Gradient Descent with Early Stopping is Provably Robust to Label Noise for Overparameterized Neural Networks

Mingchen Li
University of California
Riverside, CA

Mahdi Soltanolkotabi
University of Southern California
Los Angeles, CA

Samet Oymak
University of California
Riverside, CA

Abstract

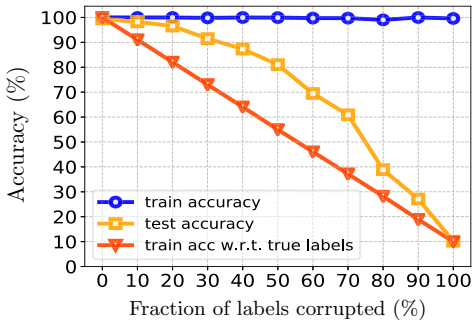
Modern neural networks are typically trained in an over-parameterized regime where the parameters of the model far exceed the size of the training data. Such neural networks in principle have the capacity to (over)fit any set of labels including significantly corrupted ones. Despite this (over)fitting capacity in this paper we demonstrate that such over-parameterized networks have an intriguing robustness capability: they are surprisingly robust to label noise when first order methods with early stopping is used to train them. This paper also takes a step towards demystifying this phenomena. Under a rich dataset model, we show that gradient descent is provably robust to noise/corruption on a constant fraction of the labels. In particular, we prove that: (i) In the first few iterations where the updates are still in the vicinity of the initialization gradient descent only fits to the correct labels essentially ignoring the noisy labels. (ii) To start to overfit to the noisy labels network must stray rather far from the initialization which can only occur after many more iterations. Together, these results show that gradient descent with early stopping is provably robust to label noise and shed light on the empirical robustness of deep networks as well as commonly adopted heuristics to prevent overfitting.

1 Introduction

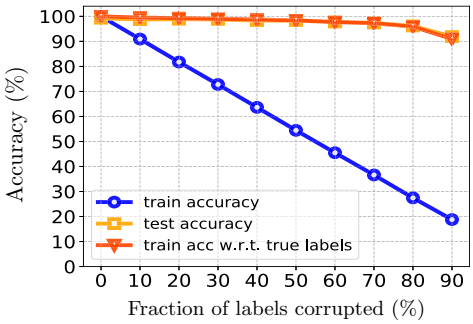
This paper focuses on an intriguing phenomena: over-parameterized neural networks are surprisingly robust to label noise when first order methods with early stopping is used to train them. To observe this phenomena consider Figure 1 where we perform experiments on the MNIST data set. Here, we corrupt a fraction of the labels of the training data by assigning their label uniformly at random. We then fit a four layer model via stochastic gradient descent and plot various performance metrics in Figures 1a and 1b. Figure 1a (blue curve) shows that indeed with a sufficiently large number of iterations the neural network does in fact perfectly fit the corrupted training data. However, Figure 1a also shows that such a model does not generalize to the test data (yellow curve) and the accuracy with respect to the ground truth labels degrades (orange curve). These plots clearly demonstrate that the model overfits with many iterations. In Figure 1b we repeat the same experiment but this time stop the updates after a few iterations (i.e. use early stopping). In this case the train accuracy degrades linearly (blue curve). However, perhaps unexpected, the test accuracy (yellow curve) remains high even with a significant amount of corruption. This suggests that with early stopping the model does not overfit but generalizes to new test data. Even more surprising, the train accuracy (orange curve) with respect to the ground truth labels continues to stay around 100% even when 50% of the labels are corrupted (see also Guan et al. (2018) and Rolnick et al. (2017) for related empirical experiments). That is, with early stopping overparameterized neural networks even correct the corrupted labels! These plots collectively demonstrate that over-parameterized neural networks when combined with early stopping have unique generalization and robustness capabilities. As we detail further in Section 3 this phenomena holds (albeit less pronounced) for richer data models and architectures.

This paper aims to demystify the surprising robustness of overparameterized neural networks when early stopping is used. We show that gradient descent is indeed provably robust to noise/corruption on a *constant fraction of the labels* in such over-parameterized learning scenarios. In particular, under a fairly expressive

Proceedings of the 23rdInternational Conference on Artificial Intelligence and Statistics (AISTATS) 2020, Palermo, Italy. PMLR: Volume 108.



(a) Trained model after many iterations



(b) Trained model with early stopping

Figure 1: In these experiments we use a 4 layer neural network consisting of two convolution layers followed by two fully-connected layers to train MNIST with various amounts of random corruption on the labels. In this architecture the convolution layers have width 64 and 128 kernels, and the fully-connected layers have 256 and 10 outputs, respectively. Overall, there are 4.8 million trainable parameters. We use 50k samples for training, 10k samples for validation, and we test the performance on a 10k test dataset. We depict the training accuracy both w.r.t. the corrupted and uncorrupted labels as well as the test accuracy. (a) Shows the performance after 200 epochs of Adadelta where near perfect fitting to the corrupted data is achieved. (b) Shows the performance with early stopping. We observe that with early stopping the trained neural network is robust to label corruption.

dataset model and focusing on one-hidden layer networks, we show that after a few iterations (a.k.a. *early stopping*), gradient descent finds a model (i) that is within a small neighborhood of the point of initialization and (ii) only fits to the correct labels essentially ignoring the noisy labels. We complement these findings by proving that if the network is trained to overfit to the noisy labels, then the solution found by gradient descent must stray rather far from the initial model. Together, these results highlight the key features of a solution that *generalizes well* vs. a solution that *fits well*.

Our theoretical results further highlight the role of *the distance between final and initial network weights* as a key feature that determines noise robustness vs. overfitting. This is inherently connected to the commonly used early stopping heuristic for DNN training as this heuristic helps avoid models that are too far from the point of initialization. In the presence of label noise, we show that gradient descent *implicitly* ignores the noisy labels as long as the model parameters remain close to the initialization. Hence, our results help explain why early stopping improves robustness and helps prevent overfitting. Under proper normalization, the required distance between the final and initial network and the predictive accuracy of the final network is independent of the size of the network such as number of hidden nodes. Our extensive numerical experiments corroborate our theory and verify the surprising robustness of DNNs to label noise. Finally, we would like to note that while our results show that solutions found by gradient descent are inherently robust to label noise, specialized techniques such as ℓ_1 penalization or sample reweighting are known to further improve robustness. Our theoretical framework may enable more rigorous understandings of the benefits of such heuristics when training overparameterized models.

1.1 Prior Art

Our work is connected to recent advances on theory for deep learning as well as heuristics and theory surrounding outlier robust optimization.

Robustness to label corruption: DNNs have the ability to fit to pure noise Zhang et al. (2016), however they are also empirically observed to be highly resilient to label noise and generalize well despite large corruption Rolnick et al. (2017). In addition to early stopping, several heuristics have been proposed to specifically deal with label noise Reed et al. (2014); Malach and Shalev-Shwartz (2017); Scott et al. (2013); Han et al. (2018); Zhang and Sabuncu (2018); Khetan et al. (2017); Basri et al. (2019); Bartlett et al. (2019). See also Frénay et al. (2014); Shen and Sanghavi (2018); Menon et al. (2018); Ren et al. (2018); Arazo et al. (2019) for additional work on dealing with label noise in classification tasks. Label noise is also connected to outlier robustness in regression which is a traditionally well-studied topic. In the context of robust regression and high-dimensional statistics, much of the focus is on regularization techniques to automatically detect and discard outliers by using tools such as ℓ_1 penalization Chen et al. (2013); Li (2013); Balakrishnan et al. (2017); Liu et al. (2018); Bhatia et al. (2015); Foygel and Mackey (2014); Candès et al. (2011). We would also like to note that there is an interesting line of work that focuses on developing robust algorithms for corruption not only in the labels but also input data Diakonikolas et al. (2018); Prasad et al. (2018); Klivans et al. (2018). Finally, noise robustness is particularly

important in safety critical domains. Noise robustness of neural nets has been empirically investigated by Hinton and coauthors in the context of automated medical diagnosis Guan et al. (2018).

Overparameterized neural networks: Intriguing properties and benefits of overparameterized networks have been the focus of a growing list of publications Zhang et al. (2016); Soltanolkotabi et al. (2018); Brutzkus et al. (2017a); Chizat and Bach (2018); Arora et al. (2018a); Ji and Telgarsky (2018); Venturi et al. (2018); Zhu et al. (2018); Soudry and Carmon (2016); Brutzkus and Globerson (2018); Azizan and Hassibi (2018); Neyshabur et al. (2018). A recent line of work Li and Liang (2018); Allen-Zhu et al. (2018a,b); Du et al. (2018b); Zou et al. (2018); Du et al. (2018a); Oymak and Soltanolkotabi (2019); Papyan (2019) shows that overparameterized neural networks can fit the data with random initialization if the number of hidden nodes are polynomially large in the size of the dataset. This line of work however is not informative about the robustness of the trained network against corrupted labels. Indeed, such theory predicts that (stochastic) gradient descent will eventually fit the corrupted labels. In contrast, our focus here is not in finding a global minima, rather a solution that is robust to label corruption. In particular, we show that with early stopping we fit to the correct labels without overfitting to the corrupted training data. Our result also differs from this line of research in another way. The key property utilized in this research area is that the Jacobian of the neural network is well-conditioned at a random initialization if the dataset is sufficiently diverse (e.g. if the points are well-separated). In contrast, in many practical settings the Jacobian is approximately low-rank. We leverage this low-rankness to prove that gradient descent is robust to label corruptions. We further utilize this to explain why neural nets can work with much smaller amounts of overparameterization where the number of parameters grow with rank rather than the sample size. Furthermore, our numerical experiments verify that the Jacobian matrix of real datasets (such as CIFAR10) indeed exhibit low-rank structure. This is related to the observations on the Hessian of deep networks which is empirically observed to be low-rank Sagun et al. (2017); Chaudhari et al. (2016); Javadi et al. (2019); Ghorbani et al. (2019). Recent papers Su and Yang (2019); Oymak et al. (2019); Rahaman et al. (2018) leverage related phenomena to prove/explain generalization and approximation ability of deep nets. More recently, Hu et al. (2019)[1] shows label noise robustness by utilizing the Rademacher complexity based generalization results of Arora et al. (2019). Also see Arora et al. (2018b); Bartlett et al. (2017); Golowich et al. (2017); Song et al. (2018); Brutzkus et al. (2017b); Belkin et al. (2018a,b); Liang and Rakhlin (2018); Oymak et al. (2019); Cao and Gu (2019); Arora et al. (2019); Ma et al. (2019); Allen-Zhu et al. (2018a) for further recent neural network generalization results. While this work does not tackle generalization in the traditional sense, we do show that solutions found by gradient descent are robust to label noise/corruption which demonstrates their predictive capabilities and in turn suggests better generalization. Finally, related to our work, the role of early-stopping in gradient descent is studied by Yao et al. (2007) in the context of function approximation via kernels.

[1]We note that the first draft of this manuscript appeared earlier than Hu et al. (2019); Su and Yang (2019); Oymak et al. (2019).

1.2 Models

We now describe the dataset model used in our theoretical results. We note that while we mainly focus on this model for exposition purposes our results holds for any data set for which the Jacobian of the network is approximately low-rank with a range that is not too spiky (See Section 4 and the supplementary for further detail). In this model we assume that the input samples $\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n \in \mathbb{R}^d$ come from K clusters which are located on the unit Euclidean ball in $\mathbb{R}^d$. We also assume our dataset consists of $\bar{K} \leq K$ classes where each class can be composed of multiple clusters. We consider a deterministic dataset with n samples with roughly balanced clusters each consisting on the order of n/K samples.[2] Finally, while we allow for multiple classes, in our model we assume the labels are scalars and take values in $[-1, 1]$ interval. Each unit Euclidean norm $\boldsymbol{x}$ is assigned to one of these class labels as described next. We formally define our dataset model below and provide an illustration in Figure 2.

Definition 1.1 ((ε_0, δ) Clusterable dataset) *A clusterable dataset $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n \in \mathbb{R}^d \times \mathbb{R}$ is described as follows. The input samples have unit Euclidean norm and originate from K clusters with the ℓth cluster containing n_ℓ data points where $c_{low}\frac{n}{K} \leq n_\ell \leq c_{up}\frac{n}{K}$ for some positive constants c_{low} and c_{up}. Cluster centers are unit norm vectors denoted by $\{\boldsymbol{c}_\ell\}_{\ell=1}^K \subset \mathbb{R}^d$. An input $\boldsymbol{x}$ that belong to the ℓth cluster obey $\|\boldsymbol{x} - \boldsymbol{c}_\ell\|_{\ell_2} \leq \varepsilon_0$, with ε_0 denoting the input noise level.*

The labels y_i belong to one of $\bar{K} \leq K$ classes. Specifically, $y_i \in \{\alpha_1, \ldots, \alpha_{\bar{K}}\}$ with $\{\alpha_\ell\}_{\ell=1}^{\bar{K}} \in [-1, 1]$ denoting the labels associated with each class. All elements of the same cluster belong to the same class and have the same label. However, a class can contain multiple clusters. The labels are separated in the sense that

[2]This is for ease of exposition rather than a particular challenge arising in the analysis.

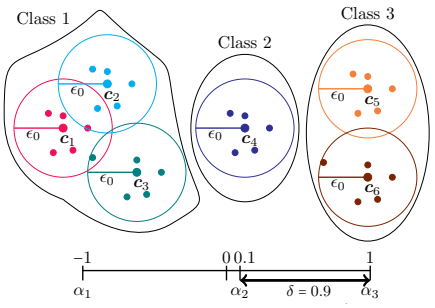


Figure 2: Visualization of the input/label samples and classes according to the clusterable model in Definition 1.1. In the depicted example there are $K = 6$ clusters, $\bar{K} = 3$ classes. In this example the number of data points is $n = 30$ with each cluster containing 5 data points. The labels associated to classes 1, 2, and 3 are $\alpha_1 = -1$, $\alpha_2 = 0.1$, and $\alpha_3 = 1$, respectively so that $\delta = 0.9$. We note that the placement of points are exaggerated for clarity. In particular, per definition the cluster center and data points all have unit Euclidean norm.

$$|\alpha_r - \alpha_s| \geq \delta \quad \textit{for} \quad r \neq s, \tag{1}$$

for some separation $\delta > 0$. Any two clusters ℓ, ℓ' belonging to different classes obey $\|\boldsymbol{c}_\ell - \boldsymbol{c}_{\ell'}\|_{\ell_2} \geq 2\varepsilon_0$.

In the data model above $\{\boldsymbol{c}_\ell\}_{\ell=1}^K$ are the K cluster centers that govern the input distribution. We note that in this model different clusters can be assigned to the same label. Hence, this setup is rich enough to model data which is not linearly separable: e.g. over $\mathbb{R}^2$, we can assign cluster centers $(0,1)$ and $(0,-1)$ to label 1 and cluster centers $(1,0)$ and $(-1,0)$ to label -1. Note that the maximum number of classes are dictated by the separation δ, in particular, $\bar{K} \leq \frac{2}{\delta} + 1$. Our dataset model is inspired from mixture models and is also related to the setup of Li and Liang (2018) which provides polynomial guarantees for learning shallow networks. Next, we introduce our noisy/corrupted dataset model.

Definition 1.2 ($(\rho, \varepsilon_0, \delta)$ corrupted dataset) *A $(\rho, \varepsilon_0, \delta)$ noisy/corrupted dataset $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$ is generated from an (ε_0, δ) clusterable dataset $\{(\boldsymbol{x}_i, \widetilde{y}_i)\}_{i=1}^n$ as follows. For each cluster $1 \leq \ell \leq K$, at most ρn_ℓ of the labels associated with that cluster (which contains n_ℓ points) is assigned to another label value chosen from $\{\alpha_\ell\}_{\ell=1}^{\bar{K}}$. We shall refer to the initial labels $\{\widetilde{y}_i\}_{i=1}^n$ as the ground truth labels.*

We note that this definition allows for a fraction ρ of corruptions in each cluster. Next we define the ground truth label function.

Definition 1.3 (Ground truth label function) *Consider the setting of Def. 1.1 with cluster centers $\{\boldsymbol{c}_\ell\}_{\ell=1}^K \subset \mathbb{R}^d$ and class labels $\{\alpha_\ell\}_{\ell=1}^{\bar{K}}$. Define the ground truth label function $\boldsymbol{x} \mapsto \widetilde{y}(\boldsymbol{x})$ as the function that maps a point $\boldsymbol{x} \in \mathbb{R}^d$ to a class label $\{\alpha_1, \alpha_2, \ldots, \alpha_{\bar{K}}\}$ by assigning it the label corresponding to the closest cluster center. Mathematically*

$$\widetilde{y}(\boldsymbol{x}) = \textit{label of } \boldsymbol{c}_{\hat{\ell}} \quad \textit{where} \quad \hat{\ell} = \operatorname*{arg\,min}_{1 \leq \ell \leq K} \|\boldsymbol{x} - \boldsymbol{c}_\ell\|_{\ell_2}.$$

In particular, when applied to the training data it yields the ground truth labels i.e. $\tilde{y}(\boldsymbol{x}_i) = \tilde{y}_i$.

Network model: We will study the ability of neural networks to learn this corrupted dataset model. To proceed, let us introduce our neural network model. We consider a network with one hidden layer that maps $\mathbb{R}^d$ to $\mathbb{R}$. Denoting the number of hidden nodes by k, this network is characterized by an activation function ϕ, input weight matrix $\boldsymbol{W} \in \mathbb{R}^{k \times d}$ and output weight vector $\boldsymbol{v} \in \mathbb{R}^k$. In this work, we will fix output $\boldsymbol{v}$ to be a unit vector where half the entries are $1/\sqrt{k}$ and other half are $-1/\sqrt{k}$ to simplify the exposition. We will only optimize over the weight matrix $\boldsymbol{W}$ which contains most of the network parameters and will be shown to be sufficient for robust learning. We will also assume ϕ has bounded first and second order derivatives, i.e. $|\phi'(z)|, |\phi''(z)| \leq \Gamma$ for some constant $\Gamma > 0$ for all z. The network's prediction at an input sample $\boldsymbol{x}$ is given by

$$\boldsymbol{x} \mapsto f(\boldsymbol{W}, \boldsymbol{x}) = \boldsymbol{v}^T \phi(\boldsymbol{W}\boldsymbol{x}), \tag{2}$$

where the activation function ϕ applies entrywise. Given a dataset $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n$, we shall train the network via minimizing the empirical risk over the training data via a quadratic loss

$$\mathcal{L}(\boldsymbol{W}) = \frac{1}{2} \sum_{i=1}^n (y_i - f(\boldsymbol{W}, \boldsymbol{x}_i))^2. \tag{3}$$

In particular, we will run gradient descent with a constant learning rate η, starting from a random initialization $\boldsymbol{W}_0$ via the following gradient descent updates

$$\boldsymbol{W}_{\tau+1} = \boldsymbol{W}_\tau - \eta \nabla \mathcal{L}(\boldsymbol{W}_\tau). \tag{4}$$

2 Main Results

Our main result shows that overparameterized neural networks, when trained via gradient descent using early stopping are fairly robust to label noise. Throughout, $\|\cdot\|$ denotes the largest singular value of a given matrix. c, c_0, C, C_0 etc. represent numerical constants. The ability of neural networks to learn from the training data, even without label corruption, naturally depends on the diversity of the input training data. Indeed, if two input data are nearly the same but have different uncorrupted labels reliable learning is difficult. We will quantify this notion of diversity via a notion of condition number related to a covariance matrix involving the activation ϕ and the cluster centers $\{\boldsymbol{c}_\ell\}_{\ell=1}^K$. This definition is induced by the Neural Tangent Kernel (Jacot et al. (2018)) which provides a linearization of the network at random initialization.

Definition 2.1 (Neural Net Cluster Covariance) *Define the matrix of cluster centers*

$$\boldsymbol{C} = [\boldsymbol{c}_1 \ \ldots \ \boldsymbol{c}_K]^T \in \mathbb{R}^{K\times d}.$$

Let $\boldsymbol{g} \sim \mathcal{N}(0, \boldsymbol{I}_d)$. *Define the neural net covariance matrix* $\boldsymbol{\Sigma}(\boldsymbol{C})$ *as*

$$\boldsymbol{\Sigma}(\boldsymbol{C}) = (\boldsymbol{C}\boldsymbol{C}^T) \bigodot \mathbb{E}_{\boldsymbol{g}}[\phi'(\boldsymbol{C}\boldsymbol{g})\phi'(\boldsymbol{C}\boldsymbol{g})^T].$$

Here $\odot$ *denotes the elementwise product. Also denote the minimum eigenvalue of* $\boldsymbol{\Sigma}(\boldsymbol{C})$ *by* $\lambda(\boldsymbol{C})$.

One can view $\boldsymbol{\Sigma}(\boldsymbol{C})$ as an empirical kernel matrix associated with the network where the kernel is given by $\mathcal{K}(\boldsymbol{c}_i, \boldsymbol{c}_j) = \boldsymbol{\Sigma}_{ij}(\boldsymbol{C})$. Note that $\boldsymbol{\Sigma}(\boldsymbol{C})$ is trivially rank deficient if there are two cluster centers that are identical. In this sense, the minimum eigenvalue of $\boldsymbol{\Sigma}(\boldsymbol{C})$ will quantify the ability of the neural network to distinguish between distinct cluster centers. The more distinct the cluster centers, the larger $\lambda(\boldsymbol{C})$ is. Throughout we shall assume that $\lambda(\boldsymbol{C})$ is strictly positive. Related assumptions are empirically and theoretically studied in earlier works by Allen-Zhu et al. (2018b); Xie et al. (2016); Du et al. (2018b,a). For instance, when the cluster centers are maximally diverse e.g. uniformly at random from the unit sphere $\lambda(\boldsymbol{C})$ scales like a constant (Oymak and Soltanolkotabi (2019)). Additionally, for ReLU activation, if the cluster centers are separated by a distance $\nu > 0$, then $\lambda(\boldsymbol{C}) \geq \frac{\nu}{100K^2}$ (Zou et al. (2018); Oymak and Soltanolkotabi (2019)).

Now that we have a quantitative characterization of distinctiveness/diversity in place we are now ready to state our main result. We note that this theorem is slightly simplified by ignoring logarithmic terms and precise dependencies on Γ. The supplementary provides precise statements.

Theorem 2.2 (Main result) *Consider a dataset* $\{(\boldsymbol{x}_i, y_i)\}_{i=1}^n \in \mathbb{R}^d \times \mathbb{R}$ *per Def. 1.2. Starting from an initial weight matrix* $\boldsymbol{W}_0 \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$ *entries, run gradient updates* $\boldsymbol{W}_{\tau+1} = \boldsymbol{W}_\tau - \eta\nabla\mathcal{L}(\boldsymbol{W}_\tau)$ *with properly chosen constant step size* η *and assume*

$$k \gtrsim \frac{K^2\|\boldsymbol{C}\|^4}{\lambda(\boldsymbol{C})^4},$$

If $\varepsilon_0 \lesssim \delta\lambda(\boldsymbol{C})^2/K^2$ *and* $\rho \leq \delta/8$ *with high probability, after* $T \propto \frac{\|\boldsymbol{C}\|^2}{\lambda(\boldsymbol{C})}$ *iterations, the model* $\boldsymbol{W}_T$ *predicts the true label function* $\widetilde{y}(\boldsymbol{x})$ *for all input* $\boldsymbol{x} \in \mathbb{R}^d$ *that lie within* ε_0 *neighborhood of a cluster center* $\{\boldsymbol{c}_k\}_{k=1}^K$. *That is,*

$$\arg\min_{\alpha_\ell:1\leq\ell\leq\bar{K}} |f(\boldsymbol{W}_T, \boldsymbol{x}) - \alpha_\ell| = \widetilde{y}(\boldsymbol{x}). \tag{5}$$

Eq. (5) applies to all training samples. Finally, for all $0 \leq \tau \leq T$, *the distance to initialization obeys*

$$\|\boldsymbol{W}_\tau - \boldsymbol{W}_0\|_F \lesssim \left(\sqrt{K} + \frac{K^2}{\|\boldsymbol{C}\|^2}\tau\varepsilon_0\right).$$

Theorem 2.2 shows that gradient descent with early stopping is robust and predicts the correct labels despite the corruption. See below for further properties.

Robustness. The solution found by gradient descent with early stopping degrades gracefully as the label corruption level ρ grows. In particular, as long as $\rho \leq \delta/8$, the final model is able to correctly classify any input data. In particular, when applied to the training data (5) yields $\arg\min_{\alpha_\ell:1\leq\ell\leq\bar{K}} |f(\boldsymbol{W}_\tau, \boldsymbol{x}) - \alpha_\ell| = \widetilde{y}_i$ so that the network labels are identical to the ground truth labels completely removing the corruption on the training data. In our setup, intuitively the label gap obeys $\delta \sim \frac{1}{\bar{K}}$, hence, we prove robustness to

$$\text{Total number of corrupted labels} \lesssim \frac{n}{\bar{K}}.$$

This result is independent of number of clusters and only depends on number of classes. An interesting future direction is to improve this result to allow on the order of n corrupted labels.

Early stopping time. Only a few iterations suffice to find a good model (at most order K iterations typically $\max(1, K/d)$ modulo condition numbers).

Modest overparameterization. Our result applies as soon as the number of hidden units in the network exceeds $K^2\|\boldsymbol{C}\|^4$ which lies between K^2 and K^4 which is independent of the sample size n. This can be interpreted as network having enough capacity to fit the cluster centers $\{\boldsymbol{c}_\ell\}_{\ell=1}^K$ and their true labels. If cluster centers are incoherent (e.g. random) and $K \geq d$, the required number of parameters in the network $(k \times d)$ scales as $dK^2\|\boldsymbol{C}\|^4 \lesssim K^4$.

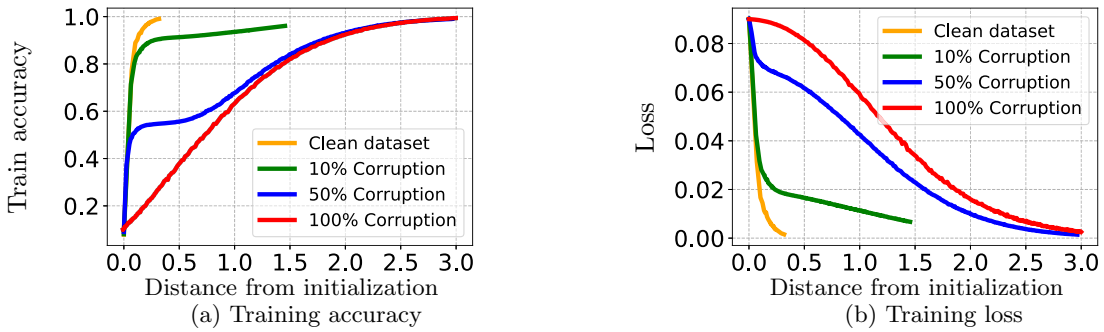
Distance from initialization. Another feature of Theorem 2.2 is that the network weights do not stray far from the initialization as the distance between the initial model and the final model (at most) grows with the square root of the number of clusters ($\sqrt{K}$). Intuitively, more clusters correspond to a richer data distribution, hence we need to travel further away to find a viable model. While our focus in this work is early stopping, the importance of distance to initialization motivates the use of ℓ_2-regularization with respect to the initial point i.e. solving the regularized empirical risk minimization

$$\boldsymbol{W}_{\text{ridge}} = \arg\min_{\boldsymbol{W}} \frac{1}{2}\sum_{i=1}^n (y_i - f(\boldsymbol{W}, \boldsymbol{x}_i))^2 + \lambda\|\boldsymbol{W} - \boldsymbol{W}_0\|_F^2,$$

where $\boldsymbol{W}_0$ is the point of initialization for the gradient based algorithm that will be used to solve above.

2.1 To (over)fit to corrupted labels requires straying far from initialization

In this section we wish to provide further insight into why early stopping enables robustness and generaliz-



(a) Training accuracy (b) Training loss

Figure 3: We depict the training accuracy of a LENET model trainined on 3000 samples from MNIST as a function of relative distance from initialization. Here, the x-axis keeps track of the distance between the current and initial weights of all layers combined.

able solutions. Our main insight is that while a neural network maybe expressive enough to fit a corrupted dataset, the model has to travel a longer distance from the point of initialization as a function of the distance from the cluster centers ε_0 and the amount of corruption. We formalize this idea as follows. Suppose (1) two input points are close to each other (e.g. they are from the same cluster), (2) but their labels are different, hence the network has to map them to distant outputs. Then, the network has to be large enough so that it can amplify the small input difference to create a large output difference. Our first result formalizes this for a randomly initialized network. Our random initialization picks $\boldsymbol{W}$ with i.i.d. standard normal entries which ensures that the network is isometric i.e. given input $\boldsymbol{x}$, $\mathbb{E}[f(\boldsymbol{W},\boldsymbol{x})^2] = \mathcal{O}(\|\boldsymbol{x}\|_{\ell_2}^2)$.

Theorem 2.3 *Let $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathbb{R}^d$ be two vectors with unit ℓ_2 norm obeying $\|\boldsymbol{x}_2 - \boldsymbol{x}_1\|_{\ell_2} \le \epsilon_0$. Let $f(\boldsymbol{W},\boldsymbol{x}) = \boldsymbol{v}^T\phi(\boldsymbol{W}\boldsymbol{x})$ where $\boldsymbol{v}$ is fixed, $\boldsymbol{W} \in \mathbb{R}^{k\times d}$, and $k \ge cd$ where c, c', c'' are constants and $|\phi'|, |\phi''| \le \Gamma$. Let y_1 and y_2 be two scalars satisfying $|y_2 - y_1| \ge \delta$. Suppose $\boldsymbol{W}_0 \overset{i.i.d.}{\sim} \mathcal{N}(0,1)$. Then, with probability $1 - 2e^{-(k+d)} - 2e^{-\frac{t^2}{2}}$, for any $\boldsymbol{W}$ such that $\|\boldsymbol{W} - \boldsymbol{W}_0\|_F \le c'\sqrt{k}$ and*

$$f(\boldsymbol{W},\boldsymbol{x}_1) = y_1 \quad \text{and} \quad f(\boldsymbol{W},\boldsymbol{x}_2) = y_2,$$

holds, we have $\|\boldsymbol{W} - \boldsymbol{W}_0\| \ge \frac{c''\delta}{\Gamma\varepsilon_0} - \frac{t}{1000}$.

In words, this result shows that in order to fit to a dataset with a *single corrupted label*, a randomly initialized network has to traverse a distance of at least δ/ε_0. The supplementary clarifies the role of the corruption amount s and shows that more label corruption within a fixed class requires a model with a larger norm in order to fit the labels.

Can we really overfit to corruption? A natural question is whether early stopping is necessary i.e. can we perfectly interpolate to the corrupted dataset model of Definition 1.2. The recent works Du et al. (2018a); Allen-Zhu et al. (2018b); Oymak and Soltanolkotabi (2019) on neural net optimization answers this affirmatively. In particular, as long as no two input samples are identical, sufficiently wide neural networks trained with gradient descent can provably and perfectly interpolate a corrupted dataset.

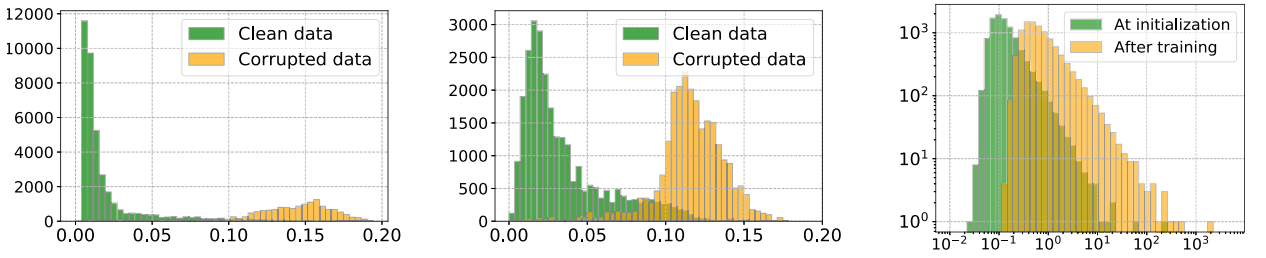
3 Numerical experiments

We conduct several experiments to investigate the robustness capabilities of deep networks to label corruption. In our first set of experiments, we explore the relationship between loss, accuracy, and amount of label corruption on the MNIST dataset to corroborate our theory. Our next experiments study the distribution of the loss and the Jacobian on the CIFAR-10 dataset. Finally, we simulate our theoretical model by generating data according to the corrupted data model of Definition 1.2 and verify the robustness capability of gradient descent with early stopping in this model[3].

In Figure 3, we train the same model used in Figure 1 with $n = 3,000$ MNIST samples for different amounts of corruption. Our theory predicts that more label corruption leads to a larger distance to initialization. To probe this hypothesis, Figure 3a and 3b visualizes training accuracy and training loss as a function of the distance from the initialization. These results demonstrate that the distance from initialization gracefully increase with more corruption.

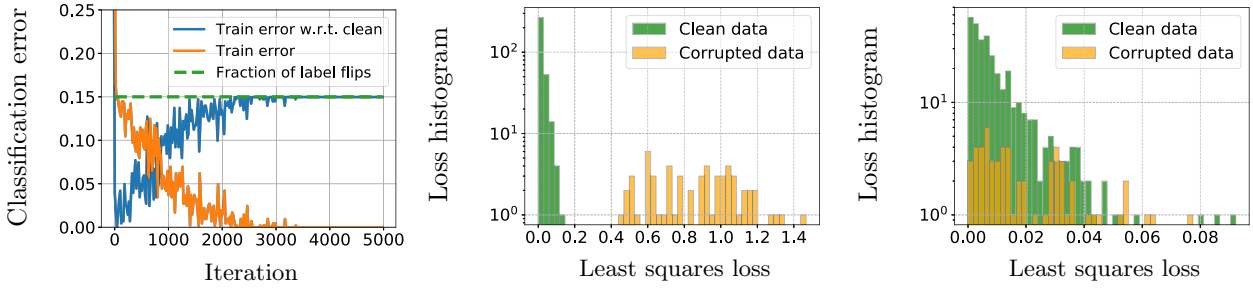
Next, we study the distribution of the individual sample losses on CIFAR-10. We conducted two experiments using Resnet-20 with least square loss. In Figure 4a and 4b we assess the noise robustness of gradient descent where we used all 50,000 samples with either 30% random corruption or 50% random corruption. The supplementary shows that when the corruption

[3] All experiments use least square loss corresponding to our theory, but we have same observation on cross entropy loss and provide figures in appendix.



(a) 30% corruption (b) 50% corruption (c) Class Airplane and Automobile

Figure 4: (a)(b) Are histograms of the least square loss of individual data points based on a model trained on 50,000 samples from CIFAR-10 with early stopping. The loss distribution of clean and corrupted data are separated but gracefully overlap as corruption increases. (c) is histogram of singular values obtained by forming the Jacobian by taking partial derivatives of class Airplane and Automobile on 10000 samples.



(a) Fraction of incorrect predictions (b) Loss histogram at iteration 80 (c) Loss histogram at iteration 4500

Figure 5: We experiment with the corrupted dataset model of Definition 1.2. We picked $K = 2$ classes and set $n = 400$ and $\varepsilon_0 = 0.5$. Trained 30% corrupted data with $k = 1000$ hidden units. In average 15% of labels actually flip which is highlighted by the dashed green line.

level is small, the loss distribution of corrupted vs. clean samples should be separable. Figure 4a shows that when 30% of the data is corrupted the distributions are approximately separable. When we increase the corruption to 50% in Figure 4b, the training loss on the clean data increases as predicted by our theory and the distributions start to gracefully overlap.

As we briefly discuss in Section 4 (see proofs in the supplementary for more extensive discussion), our technical framework utilizes the low-rank structure of the Jacobian matrix of the model. We now further investigate this hypothesis. For a binary class task, size of the Jacobian matrix is sample size (n) $\times$ total number of parameters in the model (p). The neural network model we used for CIFAR 10 has around $p = 270,000$ parameters in total. In Figure 4c we illustrate the singular value histogram of binary Jacobian model where the training classes are Airplane and Automobile. We trained the model with all samples and focus on the histogram of all training data ($n = 10,000$) before and after the training. In particular, only 10 to 20 singular values are larger than 0.1$\times$ the top one. This is consistent with earlier works that studied the Hessian spectrum. Another intriguing finding is that the distribution of before and after training are fairly close to each other highlighting that even at random initialization, the Jacobian spectrum exhibits bimodal structure.

In Figure 5, we turn our attention to verifying our findings for the corrupted dataset model of Definition 1.2. We generated $K = 2$ classes where the associated clusters centers are generated uniformly at random on the unit sphere of $\mathbb{R}^{d=20}$. We also generate the input samples at random around these two clusters uniformly at random on a sphere of radius $\varepsilon_0 = 0.5$ around the corresponding cluster center. Hence, the clusters are guaranteed to be at least 1 distance from each other to prevent overlap. Overall we generate $n = 400$ samples (200 per class/cluster). Here, $\bar{K} = K = 2$ and the class labels are 0 and 1. We picked a network with $k = 1000$ hidden units and trained on a data set with 400 samples where 30% of the labels were corrupted. Figure 5a plots the trajectory of training error and highlights the model achieves good classification in the first few iterations and ends up overfitting later on. In Figures 5b and 5c, we focus on the loss distribution of 5a at iterations 80 and 4500. In this figure, we visualize the loss distribution of clean and corrupted data. Figure 5b highlights the loss distribution with early stopping and implies that the gap between corrupted and clean loss distributions is surprisingly resilient despite a large amount of corruption and the high-capacity of the model. In Figure 5c, we

repeat plot after many more iterations at which point the model overfits. This plot shows that the distribution of the two classes overlap demonstrating that the model has overfit the corruption and lacks generalization/robustness.

4 Key Insights and Technical Ideas

Our key idea is that semantically meaningful datasets (such as the clusterable dataset model) should have a low-dimensional representation. We use Jacobian mapping of the neural network to capture such structure in data which is represented as follow.

$$\mathcal{J}(\boldsymbol{W}) = \left[\frac{\partial f(\boldsymbol{x}_1,\boldsymbol{W})}{\partial \boldsymbol{W}} \ \ldots \ \frac{\partial f(\boldsymbol{x}_n,\boldsymbol{W})}{\partial \boldsymbol{W}}\right]^T.$$

The key insight that enable our proofs is that the Jacobian mapping of neural networks typically exhibit (1) low-rank structure with a few large singular values and many small ones and (2) the sparse corruptions are mostly aligned with the small singular directions. We have empirically verified that both properties hold for a variety of neural networks and data sets.

Using these insights we show that the optimization is implicitly decomposed into two stages which corresponds to the column subspaces induced by the large and small singular values of the Jacobian. To make this precise let us denote the overall network prediction by $f(\boldsymbol{W}) = [f(\boldsymbol{W},\boldsymbol{x}_1) \ \ldots \ (\boldsymbol{W},\boldsymbol{x}_n)]$ and note that the gradient mapping takes the form

$$\mathcal{J}^T \underbrace{(f(\boldsymbol{W}_\tau)-\boldsymbol{y})}_{\text{corrupted residual}} = \mathcal{J}^T \underbrace{(f(\boldsymbol{W}_\tau)-\tilde{\boldsymbol{y}})}_{\text{clean residual}} + \underbrace{(\tilde{\boldsymbol{y}}-\boldsymbol{y})}_{\text{label corruption}}$$

We prove that the clean residual is aligned with the top singular direction whereas label noise is aligned with the small singular directions. The latter is a consequence of the fact that the top singular vectors are diffused and the label noise is sparse (constant fraction of corruption). As a result, gradient descent learns the useful information (clean residual) in few iterations whereas it takes much longer to overfit to noise justifying the use of early stopping.

The following meta theorem focuses on the first stage of the optimization and shows that in a general nonlinear learning problem if the Jacobian is low-rank and has a diffused range then the label noise is effectively suppressed in the first few iterations. This in turn provides a sharp control on the impact of noise on the final model for each input example. Formally, we assume that the range space $\mathcal{S} = \text{range}(\mathcal{J}(\boldsymbol{\theta}))$ is diffused in the sense that any unit length $\boldsymbol{v} \in \mathcal{S}$ satisfies $\|\boldsymbol{v}\|_{\ell_\infty} \le \sqrt{\gamma/n}$ for a small γ (e.g. $\boldsymbol{v}$ is scaled all ones vector). We note that for this diffuseness property to hold it is sufficient for the Jacobian to be approximately low-rank and the prominent directions to be diffused.

Theorem 4.1 (Robustness via diffuseness) *Consider a nonlinear least squares problem of the form $\mathcal{L}(\boldsymbol{\theta}) = \frac{1}{2}\|f(\boldsymbol{\theta})-\boldsymbol{y}\|_{\ell_2}^2$. Suppose $f(\boldsymbol{\theta}_0) = 0$ and assume that $\mathcal{J}(\boldsymbol{\theta})$ is sufficiently smooth function of $\boldsymbol{\theta}$ (see Assumption 3 in supplementary) and $\mathcal{S} = \text{range}(\mathcal{J}(\boldsymbol{\theta}))$ is γ-diffused as above. Let $\widetilde{\boldsymbol{y}} = [\widetilde{y}_1 \ \ldots \ \widetilde{y}_n] \in \mathcal{S}$ denote the uncorrupted labels and $\boldsymbol{e} = \boldsymbol{y} - \widetilde{\boldsymbol{y}}$ denote the label corruption. Also assume $\boldsymbol{e}$ is ρn-sparse and its entries are bounded by 1 in absolute value. Then, running gradient descent with a constant learning rate, after polynomially many iterations, we have*

$$\|f(\boldsymbol{\theta}_\tau) - \widetilde{\boldsymbol{y}}\|_{\ell_\infty} \le \gamma\rho.$$

In words, more diffused subspace and sparser vector leads to smaller entrywise prediction error. Note that as long as $\gamma\rho < \delta/2$ (where δ is class label separation), network accurately classifies all examples. For our proofs surrounding the clusterable dataset model, we show that $\mathcal{S}$ is indeed very diffused (essentially constant γ) to obtain such tight entrywise error control.

5 Conclusions

In this paper, we studied the robustness of overparameterized neural networks to label corruption from a theoretical lens. We provided robustness guarantees for training networks with gradient descent when early stopping is used and complemented these guarantees with lower bounds. Our results point to the distance between final and initial network weights as a key feature to determine robustness vs. overfitting which is inline with weight decay and early stopping heuristics. We also carried out extensive numerical experiments to verify the theoretical predictions as well as technical assumptions. While our results shed light on the intriguing properties of overparameterized neural network optimization, it would be appealing (i) to extend our results to deeper network architecture, (ii) to more complex data models, and also (iii) to explore other heuristics that can further boost the robustness of gradient descent methods.

6 Acknowledgements

Samet Oymak is supported by NSF-CNS award #1932254. Mahdi Soltanolkotabi is supported by the Packard Fellowship in Science and Engineering, a Sloan Research Fellowship in Mathematics, an NSF-CAREER under award #1846369, the Air Force Office of Scientific Research Young Investigator Program (AFOSR-YIP) under award #FA9550-18-1-0078, and an NSF-CIF award #1813877.

References

Allen-Zhu, Z., Li, Y., and Liang, Y. (2018a). Learning and generalization in overparameterized neural networks, going beyond two layers. *arXiv preprint arXiv:1811.04918*.

Allen-Zhu, Z., Li, Y., and Song, Z. (2018b). A convergence theory for deep learning via over-parameterization. *arXiv preprint arXiv:1811.03962*.

Arazo, E., Ortego, D., Albert, P., O'Connor, N. E., and McGuinness, K. (2019). Unsupervised label noise modeling and loss correction. *arXiv preprint arXiv:1904.11238*.

Arora, S., Cohen, N., and Hazan, E. (2018a). On the optimization of deep networks: Implicit acceleration by overparameterization. *arXiv preprint arXiv:1802.06509*.

Arora, S., Du, S. S., Hu, W., Li, Z., and Wang, R. (2019). Fine-grained analysis of optimization and generalization for overparameterized two-layer neural networks. *arXiv preprint arXiv:1901.08584*.

Arora, S., Ge, R., Neyshabur, B., and Zhang, Y. (2018b). Stronger generalization bounds for deep nets via a compression approach.

Azizan, N. and Hassibi, B. (2018). Stochastic gradient/mirror descent: Minimax optimality and implicit regularization. *arXiv preprint arXiv:1806.00952*.

Balakrishnan, S., Du, S. S., Li, J., and Singh, A. (2017). Computationally efficient robust sparse estimation in high dimensions. In *Conference on Learning Theory*, pages 169–212.

Bartlett, P., Foster, D. J., and Telgarsky, M. (2017). Spectrally-normalized margin bounds for neural networks.

Bartlett, P. L., Long, P. M., Lugosi, G., and Tsigler, A. (2019). Benign overfitting in linear regression. *arXiv preprint arXiv:1906.11300*.

Basri, R., Jacobs, D., Kasten, Y., and Kritchman, S. (2019). The convergence rate of neural networks for learned functions of different frequencies. *arXiv preprint arXiv:1906.00425*.

Belkin, M., Hsu, D., and Mitra, P. (2018a). Overfitting or perfect fitting? risk bounds for classification and regression rules that interpolate.

Belkin, M., Rakhlin, A., and Tsybakov, A. B. (2018b). Does data interpolation contradict statistical optimality?

Bhatia, K., Jain, P., and Kar, P. (2015). Robust regression via hard thresholding. In *Advances in Neural Information Processing Systems*, pages 721–729.

Brutzkus, A. and Globerson, A. (2018). Over-parameterization improves generalization in the xor detection problem.

Brutzkus, A., Globerson, A., Malach, E., and Shalev-Shwartz, S. (2017a). Sgd learns over-parameterized networks that provably generalize on linearly separable data. *arXiv preprint arXiv:1710.10174*.

Brutzkus, A., Globerson, A., Malach, E., and Shalev-Shwartz, S. (2017b). Sgd learns over-parameterized networks that provably generalize on linearly separable data.

Candès, E. J., Li, X., Ma, Y., and Wright, J. (2011). Robust principal component analysis? *Journal of the ACM (JACM)*, 58(3):11.

Cao, Y. and Gu, Q. (2019). A generalization theory of gradient descent for learning over-parameterized deep relu networks. *arXiv preprint arXiv:1902.01384*.

Chaudhari, P., Choromanska, A., Soatto, S., LeCun, Y., Baldassi, C., Borgs, C., Chayes, J., Sagun, L., and Zecchina, R. (2016). Entropy-sgd: Biasing gradient descent into wide valleys. *arXiv preprint arXiv:1611.01838*.

Chen, Y., Caramanis, C., and Mannor, S. (2013). Robust sparse regression under adversarial corruption. In *International Conference on Machine Learning*, pages 774–782.

Chizat, L. and Bach, F. (2018). On the global convergence of gradient descent for over-parameterized models using optimal transport. *arXiv preprint arXiv:1805.09545*.

Diakonikolas, I., Kamath, G., Kane, D. M., Li, J., Steinhardt, J., and Stewart, A. (2018). Sever: A robust meta-algorithm for stochastic optimization. *arXiv preprint arXiv:1803.02815*.

Du, S. S., Lee, J. D., Li, H., Wang, L., and Zhai, X. (2018a). Gradient descent finds global minima of deep neural networks. *arXiv preprint arXiv:1811.03804*.

Du, S. S., Zhai, X., Poczos, B., and Singh, A. (2018b). Gradient descent provably optimizes over-parameterized neural networks. *arXiv preprint arXiv:1810.02054*.

Foygel, R. and Mackey, L. (2014). Corrupted sensing: Novel guarantees for separating structured signals. *IEEE Transactions on Information Theory*, 60(2):1223–1247.

Frénay, B., Kabán, A., et al. (2014). A comprehensive introduction to label noise. In *ESANN*.

Ghorbani, B., Krishnan, S., and Xiao, Y. (2019). An investigation into neural net optimization

via hessian eigenvalue density. *arXiv preprint arXiv:1901.10159*.

Golowich, N., Rakhlin, A., and Shamir, O. (2017). Size-independent sample complexity of neural networks.

Guan, M. Y., Gulshan, V., Dai, A. M., and Hinton, G. E. (2018). Who said what: Modeling individual labelers improves classification. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., and Sugiyama, M. (2018). Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in Neural Information Processing Systems*, pages 8536–8546.

Hu, W., Li, Z., and Yu, D. (2019). Understanding generalization of deep neural networks trained with noisy labels. *arXiv preprint arXiv:1905.11368*.

Jacot, A., Gabriel, F., and Hongler, C. (2018). Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in neural information processing systems*, pages 8571–8580.

Javadi, H., Balestriero, R., and Baraniuk, R. (2019). A hessian based complexity measure for deep networks. *arXiv preprint arXiv:1905.11639*.

Ji, Z. and Telgarsky, M. (2018). Gradient descent aligns the layers of deep linear networks.

Khetan, A., Lipton, Z. C., and Anandkumar, A. (2017). Learning from noisy singly-labeled data. *arXiv preprint arXiv:1712.04577*.

Klivans, A., Kothari, P. K., and Meka, R. (2018). Efficient algorithms for outlier-robust regression. *arXiv preprint arXiv:1803.03241*.

Li, X. (2013). Compressed sensing and matrix completion with constant proportion of corruptions. *Constructive Approximation*, 37(1):73–99.

Li, Y. and Liang, Y. (2018). Learning overparameterized neural networks via stochastic gradient descent on structured data. In *Advances in Neural Information Processing Systems*, pages 8168–8177.

Liang, T. and Rakhlin, A. (2018). Just interpolate: Kernel "ridgeless" regression can generalize.

Liu, L., Shen, Y., Li, T., and Caramanis, C. (2018). High dimensional robust sparse regression. *arXiv preprint arXiv:1805.11643*.

Ma, C., Wu, L., et al. (2019). A comparative analysis of the optimization and generalization property of two-layer neural network and random feature models under gradient descent dynamics. *arXiv preprint arXiv:1904.04326*.

Malach, E. and Shalev-Shwartz, S. (2017). Decoupling" when to update" from" how to update". In *Advances in Neural Information Processing Systems*, pages 960–970.

Menon, A. K., van Rooyen, B., and Natarajan, N. (2018). Learning from binary labels with instance-dependent noise. *Machine Learning*, pages 1–35.

Neyshabur, B., Li, Z., Bhojanapalli, S., LeCun, Y., and Srebro, N. (2018). Towards understanding the role of over-parametrization in generalization of neural networks. *arXiv preprint arXiv:1805.12076*.

Oymak, S., Fabian, Z., Li, M., and Soltanolkotabi, M. (2019). Generalization guarantees for neural networks via harnessing the low-rank structure of the jacobian. *arXiv preprint arXiv:1906.05392*.

Oymak, S. and Soltanolkotabi, M. (2019). Towards moderate overparameterization: global convergence guarantees for training shallow neural networks. *arXiv preprint arXiv:1902.04674*.

Papyan, V. (2019). Measuring the spectrum of deepnet hessians.

Prasad, A., Suggala, A. S., Balakrishnan, S., and Ravikumar, P. (2018). Robust estimation via robust gradient estimation. *arXiv preprint arXiv:1802.06485*.

Rahaman, N., Baratin, A., Arpit, D., Draxler, F., Lin, M., Hamprecht, F. A., Bengio, Y., and Courville, A. (2018). On the spectral bias of neural networks. *arXiv preprint arXiv:1806.08734*.

Reed, S., Lee, H., Anguelov, D., Szegedy, C., Erhan, D., and Rabinovich, A. (2014). Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*.

Ren, M., Zeng, W., Yang, B., and Urtasun, R. (2018). Learning to reweight examples for robust deep learning. *arXiv preprint arXiv:1803.09050*.

Rolnick, D., Veit, A., Belongie, S., and Shavit, N. (2017). Deep learning is robust to massive label noise. *arXiv preprint arXiv:1705.10694*.

Sagun, L., Evci, U., Guney, V. U., Dauphin, Y., and Bottou, L. (2017). Empirical analysis of the hessian of over-parametrized neural networks. *arXiv preprint arXiv:1706.04454*.

Scott, C., Blanchard, G., and Handy, G. (2013). Classification with asymmetric label noise: Consistency and maximal denoising. In *Conference On Learning Theory*, pages 489–511.

Shen, Y. and Sanghavi, S. (2018). Iteratively learning from the best. *arXiv preprint arXiv:1810.11874*.

Soltanolkotabi, M., Javanmard, A., and Lee, J. D. (2018). Theoretical insights into the optimization landscape of over-parameterized shallow neural networks. *IEEE Transactions on Information Theory*.

Song, M., Montanari, A., and Nguyen, P. (2018). A mean field view of the landscape of two-layers neural networks. In *Proceedings of the National Academy of Sciences*, volume 115, pages E7665–E7671.

Soudry, D. and Carmon, Y. (2016). No bad local minima: Data independent training error guarantees for multilayer neural networks.

Su, L. and Yang, P. (2019). On learning over-parameterized neural networks: A functional approximation prospective. *arXiv preprint arXiv:1905.10826*.

Venturi, L., Bandeira, A., and Bruna, J. (2018). Spurious valleys in two-layer neural network optimization landscapes. *arXiv preprint arXiv:1802.06384*.

Xie, B., Liang, Y., and Song, L. (2016). Diverse neural network learns true target functions. *arXiv preprint arXiv:1611.03131*.

Yao, Y., Rosasco, L., and Caponnetto, A. (2007). On early stopping in gradient descent learning. *Constructive Approximation*, 26(2):289–315.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. (2016). Understanding deep learning requires rethinking generalization. *International Conference on Learning Representations*.

Zhang, Z. and Sabuncu, M. R. (2018). Generalized cross entropy loss for training deep neural networks with noisy labels. *arXiv preprint arXiv:1805.07836*.

Zhu, Z., Soudry, D., Eldar, Y. C., and Wakin, M. B. (2018). The global optimization geometry of shallow linear neural networks.

Zou, D., Cao, Y., Zhou, D., and Gu, Q. (2018). Stochastic gradient descent optimizes over-parameterized deep relu networks. *arXiv preprint arXiv:1811.08888*.