We thank all reviewers for their reviews, for their constructive comments and suggestions and for appreciating our results. We apologize for the typos; all these minor points will be dealt with in the final version of our article.

R#1: For Theorem 3.2, ... should be $c \geq \frac{\log d - 2}{2}$. Thank you for checking the proof of Theorem 3.2. Maybe there is a misunderstanding here but one of the main conclusions of Theorem 3.2 is that c has to go to ∞ as $\delta \to 0$. Hence for small δ we cannot see how c can be equal to $(\log d - 2)/2$. If you meant $c \geq \frac{\log d - 2}{2\delta}$ then our statement allows this.

In the Appendix B, ... are redundant. We agree that the dashed illustration may be redundant we will update our manuscript accordingly.

In line 173(Appendix), ... (this will not influence the result). Indeed this is a typo, thank you very much for finding this!

R#3: Approximate incentive ... in the main text. Approximate incentive compatibility is indeed a weak concept in many cases. At the same time though it has been a successful theoretical tool for designing mechanisms in settings where exact incentive compatibility is very difficult to achieve. Additionally the use of approximate incentive compatibility can vastly increase the theoretical performance of designed mechanisms [1].

We also commit on moving at least a proof sketch for our main theorems in the main part of the paper with more specific references to low level technical lemmas in the supplementary material.

The paper is quite .. for future work. Thank you for the suggestions! We will add more details in the related literature discussion and we will also add a conclusion section with open problems and future work.

R#5: (1) The motivation of why a better ... should be used. We commit on moving the main intuitions of the construction and our proofs to the main part of the paper to help the reader. We will also clarify the reasons for using our soft-max function in practice. We briefly name a few reason apart from the achievement of a better approximation - smoothness tradeoff:

- the worst-case approximation guarantee can be very appealing in many practical applications where bad outcomes have severe consequences even when they happen with low probability,
- our soft-max function has a simple piecewise linear form which allows the computation of both the function and its derivatives to be done very efficiently. This might be relevant in applications where computations and differentiations of a softmax function appear very frequently in some code.

(2) No empirical support is provided for two of the three applications as claimed in the paper. Thank you for the suggestion! In this work our target was to establish the necessary theoretical framework for this problem. We certainly agree that the next relevant step is to try our findings both on simulated and on real-world data where truncation occurs.

(1) Line 135: some ... can be derived? It is both actually. (l_p, l_q)-Lipschitz is of particular interest to some applications, for example in mechanism design the notion of smoothness related to incentive compatibility is the (l_p, l_1)-Lipschitzness. Additionally, to the best of our knowledge, in settings where worst-case approximation is very important then (l_p, l_q)-Lipschitzness is the only available notion, so far, of smoothness that gives non-trivial results.

(2) Line 346, the ... gradient descent algorithms". We will make sure to add more on that. The most important feature of our soft-max function that can be utilized in ML applications is the fact that favors sparse outcomes. This is formalized via the worst-case approximation guarantee of our soft-max functions. This sparsity in the output can be useful both because it is known to help generalization but also because it makes the execution of every step of SGD more efficient and hence can potentially have a big impact on the efficiency of training algorithms.

(3) In the appendix line 513, ... significance visually. Thank you for the suggestion! We will certainly add the standard deviation in the plots for the final version of the paper.

References

[1] Frank McSherry and Kunal Talwar. Mechanism design via differential privacy. In *Foundations of Computer Science, 2007. FOCS'07. 48th Annual IEEE Symposium on*, pages 94–103. IEEE, 2007.