We want to thank the reviewers for their careful reading of the paper and their many constructive comments. Due to space constraints, we are not able to reply to every point made by the reviewers and focus our response on the most important points. Naturally, all of our responses here will be incorporated into the final version of the paper.

Additional baselines and the two-coin model: We would like to thank Reviewer 3 for suggesting several highly relevant references with additional algorithms. We therefore compared the M-MSR method with 8 new baseline methods, where 6 of them are two-coin methods (Minmax[1], Entropy(O)[2] , EM-twocoin[3], BP-twocoin[3], MFA-twocoin[3], and M-MSR-twocoin) and 2 of them are tensor methods (EM-MV[4] , EM-SM[4]).

Additionally, all reviewers suggest we should test our method on the two-coin model: here the labels are binary and the probability of giving the correct answer depends on the label, so that each worker is parametrized by two parameters. Besides the above methods and M-MSR, we will also simulate a natural extension of M-MSR to the rank-2 case:

$$x_i^{t+1}=\arg\min_x \sum_{j \mid (i,j)\in\Omega}(x^T y_j - C_{ij})^2; \qquad y_j^{t+1}=\arg\min_y \sum_{i \mid (i,j)\in\Omega}(x_i^T y - C_{ij})^2,$$

where $x_i \in \mathbb{R}^2$ denotes the ith row of X and $y_j \in \mathbb{R}^2$ denotes the jth row of Y; to deal with the corrupted entries, we throw out the largest and smallest F values among the quantities $|C_{ij}|/\left\|y_j\right\|_2$ inside the minimization problem in each step. We call this algorithm the M-MSR-twocoin algorithm. It is not hard to see that, if the true proportion of "0" and "1" labels in the data-set is known (more on this later), skill estimation in the two-coin can be reduced to a rank-2 matrix recovery problem to which this algorithm can be applied.

We had time to implement the same experiments as in the paper on 5 real datasets and one synthetic dataset. The synthetic dataset is created following the two-coin model. The results are shown in Figure 1. *We can see in the figure that M-MSR outperforms all the baseline methods on all 6 datasets. Surprisingly, M-MSR even beats its own natural generalization to rank-2, even though this generalization has the implicit advantage of knowing the true proportion of "0" and "1" answers; and it can also beat all the methods tailored to the two-coin model on the synthetic data set once adversaries are introduced.*

We have spent substantial efforts during the rebuttal period implementing these new baselines and we believe the results considerably strengthen the case for the effectiveness of M-MSR. While it may not be surprising that M-MSR outperforms various two-coin methods (since M-MSR is designed with adversaries in mind) it is quite surprising that M-MSR outperforms its rank-2 generalization.

Reviewer 2: More discussion for the consensus literature. Indeed, the reviewer is correct. We will move some of Section 5.1 to the supplementary information which will give us the space to discuss this connection.

Reviewer 3:Reproducibility. We will publish all the code with the final version paper on github.

Differences between our work and literature[5], literature[6]. One difference from literature[5] is that we aim to label the class of the tasks, and they aim to rate the tasks (not necessarily integer) and find the ones with high quality. The main difference, however, is that they require the ratings of a small portion of tasks are known. Literature[6] and our work both study the issue of worker ability estimation, however, we mainly focus on the adversarial setting while they do not.

Use of "gold standard" questions. One advantage of M-MSR is that it can tolerate arbitrary adversaries, including adversaries that return large fraction of correct answers (this can be be seen in Figure 1(g) in the paper), which means M-MSR can deal even with smart adversaries that manage to pass the gold question test.

Experimental results using raw datasets. Among the 17 real datasets we used, 11 of them had each worker with 10 tasks or more, so no removal of workers was necessary. For the other 6 datasets, we implemented experiments on 4 of them (Dog, Adult, Fashion1, and Fashion2) using raw data without removals; this did not change the results. The results in Figure 1 below also use raw data without removals. Experiments on remaining 2 raw datasets (Web and TREC) will be added in final version of the paper (Web is a multiclass data set and some of the two-coin methods cannot deal with that; TREC is very large and some of the methods take a very long time to run on it).

Reviewer 4: Notation. Note that we consider two problems: recovery of an $m \times n$ rank-1 matrix and crowdsourcing with W workers and M classes. We reduce the crowdsourcing problem to rank-1 recovery of a $W \times W$ matrix.

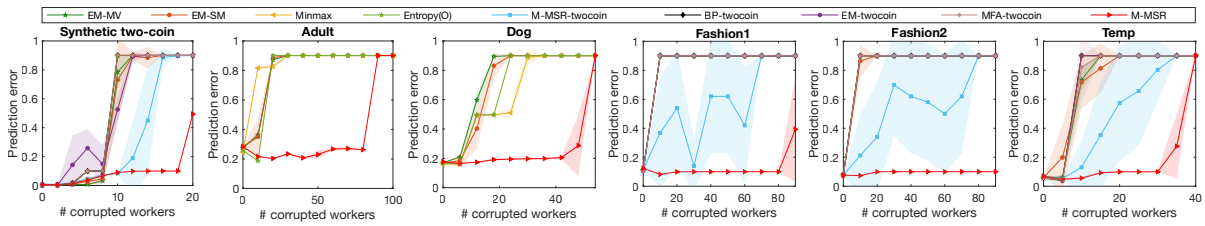

Figure 1: Experimental results. Solid lines are the average of 20 runs, and the shaded region show the standard deviation. The total number of workers of the datasets: Adult 269, Dog 109, Fashion1 196, Fashion2 198, Temp 76, Synthetic two-coin 40. Adult and Dog are multiclass datasets, hence the methods which are designed only for binary datasets are not shown. The synthetic dataset is created following two-coin model rule, where the worker probabilities of giving correct answers are uniform in $[0.5, 1]$ for both classes, and the true answers have equal probabilities of coming from each class.

[1] Zhou et al. Learning from the Wisdom of Crowds... [2] Zhou et al. Aggregating Ordinal Labels... [3] Liu et al. Variational Inference for Crowdsourcing [4] Zhang et al. Spectral Methods meet EM... [5] Steinhardt et al. Avoiding Imposters... [6] Zhou et al. MultiC^2: An Optimization framework...