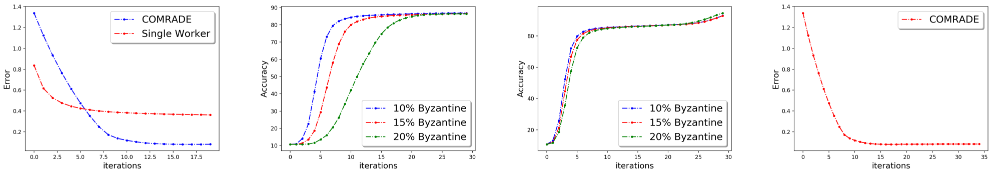
We like to thank the reviewers for their insightful feedback. We answer the questions raised by them below.

Reviewer 1: We thank the reviewer for appreciating our work, and plan to add short proof sketches. Regarding experiments, note that we only report the training accuracy of using the entire data. However, we have verified similar performance via a $80-20$ spilt between train and test data. We choose regularization parameter to be 1.

Reviewer 2 ***"Error grows with the number of workers*** m***"***: As seen in Theorem 1, the error (ϵ) indeed is proportional to $\sqrt{\log(m)}$, where m is the number of workers. But, if m increases, the error increases at a much lower rate. Moreover, the error is also inversely proportional to $\sqrt{s}$, where s is the number of samples at each machine. Usually, in practice, s is much larger than $\log m$, and hence in this regime, the error remains small and our results are useful.

"Unnecessary to use multiple Hessians and gradients": To verify the intuition about similarity between global and local gradients, we ran the following experiment as per the reviewer's suggestion. Given a total of n data points, we allow each machine to sample s data points. In Figure (a), we compare the convergence result (loss vs iteration) between our setup with multiple worker nodes and single worker with sampled data (as the reviewer asked). It is evident from the plot that our set up provides better result. For the single worker with exact Newton, the result is worse due to the high variance as it is based on a fraction of the data. For our case the average of the local update reduce the variance and provide better results. Hence, averaging the approximate update from local machines is better than the exact Newton method in one machine.

"Different Kinds of attacks" We show the robustness of COMRADE when the byzantine machines send $-c \times p$ in Figure (b), and $p + \mathcal{N}(0, 100)$ ('noise') in Figure (c), where we choose $c = 0.9$ and p as the update only with the good machines. It is evident that our algorithm is able to handle such byzantine attacks with norm based thresholding.



(a) (b) (c) (d)

Reviewer 3: ***"exact gradient is equivalent to classical sketch "***: In the special case of a squared loss, such connection indeed exists, although we are considering more general loss functions. We will add a discussion on this.

"GIANT seems to converge to a lower accuracy than the proposed algorithm": We believe that with rigorous tuning of the parameters (like step size), both the algorithms will yield similar accuracy. In our paper, however, we choose same parameter choices for both the algorithms. Our intent is to show even with one round communication, we can match the accuracy of GIANT (which uses 2 rounds). In some experiments, owing to this fixed parameter setting, COMRADE seems to achieve better accuracy.

"how to choose T is not mentioned" We run our algorithm, COMRADE for a sufficiently many iterations to ensure convergence. Alternatively, for the stopping criteria we can also choose the norm of the update as an indicator. In Figure (d), we plot the loss vs iteration with norm of the update ($\|p\|_2 < 0.1$) as a stopping criteria.

"loss vs iteration plots": Agreed. Note that in response to the previous question, we do provide a "loss vs iteration" plot.

'Step size and line search': We choose a fixed step size for all the experiments. GIANT uses Armijo–Goldstein condition to choose the step size, which requires the knowledge of the full gradient. As the first round of communication for the GIANT type algorithm computes the full gradient explicitly, the line search is feasible. In our case gradient information is not available as our algorithm only computes local updates (Hessian inverse times the gradient based on the local data). This provides the advantage of **less communication** but robs the opportunity to implement line search type algorithm. But we show in our numerical results that we achieve good convergence even with fixed step-size.

"The cost functions and regularization term": Apologies. In experiments, we choose regularization parameter to be 1.

"Typos" We will fix the typos and provide better visibility to plots.

Reviewer 4 ***"qualitative' rather than 'quantitative":*** We agree with the assessment of the reviewer. Indeed the error depends on several problem dependent parameters, and a clear choice of s is difficult to obtain in practice. However, our focus is to show that COMRADE converges to a vanishingly small error floor with sufficiently large s. We also show that error decays in $\mathcal{O}(1/\sqrt{s})$ rate.

"Numerical experiments": By accuracy, we meant $1-$ the classification error (in percentage). With a rigorous tuning of parameters and hyper-parameters, the algorithms may achieve close to 100% accuracy. However, our goal is to show that, for any fixed parameter setting, COMRADE (which communicates one round per iteration) can achieve the same accuracy as the standard two round based algorithms such as GIANT.

"$\lambda > 0$ and Remark 3": Indeed, we only require $\lambda > 0$ to ensure the invertibility of the Hessian. We will rephrase the remark. We will also add the strong convexity comment.

"Linear-quadratic convergence reference" Note that GIANT ([30] in the paper) has linear-quadratic rate of convergence; also see [21] in the paper. We will add a few other references in the revised version as well.