We thank the reviewers for the detailed and insightful reviews. As the reviews noted, our work 1) introduces “novel concepts” such as learning order, 2) provides theoretical and empirical evidence for our claims 3) proposes a mitigation strategy for small learning rate which works both theoretically and empirically. We will address questions below and incorporate feedback into our final revision.

[R1]: “I don’t exactly see if small batch vs large batch captures this phenomenon; if yes ... should say explicitly.”

• Yes, modulo some minor nuisances, the connection (smaller learning rate corresponds to larger batch) is simply through the relative scale of the noise. Smith et al. [2017] make an explicit connection between small vs. large batch and learning rate – this connection is due to the scaling of SGD noise, which can be increased by either increasing learning rate or decreasing batch size.

[R1]: “A small discussion on if the phenomenon has been observed for different datasets/tasks with different optimizers”

• The existing literature on large/small batch or large/small learning rate largely focuses on many vision tasks (including ImageNet) using SGD. The phenomenon may not be true for other optimizers such as Adam, though.

[R1]: “concept of "memorizable and generalizable", though intuitive, is sketchy and not formally explained ... authors should attempt to formalize these - perhaps identify based on sample complexity”

• We acknowledge that the terms “memorizable” and “generalizable” are potentially confusing. Memorizable refers to patterns that require low sample complexity but complex models to fit. On the other hand, “generalizable” patterns require high sample complexity, but can be fit by linear models. We will revise our terminology to clarify this distinction.

[R1]: “what is "inherently noisy"?”, “what do you mean by "getting annealed"”

• By “inherently noisy”, we refer to the fact that high noise in the datapoints will necessitate larger sample complexity. For example, in our distribution $\mathcal{P}$, the norm of the noise is $\sqrt{d}$ times that of the signal $w^\star$, resulting in a $\Omega(d)$ sample complexity. By “getting annealed”, we mean reducing the pre-activation noise by a constant factor at a certain epoch.

[R1]: “How important is the Gaussian noise injected in every step for the analysis?”

• The Gaussian noise is essential for our analysis, as it models noise from SGD. Our analysis relies on the fact that the scale of this Gaussian noise is larger with a larger learning rate.

[R1]: “Gaussian noise . . . why ... add before activation? ... add noise after, or maybe just to the SGD iterates?”

• Our theory suggests that SGD affects the training dynamics and generalization through adding the noise to the pre-activations, and therefore we did exactly the same thing as a mitigation strategy to imitate SGD. Adding noise to the gradient would also serve as a mitigation strategy for our particular data distribution that we analyze. However, because the analysis says that the fundamental benefit of the noise is perturbing the pre-activations, we suspect that pre-activation noise will transfer better to large-scale real datasets (which is indeed true).

[R1]: “There is hardly any discussion on contribution in proof techniques”, “"In our analysis, the underlying kernel is changing over time" (line 100) ... what tools are used, and moreover what analysis tools do they contribute”

• We will clarify the intuitions and contributions of our proof techniques more in the revision of our paper. A main contribution of our proof techniques is that we can deal with the changing kernel caused by the rapid changes of the activation pattern in contrast to the NTK results that often require stable activation patterns. We extend the neural tangent kernel techniques to the case with a sequence of kernels that share a common optimal classifier (Theorem C.2).

[R2]: “whether this analysis can be extended to other losses (for example, mean square loss)”

• It’s unclear whether the analysis can be extended – with squared loss it’s empirically unclear whether such a phenomenon still exists. Theoretically, we at least need to modify the construction of the data distribution. We used the property of logistic loss that as long as the sign of the prediction is accurate and the magnitude is sufficiently large, the loss gradient vanishes. (This results in the behavior that the small learning rate ignores the x_1 components of examples containing both x_1 and x_2 patterns.) However, this property does not immediately extend to squared loss.

[R2]: “It would also be good to explain the reason we need regularization here.”

• The regularization simplifies our analysis. Recall that in each iteration, we add fresh Gaussian noise Ξ_t. Without regularization, the noise part of the iterate $\widetilde{U}_t$ will keep accumulating. Adding mild regularization will balance the noise level at a stable level. Though it is possible to extend our analysis to deal with no regularization, the main message would remain the same. Therefore we considered a technically simpler setting.

[R2]: “whether it’s possible to identify a data distribution with only one type of features in which the large learning rate schedule still generalizes better than small learning rate ..."

• This is a great question for future work. We will speculate on an alternative perspective which relates to recent work on the NTK perspective of neural nets [Du et al., 2018, Li and Liang, 2018, Jacot et al., 2018][1]. With sufficient overparameterization and correct initialization, a small learning rate and no weight decay places the neural net in the NTK regime. In this regime, the generalization of the neural net will only be as good as that of a kernel method on random features. On the other hand, a larger learning rate could allow the optimization trajectory to depart the kernel regime and make the neural net features non-random. Recent work [Wei et al., 2018][2] has shown that NTK can have worse generalization than if the neural net features are allowed to be non-random.

[1] Jacot, A. et. al. “Neural Tangent Kernel: Convergence and Generalization in Neural Networks.”

[2] Wei, C. et. al. “Regularization Matters: Generalization and Optimization of Neural Nets v.s. their Induced Kernel.”