Investigating Why Contrastive Learning Benefits Robustness against Label Noise

Yihao Xue [1] Kyle Whitecross [1] Baharan Mirzasoleiman [1]

Abstract

Self-supervised contrastive learning has recently been shown to be very effective in preventing deep networks from overfitting noisy labels. Despite its empirical success, the theoretical understanding of the effect of contrastive learning on boosting robustness is very limited. In this work, we rigorously prove that learned the representation matrix has certain desirable properties in terms its SVD that benefit robustness against label noise. We further show that the low-rank structure of the Jacobian of deep networks pre-trained with contrastive learning allows them to achieve a superior performance initially, when fine-tuned on noisy labels. Finally, we demonstrate that the initial robustness provided by contrastive learning enables robust training methods to achieve state-of-the-art performance under extreme noise levels.

1. Introduction

Very recently, self-supervised contrastive learning has shown a lot of promise in boosting robustness of deep networks against noisy labels (Zheltonozhskii et al., 2022; Hendrycks et al., 2019; Ghosh & Lan, 2021). Despite its empirical success, the theoretical understanding of the effect of contrastive learning on improving robustness of deep networks against noisy labels is very limited.

In this work, we theoretically characterize the beneficial properties of representations obtained by contrastive learning for enhancing robustness against noisy labels. We prove that contrastive learning produces a representation matrix that has: (i) a prominent singular value corresponding to each sub-class in the data, and significantly smaller remaining singular values; and (ii) a large alignment between the prominent singular vectors and the ground-truth labels. Then we analyze the case where a linear model is trained on the obtained representations with labels that are either perturbed with Gaussian noise, or flipped at random to other classes. We show that noise has minimal effect on learning the clean labels and the model can hardly memorize the wrong labels.

We further show that deep networks pre-trained with contrastive learning can be fine-tuned on noisy labels to achieve a superior performance initially, before overfitting the noise. We attribute this to the initial low-rank structure of the Jacobian with a few large singular values associated to its prominent directions and insignificant singular values otherwise.

Finally, we conduct extensive experiments on noisy CIFAR-10 and CIFAR-100 (Krizhevsky & Hinton, 2009) and the mini Webvision datasets (Li et al., 2017) to demonstrate that the initial robustness provided by contrastive learning can be further leveraged by robust methods to achieve state-of-the-art performance under extreme levels of noise.

2. Problem Formulation and Background

Suppose we have a dataset $\mathcal{D} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n$, where $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ denotes the i-th sample with input $\boldsymbol{x}_i \in \mathbb{R}^d$ and its clean one-hot encoded label $\boldsymbol{y}_i \in \mathbb{R}^K$ corresponding to one of the K classes. For example, for a data point $\boldsymbol{x}_i$ from class $j \in [K]$, we have $\boldsymbol{y}_i = \boldsymbol{e}_j$ where $\mathbf{e}_j$ denotes the vector with a 1 in the jth coordinate and 0's elsewhere. We further assume that there are $\bar{K} \geq K$ sub-classes in the data. Sub-classes of a class share the same label, but are distinguishable from each other. We assume that for every data point $\boldsymbol{x}_i$, we only observe a noisy version of its label $\hat{\boldsymbol{y}}_i$. We denote by $\boldsymbol{Y}, \hat{\boldsymbol{Y}} \in \mathbb{R}^{n \times K}$ the matrices of all the one-hot encoded clean and noisy labels of the training data points.

2.1. Self-supervised Contrastive Learning

Self-supervised contrastive learning learns representations of different data points by maximizing agreement between differently augmented views of the same example via a contrastive loss in the latent space, as we discuss below.

Augmentation graph. The augmentations can be used to construct the *population augmentation graph* (HaoChen et al., 2021), whose vertices are all the augmented data in the

[1] Department of Computer Science, University of California, Los Angeles, CA 90095, USA. Correspondence to: Yihao Xue <yihaoxue@g.ucla.edu>.

population distribution, and two vertices are connected with an edge if they are augmentations of the same natural example. Hence, ground-truth classes naturally form connected sub-graphs. Formally, let P be the distribution of all natural data points (raw inputs without augmentation). For a natural data point $\boldsymbol{x}^* \sim P$, let $\mathcal{A}(\cdot|\boldsymbol{x}^*)$ be the distribution of $\boldsymbol{x}^*$'s augmentations. For instance, when $\boldsymbol{x}^*$ represents an image, $A(.|\boldsymbol{x}^*)$ can be the distribution of common augmentations (Chen et al., 2020) including Gaussian blur, color distortion and random cropping. Then, for an augmented data point $\boldsymbol{x}$, $\mathcal{A}(\boldsymbol{x}|\boldsymbol{x}^*)$ is the probability of generating $\boldsymbol{x}$ from $\boldsymbol{x}^*$. The edge weights $w_{x_i x_j} = \mathbb{E}_{\boldsymbol{x}^* \sim P}[\mathcal{A}(\boldsymbol{x}_i|\boldsymbol{x}^*)\mathcal{A}(\boldsymbol{x}_j|\boldsymbol{x}^*)]$ can be interpreted as the marginal probability of generating $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ from a random natural data point.

Contrastive loss. The embeddings produced by contrastive learning can be viewed as a low-rank approximation of the normalized augmentation graph. Effectively, minimizing a loss that performs spectral decomposition on the population augmentation graph can be succinctly written as a contrastive learning objective $\mathfrak{C}(f)$ on neural network representations (HaoChen et al., 2021):

$$\mathfrak{C}(f) = -2\mathbb{E}_{\boldsymbol{x},\boldsymbol{x}^+}[f(\boldsymbol{x})^\top_{,} f(\boldsymbol{x}^+)] + \mathbb{E}_{\boldsymbol{x},\boldsymbol{x}^-}[\Big(\Big(f(\boldsymbol{x})^\top_{,} f(\boldsymbol{x}^-)\Big)\Big)^2], \tag{1}$$

where $f(\boldsymbol{x}) \in \mathbb{R}^p$ is the neural network representation for an input $\boldsymbol{x}$, and $\boldsymbol{x}, \boldsymbol{x}^+$ are drawn from the augmentations of the same natural data point, and $\boldsymbol{x}, \boldsymbol{x}^-$ are two augmentations generated independently either from the same data point or two different data points. The above loss function is similar to many standard contrastive loss functions (Oord et al., 2018; Sohn, 2016; Wu et al., 2018), including SimCLR (Chen et al., 2020) that we will use in our experiments. Minimizing this objective leads to representations with provable accuracy guarantees under linear probe evaluation. We use $f_{\min}$ to denote the minimizer, i.e., $f_{\min} = \arg\min_f \mathfrak{C}(f)$.

2.2. Training the Linear Head with Label Noise

After obtaining the representations of dimension p by minimizing the contrastive loss, a linear layer parameterized by $\boldsymbol{W} \in \mathbb{R}^{p \times K}$ is trained on the representations. Given a matrix $\boldsymbol{F} \in \mathbb{R}^{n \times p}$ where each row $\boldsymbol{F}_i = f_{\min}(\boldsymbol{x}_i)^\top$ is the learned representation of a data point $\boldsymbol{x}_i$, we consider the downstream task of training a linear model, parameterized by $\boldsymbol{W} \in \mathbb{R}^{p \times K}$, to minimize the MSE loss with l_2 regularization with parameter β

$$\min_{\boldsymbol{W} \in \mathbb{R}^{p \times K}} \|\hat{\boldsymbol{Y}} - \boldsymbol{F}\boldsymbol{W}\|_F^2 + \beta\|\boldsymbol{W}\|. \tag{2}$$

Let $\hat{\boldsymbol{W}}^*$ denote the solution that has the following closed-form expression

$$\hat{\boldsymbol{W}}^* = (\boldsymbol{F}^\top \boldsymbol{F} + \beta \boldsymbol{I})^{-1} \boldsymbol{F}^\top \hat{\boldsymbol{Y}}. \tag{3}$$

While we use MSE in our analysis, we empirically show that our results hold for other losses, such as cross-entropy.

3. Contrastive learning Boosts Robustness

3.1. Provable Robustness of the Linear Head

To theoretically understand the robustness provided by contrastive learning, we assume certain properties of the augmentation graph and analyze the low-rank structure of the the resulting representation matrix. In particular, we utilize the following natural assumptions that formalize the following two properties on the data augmentation: (1) the augmented examples of one sub-class are similar to each other; and (2) the augmented examples of one sub-class are different from the augmented examples of other sub-classes.

Assumption 3.1 (Compact sub-class structure). For a triple of augmented examples $\boldsymbol{x}_j, \boldsymbol{x}_s$ and $\boldsymbol{x}_t$ from the same sub-class, the marginal probability of $\boldsymbol{x}_s, \boldsymbol{x}_j$ being generated from a natural data point is close to that of $\boldsymbol{x}_t, \boldsymbol{x}_j$. Formally, we have $w_{\boldsymbol{x}_s \boldsymbol{x}_j}/w_{\boldsymbol{x}_t \boldsymbol{x}_j} \in [\frac{1}{1+\delta}, 1+\delta]$, for small $\delta \in [0,1)$.

Assumption 3.2 (Distinguishable sub-class structure). For two pairs of augmentated examples $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ and $(\boldsymbol{x}_s, \boldsymbol{x}_t)$ where $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ are from different sub-classes and $\boldsymbol{x}_s$, $\boldsymbol{x}_t$ are from the same sub-class, the marginal probability of $\boldsymbol{x}_i$, $\boldsymbol{x}_j$ being generated from a natural data is much smaller than that of $\boldsymbol{x}_s$, $\boldsymbol{x}_t$. Formally, we have $w_{\boldsymbol{x}_i \boldsymbol{x}_j}/w_{\boldsymbol{x}_s \boldsymbol{x}_t} \leq \xi$, for small $\xi \in$ [0,1).

The above assumptions result in an augmentation graph where augmented data points from different subclasses form nearly disconnected subgraphs with similar edge weights. In particular for $\xi = 0$, we get diconnected subgraph structure.

Desirable Properties of Contrastive Representations

The key to our analysis is that, based on compact and distinguishable sub-class structure assumptions 3.1, 3.2, contrastive learning produces a low-rank representation matrix $\boldsymbol{F}$ that captures the sub-class structure. More formally, the representation matrix has $\bar{K}$ singular vectors that align well with the ground-truth labels, and the corresponding $\bar{K}$ singular values are significant larger than the other singular values. The following theorem is a summary of Lemmas A.2 A.3 B.2 B.3 and Corollary A.6 in the Appendix which details the desirable properties of the representation matrix.

Theorem 3.3. *Having $\bar{K}$ compact and distinguishable sub-classes in the data, the representation matrix $\boldsymbol{F}$ learned by contrastive learning has $\bar{K}$ prominent singular values of magnitude $\mathcal{O}(1)$. At the same time, the sum of the remaining singular values is significantly smaller, i.e., $\mathcal{O}(\sqrt{\delta} + \xi)$. Furthermore, the most prominent $\bar{K}$ singular vectors and the ground-truth labels has a $\mathcal{O}(1)$ alignment, measured by the normalized projection of the clean labels $\boldsymbol{Y}$ onto the*

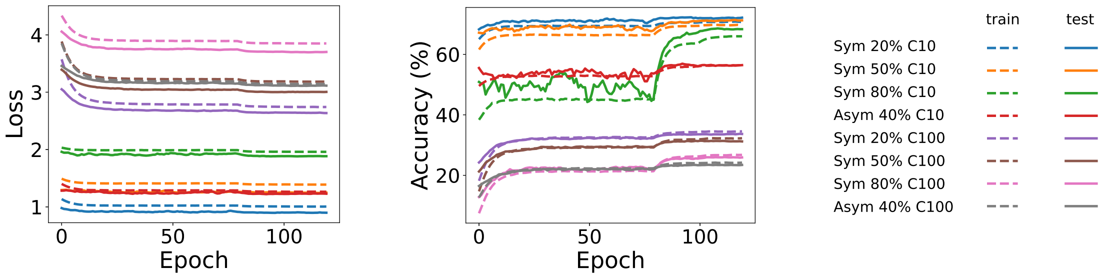


Figure 1. Training accuracy w.r.t. ground-truth labels and test accuracy of a linear classifier trained on representations learned by contrastive learning (SimCLR). Experiments are conducted on CIFAR-10 (C10) and CIFAR-100 (C100) under different noise levels. Dashed lines show loss and accuracy on training set w.r.t. ground-truth labels and solid lines show test loss and accuracy.

span of the singular vectors.

Intuitively, the above three properties of the representation matrix affect the downstream training in the following sense: (1) the magnitude of largest singular values determines the speed at which the model evolves as well as the extent to which the model can fit the training data; (2) the alignment between prominent singular vectors and clean labels indicates whether the model evolves in the right direction; and (3) the magnitude of smaller singular values dictates the amount of overfitting. As a result, Theorem 3.3 implies that the model trained on such representation learns mainly the correct information from the training data, which we formally show in Theorems 3.4 and 3.5.

Training performance w.r.t. Ground-truth Labels Reflects Robustness

To simplify the theoretical analysis, instead of studying the generalization performance (usually measured by the expected loss over the data distribution), we will examine the loss and accuracy on the training data w.r.t. ground-truth labels. This strongly correlates with the test accuracy, especially under large noise. We empirically confirm this correlation in Figure 1, where the dashed lines show training loss and training accuracy w.r.t. ground-truth labels, and solid lines show test loss and test accuracy. We clearly see the high correlation between training and performance, in particular under significant levels of label noise.

Gaussian Label Noise

We first consider the case where label noise is generated from a Gaussian distribution. Formally, $\hat{\boldsymbol{Y}} = \boldsymbol{Y} + \Delta\boldsymbol{Y}$, where $\boldsymbol{Y}$ is the clean label matrix containing all the one-hot encoded labels, and $\Delta\boldsymbol{Y}$ is the label noise matrix, where each column drawn independently from $\mathcal{N}(0, \sigma^2\mathbf{I}_n/K)$. We consider this setting first, as it provides the most convenient way to analyze robustness. Here, our analysis mainly aims at breaking down the effect of label perturbations on training dynamics, in terms of bias and variance. This could provide theoretical insights into the benefits of contrastive learning for boosting robustness.

The following theorem bounds the expected error on training data w.r.t. ground-truth labels, and shows how contrastive learning exploits the augmented sub-class structure to improve robustness.

Theorem 3.4. *For a dataset of size n with K classes, $\bar{K}$ balanced compact and distinguishable sub-classes (c.f. assumptions 3.2, 3.1) and labels corrupted with Gaussian noise $\mathcal{N}(0, \sigma^2\mathbf{I}_n/K)$, a linear model trained by minimizing the objective in Eq. (2) with the representations obtained by minimizing contrastive loss in Eq. (1) has the following expected error on the training set w.r.t. the ground-truth labels $\boldsymbol{Y}$:*

$$\mathbb{E}_{\Delta\boldsymbol{Y}}\frac{1}{n}\|\boldsymbol{Y}-\boldsymbol{F}\hat{\boldsymbol{W}}^*\|_F^2 \tag{4}$$

$$\leq \underbrace{(\frac{\beta}{\beta+1})^2+\mathcal{O}(\delta+\xi)}_{\textit{bias}^2}+\underbrace{\sigma^2\frac{\bar{K}}{n}(\frac{1}{\beta+1})^2+\sigma^2\mathcal{O}(\frac{\sqrt{\delta}+\xi}{\beta})}_{\textit{variance}}.$$

We note that the above results can be easily extended to imbalanced sub-class structure.

The proof can be found in the Appendix. The proof follows the intuition discussed in Section 3.1 that the desirable properties of the learned representation benefit the downstream training. In a nutshell, we derive the bound by writing the error in terms of singular values and vectors of $\boldsymbol{F}$ and then applying Theorem 3.3.

In Eq. (4), the error is decomposed into bias and variance. The bias captures the mismatch between the average prediction of the model and the ground-truth labels. It depends on the magnitude of the prominent singular values as well as the alignment of the corresponding singular vectors with the ground-truth labels. Contrastive learning reduces the bias by aligning the first $\bar{K}$ singular vectors with ground-truth labels (Theorem 3.3), thus producing a small second term in the bias. The variance quantifies the sensitivity to label noise, and is controlled by the magnitude of the non-prominent singular values, which is guaranteed to be small by Theorem 3.3. The regularization parameter β appears in both terms and can be tuned as a trade-off between underfitting and overfitting.

With small enough δ and ξ, one can select a small β to not explicitly penalize the variance much. This results in a small bias, and subsequently a small total error. For example, when there exists a β such that $\sqrt{\delta}+\xi \ll \beta \ll 1$, the error $\approx \sigma^2 \frac{\bar{K}}{n}(\frac{1}{\beta+1})^2$, which is the inevitable cost of achieving a small bias, when the representation matrix $\boldsymbol{F}$ has $\bar{K}$ prominent singular values.

Random Label Flipping

Next, we study the case where the label noise $\Delta \boldsymbol{Y} = \hat{\boldsymbol{Y}} - \boldsymbol{Y}$ is generated by flipping a fraction of the clean labels at random. Formally, for an example $\boldsymbol{x}_i$ belongs to class j with $\boldsymbol{y}_i = \boldsymbol{e}_j$, if its label is flipped to class k, we have $\Delta \boldsymbol{y}_i = \boldsymbol{e}_k - \boldsymbol{e}_j$. We introduce the following notations to analyze the case of asymmetric label noise which mimics the real-world noise, where wrong labels are generated in a (sub)class-dependent way. Let $m_{\bar{k}}$ be the number of mislabeled examples in subclass $\bar{k}$, $m_{\bar{k},k}$ be the number of mislabeled examples in subclass $\bar{k}$ that are labeled as class k, and Z_k be the set of sub-classes in class k. We define $c_{k|\bar{k}} := \frac{m_{\bar{k},k}}{m_{\bar{k}}}$ for all $\bar{k}$,k such that $\bar{k} \notin Z_k$, to be the fraction of mislabeled examples in sub-class $\bar{k}$ that are mislabeled as k. We have that $\sum_{k:\ \bar{k}\notin Z_k} c_{k|\bar{k}} = 1,\ \forall \bar{k} \in [\bar{K}]$. When $c_{k|\bar{k}} = \frac{1}{K-1}\ \ \forall k \in [K], \bar{k} \in [\bar{K}]$, the noise is symmetric.

For simplicity we assume $\xi = 0$. The general case of $\xi \geq 0$ requires more involved analysis which we discuss in the Appendix. The following theorem shows that for a dataset with compact and distinguishable sub-class structure the linear classifier trained on the representations obtained by contrastive learning can recover the clean label for all training data.

Theorem 3.5 (Asymmetric Noise)**.** *For a dataset with K classes and $\bar{K}$ compact and distinguishable sub-class structure (c.f. assumptions 3.2 3.1) with $\xi = 0$, let $n_{\min}, n_{\max}$ be the size of the smallest and largest sub-class, and α be the fraction of mislabeled examples in the training set. Let $c_{\max} := \max_{k\in[K],\bar{k}\in[\bar{K}]} c_{k|\bar{k}} \in [\frac{1}{K-1}, 1]$ be the maximum fraction of wrong labels in a subclass that are flipped to another class. Then as long as*

$$\alpha < \frac{1}{1+\frac{n_{\max}}{n_{\min}}c_{\max}} - \mathcal{O}\left(\frac{\sqrt{\delta}}{\beta}\right), \tag{5}$$

a linear model trained by minimizing the objective in Eq. (2) *with the representations obtained by minimizing contrastive loss in Eq.* (1) *can predict the ground-truth labels for all training examples, i.e.,*

$$\forall i,\ \ \underset{j\in[K]}{\arg\max}(\boldsymbol{F}\hat{\boldsymbol{W}})_{i,j} {=} \arg_{j\in[K]}(\boldsymbol{Y}_{i,j} = 1)$$

In other words contrastive learning can prevent the linear model from memorizing any wrong label even under large noise. Theorem 3.5 also shows that the model can tolerate more noise when the sub-class structure is more compact, i.e., δ is smaller, or the noise is more symmetric, or the sub-classes are more balanced. The following corollary for symmetric noise is simply obtained by setting $c_{\max} = \frac{1}{K-1}$ in Theorem 3.5.

Corollary 3.6 (Symmetric Noise)**.** *For symmetric noise, under the same assumption as in Theorem 3.5, as long as*

$$\alpha < \frac{K-1}{K+\frac{n_{\max}}{n_{\min}}-1} - \mathcal{O}\left(\frac{\sqrt{\delta}}{\beta}\right), \tag{6}$$

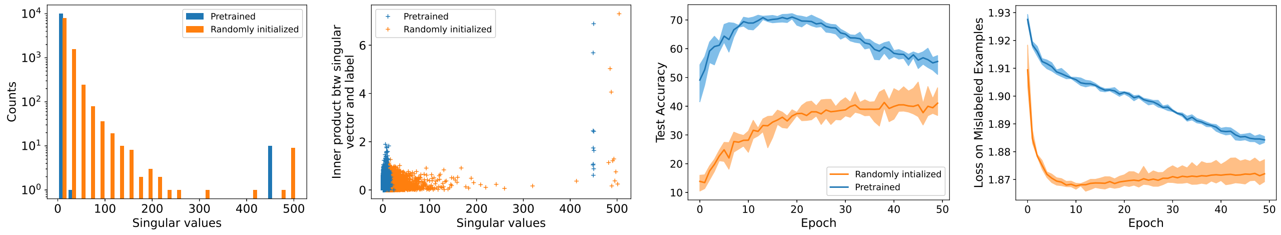
a linear model trained by minimizing the objective in Eq. (2) *with the representations obtained by minimizing contrastive loss in Eq.* (1) *can predict the ground-truth labels for all training examples.*

If we further let $\delta = 0$ and $n_{\max}/n_{\min} = 1$, we get $(K-1)/K$ noise tolerance. We note that this, however, does not imply that a dataset with more classes necessarily has a higher noise tolerance. In Appendix B.1, we show that less distinguishable sub-class structure, i.e. $\xi > 0$, introduces a $\mathcal{O}(\bar{K}^{5/2}\xi)$ perturbation to the singular values and a $\mathcal{O}(\bar{K}^{5/2}\xi)$ rotation in the direction of singular vectors of the representation matrix. Datasets with more classes usually contains more sub-classes, which greatly reduces the noise tolerance. This is also reflected by our empirical results (Figure 1) where the performance of the linear model is worse on CIFAR-100 compared to CIFAR-10 under the same noise level.

3.2. Contrastive Learning Slows down Overfitting for Fine-tuning

In the previous section, we showed that training a linear model on representations learned by contrastive learning is provably robust. Here, we study fine-tuning all layers of the deep network. Interestingly, as is shown in Fig. 2c, finetuning achieves a very high test accuracy under 80% label noise in the early phase of training.

Recall that the theoretical guarantee for linear model (theorems 3.4 and 3.5) is obtained by examining singular values and singular vectors of $\boldsymbol{F}$. Here, we use a similar idea to understand benefits of contrastive learning on robustness when all the layers are trained. Intuitively, during the early stage of training, it is natural to assume that the gradient does not considerably change, and therefore the model is



(a) (b) (c) (d)

Figure 2. Jacobian spectrum and dynamics of training a randomly initialized vs. fine-tuning a pre-trained ResNet32 on CIFAR-10 with 80% randomly flipped labels. (a) distribution of singular values of the initial Jacobian, (b) alignment of the clean labels with the initial Jacobian, (c) test accuracy and (d) loss on mislabeled data points within the first 50 epochs. While pre-training does not improve the alignment of the Jacobian with the labels, it significantly slows down overfitting at the beginning by shrinking the smaller singular values of the Jacobian matrix.

nearly linear. In this case, the initial Jacobian matrix plays the same role as the representation matrix, $\boldsymbol{F}$, to the linear model. This is supported by the recent studies suggesting the following properties of training neural networks: the early learning dynamics can by mimicked by training a linear model (Hu et al., 2020), SGD on neural networks learns a linear model first (Kalimeris et al., 2019), and a network that provides a better alignment between prominent directions of the Jacobian matrix and the label vector is more likely to generalize well (Oymak et al., 2019).

We examine the SVD of the Jacobian of a ResNet pretrained with contrastive learning and compare it to that of a randomly initialized network. Fig. 2a, 2b present the distribution of singular values and the alignment of singular vectors with clean labels. The Jacobian is computed on a random sample of 1000 data points from CIFAR10. Interestingly, Fig. 2b shows that while pre-training does not considerably improve (in Appendix C we show a slight improvement) the alignment between singular vectors of the Jacobian and the clean label vector, it greatly shrinks the smaller singular values of the Jacobian, as is illustrated by Fig. 2a. As a result, it takes substantially longer for the pre-trained network to overfit the noisy labels. As Fig. 2d shows, while a randomly initialized network experience a sharp drop in loss of noisy labeled data points during the first few epochs of training, it takes much longer for a pre-trained network to overfit the noise. Details of the experiment can be found in Appendix E.

3.3. Contrastive Learning Boosts Robust Methods

As discussed, pre-training the network with contrastive learning effectively shrinks the smaller singular values of the Jacobian and slows down overfitting the noisy labels. The following experiments show that the initial level of robustness provided by contrastive learning can be leveraged by existing robust training methods to achieve a superior performance under extreme noise levels. In particular, we consider three methods that prevent overfitting through regularization, namely, Mixup (Zhang et al., 2017) and ELR (Liu et al., 2020), or identifying clean examples, namely, CRUST (Mirzasoleiman et al., 2020).

For the datasets we use artificially corrupted versions of CIFAR-10 and CIFAR-100 (Krizhevsky & Hinton, 2009), as well as a subset of the real-world dataset Webvision (Li et al., 2017), which naturally contains noisy labels. Our method was developed using PyTorch (Paszke et al., 2017). We use 1 Nvidia A40 for all experiments. All baselines and training setups are described in Appendix D.

Empirical results on artificially corrupted CIFAR We first evaluate our method on CIFAR-10 and CIFAR-100 with the same testing protocol as (Xu et al., 2019; Liu et al., 2020; Xia et al., 2019). We consider symmetric noise ratios of 0.2, 0.5, 0.8, and an asymmetric noise ratio of 0.4. In the experiments, we first pre-train ResNet-32 (He et al., 2016) using SimCLR (Chen et al., 2020; SimCLR), then add a linear layer on top of the learned encoder and train the whole network. We also report the results when ELR, Mixup, and Crust are applied

The results are shown in Table 1. We note that SimCLR pretraining leads to an across the board improvement for Crust, ELR, and Mixup. For lower noise ratios, the improvement is marginal. However, for extreme noise ratios, the improvement is more dramatic. In particular, pre-training boosts the performance of Crust by up to 44.1%, ELR by up to 8.2%, and Mixup by up to 34.1% under 80% noise. We also note that under 80% noise, SimCLR pretraining alone outperforms all methods without SimCLR pretraining on CIFAR-10 and CIFAR-100.

Empirical Results on WebVision WebVision is a large scale image dataset with noisy labels (Li et al., 2017). It contains 2.4 million images crawled from Google Images search and Flickr that share the same 1000 classes as the ImageNet dataset.

Table 1. Average test accuracy (3 runs) on CIFAR-10 and CIFAR-100. The best test accuracy is marked in bold. We note the higher performance of methods that use SimCLR (CL) pretraining, especially under higher noise levels. In particular, under 80% noise, methods see an average of 27.18%, and 15.58% increase in test accuracy for CIFAR-10 and CIFAR-100 respectively. Results marked with (*) are reproduced from publicly available code. E2E refers to end to end fine-tuning the pre-trained network.

Dataset	CIFAR-10				CIFAR-100			
Noise Type	Sym			Asym	Sym			Asym
Noise Ratio	20	50	80	40	20	50	80	40
F-correction	85.1 ± 0.4	76.0 ± 0.2	34.8 ± 4.5	83.6 ± 2.2	55.8 ± 0.5	43.3 ± 0.7	–	42.3 ± 0.7
Decoupling	86.7 ± 0.3	79.3 ± 0.6	36.9 ± 4.6	75.3 ± 0.8	57.6 ± 0.5	45.7 ± 0.4	–	43.1 ± 0.4
Co-teaching	89.1 ± 0.3	82.1 ± 0.6	16.2 ± 3.2	84.6 ± 2.8	64.0 ± 0.3	52.3 ± 0.4	–	47.7 ± 1.2
MentorNet	88.4 ± 0.5	77.1 ± 0.4	28.9 ± 2.3	77.3 ± 0.8	63.0 ± 0.4	46.4 ± 0.4	–	42.4 ± 0.5
D2L	86.1 ± 0.4	67.4 ± 3.6	10.0 ± 0.1	85.6 ± 1.2	12.5 ± 4.2	5.6 ± 5.4	–	14.1 ± 5.8
INCV	89.7 ± 0.2	84.8 ± 0.3	52.3 ± 3.5	86.0 ± 0.5	60.2 ± 0.2	53.1 ± 0.4	–	50.7 ± 0.2
T-Revision	79.3 ± 0.5	78.5 ± 0.6	36.2 ± 1.6	76.3 ± 0.8	52.4 ± 0.3	37.6 ± 0.3	–	32.3 ± 0.4
L_DMI	84.3 ± 0.4	78.8 ± 0.5	20.9 ± 2.2	84.8 ± 0.7	56.8 ± 0.4	42.2 ± 0.5	–	39.5 ± 0.4
Crust*	85.3 ± 0.5	86.8 ± 0.3	33.8 ± 1.3	76.7 ± 3.4	62.9 ± 0.3	55.5 ± 1.1	18.5 ± 0.8	52.5 ± 0.4
Mixup	89.7 ± 0.7	84.5 ± 0.3	40.7 ± 1.1	86.3 ± 0.1	64.0 ± 0.4	53.4 ± 0.5	15.1 ± 0.1	54.4 ± 2.0
ELR*	90.6 ± 0.6	87.7 ± 1.0	69.5 ± 5.0	86.6 ± 2.9	63.6 ± 1.7	52.5 ± 4.2	23.4 ± 1.9	59.7 ± 0.1
CL+E2E*	88.8 ± 0.5	82.8 ± 0.2	72.0 ± 0.3	83.5 ± 0.5	63.5 ± 0.2	56.1 ± 0.3	**36.7 ± 0.3**	52.4 ± 0.2
CL+Crust*	86.5 ± 0.7	87.6 ± 0.3	**77.9 ± 0.3**	85.9 ± 0.4	63.0 ± 0.8	**58.3 ± 0.1**	34.8 ± 1.5	53.3 ± 0.7
CL+Mixup*	90.8 ± 0.2	84.6 ± 0.4	74.8 ± 0.3	87.5 ± 1.3	64.4 ± 0.4	55.5 ± 0.1	30.3 ± 0.4	55.5 ± 0.8
CL+ELR*	**91.3 ± 0.0**	**89.1 ± 0.1**	77.7 ± 0.2	**89.7 ± 0.3**	**64.7 ± 0.2**	55.6 ± 0.2	35.9 ± 0.3	**63.6 ± 0.1**

Table 2. Test accuracy on mini WebVision. The best test accuracy is marked in bold. SimCLR (CL) pre-training leads to average improvements of 4.11% and 3.20% for mini Webvision and ImageNet respectively.

	WebVision		ImageNet	
Method	Top-1	Top-5	Top-1	Top-5
F-correction	61.12	82.68	57.36	82.36
Decoupling	62.54	84.74	58.26	82.26
Co-teaching	63.58	85.20	61.48	84.70
MentorNet	63.00	81.40	57.80	79.92
D2L	62.68	84.00	57.80	81.36
INCV	65.24	85.34	61.60	84.98
Crust	72.40	89.56	67.36	87.84
Mixup	71.38	87.36	68.34	88.44
ELR	76.26	91.26	68.71	87.84
CL + E2E	71.84	88.84	68.48	89.32
CL + Mixup	76.34	90.52	**72.25**	**89.72**
CL + ELR	**79.52**	**93.80**	71.20	90.80

The noise ratio in classes varies from 0.5% to 88%, and the number of images per class varies from 300 to more than 10,000 (Fig. 4 in (Li et al., 2017) shows the noise distribution). We follow the setting in (Jiang et al., 2018) and create a mini WebVision dataset that consists of the top 50 classes in the Google subset with 66,000 images. We use both WebVision and ImageNet test sets for testing the performance of the model. We train a InceptionResNet-v2 (Szegedy et al., 2017).

Table 2 shows the Top-1 and Top-5 accuracy of different methods evaluated on WebVision and ImageNet. We see that for both ELR and Mixup, SimCLR pretraining leads to average improvements of 4.11% and 3.20% for mini Webvision and ImageNet respectively. Furthermore, we note that SimCLR pretraining on its own outperforms every method without SimCLR pretraining, except ELR and Crust.

4. Conclusion

We showed that representations learned by contrastive learning provably boosts robustness against noisy labels. In particular, contrastive learning provides a representation matrix that has: (i) a significant gap between the prominent singular values and the remaining ones; (ii) a large alignment between the prominent singular vectors and the clean labels. The above properties allow a linear layer trained on the representations to effectively learn the clean labels well while barely overfitting the noise. Then we explained why fine-tuning all layers of a network pre-trained with contrastive learning can also achieve a good performance in the early phase. Crucially, contrastive learning greatly reduces the magnitude of nonprominant singular values of the initial Jacobian matrix, which slows down the overfitting. Finally, we demonstrated that the initial robustness provided by contrastive learning can boost robust methods and achieve state-of-the-art performance under extreme noise levels. Our results confirm benefits of contrastive pretraining for robust machine learning.

Acknowledgements

This research was supported in part by Cisco Systems and UCLA-Amazon Science Hub for Humanity and Artificial Intelligence.

References

Chen, P., Liao, B. B., Chen, G., and Zhang, S. Understanding and utilizing deep neural networks trained with noisy labels. In *International Conference on Machine Learning*, pp. 1062–1070, 2019.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Eckart, C. and Young, G. The approximation of one matrix by another of lower rank. *Psychometrika*, 1(3):211–218, 1936.

Ghosh, A. and Lan, A. Contrastive learning improves model robustness under label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2703–2708, 2021.

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., and Sugiyama, M. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in neural information processing systems*, pp. 8527–8537, 2018.

HaoChen, J. Z., Wei, C., Gaidon, A., and Ma, T. Provable guarantees for self-supervised deep learning with spectral contrastive loss. *arXiv preprint arXiv:2106.04156*, 2021.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Hendrycks, D., Mazeika, M., Kadavath, S., and Song, D. Using self-supervised learning can improve model robustness and uncertainty. *arXiv preprint arXiv:1906.12340*, 2019.

Hu, W., Xiao, L., Adlam, B., and Pennington, J. The surprising simplicity of the early-time learning dynamics of neural networks. *Advances in Neural Information Processing Systems*, 33:17116–17128, 2020.

Jiang, L., Zhou, Z., Leung, T., Li, L.-J., and Fei-Fei, L. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels. In *International Conference on Machine Learning*, pp. 2309–2318, 2018.

Kalimeris, D., Kaplun, G., Nakkiran, P., Edelman, B., Yang, T., Barak, B., and Zhang, H. Sgd on neural networks learns functions of increasing complexity. *Advances in neural information processing systems*, 32, 2019.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. Technical report, Citeseer, 2009.

Li, W., Wang, L., Li, W., Agustsson, E., and Van Gool, L. Webvision database: Visual learning and understanding from web data. *arXiv preprint arXiv:1708.02862*, 2017.

Liu, S., Niles-Weed, J., Razavian, N., and Fernandez-Granda, C. Early-learning regularization prevents memorization of noisy labels. *arXiv preprint arXiv:2007.00151*, 2020.

Ma, X., Wang, Y., Houle, M. E., Zhou, S., Erfani, S., Xia, S., Wijewickrema, S., and Bailey, J. Dimensionality-driven learning with noisy labels. In *International Conference on Machine Learning*, pp. 3355–3364, 2018.

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

Minc, H. On the maximal eigenvector of a positive matrix. *SIAM Journal on Numerical Analysis*, 7(3):424–427, 1970.

Mirzasoleiman, B., Cao, K., and Leskovec, J. Coresets for robust training of deep neural networks against noisy labels. *Advances in Neural Information Processing Systems*, 33, 2020.

Oord, A. v. d., Li, Y., and Vinyals, O. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

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

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., and Lerer, A. Automatic differentiation in pytorch. 2017.

Patrini, G., Rozza, A., Krishna Menon, A., Nock, R., and Qu, L. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1944–1952, 2017.

SimCLR. https://github.com/spijkervet/simclr.

Sohn, K. Improved deep metric learning with multi-class n-pair loss objective. In *Advances in neural information processing systems*, pp. 1857–1865, 2016.

Stewart, G. W. Matrix perturbation theory. 1990.

Szegedy, C., Ioffe, S., Vanhoucke, V., and Alemi, A. A. Inception-v4, inception-resnet and the impact of residual connections on learning. In *Thirty-first AAAI conference on artificial intelligence*, 2017.

Wedin, P.-Å. Perturbation bounds in connection with singular value decomposition. *BIT Numerical Mathematics*, 12(1):99–111, 1972.

Wu, Z., Xiong, Y., Yu, S. X., and Lin, D. Unsupervised feature learning via non-parametric instance discrimination. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3733–3742, 2018.

Xia, X., Liu, T., Wang, N., Han, B., Gong, C., Niu, G., and Sugiyama, M. Are anchor points really indispensable in label-noise learning? In *Advances in Neural Information Processing Systems*, pp. 6838–6849, 2019.

Xu, Y., Cao, P., Kong, Y., and Wang, Y. L_dmi: A novel information-theoretic loss function for training deep nets robust to label noise. In *Advances in Neural Information Processing Systems*, pp. 6225–6236, 2019.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. *arXiv preprint arXiv:1710.09412*, 2017.

Zheltonozhskii, E., Baskin, C., Mendelson, A., Bronstein, A. M., and Litany, O. Contrast to divide: Self-supervised pre-training for learning with noisy labels. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1657–1667, 2022.

A. Analysis for Disconnected Subclasses

In this section we consider the case where $\xi = 0$ in assumption 3.2, which implies that the probability of two augmentation data from different subclasses being generated from the same random natural datum is exactly zero. And in section B we extend the results to any $\xi \in [0, 1)$ via eigenvalue and eigenvector perturbation. We use $\| \cdot \|_1$, $\| \cdot \|_2$ and $\|\|_F$ to denote the 1-norm, operator norm and Frobenius norm, respectively.

A.1. Spectral Decomposition of Augmentation Graph

An important technical idea we use to formalize the representations obtained by contrastive learning is augmentation graph (HaoChen et al., 2021), which is an undirected graph with all augmentation data $\{\boldsymbol{x}_1, \boldsymbol{x}_2, \ldots, \boldsymbol{x}_n\}$ as its vertices and $w_{\boldsymbol{x}_i\boldsymbol{x}_j}$ as the weight for edge $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Let $\boldsymbol{A}$ denote the adjacency matrix of the augmentation graph, that is, each entry $a_{ij} = w_{\boldsymbol{x}_i\boldsymbol{x}_j}$. And the normalized adjacency matrix $\overline{\boldsymbol{A}}$ is defined as

$$\overline{\boldsymbol{A}} := \boldsymbol{D}^{-1/2}\boldsymbol{A}\boldsymbol{D}^{-1/2},$$

where $\boldsymbol{D} = \mathrm{diag}(w_{\boldsymbol{x}_1}, w_{\boldsymbol{x}_2}, \ldots, w_{\boldsymbol{x}_n})$ with $w_{\boldsymbol{x}_i} = \sum_{j=1}^{n} w_{\boldsymbol{x}_i\boldsymbol{x}_j}$. For simplicity we index the augmentation data in the following way: the first n_1 data are from subclass 1, the next n_2 data are from subclass 2, ..., the last $n_{\bar K}$ data are from subclass $\bar K$. Lemma A.1 states an important property of $\overline{\boldsymbol{A}}$.

Lemma A.1. *Assumption 3.2 with $\xi = 0$ implies that $\overline{\boldsymbol{A}}$ is a block diagonal matrix $\boldsymbol{diag}(\overline{\boldsymbol{A}}_1, \overline{\boldsymbol{A}}_2, \ldots, \overline{\boldsymbol{A}}_{\bar K})$ where $\overline{\boldsymbol{A}}_{\bar k} \in \mathbb{R}^{n_{\bar k}\times n_{\bar k}}$. This combined with 3.1 gives us: for each block $\overline{\boldsymbol{A}}_{\bar k}$, the ratio between two entries in the same column is bounded as follows*

$$\forall \bar k \in [\bar K], \max_{j,s,t} \frac{\bar a_{\bar k,s,j}}{\bar a_{\bar k,t,j}} \le 1 + \delta',$$

where $\bar a_{k,i,j}$ are the entries of $\overline{\boldsymbol{A}}_{\bar k}$ and $\delta' = (1+\delta)^{3/2} - 1 = \frac{3}{2}\delta + \mathcal{O}(\delta^2)$.

Let $f_{\min} = \arg\min_f \mathfrak{C}(f)$ and $\boldsymbol{F}_{\min} = [f_{\min}(\boldsymbol{x}_1)\ f_{\min}(\boldsymbol{x}_2) \ldots f_{\min}(\boldsymbol{x}_n)]^\top$, according the theorem in (HaoChen et al., 2021), $\boldsymbol{F}_{\min}$ is also the minimizer of the following matrix factorization problem

$$\min_{\boldsymbol{F}\in\mathbb{R}^{n\times p}} \|\overline{\boldsymbol{A}} - \boldsymbol{F}\boldsymbol{F}^\top\|_F^2, \tag{7}$$

and therefore can be further decomposed as

$$\boldsymbol{F}_{\min} = \boldsymbol{F}^*\boldsymbol{\Sigma}\boldsymbol{R}, \tag{8}$$

by Eckart–Young–Mirsky theorem (Eckart & Young, 1936), where $\boldsymbol{F}^* \in \mathbb{R}^{n\times p} = [\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_p] \in \mathbb{R}^{n\times p}$, $\boldsymbol{\Sigma} = \mathbf{diag}(\sqrt{\lambda_1}, \sqrt{\lambda_2}, \ldots, \sqrt{\lambda_p})$, $\boldsymbol{R} \in \mathbb{R}^{p\times p}$ is some orthogonal matrix, $\lambda_1, \lambda_2, \ldots, \lambda_p$ are the p largest eigenvalues of $\overline{\boldsymbol{A}}$ and $\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_p$ are the corresponding unit-norm eigenvectors. W.l.o.g., we assume $\lambda_1 \ge \lambda_2 \ge \cdots \ge \lambda_p$. Our following proofs are based on this decomposition. To avoid cluttered notation we drop the subscript of $\boldsymbol{F}_{\min}$, i.e., we use $\boldsymbol{F}$ for the *learned* representation.

A.2. Properties of Singular Values/Vectors of the Representation Matrix

From the above section we know that the singular values/vectors of $\boldsymbol{F}$ are the first p eigenvalues/vectors of $\overline{\boldsymbol{A}}$. For each block $\overline{\boldsymbol{A}}_{\bar k}$, let $\lambda_{\bar k,1}, \lambda_{\bar k,2}, \ldots, \lambda_{\bar k,n_{\bar k}}$ denote the eigenvalues (in descending order) and $\boldsymbol{v}_{\bar k,1}, \boldsymbol{v}_{\bar k,2}, \ldots, \boldsymbol{v}_{\bar k,n_{\bar k}}$ denote the corresponding eigenvectors. The eigenvalues of $\overline{\boldsymbol{A}}$ are the list of the eigenvalues of all blocks. The corresponding eigenvectors are the block vectors $(\vec{0}_1, \vec{0}_2, \ldots, \vec{0}_{\bar k-1}, \boldsymbol{v}_{\bar k,i}, \vec{0}_{\bar k+1}, \ldots, \vec{0}_{\bar K}) := \dot{\boldsymbol{v}}_{\bar k,i}$ with each $\vec{0}_j$ being a zero vector of length n_j. Since $\overline{\boldsymbol{A}}$ is a normalized adjacency matrix, each block $\overline{\boldsymbol{A}}_{\bar k}$ is also normalized. Then the largest eigenvalue for each block is exactly 1, i.e., $\lambda_{\bar k,1} = 1$. It immediately follow Lemma A.2.

Lemma A.2. *The $\bar K$ largest eigenvalues of $\overline{\boldsymbol{A}}$ are all 1, i.e., $\lambda_1 = \lambda_2 = \cdots = \lambda_{\bar K} = 1$.*

As long as $p \ge \bar K$, all $\lambda_{\bar k,1}$ and $\boldsymbol{v}_{\bar k,1}$ appear in the decomposition of $\boldsymbol{F}$. Let $p_{\bar k} \ge 1$ be the number of eigenvalues/eigenvectors of block $\overline{\boldsymbol{A}}_{\bar k}$ that also appear in the decomposition of $\boldsymbol{F}$. The following Lemmas and Corollaries states other important properties of eigenvalues/eigenvectors of $\overline{\boldsymbol{A}}$.

Lemma A.3. *By assumption 3.1, the* 1*-norm of* $\boldsymbol{v}_{\bar{k},1}$ *has the following lower bound.*

$$\|\boldsymbol{v}_{\bar{k},1}\|_1^2 \geq \frac{n_{\bar{k}}^2}{1+(n_{\bar{k}}-1)(1+\delta')^2} = n_{\bar{k}} - 2(n_{\bar{k}}-1)\delta' - \mathcal{O}(\delta'^2).$$

Proof. Write $\boldsymbol{v}_{\bar{k},1}$ as $[e_1, e_2, \dots, e_{n_{\bar{k}}}]^T$. By Perron-Frobenius theorem, all the elements here are positive since $\overline{\boldsymbol{A}}_{\bar{k}}$ is a positive matrix. Then the quotient of any two elements in $\boldsymbol{v}_{\bar{k},1}$ can be bounded in terms of the entries of $\overline{\boldsymbol{A}}_{\bar{k}}$ (Minc, 1970) and therefore $1+\delta'$ by lemma A.1:

$$\max_{i,j} \frac{e_i}{e_j} \leq \max_{j,s,t} \frac{\bar{a}_{\bar{k},s,j}}{\bar{a}_{\bar{k},t,j}} \leq 1+\delta'.$$

Let $e_{min} = \min_i e_i$, we have

$$1 = \|\boldsymbol{v}_{\bar{k},1}\|_2^2 = \sum_{i=1}^{n_{\bar{k}}} e_i^2 \leq e_{min}^2 + (n_{\bar{k}}-1)(1+\delta')^2 e_{min}^2,$$

and

$$\|\boldsymbol{v}_{\bar{k},1}\|_1^2 = (\sum_{i=1}^{n_{\bar{k}}} e_i)^2 \geq n_{\bar{k}}^2 e_{min}^2.$$

Combining the preceding two equations yields

$$\|\boldsymbol{v}_{\bar{k},1}\|_1^2 \geq \frac{n_{\bar{k}}^2}{1+(n_{\bar{k}}-1)(1+\delta')^2}.$$

□

Lemma A.4. *The sum of squared eigenvalues of each block* $\overline{\boldsymbol{A}}_{\bar{k}}$ *can be bounded.*

$$\sum_{i=1}^{n_{\bar{k}}} \lambda_{\bar{k},i}^2 \leq \frac{(1+(n_{\bar{k}}-1)(1+\delta')^2)(1+\delta')^2}{n_{\bar{k}}}.$$

Proof. First we have

$$\sum_{i=1}^{n_{\bar{k}}} \lambda_{\bar{k},i}^2 = \|\overline{\boldsymbol{A}}_{\bar{k}}\|_F^2 = \sum_{j=1}^{n_{\bar{k}}} \|\boldsymbol{c}_{\bar{k},j}\|_2^2, \tag{9}$$

where $\boldsymbol{c}_{\bar{k},j} = [\bar{a}_{\bar{k},1,j}, \bar{a}_{\bar{k},2,j}, \dots, \bar{a}_{\bar{k},n_{\bar{k}},j}]^T$ denotes the j-th column in $\overline{\boldsymbol{A}}_{\bar{k}}$. Analogous to lemma A.3, here we can bound $\|\boldsymbol{c}_{\bar{k},j}\|_2^2$ in terms of $\|\boldsymbol{c}_{\bar{k},j}\|_1^2$ by lemma A.1

$$\|\boldsymbol{c}_{\bar{k},j}\|_2^2 \leq \frac{(1+(n_{\bar{k}}-1)(1+\delta')^2)\|\boldsymbol{c}_{\bar{k},j}\|_1^2}{n_{\bar{k}}^2}. \tag{10}$$

We also have

$$\|\boldsymbol{c}_{\bar{k},j}\|_1^2 \leq \max_j \|\boldsymbol{c}_{\bar{k},j}\|_1^2 \leq (1+\delta')^2 \min_j \|\boldsymbol{c}_{\bar{k},j}\|_1^2 \leq (1+\delta')^2 \lambda_{\bar{k},1}^2 = (1+\delta')^2, \tag{11}$$

where the second inequality holds because of assumption 3.1 and the third inequality holds because of Perron-Frobenius theorem. Combining equations 9, 10 and 11 completes the proof. □

Corollary A.5. *The eigenvalues except the* $\bar{K}$ *largest ones are each upper bounded by*

$$\lambda_{\bar{k},i} \;\leq\; \sqrt{\frac{(1+(n_{\bar{k}}-1)(1+\delta')^2)(1+\delta')^2}{n_{\bar{k}}} - 1} = \; \mathcal{O}(\sqrt{\delta'}), \quad \forall i = 2,3,\dots,n_{\bar{k}}, \quad \forall \bar{k} \in [\bar{K}].$$

Proof. By Lemmas A.2 and A.4

$$\begin{aligned}\sum_{i=2}^{n_{\bar{k}}}\lambda_{\bar{k},i}^2 =&\sum_{i=1}^{n_{\bar{k}}}\lambda_{\bar{k},i}^2-\lambda_{\bar{k},1}^2\\ \leq&\frac{(1+(n_{\bar{k}}-1)(1+\delta')^2)(1+\delta')^2}{n_{\bar{k}}}-1\\ =&4\delta'+\mathcal{O}(\delta'^2).\end{aligned} \tag{12}$$

Then

$$\begin{aligned}\lambda_{\bar{k},i}\leq&\sqrt{\sum_{i=2}^{n_{\bar{k}}}\lambda_{\bar{k},i}^2}\\ \leq&\sqrt{\frac{(1+(n_{\bar{k}}-1)(1+\delta')^2)(1+\delta')^2}{n_{\bar{k}}}-1}\end{aligned}$$

□

Corollary A.6. *For each block $\overline{\boldsymbol{A}}_{\bar{k}}$, the sum of the eigenvalues from the second to the $p_{\bar{k}}$-th is bounded by*

$$\sum_{i=2}^{p_{\bar{k}}}\lambda_{\bar{k},i}\leq 2\sqrt{(p_{\bar{k}}-1)}\sqrt{\delta'}+\mathcal{O}(\delta').$$

And the sum of eigenvalues of $\overline{\boldsymbol{A}}$ from the $\bar{K}+1$-th to the p-th is bounded by

$$\sum_{i=\bar{K}+1}^{p}\lambda_i\leq 2\sqrt{(p-\bar{K})\bar{K}\delta'}+\mathcal{O}(\delta').$$

Proof. Applying Cauchy–Schwarz inequality to equation 12 yields the bound for $\sum_{i=2}^{p_{\bar{k}}}\lambda_{\bar{k},i}$. Summing both sides of equation 12 over $\bar{k}\in[\bar{K}]$ and then applying Cauchy–Schwarz inequality give the bound for $\sum_{i=\bar{K}+1}^{p}\lambda_i$. □

A.3. Error under Gaussian Noise when $\xi=0$

With the decomposition in Equation 8, the learned parameter of the linear model in Equation 3 can be rewritten as

$$\hat{\boldsymbol{W}}^*=\boldsymbol{R}^\top\textbf{diag}(\frac{\sqrt{\lambda_1}}{\lambda_1+\beta},\frac{\sqrt{\lambda_2}}{\lambda_2+\beta},\dots,\frac{\sqrt{\lambda_p}}{\lambda_p+\beta})\boldsymbol{F}^{*\top}\hat{\boldsymbol{Y}}.$$

The output on the training set $\boldsymbol{F}\hat{\boldsymbol{W}}^*$ is

$$\boldsymbol{F}\hat{\boldsymbol{W}}^*=\boldsymbol{F}^*\boldsymbol{B}\boldsymbol{F}^{*\top}\hat{\boldsymbol{Y}},$$

where $\boldsymbol{B}=\textbf{diag}(b_1,b_2,\dots,b_p)$ with $b_i=\frac{\lambda_i}{\lambda_i+\beta}$. And the error on training set w.r.t. ground-truth labels can be therefore written in terms of the eigenvalues and eigenvectors of $\overline{\boldsymbol{A}}$

$$\begin{aligned}\mathbb{E}_{\Delta\boldsymbol{Y}}\frac{1}{n}\|\boldsymbol{Y}-\boldsymbol{F}\hat{\boldsymbol{W}}^*\|_F^2=&\mathbb{E}_{\Delta\boldsymbol{Y}}\left[\frac{1}{n}\|\boldsymbol{Y}-\boldsymbol{F}^*\boldsymbol{B}\boldsymbol{F}^{*\top}(\boldsymbol{Y}+\Delta\boldsymbol{Y})\|_F^2\right]\\ =&\frac{1}{n}\|\boldsymbol{Y}-\boldsymbol{F}^*\boldsymbol{B}\boldsymbol{F}^{*\top}\boldsymbol{Y}\|_F^2+\mathbb{E}_{\Delta\boldsymbol{Y}}\left[\|\boldsymbol{F}^*\boldsymbol{B}\boldsymbol{F}^{*\top}\Delta\boldsymbol{Y}\|_F^2\right]\\ =&\frac{1}{n}\|\boldsymbol{Y}\|_F^2+\frac{1}{n}\sum_{i=1}^{p}\sum_{j=1}^{K}(b_i^2-2b_i)(\boldsymbol{v}_i^\top\boldsymbol{y}_j)^2+\frac{\sigma^2}{n}\sum_{i=1}^{p}b_i^2\\ =&\underbrace{1+\frac{1}{n}\sum_{i=1}^{p}\sum_{j=1}^{K}(b_i^2-2b_i)(\boldsymbol{v}_i^\top\boldsymbol{y}_j)^2}_{\textbf{bias}^2}+\underbrace{\frac{\sigma^2}{n}\sum_{i=1}^{p}b_i^2}_{\textbf{variance}},\end{aligned} \tag{13}$$

where $\boldsymbol{y}_j \in \mathbb{R}^n$ is the j-th *column* of $\boldsymbol{Y}$.

We first calculate the **bias**^{2} term. We have $b_i^2 - 2b_i \leq 0$ since $b_i \in (0, 1]$. Then we drop items with $i \geq \bar{K} + 1$ in the summation and apply Lemma A.2 to get an upper bound

$$\begin{aligned}
\textbf{bias}^2 \leq & 1 - \frac{1}{n} \sum_{i=1}^{\bar{K}} \sum_{j=1}^{K} (2b_i - b_i^2)(\boldsymbol{v}_i^\top \boldsymbol{y}_j)^2 \\
= & 1 - \frac{1}{n} \left(\frac{2}{1+\beta} - (\frac{1}{1+\beta})^2 \right) \sum_{i=1}^{\bar{K}} \sum_{j=1}^{K} (\boldsymbol{v}_i^\top \boldsymbol{y}_j)^2 \\
= & 1 - \frac{1}{n} \left(\frac{2}{1+\beta} - (\frac{1}{1+\beta})^2 \right) \sum_{\bar{k}=1}^{\bar{K}} \sum_{j=1}^{K} (\dot{\boldsymbol{v}}_{\bar{k},1}^\top \boldsymbol{y}_j)^2
\end{aligned}$$

By Perron-Frobenious theorem all elements in $\boldsymbol{v}_{\bar{k},1}$ are positive, thus the sum of elements of $\boldsymbol{v}_{\bar{k},1}$ can be written as $\|\boldsymbol{v}_{\bar{k},1}\|_1$. With the observation that $\dot{\boldsymbol{v}}_{\bar{k},1}^T \boldsymbol{y}_j = \|\boldsymbol{v}_{\bar{k},1}\|_1$ when $\bar{k}$ is a subclass within class j and otherwise $\dot{\boldsymbol{v}}_{\bar{k},1}^T \boldsymbol{y}_j = 0$, the above equation can be rewritten as

$$\textbf{bias}^2 \leq 1 - \frac{1}{n} \left(\frac{2}{1+\beta} - (\frac{1}{1+\beta})^2 \right) \sum_{\bar{k}=1}^{\bar{K}} \|\boldsymbol{v}_{\bar{k},1}\|_1^2. \tag{14}$$

Then by Lemma A.3,

$$\begin{aligned}
\textbf{bias}^2 \leq & 1 - \frac{1}{n} \left(\frac{2}{1+\beta} - (\frac{1}{1+\beta})^2 \right) \sum_{\bar{k}=1}^{\bar{K}} \left(n_{\bar{k}} - 2(n_{\bar{k}} - 1)\delta' - \mathcal{O}(\delta'^2) \right) \\
= & (\frac{\beta}{1+\beta})^2 + 2(1 - \frac{\bar{K}}{n}) \frac{(2\beta+1)}{(\beta+1)^2} \delta' + \mathcal{O}(\delta'^2) \\
= & (\frac{\beta}{1+\beta})^2 + 3(1 - \frac{\bar{K}}{n}) \frac{(2\beta+1)}{(\beta+1)^2} \delta + \mathcal{O}(\delta^2)
\end{aligned} \tag{15}$$

Now we bound the **variance** term. By Lemma A.2

$$\begin{aligned}
\textbf{variance} = & \frac{\sigma^2}{n} \sum_{i=1}^{p} b_i^2 \\
= & \frac{\sigma^2}{n} \sum_{i=1}^{\bar{K}} b_i^2 + \frac{\sigma^2}{n} \sum_{i=\bar{K}+1}^{p} b_i^2 \\
= & \sigma^2 \frac{\bar{K}}{n} (\frac{1}{\beta+1})^2 + \frac{\sigma^2}{n} \sum_{i=\bar{K}+1}^{p} (1 - \frac{\beta}{\lambda_i + \beta}) \\
= & \sigma^2 \frac{\bar{K}}{n} (\frac{1}{\beta+1})^2 + \frac{\sigma^2}{n} (p - \bar{K}) - \frac{\sigma^2}{n} \sum_{i=\bar{K}+1}^{p} \frac{\beta}{\lambda_i + \beta}.
\end{aligned} \tag{16}$$

Apply Cauchy–Schwarz inequality and Corollary A.6 to bound the summation in the last term

$$\begin{aligned}
\sum_{i=\bar{K}+1}^{p} \frac{\beta}{\lambda_i + \beta} & \geq \frac{\beta(p - \bar{K})^2}{\sum_{i=\bar{K}+1}^{p} (\lambda_i + \beta)} \\
& \geq \frac{\beta(p - \bar{K})^2}{2\sqrt{(p_{\bar{k}} - 1)}\sqrt{\delta'} + \mathcal{O}(\delta') + (p - \bar{K})\beta} \\
& = p - \bar{K} + \frac{2\sqrt{p - \bar{K}}}{\beta} \sqrt{\bar{K}\delta'} + \mathcal{O}(\delta').
\end{aligned}$$

Plugging the preceding into Equation 16 yields

$$
\begin{aligned}
\textbf{variance} \le & \sigma^2 \frac{\bar{K}}{n}(\frac{1}{\beta+1})^2 + \sigma^2 \frac{2\sqrt{\bar{K}(p-\bar{K})}}{n}\frac{\sqrt{\delta'}}{\beta} + \mathcal{O}(\delta') \\
= & \sigma^2 \frac{\bar{K}}{n}(\frac{1}{\beta+1})^2 + \sigma^2 \frac{\sqrt{6\bar{K}(p-\bar{K})}}{n}\frac{\sqrt{\delta}}{\beta} + \mathcal{O}(\delta)
\end{aligned}
$$

A.4. Accuracy under Label Flipping (Proof for Theorem 3.5)

We study the accuracy by looking at the entries of the output $\boldsymbol{F}\hat{\boldsymbol{W}}^*$.

$$
\begin{aligned}
\boldsymbol{F}\hat{\boldsymbol{W}}^* &= \boldsymbol{F}^*\boldsymbol{B}\boldsymbol{F}^{*\top}(\boldsymbol{Y}+\Delta\boldsymbol{Y}) \\
&= \left[\boldsymbol{F}^*\boldsymbol{B}\boldsymbol{F}^{*\top}(\boldsymbol{y}_1+\Delta\boldsymbol{y}_1),\ \boldsymbol{F}^*\boldsymbol{B}\boldsymbol{F}^{*\top}(\boldsymbol{y}_2+\Delta\boldsymbol{y}_2),\ \ldots,\ \boldsymbol{F}^*\boldsymbol{B}\boldsymbol{F}^{*\top}(\boldsymbol{y}_K+\Delta\boldsymbol{y}_K)\right] \\
&= \left[\sum_{i=1}^{p} b_i\boldsymbol{v}_i\boldsymbol{v}_i^\top(\boldsymbol{y}_1+\Delta\boldsymbol{y}_1),\ \sum_{i=1}^{p} b_i\boldsymbol{v}_i\boldsymbol{v}_i^\top(\boldsymbol{y}_2+\Delta\boldsymbol{y}_2),\ \ldots,\ \sum_{i=1}^{p} b_i\boldsymbol{v}_i\boldsymbol{v}_i^\top(\boldsymbol{y}_K+\Delta\boldsymbol{y}_K)\right] \\
&\coloneqq [\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots, \boldsymbol{z}_K]
\end{aligned}
$$

For convenience we define the notations C_k and $S_{\bar{k}}$ as the sets of indices of examples from class k and subclass $\bar{k}$, respectively

$$
\begin{aligned}
C_k \coloneqq & \{i : \boldsymbol{x}_i \text{ belongs to class } k\} \\
S_{\bar{k}} \coloneqq & \{i : \boldsymbol{x}_i \text{ belongs to subclass } \bar{k}\} = \{i : \sum_{j=1}^{\bar{k}-1} n_j < i \le \sum_{j=1}^{\bar{k}} n_j\}.
\end{aligned}
$$

Let the notation $\boldsymbol{\mu}^{(j)}$ denote the j-th element of vector $\boldsymbol{\mu}$. Then $\boldsymbol{z}_k^{(j)}$ can be written as $\sum_{i=1}^{p} b_i\boldsymbol{v}_i^{(j)}\boldsymbol{v}_i^\top(\boldsymbol{y}_k+\Delta\boldsymbol{y}_k)$. Let $\bar{k}_j$ denote the subclass that $\boldsymbol{x}_j$ belongs to, i.e., $j \in S_{\bar{k}}$ and define $e_{\min,s} \coloneqq \min_j \boldsymbol{v}_{\bar{k}_j,1}^{(s)}$ and $e_{\max,s} \coloneqq \max_s \boldsymbol{v}_{\bar{k}_j,1}^{(s)}$. We have the following two lemmas.

Lemma A.7. $\boldsymbol{z}_k^{(j)}$ *can be bounded*

$$
\begin{cases}
\boldsymbol{z}_k^{(j)} \ge & \frac{1}{\beta+1} e_{\min,j}\dot{\boldsymbol{v}}_{\bar{k}_j,1}^{(j)} n_{\min}(1-\alpha) - \sqrt{n_{\max}}\frac{2\sqrt{p-1}}{\beta}\sqrt{\delta'} - \mathcal{O}(\delta'), \quad j \in C_k \\
\boldsymbol{z}_k^{(j)} \le & \frac{1}{\beta+1} e_{\max,j}\dot{\boldsymbol{v}}_{\bar{k}_j,1}^{(j)} n_{\max}\alpha c_{\max} + \sqrt{n_{\max}c_{\max}}\frac{2\sqrt{p-1}}{\beta}\sqrt{\delta'} + \mathcal{O}(\delta'), \quad j \notin C_k.
\end{cases}.
$$

Proof. Let $b_{\bar{k},i}$ denote $\frac{\lambda_{\bar{k},i}}{\beta+\lambda_{\bar{k},i}}$. Recalling that one property of the block vector $\dot{\boldsymbol{v}}_{\bar{k},i}$ is that $\dot{\boldsymbol{v}}_{\bar{k},i}^{(j)} = 0$ when $j \notin S_{\bar{k}}$, we have

$$
\begin{aligned}
\boldsymbol{z}_k^{(j)} &= \sum_{i=1}^{p} b_i\boldsymbol{v}_i^{(j)}\boldsymbol{v}_i^\top(\boldsymbol{y}_k+\Delta\boldsymbol{y}_k) \\
&= \sum_{\bar{k}=1}^{\bar{K}}\sum_{i=1}^{p_{\bar{k}}} b_{\bar{k},i}\dot{\boldsymbol{v}}_{\bar{k},i}^{(j)}\dot{\boldsymbol{v}}_{\bar{k},i}^\top(\boldsymbol{y}_k+\Delta\boldsymbol{y}_k) \\
&= \sum_{i}^{p_{\bar{k}_j}} b_{\bar{k}_j,i}\dot{\boldsymbol{v}}_{\bar{k}_j,i}^{(j)}\dot{\boldsymbol{v}}_{\bar{k}_j,i}^\top(\boldsymbol{y}_k+\Delta\boldsymbol{y}_k) \\
&= b_{\bar{k}_j,1}\dot{\boldsymbol{v}}_{\bar{k}_j,1}^{(j)}\dot{\boldsymbol{v}}_{\bar{k}_j,1}^\top(\boldsymbol{y}_k+\Delta\boldsymbol{y}_k) + \sum_{i=2}^{p_{\bar{k}_j}} b_{\bar{k}_j,i}\dot{\boldsymbol{v}}_{\bar{k}_j,i}^{(j)}\dot{\boldsymbol{v}}_{\bar{k}_j,i}^\top(\boldsymbol{y}_k+\Delta\boldsymbol{y}_k) \\
&= \frac{1}{\beta+1}\dot{\boldsymbol{v}}_{\bar{k}_j,1}^{(j)}\dot{\boldsymbol{v}}_{\bar{k}_j,1}^\top(\boldsymbol{y}_k+\Delta\boldsymbol{y}_k) + \sum_{i=2}^{p_{\bar{k}_j}} b_{\bar{k}_j,i}\dot{\boldsymbol{v}}_{\bar{k}_j,i}^{(j)}\dot{\boldsymbol{v}}_{\bar{k}_j,i}^\top(\boldsymbol{y}_k+\Delta\boldsymbol{y}_k)
\end{aligned}
\tag{17}
$$

For the nonzero elements in $\dot{\boldsymbol{v}}_{\bar{k}_j,1}$, if $j \in C_k$, there are at least $n_{\min}(1-\alpha)$ elements being 1 at corresponding positions in $\boldsymbol{y}_k + \Delta\boldsymbol{y}_k$; if $j \notin C_k$, there are at most $n_{\max}\alpha c_{\max}$ elements being 1 at corresponding positions in $\boldsymbol{y}_k + \Delta\boldsymbol{y}_k$.Then the inner product in the first term in equation 17 can be bounded by

$$\dot{\boldsymbol{v}}_{\bar{k}_j,1}^{\top}(\boldsymbol{y}_k + \Delta\boldsymbol{y}_k) = \sum_{s=1}^{n_{\bar{k}_j}} \dot{\boldsymbol{v}}_{\bar{k}_j,1}^{(s)}(\boldsymbol{y}_k + \Delta\boldsymbol{y}_k)^{(s)} \begin{cases} \geq e_{\min,j} n_{\min}(1-\alpha), & j \in C_k \\ \leq e_{\max,j} \dot{\boldsymbol{v}}_{\bar{k}_j,1}^{(j)} n_{\max}\alpha c_{\max}, & j \notin C_k \end{cases}$$

For the inner product in the second term in equation 17, if $j \in C_k$, then $\dot{\boldsymbol{v}}_{\bar{k}_j,i}^{\top}(\boldsymbol{y}_k + \Delta\boldsymbol{y}_k)$ is the sum of at most $n_{\max}$ elements in $\dot{\boldsymbol{v}}_{\bar{k}_j,i}$. Since $\|\dot{\boldsymbol{v}}_{\bar{k}_j,i}\|_2^2 = 1$, the sum is bounded by $[-\sqrt{n_{\max}}, \sqrt{n_{\max}}]$. Similarly we can get the bound $[-\sqrt{n_{\max}c_{\max}}, \sqrt{n_{\max}c_{\max}}]$ for $j \notin C_k$. We also know that $|\dot{\boldsymbol{v}}_{\bar{k}_j,i}^{(j)}| \leq 1$. Now it remains to bound $\sum_{i=2}^{p_{\bar{k}_j}} b_{\bar{k}_j,i}$ by applying Cauchy–Schwarz inequality and Corollary A.6

$$\begin{aligned}
\sum_{i=2}^{p_{\bar{k}_j}} b_{\bar{k}_j,i} =& \sum_{i=2}^{p_{\bar{k}_j}} \frac{\lambda_{\bar{k},i}}{\beta + \lambda_{\bar{k},i}} \\
=& p_{\bar{k}_j} - 1 - \sum_{i=2}^{p_{\bar{k}_j}} \frac{\beta}{\beta + \lambda_{\bar{k},i}} \\
\leq& p_{\bar{k}_j} - 1 - \frac{\beta(p_{\bar{k}_j} - 1)^2}{\sum_{i=2}^{p_{\bar{k}_j}}(\lambda_{\bar{k}_j,i} + \beta)} \\
\leq& p_{\bar{k}_j} - 1 - \frac{\beta(p_{\bar{k}_j} - 1)^2}{2\sqrt{(p_{\bar{k}} - 1)}\sqrt{\delta'} + \mathcal{O}(\delta') + \beta(p_{\bar{k}_j} - 1)} \\
\leq& 2\sqrt{p-1}\frac{\sqrt{\delta'}}{\beta} + \mathcal{O}(\delta')
\end{aligned}$$

□

Lemma A.8. *$e_{\min,j}$, $e_{\max,j}$ and $e_{max,j}/e_{\min,j}$ can be bounded.*

$$\begin{aligned}
e_{\min,j} &\geq \sqrt{\frac{1}{1 + (n_{\max} - 1)(1+\delta')^2}} \\
e_{\max,j} &\leq \sqrt{\frac{1}{1 + (n_{\min} - 1)/(1+\delta')^2}} \\
\frac{e_{\max,j}}{e_{\min,j}} &\leq (1+\delta')\sqrt{\frac{(1 + (1+\delta')^2(n_{\max} - 1))}{n_{\max} - 1 + (1+\delta')^2}}
\end{aligned}$$

Proof. The proof is analogous to that for lemma A.3. □

Now we are ready to calculate the maximum noise level that allows correct prediction on all training examples, i.e., $\boldsymbol{z}_{k:j\in C_k}^{j} > \boldsymbol{z}_{k':j\notin C_{k'}}^{j}$. Let the preceding hold, then by lemma A.7 we get

$$\alpha < \frac{1 - \frac{2\sqrt{n_{\max}}(1+\sqrt{c_{\max}})\sqrt{p-1}}{\frac{\beta}{\beta+1} e_{\min,j} \dot{\boldsymbol{v}}_{\bar{k}_j,1}^{(j)} n_{\min}}\sqrt{\delta'}}{1 + \frac{n_{\max}e_{\max,j}}{n_{\min}e_{\min,j}}c_{\max}} - \mathcal{O}(\delta') \tag{18}$$

Plugging lemma A.8 into equation 18 with some algebraic manipulation yields

$$\alpha < \frac{1}{1 + \frac{n_{\max}}{n_{\min}}c_{\max}} - \mathcal{O}\left(\frac{\sqrt{\delta'}}{\beta}\right) = \frac{1}{1 + \frac{n_{\max}}{n_{\min}}c_{\max}} - \mathcal{O}\left(\frac{\sqrt{\delta}}{\beta}\right). \tag{19}$$

B. Considering Off-Diagonal Entries in the Adjacency Matrix (Connected Subclasses)

For here on we assume $n_{\bar{k}} = \frac{n}{\bar{K}}, \forall \bar{k} \in \bar{K}$ for simplicity, despite that our results can easily extend to unbalanced dataset.

Lemma B.1. *Under assumption 3.2, the off-diagonal entries in A is no longer zero. Let $\widetilde{\boldsymbol{A}}$ denote the new normalized matrix, which also contains non-zero off-diagonal entries. With a bit abuse of notation, in the following we use $\overline{\boldsymbol{A}}$ to denote the matrix obtained by normalizing A with off-diagonal elements ignored. Then all the properties of eigenvectors and eigenvalues of $\overline{\boldsymbol{A}}$ stated before including those lemma A.1, A.3, A.4 still hold. And $\widetilde{\boldsymbol{A}}$ can be written as a perturbation of $\overline{\boldsymbol{A}}$*

$$\widetilde{\boldsymbol{A}} = \overline{\boldsymbol{A}} + \boldsymbol{E},$$

with $\|\boldsymbol{E}\|_F = \mathcal{O}\left(\bar{K}^{5/2}\xi\right)$.

Proof. Let $\mathring{\boldsymbol{A}}$ be a matrix in the same shape of $\overline{\boldsymbol{A}}$ containing all elements of $\widetilde{\boldsymbol{A}}$ in the diagonal blocks. Let $\boldsymbol{H}$ be a matrix that collects the remaining off-diagonal elements. Therefore $\widetilde{\boldsymbol{A}} = \mathring{\boldsymbol{A}} + \boldsymbol{H}$, which can be rewritten as

$$\widetilde{\boldsymbol{A}} = \overline{\boldsymbol{A}} + \mathring{\boldsymbol{A}} - \overline{\boldsymbol{A}} + \boldsymbol{H}. \tag{20}$$

For all off-diagonal elements $h_{i,j}$ in $\boldsymbol{H}$

$$h_{i,j} \leq \frac{\bar{K}\xi}{n}.$$

Since there are $n^2 - \sum_{\bar{k}=1}^{\bar{K}} n_{\bar{k}}^2$ entries outside of the diagonal blocks, the norm of $\boldsymbol{H}$ can be bounded by

$$\|\boldsymbol{H}\|_F \leq \sqrt{1 - \frac{1}{\bar{K}}}\bar{K}\xi \tag{21}$$

Each element in the diagonal blocks of $\overline{\boldsymbol{A}} - \mathring{\boldsymbol{A}}$ is non-negative. Also, supposing $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are from subclass $\bar{k}$, we have

$$\begin{aligned}
(\overline{\boldsymbol{A}} - \mathring{\boldsymbol{A}})_{i,j} =& \frac{w_{\boldsymbol{x}_i\boldsymbol{x}_j}}{\sqrt{\sum_{s:\boldsymbol{x}_s \in C_{\bar{k}}} w_{\boldsymbol{x}_i\boldsymbol{x}_s}}\sqrt{\sum_{t:\boldsymbol{x}_t \in C_{\bar{k}}} w_{\boldsymbol{x}_t\boldsymbol{x}_j}}} - \frac{w_{\boldsymbol{x}_i\boldsymbol{x}_j}}{\sqrt{\sum_{s=1}^{n} w_{\boldsymbol{x}_i\boldsymbol{x}_s}}\sqrt{\sum_{t=1}^{n} w_{\boldsymbol{x}_t\boldsymbol{x}_j}}} \\
\leq& \frac{n(\bar{K}-1)\xi}{\frac{n}{\bar{K}(1+\delta)}\left(\frac{n}{\bar{K}(1+\delta)} + n(\bar{K}-1)\xi\right)} \\
=& \mathcal{O}\left(\frac{\bar{K}^3(1+\delta)^2\xi}{n}\right),
\end{aligned}$$

by which the norm of $\overline{\boldsymbol{A}} - \mathring{\boldsymbol{A}}$ is bounded

$$\|\overline{\boldsymbol{A}} - \mathring{\boldsymbol{A}}\|_F = \mathcal{O}\left(\bar{K}^{5/2}(1+\delta)^2\xi\right) = \mathcal{O}\left(\bar{K}^{5/2}\xi\right). \tag{22}$$

Combining equations 20, 21 and 22 completes the proof. □

$\overline{\boldsymbol{A}}$ has the following eigendecomposition

$$\overline{\boldsymbol{A}} = \left[\boldsymbol{V}_I\boldsymbol{V}_N\right] \begin{bmatrix} \boldsymbol{\Sigma}_I & 0 \\ 0 & \boldsymbol{\Sigma}_N \end{bmatrix} \begin{bmatrix} \boldsymbol{V}_I^\top \\ \boldsymbol{V}_N^\top \end{bmatrix},$$

where $\boldsymbol{\Sigma}_I$ collects the $\bar{K}$ largest eigenvalues $\lambda_1, \dots, \lambda_{\bar{K}}$ on the diagonal and $\boldsymbol{\Sigma}_N$ collects the remaining $\lambda_{\bar{K}+1}, \dots, \lambda_{\bar{n}}$. $\boldsymbol{V}_I$ and $\boldsymbol{V}_N$ collects the corresponding $\bar{K}$ and $n - \bar{K}$ eigenvectors, respectively. Let $\widetilde{\boldsymbol{A}}$ has analogous decomposition

$$\widetilde{\boldsymbol{A}} = \left[\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_N\right] \begin{bmatrix} \widetilde{\boldsymbol{\Sigma}}_I & 0 \\ 0 & \widetilde{\boldsymbol{\Sigma}}_N \end{bmatrix} \begin{bmatrix} \widetilde{\boldsymbol{V}}_I^\top \\ \widetilde{\boldsymbol{V}}_N^\top \end{bmatrix},$$

with eigenvalues $\widetilde{\lambda}_1, \dots, \widetilde{\lambda}_n$ and eigenvectors $\widetilde{\boldsymbol{v}}_1, \dots, \widetilde{\boldsymbol{v}}_n$. Eigenvalues of both matrices are indexed in descending order.

B.1. Perturbation in Eigenvalues and Eigenvectors

The following two lemmas bound the changes in eigenvalues, eigenvectos and the alignment between labels and eigenvectors caused by ξ.

Lemma B.2. *We have the following bound for eigenvalues of* $\widetilde{\boldsymbol{A}}$*:*

$$\begin{cases} 1-\mathcal{O}(K^{5/2}\xi) \leq \widetilde{\lambda}_i \leq 1, & i=1,2,\dots,\bar{K} \\ \sum_{i=\bar{K}+1}^{p} \lambda_i \leq \mathcal{O}(\sqrt{\delta}+\bar{K}^{5/2}\xi) & \end{cases}$$

Proof. From Lemma A.2 and Corollary A.6 we know that

$$\begin{cases} \lambda_i = 1, & i=1,2,\dots,\bar{K}, \\ \sum_{i=\bar{K}+1}^{p} \lambda_i \leq \mathcal{O}(\sqrt{\delta}). & \end{cases} \tag{23}$$

By Weyl's inequality on perturbation, we have

$$|\widetilde{\lambda}_i - \lambda_i| \leq \|\boldsymbol{E}\|_2.$$

The right-hand-side is $\leq \|\boldsymbol{E}\|_F$ and therefore $\mathcal{O}(\bar{K}^{5/2}\xi)$ by lemma B.1. Combining the preceding with equation 23 completes the proof.

□

Lemma B.3. *The norm of the projection of* $\boldsymbol{Y}$ *onto the range of* $\widetilde{\boldsymbol{V}}_I$ *is bounded from below, i.e.,*

$$\|\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_I^T\boldsymbol{Y}\|_F^2 \geq \|\boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2 - \mathcal{O}(n\bar{K}^2\xi).$$

Proof. By Lemma A.2 and Corollary A.5 we have

$$\lambda_{\bar{K}} - \lambda_{\bar{K}+1} \geq 1-\mathcal{O}(\sqrt{\delta}).$$

By Wedin's Theorem (Wedin, 1972; Stewart, 1990), we have the following bound on the principle angle between the range of $\widetilde{\boldsymbol{V}}_I$ and the range of $\boldsymbol{V}_I$ as long as $1 \geq \mathcal{O}(\sqrt{\delta}+\bar{K}^{5/2}\xi)$

$$\|\boldsymbol{V}_I\boldsymbol{V}_I^T(\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_I^T - I)\|_F \leq \mathcal{O}\left(\frac{\|\boldsymbol{E}\|_F}{\lambda_{\bar{K}}-\lambda_{\bar{K}+1}}\right) \leq \mathcal{O}\left(\frac{\mathcal{O}(\bar{K}^{5/2}\xi)}{1-\mathcal{O}(\sqrt{\delta})}\right) = \mathcal{O}(\bar{K}^{5/2}\xi).$$

Thus

$$\begin{aligned}
\|\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_I^T\boldsymbol{Y}\|_F^2 =& \|\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_I^T\boldsymbol{Y} - \boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y} + \boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2 \\
\geq& \|\boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2 + 2\langle\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_I^T\boldsymbol{Y} - \boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y},\ \boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\rangle_F \\
\geq& \|\boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2 + 2\,\mathrm{Tr}((\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_I^T - \boldsymbol{V}_I\boldsymbol{V}_I^T)\boldsymbol{Y}\boldsymbol{Y}^T\boldsymbol{V}_I\boldsymbol{V}_I^T) \\
=& \|\boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2 + 2\,\mathrm{Tr}(\boldsymbol{V}_I\boldsymbol{V}_I^T(\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_I^T - I)\boldsymbol{Y}\boldsymbol{Y}^T) \\
=& \|\boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2 + 2\langle\boldsymbol{V}_I\boldsymbol{V}_I^T(\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_I^T - \boldsymbol{I}),\ \boldsymbol{Y}\boldsymbol{Y}^T\rangle_F \\
\geq& \|\boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2 - 2\|\boldsymbol{V}_I\boldsymbol{V}_I^T(\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_I^T - \boldsymbol{I})\|_F\|\boldsymbol{Y}\boldsymbol{Y}^T\|_F \\
\geq& \|\boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2 - \mathcal{O}(\bar{K}^{5/2}\xi)\frac{n}{\sqrt{K}} \\
=& \|\boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2 - \mathcal{O}(n\bar{K}^2\xi).
\end{aligned}$$

□

B.2. Error under Gaussian Noise (Proof for Theorem 3.4)

Considering ξ, rewrite **bias**^{2} as

$$
\begin{aligned}
\textbf{bias}^2 =& 1 + \frac{1}{n}\sum_{i=1}^{p}\sum_{j=1}^{K}(\widetilde{b}_i^2 - 2\widetilde{b}_i)(\widetilde{\boldsymbol{v}}_i{}^\top \boldsymbol{y}_j)^2 \\
\leq& 1 - \frac{1}{n}\sum_{i=1}^{\bar{K}}\sum_{j=1}^{K}(2\widetilde{b}_i - \widetilde{b}_i^2)(\widetilde{\boldsymbol{v}}_i{}^\top \boldsymbol{y}_j)^2 \\
\leq& 1 - \frac{1}{n}(2\widetilde{b}_{\bar{K}} - \widetilde{b}_{\bar{K}}^2)\sum_{i=1}^{\bar{K}}\sum_{j=1}^{K}(\widetilde{\boldsymbol{v}}_i{}^\top \boldsymbol{y}_j)^2 \\
\leq& 1 - \frac{1}{n}(2\widetilde{b}_{\bar{K}} - \widetilde{b}_{\bar{K}}^2)\|\widetilde{\boldsymbol{V}}_I\widetilde{\boldsymbol{V}}_I^T\boldsymbol{Y}\|_F^2, \qquad (24)
\end{aligned}
$$

where $\widetilde{b}_i = \frac{\widetilde{\lambda}_i}{\widetilde{\lambda}_i+\beta}$. Also, Lemma A.3 gives us the lower bound for $\|\boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2$

$$
\begin{aligned}
\|\boldsymbol{V}_I\boldsymbol{V}_I^T\boldsymbol{Y}\|_F^2 =& \sum_{\bar{k}}^{\bar{K}}\|\boldsymbol{v}_{\bar{k},1}\|_1^2 \\
\geq& n - \mathcal{O}(\delta). \qquad (25)
\end{aligned}
$$

Combining lemma B.2, lemma B.3, equation 24 and equation 25 yields the bound for the bias

$$
\textbf{bias}^2 = (\frac{\beta}{1+\beta})^2 + \mathcal{O}(\delta + \xi).
$$

We bound the variance in the same manner as in Section A.3 by applying Cauchy–Schwarz inequality, Corollary A.6 and Lemma B.2

$$
\begin{aligned}
\textbf{variance} =& \frac{\sigma^2}{n}\sum_{i=1}^{p}\widetilde{b}_i^2 \\
=& \frac{\sigma^2}{n}\sum_{i=1}^{\bar{K}}\widetilde{b}_i^2 + \frac{\sigma^2}{n}\sum_{i=\bar{K}+1}^{p}\widetilde{b}_i^2 \\
\leq& \sigma^2\frac{\bar{K}}{n}(\frac{1}{\beta+1})^2 + \frac{\sigma^2}{n}\sum_{i=\bar{K}+1}^{p}(1 - \frac{\beta}{\widetilde{\lambda}_i+\beta}) \\
=& \sigma^2\frac{\bar{K}}{n}(\frac{1}{\beta+1})^2 + \frac{\sigma^2}{n}(p-\bar{K}) - \frac{\sigma^2}{n}\sum_{i=\bar{K}+1}^{p}\frac{\beta}{\widetilde{\lambda}_i+\beta} \\
\leq& \sigma^2\frac{\bar{K}}{n}(\frac{1}{\beta+1})^2 + \frac{\sigma^2}{n}(p-\bar{K}) - \frac{\sigma^2}{n}\frac{\beta(p-\bar{K})^2}{\sum_{i=\bar{K}+1}^{p}(\widetilde{\lambda}_i+\beta)} \\
\leq& \sigma^2\frac{\bar{K}}{n}(\frac{1}{\beta+1})^2 + \sigma^2\mathcal{O}(\frac{\sqrt{\delta}+\xi}{\beta}).
\end{aligned}
$$

C. Contrastive Learning Slightly Improves the Alignment Between Jacobian Matrix and Ground-truth Labels

We compare the alignments between the clean label vector and the initial Jacobian matrix of (1) network pretrained using SimCLR for 1000 epochs, (2) network pretrained using SimCLR for 100 epochs and (3) randomly initialized network. $\boldsymbol{y} \in \mathbb{R}^{nK}$ is the vector obtained by flattening the label matrix $\boldsymbol{Y}$, i.e., concatenating the n rows of $\boldsymbol{Y}$. Let $\boldsymbol{z}(\boldsymbol{x}_i, \boldsymbol{W}) \in \boldsymbol{R}^K$

	$\|\Pi_{\mathcal{I}}(\boldsymbol{y})\|_F$	$\|\Pi_{\mathcal{N}}(\boldsymbol{y})\|_F$	$\|\boldsymbol{J}\boldsymbol{J}^T\boldsymbol{y}\|_F/\|\boldsymbol{J}\boldsymbol{J}^T\|_F$
Pretrained for 1000 epochs	10.063	29.979	3.184
Pretrained for 100 epochs	10.036	29.988	3.175
Randomly initialized	10.014	29.995	3.055

Table 3. Alignment between the Jacobian matrix and the clean labels.

be the output of the network given example $\boldsymbol{x}_i$ and parameters $\boldsymbol{W} \in \mathbb{R}^d$ (we see the parameters of the network as a vector). Then the Jacobian $\boldsymbol{J}$ is defined as

$$\boldsymbol{J}(\boldsymbol{W}) = \left[\frac{\partial \boldsymbol{z}(\boldsymbol{x}_1, \boldsymbol{W})}{\boldsymbol{W}} \dots \frac{\partial \boldsymbol{z}(\boldsymbol{x}_n, \boldsymbol{W})}{\boldsymbol{W}}\right]^\top .$$

Note that $\frac{\partial \boldsymbol{z}(\boldsymbol{x}_i, \boldsymbol{W})}{\boldsymbol{W}} \in \mathbb{R}^{d \times K}$, therefore $\boldsymbol{J}(\boldsymbol{W}) \in \mathbb{R}^{nK \times d}$. In table 3 $\Pi_I(\boldsymbol{y})$ is the projection of $\boldsymbol{y}$ onto the span of the 10 singular vectors of $\boldsymbol{J}(\boldsymbol{W}_0)$ with larges singular values and $\Pi_N(\boldsymbol{y})$ is the projection of $\boldsymbol{y}$ onto the span of the remaining singular vectors. Interestingly, pretraining for more epochs leads to larger $\Pi_I(\boldsymbol{y})$ and smaller $\Pi_N(\boldsymbol{y})$ and therefore larger $\|\boldsymbol{J}\boldsymbol{J}^T\boldsymbol{y}\|_F/\|\boldsymbol{J}\boldsymbol{J}^T\|_F$. How much this slight improvement in the alignment contributes to the robustness deserves further investigation.

D. Baselines for Experiments in Section 3.3

We compare our results with many commonly used baselines for robust training against label noise: (1) F-correction (Patrini et al., 2017) is a two step process, where a neural network is first trained on noisily-labelled data, then retrained using a corrected loss function based on an estimation of the noise transition matrix. (2) Decoupling (Malach & Shalev-Shwartz, 2017) is a meta-algorithm that trains two networks concurrently, only training on examples where the two networks disagree. (3) Co-teaching (Han et al., 2018) also trains two networks simultaneously. Each network selects subsets of clean data with high probability for the other network to train on. (4) MentorNet (Jiang et al., 2018) uses two neural networks, a student and a mentor. The mentor dynamically creates a curriculum based on the student, while the student trains on the curriculum provided by the mentor. (5) D2L (Ma et al., 2018) learns the training data distribution, then dynamically adapts the loss function based on the changes in dimensionality of subspaces during training. (6) INCV (Chen et al., 2019) identifies random subsets of the training data with fewer noisy labels, then applies Co-teaching to iteratively train on subsets found with the most clean labels. (7) T-Revision (Xia et al., 2019) learns the transition matrix efficiently using an algorithm that does not rely on known points with clean labels. (8) L_DMI (Xu et al., 2019) uses a novel information-theoretic loss function based on determinant based mutual information. (9) ELR (Liu et al., 2020) uses semi-supervised learning techniques to regularize based on the early-learning phase of training, to ensure the noisy labels are not overfit. (10) CRUST (Mirzasoleiman et al., 2020) dynamically selects subsets of clean data points by clustering in the gradient space. (11) Mixup (Zhang et al., 2017) smooths the decision boundary by adding linear interpolations of feature vectors and their labels to the dataset.

E. Training Only the Linear Layer v.s. Training All Layers

Figure 3 compares the performance of fine-tuning only the last linear layer (i.e., with the encoder frozen) and fine-tuning all layers (i.e., with the encoder unfrozen). For both CIFAR-10 and CIFAR-100 we first pretrain Res-Net 32 using SimCLR for 1000 epochs using the Adam optimizer with a learning rate of 3×10^{-4}, a weight decay of 1×10^{-6} and a batch size of 128. Then, for both frozen and unfrozen fine-tunings, we use the SGD optimizer with a learning rate of 5×10^{-3}, a weight decay of 1×10^{-3}, a batch size of 64. Interestingly, in most cases training all layers achieves a higher test accuracy, which, however, does not mean it is necessarily better under some other hyperparameter settings. Also, we note that training all layers is more likely to overfit, especially under large noise level (column 3 in figure 3).

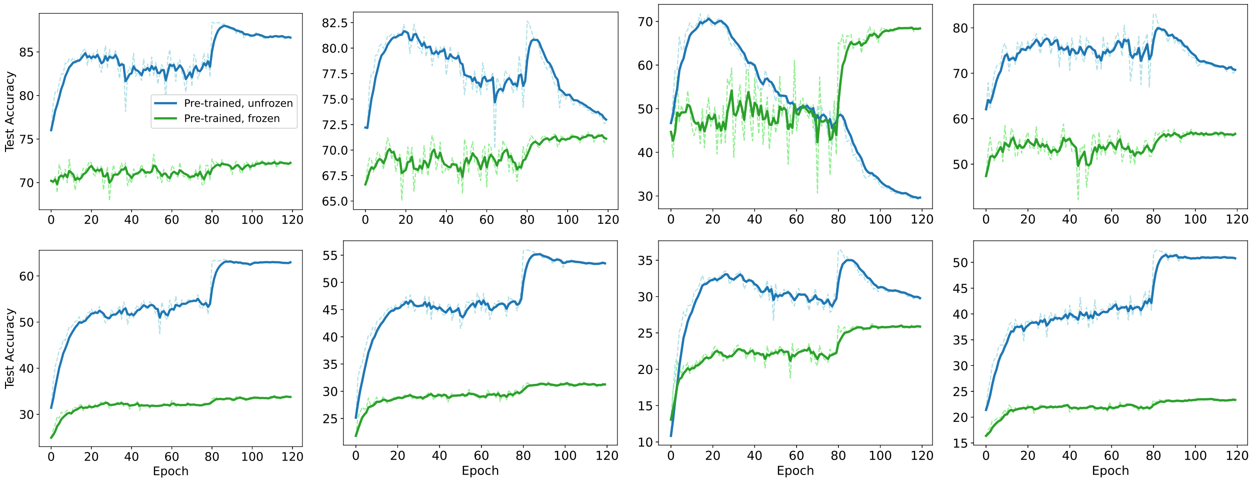


Figure 3. Test accuracy of fine-tuning a pre-trained network with frozen encoder v.s. unfrozen encoder on CIFAR-10 (top) and CIFAR-100 (bottom) under 20%, 50%, 80% symmetric noise and 40% asymmetric noise (left to right).