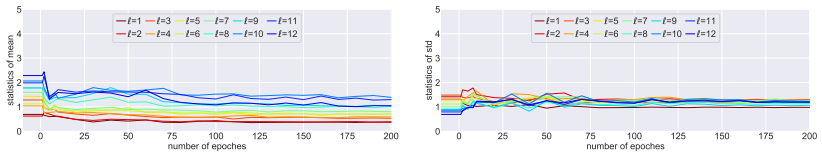

(a) (b)

Figure 1: The statistics of (a) mean and (b) variance of features in layer $\ell \in [12]$ during training.

We thank all the reviewers for their constructive comments. We appreciate their suggestions like add a table of notations (Reviewer #3) and provide more detailed discussion (Reviewer #2). We will take these suggestions in later version.

To Reviewer #1:

Q1. As the paper tries to argue that the feature distributions are similar for differently initialized networks, I wonder if this can be shown by using standard statistics such as mean and variance.

A1. Yes, we agree with you that when two finite networks are sampled from a common continuous CNN, their standard statistics would be similar. We report the difference of mean/variance of the feature functions in two different initialized networks θ_1 and θ_2 in Fig 1. We notice that the nets θ_1 and θ_2 at initialization are exactly sampled from a common continuous VGG-16 as we use the same initialization method, so the differences of mean/variance at initialization can be used as a baseline, i.e., we can compare the differences during training with this baseline to see whether the mean/variance are close. We adopt widely used statistics in hypothesis test as metrics to measure such difference. To be specific, we use $|\bar{X}-\bar{Y}|/\sqrt{2S_p^2/m}$ for mean (smaller means closer) and $\frac{1}{m-1}\sum_{i=1}^m (X_i-\bar{X})^2 / \frac{1}{m-1}\sum_{i=1}^m (Y_i-\bar{Y})^2$ for variance (closer to 1 means closer), where $S_p^2 = (\sum_{i=1}^m (X_i-\bar{X})^2 + \sum_{i=1}^m (Y_i-\bar{Y})^2)/(2m-2)$, $\{X_i\}_{i=1}^m$ and $\{Y_i\}_{i=1}^m$ are the m feature values for fixed input image in one layer of nets θ_1 and θ_2, respectively. We can see that the differences of both mean and variance during training are always at the same level with those at initialization (the parts where x-axis<0). Therefore, the results above indicate that mean and variance of the features in θ_1 and θ_2 are similar during the whole training process.

Moreover, we'd like to point out that the standard statistics mean and variance are not enough to differentiate two feature distributions. Besides the difficulty in choosing a proper threshold, there exist some networks, whose feature functions have close mean and variance but are essentially different.

To Reviewer #2:

Q1. The proposed method just does a change of variable where the new variables hide the non-convexity of the system.

A1. The reviewer might have missed some results in our appendix. Our convex formulation obtained by a change of variable is highly nontrivial. Below, we explain it from three aspects:

- The technique of reformulating two layer networks with respect to the distributions of hidden nodes was proposed in mean field theory based studies, such as [8] and [22], which have been widely accepted. This work is a nontrivial extension of this technique in multi layer neural networks.
- In appendix B, we explicitly present the connection between original formulation and convex reformulation in Theorem 2. It claims that for arbitrary point (w_*, p_*, u_*) in original formulation, if varying (w_*, u_*) cannot decrease the objective then the corresponding point $(\tilde{w}_*, p_*, \tilde{u}_*)$ after our proposed change of variable is a stationary point in the convex reformulation. This suggests that when CNN converges (GD only updates (w_*, u_*)), it converges to a stationary point of our convex reformulation. Therefore, our formulation can explain why one usually does not observe bad local minima when the widths are in the limit of infinity.
- We do obtain a convex reformulation for continuous CNN by changing the variables (w, u) in the original formulation into $(\tilde{w}, p, \tilde{u})$, but it should be noted that the whole system is essentially determined by $(p, \tilde{u})$ instead of $(\tilde{w}, p, \tilde{u})$ since given $(p, \tilde{u})$, $\tilde{w}$ could be calculated by minimizing the regularizer $\tilde{R}(\tilde{w}, p, \tilde{u})$ under the first two constraints in Eqn 7. Therefore, we do not hide any non-convexity in the new variable $\tilde{w}$.

To Reviewer #3:

Q1. More details about training process (whether to shuffle the training data) and whether we can obtain consistent results when the training data is shuffled and not shuffled in training.

A1. We use independent shuffling and data augmentation scheme in training each network in order to obtain the same test accuracy as previous work. To be specific, in training each network, we use the data augmentation used in previous work for training VGG nets (we mentioned in appendix D), which means that for each epoch, the training data was shuffled first and then standard data augmentation scheme is used (to be specific, the images are zero-padded with 4 pixels on each side, randomly cropped to produce 32×32 images, and horizontally mirrored with probability 0.5).