We thank the reviewers for the detailed and insightful reviews. As the reviewers noted, our work 1) contributes to "a deeper understanding of NTK and its limitations" and 2) develops novel analysis tools and techniques. We answer reviewers' questions below and will incorporate feedback into the final revision.

Reviewer 1:

—"there is no intuition provided how eqn 3.1 came about or what it means... explaining this will help readability."

Thank you for the valuable feedback on this section — we will incorporate this in our next revision. Equation 3.1 and the first term of Equation 3.2 are due to the formula for the Wasserstein gradient flow dynamics (see, e.g., [Santambrogio, 2017]), which are derived via continuous time steepest descent with respect to Wasserstein distance over the space of probability distributions on the neurons.

—"intuition/details of proof techniques for the optimization result in the main paper"

The intuition for the proof of Theorem 3.3 is that the optimization problem is convex over the space of probability distributions on neurons. We use this convexity to argue that if there is a descent direction, the uniform noise (U in equation 3.2) along with the 2-homogeneity will allow the optimization dynamics to increase the mass in this direction exponentially fast, which results in decrease of the loss by a polynomial amount. Note that though the problem is convex over the space of distributions, SGD on the network weights does not use the gradient with respect to the probability distribution so the convergence claim is not immediate.

— "Minor comment: Please explain somewhere why it is called "weak" regularization."

By weak regularization, we refer to the fact that $\lambda \to 0$ for our Theorem 4.1 to hold.

Reviewer 2:

— "might be better to discuss the challenges to extend to ReLU nets and the possible approaches."

The difficulty with ReLU networks is that if the gradient flow pushes neurons towards 0, issues of differentiability arise. One potential approach to circumvent this issue is arguing that with correct initialization, the iterates will never reach 0.

— "more interesting if the authors can show using a wider network can strictly improve the maximum margin achievable ... discuss a bit more ... gap of generalizations between narrow nets and wide nets."

It is possible to construct an instance where a narrower and wider network can both fit the training data, but the wider networks allow larger maximum normalized margin. Consider a distribution where the first two coordinates are the same as in our construction for Theorem 2.1, and the third coordinate is $\pm\epsilon$ with sign matching the label. Then a two neuron network separates this data with margin $\epsilon/2$ by using the third coordinate, whereas there exists a four neuron network – the one described in line 166 of the paper – that separates this data with margin $1/4$. We suspect that an intuition such as this can be used to prove a formal generalization gap (instead of only a gap of margins) based on width. This is an interesting direction for future work and we thank the reviewer for this suggestion.

Reviewer 3:

— "prove a similar result of Theorem 2.1 when the first k coordinates of the inputs are informative? And how the sample complexity depends on (k, d) as $k, d \to \infty$?"

It seems likely that some result in the form of Theorem 2.1 could be shown for some distribution where the first k input coordinates contain signal, although we have not fully explored this possibility at the moment. This is an interesting direction for future work and we thank the reviewer for this suggestion.

— "how do the sample complexity in Thm 2.1 change if adding a regularizer to NTK (i.e. ridge regression)?"

Theorem 2.1 will still apply in this setting. The optimizer of ridge regression with NTK will lie in the RKHS spanned by the data, and the sample complexity lower bound of Theorem 2.1 applies to all such functions.

— "possible to prove that for regularized *finite* width NN, SGD could learn distribution (2.1)?"

Empirically, we have found that SGD on finite width networks does indeed learn distribution (2.1). With over-parameterization, each of the learned neurons converges to one of the directions $e_1, -e_1, e_2, -e_2$. The question of rigorously proving that this behavior holds is an interesting and very challenging question for future work.

References

Filippo Santambrogio. {Euclidean, metric, and Wasserstein} gradient flows: an overview. *Bulletin of Mathematical Sciences*, 7(1):87–154, 2017.