Thanks to all of the reviewers for the time spent reading and commenting on our work.

General Comments: Our main contributions are efficient outlier-robust algorithms for sparse recovery problems (sparse mean estimation and sparse PCA) that rely on a novel spectral filtering method. Previous polynomial time algorithms for these problems inherently relied on convex optimization and in particular required solving a large SDP polynomially many times. In more detail, as mentioned in our introduction (lines 53–57), prior work gave an ellipsoid-based method whose separation oracle is an SDP. As a result, this prior method is extremely impractical (and, unsurprisingly, has not been implemented). One could also construct polynomial time algorithms for our problems that iteratively filter outliers using an SDP in each iteration. (For sparse mean estimation, this algorithm is briefly described in lines 151–157.) However, even this filter-based approach is quite slow, and in particular is very different than the algorithms we design. As pointed out by the first reviewer, we are able to use the problem structure to eliminate the need for *any* SDP. Ours are the first potentially practical robust algorithms for the problems considered.

Reviewer 1: We thank the reviewer for their careful reading of the paper and their positive feedback.

Reviewer 2: *Runtime and Comparison:* In the revised version of our paper we will include a plot of the running time. For reference, we give a few numbers for robust sparse mean estimation here: (a) For $k = 1$, $d = 10$, $m = 50$: 0.005 seconds; (b) For $k = 10$, $d = 300$, $m = 50$: 0.014 seconds; (c) For $k = 40$, $d = 1000$, $m = 8000$: 1.4 seconds.

We would like to be able to directly compare to the prior published work on robust sparse mean estimation, but that algorithm is quite complicated and has never been implemented. It also will surely be much worse: it uses the ellipsoid algorithm and requires a large SDP as a separation oracle. For the same $k = 10$, $d = 300$, $m = 50$ case that our algorithm solves in 0.014 seconds, the very first SDP takes 10 seconds to solve with CVXOPT; the full ellipsoid-based algorithm, if implemented, would take many times that.

Overview of our Algorithms. Here we expand upon the intuition for our robust sparse mean algorithm, in particular lines 167–170. The goal is to produce a filter if the Frobenius norm of the difference between the empirical covariance ($\tilde{\Sigma}$) and the true covariance (I) is large on the largest k^2 entries — otherwise, if the difference is small, we don't need to filter at all.

We use the terminology "good samples" (as in previous works in the area) to mean a set of samples satisfying certain deterministic conditions that an uncorrupted Gaussian dataset will satisfy with high probability for a large enough sample size. Our algorithms succeed under these deterministic conditions, which are described in the supplementary material (e.g., Definition A.2 for the sparse mean case). One such condition is that the empirical expectation of every degree-2 homogeneous polynomial $p(x)$ with k^2 nonzero coefficients is close to its true value. If $\|(\tilde{\Sigma} - I)_U\|_F$ is too large, then we show that there exists such a polynomial p that takes a large value in expectation, and hence on a reasonable fraction of the sample points. But $p(x)$ is not large on average over a good set, so most of the points x with large $p(x)$ must be outliers. Therefore, we can remove the points with large $p(x)$ to filter out a set of mostly corrupted points. (Notation clarification: As defined in the Appendix, $h_k(\cdot)$ is the thresholding operator which zeros out all but the k largest-magnitude entries of a vector.)

Comments on Related Work: The Cheng et al. [CDG19] robust mean estimation algorithm works for the dense case. In particular, it has sample complexity $N = \tilde{\Omega}(d/\epsilon^2)$. That algorithm has no implications for the k-sparse setting studied here, where we are interested in algorithms with sample complexity $N = \mathrm{poly}(k, \log d)/\epsilon^2$.

While recent literature has developed robust mean estimation algorithms under more general distribution families, this is *not* the case for the sparse setting studied here. The only previous algorithm for the sparse setting is the ellipsoid-based method from [BDLS17] working under the same Gaussian assumptions as ours.

Reviewer 3: *Sample Complexity optimality:* The claim we make in lines 72–73 is that our $\tilde{O}(k^2 \log d/\epsilon^2)$ sample complexity matches existing Statistical Query lower bounds, which hold for $k < \tilde{\Omega}(\sqrt{d})$. As the reviewer notes, above this threshold, dense mean estimation algorithm performs better (and matches the SQ lower bound).

Novelty of Our Filtering Algorithm: The idea of filtering out outliers is of course not new. The question is how to find a filter that removes the outliers. This is a problem-specific task that can be highly non-trivial.

As the reviewer seems to be suggesting, there is a similarity between ellipsoid-based methods and filtering methods. But the existing ellipsoid-based method for robust sparse mean estimation relies on an SDP for its separation oracle. The key contribution of our paper is to avoid convex programming entirely, producing a faster filter. To get such a filter for robust sparse mean estimation, we actually need two filters: one linear and one *quadratic*. The quadratic filter has no analog in prior work, yet (as we demonstrate in our experiments) is *necessary* for our approach. Analogous comments apply for the robust sparse PCA setting.