FINE Samples for Learning with Noisy Labels

Anonymous Author(s)
Affiliation
Address
email

Abstract

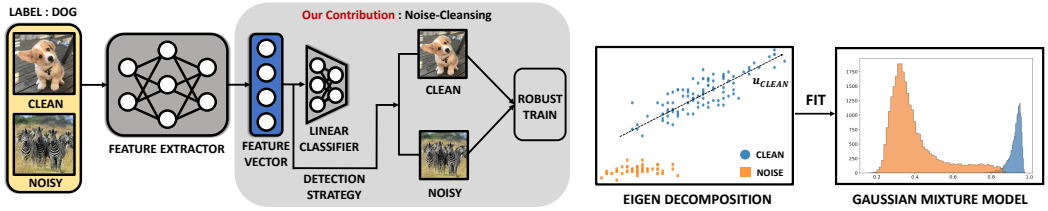
Modern deep neural networks (DNNs) become frail when the datasets contain noisy (incorrect) class labels. Robust techniques in the presence of noisy labels can be categorized into two folds: developing *noise-robust* functions or using *noise-cleansing* methods by detecting the noisy data. Recently, *noise-cleansing* methods have been considered as the most competitive noisy-label learning algorithms. Despite their success, their noisy label detectors are often based on heuristics more than a theory, requiring a robust classifier to predict the noisy data with loss values. In this paper, we propose a novel detector for filtering label noise. Unlike most existing methods, we focus on each data's latent representation dynamics and measure the alignment between the latent distribution and each representation using the eigen decomposition of the data covariance matrix. Our framework, coined as *filtering noisy instances via their eigenvectors* (FINE), provides a robust detector with derivative-free simple methods having theoretical guarantees. Under our framework, we propose three applications of the FINE: sample-selection approach, semi-supervised learning approach, and collaboration with *noise-robust* loss functions. Experimental results show that the proposed methods consistently outperform corresponding baselines for all three applications on various benchmark datasets.

1 Introduction

Deep neural networks have achieved remarkable success in numerous tasks as the amount of accessible data dramatically increases [1, 2]. On the other side, accumulated datasets are typically labeled through a labor-intensive job by a human or through web crawling [3] so that they may be easily corrupted (*label noise*) in real-world situations. Recent studies have shown that deep neural networks have the capacity to memorize any labeling of the data essentially [4]. Even a small amount of such noisy data can hinder the generalization of deep neural networks owing to their strong memorization of noisy labels [4, 5]. Hence, it becomes crucial to train deep neural networks that are robust to corrupted labels. As label noise problems may appear everywhere, such robustness contributes more reliability in many applications such as the e-commerce market [6], medical fields [7], on-device AI [8], and autonomous driving systems [9].

To improve the robustness against noisy data, the methods for learning with noisy labels (LNL) have been evolving mainly in two directions [10]: (1) designing noise-robust objective functions or regularizations and (2) detecting and cleansing the noisy data. In general, the former *noise-robust* direction uses explicit regularization techniques [11, 12, 13] or robust loss functions [14, 15, 16, 17], but they become far from the state-of-the-art performance [4, 18] on the dataset with a severe noise rate. Recently, researchers have designed *noise-cleansing* algorithms focused on segregating the clean data (i.e., samples with uncorrupted labels) from the corrupted data [19, 20, 21, 10, 22, 23]. One of the popular criteria for the segregation process is the loss value between the prediction of the noisy classifier and its noisy label, where it is generally assumed that the noisy data have a large loss [19, 20, 21, 10] or the magnitude of the gradient during training [17, 16]. However, these methods may still be biased by the corrupted linear classifier towards label noise because their criterion (e.g.,

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



(a) Noise-Cleansing-based Approach (b) FINE

Figure 1: Illustration for (a) the basic concept of this work and (b) the proposed detection framework, FINE. Noise-cleansing learning generally separates clean data from the original dataset by using the prediction outputs. We propose a novel derivative-free detector based on an unsupervised clustering algorithm on the high-order topological space. FINE measures the alignment of pre-logits (i.e., penultimate layer representation vectors) towards the class-representative vector that is extracted through the eigen decomposition of the covariance matrix of data representations.

loss values or weight gradient) uses the posterior information of such a linear classifier [24]. Maennel et al. [25] analytically showed that the principal components of the weights of a neural network align with the randomly labeled data; this phenomenon can yield more negative effects on the classifier as the number of randomly labeled classes increases. Recently, Wu et al. [23] used an inherent geometric structure induced by nearest neighbors (NN) in latent space and filtered out isolated data in such topology, and its quality is sensitive to its hyperparameters regarding NN clustering in the presence of severe noise rates.

In order to mitigate such issues for label noise detectors, we provide a novel yet simple detector framework, *filtering noisy labels via their eigenvectors* (FINE) with theoretical guarantees to provide a high-quality splitting of clean and corrupted examples (without the need to estimate noise rates). Instead of using the neural network's linear classifier, FINE utilizes the principal components of latent representations made by eigen decomposition which is one of the most widely used unsupervised learning algorithms and separates clean data and noisy data by these components (Figure 1a). To motivate our approach, as Figure 1b shows, we find that the clean data (blue points) are mainly aligned on the principal component (black dotted line), whereas the noisy data (orange points) are not; thus, the dataset is well clustered with the alignment of representations towards the principal component by fitting them into Gaussian mixture models. We apply our framework to various *LNL* methods: sample selection approach, a semi-supervised learning approach, and conjunction with noise-robust loss functions. The key contributions of this work are summarized as follows:

- We propose a novel framework, termed FINE (*filtering noisy labels via their eigenvectors*), for detecting clean instances from noisy datasets. FINE makes robust decision boundary for the high-order topological information of data in latent space by using eigen decomposition of their covariance matrix.
- We provide a provable evidence that FINE allows meaningful decision boundary made by eigenvectors in latent space. We support our theoretical analysis with various experimental results regarding the characteristics of the principal components extracted by our FINE detector.
- We develop a simple sample-selection method by replacing the existing detector method with FINE. We empirically validate that a sample-selection learning with FINE provides consistently more superior detection quality and higher test accuracy than other existing alternative methods such as Co-teaching familie [20, 21], TopoFilter [23], and CRUST [22].
- We experimentally show that our detection framework can be applied into various ways to existing *LNL* methods and validate that ours consistently improves the generalization in the presence of noisy data: sample selection approach [20, 21], semi-supervised learning approach [26], and collaboration with noise-robust loss functions [17, 16, 5].

Organization. The remainder of this paper is organized as follows. In Section 2, we discuss the recent literature for the solutions of learning with noisy labels and meaningful detector. In Section 3, we address our motivation for noisy label detector with theoretical insights and provide our main method, filtering the noisy labels via their eigenvectors (FINE). In Section 4, we present the experimental results. Finally, Section 5 concludes the paper.

2 Related Works

Zhang et al. [4] empirically showed that any convolutional neural networks trained using stochastic gradient methods easily fit a random labeling of the training data. To tackle this issue, numerous works have examined the classification task with noisy labels. We do not consider the works that assumed the availability of small subsets of training data with clean labels [27, 28, 29, 30, 31].

Noise-Cleansing-based Approaches. Noise-cleansing methods have evolved following the improvement of noisy detectors. Han et al. [20] suggested a noisy detection approach, named co-teaching, that utilizes two networks, extracts subsets of instances with small losses from each network, and trains each network with subsets of instances filtered by another network. Yu et al. [21] combined a disagreement training procedure with co-teaching, which only selects instances predicted differently by two networks. Huang et al. [10] provided a simple noise-cleansing framework, training-filtering-training; the empirical efficacy was improved by first finding label errors, then training the model only on data predicted as clean. Recently, new noisy detectors with theoretical support have been developed. Wu et al. [23] proposed a method called TopoFilter that filters noisy data by utilizing the k-nearest neighborhood algorithm and Euclidean distance between pre-logits. Mirzasoleiman et al. [22] introduced an algorithm that selects subsets of clean instances that provide an approximately low-rank Jacobian matrix and proved that gradient descent applied to the subsets prevents overfitting to noisy labels. Pleiss et al. [32] proposed an area under margin (AUM) statistic that measures the average difference between the logit values of the assigned class and its highest non-assigned class to divide clean and noisy samples. Cheng et. al [33] progressively filtered out corrupted instances using a novel confidence regularization term. The noise-cleansing method was also developed in a semi-supervised learning (SSL) manner. Li et al. [26] modeled the per-sample loss distribution and divide it into a labeled set with clean samples and an unlabeled set with noisy samples, and they leverage the noisy samples through the well-known SSL technique MixMatch [34].

Noise-Robust Models. Noise-robust models have been studied in the following directions: robust-loss functions, regularizations, and strategies. First, for robust-loss functions, Ghosh et al. [15] showed that the mean absolute error (MAE) might be robust against noisy labels. Zhang & Sabuncu et al. [17] argued that MAE performed poorly with DNNs and proposed a GCE loss function, which can be seen as a generalization of MAE and CE. Wang et al. [16] introduced the reverse version of the cross-entropy term (RCE) and suggested that the SCE loss function is a weighted sum of the CE and RCE. Some studies have stated that the early-stopped model can prevent the memorization phenomenon for noisy labels [35, 4] and theoretically analyzed it [18]. Base on this intuition, Liu et al. [5] proposed an ELR loss function to prohibit memorizing noisy data by leveraging the semi-supervised learning techniques. Xia et al. [36] clarified which neural network parameters cause memorization and proposed a robust training strategy for these parameters. Efforts have been made to develop regularizations on the prediction level by smoothing the one-hot vector [37], using linear interpolation between data instances [13], and distilling the rescaled prediction of other models [38]. However, these works have limitations in terms of performance as the noise rate of the dataset increases.

Dataset Resampling. Label-noise detection may be a category of data resampling which is a common technique that extracts a "*helpful*" dataset from the distribution of the original dataset to remove the dataset bias in the machine learning community. In class-imbalance tasks, numerous studies have conducted over-sampling of minority classes [39, 40] or undersampling the majority classes [41] to balance the amount of data per class. Li & Vasconcelos et al. [42] proposed a resampling procedure to reduce the representation bias of the data by learning a weight distribution that favors difficult instances for a given feature representation. Le Bras et al. [43] suggested an adversarial filtering-based approach to remove spurious artifacts in a dataset. Analogously, in anomaly detection and out-of-distribution detection problems [44, 45, 46], the malicious data are usually detected by examining the loss value or negative behavior in the feature representation space. While our research is motivated by such previous works, this paper focuses on the noisy image classification task.

3 Method

In this section, we present our detector framework and the theoretical motivation behind the detector in the high-dimensional classification. To segregate the clean data, we utilize the degree of alignment between the representations and the eigenvector of the representations' covariance matrices for all classes, called **FINE** (*FIltering Noisy instnaces via their Eigenvectors*). Our algorithm is as

Algorithm 1: FINE Algorithm for Sample Selection

INPUT : Noisy training data $\mathcal{D}$, feature extractor g, number of classes K, clean probability threshold ζ, set of FINE scores for class k $\mathcal{F}_k$

OUTPUT : Collected clean data $\mathcal{C}$

Initialize $\mathcal{C} \leftarrow \emptyset$, $\hat{\mathcal{D}} \leftarrow \mathcal{D}$, $\mathbf{\Sigma}_k \leftarrow \mathbf{0}$ for all $k = 1, \ldots, K$
/* Update the convariance matrices for all classes */
for $(\boldsymbol{x}_i, y_i) \in \mathcal{D}$ **do**
  $\boldsymbol{z}_i \leftarrow g(\boldsymbol{x}_i)$
  Update the covariance matrix $\mathbf{\Sigma}_{y_i} \leftarrow \mathbf{\Sigma}_{y_i} + \boldsymbol{z}_i \boldsymbol{z}_i^\top$
end for
/* Generate the principal component with eigen decomposition */
for $k = 1, \ldots, K$ **do**
  $\mathbf{U}_k, \mathbf{\Lambda}_k \leftarrow$ EIGEN DECOMPOSITION OF $\mathbf{\Sigma}_k$
  $\mathbf{u}_k \leftarrow$ THE FIRST COLUMN OF $\mathbf{U}_k$
end for
/* Compute the alignment score and get clean subset $\mathcal{C}$ */
for $(\boldsymbol{x}_i, y_i) \in \mathcal{D}$ **do**
  Compute the FINE score $f_i = \langle \mathbf{u}_{y_i}, \boldsymbol{z}_i \rangle^2$ and $\mathcal{F}_{y_i} \leftarrow \mathcal{F}_{y_i} \cup \{f_i\}$
end for
/* Finding the samples whose clean probability is larger than ζ */
$\mathcal{C} \leftarrow \mathcal{C} \cup$ GMM $(\mathcal{F}_k, \zeta)$ for all $k = 1, \ldots, K$

follows (Algorithm 1). FINE first creates a covariance matrix of the representation in noisy training dataset for each class and conducts the eigen decomposition for those covariance matrices. Then, FINE finds clean and noisy instances using the square of inner product values between the representations and the first eigenvector having the largest eigenvalue. In this manner, we treat the data as clean if aligned onto the first eigenvector while the most of noisy instances are not. Here, we formally define '*alignment*' and '*alignment clusterability*' in Definition 1 and Definition 2.

Definition 1. *(Alignment) Let $\boldsymbol{x}$ be the data labeled as class k and $\boldsymbol{z}$ is the output of feature extractor with input $\boldsymbol{x}$, and the covariance matrix $\mathbf{\Sigma}_k$ of all features labeled as class k in dataset $\mathcal{D}$ (i.e., $\mathbf{\Sigma}_k = \sum_{\boldsymbol{z} \in \{class=k\}} \boldsymbol{z}\boldsymbol{z}^\top$). Then, we estimate the alignment of the data $\boldsymbol{x}$ via $\langle \boldsymbol{u}_1, \boldsymbol{z} \rangle^2$ where $\mathbf{u}_1$ is the first column of $\mathbf{U_k}$ from the eigen decomposition of $\mathbf{\Sigma_k}$, i.e., $\mathbf{\Sigma_k} = \mathbf{U_k \Lambda_k U_k}^\top$ when the eigenvalues are in descending order.*

Definition 2. *(Alignment Clusterability) For all features labeled as class k in dataset $\mathcal{D}$, let fit a Gaussian Mixture Model (GMM) on their alignment (Definition 1) distribution to divide current samples into a clean set and a noisy set; the set having larger mean value is treated as a clean set, and another one is a noisy set. Then, we call a dataset $\mathcal{D}$ satisfies alignment clusterability if the representation $\boldsymbol{z}$ labeled as the same true class belongs to the clean set.*

As an empirical evaluation, the quality of our detector for noisy data is presented along the ***F-score***, a widely used criterion in noisy label detection, anomaly detection and out-of-distribution detection [33, 44, 45, 46]. We treat the selected clean samples as positive class and noisy samples as negative class. ***F-score*** is the harmonic mean of the precision and the recall; the *precision* indicates the fraction of clean samples among all samples that are predicted as clean, and the *recall* indicates the portion of clean samples that are identified correctly.

3.1 Alignment Analysis for a Noisy Label Detector

To design a robust label noise filtering framework, we explore a linear nature of the topological space of feature vectors for data resampling techniques and deal with the classifier contamination due to random labels. Recent studies on the distribution of latent representations in deep neural networks provide intuition regarding how correctly the outlier samples can be filtered with the hidden space's geometrical information. For instance, in [46, 24], the authors proposed frameworks for novelty detection using the topological information of pre-logit based on the Mahalanobis distance, and, in [23], the authors filtered the noisy data based on the Euclidean distance between pre-logits. Maennel et al. [25] analytically showed that an alignment between the principal components of

network parameters and those of data takes place when training with random labels. This finding points out that random labels can corrupt a classifier, and thus building a robust classifier is required.

Motivated by these works, we aim to design a novel detector using the principal components of latent features to satisfy Definition 2. However, it is intractable to find the optimal classifier to maximize the separation of *alignment clusterability* because clean data distribution and noisy data distribution are inaccessible. To handle this issue, we attempt to approximate the clean eigenvector to maximize the alignment values of clean data rather than to maximize the separation; the algorithm utilizes the eigenvector of the data for each class (Figure 2). Below, we provide the upper bound for the perturbation towards the clean data's eigenvector under simple problem settings with noisy labels referred to in other studies [5, 23]. We first introduce notations. Next, we establish the theoretical evidence that our FINE algorithm approximates the clean data's eigenvectors under some assumptions for its analytic tractability (Theorem 1). We mainly present the theorem and its interpretation; details of the proofs can be found in Appendix.

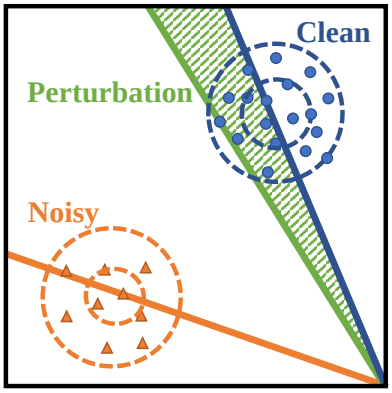


Figure 2: Illustration for the problem settings and Theorem 1. The perturbation (green shade) is the angle between the first eigenvector of clean instances (blue line) and the estimated first eigenvector (green line) which is perturbed by that of noisy instances (orange line). Note that blue and orange points are clean instances and noisy instances, respectively.

Notations. Consider a binary classification task. Assume that the data points and labels lie in $\mathcal{X} \times \mathcal{Y}$, where the feature space $\mathcal{X} \subset \mathbb{R}^d$ and label space $\mathcal{Y} = \{-1, +1\}$. A single data point $\boldsymbol{x}$ and its true label y follow a distribution $(\boldsymbol{x}, y) \sim P_{\mathcal{X} \times \mathcal{Y}}$. Denote by $\tilde{y}$ the observed label (potentially corrupted). Without loss of generality, we focus on the set of data points whose observed label is $\tilde{y} = +1$.

Let $\mathbf{X} \subset \mathcal{X}$ be the finite set of features with clean instances whose true label is $y = +1$. Similarly, let $\tilde{\mathbf{X}} \subset \mathcal{X}$ be the set of noisy instances whose true label is $y = -1$. To establish our theorem, we assume the following reasonable conditions referred to other works using linear discriminant analysis (LDA) assumptions [24, 47]:

Assumption 1. *The feature distribution is comprised of two Gaussians, each identified as a clean cluster and a noisy cluster.*

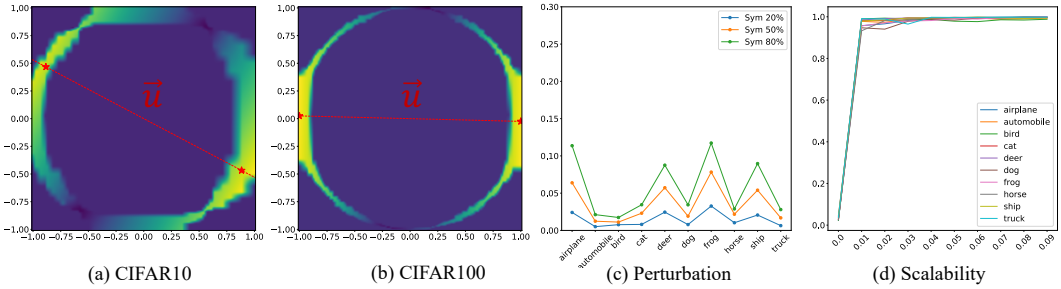
Assumption 2. *The features of all instances with $y = +1$ are aligned on the unit vector $\mathbf{v}$ with the white noise, i.e., $\mathbb{E}_{\boldsymbol{x} \in \mathbf{X}}[\boldsymbol{x}] = \boldsymbol{v}$. Similarly, features of all instances with $y = -1$ are aligned on the unit vector $\boldsymbol{w}$, i.e., $\mathbb{E}_{\boldsymbol{x} \in \tilde{\mathbf{X}}}[\boldsymbol{x}] = \boldsymbol{w}$.*

Theorem 1. ***(Upper bound for the perturbation towards the clean data's eigenvector $\boldsymbol{v}$)*** *Let N_+ and N_- be the number of clean instances and noisy instances, respectively, and $\boldsymbol{u}$ be the FINE's eigenvector which is the first column of $\mathbf{U}$ from the eigen decomposition of the whole data's covariance matrix $\boldsymbol{\Sigma}$. For any $\delta \in (0, 1)$, its perturbation towards the $\boldsymbol{v}$ in assumption 2 (i.e., 2-norm for difference of projection matrices; left hand side of Eq. (1)) holds the following with probability $1 - \delta$:*

$$\left\| \boldsymbol{u}\boldsymbol{u}^\top - \boldsymbol{v}\boldsymbol{v}^\top \right\|_2 \leq \frac{3\tau \cos\theta + \mathcal{O}(\sigma^2 \sqrt{\frac{d + \log(4/\delta)}{N_+}})}{1 - \tau(\sin\theta + 3\cos\theta) - \mathcal{O}(\sigma^2 \sqrt{\frac{d + \log(4/\delta)}{N_+}})} \tag{1}$$

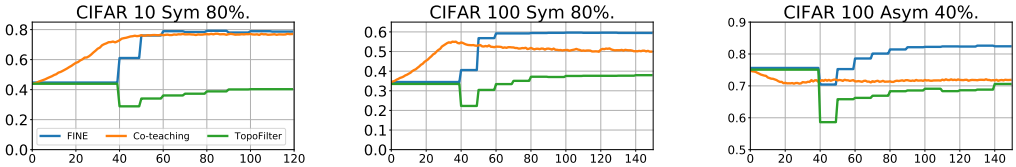
where $\boldsymbol{w}$ is the first eigenvector of noisy instances, τ is the fraction between noisy and clean instances ($\frac{N_-}{N_+}$), θ is $\angle(\boldsymbol{w}, \boldsymbol{v})$, and σ^2 is a variance of white noise.

Theorem 1 states that the upper bound for the perturbation towards $\boldsymbol{v}$ are dependent on both the ratio τ and the angle θ between $\boldsymbol{w}$ and $\boldsymbol{v}$; small upper bound can be guaranteed as the number of clean data increases, τ decreases, and θ approaches $\frac{\pi}{2}$. We also derive the lower bound for the precision and the recall when using the eigenvector $\boldsymbol{u}$ in Appendix. In this theoretical analysis, we can ensure that such lower bound values become larger as $\boldsymbol{v}$ and $\boldsymbol{w}$ go to be orthogonal each other. To verify these intuitions, we provide various experimental results for the separation of *alignment clusterability*,



(a) CIFAR10 (b) CIFAR100 (c) Perturbation (d) Scalability

Figure 3: (a), (b): Heatmaps of the Eq. (2) values on the unit circle in random hyperplane. We evaluate this visualization on the ResNet34 model trained with common cross-entropy loss on CIFAR-10 with asymmetric noise 40% and CIFAR-100 with symmetric noise 80%, respectively. The closer to yellow, the larger the value; (c): comparison of the perturbations of Eq. (1) on CIFAR-10 with symmetric noise 20%; (d): comparison of the cosine similarity values between the FINE's principal components and the approximated principal components using the fraction of data on CIFAR-10 with symmetric noise 80%.



(a) Sym 80% (C10) (b) Sym 80% (C100) (c) Asym 40% (C100)

Figure 4: Comparisons of F-scores on CIFAR10 and CIFAR100 under symmetric and asymmetric label noise. C10 and C100 indicates the abbreviation of CIFAR-10 and CIFAR-100, respectively.

the perturbation values, the scalability to the number of samples, the quality of our detector in the application of sample selection approach, and the comparison with an alternative clustering based estimator [24].

Validation for our estimated eigenvector. To validate our FINE's principal components, we first propose a simple visualization scheme based on the following steps: (1) Pick the first eigenvector ($\boldsymbol{u}$) extracted by FINE algorithm, (2) Generate a random hyperplane spanned by such eigenvector ($\boldsymbol{u}$) and a random vector, (3) Calculate the value of the following Eq. (2) on any unit vectors ($\boldsymbol{a}$) in such hyperplane and plot a heatmap with them:

$$\frac{1}{|\mathbf{X}|}\sum_{\boldsymbol{x}_i\in\mathbf{X}}\langle \boldsymbol{a},\boldsymbol{x}_i\rangle^2-\frac{1}{|\tilde{\mathbf{X}}|}\sum_{\boldsymbol{x}_j\in\tilde{\mathbf{X}}}\langle \boldsymbol{a},\boldsymbol{x}_j\rangle^2 \tag{2}$$

Eq. (2) is maximized when the unit vector $\boldsymbol{a}$ not only maximizes the FINE scores of clean data for the first term in Eq. (2), but also minimizes those of noisy data for the second term in Eq. (2). This visualization shows in 2-D how our FINE's first eigenvector ($\boldsymbol{u}$) optimizes such values in the presence of noisy instances (Figure 3a and 3b). As the figures show, the FINE's eigenvector $\boldsymbol{u}$ (red dotted line) has almost maximum value of Eq. (2). Furthermore, we empirically evaluate the perturbation values in Theorem 1 as the noise rate changes (Figure 3c); FINE has small perturbation values even in a severe noise rate.

Scalability to the number of samples. Despite FINE's superiority, it may require expensive computational costs if the whole data is used for eigen decomposition. To handle this issue, we approximate the eigenvector with a small portion of the dataset and measure the cosine similarity values between the approximated term and the original one ($\boldsymbol{u}$) (Figure 3d). Interestingly, we verify that far accurate eigenvector is computed even using 1% data (i.e., a cosine similarity value is 0.99), and thus the eigenvector can be accurately estimated with little computation time.

Validation for the dynamics of sample-selection approach. We evaluate the F-score dynamics of every training epoch on the symmetric and the asymmetric label noise in Figure 4. We compare the FINE with the following sample selection approaches: Co-teaching [20] and TopoFilter [23].

In Figure 4, during the training process, F-scores of FINE becomes consistently higher on both symmetric noise and asymmetric noise settings, while Co-teaching and TopoFilter achieve lower quality. Unlike TopoFilter and FINE, Co-teaching even performs the sample-selection with the access of noise rate. These evidences show that FINE is also applicable to the naive sample-selection approaches.

Comparison for Mahalanobis distance estimator Under similar conditions, Lee et al. [24] measured the Mahalanobis distance of pre-logits using mininum covariance determinant (MCD) estimator and selected clean samples based on this distance. While they also utilized the LDA assumptions on pre-logits, FINE consistently outperforms MCD in both precision and recall, thus yielding better F-score (Figure 5). The experimental results justify our proposed detector, in comparison with a similar alternative.

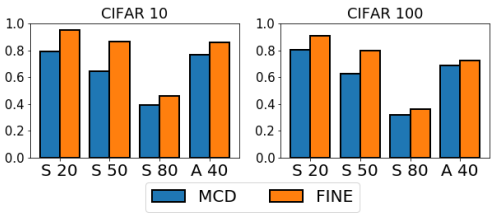


Figure 5: Comparisons of F-scores on CIFAR-10 and CIFAR-100 under symmetric (S) and asymmetric noise (A) settings.

4 Experiment

In this section, we demonstrate the effectiveness of our FINE detector for three applications: sample-selection approach, semi-supervised learning approach, and collaboration with noise-robust loss functions.

4.1 Experimental Settings

Noisy Bechmark Dataset. Following the previous setups [26, 5], we artificially generate two types of random noisy labels: injecting uniform randomness into a fraction of labels (symmetric) and corrupting a label only to a specific class (asymmetric). For example, we generate noise by mapping TRUCK $\rightarrow$ AUTOMOBILE, BIRD $\rightarrow$ AIRPLANE, DEER $\rightarrow$ HORSE, CAT $\leftrightarrow$ DOG to make asymmetric noise for CIFAR-10. For CIFAR-100, we create 20 five-size super-classes and generate asymmetric noise by changing each class to the next class within super-classes circularly. For a real-world dataset, Clothing1M [48] containing inherent noisy labels is used. This dataset contains 1 million clothing images obtained from online shopping websites with 14 classes[1]. The dataset provides $50k, 14k, and 10k$ verified as clean data for training, validation, and testing. Instead of using the $50k$ clean training data, we use a randomly sampled pseudo-balanced subset as a training set with $120k$ images. For evaluation, we compute the classification accuracy on the $10k$ clean dataset.

Networks and Hyperparameter Settings. We use the architectures and hyperparameter settings for all baseline experiments following the setup of Liu et al. [5] except SSL approaches. For SSL approaches, we follow the setup of Li et al. [26]. We set the threshold ζ as 0.5.

4.2 FINE as Application

4.2.1 Sample-Selection based Approaches

We apply our FINE detector for various sample selection algorithms. In detail, after warmup training, at every epoch, FINE select the clean data with the eigenvectors generated from the covariance matrices of data predicted to be clean in the previous round, and then the neural networks are trained with them. We compare our proposed method with the following sample selection approaches.; (1) *Bootstrap* [50], (2) *Forward* [51], (3) *Co-teaching* [20]; (4) *Co-teaching+* [21]; (5) *TopoFilter* [23]; (6) *CRUST* [22]. We evaluate these algorithms for three times and report error bars.

Table 1 summarizes the performances of different sample selection approaches on various noise distribution and datasets. We observe that our FINE method consistently outperforms the competitive methods over the various noise rates. Our FINE methods can filter the clean instances without losing essential information, leading to training the robust network.

To go further, we improve the performances of the Co-teaching [20] by substituting its sample selection state to our FINE algorithm. To combine FINE and Co-teaching families, unlike the original

[1] T-shirt, Shirt, Knitwear, Chiffon, Sweater, Hoodie, Windbreaker, Jacket, Down Coat, Suit, Shawl, Dress, Vest, and Underwear. The labels in this dataset are extremely noisy (estimated noisy level is 38.5%) [49].

Table 1: Test accuracies (%) on CIFAR-10 and CIFAR-100 under different noisy types and fractions. All comparison methods were reproduced with publicly available code while the results for Bootstrap [50] and Forward [51] are taken from [5]. For CRUST [22], we experiment without mixup to compare intrinsic sample selection effect of each method. The average accuracies and standard deviations over three trials are reported. Here, we substitute the sample selection method of Co-teaching [20, 21] to FINE (i.e., F-Coteaching). The best results sharing the noisy fraction and method are highlighted in bold.

Dataset	CIFAR-10				CIFAR-100			
Noisy Type	Sym			Asym	Sym			Asym
Noise Ratio	20	50	80	40	20	50	80	40
Standard	87.0 ± 0.1	78.2 ± 0.8	53.8 ± 1.0	85.0 ± 0.0	58.7 ± 0.3	42.5 ± 0.3	18.1 ± 0.8	42.7 ± 0.6
Bootstrap [50]	86.2 ± 0.2	-	54.1 ± 1.3	81.2 ± 1.5	58.3 ± 0.2	-	21.6 ± 1.0	45.1 ± 0.6
Forward [51]	88.0 ± 0.4	-	54.6 ± 0.4	83.6 ± 0.6	39.2 ± 2.6	-	9.0 ± 0.6	34.4 ± 1.9
Co-teaching [20]	89.3 ± 0.3	83.3 ± 0.6	66.3 ± 1.5	88.4 ± 2.8	63.4 ± 0.0	49.1 ± 0.4	20.5 ± 1.3	47.7 ± 1.2
Co-teaching+ [21]	89.1 ± 0.5	84.9 ± 0.4	63.8 ± 2.3	86.5 ± 1.2	59.2 ± 0.4	47.1 ± 0.3	20.2 ± 0.9	44.7 ± 0.6
TopoFilter [23]	90.4 ± 0.2	86.8 ± 0.3	46.8 ± 1.0	87.5 ± 0.4	66.9 ± 0.4	53.4 ± 1.8	18.3 ± 1.7	56.6 ± 0.5
CRUST [22]	89.4 ± 0.2	87.0 ± 0.1	64.8 ± 1.5	82.4 ± 0.0	69.3 ± 0.2	62.3 ± 0.2	21.7 ± 0.7	56.1 ± 0.5
FINE	**91.0 ± 0.1**	**87.3 ± 0.2**	**69.4 ± 1.1**	**89.5 ± 0.1**	**70.3 ± 0.2**	**64.2 ± 0.5**	**25.6 ± 1.2**	**61.7 ± 1.0**
F-Coteaching	**92.0 ± 0.1**	**87.5 ± 0.1**	**74.2 ± 0.8**	**90.5 ± 0.2**	**71.1 ± 0.2**	**64.7 ± 0.3**	**31.6 ± 1.0**	**64.8 ± 0.7**

Table 2: Comparison of test accuracies (%) for FINE collaborating with DivideMix and existing semi-supervised approaches on CIFAR-10 and CIFAR-100 under different noisy types and fractions. The results for all comparison methods are taken from their original works.

Dataset		CIFAR-10					CIFAR-100			
Noisy Type		Sym				Asym	Sym			
Noise Ratio		20	50	80	90	40	20	50	80	90
DivideMix [26]	Best	96.1	94.6	93.2	76.0	93.4	77.3	74.6	60.2	31.5
	Last	95.7	94.4	92.9	75.4	92.1	76.9	74.2	59.6	31.0
DST [53]	Best	96.1	95.2	92.9	-	94.3	78.0	74.3	57.8	-
	Last	95.9	94.7	92.6	-	92.3	77.4	73.6	55.3	-
LongReMix [52]	Best	96.2	95.0	93.9	82.0	94.7	77.8	75.6	62.9	33.8
	Last	96.0	94.7	93.4	81.3	94.3	77.5	75.1	62.3	33.2
F-DivideMix	Best	96.1	94.9	93.5	**90.5**	93.8	**79.1**	74.6	61.0	**34.3**
	Last	96.0	94.5	93.2	**89.6**	92.8	**78.8**	74.3	60.1	**31.2**

methods that utilize the small loss instances to train with clean labels, we train one model with extracted samples by conducting FINE on another model. The results of the experiments are shown in the eighth and the ninth rows of Table 1.

4.2.2 Semi-Supervised Learning based Approaches

SSL approaches [26, 52, 53] divide the training data into the clean instances as labeled instances and noisy instances as unlabeled instances and use both the labeled and unlabeled samples to train the networks in semi-supervised learning. Recently, methods belong to this category have shown the best performance among the various *LNL* methods, and these methods possibly train robust networks for even extremely high noise rates. We compare the performances of the existing semi-supervised approaches and substituting the sample selection state of DivideMix [26] to our FINE algorithm (i.e., F-Dividemix). The results of the experiments are shown in Table 2. We achieve consistently higher performance than DividMix by utilizing FINE instead of its loss-based filtering method and show comparable performance to the state-of-the-art SSL methods such as DST [53], and LongReMix [52]. Interestingly, as Figure 6 shows, clean and noisy data is well-classified in F-DivideMix under extreme noise case.

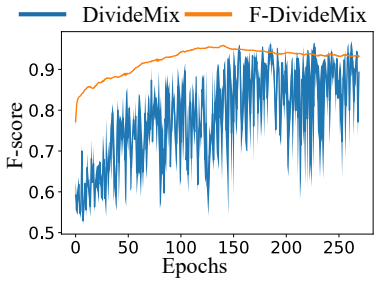
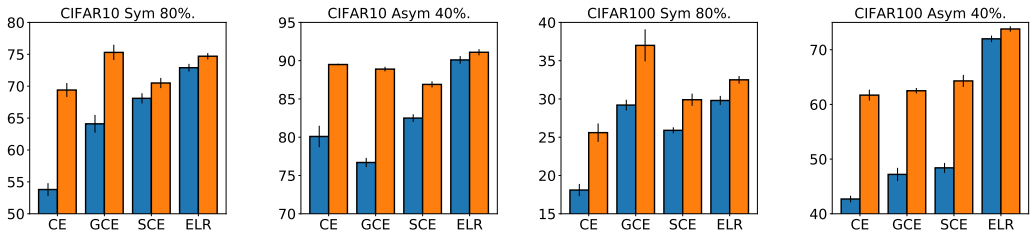


Figure 6: Comparisons of F-scores on CIFAR-10 under symmetric 90% noise. Blue line indicates the error bar of two networks' F-score used in Dividemix [26], and Orange line indicates those replaced by our FINE detector.



(a) Sym 80% (C10) (b) Asym 40% (C10) (c) Sym 80% (C100) (d) Asym 40% (C100)

Figure 7: Test accuracies (%) on CIFAR-10 and CIFAR-100 under different noisy types and fractions for noise-robust loss approaches. Note that the blue box and orange box are results for without and with FINE, respectively. The average accuracies and standard deviations over three trials are reported.

4.2.3 Collaboration with Noise-Robust Loss Functions

The goal of the noise-robust loss function is to achieve a small risk for unseen clean data even when noisy labels exist in the training data. There have been few collaboration studies of the noise-robust loss function methodology and dynamic sample selection. Most studies have selected clean and noisy data based on cross-entropy loss.

Here, we state the conjunction effects of FINE with various noise-robust loss functions: generalized cross entropy (GCE) [17], symmetric cross entropy (SCE) [16], and early-learning regularization (ELR) [5]. Figure 7 shows that FINE facilitates generalization in the application of noise-robust loss functions on severe noise rate settings. The detailed results are reported in Appendix. Unlike other methods, it is still theoretically supported because FINE extracts clean data with a robust classifier using representation.

4.3 Experiments on Real-world Dataset

As Table 3 shows, FINE and F-Dividemix work fairly well on the Clothing 1M dataset compared to other approaches when we reproduce the experimental results under the same settings.

Table 3: Test accuracy on Clothing1M dataset

Method	Standard	GCE [17]	SCE [16]	ELR [5]	DivideMix [26]	CORES^2 [33]	FINE	F-DivideMix
Accuracy	68.94	69.75	71.02	72.87	74.30	73.24	72.91	**74.37**

5 Conclusion

This paper introduces FINE for detecting the label noise by designing a robust noise detector. Our main idea is utilizing the principal components of latent representations made by eigen decomposition. Most existing detection methods are dependent on the loss values, while such losses may be biased by corrupted classifier [24, 25]. Our methodology alleviates this issue by extracting key information from representations, but without using explicit knowledge of the noise rates. We show that FINE detector has a high quality on detecting the noisy labels in theoretical and experimental results. We propose three applications of the FINE detector: sample-selection approach, semi-supervised learning approach, and collaboration with noise-robust loss functions. FINE yields strong results on standard benchmarks and a real-world dataset for various *LNL* approaches.

We believe that our work opens the door to detect the samples having noisy label with explainable results. It is a non-trivial task and of social significance, and thus our work provides a substantial social impact to DL practitioners because it does not require a labor-intensive job for checking data label quality. As future work, we hope that our work will trigger interest in the design of new label-noise detectors and bring a refreshing perspective for other data-resampling approaches, e.g., anomaly detection and novelty detection. The development of robustness against label noise even leads to an improvement in the network's performance trained with data collected through web-crawling. We believe that our contribution will lower the barriers to entry for developing robust models for DL practitioners and greatly impact the internet industry. On the other side, we concern that it can be exploited to train robust models using data collected illegally and indiscriminately in the dark (e.g., web crawling), and thus it may threaten the privacy considerations (e.g., copyright).

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25:1097–1105, 2012.

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

[3] Xiyu Yu, Tongliang Liu, Mingming Gong, and Dacheng Tao. Learning with biased complementary labels. In *Proceedings of the European conference on computer vision (ECCV)*, pages 68–83, 2018.

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

[5] Sheng Liu, Jonathan Niles-Weed, Narges Razavian, and Carlos Fernandez-Granda. Early-learning regularization prevents memorization of noisy labels. *arXiv preprint arXiv:2007.00151*, 2020.

[6] Charles Corbiere, Hedi Ben-Younes, Alexandre Ramé, and Charles Ollion. Leveraging weakly annotated data for fashion image retrieval and label prediction. In *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pages 2268–2274, 2017.

[7] Yilun Xu, Peng Cao, Yuqing Kong, and Yizhou Wang. L_dmi: A novel information-theoretic loss function for training deep nets robust to label noise. In *NeurIPS*, pages 6222–6233, 2019.

[8] Seunghan Yang, Hyoungseob Park, Junyoung Byun, and Changick Kim. Robust federated learning with noisy labels, 2020.

[9] Di Feng, Christian Haase-Schutz, Lars Rosenbaum, Heinz Hertlein, Claudius Glaser, Fabian Timm, Werner Wiesbeck, and Klaus Dietmayer. Deep multi-modal object detection and semantic segmentation for autonomous driving: Datasets, methods, and challenges. *IEEE Transactions on Intelligent Transportation Systems*, 22(3):1341–1360, Mar 2021.

[10] Jinchi Huang, Lie Qu, Rongfei Jia, and Binqiang Zhao. O2u-net: A simple noisy label detection approach for deep neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3326–3334, 2019.

[11] Kaidi Cao, Yining Chen, Junwei Lu, Nikos Arechiga, Adrien Gaidon, and Tengyu Ma. Heteroskedastic and imbalanced deep learning with adaptive regularization. *arXiv preprint arXiv:2006.15766*, 2020.

[12] Zizhao Zhang, Han Zhang, Sercan O Arik, Honglak Lee, and Tomas Pfister. Distilling effective supervision from severe label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9294–9303, 2020.

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

[14] Brendan Van Rooyen, Aditya Krishna Menon, and Robert C Williamson. Learning with symmetric label noise: The importance of being unhinged. *arXiv preprint arXiv:1505.07634*, 2015.

[15] Aritra Ghosh, Himanshu Kumar, and PS Sastry. Robust loss functions under label noise for deep neural networks. In *AAAI*, 2017.

[16] Yisen Wang, Xingjun Ma, Zaiyi Chen, Yuan Luo, Jinfeng Yi, and James Bailey. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 322–330, 2019.

[17] Zhilu Zhang and Mert R Sabuncu. Generalized cross entropy loss for training deep neural networks with noisy labels. *arXiv preprint arXiv:1805.07836*, 2018.

[18] Mingchen Li, Mahdi Soltanolkotabi, and Samet Oymak. Gradient descent with early stopping is provably robust to label noise for overparameterized neural networks. In *International Conference on Artificial Intelligence and Statistics*, pages 4313–4324. PMLR, 2020.

[19] Lu Jiang, Zhengyuan Zhou, Thomas Leung, Li-Jia Li, and Li Fei-Fei. Mentornet: Learning data-driven curriculum for very deep neural networks on corrupted labels, 2018.

[20] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. In *Advances in neural information processing systems*, pages 8527–8537, 2018.

[21] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor W Tsang, and Masashi Sugiyama. How does disagreement help generalization against label corruption? *arXiv preprint arXiv:1901.04215*, 2019.

[22] Baharan Mirzasoleiman, Kaidi Cao, and Jure Leskovec. Coresets for robust training of neural networks against noisy labels. *arXiv preprint arXiv:2011.07451*, 2020.

[23] Pengxiang Wu, Songzhu Zheng, Mayank Goswami, Dimitris Metaxas, and Chao Chen. A topological filter for learning with label noise. *arXiv preprint arXiv:2012.04835*, 2020.

[24] Kimin Lee, Sukmin Yun, Kibok Lee, Honglak Lee, Bo Li, and Jinwoo Shin. Robust inference via generative classifiers for handling noisy labels. In *International Conference on Machine Learning*, pages 3763–3772. PMLR, 2019.

[25] Hartmut Maennel, Ibrahim Alabdulmohsin, Ilya Tolstikhin, Robert JN Baldock, Olivier Bousquet, Sylvain Gelly, and Daniel Keysers. What do neural networks learn when trained with random labels? *arXiv preprint arXiv:2006.10455*, 2020.

[26] Junnan Li, Richard Socher, and Steven C.H. Hoi. Dividemix: Learning with noisy labels as semi-supervised learning. In *International Conference on Learning Representations*, 2020.

[27] Dan Hendrycks, Mantas Mazeika, Duncan Wilson, and Kevin Gimpel. Using trusted data to train deep networks on labels corrupted by severe noise. *arXiv preprint arXiv:1802.05300*, 2018.

[28] Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to reweight examples for robust deep learning. In *International Conference on Machine Learning*, pages 4334–4343. PMLR, 2018.

[29] Andreas Veit, Neil Alldrin, Gal Chechik, Ivan Krasin, Abhinav Gupta, and Serge Belongie. Learning from noisy large-scale datasets with minimal supervision. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 839–847, 2017.

[30] Zizhao Zhang, Han Zhang, Sercan O. Arik, Honglak Lee, and Tomas Pfister. Distilling effective supervision from severe label noise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[31] Dara Bahri, Heinrich Jiang, and Maya Gupta. Deep k-nn for noisy labels. In *International Conference on Machine Learning*, pages 540–550. PMLR, 2020.

[32] Geoff Pleiss, Tianyi Zhang, Ethan R Elenberg, and Kilian Q Weinberger. Identifying mislabeled data using the area under the margin ranking. *arXiv preprint arXiv:2001.10528*, 2020.

[33] Hao Cheng, Zhaowei Zhu, Xingyu Li, Yifei Gong, Xing Sun, and Yang Liu. Learning with instance-dependent label noise: A sample sieve approach. *arXiv preprint arXiv:2010.02347*, 2020.

[34] David Berthelot, Nicholas Carlini, Ian Goodfellow, Nicolas Papernot, Avital Oliver, and Colin A Raffel. Mixmatch: A holistic approach to semi-supervised learning. In *Advances in Neural Information Processing Systems*, pages 5049–5059, 2019.

[35] Devansh Arpit, Stanisław Jastrzębski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, et al. A closer look at memorization in deep networks. In *International Conference on Machine Learning*, pages 233–242. PMLR, 2017.

[36] Xiaobo Xia, Tongliang Liu, Bo Han, Chen Gong, Nannan Wang, Zongyuan Ge, and Yi Chang. Robust early-learning: Hindering the memorization of noisy labels. In *International Conference on Learning Representations*, 2021.

[37] Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. Does label smoothing mitigate label noise? In *International Conference on Machine Learning*, pages 6448–6458. PMLR, 2020.

[38] Taehyeon Kim, Jaehoon Oh, NakYil Kim, Sangwook Cho, and Se-Young Yun. Comparing kullback-leibler divergence and mean squared error loss in knowledge distillation, 2021.

[39] Nitesh V Chawla, Kevin W Bowyer, Lawrence O Hall, and W Philip Kegelmeyer. Smote: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 16:321–357, 2002.

[40] Shin Ando and Chun Yuan Huang. Deep over-sampling framework for classifying imbalanced data. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 770–785. Springer, 2017.

[41] Mateusz Buda, Atsuto Maki, and Maciej A Mazurowski. A systematic study of the class imbalance problem in convolutional neural networks. *Neural Networks*, 106:249–259, 2018.

[42] Yi Li and Nuno Vasconcelos. Repair: Removing representation bias by dataset resampling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9572–9581, 2019.

[43] Ronan Le Bras, Swabha Swayamdipta, Chandra Bhagavatula, Rowan Zellers, Matthew Peters, Ashish Sabharwal, and Yejin Choi. Adversarial filters of dataset biases. In *International Conference on Machine Learning*, pages 1078–1088. PMLR, 2020.

[44] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

[45] Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017.

[46] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, pages 7167–7177, 2018.

[47] Ronald A Fisher. The use of multiple measurements in taxonomic problems. *Annals of eugenics*, 7(2):179–188, 1936.

[48] Tong Xiao, Tian Xia, Yi Yang, Chang Huang, and Xiaogang Wang. Learning from massive noisy labeled data for image classification. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2691–2699, 2015.

[49] Hwanjun Song, Minseok Kim, Dongmin Park, and Jae-Gil Lee. Prestopping: How does early stopping help generalization against label noise?, 2020.

[50] Scott Reed, Honglak Lee, Dragomir Anguelov, Christian Szegedy, Dumitru Erhan, and Andrew Rabinovich. Training deep neural networks on noisy labels with bootstrapping. *arXiv preprint arXiv:1412.6596*, 2014.

[51] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1944–1952, 2017.

[52] Filipe R Cordeiro, Ragav Sachdeva, Vasileios Belagiannis, Ian Reid, and Gustavo Carneiro. Longremix: Robust learning with high confidence samples in a noisy label environment. *arXiv preprint arXiv:2103.04173*, 2021.

[53] Yi Wei, Xue Mei, Xin Liu, and Pengxiang Xu. Dst: Data selection and joint training for learning with noisy labels. *arXiv preprint arXiv:2103.00813*, 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See abstract and Section 1. We categorize our key contributions and provide applicable areas.
   (b) Did you describe the limitations of your work? [Yes] We describe the limitations in subsection 3.1. Main concern is that our algorithm may require expensive computational costs when using the whole dataset (scalability to the number of samples). However, we also find that the eigenvector can be well-approximated using 1% portion of the dataset, and thus such limitation can be significantly mitigated.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See the end of Section 5. We discuss the unintended uses in the dark.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes] We've read the ethics review guidelines. We ensure that our paper satisfies them.
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See subsection 3.1. We clearly state the assumptions and provide theoretical guarantees for our algorithms in this part.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See Section 4 and Appendix.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section 4 and Appendix.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] We report error bars for subsubsection 4.2.1 and subsubsection 4.2.3. On the other side, we report the best and last accuracy for subsubsection 4.2.2.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] We report the total amount of compute and the type of resources in Appendix.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See subsection 4.1
   (b) Did you mention the license of the assets? [N/A] Some code and data are proprietary.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We include code for our algorithms in URL
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [Yes] We have an approval form the creaters to use the data for publication.
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A] We ensure that our work does not correspond to the question situation.
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A] We ensure that our work does not correspond to the question situation.
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A] We ensure that our work does not correspond to the question situation.
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A] We ensure that our work does not correspond to the question situation.