Firstly, we would like to thank the anonymous reviewers for their very helpful comments and suggestions which greatly enriched our contribution. We address each point raised by **R1**, **R2**, **R3**, and **R4** separately.

R2 [Subspace errors] The error analyses of our algorithm are summarised in Lemma 11, and Theorems 2-3 in Appendix **C.4**. These results assume data matrices in $\mathbb{R}^{d \times tMb}$. In this context, it is assumed that at each time t each of the M nodes has observed at most b vectors, so dependence on $n \leq tMb$ is implicit. We chose this formulation because it better reflects the streaming nature of our algorithms and simplifies the exposition.

R2 [DP for the 1st eigenvector] It is true that the utility guarantees we present are limited to the first eigenvector. We point out that our key contribution (wrt DP) is in adapting Mod-SULQ to the limited-memory and federated settings. We have edited Section 3.3, Lemma 2, and the introduction to emphasise this. That being said, we think a theoretical guarantee for the leading r eigenvectors is a plausible direction for future work. Indeed, our experiments provide numerical evidence that the utility of Federated-PCA extends beyond the first eigenvector.

R2 [Claim on $\mathcal{O}(cdn)$ memory requirement] While Lemma 2 considers matrices in $\mathbb{R}^{d \times n}$, it is invoked on $\mathbf{B} = \mathbf{X} \in \mathbb{R}^{d \times b}$ which implies $n = b$. This is remarked in line 204, but we agree that it's not sufficiently clear. We have reformulated Lemma 2 to remove ambiguity.

R2 [DP concerns] We state in numerous places of the paper (see, for example, lines 88, 192, 256) that our DP results *extend* the results in [8] to the context of streaming computation. As clarified previously, our key contribution is to guarantee that PCA can be computed in a federated and differentially private way in devices with limited memory. In doing so we obtained an asymptotic on the variance of the perturbation that improves the state-of-the-art for the non-symmetric Gaussian case.

R2 [Concerns on case $M \gg n/M$] In Theorem 3 (Appendix) we observe that as long as the tree depth is shallow for each t, then the overall error will be very close to $\min(\operatorname{rank} \mathbf{Y}_t^i) \forall\, i \in [M]$. This holds regardless of the fan-out of the tree. The assumption about the tree-depth, in the federated setting, seems justified: we expect to have many leaf nodes and a few upwards aggregation points. More importantly, one of our objectives was to have a balanced, flexible approach in terms of communication and performance.

R2 [Dependence on c for the utility bound] As explained above, Lemma 2 is invoked on batches $\mathbf{B} = \mathbf{X} \in \mathbb{R}^{d \times b}$, so the utility bounds depend on b rather than n. The parameter c is an small "local" variable that ensures that computation can be handled by limited-memory devices. We have rephrased Lemma 2 to avoid confusion.

R2 [Experiments in Fig. 2] We believe the reason we observe this performance benefit is that in our experiments we used the r-truncated SVD (with r=6) to compute the final subspace before selecting to project using the first 2 PC's. Quantitatively, this empirically indicates that block based methods are able to better capture the progressive geometry and retain a better shape structure overall, which translates to savings in computation and memory. However, this property only holds in cases where the sample size is sufficiently large (see sample complexity guarantees in Lemma 9). In particular, we believe this is why our MNIST results are better than the Wine-quality dataset.

R2 [Comparison with distributed stochastic PCA methods] For our experiments we elected well-established algorithms in the field that were easy to implement or had source code readily available. This is because replicating the distributed setups and experiments described in these works was not practically feasible. However, we qualitatively compared the references brought forth in our lit-review. In addition one key difference in these works is that all require some form of synchronisation, which in a federated setting would add considerable overhead. Further, our algorithm is limited by potential shortcomings of the SVD itself. Motivated by the federated setting constraints, our federated error analysis, and our empirical results struck a balance between flexibility, performance, and accuracy.

R2 [Related functions] Related functions are defined in Section 3.2 together with their full implementation. The intuition for the choice of α, β is to damp the singular values to achieve an adaptive estimation of the rank. We have included a note to explain this together with a remark about how these parameters should be tuned and initialised (concretely, nominal values are 1 for α and 10 for β).

R4 [Limited details in tree-shaped aggregation] For simplicity, a tree structure was used to implement the merging strategy. However, we note that this structure can be violated without performance loss due to the the consequences of Lemma 10 (See Appendix C.2).

R4 [10k samples of MNIST] We confirm that, for efficiency, we only used MNIST's standard test set. However, a similar result holds for the full dataset. We included a an extra remark to explain this in the paper.

R1, R2, R3, R4 [Typos, clarity, grammar, notation, extra references] Finally, we have addressed all the suggestions regarding typos, clarity, grammar, notation, and additional references. In particular, we reworked the writing in sections 3.1, 3.2, 4.1, and 4.2. We also clarified our algorithms and the readability of the figures.