We thank the reviewers for their constructive feedback, and first address **adding more experiments**, common to Reviewers 1, 2 and 4. Our main *algorithmic* innovation is in step 1 (robust subspace estimation), and our contribution in steps 2 and 3 (robust clustering and robust estimation) are *theoretical*, as we borrow existing algorithms. Hence, it was natural for us to focus on experimentally verifying step 1. However, we agree that more experiments will help solidify the theoretical guarantees, and will verify the following experimentally: (a) an experiment showing that the SOS approach (step 2) is robust under the linear regression probabilistic model which is not Poincaré; and (b) add an experiment showing that robust parameter estimation succeeds when applied with the classification together (step 3).

Detailed response to Reviewer 3:

- *Q: Could the adversary first look at *all* batches and then pick the corruptions?* A: Yes, the adversary can take a look at all three groups of batches and add corruption. We have revised Assumption 2 accordingly.
- *Q: more *direct* approach ...* A: [31] defines "meta-learning" as Eq. (2). However, we agree there are many ways to meta-learn, some of which are more direct but less understood. We will survey those approaches in Section 3.
- In practice, we oftentimes have problems with a large ambient data dimension, but a simple structure among the meta-training tasks (captured by small k in our setting). Our approach is tailored for such settings with $k \ll d$.
- We will address all the comments and typos in the final version of the paper.

Detailed response to Reviewer 4:

- *Q: ...why these types of extensions are necessary in practice...why are the existing methods they mention brittle...* A: Existing methods can completely break down with a single corrupted user. We will add the following remark and references: " [41] builds upon principal component analysis and linear regression, both of which are known to be brittle to outliers [39,19]. For example, a single corrupted user can result in an arbitrarily bad subspace estimation in the first step of [41]. This causes the meta-learning algorithm to learn nothing about the true regression parameters, resulting in an completely random prediction in the subsequent step."
  In particular, under the setting of Corollary 1.3, with a tiny fraction of adversary with $\alpha = 1/n$, our approach guarantees error $\mathcal{O}(k/n)$ that decreases with n whereas [41] will have error $\Omega(\rho/\Delta)$ that does not decrease with n. Further, for general $\alpha \in (0, 1/2)$, naive pruning techniques, for example removing the datapoints with extremely large magnitude, is not enough to resolve the issue in a high dimension.
- *Q: Why is this the right adversarial model?* A: This is the right model for security, in the sense that it is the strongest adversarial model (among those that can corrupt the same number of samples), and more importantly, we can still make the algorithm robust. We will add a remark that "Following robust learning literature [44, 25], we assume a general adversary who can adaptively corrupt any α fraction of the tasks, formally defined in Assumption 2. This parameter $\alpha \in [0, 1]$ captures how powerful an adversary is. Among all adversaries that can corrupt an α fraction of the dataset, we assume the strongest possible one that can *adaptively* select which samples to corrupt and replace them with *arbitrary* data points.". This is also a realistic adversarial model, in settings like federated learning where an α fraction of devices can be compromised.
- *Q: ...unhelpful as it presents results before we've even seen what the model in question is...* A: We moved the generative model earlier than Corollary 1.1. We moved the adversarial model earlier than Corollary 1.3.
- *Q: Much more context should be given into SOS methods.* A: Due to the space limitation, we had to be selective. In the revision, we will add a subsection in Section 3 with preliminary on SOS methods applied to robust estimation.
- *Q: Corollary 1.1: I am a bit surprised that there is no dependence...* A: Since Corollary 1.1 is an informal version, we restrict our focus on d and k and assumed that the error ϵ is a positive constant. A more formal version of Corollary 1.1 is Corollary 1.3 and Theorem 1, where the dependencies on the final accuracy are highlighted in adversarial tolerance, sample and running time complexity. Below we re-write Corollary 1.1 with dependency in ϵ:

  Corollary. *For any $\epsilon > 0$, given a collection of n tasks each associated with $t = \tilde{\Omega}(1)$ labeled examples, if the effective sample size $nt = \tilde{\Omega}(dk^2 + k^{\Theta(\log k)} + dk/\epsilon^2)$, then Algorithm 4 estimates the meta-parameters up to the accuracy of ϵ w.h.p. in time $\mathrm{poly}(d, k^{(\log k)^2}, 1/\epsilon)$, under certain assumptions on the meta-parameters.*
- *Q: L74: What is the dependence on k in the $\tilde{\mathcal{O}}(d^2)$?* A: $\tilde{\mathcal{O}}(d^2)$ sample suffices for estimating the covariance matrix itself accurately under Frobenius norm, which implies accurate estimation of top-k subspaces for any k. Therefore, there is no dependence of k in $\tilde{\mathcal{O}}(d^2)$, as we explicitly write in Remark K.1 in the supplementary material.
- *Q: Thm. 1: Why is the only dependence on δ in t_{L2}?* A: The target guarantee is parametrized by the failure probability δ and the accuracy ϵ. For subspace estimation and clustering, we apply concentration of measure on the whole dataset, and hence n_{L1} and n_H depends on $\log(1/\delta)$, which is hidden in the $\tilde{\Omega}(\cdot)$ notation. For classification, we apply concentration to each task, and hence t_{L2} depends on $\log(1/\delta)$. As for the accuracy ϵ, (as we explain in L159 of the submission), the subspace estimation and clustering steps succeed with high probability as soon as they achieve accuracy of $\mathcal{O}(\Delta/\rho)$, regardless of the final target accuracy ϵ. The refinement with classification is solely in charge of achieving the target ϵ accuracy, and hence $1/\epsilon^2$ dependence only shows up in n_{L2}.
- We will modify the presentation of the setting and priors for better readability in the final submission as suggested.