We thank all the reviewers for the helpful comments and questions. Before moving on to answer reviewers' questions, we first clarify the main **contributions** and **motivations** of our paper.

The goal of the paper is to better understand neural network optimization, especially the interactions between algorithmic choices and batch-size. We proposed a simple, yet useful toy model – Noisy Quadratic Models. This model captures the essential behavior we see in real neural networks (as validated by our experiments) while **allowing us to run experiments in seconds**. Note that most assumptions we made are based on existing theoretical results or experiments in real neural networks. We stress that NQM makes it easy to test new empirical ideas for practitioners and derive new, testable theoretical results for theorists (as an example, we studied the role of momentum analytically, showing that the momentum and learning rate are interchangeable in the small batch regime, which is non-trivial in stochastic setting).

We now address all comments and questions in order.

Interaction of momentum and preconditioning (R1). For neural net experiments, we applied preconditioning before the momentum. For the NQM, with fixed preconditioning, both methods are equivalent. We will update the paper to clarify these points.

Relationship between the Hessian and Fisher matrices (R1). For the models we study, the Fisher matrix is equivalent to the Gauss-Newton Hessian; see Martens, (2014) for details. We will make this explicit in the paper.

Assumption of diagonal Hessian (R2, R3). The assumption that the Hessian is diagonal can be made *without loss of generality* in the sense that (as noted in lines 94–98) we focus on algorithms such as SGD and Heavy-ball momentum that are invariant under variable rotations. For such algorithms, we can analyze the evolution of iterates in a basis that makes the Hessian diagonal, without changing the dynamics of the system.

Theoretical contribution (R2). We stress here that the analysis of momentum in stochastic setting is non-trivial. In particular, stochasticity (gradient noise) introduces extra difficulty in analyzing heavy-ball momentum. For the NQM, we showed theoretically that momentum SGD performs similarly to plain SGD in the regime of small batch sizes but helps in the large-batch regime, which also matches previous studies and our large-scale experiments.

NQM is too simple, and some assumptions are false for neural nets (R2, R3). When constructing a model for some phenomenon, the model does not need to be completely faithful in all respects; rather, it is the act of abstracting away inessential features that allows one to tractably analyze phenomena. Simplicity is a virtue, as long as the model captures the effects relevant to the phenomenon. One of our main contributions is showing that (empirically) features such as non-convexity, non-stationary gradient noise, etc., are not needed to explain certain neural net training phenomena which have recently received a lot of attention. Abstracting away these features will make it much easier for others to build on our work by further analyzing these phenomena. (Of course, for explaining other phenomena we don't consider, such as local optima, one might need to re-introduce features such as non-convexity.)

Assumptions of fixed gradient noise and particular form of preconditioner (R2). (Note that we *don't* require the noise distribution to be Gaussian.) Both assumptions are required for analytic analysis and fast simulation, yet reflect (to some extent) the reality of neural networks. Particularly, we showed in the appendix that the Fisher matrix stabilizes after a few epochs of training on ResNet, supporting the assumption of fixed gradient noise. In other words, we expect the assumption fixed gradient noise roughly captures the essence of real neural networks in throughout all of training except the initial phase.

Quadratic loss function (R2, R3). We stress here that there are several ways one might try to justify the use of convex quadratic models of the objective (also see lines 83-89 of our submission). First is the well-known fact that any smooth function will resemble a convex quadratic in a small enough neighborhood around a local minimizer. And recent theoretical work (Jacot et al., 2018, Du et al., 2018, Lee et al., 2019) has argued that very wide neural networks will deviate only a small distance from the initial random point in parameter space throughout training, and that they thus behave similarly to linearized networks for the purposes of training. This implies that the objective will appear "locally convex" throughout training, and that for the squared error loss will even resemble a convex quadratic. Adding further support for the "networks as linearized models" approximation is the fact that the Generalized Gauss-Newton matrix (Martens, 2014), which is the matrix of choice behind the most powerful 2nd-order neural network optimization methods, can be seen as the Hessian of the training objective under said approximation. Finally, we note that quadratic approximations also motivated Laplace approximation (MacKay, 1992) and variational inference (Zhang et al., 2018) in Bayesian neural networks.

We thank all reviewers again. We hope that our responses address your comments and concerns.