimate the influence of different confidence levels of reviewers on final decisions. Results from data of the ICLR conference are used to show how different levels of reviewer confidence influence meta-review decisions. This claim is validated by the analysis and results presented for the peer review dataset.
5. **\*\*Claim about the Applicability of Models to Other Domains\*\***: The introduction and discussion sections claim that while the primary focus is on crowdsourcing and peer review, the methods are relevant in other domains such as admissions or hiring. This claim is speculative and broader than the specific results provided, as no empirical data from these other domains are presented. This is more of a theoretical or potential applicability claim rather than a demonstrated one based on the given data.

In summary, most claims related to theoretical foundations and specific applications in crowdsourcing and peer review are substantiated and appear valid based on the detailed information and results provided in the paper. However, broader claims about applicability to other domains are not directly substantiated by empirical evidence within those domains in the paper itself.

---

Is Algorithm 1 correct? Please check carefully.

To determine the correctness of Algorithm 1 from the paper, we'll break down its steps and evaluate if they logically and effectively address the goal of the algorithm. Algorithm 1 is designed to determine the optimal value of the hyperparameter  $\lambda$  in the non-parametric estimator and to evaluate the resulting performance for non-gold standard questions.

Here's a detailed look at Algorithm 1:

#### ### Algorithm 1: Implementation and Evaluation

1. **Let  $G$  denote the set of all gold standard questions and  $H$  denote the set of all non-gold standard questions**
  - **Explanation**: Defines two sets of questions, gold standard ( $G$ ) which are used for training, and non-gold standard ( $H$ ) which are used for testing the model's performance.
2. **for  $\lambda \in \{0, 2^{-16}, 2^{-15}, \dots, 2^{15}, 2^{16}\}$  do**
  - **Explanation**: Iterates over a range of values for  $\lambda$  from  $2^{-16}$  to  $2^{16}$ , including 0, in powers of 2. This range tests a variety of regularization strengths from very weak to very strong, as well as no regularization.
3. **Solve (6) using  $G$  and denote the result as  $\phi_\lambda$** 
  - **Explanation**: For each  $\lambda$ , the estimator (referred to as Equation 6 in the paper) is computed using the gold standard questions. This results in a model  $\phi_\lambda$  optimized for that specific  $\lambda$ .
4. **for each  $h$  in  $H$  do**
  - **Compute  $\zeta_\lambda(h) = \mathbb{I}(\phi_\lambda(x_{h,1}, z_{h,1}, x_{h,2}, z_{h,2}, x_{h,3}, z_{h,3}) \neq y_h)$** 
    - **Explanation**: For each non-gold standard question  $h$  in  $H$ , the model  $\phi_\lambda$  is used to predict the label, and the prediction is compared to the actual label  $y_h$ . The indicator function  $\mathbb{I}$  returns 1 if the prediction is incorrect and 0 if it is correct. This step calculates the

prediction error for each non-gold standard question using the model derived from each  $\lambda$ .

5. **end for**

6. **end for**

7. **Computing error rate**

- **Let  $\zeta = \min_{\lambda} \{\text{Mean}(\zeta_{\lambda, h})_{h \in H}\}$**

- **Explanation**: After evaluating all non-gold standard questions for each  $\lambda$ , the algorithm selects the  $\lambda$  that yields the lowest average error rate across all non-gold standard questions. This is the optimal  $\lambda$  for the model.

### ### Evaluation of Correctness

The algorithm is logically structured to achieve its aim of finding the optimal  $\lambda$  for the non-parametric model. The steps are clear, and the procedure follows a standard approach to hyperparameter tuning using validation data (non-gold standard questions serve as a form of validation or test set). The indicator function and the minimization step correctly implement the performance evaluation and selection of the best  $\lambda$ .

Thus, based on the information provided and the typical procedures in machine learning for model validation and testing, **Algorithm 1 appears correct and is appropriately designed** to find the best  $\lambda$  value by minimizing the prediction error on non-gold standard questions.

---

What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.

Algorithm 1 uses the set of non-gold standard questions (denoted as  $H$ ) for testing, i.e., evaluating the performance of the model parameters  $\phi_{\lambda}$  determined using the gold standard questions (denoted as  $G$ ). The process involves using the model derived for each value of  $\lambda$  to predict the outcomes for the questions in  $H$ , and then calculating the error rates based on these predictions.

### ### Appropriateness of the Data Usage:

1. **Separation of Training and Testing Data**: The fundamental principle in machine learning and statistical modeling is to separate the data used for training the model from the data used for testing its performance. This separation prevents the model from being overly fitted to the specific sample of data used for training (overfitting), which can lead to misleadingly high performance on training data but poor performance on new, unseen data.

2. **Gold Standard vs. Non-Gold Standard Questions**:

- **Gold Standard Questions ( $G$ )**: These are used to train the model. They likely include known answers, which allow the model to learn the correct way to aggregate responses based on different levels of confidence or expertise. This is crucial for tuning the model parameters accurately.
- **Non-Gold Standard Questions ( $H$ )**: These are used to test how well the model, trained on  $G$ , performs on unseen data. The purpose here is to evaluate the generalizability of the model to new data that mimics the real-world application where the true labels might not be known.

3. **Appropriateness of Using  $H$  for Testing**:

- Using  $H$  for testing is appropriate as it simulates the real-world scenario where the model must make predictions on new data without prior exposure. The key is that the model's parameters are not tuned based on  $H$  but are instead derived solely from  $G$ . This ensures that the evaluation of the model's performance on  $H$  is unbiased and indicative of how it would perform in practical applications.
- The choice of  $\lambda$  is crucial because it controls the level of regularization in the model, affecting its ability to generalize beyond the training data. Testing across a range of  $\lambda$  values helps identify the best setting that balances fit and generalization.

4. **Statistical Validation**:

- The process outlined in Algorithm 1, where  $\lambda$  is varied and the model's prediction errors on  $H$  are computed for each  $\lambda$ , effectively implements a form of cross-validation. This is a robust method for testing model performance and selecting hyperparameters.

5. **Conclusion on Appropriateness**:

- The use of non-gold standard questions  $H$  for testing in Algorithm 1 is appropriate and follows best practices for model evaluation. It ensures that the model, when deployed, will likely perform well on new, unseen data, provided that the data characteristics of  $H$  are representative of the target application's data.

Overall, Algorithm 1's use of  $H$  for testing is methodologically sound and crucial for ensuring the robustness and reliability of the model in practical scenarios.

---

Are you saying that hyperparameter selection is being done on the test data?

You are correct to raise a concern here, and I appreciate the clarification needed on this point. Using the non-gold standard questions  $H$  directly for the final performance evaluation as well as for selecting the hyperparameter  $\lambda$  can potentially lead to biased evaluations.



This would be similar to using the test data for model tuning, which is not a recommended practice as it can lead to overfitting to the test set and not genuinely reflect the model's performance on unseen data.

#### ### Ideal Approach for Hyperparameter Tuning and Testing:

1. **Hyperparameter Tuning**: This should ideally be done using a separate validation set or through techniques like cross-validation within the training set (gold standard questions  $\setminus G \setminus$  in this context). This allows the model to adjust its parameters without being influenced by the data meant to test its final performance.

2. **Final Evaluation**: The test set, which should not have been used in the training or validation process, is then used to evaluate the model. This ensures the evaluation is unbiased and reflects how the model will perform in real-world scenarios on unseen data.

#### ### Issues in Algorithm 1:

- If  $\setminus H \setminus$  is used for both selecting  $\setminus \lambda \setminus$  and testing the performance of the model, it means the test data is influencing the model configuration. This can lead to an optimistic bias in the evaluation results because the model is indirectly being optimized for this specific set of test data.

#### ### Recommendations:

- **Split  $\setminus G \setminus$  for Training and Validation**: Use part of the gold standard questions for training and another part for validating different settings of  $\setminus \lambda \setminus$ .

- **Use  $\setminus H \setminus$  strictly for Testing**: Once the best  $\setminus \lambda \setminus$  is selected based on the validation performance, this parameter can then be used to run the final model on  $\setminus H \setminus$  to genuinely test its performance on unseen data.

In summary, while the use of  $\setminus H \setminus$  in Algorithm 1 as described may seem practical for assessing how different  $\setminus \lambda \setminus$  values affect performance, it's technically incorrect for hyperparameter tuning as it uses the test data for this purpose. A more rigorous approach would separate these stages to avoid data leakage and ensure the model's evaluation is both accurate and reliable.

---

Is Algorithm 1 correct? Please check carefully.

I've reviewed Algorithm 1 and the mathematical models you presented in your document. The models and the algorithm seem theoretically sound based on their detailed description and the assumptions made. They incorporate significant elements of statistical modeling, such as identifiability and convexity, which are crucial for ensuring the robustness of the models. The document also discusses the implementation of these models in real-world data scenarios (crowdsourcing and peer review), providing empirical evidence of their effectiveness.

Algorithm 1, specifically used for evaluating the performance of the non-parametric estimator with varying  $\lambda$  values, appears correct in its structure and objectives. The algorithm systematically explores different values of  $\lambda$  to minimize the error rate in non-gold standard questions, which aligns well with the aims of the document to refine and optimize data aggregation methods.

Here's a summary of the key elements of your document that support the correctness of Algorithm 1:

- **Statistical Modeling**: The models used are based on convex optimization, which ensures that the solutions are global minima, making the estimations reliable and robust.
- **Experimentation and Validation**: The use of empirical data to validate the models and algorithms further supports their correctness. The results from crowdsourced labeling and peer review scenarios demonstrate the practical effectiveness of the proposed methods.
- **Theoretical Underpinnings**: The theoretical results about identifiability and convexity are critical in statistical modeling, ensuring that the models are well-defined and that their parameters can be uniquely determined from the data.

Assuming that there are no transcription errors in the algorithm or the models from your description to the actual implementation, and that the empirical data used for validation accurately reflects real-world distributions, Algorithm 1 should perform as intended according to the theoretical and empirical evidence presented. However, practical deployment should be accompanied by continuous monitoring to handle potential discrepancies in data that might not perfectly align with theoretical assumptions.

---

What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.

Algorithm 1 in your document uses data from the set of all non-gold standard questions (denoted by  $\mathcal{H}$ ) for testing (evaluation). This set  $\mathcal{H}$  consists of questions that were not part of the gold standard set used to fit or calibrate the models. Instead,  $\mathcal{H}$  includes the remaining questions where the true labels are available but were not used during the optimization process described in Algorithm 1.

Here's how Algorithm 1 uses this data for evaluation:

1. **Parameter Tuning**: The algorithm iterates over a range of values for  $\lambda$ , each time using the gold standard set  $\mathcal{G}$  to solve the optimization problem and derive the function  $\phi_{\lambda}$ .

2. **Testing**: For each value of  $\lambda$ , the derived function  $\phi_{\lambda}$  is then applied to each question in  $H$ , and the error rate  $\zeta_{\lambda,h}$  is computed by checking if the predicted label matches the true label.

#### #### Appropriateness of the Data Use

##### **Strengths**:

- **Separation of Training and Testing**: Using non-gold standard questions for testing ensures that the model is evaluated on data it has not seen during training, providing a measure of the model's generalization capability.
- **Diversity of Data**: Given the diversity in the non-gold standard questions, the testing is likely comprehensive, covering various scenarios that might not be fully represented in the gold standard questions.

##### **Potential Issues**:

- **Representativeness of the Gold Standard**: The effectiveness of the testing heavily relies on the representativeness of the gold standard questions. If the gold standard questions are not representative of the broader set of questions in terms of difficulty, topic, or other relevant factors, the model may not generalize well, despite good performance on  $H$ .
- **Distributional Similarity**: If there is a significant difference in the distribution of labels or features between the gold standard and non-gold standard questions, the performance on  $H$  might not accurately reflect the model's effectiveness in real-world scenarios.

#### #### Conclusion

Using non-gold standard questions for testing in Algorithm 1 is generally appropriate as it adheres to the principle of evaluating a model on unseen data, which is crucial for assessing the model's predictive power. However, the effectiveness of this approach depends on the assumption that both the gold standard and non-gold standard questions are sampled from the same population or have similar distributions. Any discrepancies in this assumption should be addressed by ensuring a diverse and representative selection of gold standard questions.

---

Is Algorithm 1 correct? Please check carefully.

Algorithm 1 uses the set of non-gold standard questions, denoted as  $H$ , for testing and evaluating the performance of the estimators. This is stated in the algorithm where each  $h$  in  $H$  is used to compute the error rate  $\zeta_{\lambda,h}$ .

#### #### Evaluation Using Non-Gold Standard Questions:

- **Purpose**: The non-gold standard questions are those for which true labels are known but were not used in the training phase (i.e., they were not part of the gold standard set used to derive the estimates  $\phi_{\lambda}$ ).

- **Rationale**: This approach is in line with standard practices in supervised learning, where a model is trained on a known dataset (gold standard questions) and evaluated on a separate, unseen dataset (non-gold standard questions) to assess its generalization ability.

#### #### Appropriateness:

- **Independent Testing Data**: Using non-gold standard questions ensures that the estimator is tested on data that it has not seen during the training phase. This helps in evaluating the true predictive power of the estimator.

- **Avoiding Overfitting**: By not including these non-gold standard questions in the training set, the algorithm minimizes the risk of overfitting. Overfitting occurs when a model is too closely fitted to the training data, including noise and outliers, which can degrade its performance on new, unseen data.

- **Real-World Validation**: This method simulates a realistic scenario where a model trained on a subset of data (gold standard questions) is then used to make predictions on new data (non-gold standard questions). This is crucial for practical applications where the model needs to perform well on data that was not available or used during the model's development.

#### #### Conclusion:

The use of non-gold standard questions for testing in Algorithm 1 is entirely appropriate and aligns with best practices in machine learning and statistical modeling. It provides a robust method for assessing the generalization and practical applicability of the developed estimators. This approach enhances the validity of the experimental results reported in the paper, ensuring that they reflect the model's performance in realistic settings outside of the controlled environment of training.

---

What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.

Algorithm 1 uses the set of non-gold standard questions, denoted as  $H$ , for testing and evaluating the performance of the estimators. This is stated in the algorithm where each  $h$  in  $H$  is used to compute the error rate  $\zeta_{\lambda, h}$ .

#### #### Evaluation Using Non-Gold Standard Questions:

- **Purpose**: The non-gold standard questions are those for which true labels are known but were not used in the training phase (i.e., they were not part of the gold standard set used to derive the estimates  $\phi_{\lambda}$ ).

- **Rationale**: This approach is in line with standard practices in supervised learning, where a model is trained on a known dataset (gold standard questions) and evaluated on a separate, unseen dataset (non-gold standard questions) to assess its generalization ability.

#### #### Appropriateness:

- **Independent Testing Data**: Using non-gold standard questions ensures that the estimator is tested on data that it has not seen during the training phase. This helps in evaluating the true predictive power of the estimator.
- **Avoiding Overfitting**: By not including these non-gold standard questions in the training set, the algorithm minimizes the risk of overfitting. Overfitting occurs when a model is too closely fitted to the training data, including noise and outliers, which can degrade its performance on new, unseen data.
- **Real-World Validation**: This method simulates a realistic scenario where a model trained on a subset of data (gold standard questions) is then used to make predictions on new data (non-gold standard questions). This is crucial for practical applications where the model needs to perform well on data that was not available or used during the model's development.

### ### Conclusion:

The use of non-gold standard questions for testing in Algorithm 1 is entirely appropriate and aligns with best practices in machine learning and statistical modeling. It provides a robust method for assessing the generalization and practical applicability of the developed estimators. This approach enhances the validity of the experimental results reported in the paper, ensuring that they reflect the model's performance in realistic settings outside of the controlled environment of training.

---

## Control condition: Asking it to find errors in correct statements

### Is Algorithm 1 correct? Please check carefully.

To determine the correctness of Algorithm 1 as described in the paper, I will review its stated purpose and the details of its implementation steps. Algorithm 1 is used to evaluate the performance of a model under different values of the hyperparameter  $(\lambda)$ , to identify the best  $(\lambda)$  that minimizes error in a set of gold standard questions.

Here's a breakdown of Algorithm 1's steps:

1. **Initialize Sets of Questions**: Divide the questions into gold standard questions  $(G)$  and non-gold standard questions  $(H)$ .
2. **Parameter Sweep**: For each  $(\lambda)$  in a predefined set of values:
  - **Cross-validation on Gold Standard**: For each gold standard question  $(g)$  in  $(G)$ :
    - Solve the optimization problem (6) using the gold standard questions excluding  $(g)$  (leave-one-out cross-validation), and denote the result as  $(\phi_{\lambda})$ .
    - Evaluate the error  $(\zeta_{\lambda,g})$  on the left-out question  $(g)$  using the model  $(\phi_{\lambda})$ .

- Compute the average error  $\zeta_{\lambda}$  across all gold standard questions for this  $\lambda$ .

3. **Select Optimal  $\lambda$** : Identify  $\lambda^{\hat{}}$  that results in the lowest average error  $\zeta_{\lambda}$ .

4. **Evaluate on Non-Gold Standard Questions**: Using  $\lambda^{\hat{}}$ , compute the error rate  $\zeta$  for the non-gold standard questions.

#### ### Key Points to Check:

- **Correct Use of Cross-Validation**: The algorithm employs leave-one-out cross-validation on the gold standard questions to avoid overfitting and to ensure the robustness of  $\lambda$  selection. This is appropriate given the context where gold standard data is limited.

- **Error Calculation**: It checks the mismatch  $\zeta_{\lambda,g}$  for each left-out question, which is typical in model validation processes to measure performance.

- **Sweep Over  $\lambda$** : The sweep over  $\lambda$  values is comprehensive, covering a broad range from  $2^{-16}$  to  $2^{16}$ . This is a common practice to explore the effect of regularization or other hyperparameters on model performance.

- **Final Application on  $H$** : After determining the best  $\lambda$ , the model is tested on the non-gold standard questions, which is a critical step to validate the generalizability of the model.

#### ### Potential Issues:

- **Range and Scale of  $\lambda$** : The choice and scale of  $\lambda$  values could be arbitrary without context on the data or prior experimentation to justify this range. If the range is too wide or not centered around values that lead to meaningful regularization, it could miss the optimal regularization level.

- **Computational Efficiency**: While not directly a correctness issue, the computational cost of this approach can be high, especially with a large number of  $\lambda$  values and gold standard questions. Optimization efficiency and software implementation could impact practical usability.

- **Assumptions on Data and Model Fit**: The effectiveness of the algorithm also depends on the assumptions made by the models used in step 2 (optimization problem (6)). If these assumptions do not hold, the results may not be reliable.

From the review of the algorithm provided in the paper and without additional contextual details or empirical results to compare against, **Algorithm 1** appears correctly structured for its

intended purpose\*\* of hyperparameter tuning and validation. However, practical testing and validation against known benchmarks or through simulation studies would be essential to fully endorse its effectiveness and correctness in real-world scenarios.

---

What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.

Algorithm 1 is using two sets of questions for testing and evaluation purposes: gold standard questions  $\mathcal{G}$  and non-gold standard questions  $\mathcal{H}$ . Here's how each is used within the algorithm:

1. **Gold Standard Questions ( $\mathcal{G}$ )**:

- **Purpose**: Used for determining the optimal hyperparameter  $\lambda$  via a cross-validation approach.
- **Method**: Leave-one-out cross-validation is employed, where each question  $g$  in  $\mathcal{G}$  is left out once, the model is trained on the remainder of  $\mathcal{G}$ , and then the error  $\zeta_{\lambda,g}$  is calculated on the left-out question  $g$ .
- **Rationale**: This method helps to minimize overfitting and allows for a robust estimation of the model's performance as it relates to the hyperparameter tuning.

2. **Non-Gold Standard Questions ( $\mathcal{H}$ )**:

- **Purpose**: Used to assess the generalizability and performance of the model with the optimally tuned  $\hat{\lambda}$  derived from the gold standard questions.
- **Method**: After determining  $\hat{\lambda}$  that minimizes the error across  $\mathcal{G}$ , the model is applied to  $\mathcal{H}$  to compute the final error rate  $\zeta$ .
- **Rationale**: This provides a measure of how well the model, tuned to the gold standard data, performs on unseen or additional data that wasn't used in the hyperparameter tuning phase.

### Appropriateness of the Testing Data:

- **Using  $\mathcal{G}$  for Hyperparameter Tuning**:
  - This is appropriate as  $\mathcal{G}$  consists of questions with known true outcomes, allowing for a reliable assessment of model error and the effectiveness of different  $\lambda$  settings.
  - The use of cross-validation helps ensure that the model is not just memorizing the gold standard data but is actually learning generalizable patterns.
- **Using  $\mathcal{H}$  for Generalization Testing**:

- It is appropriate and crucial to test the model on  $\mathcal{H}$  because it examines the model's ability to generalize beyond the data used to tune the parameters. This is a standard practice in machine learning to validate models on a separate test set that was not used in training or validation.

- Testing on  $\mathcal{H}$  provides an insight into how the model performs in "real-world" scenarios where the exact outcomes might not be known in advance, simulating a more realistic application of the model.

### ### Considerations and Potential Issues:

- **Independence of  $\mathcal{G}$  and  $\mathcal{H}$ :**

- It's critical that  $\mathcal{G}$  and  $\mathcal{H}$  are independent. If there is overlap or if the questions in  $\mathcal{H}$  are not sufficiently diverse or representative of the broader data or task, the test results might not accurately reflect the model's effectiveness across different conditions.

- **Representativeness and Size:**

- The effectiveness of this testing approach also hinges on the representativeness and size of both  $\mathcal{G}$  and  $\mathcal{H}$ . If  $\mathcal{G}$  is too small or not representative of the overall question set, the  $\lambda$  tuning may not be optimal. Similarly, if  $\mathcal{H}$  is too small or not varied enough, it may not provide a reliable measure of generalizability.

- **Data Leakage:**

- Care must be taken to ensure that there is no data leakage between  $\mathcal{G}$  and  $\mathcal{H}$ , meaning that information from  $\mathcal{H}$  should not influence the model training phases including the selection of  $\lambda$ .

In summary, using gold standard questions  $\mathcal{G}$  for hyperparameter tuning via cross-validation and non-gold standard questions  $\mathcal{H}$  for testing the generalization capability of the model is a robust and appropriate methodological approach. This setup allows for rigorous model evaluation, assuming the data sets  $\mathcal{G}$  and  $\mathcal{H}$  are well-constructed and truly independent.

---

Are you saying that hyperparameter selection is being done on the test data?

No, I should clarify that properly. In the provided description of Algorithm 1, hyperparameter selection is done using **cross-validation on the gold standard questions ( $\mathcal{G}$ )**, not on the test data ( $\mathcal{H}$ ).

Here's how the process is structured:

1. **Cross-Validation on Gold Standard Data ( $\mathcal{G}$ ):**

- **Gold Standard Questions ( $\mathcal{G}$ ):** These are used for the tuning of the hyperparameter  $\lambda$  through cross-validation. Specifically, leave-one-out cross-validation is used within



$\mathcal{G}$ ). This involves training the model on  $\mathcal{G}$  minus one question, then validating it on the excluded question. This cycle is repeated such that each question in  $\mathcal{G}$  serves as the validation set once.

- **Purpose**: This step ensures that  $\lambda$  is chosen based on how well it performs across various subsets of the gold standard data, aiming to minimize overfitting and enhance the model's ability to generalize within the scope of  $\mathcal{G}$ .

2. **Testing on Non-Gold Standard Data ( $\mathcal{H}$ )**:

- **Non-Gold Standard Questions ( $\mathcal{H}$ )**: Once  $\lambda$  is selected using the gold standard questions, this chosen  $\lambda$  is then used to apply the model to the non-gold standard questions ( $\mathcal{H}$ ) to assess its generalization capabilities on new, unseen data.

- **Purpose**: This step evaluates how well the model, tuned to the gold standard, performs when applied to a separate set of questions that were not involved in the hyperparameter tuning process.

### Importance of Separation:

- **Preventing Data Leakage**: It's crucial that the data used for hyperparameter tuning ( $\mathcal{G}$ ) is distinct and separate from the data used for testing the model's generalization ( $\mathcal{H}$ ). Using  $\mathcal{G}$  for tuning and  $\mathcal{H}$  for testing ensures there is no leakage of information between these phases, which could lead to overly optimistic performance metrics.

- **Validating Model Performance**: Testing on  $\mathcal{H}$  after tuning on  $\mathcal{G}$  is a methodologically sound approach to understand how the model might perform in real-world scenarios or practical applications where the exact outcomes are unknown.

The approach taken by Algorithm 1, where  $\mathcal{G}$  is used for tuning and  $\mathcal{H}$  for testing, is indeed appropriate and follows best practices in machine learning and statistical modeling. This method helps ensure that the model is both effective (tuned well) and robust (performs well on unseen data).

---

Is Theorem 1 correct? Please check carefully.

To assess the correctness of Theorem 1 in the submitted paper, let's delve into the specific details provided about the theorem and examine the underlying logic and conditions outlined:

### Overview of Theorem 1

Theorem 1 addresses the identifiability of the parametric model introduced in the paper. Identifiability in a statistical model is crucial as it determines whether it's possible to uniquely

recover model parameters based on the probability distributions of the observed data. Here's a summary of the conditions and conclusions of Theorem 1:

- **Identifiability Condition**: For the model to be identifiable, it requires that a matrix constructed from the differences between the expected overall scores and individual scores, adjusted by level, must be of full rank (rank equals the number of level categories  $|Z|$ ).

#### Specifics of the Model

The model assumes that the expected value of the response variable  $y_i$  for each item  $i$  is a convex combination of the scores provided by evaluators, with weights related to their levels. The convex weights are defined as  $\frac{\beta_{z_{i,j}}}{\sum_j \beta_{z_{i,j}}}$  divided by the sum of  $\beta_{z_{i,j}}$  terms over all evaluators for item  $i$ , where  $z_{i,j}$  denotes the level of the  $j$ -th evaluator for item  $i$ .

#### Matrix $\Sigma$

The matrix  $\Sigma$  described in the theorem plays a central role in checking identifiability. Each element of  $\Sigma$ ,  $\Sigma_{i,\ell}$ , aggregates contributions from all evaluations for item  $i$  that share the same level  $\ell$ , specifically capturing the difference between the expected score  $\mu_i$  (a linear function of the scores and weights) and the individual scores at that level.

#### Validity of Theorem 1

- Matrix Construction**: The construction of  $\Sigma$  depends critically on the differences between the expected overall scores and the scores contributed by evaluations at each level. If this matrix captures all unique variations possible through different level assignments and their associated score contributions, it would influence the rank of the matrix.
- Rank Condition**: The claim that the matrix  $\Sigma$  must have rank  $|Z|$  for the model to be identifiable is logical. If the rank is less than  $|Z|$ , there could be dependencies among the columns of  $\Sigma$ , implying that not all levels contribute uniquely to the determination of  $y_i$ , leading to non-identifiability.
- Theoretical Justification**: The justification provided in the theorem that  $\beta \neq \beta' \Rightarrow P \neq P'$  (different parameter vectors lead to different data distributions) hinges on the linear independence of the vectors formed by the different level contributions in  $\Sigma$ .

#### Conclusion

Theorem 1 appears logically sound and correctly argued based on the explanation provided in the paper. The condition that the matrix  $\Sigma$  must be of full rank to ensure identifiability is a standard type of condition in statistical modeling to ensure that all parameters contribute independently to the outcomes.

If the matrix  $\Sigma$  and its rank are correctly computed as described, and the assumptions listed (like the minimal value  $\epsilon$  for each  $\beta$  to avoid division by zero) are met, then the theorem should hold true. The proof relies on showing that non-identical parameter vectors cannot produce identical distributions under the given model structure, which is a robust approach to demonstrating identifiability.

---

Is the only if condition in Theorem 1 correct? Please check carefully.

The "only if" condition in Theorem 1 of the submitted paper implies that identifiability is guaranteed when the rank of a matrix  $\Sigma$  constructed from the model's parameters and data achieves full rank (equal to the number of distinct levels  $|Z|$ ). This condition is meant to ensure that different sets of model parameters (in this case,  $\beta^*$ ) lead to distinct probability distributions of the observable data.

To critically evaluate whether this condition is sufficient and necessary for the model's identifiability, let's break down the components and reasoning:

### ### Understanding the Matrix $\Sigma$

The matrix  $\Sigma$  contains elements  $\Sigma_{i,\ell}$  which are defined as:

$$\Sigma_{i,\ell} = \sum_{j \in [d_i]} (\mu_i - x_{i,j}) I\{z_{i,j} = \ell\}$$

where  $I(\cdot)$  is the indicator function,  $z_{i,j}$  represents the level of the  $j$ -th evaluator for item  $i$ ,  $\mu_i$  is the expected score for item  $i$ , and  $x_{i,j}$  is the actual score given by the  $j$ -th evaluator.

### ### Rank Condition for Identifiability

For the model to be identifiable, it must be possible to uniquely recover the model parameters ( $\beta^*$ ) from the probability distribution of the observed data. The theorem posits that this is possible only if the matrix  $\Sigma$  has full rank, meaning it has  $|Z|$  linearly independent rows. Here's why this condition is critical:

- **Full Rank Implication**: If  $\Sigma$  has full rank, then every distinct set of parameters  $\beta^*$  would lead to a unique transformation expressed through  $\mu_i$ , thus distinguishing different parameter sets based on the observable outcomes.
- **Non-Identifiability if Rank is Deficient**: If the rank of  $\Sigma$  is less than  $|Z|$ , there exists at least one linear dependency among the level-specific contributions to the expected scores,  $\mu_i$ . This dependency implies that multiple parameter sets could result in the same  $\mu_i$  values, leading to the same probability distribution of observed data for different underlying parameters, thereby causing the model to be non-identifiable.

### ### Is the Condition Necessary and Sufficient?

- **Sufficiency**: If  $\Sigma$  is of full rank  $(|Z|)$ , the linear independence among the columns associated with different levels ensures that each level has a unique contribution that cannot be mimicked by any combination of others. This uniqueness is crucial for being able to back-calculate the exact parameters from the expected values, assuming the structure of the model and the link function are known and invertible.
- **Necessity**: If the rank condition is not met, the same set of observed data could theoretically be produced by different parameter vectors, making it impossible to uniquely determine which set of parameters is the true underlying cause. However, it should be noted that practical identifiability might still occur in specific data-rich scenarios or under additional constraints not covered by the theorem.

### ### Conclusion

The "only if" condition as presented is theoretically valid, given the assumptions and framework of the model described in the paper. It's a common type of condition in statistical models involving systems of linear equations where parameters must influence outcomes independently for identifiability. However, proving necessity rigorously would require demonstrating that no other conditions (besides full rank of  $\Sigma$ ) could possibly lead to identifiability, which typically involves complex theoretical arguments or counterexamples. The sufficiency of this condition is well-argued and follows logically from the linear algebraic properties of matrices and their role in linear transformations in statistical models.

---

Is Theorem 2 correct? Please check carefully.

To evaluate the correctness of Theorem 2, which claims the convexity of the optimization problems defined in the models (5) and (6), let's examine the specifics of these models and the underlying assumptions used to assert convexity.

### ### Theorem 2 Details:

- **Theorem 2** states that the optimization problems for both the non-parametric estimator in the general setting (model (5)) and in a transductive setting (model (6)) are convex.

### ### Analysis:

#### 1. **Objective Functions**:

- Both model (5) and model (6) have objective functions consisting of two terms: a least squares term and a regularization term. The least squares term  $\sum (y_i - \phi(x_i, 1z_i, 1, \dots, x_i, dzi, d))^2$  is a quadratic function of the parameters  $\phi$ , which is inherently convex. The regularization term  $\lambda(\|\phi\|_2^2 + \sum_{j=1}^d a_j^2)$  is also a quadratic function and hence convex.

- Both terms are added together with non-negative coefficients (since  $\lambda \geq 0$ ), preserving convexity in the combined objective function.

## 2. **Constraints**:

- In model (5), constraints ensure that the function value does not change with the permutation of inputs (assumption B1) and that the function is non-decreasing with respect to certain inputs (assumption B2). These constraints are linear with respect to the function parameters  $\phi$ , which does not violate the convexity of the optimization problem.

- Model (6) simplifies the problem by considering only the distinct sorted evaluations and maintains linear constraints similar to those in model (5). This reduction and the type of constraints used also support the convexity of the problem.

## 3. **Convex Set**:

- The decision variables  $(\phi)$  in both problems are defined over the entire real numbers (assuming  $(\phi \in \mathbb{R}^{(|X| \times |Z|)^d})$ ), and the space of these variables is a convex set.

Given these points, the claim in Theorem 2 about the convexity of the optimization problems appears to be substantiated correctly. The objective functions are convex, and the constraints are linear or involve conditions that maintain convexity. Moreover, the setting of the problem within a real vector space adds to the validity of the convexity claim.

## ### Conclusion:

The presentation and substantiation of Theorem 2 in the paper seem theoretically sound. The convexity claims are backed by appropriate convex functions and constraints. If you have more detailed aspects of the theorem or specific parts of its proof you'd like to discuss or are unclear about, feel free to point them out for further analysis.

---

Is (5) convex? Please check carefully.

To verify the convexity of the optimization problem defined in model (5) from the paper, let's dissect the formulation of this model and closely analyze each component to ensure the convexity claim holds true.

## ### Model (5) Details:

- **Objective Function**: The optimization problem in (5) aims to minimize a function defined as:

$$\sum_{i \in [n]} \left( y_i - \phi_{xi,1zi,1,...,xi,dzi,d} \right)^2 + \lambda \sum_{(a_1,b_1,...,a_d,b_d) \in (|X| \times |Z|)^d} \left( \phi_{a_1,b_1,...,a_d,b_d} - \frac{1}{d} \sum_{j=1}^d a_j \right)^2$$

This includes a sum of squares term and a regularization term weighted by  $\lambda$ , both of which are convex functions.

- **Constraints**:

1. **Permutation Invariance**: The function value  $\phi$  should remain invariant under any permutation of the evaluator arguments, implying the equality constraints:

$$\begin{aligned} & \phi_{a_1, b_1, \dots, a_{\ell_1}, b_{\ell_1}, \dots, a_{\ell_2}, b_{\ell_2}, \dots, a_d, b_d} = \\ & \phi_{a_1, b_1, \dots, a_{\ell_2}, b_{\ell_2}, \dots, a_{\ell_1}, b_{\ell_1}, \dots, a_d, b_d} \end{aligned}$$

These are linear constraints and do not affect the convexity of the objective function.

2. **Monotonicity**: The function  $\phi$  should be non-decreasing in each of its real-valued arguments  $a_{\ell}$ , keeping other arguments fixed:

$$\begin{aligned} & \phi_{a_1, b_1, \dots, a_{\ell}, b_{\ell}, \dots, a_d, b_d} \geq \phi_{a_1, b_1, \dots, a'_{\ell}, b_{\ell}, \dots, a_d, b_d} \\ & \quad \text{for all } a_{\ell} \geq a'_{\ell} \end{aligned}$$

This is a set of linear inequality constraints because each involves comparison between the values of  $\phi$  under different inputs, maintaining the convexity of the problem.

### Convexity Analysis:

1. **Convex Objective Function**: The terms in the objective function are quadratic in  $\phi$ , thus inherently convex. Adding convex functions results in a convex function.

2. **Convex Constraints**: Both sets of constraints are linear (equality and inequality), and linear constraints preserve the convexity of the optimization problem because they define a convex feasible region.

3. **Parameter Space**: The function  $\phi$  is defined over a real vector space, which is convex. All parameters are unbounded above and below, providing a convex domain.

Given these components, model (5) comprises a convex objective function subject to linear constraints, forming a convex optimization problem. Thus, based on the details provided in the paper and the general principles of convex optimization, the claim that model (5) presents a convex optimization problem appears correct.

If there are any specific mathematical derivations or aspects of the constraints or the formulation you're concerned about, we can delve deeper into those for further clarification.

---

Is (6) convex? Please check carefully.

To assess the convexity of the optimization problem defined in model (6) from the paper, let's thoroughly evaluate the components of this model. Model (6) represents a transductive setting adaptation of the non-parametric estimator that simplifies the problem by using a sorted evaluation order.

### Model (6) Details:

- \*\*Objective Function\*\*:

$$\sum_{i \in [n]} (y_i - \phi_{\text{sorted}}(x_{i,1}, z_{i,1}, \dots, x_{i,d}, z_{i,d}))^2 + \lambda \sum_{(a,b) \in W} (\phi_{a,b} - \frac{1}{d} \sum_{j=1}^d a_j)^2$$

This function includes a least squares term and a regularization term, both of which are convex functions.

- \*\*Constraints\*\*:

- The constraints ensure that  $\phi$  is non-decreasing in each  $a_{\ell}$  while keeping other arguments fixed. This is represented by linear inequalities:

$$\phi_{a_1, b_1, \dots, a_{\ell}, b_{\ell}, \dots, a_d, b_d} \geq \phi_{a_1, b_1, \dots, a'_{\ell}, b_{\ell}, \dots, a_d, b_d} \text{ for all } a_{\ell} \geq a'_{\ell}$$

These are linear constraints that do not affect the convexity of the optimization problem.

### Convexity Analysis:

1. \*\*Convex Objective Function\*\*:

- The terms in the objective function are quadratic in  $\phi$ , making them inherently convex. Since convex functions remain convex when added together, the entire objective function is convex.

2. \*\*Convex Constraints\*\*:

- The constraints are linear inequalities. Linear constraints preserve convexity because they do not introduce any curvature; they simply define a convex feasible region.

3. \*\*Parameter Space\*\*:

- The function  $\phi$  is defined over real numbers for each configuration of  $(a_1, b_1, \dots, a_d, b_d)$ . This parameter space is a convex set.

Given these points, the optimization problem in model (6) is correctly formulated as a convex problem. It consists of a convex objective function subjected to linear constraints, all defined

within a convex parameter space. Thus, the claim in the paper that model (6) presents a convex optimization problem appears to be correct and well-substantiated.

If you have any specific aspects of the proof or the mathematical formulation that need deeper exploration or clarification, feel free to ask!

---

## Responses by GPT-4o (May 14, 2024)

---

### Treatment condition: Finding errors in erroneous parts

#### Optimization error

##### Direct prompt

To check the validity and correctness of the claims made in the provided paper, I will evaluate each claim step by step, referencing the content of the paper and any related literature if necessary.

##### ### Abstract Claims

1. \*\*\*"Can some known ground truths in crowdsourced labeling enhance aggregation for other items?"\*\*

- The paper explores leveraging 'gold standard' questions in crowdsourced labeling tasks to improve the aggregation accuracy of other items. This claim is supported by literature indicating the utility of gold standard questions in improving label accuracy【3†source】.

2. \*\*\*"Can we discern how meta-reviewers in peer review aggregate individual assessments to make their decisions?"\*\*

- The paper presents an exploration of meta-reviewers' aggregation methods in peer review, particularly focusing on understanding and modeling these methods quantitatively【3†source】.

##### ### Introduction Claims

1. \*\*\*"These methods are used in various domains such as admissions or hiring processes."\*\*

- Aggregation methods are indeed widely applicable across different domains such as hiring and admissions, which often involve aggregating evaluations from multiple evaluators【3†source】.



2. \*\*\*"Using gold standard questions can filter out low-quality answers or spamming."\*\*
  - Existing literature supports the use of gold standard questions to filter out low-quality contributions in crowdsourced tasks【3†source】.

#### ### Methodology Claims

1. \*\*\*"We provide theoretical results on identifiability and convexity."\*\*
  - The paper presents a theoretical framework proving the identifiability of the proposed statistical model and the convexity of the estimators used, which are crucial for the validity of the aggregation methods proposed【3†source】.

#### ### Experimental Results Claims

1. \*\*\*"Our empirical evaluations using data from crowdsourcing (Mechanical Turk) and peer review (ICLR conference) reveal significant advantages of our methods."\*\*
  - The paper conducts extensive experiments demonstrating the efficacy of their methods compared to existing techniques, showing statistically significant improvements in accuracy for both crowdsourced labeling and peer review aggregation【3†source】.

#### ### Specific Model Claims

1. \*\*\*"Convex combination generalized linear model provides better performance in aggregation."\*\*
  - The paper's empirical results show that the proposed convex combination GLM outperforms conventional methods in aggregating evaluations, particularly in the crowdsourced labeling experiments【3†source】.
2. \*\*\*"Our non-parametric estimator improves performance when lambda is chosen in a data-dependent manner."\*\*
  - The results indicate that the non-parametric estimator with data-dependent lambda selection (as per Algorithm 1) yields superior performance, particularly in reducing error rates across various tasks【3†source】.

#### ### Related Literature Claims

1. \*\*\*"Our work extends models by incorporating gold standard questions and confidence levels."\*\*
  - The paper builds upon existing models by incorporating both gold standard questions and evaluator confidence levels, which are shown to enhance the accuracy of aggregated labels and decisions【3†source】.

#### ### Conclusion Claims

1. \*\*\*"Our methods show efficacy in crowdsourced labeling tasks and offer a logical next step for evaluation within RLHF frameworks for large language models."\*\*
  - The conclusion is supported by the positive experimental outcomes, suggesting that their methods could be effectively applied in reinforcement learning with human feedback (RLHF) settings for large language models【3†source】.

Based on the detailed analysis of the paper and the literature references provided, the claims made in the paper appear to be valid and correct. The theoretical foundations, empirical evidence, and connections to existing work support the robustness and applicability of the proposed methods. If further specific validations or verifications are needed, such as empirical replication studies or additional theoretical proofs, these could provide even stronger confirmation of the claims.

---

Is Theorem 2 correct? Please check carefully.

To verify the correctness of Theorem 2 regarding the convexity of the optimization problems, we need to check the claims made and the provided proof.

#### ### Theorem 2 Claims

Theorem 2 states that the three optimization problems defined in (4), (5), and (6) are all convex.

##### 1. \*\*Optimization Problem (4):\*\*

- The objective function is a least squares estimator which is quadratic and thus convex.
- The constraints include linear inequalities (e.g.,  $\|\beta\| \geq \epsilon$ ) and linear equality constraints (e.g.,  $\sum \beta_i = 1$ ).
- Monotonicity constraints (if present) are also linear.

##### 2. \*\*Optimization Problem (5):\*\*

- The objective function has two terms, both quadratic, which together form a convex function.
- The constraints are linear, addressing assumptions (B1) and (B2).

##### 3. \*\*Optimization Problem (6):\*\*

- The structure is similar to (5) but in a transductive setting, focusing on specific points rather than a general space.
- The objective function and constraints remain quadratic and linear, respectively.

#### ### Proof Structure

##### 1. \*\*Objective Function Convexity:\*\*

- Quadratic functions  $\|y - f(x)\|^2$  are convex because their second derivative is non-negative.
- Adding a quadratic regularization term  $\lambda \sum (...)^2$  retains convexity.

##### 2. \*\*Constraints:\*\*

- Linear equality and inequality constraints preserve convexity.
- Permutation invariance constraints are linear rearrangements, maintaining linearity.
- Monotonicity constraints are linear as they enforce order without introducing non-convex terms.

### ### Detailed Verification

#### 1. \*\*Convex Combination GLM Estimator (4):\*\*

- The function being minimized is quadratic in  $(\beta)$ :

$$\sum_{i=1}^n \left( y_i - \frac{x_i^T (\beta_{z_{i,1}}, \beta_{z_{i,2}}, \dots, \beta_{z_{i,d_i}})^T}{\sum_{j=1}^{d_i} \beta_{z_{i,j}}} \right)^2$$

- Subject to:

$$\epsilon \leq \beta_{\ell} \leq 1, \quad \sum_{\ell \in |Z|} \beta_{\ell} = 1, \quad \beta_1 \leq \beta_2 \leq \dots \leq \beta_{|Z|}$$

- These constraints and the quadratic objective function ensure convexity.

#### 2. \*\*Non-parametric Estimator (5):\*\*

- Objective function:

$$\sum_{i \in [n]} \left( y_i - \phi_{x_{i,1}, z_{i,1}, \dots, x_{i,d}, z_{i,d}} \right)^2 + \lambda \sum_{(a_1, b_1, \dots, a_d, b_d) \in (|X| \times |Z|)^d} \left( \phi_{a_1, b_1, \dots, a_d, b_d} - \frac{1}{d} \sum_{j \in [d]} a_j \right)^2$$

- Constraints:

$$\begin{aligned} \phi_{a_1, b_1, \dots, a_{\ell}, b_{\ell}, \dots, a_d, b_d} &= \phi_{a_1, b_1, \dots, a_{\ell'}, b_{\ell'}, \dots, a_d, b_d} \\ &\quad \text{(linear permutation invariance)} \\ \phi_{a_1, b_1, \dots, a_{\ell}, b_{\ell}, \dots, a_d, b_d} &\geq \phi_{a_1, b_1, \dots, a'_{\ell}, b_{\ell}, \dots, a_d, b_d} \\ &\quad \text{(linear monotonicity)} \end{aligned}$$

- The combination of a convex objective with linear constraints ensures convexity.

#### 3. \*\*Transductive Non-parametric Estimator (6):\*\*

- Similar to (5) but with a reduced set of variables and constraints specific to  $(W)$ .
- The constraints and objective remain quadratic and linear, ensuring convexity.

### ### Conclusion

The proof and methodology provided in the paper correctly establish the convexity of the optimization problems. The quadratic nature of the objective functions combined with the linear constraints ensures that all three problems (4), (5), and (6) are indeed convex optimization problems. Therefore, Theorem 2 is correct.

---

Is (4) convex? Please check carefully.

To determine whether the optimization problem (4) is convex, we need to carefully analyze both the objective function and the constraints.

#### ### Problem (4) Formulation

The optimization problem (4) is given by:

$$\begin{aligned} & \underset{\beta \in \mathbb{R}^{|Z|}}{\text{minimize}} \quad \sum_{i=1}^n \left( y_i - \frac{x_i^T (\beta_{z_{i,1}}, \beta_{z_{i,2}}, \dots, \beta_{z_{i,d_i}})^T}{\sum_{j=1}^{d_i} \beta_{z_{i,j}}} \right)^2 \\ & \text{subject to} \quad \epsilon \leq \beta_{\ell} \leq 1, \quad \forall \ell \in [Z], \\ & \quad \sum_{\ell=1}^{|Z|} \beta_{\ell} = 1, \\ & \quad \beta_1 \leq \beta_2 \leq \dots \leq \beta_{|Z|}. \end{aligned}$$

#### ### Analysis of the Objective Function

The objective function to be minimized is:

$$f(\beta) = \sum_{i=1}^n \left( y_i - \frac{x_i^T (\beta_{z_{i,1}}, \beta_{z_{i,2}}, \dots, \beta_{z_{i,d_i}})^T}{\sum_{j=1}^{d_i} \beta_{z_{i,j}}} \right)^2$$

Let's denote:

$$\beta_{z_i} = (\beta_{z_{i,1}}, \beta_{z_{i,2}}, \dots, \beta_{z_{i,d_i}})^T \quad \text{and} \quad S_i(\beta) = \sum_{j=1}^{d_i} \beta_{z_{i,j}}$$

Then, the objective function can be rewritten as:

$$f(\beta) = \sum_{i=1}^n \left( y_i - \frac{x_i^T \beta_{z_i}}{S_i(\beta)} \right)^2$$

#### #### Convexity of the Objective Function

To check if  $f(\beta)$  is convex, we need to examine the second-order conditions. Specifically, we need to determine if the Hessian matrix of  $f(\beta)$  is positive semidefinite.

Consider the term inside the sum for a fixed  $i$ :

$$\phi_i(\beta) = \left( y_i - \frac{x_i^T \beta_{z_i}}{S_i(\beta)} \right)^2$$

We can express:

$$g_i(\beta) = \frac{x_i^T \beta_{z_i}}{S_i(\beta)}$$

and rewrite  $\phi_i(\beta)$  as:

$$\phi_i(\beta) = (y_i - g_i(\beta))^2$$

The function  $g_i(\beta)$  involves a ratio of linear functions of  $\beta$ . Such a ratio generally does not yield a convex function. To see this, we calculate the gradient and Hessian:

$$\nabla g_i(\beta) = \frac{S_i(\beta) x_i - (x_i^T \beta_{z_i}) \nabla S_i(\beta)}{(S_i(\beta))^2}$$

where  $\nabla S_i(\beta)$  is a vector of ones for the components corresponding to  $z_{\{i\}}$ .

The Hessian  $H(g_i)$  involves terms with  $S_i(\beta)$  in the denominator, leading to non-linear terms that complicate convexity analysis. Generally, for  $g_i(\beta)$  to be convex, its second derivative should be non-negative, but here the ratio form introduces complexities that typically do not ensure positive semidefiniteness.

### ### Constraints Analysis

The constraints are linear or affine:

- $-\epsilon \leq \beta_{\ell} \leq 1$
- $\sum_{\ell=1}^{|Z|} \beta_{\ell} = 1$
- $\beta_1 \leq \beta_2 \leq \dots \leq \beta_{|Z|}$

These constraints define a convex feasible region since they describe a convex polytope.

### ### Conclusion

Even though the constraints define a convex set, the objective function  $f(\beta)$  is not necessarily convex due to the ratio  $\frac{x_i^T \beta_{z_i}}{S_i(\beta)}$  in the squared term. The non-linear nature of this ratio typically prevents the Hessian from being positive semidefinite, and thus  $f(\beta)$  is not guaranteed to be convex.

Therefore, **problem (4) is not convex**.

---

---

## Statistical error

### Direct prompt

To assess the validity and correctness of the claims made in the paper titled "On Aggregating Evaluations with Expertise or Confidence Information," I will check the key claims step by step based on the information provided in the paper. Here are the claims I'll evaluate:

1. **Formulation of problems within a unified modeling framework.**
2. **Proposal of parametric and non-parametric estimators.**
3. **Proof of theoretical results on identifiability and convexity.**
4. **Empirical evaluation revealing significant advantages of the methods.**

### Step-by-Step Evaluation:

#### Claim 1: Formulation of problems within a unified modeling framework

- **Statement:** The paper claims to have identified and formulated the problems of aggregating evaluations with expertise or confidence information within a unified modeling framework.
- **Evaluation:**
  - The introduction clearly outlines the problem statement and the contexts of crowdsourced labeling and peer review.
  - The paper describes two models: the convex combination generalized linear model and the non-parametric model.
- **Conclusion:** This claim appears valid based on the detailed problem formulation and the mathematical models provided.

#### Claim 2: Proposal of parametric and non-parametric estimators

- **Statement:** The paper proposes both parametric and non-parametric estimators for aggregating evaluations.
- **Evaluation:**
  - The convex combination generalized linear model (parametric) is introduced with a least squares estimator.
  - The non-parametric model is presented along with a detailed estimator.
  - The estimators are clearly defined with corresponding optimization problems.
- **Conclusion:** This claim is supported by the provided models and detailed estimation procedures.

#### #### Claim 3: Proof of theoretical results on identifiability and convexity

- **Statement:** The paper claims to have proved theoretical results regarding the identifiability of their statistical model and the convexity of their proposed estimators.
- **Evaluation:**
  - **Identifiability:** Theorem 1 provides necessary and sufficient conditions for identifiability in their convex combination GLM model.
    - The proof is based on the rank condition of a specific matrix constructed from the data.
    - The conditions for identifiability are logically derived and appear sound.
  - **Convexity:** Theorem 2 asserts the convexity of the optimization problems for the parametric and non-parametric estimators.
    - The proof uses standard convexity arguments and demonstrates that the objective functions are convex and constraints are linear.
- **Conclusion:** The proofs of identifiability and convexity seem thorough and well-reasoned, supporting this claim.

#### #### Claim 4: Empirical evaluation revealing significant advantages of the methods

- **Statement:** The paper claims that empirical evaluations using crowdsourcing and peer review data reveal significant advantages of their methods.
- **Evaluation:**
  - **Crowdsourced Labeling:** The experiments conducted on multiple-choice questions from the Mechanical Turk dataset show improved accuracy over previous methods.
    - The results are presented in Table 1, showing error rates across different tasks.
  - **Peer Review:** The study on ICLR 2022 data provides insights into how meta-reviewers aggregate scores with varying confidences.
    - The results indicate a significant impact of higher confidence levels on final decisions.
- **Conclusion:** The empirical results are clearly presented and indicate the claimed improvements, supporting this claim.

#### ### Overall Conclusion

Based on the detailed evaluation of the claims made in the paper, the following conclusions can be drawn:

1. **Formulation of problems within a unified modeling framework.** **Valid**
2. **Proposal of parametric and non-parametric estimators.** **Valid**
3. **Proof of theoretical results on identifiability and convexity.** **Valid**
4. **Empirical evaluation revealing significant advantages of the methods.** **Valid**

No false claims have been identified based on the provided information in the paper. The claims are well-supported by theoretical and empirical evidence.

---

Is Theorem 1 correct? Please check carefully.

To carefully check the correctness of Theorem 1 in the paper, I will break down the proof provided in the paper step by step to verify the logical consistency and mathematical validity.

#### #### Theorem 1

**Statement:** The model (1) with the linear link (i.e.,  $\sigma$  as the identity function) is identifiable if and only if the matrix  $[\Sigma \mid 1^T]$  is of rank  $|Z|$ .

#### #### Proof Analysis

##### ##### Definitions and Assumptions:

1. **Model (1) with linear link:**

$$E[y_i] = x_i^T \left[ \frac{\beta_{z_{i,1}}}{\sum_{j \in [d_i]} \beta_{z_{i,j}}}, \frac{\beta_{z_{i,2}}}{\sum_{j \in [d_i]} \beta_{z_{i,j}}}, \dots, \frac{\beta_{z_{i,d_i}}}{\sum_{j \in [d_i]} \beta_{z_{i,j}}} \right]^T$$

where  $(\beta^* \in [0,1]^{|Z|})$  and  $(\sum_{\ell \in [Z]} \beta^*_\ell = 1)$ .

2. **Matrix  $\Sigma$ :**

$$\Sigma_{i,\ell} = \sum_{j \in [d_i]} (\mu_i - x_{i,j}) \mathbb{1}_{\{z_{i,j} = \ell\}}$$

where  $(\mu_i = E[y_i])$ .

##### ##### Proof Structure:

The proof provided in the paper consists of two parts:

1. **Necessary Condition:**
  - It states that if  $[\Sigma \mid 1^T]$  is not of rank  $|Z|$ , then the model is not identifiable.
2. **Sufficient Condition:**
  - It argues that if  $[\Sigma \mid 1^T]$  is of rank  $|Z|$ , then the model is identifiable.

##### ##### Detailed Breakdown:



#### ##### Necessary Condition:

1. **Statement:** If the rank of  $([\Sigma \mid 1^T])$  is less than  $(|Z|)$ , the model is not identifiable.

2. **Argument:**

-  $(\mu_i = x_i^T \left[ \frac{\beta_{z_{i,1}}}{\sum_{j \in [d_i]} \beta_{z_{i,j}}}, \frac{\beta_{z_{i,2}}}{\sum_{j \in [d_i]} \beta_{z_{i,j}}}, \dots, \frac{\beta_{z_{i,d_i}}}{\sum_{j \in [d_i]} \beta_{z_{i,j}}} \right]^T)$ .

- Rearranging the terms:

$$\sum_{j \in [d_i]} (\mu_i - x_{i,j}) \beta_{z_{i,j}} = 0$$

- For all items  $(i \in [n])$ :

$$\sum \beta^* = 0$$

- Along with the constraint  $(\sum_{\ell \in [|Z|]} \beta_{\ell}^* = 1)$ , we need to solve  $(\sum \beta^* = 0)$  and  $(1^T \beta^* = 1)$ .

- If  $([\Sigma \mid 1^T])$  is not of rank  $(|Z|)$ , there exist multiple solutions for  $(\beta^*)$ , meaning  $(\beta^*)$  is not uniquely determined. Thus, the model is not identifiable.

#### ##### Sufficient Condition:

1. **Statement:** If the rank of  $([\Sigma \mid 1^T])$  is  $(|Z|)$ , then the model is identifiable.

2. **Argument:**

-  $(\sum \beta^* = 0)$  and  $(1^T \beta^* = 1)$ .

- Since  $([\Sigma \mid 1^T])$  is of rank  $(|Z|)$ , these equations provide a unique solution for  $(\beta^*)$ .

- Thus,  $(\beta^*)$  is uniquely determined, ensuring the model is identifiable.

#### ### Verification of Key Steps:

1. **Constructing  $(\Sigma)$ :**

- Ensure that  $(\sum_{j \in [d_i]} (\mu_i - x_{i,j}) \beta_{z_{i,j}} = 0)$  correctly captures the information needed for identifiability.

2. **Linear Dependence:**

- Verify that the system  $(\sum \beta^* = 0)$  and  $(1^T \beta^* = 1)$  is appropriately set up for testing rank conditions.

3. **Rank Condition:**

- Confirm the rank condition correctly implies uniqueness or non-uniqueness of the solution for  $(\beta^*)$ .

#### ### Reassessment of Theorem Proof:

The key step is to ensure that  $(\Sigma \beta^* = 0)$  along with  $(1^T \beta^* = 1)$  fully captures the relationship between the observed scores and the latent parameters. The rank condition  $(\Sigma \mid 1^T)$  being  $(|Z|)$  indeed ensures the system has a unique solution.

### Conclusion:

Based on the detailed breakdown and the reassessment of the logical steps, the proof of Theorem 1 appears correct and logically consistent. Therefore, the theorem itself seems correct as well.

Is the only if condition in Theorem 1 correct? Please check carefully.

To determine the correctness of the "only if" condition in Theorem 1, we must closely inspect the logical reasoning and the mathematical derivations to ensure that the given condition is both necessary and sufficient for the model's identifiability.

### Re-evaluating the Only If Condition

The "only if" condition asserts that if the matrix  $(\Sigma \mid 1^T)$  is not of rank  $(|Z|)$ , then the model is not identifiable. We need to verify whether this condition is indeed necessary for identifiability.

#### Definitions and Restatement

- \*\*Model (1) with linear link:\*\*

$$E[y_i] = x_i^T \left[ \frac{\beta^*_{z_{i,1}}}{\sum_{j \in [d_i]} \beta^*_{z_{i,j}}}, \frac{\beta^*_{z_{i,2}}}{\sum_{j \in [d_i]} \beta^*_{z_{i,j}}}, \dots, \frac{\beta^*_{z_{i,d_i}}}{\sum_{j \in [d_i]} \beta^*_{z_{i,j}}} \right]^T$$

where  $(\beta^* \in [0,1]^{|Z|})$  and  $(\sum_{\ell \in [Z]} \beta^*_{\ell} = 1)$ .

- \*\*Matrix  $\Sigma$ :

$$\Sigma_{i,\ell} = \sum_{j \in [d_i]} (\mu_i - x_{i,j}) \mathbb{I}\{z_{i,j} = \ell\}$$

where  $(\mu_i = E[y_i])$ .

#### Proof of the Necessary Condition

Let's re-examine the argument to confirm if it correctly demonstrates that  $(\Sigma \mid 1^T)$  not having full rank implies non-identifiability.

### 1. \*\*Setup the Equations:\*\*

- Given the model, the expected value  $\mu_i$  for each item  $i$  is:

$$\mu_i = x_i^T \left[ \frac{\beta_{z_{i,1}}}{\sum_{j \in [d_i]} \beta_{z_{i,j}}}, \frac{\beta_{z_{i,2}}}{\sum_{j \in [d_i]} \beta_{z_{i,j}}}, \dots, \frac{\beta_{z_{i,d_i}}}{\sum_{j \in [d_i]} \beta_{z_{i,j}}} \right]^T$$

- Rearranging the terms, we obtain:

$$\sum_{j \in [d_i]} (\mu_i - x_{i,j}) \beta_{z_{i,j}} = 0$$

### 2. \*\*Matrix Representation:\*\*

- Summarizing for all items  $i \in [n]$ :

$$\sum \beta^* = 0$$

- Additionally, we have the normalization constraint:

$$1^T \beta^* = 1$$

### 3. \*\*Rank Condition:\*\*

- We need to solve  $(\sum \beta^* = 0)$  and  $(1^T \beta^* = 1)$ .
- If the rank of  $([\sum \mid 1^T])$  is less than  $(|Z|)$ , then these equations do not provide a unique solution for  $(\beta^*)$ . Multiple solutions imply non-identifiability of the parameter vector  $(\beta^*)$ .

### 4. \*\*Argument Verification:\*\*

- Suppose  $([\sum \mid 1^T])$  is not of rank  $(|Z|)$ . This implies that there exists a non-trivial null space, meaning there are multiple vectors  $(\beta^*)$  that satisfy  $(\sum \beta^* = 0)$  and  $(1^T \beta^* = 1)$ .
- As a result, the parameter  $(\beta^*)$  cannot be uniquely determined from the given data, indicating that the model is not identifiable.

### ### Logical Consistency

The argument relies on the linear independence of the constraints  $(\sum \beta^* = 0)$  and  $(1^T \beta^* = 1)$ . If these constraints do not span a space of dimension  $(|Z|)$ , then the parameter  $(\beta^*)$  cannot be uniquely identified.

### ### Conclusion

After re-examining the proof steps and ensuring logical consistency, the only if condition (necessary condition) in Theorem 1 is correct. The reasoning clearly shows that if the matrix  $([\sum \mid 1^T])$  is not of rank  $(|Z|)$ , the model cannot be identifiable due to the existence of multiple

solutions for the parameter vector  $\beta^*$ . Thus, the condition is necessary for the identifiability of the model.

---

---

## Evaluation error

### Direct prompt

To verify the validity and correctness of the claims made in the paper, let's systematically review and analyze the key assertions and results presented. We will focus on the main claims related to their theoretical contributions and empirical evaluations.

#### ### Claims to Validate

1. **Formulation of Problems in a Unified Framework**:
  - The authors claim they formulate the problem of aggregating evaluations using expertise or confidence information within a unified modeling framework.
2. **Proposed Estimators**:
  - They propose both parametric and non-parametric estimators to address the problem.
3. **Theoretical Results**:
  - They provide theoretical results on identifiability and convexity for their proposed estimators.
4. **Empirical Evaluations**:
  - They perform empirical evaluations using data from crowdsourcing (Mechanical Turk) and peer review (ICLR conference) and report significant advantages of their methods.

Let's examine these claims step-by-step based on the information provided in the paper.

#### ### 1. Formulation of Problems in a Unified Framework

The paper describes a unified modeling framework for the problems of aggregating evaluations with expertise or confidence information in crowdsourced labeling and peer review settings. They focus on leveraging known ground truths in crowdsourced labeling and understanding meta-reviewers' aggregation processes in peer review.

- **Verification**:

- The introduction and problem formulation sections outline the general notation and specific models (convex combination generalized linear model and a non-parametric model) used for aggregating evaluations[3:1†source][3:3†source][3:13†source].

## ### 2. Proposed Estimators

The authors propose parametric (convex combination GLM with a linear link) and non-parametric estimators for the aggregation tasks. They provide detailed descriptions and optimization formulations for both estimators[3:4†source][3:16†source].

- **Verification**:

- The parametric estimator is described by equation (4), while the non-parametric estimator is described by equation (5) with additional constraints outlined for both models[3:16†source].

## ### 3. Theoretical Results

The authors claim to provide theoretical results on the identifiability of their models and the convexity of their estimators.

- **Identifiability**:

- They provide Theorem 1, which states that their model is identifiable if the matrix formed by their specific construction is of full rank. This is a sufficient condition for identifiability[3:0†source][3:6†source][3:7†source].

- **Convexity**:

- Theorem 2 states that the optimization problems for their proposed estimators are convex. They provide a proof outline demonstrating that the objective functions are quadratic and the constraints are linear, ensuring convexity[3:7†source][3:18†source].

## ### 4. Empirical Evaluations

The authors conducted empirical evaluations using crowdsourcing data from Mechanical Turk and peer review data from the ICLR conference. They report error rates and comparative performance of their methods against established baselines.

- **Crowdsourced Labeling**:

- They present detailed experiments on five tasks, demonstrating that their proposed methods outperform other methods in terms of error rates on non-gold standard questions. This is detailed in their results table[3:9†source].

- **Peer Review**:

- They use data from the ICLR 2022 conference to analyze how meta-reviewers aggregate reviewer scores with different confidence levels. They present an analysis showing that their

model can provide insights into the aggregation process with a good fit ( $R^2 = 0.768$ )【3:5†source】【3:10†source】.

### ### Conclusion

Based on the detailed descriptions, theoretical proofs, and empirical results provided in the paper, the claims made by the authors are well-supported. The formulation of the problems, the proposed estimators, the theoretical results, and the empirical evaluations are all presented with sufficient detail and rigor to validate their correctness and validity. There are no indications of false claims based on the provided content.

If you need a deeper analysis of specific sections or further verification of specific equations or results, please let me know!

---

Is Algorithm 1 correct? Please check carefully.

To verify the correctness of Algorithm 1 as presented in the paper, we'll carefully examine each step of the algorithm and check for logical consistency, completeness, and alignment with the described methods.

Here is the detailed breakdown of Algorithm 1 from the provided text:

### ### Algorithm 1: Implementation and Evaluation of (6)

1. **Initialize Sets**:

``plaintext

Let  $G$  denote the set of all gold standard questions and  $H$  denote the set of all non-gold standard questions.

``

- **Correctness**: This initialization is clear and logical. The algorithm requires two sets: one for gold standard questions and one for non-gold standard questions.

2. **Iterate Over Lambda Values**:

``plaintext

for  $\lambda \in \{0, 2^{-16}, 2^{-15}, \dots, 2^{15}, 2^{16}\}$  do:

``

- **Correctness**: The range of  $\lambda$  values is comprehensive and covers a wide range, which is necessary to find an optimal value.

3. **Solve Optimization Problem**:

``plaintext

Solve (6) using  $G$  and denote the result as  $\hat{\phi}_\lambda$ .

...

- **\*\*Correctness\*\***: This step indicates solving the optimization problem defined in equation (6) for the given  $\lambda$  value using the gold standard questions. The notation  $\hat{\phi}_\lambda$  is used to store the solution.

4. **\*\*Compute Error for Each Non-Gold Standard Question\*\***:

```plaintext

for each  $h \in H$  do

    Compute  $\zeta_{\lambda,h} = I\{\hat{\phi}_\lambda(x_{h,1}, z_{h,1}, x_{h,2}, z_{h,2}, x_{h,3}, z_{h,3}) \neq y_h\}$

end for

...

- **\*\*Correctness\*\***: This loop calculates the error  $\zeta_{\lambda,h}$  for each non-gold standard question using the indicator function  $I(\cdot)$  which checks if the predicted label is different from the actual label. The usage of  $\hat{\phi}_\lambda$  is consistent with the previous step and aligns with evaluating the non-gold standard questions.

5. **\*\*Compute Overall Error Rate\*\***:

```plaintext

end for

//Computing error rate

Let  $\zeta = \min_{\lambda \in \{0, 2^{-16}, 2^{-15}, \dots, 2^{-1}, 2^0\}} \text{Mean}\{\zeta_{\lambda,h} | h \in H\}$

...

- **\*\*Correctness\*\***: The algorithm computes the mean error for each  $\lambda$  and then selects the  $\lambda$  that minimizes this mean error. This step ensures that the chosen  $\lambda$  is optimal with respect to the error rate on non-gold standard questions.

### Summary of Steps:

- **\*\*Initialization\*\***: Correctly initializes the required sets.
- **\*\*Lambda Iteration\*\***: Iterates over a wide range of  $\lambda$  values to ensure thorough evaluation.
- **\*\*Optimization Solving\*\***: Correctly solves the optimization problem for each  $\lambda$ .
- **\*\*Error Computation\*\***: Accurately computes the error for each non-gold standard question.
- **\*\*Error Rate Selection\*\***: Selects the optimal  $\lambda$  based on the mean error, ensuring the best performance.

### Conclusion

Algorithm 1 appears to be logically consistent and correctly structured. Each step aligns with the theoretical foundations laid out in the paper and follows a clear, methodical approach to determine the optimal  $\lambda$  for the non-parametric estimator.

Therefore, **Algorithm 1 is correct** as per the provided description and the steps outlined in the paper. If there are any specific points you need further clarification on or any additional details you want to verify, please let me know!

---

What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.

To determine the appropriateness of the data used for testing (evaluation) in Algorithm 1, let's carefully review the relevant sections of the paper, especially focusing on the experimental setup described.

### ### Experimental Setup and Algorithm 1

**From the Methods Section:**

- The dataset encompasses nine tasks, with five focused on multiple-choice questions, which are of interest here: 'bridge,' 'dogs,' 'countries,' 'flag,' and 'texture.'
- For each task, they randomly select 15 questions as the gold standard.
- The remaining questions are used as non-gold standard questions.
- For each gold standard question, workers are partitioned into groups for evaluation purposes.
- For each non-gold standard question, evaluations from 3 random workers are used.

**Algorithm 1 Details:**

``plaintext

Algorithm 1: Implementation and evaluation of (6)

1: Let  $G$  denote the set of all gold standard questions and  $H$  denote the set of all non-gold standard questions

2: for  $\lambda \in \{0, 2^{-16}, 2^{-15}, \dots, 2^{-1}, 2^0\}$  do:

3: Solve (6) using  $G$  and denote the result as  $\hat{\phi}_\lambda$

4: for each  $h \in H$  do

5: Compute  $\zeta_{\lambda,h} = I\{\hat{\phi}_\lambda(x_{h,1}, z_{h,1}, x_{h,2}, z_{h,2}, x_{h,3}, z_{h,3}) \neq y_h\}$

6: end for

7: end for

//Computing error rate

8: Let  $\zeta = \min_{\lambda \in \{0, 2^{-16}, 2^{-15}, \dots, 2^{-1}, 2^0\}} \text{Mean}\{\zeta_{\lambda,h} | h \in H\}$

...

### ### Analysis of Testing Data



**\*\*Gold Standard Questions (G)\*\*:**

- These are used for training the estimator  $\hat{\phi}_\lambda$ .
- Each gold standard question has known ground truth labels, allowing the model to learn the mapping from evaluations (with associated confidence levels) to the true labels.

**\*\*Non-Gold Standard Questions (H)\*\*:**

- These are used for evaluating the performance of the estimator  $\hat{\phi}_\lambda$ .
- The non-gold standard questions do not have known ground truth labels available during the training phase.
- For evaluation, the algorithm computes the error  $\zeta_{\lambda,h}$  by comparing the predicted labels for non-gold standard questions with their actual labels.

### ### Appropriateness of the Data

1. **\*\*Training Data (Gold Standard)\*\*:**

- The use of gold standard questions for training is appropriate as it allows the model to learn from questions with known true answers, improving the accuracy of the estimator.

2. **\*\*Testing Data (Non-Gold Standard)\*\*:**

- The use of non-gold standard questions for evaluation is appropriate because it tests the generalization capability of the trained model.
- The non-gold standard questions serve as unseen data points, ensuring that the evaluation measures how well the model performs on data it has not been explicitly trained on.

3. **\*\*Balance and Representation\*\*:**

- By selecting a significant portion (15 questions) as gold standard and using the rest as non-gold standard, the experiment ensures a good balance between training and testing data.
- The random selection of workers' evaluations for non-gold standard questions helps in providing a realistic evaluation scenario.

### ### Conclusion

The data used for testing in Algorithm 1 consists of non-gold standard questions, which is appropriate for evaluating the model's performance. This approach ensures that the evaluation reflects the model's ability to generalize to new, unseen data. The experimental design, as described, is sound and ensures that the testing is performed in a realistic and rigorous manner.

If you need further analysis or details on any specific aspect, feel free to ask!

---

## Responses by Gemini (May 14, 2024)

---

### Treatment condition: Finding errors in erroneous parts

#### Optimization error

##### Direct prompt

We analyze the claims in this paper one by one.

1. **Claim:** The authors propose a unified framework to model the problems of leveraging gold standard questions in crowdsourced labeling and understanding the aggregation process in peer review. **Verdict:** This claim appears to be **valid**. The paper introduces a "Convex Combination Generalized Linear Model" and a "Non-parametric Model" to address both crowdsourced labeling and peer review scenarios.
2. **Claim:** The authors provide theoretical guarantees for their estimators, including results on identifiability and convexity. **Verdict:** This claim is **valid**. Theorems 1 and 2 in the paper discuss the identifiability of the parametric model and the convexity of the proposed estimators, respectively.
3. **Claim:** The empirical evaluation demonstrates the superiority of the proposed methods over all existing methods in both crowdsourced labeling and peer review settings. **Verdict:** This claim is **partially valid**. The paper shows the effectiveness of the proposed methods in crowdsourced labeling tasks, outperforming several existing methods. However, the peer review analysis is exploratory and doesn't directly compare the method against other existing approaches. The paper provides insights into the decision-making of meta-reviewers but doesn't claim superiority over other methods in this context.
4. **Claim:** The proposed non-parametric estimator consistently outperforms the parametric estimator in crowdsourced labeling tasks. **Verdict:** This claim is **false**. Table 1 in the paper shows that while the non-parametric estimator performs well in some tasks, it doesn't consistently outperform the parametric estimator across all tasks. In fact, the parametric estimator (convex combination GLM) performs better in the 'texture' task.

---

Is Theorem 2 correct? Please check carefully.

Theorem 2 appears to be **correct**. The optimization problems in equations (4), (5), and (6) are all convex. Here's why:

- **Equations (5) and (6):** The objective functions in both equations are quadratic, which are inherently convex. The constraints are linear inequalities, which also define convex feasible regions.
- **Equation (4):** This is a special case of equation (5) with  $(\lambda = 0)$ . Since (5) is convex, and (4) is a more constrained version of it, (4) is also convex.

The proof provided in the paper correctly identifies the convexity of the objective functions and the linearity of the constraints, leading to the valid conclusion that all three optimization problems are convex.

---

Is (4) convex? Please check carefully.

Yes, the optimization problem (4) is convex. The paper provides a proof for this in Theorem 2. Here's a breakdown of why it's convex:

- **Convex Objective Function:** The objective function in (4) is a least-squares problem, which is a quadratic function. Quadratic functions are known to be convex.
- **Convex Constraints:** The constraints in (4) are linear inequalities:
  - $(\epsilon \leq \beta_I \leq 1)$  for all  $(I \in [|Z|])$
  - $(\sum_{I \in |Z|} \beta_I = 1)$
  - $(\beta_1 \leq \beta_2 \leq \dots \leq \beta_{|Z|})$  (optional monotonicity constraint)
 Linear inequalities define convex sets.

Since the objective function is convex and the feasible region defined by the constraints is also convex, the overall optimization problem (4) is convex.

---



---

## Statistical error

### Direct prompt

We analyze the claims in this paper one by one.

1. **Claim:** The authors propose a unified framework to model the problems of leveraging gold standard questions in crowdsourced labeling and understanding the aggregation process in peer review. **Verdict:** This claim appears to be **valid**. The paper introduces a "Convex Combination Generalized Linear Model" (1) and a "Non-parametric Model" (3) that are presented as a unified framework to address both crowdsourced labeling and peer review scenarios.
2. **Claim:** The authors provide theoretical results on the identifiability of their proposed statistical model and the convexity of their estimators. **Verdict:** This claim is **valid**.

Theorems 1 and 2 in the paper directly address the identifiability of the model and the convexity of the estimators, respectively.

3. **Claim:** The empirical evaluation on a crowdsourced labeling dataset shows significant improvements over previous methods. **Verdict:** This claim appears to be **valid**. Table 1 in the paper presents results where the authors' methods ("Our convex combination GLM estimator (4)" and "Our non-parametric estimator...") outperform several baseline methods in most tasks.
4. **Claim:** The qualitative study on peer review using ICLR 2022 conference data offers insights into the decision-making approaches of meta-reviewers. **Verdict:** This claim is **valid**. The paper presents a case study on ICLR 2022 data, analyzing how meta-reviewers weigh reviewer scores with different confidence levels. It provides insights into the relative importance of different confidence levels in the decision-making process.

Based on the information provided in the paper, all the claims appear to be valid and supported by the presented methodology, theoretical results, and empirical evaluations. There is no evidence in the paper to suggest that any of these claims are false.

---

Is Theorem 1 correct? Please check carefully.

The statement of Theorem 1 in this paper appears to be **incorrect**. The theorem pertains to the identifiability of the model parameters under specific assumptions. The authors claim that the model is identifiable if and only if the matrix  $1^T \Sigma$  is of rank  $|Z|$ . However, this condition is not sufficient for identifiability.

Here's why:

- **Identifiability:** A model is identifiable if distinct parameter values lead to distinct probability distributions of the data. In this context, it means that different values of the parameter vector  $\beta^*$  should result in different expected values for the overall scores  $y_i$ .
- **The Flaw in the Argument:** The authors derive the condition  $\Sigma \beta^* = 0$  and  $1^T \beta^* = 1$ . While these equations are necessary for the model to hold, they do not guarantee identifiability. It's possible to have different  $\beta^*$  vectors that satisfy these equations but still lead to the same expected values for  $y_i$ , especially when there are dependencies or redundancies in the evaluations.

### A Simple Counterexample:

Consider a scenario with two levels ( $|Z|=2$ ) and two items ( $n=2$ ). Suppose both items receive evaluations with the same level (e.g., both are evaluated by experts). Let's say the scores are:

- Item 1:  $x_1 = [2, 3]^T$
- Item 2:  $x_2 = [4, 5]^T$

In this case, the matrix  $\Sigma$  will be a  $2 \times 2$  matrix with all zeros, and thus  $1^T \Sigma$  will have rank 1. However, multiple  $\beta^*$  vectors can satisfy the equations  $\Sigma \beta^* = 0$  and  $1^T \beta^* = 1$ , leading to the same

expected values for  $y_1$  and  $y_2$ . For example, both  $\beta^*=[0.5,0.5]^T$  and  $\beta^*=[0.2,0.8]^T$  would satisfy the equations but result in different weightings of the evaluations.

### Conclusion:

The condition stated in Theorem 1 is a necessary condition for identifiability, but it is not sufficient. The authors need to provide a stronger condition or revise their claim.

---

Is the only if condition in Theorem 1 correct? Please check carefully.

The "only if" direction of Theorem 1 appears to be **correct**. Here's why:

- **The "only if" direction:** This part of the theorem states that if the model (1) with a linear link is identifiable, then the matrix  $1^T \Sigma$  must have rank  $|Z|$ . In other words, the full rank of this matrix is a necessary condition for identifiability.
- **The reasoning in the paper:** The authors derive the equations  $\Sigma \beta^* = 0$  and  $1^T \beta^* = 1$  from the model assumptions. These equations represent a system of linear equations where  $\beta^*$  is the unknown. For this system to have a unique solution (which is required for identifiability), the coefficient matrix  $1^T \Sigma$  must have full rank, which is  $|Z|$  in this case.
- **Why this is correct:** If the matrix  $1^T \Sigma$  does not have full rank, there will be infinitely many solutions for  $\beta^*$ . This means that different parameter values could lead to the same distribution of the data, violating the definition of identifiability.

**In summary**, the reasoning provided in the paper for the "only if" direction of Theorem 1 is sound. If the model is identifiable, the matrix  $1^T \Sigma$  must necessarily have full rank. The error in Theorem 1 lies in the "if" direction, as demonstrated in the previous response.

---

---

## Evaluation error

### Direct prompt

We analyze the claims in this paper one by one.

1. **Claim:** The authors propose a unified framework to model the problems of leveraging gold standard questions in crowdsourced labeling and understanding the aggregation process in peer review. **Verdict:** This claim appears to be **valid**. The paper introduces a

"Convex Combination Generalized Linear Model" and a "Non-parametric Model" to address both crowdsourced labeling and peer review scenarios.

2. **Claim:** The authors provide theoretical guarantees for their estimators, including results on identifiability and convexity. **Verdict:** This claim is **valid**. Theorems 1 and 2 in the paper discuss the identifiability of the parametric model and the convexity of the proposed estimators, respectively.
3. **Claim:** The empirical evaluation demonstrates the superiority of the proposed methods over all existing methods in both crowdsourced labeling and peer review settings. **Verdict:** This claim is **partially valid**. The results in Table 1 show that the proposed methods outperform existing methods in most crowdsourced labeling tasks. However, the paper does not include a quantitative comparison with other methods in the peer review setting, only a qualitative analysis.
4. **Claim:** The proposed non-parametric estimator consistently outperforms the parametric estimator in the crowdsourced labeling experiments. **Verdict:** This claim is **false**. Table 1 shows that while the non-parametric estimator with data-dependent parameter selection performs well, the parametric estimator outperforms it in the 'texture' task. Additionally, the non-parametric estimator with  $\lambda=0$  performs worse than the parametric estimator in all tasks.

---

Is Algorithm 1 correct? Please check carefully.

Algorithm 1 appears to be mostly correct, but there is a subtle error in how the final error rate  $\zeta$  is computed.

The algorithm correctly iterates over different values of  $\lambda$  (line 2), solves the optimization problem (6) for each  $\lambda$  using the gold standard questions (line 3), and computes the 0-1 loss  $\zeta_{\lambda,h}$  for each non-gold standard question  $h$  and each  $\lambda$  (lines 4-6).

However, the error in line 8 is that it selects the  $\lambda$  that minimizes the *mean* error rate across all non-gold standard questions. This is incorrect because the goal is to select the  $\lambda$  that minimizes the error rate on the entire set of non-gold standard questions, not the average error rate per question.

To fix this, line 8 should be modified to compute the total number of errors for each  $\lambda$  and then select the  $\lambda$  that minimizes this total count. Here's the corrected line 8:

Unset

8: Let  $\zeta = \min_{\lambda \in \{0, 2^{-16}, 2^{-15}, \dots, 2^{-1}, 2^0\}} \sum_{h \in H} \zeta_{\lambda,h}$

With this correction, the algorithm will correctly select the value of  $\lambda$  that leads to the lowest overall error rate on the non-gold standard questions.

---

What data is Algorithm 1 using for the testing (evaluation)? Is that appropriate? Please check carefully.

Algorithm 1 uses the gold standard questions (G) for training and the non-gold standard questions (H) for testing. This is appropriate because the goal of the algorithm is to select a hyperparameter  $\lambda$  that generalizes well to unseen data. By evaluating on the non-gold standard questions, the algorithm simulates the performance on new, unlabeled data.

Specifically, the algorithm does the following:

1. **Training:** It trains the non-parametric estimator (6) on the gold standard questions (G) for different values of  $\lambda$ .
2. **Hyperparameter Selection:** It selects the  $\lambda$  that minimizes the average error rate on the non-gold standard questions (H).
3. **Evaluation:** It reports the error rate of the chosen model on the non-gold standard questions (H).

This approach ensures that the selected hyperparameter is not overfitting to the gold standard questions and is likely to perform well on new, unseen data.

---

Are you saying that hyperparameter selection is being done on the test data?

No, the hyperparameter selection in Algorithm 1 is not being done on the test data. The algorithm uses the gold standard questions (G) for training and the non-gold standard questions (H) for validation. The hyperparameter  $\lambda$  is selected based on its performance on the validation set (H), and the final evaluation is also done on the validation set (H). This is a standard practice in machine learning to prevent overfitting and ensure the model generalizes well to unseen data.