We'd like to begin by thanking the reviewers for their careful reading of the paper and insightful feedback. It has helped bring to light many typos and some poorly explained sections.

Reviewer 1

- **Figure 4:** This figure shows how the ρ_i impact performance. While small ρ_i lead to improved performance, it is critically important what the corresponding Δ_i are. This is due to the nature of $\tilde{H}_2 = \max_{i \geq 2} i\rho^2_{(i)}/\Delta^2_{(i)}$: if ρ_i are only small for large Δ_i, there will not be very large gains over $H_2 = \max_{i \geq 2} i/\Delta^2_{(i)}$. If, however, for small Δ_i we have correspondingly small ρ_i (as is empirically the case on 2 different datasets, as shown in Fig. 4), then large improvements will be realized. We will clarify this figure more in the final version.
- **Theory versus Practice:** We quantify our theoretical gain as $H_2/\tilde{H}_2$ as in lines 84-88. These theoretical gains do not capture the entire picture, only predicting a gain of around 7x over Med-dit for RNA-Seq 20k as opposed to the 50x reduction realized, as our analysis is only able to incorporate pairwise dependence. A lengthier discussion on the other gains we are able to realize and the difficulty in analyzing them can be found in Appendix C.1
- **Error bars on plots:** We will include these in the final version, thank you for the suggestion.

Reviewer 2

- **corrSH error rate:** Since corrSH takes a budget as an input, we vary the input budget and plot the error probability for various budgets, noting in the table the smallest budget above which all error probabilities were 0.
- **Typos:** We thank the reviewer for their close reading and pointing out the typos and other unclear portions, these are being corrected for the final version. For example the "middle of the road point" was very poorly characterized; we listed it in the figure caption as $i = 10000$ out of $n = 20000$ (point with the median value of $\{\theta_j\}_{j=1}^n$), but this was a very vague way of referring to it.
- **Remark 3:** Yes, the budget of corrSH is a very important question. Due to the page limit we were forced to relegate many important details to the Appendix; with the extra page allotment for the final version we will make sure to move this remark back to the main text.
- **What yields small $\tilde{H}_2$:** This is a great question that we are currently pursuing; previous works like Med-dit also tried to analyze a similar problem, examining $H = \sum_{i=1}^n \left\lceil \frac{\log n}{\Delta_i^2} \bigwedge n \right\rceil$. In this work, they took several pages to show that $\mathbb{E}[H] = O(n \log n)$ under the assumption that $\theta_i = N_{(i)}$ where $N_i \sim \mathcal{N}(0,1), i = 1, \ldots, n$ and $N_{(i)}$ denotes the i-th order statistic. Note that this is an assumption on the θ_i, and not a true generative assumption on the $\{x_i\}$. Unfortunately these techniques do not translate over, as H is much easier to analyze than $\tilde{H}_2$, as H is a summation over the Δ_i's rather than a max, and does not involve ρ_i's which are very difficult to analyze.

Reviewer 3

- **Problem Motivation:** Algorithms for finding the medoid have gained recent interest in the community; Newling and Fleuret won the best paper award in AI Stats 2017 for their work on a sub-quadratic medoid algorithm [9], and Med-dit followed after this. In addition to the basic medoid, such algorithms are building blocks for k-medoid clustering, a commonly used preprocessing step for unlabeled data. Some algorithms for this involve Voronoi iteration, where the medoid of a cluster of points is computed as a subroutine [1]; our scheme could be used to drastically speed up this step. Some alternate algorithms for k-medoid clustering are PAM, CLARA, and CLARANS [2]. Our contribution is methodological and goes beyond simply the medoid case, and the methods of correlated sampling we introduced appear to be applicable to these algorithms as well.
- **Lower bounds:** This is a line of ongoing research, as we believe that a lower bound is important and would cleanly close this problem. However, this appears to be highly nontrivial due to the complex dependence structure stemming from the underlying computational problem, as discussed in Sec 2.1 and Appendix C.

References

[1] H.-S. Park and C.-H. Jun, "A simple and fast algorithm for k-medoids clustering," *Expert systems with applications*, vol. 36, no. 2, pp. 3336–3341, 2009.

[2] L. Kaufman and P. J. Rousseeuw, "Partitioning around medoids (program pam)," *Finding groups in data: an introduction to cluster analysis*, pp. 68–125, 1990.