Taxonomizing local versus global structure in neural network loss landscapes

Anonymous Author(s)
Affiliation
Address
email

Abstract

Viewing neural network models in terms of their loss landscapes has a long history in the statistical mechanics approach to learning, and in recent years it has received attention within machine learning proper. Among other things, local metrics (such as the smoothness of the loss landscape) have been shown to correlate with global properties of the model (such as good generalization performance). Here, we perform a detailed empirical analysis of the loss landscape structure of thousands of neural network models, systematically varying learning tasks, model architectures, and quantity/quality of data. By considering a range of metrics that attempt to capture different aspects of the loss landscape, we demonstrate that the best test accuracy is obtained when: the loss landscape is globally well-connected; ensembles of trained models are more similar to each other; and models converge to locally smooth regions. We also show that globally poorly-connected landscapes can arise when models are small or when they are trained to lower quality data; and that, if the loss landscape is globally poorly-connected, then training to zero loss can actually lead to worse test accuracy. This suggests that the correlation between local smoothness and global model quality that has been observed in the past is due to the confounding factor that models are often chosen to be rather large and are typically trained to reasonably good quality data, rather than due to some fundamental property of the model itself. Based on these results, we develop a simple one-dimensional model with load and temperature-like parameters, and introduce the notion of an *effective loss landscape* depending on these parameters. When viewed through this lens, our detailed empirical results shed light on phases of learning (and related double descent behavior), determinants of good generalization, the role of load-like and temperature-like parameters in the learning process, the different influences on the loss landscape from model and data, and the relationships between local and global metrics, all topics of recent interest.

1 Introduction

Among the many approaches to understanding the behavior of neural network (NN) models, the study of their loss landscapes [1, 2] has proven to be particularly fruitful. Indeed, analyzing loss landscapes has helped shed light on the workings of many popular techniques, including large-batch training [3, 4], adversarial training [5], residual connections [6], and BatchNorm [7]. One particular concept of interest in these prior studies is the *sharpness* of local minima [3, 5, 8, 9]. While sharpness can be measured by first-order sensitivity measures, such as the Jacobian or Lipschitz constant, it is more appropriately measured by second-order sensitivity measures, typically via the Hessian spectrum. It has been suggested that NNs generalize well if they converge to a relatively flat local minimum.

While such local sharpness measures can provide insight, their focus on the local geometry of the loss landscape neglects the *global* structure of the loss landscape. Indeed, it is well-known that

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

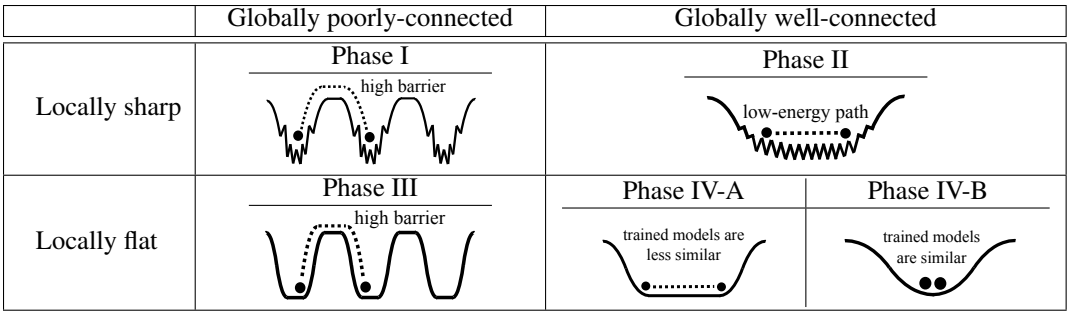


Figure 1: Globally well-connected versus globally poorly-connected, and locally sharp versus locally flat loss landscapes. Globally well-connected and locally flat landscape can be further divided into two sub-cases based on the similarity of trained models.

existing sharpness-based metrics can be altered (trivially) by reparameterization tricks or (more interestingly) by taking steps which have the effect of changing the local structures on the loss landscape [5, 8, 10]. For example, [5] shows that adversarial training can decrease the magnitude of Hessian eigenvalues and bias the model towards a locally smooth area, even though adversarial training can reduce clean test accuracy [11]. Similarly, [10] shows that Hessian eigenvalues become smaller with reduced ℓ_2 regularization, even though increased ℓ_2 regularization is known to reduce overfitting and improve training if used properly. More generally, general considerations would suggest (and indeed our own empirical results, reported in Figure 6 demonstrate) that by training to data with noisy labels, one can find models that generalize poorly, despite lying in very "flat" regions with small Hessian eigenvalues. These observations indicate that very local metrics like sharpness are insufficient to dictate larger-scale behavior of models, including their generalizability, especially when one is interested in finer-scale insight.

Motivated by these considerations, in this work, we are interested in understanding what we will call local properties / structure versus global properties / structure of the loss landscape of realistic NN models. While similar ideas underlie work that adopts a more explicitly statistical mechanics perspective [2, 12, 13], here we are interested in adopting a more operational machine learning perspective, where we employ metrics that have been used within machine learning as "experimental probes" to gain insight into local versus global properties. To do so, we employ the following metrics.

- First, we consider *Hessian-based metrics*, including the largest eigenvalue of the Hessian and the trace of the Hessian. These metrics capture *local curvature properties of the loss landscape* [14].
- Second, we use *mode connectivity* [15, 16]—in particular, the connectivity between trained models, or between a trained model and its random initialization—to try to *capture how well-connected different local minima* in the loss landscape are to each other.
- Third, we use *CKA similarity* [17] to *capture a correlation-like similarity* between the outputs of different trained models. Averaging the CKA over several pairs of models can be thought of as an approximation to the so-called overlap integrals frequently appearing in statistical mechanics [18].

We have considered many other metrics, but these seem to be particularly useful for identifying global structure versus local structure in loss landscapes. Informally, mode connectivity seems to capture the idea that if there is a single "rugged basin," then good models will be well-connected (i.e., low-energy / low-loss, potentially non-linear, paths through the loss landscape, i.e., continuous chains of models which all achieve a small loss value). We expect this property to be important since the connectivity of local minima indicates how efficiently the training dynamics can explore the loss landscape, without being stuck at saddle points or in a bad local minimum. Similarly, CKA similarity seems to capture the idea that if there is one / many "rugged basins," then an ensemble of good models will produce roughly similar outputs.

Here we briefly summarize our main contributions.

- We design an experimental setup based on two contol parameters, a *temperature-like* parameter that correlates with the magnitude of noise in SGD controlled during training, e.g., batch size (in most figures), learning rate, or weight decay, and a *load-like* parameter that measures the relationship between model size and data quantity and/or quality. By training thousands of models, under a

variety of settings of these parameters, and measuring local and global metrics of the loss landscape, we identify four distinct phases in temperature-load space, with sharp transitions between them.

- Using global connectivity (measured by mode connectivity) and local flatness (measured by Hessian), we taxonomize loss landscapes into four categories shown in Figure 1, labelled Phase I to Phase IV. For reasons observed in our empirical results in Section 3, it is often convenient to further divide Phase IV into two subcategories, based on whether or not the trained models produce similar representations (measured by CKA similarity). If the loss landscape satisfies the first property, we say it is *globally well-connected*; and if the loss landscape *also* satisfies the second property, we say it is *globally nice*.[1] Depending on whether the Hessian eigenvalues are large or small, we say the loss landscape is *locally sharp* or *locally flat*.
- We empirically demonstrate that the global structure of a loss landscape is well-correlated with good generalization performance, and that the best generalization occurs in the phase associated with a locally flat, and globally nice loss landscape. We demonstrate these results on a range of computer vision and natural language processing benchmarks (Cifar10, Cifar100, SVHN, and IWSLT 2016 De-En) and various models (ResNet, VGG, and Transformers). We also vary the amount of data by 10x, or vary the number of noisy labels, to study both the effect of the quantity of data and the quality of data on changing the loss landscape.
- Using a toy *bifrequency quadratic model*, we illustrate the main effects of varying temperature- and load-like parameters on an *effective* loss landscape, providing a simple demonstration of the shapes of Phases I–IV seen in Figure 1.

Computing connectivity and similarity requires comparing multiple distinct models. Compared to Hessian computations, this could be computationally expensive, especially if model training is expensive. For many reasonably-sized models, the metrics we consider are sufficiently tractable so as to be useful, e.g., during model training (although a full analysis of that is outside the scope of this paper). Moreover, both connectivity and similarity can be computed only from the training data and trained networks, providing non-trivial predictors of generalization performance.

The use of connectivity and similarity for studying the global structure of NN loss landscapes has motivations in classical spin-glass theory (which has been widely applied in studying NNs [12, 19–21], as well as neuroscience [22–24]); and it was inspired by [25], whose results suggest a *rugged convexity* in the NN loss landscape, as well as the concept of a *folding funnel* in the (statistical mechanics of) protein folding literature [26–29]. We discuss related work further in Section A.

2 Setup

Throughout this paper, we consider training a NN $f_\theta : \mathbb{R}^{d_{\text{in}}} \to \mathbb{R}^{d_{\text{out}}}$, with trainable parameters θ, to a dataset consisting of n datapoint/label pairs $S_{\text{train}} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$. Our training objective is to minimize a loss function of the form

$$\mathcal{L}(\theta) = \frac{1}{n} \sum_{(\boldsymbol{x},y)\in S_{\text{train}}} \ell(f_\theta(\boldsymbol{x}), y) + \lambda \|\theta\|_2^2. \tag{1}$$

Here ℓ is a loss function, typically chosen to be the cross entropy loss. The parameter λ is the weight decay parameter, which controls the level of ℓ_2 regularization. We consider optimizing networks using standard minibatch SGD, which uses iterates of the form

$$\theta \leftarrow \theta - \eta \tilde{\boldsymbol{g}}(\theta), \qquad \tilde{\boldsymbol{g}}(\theta) = \frac{1}{B} \sum_{j=1}^{B} \nabla_\theta \ell(f_\theta(\boldsymbol{x}_{i_j}), y_{i_j}), \tag{2}$$

where η is the learning rate, $0 < B \leq n$ is the batch size and the indices $i_1, \ldots, i_B$ of each minibatch are sampled without replacement from $\{1, \ldots, n\}$. For classification tasks, we consider also the training/testing accuracy, which is simply the fraction of correctly classified points $\text{acc}_{\text{train}}(\theta) = n^{-1} \sum_{(\boldsymbol{x},y)\in S_{\text{train}}} \mathbf{1}(f_\theta(\boldsymbol{x}) = y)$, and similarly for $\text{acc}_{\text{test}}(\theta)$ on a given test set S_{test}. The Hessian at a given point θ_0 in parameter space is represented by the matrix $\nabla_\theta^2 \mathcal{L}(\theta_0)$. To summarize the

[1] Our empirical results show that a loss landscape can generate dissimilar models while being globally well-connected; but we do not observe a loss landscape that generates similar models but is globally poorly-connected.

Hessian in a single scalar value, we report the dominant eigenvalue $\lambda_{\max}(\nabla^2_\theta \mathcal{L}(\theta_0))$ and/or the trace $\mathrm{tr}(\nabla^2_\theta \mathcal{L}(\theta_0))$, calculated using the `PyHessian` software [14].

We now briefly introduce the main metrics and control parameters which we will consider in this paper. Due to space constraints, we defer further details to Section B in the Appendix.

Temperature and load. In the sequel, a **load parameter** of a loss landscape refers to some quantity related to the amount and/or quality of data, relative to the size of the model. Specifically, we vary either i) model size (e.g. width), for fixed training set size n, ii) training set size n, for fixed model size, or iii) the "quality" of training data, which is varied by randomizing a fraction α of the training labels. Each of these control parameters *directly* induce a different loss landscape by changing the data S_{train} and/or architecture f_θ for which the loss $\mathcal{L}(\theta)$ is being computed. For example, we expect that increasing width will result in a smoother loss landscape [30]; we shall see this effect with CKA similarity in the transition from Phase IV-A to IV-B.

The second control parameter we vary in our experiments is **temperature**, representing the amount of noise introduced in the SGD iterates (2). Most commonly we take this to be the batch size B, though we will also use the learning rate η and the weight decay parameter λ. Increasing temperature corresponds to smaller batch size, and large learning rate or weight decay. Varying the temperature does not directly define a different loss function $\mathcal{L}(\theta)$, but rather *indirectly* induces a different *effective* loss function. This is because, at different temperatures, the iterates of SGD concentrate on different regions of the loss-landscape. Later, in Section 4, we formalize this notion by considering the stationary distributions of a Langevin model for SGD with different levels of noise.

CKA similarity. To measure the similarity of two NN representations, we use the *centered kernel alignment* (CKA) metric proposed in [17]. For a NN f_θ, let $F_\theta = [f_\theta(\boldsymbol{x}_1) \quad \cdots \quad f_\theta(\boldsymbol{x}_m)]^\top \in \mathbb{R}^{m \times d_{\text{out}}}$ denote the concatenation of the outputs[2] of the network over a set of m randomly sampled datapoints. Then the (linear) CKA similarity between two parameter configurations θ, θ' is given by

$$s(\theta, \theta') = \frac{\mathrm{Cov}(F_\theta, F_{\theta'})}{\sqrt{\mathrm{Cov}(F_\theta, F_\theta)\mathrm{Cov}(F_{\theta'}, F_{\theta'})}}, \tag{3}$$

where for $X, Y \in \mathbb{R}^{m \times d}$, we define $\mathrm{Cov}(X, Y) = (m-1)^{-2}\mathrm{tr}(XX^\top H_m YY^\top H_m)$, and $H_m = I_m - m^{-1}\mathbf{1}\mathbf{1}^\top$ is the centering matrix. The CKA similarity was studied thoroughly in [17], and is known to be an effective way to compare the representations learned by two different trained NNs. In addition, rather than computing the similarity directly on the original training points, we measure CKA on a perturbed training set comprised of Mixup samples [31]; this can reduce trivial similarity that occurs when the models are trained to zero training error. See also Section B.4.1 for the ablation study on different perturbed training sets.

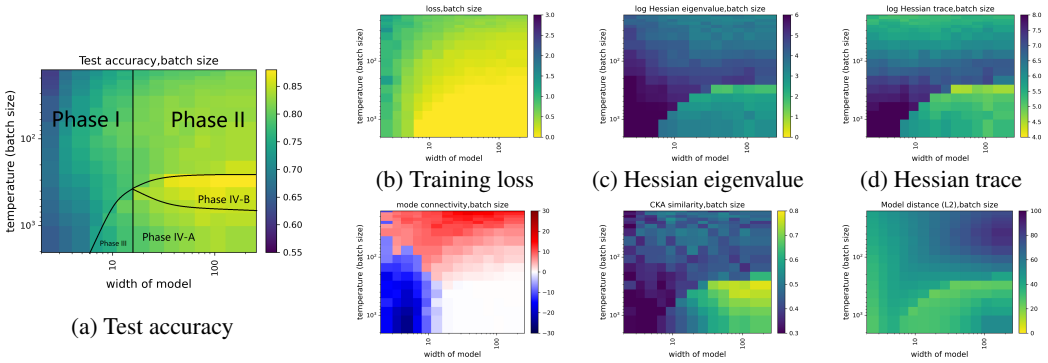
Mode connectivity. For two parameter configurations θ, θ', computing mode connectivity involves finding a *low-energy curve* $\gamma(t)$, $t \in [0, 1]$, for which $\gamma(0) = \theta, \gamma(1) = \theta'$, such that $\int \mathcal{L}(\gamma(t))dt$ is minimized [15, 16]. A number of techniques have been proposed to find such curves γ; in this work, we use the technique proposed in [15], which parameterizes the Bezier curve with $k+1$ bends, given by $\gamma_\phi(t) = \sum_{j=0}^k \binom{k}{j}(1-t)^{k-j}t^j\theta_j$ for $t \in [0, 1]$, where $\theta_0 = \theta, \theta_k = \theta'$, and $\phi = \{\theta_1, \ldots, \theta_{k-1}\}$ are trainable parameters of additional models, defining "bends" on the curve $\gamma_\phi(t)$. In this paper, we use Bezier curves with three bends ($k = 2$). Given the curve $\gamma_\phi(t)$, we define the mode connectivity of the models θ, θ' to be

$$\mathsf{mc}(\theta, \theta') = \mathcal{L}(\gamma_\phi(t^*)) - \frac{1}{2}(\mathcal{L}(\theta) + \mathcal{L}(\theta')), \tag{4}$$

where t^* maximizes the deviation $t \mapsto |\mathcal{L}(\gamma_\phi(t)) - \frac{1}{2}(\mathcal{L}(\theta) + \mathcal{L}(\theta'))|$. If $\mathsf{mc}(\theta, \theta') > 0$, then there is a "barrier" of high loss between θ, θ', while $\mathsf{mc}(\theta, \theta') \leq 0$ implies a curve of low loss connecting θ, θ', which typically occurs if training failed to locate a reasonable optimum. We provide additional details of this procedure, as well as an ablation study on different mode connectivity hyperparameters in Section B.4.2.

ℓ_2 distance. We will also occasionally report the ℓ_2 distance between two parameter configurations $\|\theta - \theta'\|_2$ as a measure of similarity between models, though we typically find that the CKA similarity is a more informative measure.

[2] For this work, we focus on the similarity of representations at the output layer, i.e. after the softmax is applied, though the CKA similarity can be used to compare the representations at any layer.



(a) Test accuracy (b) Training loss (c) Hessian eigenvalue (d) Hessian trace

(e) Mode connectivity (f) CKA similarity (g) ℓ_2 distance

Figure 2: Metrics partitioning the load-temperature diagram into different learning phases, using batch size as temperature. Models are trained with ResNet18 on CIFAR-10. All plots are on the same set of axes.

3 Empirical results on taxonomizing local versus global structure

Experiment setup. We consider ResNets [32] trained on CIFAR-10 [33] as the standard setting to demonstrate different loss landscapes. We will scale the network width to change the size of the network. For ResNet18 which contains four major blocks with channel width $\{k, 2k, 4k, 8k\}$, we select different values of k to obtain ResNets with different widths. In the standard setting, batch size, learning rate, and weight decay is kept constant throughout training to study interactions between temperature parameters and the loss landscape. Then, we apply learning rate decay in a separate experiment. More details on the experimental setup can be found in Section C.

3.1 Types of loss landscapes and phase transitions

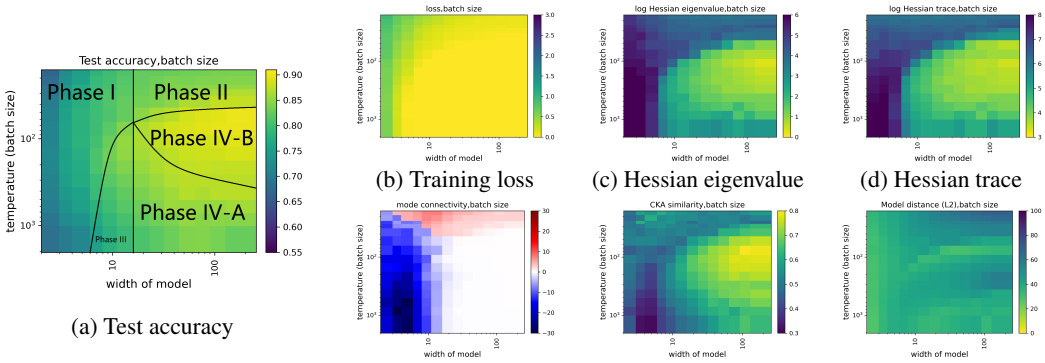
In this section, we highlight the presence of globally nice, globally well-connected/poorly-connected, and locally flat/sharp loss landscapes, and the phase transitions which separate them. In addition to test accuracy, results on six other metrics are presented, including training loss, leading Hessian eigenvalue, trace of Hessian, CKA similarity, mode connectivity, and ℓ_2 distance measured between model weights. For each metric, the results are presented in a 2D diagram, in which the horizontal dimension is the load, and the vertical dimension is the temperature. In this subsection, we discuss our standard setting, in which we vary width as the load parameter and batch size as the temperature — the results are displayed in Figure 2. Each pixel represents a specific training configuration tuple (width, batch size), averaged over five independent runs. There are two phase transitions that separate each plot into four primary regions:

1. Hessian distinguishes locally sharp and locally flat loss landscapes. The first phase transition is displayed in Figure 2(c) and 2(d), separating Phase I/II from Phase III/IV. A large Hessian eigenvalue or Hessian trace (darker color) represents a sharp local loss landscape [5, 14]. From the training loss shown on Figure 2(b), we find this transition coincides with a significant decrease in the training loss. Comparing Figures 2(a) and 2(c)-(d), categorizing loss landscapes based solely on the Hessian is insufficient to predict test accuracy.

2. Mode connectivity distinguishes globally well-connected and globally poorly-connected loss landscapes. The second phase transition is shown in Figure 2(e). The white region represents near-zero mode connectivity which, according to our definition, implies a flat curve in the loss landscape between freshly-trained weights. The loss along individual mode connectivity curves are presented in Section B.5 in the Supplementary Material. In contrast to training loss, test accuracy only appears to show significant improvements after this transition. In particular, well-connected loss landscapes improve the test accuracy one can obtain with suitable choice of temperature.This phase transition forms a vertical line separating Phase I from II and separates Phase III from IV.

Based on the two transitions, we now classify the loss landscapes into the following phases.

- **Phase I: Globally poorly-connected and locally sharp.** Training loss is high, Hessian eigenvalue and trace are large, and mode connectivity is poor due to the presence of large barriers in the landscape.



(a) Test accuracy (b) Training loss (c) Hessian eigenvalue (d) Hessian trace

(e) Mode connectivity (f) CKA similarity (g) ℓ_2 distance

Figure 3: Metrics partitioning the load-temperature phase diagram, using learning rate as temperature.

- **Phase II: Globally well-connected and locally sharp.** Training loss is high, Hessian eigenvalue and trace are large, and mode connectivity is poor due to a failure in training to localize about local minima.
- **Phase III: Globally poorly-connected and locally flat.** Training loss is small, Hessian eigenvalue and trace are small, yet mode connectivity still remains poor.
- **Phase IV: Globally well-connected and locally flat.** Training loss is small, Hessian eigenvalue and trace are small, and mode connectivity is good.

CKA further distinguishes two subcategories in Phase IV. From Figure 2(f), CKA can be used to further divide Phase IV into Phase IV-A and Phase IV-B, with the latter exhibiting larger CKA similarity.

Simple ℓ_2 distance is not enough. A major difficulty in measuring similarity between models is that the same model can be realized using different weights [8]. To reconcile this effect, the distance between two models is commonly defined in terms of their predictions instead of weights. Indeed, the representation-based CKA similarity is seen to be preferable to the weights-based ℓ_2 distance. From Figure 2(g), the ℓ-2 distance provides some limited information, but is not as informative as the CKA similarity.

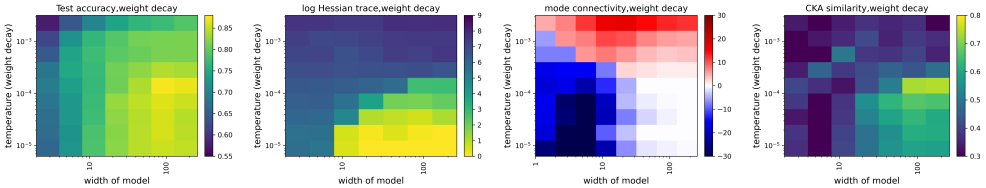
From the discussions above, we assert the following central claim of this work: **optimal test accuracy is obtained when the loss landscape is globally nice and the trained model converges to a locally flat region.** Note that both similarity and connectivity are required for a globally nice loss landscape. Phase IV-B is precisely the region with *globally nice* landscapes, exhibiting the highest test accuracies.

Training with learning rate decay. Next, we consider the same phase diagram, now with learning rate decay, which relaxes the constraint of constant temperature. The results are presented in Figure 3. Comparing Figure 3 with Figure 2, we see that the four phases are still present and the test accuracy is maximized when the loss landscape is globally nice and locally flat. Therefore, our central claim is unaffected by learning rate schedule. In Figure 3(c) and Figure 3(d), smaller temperatures in Phase IV-A appear to increase the size of the Hessian. This is a well-known issue with large-batch training [3]. Finally, note that the optimal test accuracy achieved improves in the presence of learning rate decay. See Section D in the Supplementary Material for further discussions on learning rate decay.

3.2 Ablation study

In this subsection, we present results on additional temperature parameters and load parameters.

Different temperature parameters. First, we study weight decay as an alternative temperature parameter, in addition to batch size. Results are shown in Figure 4, and are similar to those seen in Figure 2. Once again, best test accuracy is obtained when the loss landscape is both globally nice and locally flat. Another observation with Figure 4(b): when training a wide model with small weight decay (bottom right), the Hessian trace becomes extremely small. This matches observations in [10] that decreasing weight decay reduces the size of the Hessian. Since weight decay is known to improve generalization, this also demonstrates that local metrics alone are insufficient to predict test performance.



(a) Test accuracy (b) Hessian trace (c) Mode connectivity (d) CKA similarity

Figure 4: Metrics partitioning the load-temperature phase diagram, using weight decay as temperature.

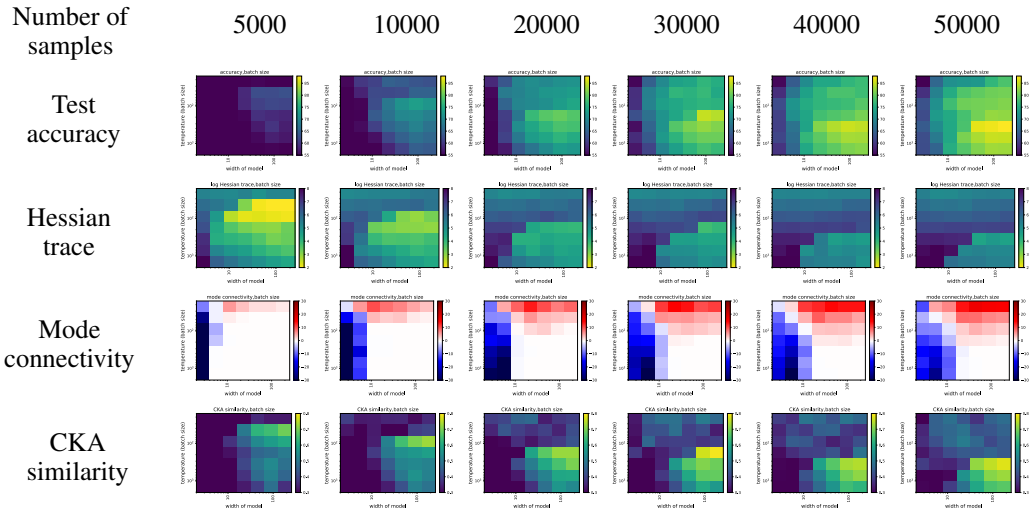


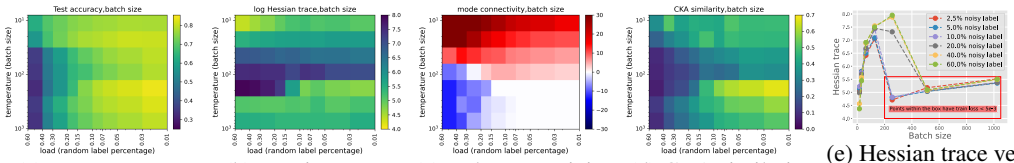
Figure 5: Metrics on the load-temperature phase diagram, using batch size as temperature, varying quantities of training data from CIFAR-10.

Different amount of training data. Next, we study how the amount of training data (as another load parameter) affects our results. We vary the number of training samples in CIFAR-10 by 10x. Results are shown in Figure 5. Again, optimal test accuracy is achieved when Hessian eigenvalue and trace are small, mode connectivity is good, and CKA similarity is large. Better test accuracy is achieved with more data. Here, CKA provides useful complementary information to the Hessian and mode connectivity for explaining the utility of large data. The Hessian alone cannot predict the correct trend as it *increases* in magnitude with data. Mode connectivity alone cannot predict the correct trend either, becoming increasingly poor with larger data (see the shrinking white region). Indeed, it appears that larger models are required to keep the loss landscape well-connected with increasing data. In contrast, CKA precisely captures the relationship of increasing test accuracy with additional data.

These observations also imply that the utilities of extra data and larger models are different. Larger models can increase connectivity in the loss landscape (e.g. Figure 2(e)), while increasing data boosts signal in the landscape, enabling trained models to become similar to each other.

Different quality of data by changing the amount of noisy labels. Next, we replace the load parameter with a proportion of noisy labels to simulate the change in the *quality* of data. To generate noisy labels, a percentage of the training data is randomly selected and altered to an incorrect target class. Results are shown in Figure 6. Once again, local information alone fails to measure the quality of training data. We can see that training with a large amount of noise does not significantly affect the Hessian — see Figure 6(b). In particular, as the temperature decreases (down to the bottom of Figure 6(b)), the Hessian becomes smaller, independent of the quantity of noisy labels. This is especially evident in Figure 6(e), where we plot Hessian trace against batch size. However, looking instead at mode connectivity in Figure 6(c) and CKA in Figure 6(d), one can easily deduce that training with more noisy labels leads to more poorly-connected loss landscapes.

Additional results on different datasets, network architectures, load/temperature parameters, and training schemes in the supplementary material. In Section E in the Supplementary Material,



(a) Test accuracy (b) Hessian trace (c) Md. connectivity (d) CKA similarity (e) Hessian trace versus batch size

Figure 6: Metrics on the load-temperature phase diagram, using batch size as temperature, varying the proportion of random labels. (e) shows that the Hessian changes slowly with the proportion of noisy labels when training loss is small.

we cover additional datasets, including SVHN, CIFAR-100, and IWSLT 2016 German to English (De-En) (a machine translation dataset), as well as additional NN architectures, including VGG11 and Transformers. All experiments support our conclusions.

In Section E.3, we study an analogous plot to Figure 2, training with 10% noisy labels. Interestingly, we observe double-descent [34–36], which can be viewed naturally as "bad fluctuations" between different learning phases [37, 38]. Using this experiment, we infer that the decision to train to zero loss (traditionally a rule-of-thumb in computer-vision tasks) should depend on the global connectivity of the loss landscape. Indeed, for small models with poor connectivity, we find that training to zero loss can harm test accuracy. This suggests that the common wisdom to fit training data to zero loss is derived from experiments involving high-quality data and models. Furthermore, in Section E.4, we show that in the setting of machine translation, the loss landscape remains poorly-connected even for a large embedding dimension (Figure 17). In this case, generalization can be quite poor when training to zero loss. This conclusion matches (with hindsight) the observations in practice, e.g., dropout and early stopping can improve test loss [39, 40].

In addition, in Section E.5, we study learning rate as an alternative temperature parameter, which produces analogous results to Figure 2. In Section E.6.1, we study large-batch training and show that it increases local sharpness. Note that for most experiments, we intentionally keep a constant learning rate when varying the batch size to study the change in the landscape with a changing temperature. Thus, in Section E.6.2, we provide additional results on tuning learning rate with changing batch size, including the commonly used "linear scaling rule" [41].

4 Temperature and the effective loss landscape

Thus far we have argued that varying temperature (e.g. batch size) induces loss landscapes with different properties. However, the loss function $\mathcal{L}(\theta)$ is defined independently of these quantities, rendering this relationship unclear. Using a simple toy model, we illustrate how varying the temperature concentrates the optimizer on different regions of the loss landscape, inducing an *effective* loss landscape, exhibiting the properties seen in Figure 1. To begin, consider the typical Langevin diffusion model for SGD dynamics (2):

$$\mathrm{d}\Theta_t = -\nabla\mathcal{L}(\Theta_t)\mathrm{d}t + \sigma\sqrt{\eta}\cdot\mathrm{d}W_t, \tag{5}$$

where W_t denotes Brownian motion, σ^2 represents the variance in the stochastic gradient (decreasing with batch size) and η is the step-size from (2). The unique stationary distribution of this process is a Gibbs distribution of the form $p_T(\theta) \propto e^{-\mathcal{L}(\theta)/T}$ (when it exists), where $T = \sqrt{\eta\sigma^2/2}$ is referred to as the *temperature*. As expected, the temperature increases proportionally to the learning rate and inversely to the batch size. As $T \to 0^+$, the Gibbs distribution p_T concentrates on the set $\Omega_0 = \arg\min_\theta \mathcal{L}(\theta)$. On the other hand, as $T \to \infty$, p_T becomes increasingly uniform. Therefore, our choice of temperature should influence the regions of the loss landscape explored. To formalize this concept, recall that the quantile function $Q(x)$ (the pseudoinverse of the distribution function) of a random variable X is the unique transformation sending uniform random variables U on $[0,1]$ to X. In particular, for any function ϕ, $\mathbb{E}\phi(X) = \int_0^1 \phi(Q(x))\mathrm{d}x$. Letting $Q_T^1, \ldots, Q_T^d$ denote the quantile functions of the marginal distributions of p_T, we define the *effective loss landscape* as the function $\mathcal{L}_{\text{eff}} : [0,1]^d \to \mathbb{R}$ with $\mathcal{L}_{\text{eff}}(u_1, \ldots, u_d) = \mathcal{L}(Q_T^1(u_1), \ldots, Q_T^d(u_d))$. In the case where $\mathcal{L}$ is replaced by the joint distribution, we recover the copula associated with p_T. In general, this formulation will fail to capture details about the correlations in p_T, so we restrict our attention to a one-dimensional model, which will suffice for basic qualitative purposes.

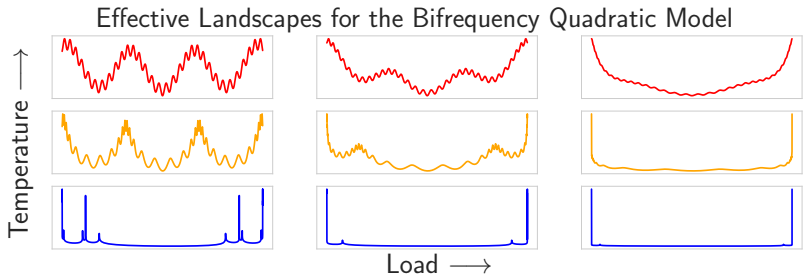


Figure 7: Illustration of the effective landscapes for the bifrequency quadratic model (6). Here we take $\lambda_1 = 10, \lambda_2 = 3, \gamma_1 = 0.5$, and $\gamma_2 = 4.5$ and vary $T \in \{100, 10, 1\}$ and $a \in \{0.01, 0.1, 0.5\}$.

Our one-dimensional model of choice is the following *bifrequency quadratic model*:

$$\mathcal{L}(\theta) = a\theta^2 + \lambda_1(1 - \cos(\gamma_1\theta)) + \lambda_2(1 - \cos(\gamma_2\theta)), \tag{6}$$

where $a, \lambda_1, \lambda_2, \gamma_1, \gamma_2 > 0$. As usual, $p_T(\theta) \propto e^{-\mathcal{L}(\theta)/T}$ denotes the associated Gibbs distribution. The two cosine waves here describe noise on both global and local scales, so $\lambda_1 \gg \lambda_2$ and $\gamma_1 \ll \gamma_2$. The a parameter is used to describe *load*, quantifying the amount of signal-to-noise in the landscape. Indeed, as a becomes large, the quadratic term dominates, while for small a, $\mathcal{L}(\theta)$ resembles a two-frequency cosine wave. In Figure 7, we see that as we decrease the temperature, the corresponding effective loss landscape $\mathcal{L}_{\text{eff}}$ becomes smoother, analogous to the transitions from Phase I to Phase III and/or Phase II to Phase IV in Figure 1.

5 Conclusions

Generalization is a complex issue, and many explanations coexist. In this paper, we try to complement existing sharpness-based arguments by measuring the global structure through connectivity and similarity, in addition to the local sharpness-based metrics. Here, we summarize a few observations from our connectivity and similarity plots.

- **A larger width improves connectivity:** From Figure 2(e), 3(e), 4(c), 13(c), 15(c), 16(e), and 18(c) (see Supplementary Materials), we see that increasing model width improves connectivity.
- **More data improves similarity:** From the CKA-similarity row in Figure 5, we see that increasing the amount of data can improve similarity.
- **Better data quality improves connectivity:** From Figure 6(c), we see that reducing the amount of noisy labels improves connectivity.
- **A larger width and a higher temperature in Phase IV improves similarity:** From Figure 2(f), 3(f), 4(d), 13(d), 15(d), 16(f), and 18(d), we see that (1) a larger width increases similarity of trained models; and (2) for Phase IV (globally well-connected and locally flat minima), using a relatively large temperature can improve similarity. From our analysis in Section 4, changing temperature does not directly change the loss landscape, but it implicitly changes the stationary distribution of SGD training, which "effectively" changes the loss landscape according to the shapes in Figure 1.

The global properties of a loss landscape are more challenging to measure than the local ones, so we resort to large-scale experiments and aim for a thorough study. Our empirical evaluation covers various data, architectures, and training schemes. In addition, we provide theoretical explanations on interactions between SGD stationary distribution and temperature, which characterizes the influence of the training schemes on the loss landscape. We stress that we do not support the replacement of sharpness-based metrics with global metrics. The inadequacy of local metrics arise since the Hessian is only a subset of the relevant factors (effectively, a Simpson's paradox [42]). Instead, by considering mode connectivity and CKA in conjunction with local metrics, we can address their deficiencies. Here, we focus only on loss landscapes derived from training data, following the convention of avoiding the use of test data. However, in certain areas such as domain adaptation and transfer learning, there is a non-negligible distributional gap between training and test, which necessitates a characterization of the difference between training and test loss landscapes. In future work, we aim to look at phase diagrams outside of the load/temperature form, especially in the low-connectivity regime, which is most challenging according to our taxonomy.

References

[1] Hao Li, Zheng Xu, Gavin Taylor, Christoph Studer, and Tom Goldstein. Visualizing the loss landscape of neural nets. In *Advances in Neural Information Processing Systems*, pages 6389–6399, 2018.

[2] Andrew J Ballard, Ritankar Das, Stefano Martiniani, Dhagash Mehta, Levent Sagun, Jacob D Stevenson, and David J Wales. Energy landscapes for machine learning. *Physical Chemistry Chemical Physics*, 19(20):12585–12603, 2017.

[3] Nitish Shirish Keskar, Dheevatsa Mudigere, Jorge Nocedal, Mikhail Smelyanskiy, and Ping Tak Peter Tang. On large-batch training for deep learning: Generalization gap and sharp minima. *arXiv preprint arXiv:1609.04836*, 2016.

[4] Zhewei Yao, Amir Gholami, Kurt Keutzer, and Michael W. Mahoney. Large batch size training of neural networks with adversarial training and second-order information. Technical Report Preprint: arXiv:1810.01021, 2018.

[5] Zhewei Yao, Amir Gholami, Qi Lei, Kurt Keutzer, and Michael W Mahoney. Hessian-based analysis of large batch training and robustness to adversaries. *arXiv preprint arXiv:1802.08241*, 2018.

[6] Yuanzhi Li and Yang Yuan. Convergence analysis of two-layer neural networks with relu activation. *arXiv preprint arXiv:1705.09886*, 2017.

[7] Shibani Santurkar, Dimitris Tsipras, Andrew Ilyas, and Aleksander Madry. How does batch normalization help optimization? *arXiv preprint arXiv:1805.11604*, 2018.

[8] Laurent Dinh, Razvan Pascanu, Samy Bengio, and Yoshua Bengio. Sharp minima can generalize for deep nets. In *International Conference on Machine Learning*, pages 1019–1028. PMLR, 2017.

[9] Behnam Neyshabur, Srinadh Bhojanapalli, and Nathan Srebro. A pac-bayesian approach to spectrally-normalized margin bounds for neural networks. *arXiv preprint arXiv:1707.09564*, 2017.

[10] Diego Granziol. Flatness is a false friend. *arXiv preprint arXiv:2006.09091*, 2020.

[11] Dimitris Tsipras, Shibani Santurkar, Logan Engstrom, Alexander Turner, and Aleksander Madry. Robustness may be at odds with accuracy. *arXiv preprint arXiv:1805.12152*, 2018.

[12] Andreas Engel and Christian P. L. Van den Broeck. *Statistical mechanics of learning*. Cambridge University Press, New York, NY, USA, 2001.

[13] Charles H Martin and Michael W Mahoney. Implicit self-regularization in deep neural networks: Evidence from random matrix theory and implications for learning. *arXiv preprint arXiv:1810.01075*, 2018.

[14] Zhewei Yao, Amir Gholami, Kurt Keutzer, and Michael W Mahoney. Pyhessian: Neural networks through the lens of the hessian. In *2020 IEEE International Conference on Big Data (Big Data)*, pages 581–590. IEEE, 2020.

[15] Timur Garipov, Pavel Izmailov, Dmitrii Podoprikhin, Dmitry Vetrov, and Andrew Gordon Wilson. Loss surfaces, mode connectivity, and fast ensembling of dnns. *arXiv preprint arXiv:1802.10026*, 2018.

[16] Felix Draxler, Kambis Veschgini, Manfred Salmhofer, and Fred Hamprecht. Essentially no barriers in neural network energy landscape. In *International conference on machine learning*, pages 1309–1318. PMLR, 2018.

[17] Simon Kornblith, Mohammad Norouzi, Honglak Lee, and Geoffrey Hinton. Similarity of neural network representations revisited. In *International Conference on Machine Learning*, pages 3519–3529. PMLR, 2019.

[18] Madhu Advani, Subhaneil Lahiri, and Surya Ganguli. Statistical mechanics of complex neural systems and high dimensional data. *Journal of Statistical Mechanics: Theory and Experiment*, 2013(03):P03014, mar 2013.

[19] Daniel J Amit, Hanoch Gutfreund, and Haim Sompolinsky. Storing infinite numbers of patterns in a spin-glass model of neural networks. *Physical Review Letters*, 55(14):1530, 1985.

[20] Anna Choromanska, Mikael Henaff, Michael Mathieu, Gérard Ben Arous, and Yann LeCun. The loss surfaces of multilayer networks. In *Artificial intelligence and statistics*, pages 192–204. PMLR, 2015.

[21] Elena Agliari, Adriano Barra, Andrea Galluzzi, Daniele Tantari, and Flavia Tavani. A walk in the statistical mechanical formulation of neural networks. *arXiv preprint arXiv:1407.5300*, 2014.

[22] Mark C Fuhs and David S Touretzky. A spin glass model of path integration in rat medial entorhinal cortex. *Journal of Neuroscience*, 26(16):4266–4276, 2006.

[23] Anthony G Hudetz, Colin J Humphries, and Jeffrey R Binder. Spin-glass model predicts metastable brain states that diminish in anesthesia. *Frontiers in systems neuroscience*, 8:234, 2014.

[24] Ibon Recio and Joaquín J Torres. Emergence of low noise frustrated states in e/i balanced neural networks. *Neural Networks*, 84:91–101, 2016.

[25] Charles H Martin and Michael W Mahoney. Rethinking generalization requires revisiting old ideas: statistical mechanics approaches and complex learning behavior. *arXiv preprint arXiv:1710.09553*, 2017.

[26] Joseph D Bryngelson and Peter G Wolynes. Spin glasses and the statistical mechanics of protein folding. *Proceedings of the National Academy of sciences*, 84(21):7524–7528, 1987.

[27] Piotr Garstecki, Trinh Xuan Hoang, and Marek Cieplak. Energy landscapes, supergraphs, and "folding funnels" in spin systems. *Physical Review E*, 60(3):3219, 1999.

[28] Konstantin Klemm, Christoph Flamm, and Peter F Stadler. Funnels in energy landscapes. *arXiv preprint arXiv:0707.3719*, 2007.

[29] Charles L Brooks, José N Onuchic, and David J Wales. Taking a walk on a landscape. *Science*, 293(5530):612–613, 2001.

[30] Brady Neal, Sarthak Mittal, Aristide Baratin, Vinayak Tantia, Matthew Scicluna, Simon Lacoste-Julien, and Ioannis Mitliagkas. A modern take on the bias-variance tradeoff in neural networks. *arXiv preprint arXiv:1810.08591*, 2018.

[31] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

[32] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[33] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[34] Mikhail Belkin, Daniel Hsu, Siyuan Ma, and Soumik Mandal. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proceedings of the National Academy of Sciences*, 116(32):15849–15854, 2019.

[35] Mikhail Belkin, Daniel Hsu, and Ji Xu. Two models of double descent for weak features. *SIAM Journal on Mathematics of Data Science*, 2(4):1167–1180, 2020.

[36] Preetum Nakkiran, Gal Kaplun, Yamini Bansal, Tristan Yang, Boaz Barak, and Ilya Sutskever. Deep double descent: Where bigger models and more data hurt. *arXiv preprint arXiv:1912.02292*, 2019.

[37] Zhenyu Liao, Romain Couillet, and Michael W Mahoney. A random matrix analysis of random fourier features: beyond the gaussian kernel, a precise phase transition, and the corresponding double descent. *arXiv preprint arXiv:2006.05013*, 2020.

[38] Michał Dereziński, Feynman Liang, and Michael W Mahoney. Exact expressions for double descent and implicit regularization via surrogate random design. *arXiv preprint arXiv:1912.04533*, 2019.

[39] Antonio Valerio Miceli Barone, Barry Haddow, Ulrich Germann, and Rico Sennrich. Regularization techniques for fine-tuning in neural machine translation. *arXiv preprint arXiv:1707.09920*, 2017.

[40] Yarin Gal and Zoubin Ghahramani. A theoretically grounded application of dropout in recurrent neural networks. *Advances in neural information processing systems*, 29:1019–1027, 2016.

[41] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017.

[42] Steven A Julious and Mark A Mullee. Confounding and simpson's paradox. *Bmj*, 309(6967):1480–1481, 1994.

[43] Chen Xing, Devansh Arpit, Christos Tsirigotis, and Yoshua Bengio. A walk with sgd. *arXiv preprint arXiv:1802.08770*, 2018.

[44] Bobby Kleinberg, Yuanzhi Li, and Yang Yuan. An alternative view: When does sgd escape local minima? In *International Conference on Machine Learning*, pages 2698–2707. PMLR, 2018.

[45] Charles H Martin and Michael W Mahoney. Traditional and heavy-tailed self regularization in neural network models. *arXiv preprint arXiv:1901.08276*, 2019.

[46] Charles H Martin, Michael W Mahoney, et al. Predicting trends in the quality of state-of-the-art neural networks without access to training or testing data. *arXiv preprint arXiv:2002.06716*, 2020.

[47] Umut Simsekli, Levent Sagun, and Mert Gurbuzbalaban. A tail-index analysis of stochastic gradient noise in deep neural networks. In *International Conference on Machine Learning*, pages 5827–5837. PMLR, 2019.

[48] Liam Hodgkinson and Michael W Mahoney. Multiplicative noise and heavy tails in stochastic optimization. *arXiv preprint arXiv:2006.06293*, 2020.

[49] Stanislav Fort, Gintare Karolina Dziugaite, Mansheej Paul, Sepideh Kharaghani, Daniel M Roy, and Surya Ganguli. Deep learning versus kernel learning: an empirical study of loss landscape geometry and the time evolution of the neural tangent kernel. *arXiv preprint arXiv:2010.15110*, 2020.

[50] Ruoyu Sun, Dawei Li, Shiyu Liang, Tian Ding, and Rayadurgam Srikant. The global landscape of neural networks: An overview. *IEEE Signal Processing Magazine*, 37(5):95–108, 2020.

[51] Yann Dauphin, Razvan Pascanu, Caglar Gulcehre, Kyunghyun Cho, Surya Ganguli, and Yoshua Bengio. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. *arXiv preprint arXiv:1406.2572*, 2014.

[52] Rong Ge, Furong Huang, Chi Jin, and Yang Yuan. Escaping from saddle points—online stochastic gradient for tensor decomposition. In *Conference on learning theory*, pages 797–842. PMLR, 2015.

[53] Simon S Du, Chi Jin, Jason D Lee, Michael I Jordan, Barnabas Poczos, and Aarti Singh. Gradient descent can take exponential time to escape saddle points. *arXiv preprint arXiv:1705.10412*, 2017.

[54] Chi Jin, Rong Ge, Praneeth Netrapalli, Sham M Kakade, and Michael I Jordan. How to escape saddle points efficiently. In *International Conference on Machine Learning*, pages 1724–1732. PMLR, 2017.

[55] Itay Safran and Ohad Shamir. Spurious local minima are common in two-layer relu neural networks. In *International Conference on Machine Learning*, pages 4433–4441. PMLR, 2018.

[56] Mahdi Soltanolkotabi, Adel Javanmard, and Jason D Lee. Theoretical insights into the optimization landscape of over-parameterized shallow neural networks. *IEEE Transactions on Information Theory*, 65(2):742–769, 2018.

[57] Simon Du, Jason Lee, Haochuan Li, Liwei Wang, and Xiyu Zhai. Gradient descent finds global minima of deep neural networks. In *International Conference on Machine Learning*, pages 1675–1685. PMLR, 2019.

[58] Roman Novak, Yasaman Bahri, Daniel A Abolafia, Jeffrey Pennington, and Jascha Sohl-Dickstein. Sensitivity and generalization in neural networks: an empirical study. *arXiv preprint arXiv:1802.08760*, 2018.

[59] Jinlong Liu, Guoqing Jiang, Yunzhi Bai, Ting Chen, and Huayan Wang. Understanding why neural networks generalize well through gsnr of parameters. *arXiv preprint arXiv:2001.07384*, 2020.

[60] David A McAllester. Pac-bayesian model averaging. In *Proceedings of the twelfth annual conference on Computational learning theory*, pages 164–170, 1999.

[61] Gintare Karolina Dziugaite and Daniel M Roy. Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data. *arXiv preprint arXiv:1703.11008*, 2017.

[62] Yiding Jiang, Behnam Neyshabur, Hossein Mobahi, Dilip Krishnan, and Samy Bengio. Fantastic generalization measures and where to find them. *arXiv preprint arXiv:1912.02178*, 2019.

[63] Mario Geiger, Stefano Spigler, Stéphane d'Ascoli, Levent Sagun, Marco Baity-Jesi, Giulio Biroli, and Matthieu Wyart. Jamming transition as a paradigm to understand the loss landscape of deep neural networks. *Physical Review E*, 100(1):012115, 2019.

[64] Levent Sagun, Leon Bottou, and Yann LeCun. Eigenvalues of the hessian in deep learning: Singularity and beyond. *arXiv preprint arXiv:1611.07476*, 2016.

[65] Guy Gur-Ari, Daniel A Roberts, and Ethan Dyer. Gradient descent happens in a tiny subspace. *arXiv preprint arXiv:1812.04754*, 2018.

[66] Vardan Papyan. Traces of class/cross-class structure pervade deep learning spectra. *Journal of Machine Learning Research*, 21(252):1–64, 2020.

[67] Stanislav Fort and Surya Ganguli. Emergent properties of the local geometry of neural loss landscapes. *arXiv preprint arXiv:1910.05929*, 2019.

[68] Haowei He, Gao Huang, and Yang Yuan. Asymmetric valleys: Beyond sharp and flat local minima. *arXiv preprint arXiv:1902.00744*, 2019.

[69] Pratik Chaudhari, Anna Choromanska, Stefano Soatto, Yann LeCun, Carlo Baldassi, Christian Borgs, Jennifer Chayes, Levent Sagun, and Riccardo Zecchina. Entropy-sgd: Biasing gradient descent into wide valleys. *Journal of Statistical Mechanics: Theory and Experiment*, 2019(12):124018, 2019.

[70] Pavel Izmailov, Dmitrii Podoprikhin, Timur Garipov, Dmitry Vetrov, and Andrew Gordon Wilson. Averaging weights leads to wider optima and better generalization. *arXiv preprint arXiv:1803.05407*, 2018.

[71] Zhen Dong, Zhewei Yao, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. Hawq: Hessian aware quantization of neural networks with mixed-precision. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 293–302, 2019.

[72] Sheng Shen, Zhen Dong, Jiayu Ye, Linjian Ma, Zhewei Yao, Amir Gholami, Michael W Mahoney, and Kurt Keutzer. Q-bert: Hessian based ultra low precision quantization of bert. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8815–8821, 2020.

[73] Ian J Goodfellow, Oriol Vinyals, and Andrew M Saxe. Qualitatively characterizing neural network optimization problems. *arXiv preprint arXiv:1412.6544*, 2014.

[74] Yaim Cooper. The loss landscape of overparameterized neural networks. *arXiv preprint arXiv:1804.10200*, 2018.

[75] C Daniel Freeman and Joan Bruna. Topology and geometry of half-rectified network optimization. *arXiv preprint arXiv:1611.01540*, 2016.

[76] Quynh Nguyen. On connected sublevel sets in deep learning. In *International Conference on Machine Learning*, pages 4790–4799. PMLR, 2019.

[77] Rohith Kuditipudi, Xiang Wang, Holden Lee, Yi Zhang, Zhiyuan Li, Wei Hu, Sanjeev Arora, and Rong Ge. Explaining landscape connectivity of low-cost solutions for multilayer nets. *arXiv preprint arXiv:1906.06247*, 2019.

[78] Alexander Shevchenko and Marco Mondelli. Landscape connectivity and dropout stability of sgd solutions for over-parameterized neural networks. In *International Conference on Machine Learning*, pages 8773–8784. PMLR, 2020.

[79] Jonathan Frankle, Gintare Karolina Dziugaite, Daniel Roy, and Michael Carbin. Linear mode connectivity and the lottery ticket hypothesis. In *International Conference on Machine Learning*, pages 3259–3269. PMLR, 2020.

[80] Stanislav Fort and Stanislaw Jastrzebski. Large scale structure of neural network loss landscapes. *arXiv preprint arXiv:1906.04724*, 2019.

[81] Vidya Muthukumar, Kailas Vodrahalli, Vignesh Subramanian, and Anant Sahai. Harmless interpolation of noisy data in regression. *IEEE Journal on Selected Areas in Information Theory*, 1(1):67–83, 2020.

[82] Song Mei and Andrea Montanari. The generalization error of random features regression: Precise asymptotics and double descent curve. *arXiv preprint arXiv:1908.05355*, 2019.

[83] Peter L Bartlett, Philip M Long, Gábor Lugosi, and Alexander Tsigler. Benign overfitting in linear regression. *Proceedings of the National Academy of Sciences*, 117(48):30063–30070, 2020.

[84] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding deep learning requires rethinking generalization. *arXiv preprint arXiv:1611.03530*, 2016.

[85] Lin Chen, Yifei Min, Mikhail Belkin, and Amin Karbasi. Multiple descent: Design your own generalization curve. *arXiv preprint arXiv:2008.01036*, 2020.

[86] Ben Adlam and Jeffrey Pennington. The neural tangent kernel in high dimensions: Triple descent and a multi-scale theory of generalization. In *International Conference on Machine Learning*, pages 74–84. PMLR, 2020.

[87] Stéphane d'Ascoli, Levent Sagun, and Giulio Biroli. Triple descent and the two kinds of overfitting: Where & why do they appear? *arXiv preprint arXiv:2006.03509*, 2020.

[88] Zitong Yang, Yaodong Yu, Chong You, Jacob Steinhardt, and Yi Ma. Rethinking bias-variance trade-off for generalization of neural networks. In *International Conference on Machine Learning*, pages 10767–10777. PMLR, 2020.

[89] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. Reading digits in natural images with unsupervised feature learning. 2011.

[90] Mauro Cettolo, Christian Girardi, and Marcello Federico. Wit3: Web inventory of transcribed and translated talks. In *Conference of european association for machine translation*, pages 261–268, 2012.

[91] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. *arXiv preprint arXiv:1706.03762*, 2017.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] Our main claims are accurately addressed in the main body of the paper.
   (b) Did you describe the limitations of your work? [Yes] See the discussion on transfer learning in the conclusion section.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A] Our work is related to general properties of optimization loss landscape. We believe our work does not have any immediate negative societal impact.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] Yes.
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [N/A]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We have provided thorough details on reproducing the results. See Section C.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section C.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] All of our 2D diagrams are average from 5 times. In Figure 12, we report the standard deviation.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section C.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We have cited all of the used code, data, and models.
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We have included our code.
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A] All assets are publicly available.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] None of the assets contain personally identifiable information or offensive content.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]