Confident Learning: Estimating Uncertainty in Dataset Labels

Curtis G. Northcutt [1] **Lu Jiang** [2] **Isaac L. Chuang** [1]

Abstract

Learning exists in the context of data, yet notions of *confidence* typically focus on model predictions, not label quality. Confident learning (CL) has emerged as an approach for characterizing, identifying, and learning with noisy labels in datasets, based on the principles of pruning noisy data, counting to estimate noise, and ranking examples to train with confidence. Here, we generalize CL, building on the assumption of a classification noise process, to directly estimate the joint distribution between noisy (given) labels and uncorrupted (unknown) labels. This generalized CL, open-sourced as `cleanlab`, is provably consistent across reasonable conditions, and experimentally performant on ImageNet and CIFAR, outperforming seven recent approaches when label noise is non-uniform. `cleanlab` also quantifies ontological class overlap, and can increase model accuracy (e.g. ResNet) by providing clean data for training.

1. Introduction

Advances in learning with noisy labels and weak supervision usually introduce a new model or loss function. Often this model-centric approach band-aids the real question: which data is mislabeled? Here, we take a data-centric approach and establish theoretical and experimental evidence that a key to learning with noisy labels lies in accurately characterizing the uncertainty of label noise in the data directly.

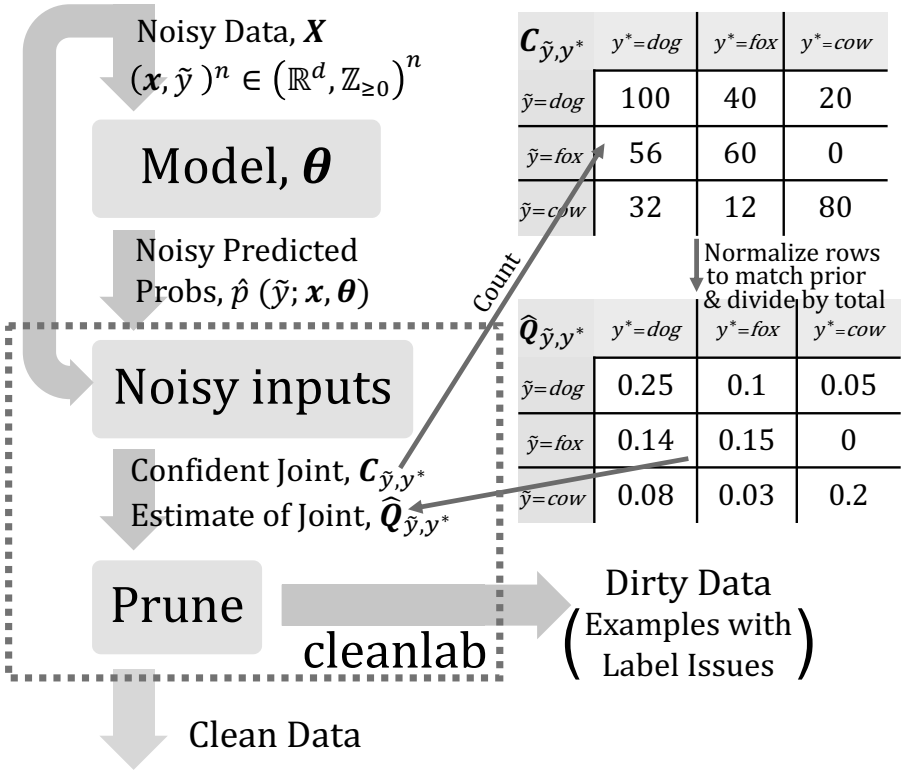
A large body of work, which may be termed "confident learning," has arisen to address the uncertainty in dataset labels, from which two aspects stand out. First, (Angluin & Laird, 1988)'s classification noise process (CNP) provides a starting assumption, that label noise is class-conditional, depending only on the latent true class, not the data. While there are exceptions, this assumption is commonly used (Goldberger & Ben-Reuven, 2017; Sukhbaatar et al., 2015) because it is reasonable. For example, in ImageNet, a *leopard* is more likely to be mislabeled *jaguar* than *bathtub*. Second, direct estimation of the joint distribution between noisy (given) labels and true (unknown) labels (see Fig. 1) can be pursued effectively based on three principled approaches: (a) **Prune**, to search for label errors, e.g. following the example of (Chen et al., 2019; Patrini et al., 2017; Van Rooyen et al., 2015), using *soft-pruning* via loss-reweighting, to avoid the convergence pitfalls of iterative re-labeling – (b) **Count**, to train on clean data, avoiding error-propagation in learned model weights from reweighting the loss (Natarajan et al., 2017) with imperfect predicted probabilities, generalizing seminal work (Forman, 2005; 2008; Lipton et al., 2018) – and (c) **Rank** which examples to use during training, to allow learning with unnormalized probabilities or decision boundary distances, building on well-known robustness findings (Page et al., 1997) and ideas of curriculum learning (Jiang et al., 2018).

To our knowledge, no prior work has thoroughly analyzed direct estimation of the joint distribution between noisy and uncorrupted labels. Here, we assemble these principled approaches to generalize confident learning (CL) for this purpose. Estimating the joint distribution is challenging, but useful because its marginals yield important statistics used in the literature, including latent noise transition rates (Sukhbaatar et al., 2015; Reed et al., 2015), latent prior of uncorrupted labels (Lawrence & Schölkopf, 2001; Graepel & Herbrich, 2001), and inverse noise rates (Katz-Samuels et al., 2019). While noise rates are useful for loss-reweighting (Natarajan et al., 2013), only the joint can directly estimate the number of label errors for each pair of true and noisy classes. Removal of these errors prior to training is an effective approach for learning with noisy labels (Chen et al., 2019). The joint is also useful to discover ontological issues in datasets, e.g. ImageNet includes two classes for the same *maillot* class (c.f. Table 3 in Sec. 5).

The resulting CL procedure (Fig. 1) is a model-agnostic family of theory and algorithms for characterizing, finding, and learning with label errors. It uses predicted probabilities and noisy labels to *count* examples in the unnormalized *confident joint*, estimate the joint distribution, and *prune* noisy data, producing clean data as output.

[1]Massachusetts Institute of Technology, Department of EECS, Cambridge, MA, USA [2]Google, Mountain View, CA, USA. Correspondence to: Curtis G. Northcutt <cgn@mit.edu>.

Under review by the *International Conference on Machine Learning (ICML)*.



$\boldsymbol{C}_{\tilde{y},y^*}$	y^*=dog	y^*=fox	y^*=cow
$\tilde{y}$=dog	100	40	20
$\tilde{y}$=fox	56	60	0
$\tilde{y}$=cow	32	12	80

$\widehat{\boldsymbol{Q}}_{\tilde{y},y^*}$	y^*=dog	y^*=fox	y^*=cow
$\tilde{y}$=dog	0.25	0.1	0.05
$\tilde{y}$=fox	0.14	0.15	0
$\tilde{y}$=cow	0.08	0.03	0.2

Figure 1. The confident learning (CL) process with examples of $\boldsymbol{C}_{\tilde{y},y^*}$, the confident joint, and $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$, the estimated joint distribution of noisy observed labels $\tilde{y}$ and latent uncorrupted labels y^*.

This paper makes three key contributions to CL. First, we prove CL exactly estimates the joint distribution of noisy and true labels with exact identification of label errors under realistic sufficient conditions. Second, we show CL is empirically performant on three tasks (a) label noise estimation, (b) label error finding, and (c) learning with noisy labels, increasing ResNet accuracy on a cleaned-ImageNet and outperforming seven recent state-of-the-art methods for learning with noisy labels. Finally, we open-sourced `cleanlab`[1] as standard Python package to reproduce all results and support future research in weak supervision.

Our contributions can be summarized as follows:

1. Proposed confident learning for characterizing, finding, & learning with label errors in datasets.
2. Proved non-trivial conditions for consistent joint estimation and exactly finding label errors.
3. Verified the efficacy of CL on CIFAR (added label noise) and ImageNet (real label noise).
4. Released the `cleanlab` Python package for accessibility and reproducibility.

[1] `cleanlab` for finding and learning with noisy labels is open-source: https://github.com/cgnorthcutt/cleanlab/

2. Framework

Here, we consider standard multiclass classification with possibly noisy labels. Let $\{1..m\}$ denote the set of m unique class labels and $\boldsymbol{X} := (\boldsymbol{x}, \tilde{y})^n \in (\mathbb{R}^d, \mathbb{N}_{>0})^n$ denote the set of n examples $\boldsymbol{x} \in \mathbb{R}^d$ with associated observed noisy labels $\tilde{y} \in \mathbb{N}_{>0}$. We couple $\boldsymbol{x}$ and $\tilde{y}$ in $\boldsymbol{X}$ to signify that *cleaning* removes data and label.

Assumptions. Prior to observing $\tilde{y}$, we assume a class-conditional classification noise process (CNP) (Angluin & Laird, 1988) maps $y^* \to \tilde{y}$ such that every label in class $j \in 1..m$ may be independently mislabeled as class $i \in 1..m$ with probability $p(\tilde{y}{=}i|y^*{=}j)$. This assumption is reasonable and has been used in prior work (Goldberger & Ben-Reuven, 2017; Sukhbaatar et al., 2015).

Notation. All notation is summarized in the Appendix (see Table 4). The discrete random variable $\tilde{y}$ takes an observed, noisy label (potentially flipped to an incorrect class), and y^* takes a latent, uncorrupted label. The subset of examples in $\boldsymbol{X}$ with noisy label i is denoted $\boldsymbol{X}_{\tilde{y}=i}$, *i.e.* $\boldsymbol{X}_{\tilde{y}=\text{cat}}$ is read, "examples labeled cat." The notation $p(\tilde{y}; x)$, as opposed to $p(\tilde{y}|x)$, expresses our assumption that input $\boldsymbol{x}$ is deterministic and error-free. We denote the discrete joint probability of the noisy and latent labels as $p(\tilde{y}, y^*)$, where conditionals $p(\tilde{y}|y^*)$ and $p(y^*|\tilde{y})$ denote probabilities of la-

bel flipping. We use $\hat{p}$ for predicted probabilities. In matrix notation, the $n \times m$ matrix of out-of-sample predicted probabilities is $\hat{\boldsymbol{P}}_{k,i} \coloneqq \hat{p}(\tilde{y}=j; \boldsymbol{x}_k, \boldsymbol{\theta})$, the prior of the latent labels is $\boldsymbol{Q}_{y^*} \coloneqq p(\tilde{y}{=}i)$; the $m \times m$ joint distribution matrix is $\boldsymbol{Q}_{\tilde{y},y^*} \coloneqq p(\tilde{y}{=}i, y^*{=}j)$; the $m \times m$ noise transition matrix (noisy channel) of flipping rates is $\boldsymbol{Q}_{\tilde{y}|y^*} \coloneqq p(\tilde{y}{=}i|y^*{=}j)$; and the $m \times m$ inverse noise matrix is $\boldsymbol{Q}_{y^*|\tilde{y}} \coloneqq p(y^*{=}i|\tilde{y}{=}j)$. At times, we abbreviate $\hat{p}(\tilde{y}=i; \boldsymbol{x}, \boldsymbol{\theta})$ as $\hat{p}_{\boldsymbol{x},\tilde{y}=i}$, where $\boldsymbol{\theta}$ denotes the model parameters. CL assumes no specific loss function: this framework is model-agnostic.

Goal. CNP implies a data-independent noise transition probability, namely $p(\tilde{y}|y^*; \boldsymbol{x}) = p(\tilde{y}|y^*)$ as well as $p(y^*|\tilde{y}; \boldsymbol{x}) = p(y^*|\tilde{y})$. Thereby, $p(\tilde{y}, y^*; \boldsymbol{x}) = p(\tilde{y}, y^*)$, completely characterizes noise and our goal is to estimate the complete matrix $\boldsymbol{Q}_{\tilde{y},y^*}$ and use it to find errors in $\boldsymbol{X}$.

Definition. *Sparsity* is the fraction of zeros in the off-diagonals of $\boldsymbol{Q}_{\tilde{y},y^*}$. High sparsity quantifies non-uniformity of label noise, common to real-world datasets. For example, in ImageNet, *missile* may have high probability of being mislabeled as *projectile*, but insignificant probability of being mislabeled as most other classes like *wool*, *ox*, or *wine*.

Definition. *Self-Confidence* is the predicted probability that an example $\boldsymbol{x}$ belongs to its given label $\tilde{y}$, expressed as $\hat{p}(\tilde{y}{=}i; \boldsymbol{x}{\in}\boldsymbol{X}_{\tilde{y}=i}, \boldsymbol{\theta})$. Low self-confidence is a heuristic likelihood of being a label error.

3. CL Methods

Confident learning estimates the joint distribution between the (noisy) observed labels and the (true) latent labels and can be used to (i) improve training with noisy labels, and (ii) identify noisy labels in existing datasets. The main procedure consists of three steps: (1) estimate the joint $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$ to characterize class-conditional label noise, (2) filter out noisy examples, and (3) train with errors removed via Co-Teaching (Han et al., 2018), re-weighting examples by class weights $\frac{\hat{\boldsymbol{Q}}_{y^*}[i]}{\hat{\boldsymbol{Q}}_{\tilde{y},y^*}[i][i]}$ for each class $i \in 1..m$. In this section, we define these three steps and discuss their expected outcomes. Only two inputs are used: out-of-sample predicted probabilities $\hat{\boldsymbol{P}}_{k,i}$ and the array of noisy labels $\tilde{y}_k$, sharing index k. Our method requires no hyperparameters.

3.1. Count: Label Noise Characterization

We estimate $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$ by counting examples in the joint distribution, calibrating estimated counts using the given count of noisy labels in each class, $|\boldsymbol{X}_{\tilde{y}=i}|$, then normalizing. Counts are captured by the *confident joint* $\boldsymbol{C}_{\tilde{y},y^*} \in \mathbb{Z}_{\geq 0}^{m \times m}$, the key structure of confident learning. Diagonal entries of $\boldsymbol{C}_{\tilde{y},y^*}$ count correct labels and non-diagonals capture asymmetric label error counts. As an example, $C_{\tilde{y}=3,y^*=1}{=}10$ is read, "Ten examples are labeled *3* but should be labeled *1*."

Confusion matrix $\boldsymbol{C}_{\text{confusion}}$. $\boldsymbol{C}_{\tilde{y},y^*}$ may be constructed as a confusion matrix of given labels $\tilde{y}_k$ and predictions $\arg\max_{i \in 1..m} \hat{p}(\tilde{y}{=}i; \boldsymbol{x}_k, \boldsymbol{\theta})$. This approach performs reasonably empirically (Sec. 5) and is a consistent estimator for noiseless predicted probabilities (Thm. 1), but fails when the distributions of probabilities are not similar for each class (Thm. 2). We deal with this sensitivity in $\boldsymbol{C}_{\tilde{y},y^*}$ via thresholding (Richard & Lippmann, 1991; Elkan, 2001).

The confident joint $\boldsymbol{C}_{\tilde{y},y^*}$. $\boldsymbol{C}_{\tilde{y},y^*}$ bins examples $\boldsymbol{x}$ labeled $\tilde{y}{=}i$ with *large enough* $\hat{p}_{\boldsymbol{x},\tilde{y}=j}$ to likely belong to label $y^*{=}j$. As a first try, we express $\boldsymbol{C}_{\tilde{y},y^*}$ as

$$\boldsymbol{C}_{\tilde{y},y^*}[i][j] \overset{?}{\coloneqq} |\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}| \text{ , where} \quad (1)$$
$$\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i} : \ \hat{p}(\tilde{y}=j; \boldsymbol{x}, \boldsymbol{\theta}) \geq t_j\}$$

and the threshold t_j is the expected (average) self-confidence for each class.

$$t_j = \frac{1}{|\boldsymbol{X}_{\tilde{y}=j}|} \sum_{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=j}} \hat{p}(\tilde{y}=j; \boldsymbol{x}, \boldsymbol{\theta}) \quad (2)$$

This formulation fixes the problems with $\boldsymbol{C}_{\text{confusion}}$ so that $\boldsymbol{C}_{\tilde{y},y^*}$ is robust for any particular class with large or small probabilities, but introduces *label collisions* when an example x is confidently counted into more than one $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$ bin. Collisions only occur along the y^* dimension of $\boldsymbol{C}_{\tilde{y},y^*}$ because $\tilde{y}$ is given. We handle collisions by selecting $\hat{y}^* \leftarrow \arg\max_{j \in 1..m} \hat{p}_{\boldsymbol{x},\tilde{y}=j}$. The result (Eqn. 3) defines the confident joint, $\boldsymbol{C}_{\tilde{y},y^*}$:

$$\boldsymbol{C}_{\tilde{y},y^*}[i][j] \coloneqq |\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}| \quad \text{where} \quad \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} \coloneqq$$
$$\left\{ \boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i} : \ \hat{p}_{\boldsymbol{x},\tilde{y}=j} \geq t_j, \ j = \underset{k \in 1..m : \hat{p}_{\boldsymbol{x},\tilde{y}=k} \geq t_k}{\arg\max} \hat{p}_{\boldsymbol{x},\tilde{y}=k} \right\} \quad (3)$$

where the $j{=}\arg\max$ term only matters when $|\{k{\in}1..m : \hat{p}(\tilde{y}{=}k; \boldsymbol{x}{\in}\boldsymbol{X}_{\tilde{y}=i}, \boldsymbol{\theta}) \geq t_k\}| > 1$ (collision). In practice with softmax, collisions sometimes occur for softmax outputs with low temperature, few collisions occur with high temperature, and no collisions occur as the temperature $\to \infty$ because this reverts to $\boldsymbol{C}_{\text{confusion}}$.

$\boldsymbol{C}_{\tilde{y},y^*}$ (Eqn. 3) has some nice properties. First, if an example has low (near-uniform) probabilities across classes, it is not counted so that $\boldsymbol{C}_{\tilde{y},y^*}$ is robust to examples from an alien class not in the dataset. Second, t_j embodies the intuition that examples with higher probability of belonging to class j than the expected probability of examples in class j probably belong to class j. Third, the 90^{th} percentile may be used in t_j instead of the mean for higher confidence.

Complexity. We provide algorithmic implementations of Eqns. 2, 3, and 4 in the Appendix. Given predicted probabilities $\hat{\boldsymbol{P}}_{k,i}$ and noisy labels $\tilde{y}$, these require $\mathcal{O}(m^2 + nm)$ operations to store and compute $\boldsymbol{C}_{\tilde{y},y^*}$.

Estimate the joint $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$. Given the confident joint $\boldsymbol{C}_{\tilde{y},y^*}$, we estimate $\boldsymbol{Q}_{\tilde{y},y^*}$ as, $\hat{\boldsymbol{Q}}_{\tilde{y}=i,y^*=j} =$

$$\frac{\frac{\boldsymbol{C}_{\tilde{y}=i,y^*=j}}{\sum_{j\in 1..m}\boldsymbol{C}_{\tilde{y}=i,y^*=j}}\cdot|\boldsymbol{X}_{\tilde{y}=i}|}{\sum_{i\in 1..m,j\in 1..m}\left(\frac{\boldsymbol{C}_{\tilde{y}=i,y^*=j}}{\sum_{j\in 1..m}\boldsymbol{C}_{\tilde{y}=i,y^*=j}}\cdot|\boldsymbol{X}_{\tilde{y}=i}|\right)} \tag{4}$$

The numerator calibrates $\sum_j \hat{\boldsymbol{Q}}_{\tilde{y}=i,y^*=j} = |\boldsymbol{X}_i|/\sum_{i\in 1..m}|\boldsymbol{X}_i|, \forall i\in 1..m$ so that row-sums match the observed marginals. The denominator calibrates $\sum_{i,j}\hat{\boldsymbol{Q}}_{\tilde{y}=i,y^*=j}=1$ so the distribution sums to 1.

Label noise characterization Using the observed prior $\boldsymbol{Q}_{\tilde{y}=i} = |\boldsymbol{X}_i|\,/\sum_{i\in 1..m}|\boldsymbol{X}_i|$ and marginals of $\boldsymbol{Q}_{\tilde{y},y^*}$, we estimate the latent prior as $\hat{\boldsymbol{Q}}_{y^*=j} \coloneqq \sum_i \hat{\boldsymbol{Q}}_{\tilde{y}=i,y^*=j}, \forall j\in 1..m$; the noise transition matrix (noisy channel) as $\hat{\boldsymbol{Q}}_{\tilde{y}=i|y^*=j} \coloneqq \hat{\boldsymbol{Q}}_{\tilde{y}=i,y^*=j}/\hat{\boldsymbol{Q}}_{y^*=j}, \forall i\in 1..m$; and the inverse noise matrix as $\hat{\boldsymbol{Q}}_{y^*=j|\tilde{y}=i} \coloneqq \hat{\boldsymbol{Q}}^{\top}_{\tilde{y}=j,y^*=i}/\boldsymbol{Q}_{\tilde{y}=i}, \forall i\in 1..m$. Whereas prior approaches estimate the noise transition matrices from error-prone predicted probabilities (Reed et al., 2015; Goldberger & Ben-Reuven, 2017), as demonstrated empirically (see Fig. 2), CL marginalizes the joint directly in favor of robustness to imperfect probability estimation.

3.2. Rank and Prune: Data Cleaning

Following estimation of the joint, we apply pruning, ranking, and other heuristics for cleaning training data. Two approaches are: (1) use the off-diagonals of $\boldsymbol{C}_{\tilde{y},y^*}$ or (2) use $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$ to estimate the number of label errors and remove errors by ranking over predicted probability. Sec. 4 and the first two methods below examine the first approach, while the second is addressed by the last three methods below:

Method: $\boldsymbol{C}_{\text{confusion}}$. Estimate label errors as $\mathbb{1}[[\tilde{y}_k \neq \arg\max_{j\in 1..m}\hat{p}(\tilde{y}=j;\boldsymbol{x}_k,\boldsymbol{\theta})]]$ for any $\boldsymbol{x}_k\in\boldsymbol{X}$. This is identical to using the off-diagonals of $\boldsymbol{C}_{\text{confusion}}$ and similar to INCV (Chen et al., 2019), but with only one iteration.

Method: $\boldsymbol{C}_{\tilde{y},y^*}$. Estimate label errors as $\{\boldsymbol{x}\in\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} : i\neq j\}$ from the off-diagonals of $\boldsymbol{C}_{\tilde{y},y^*}$.

Method: Prune by Class (PBC). For each class $i\in 1..m$, select the $n\cdot\sum_{j\in 1..m:j\neq i}\left(\hat{\boldsymbol{Q}}_{\tilde{y}=i,y^*=j}[i]\right)$ examples with lowest self-confidence $\hat{p}(\tilde{y}=i;\boldsymbol{x}\in\boldsymbol{X}_i)$.

Method: Prune by Noise Rate (PBNR). For each off-diagonal entry in $\boldsymbol{C}_{\tilde{y},y^*}$, select the $n\cdot\hat{\boldsymbol{Q}}_{\tilde{y}=i,y^*=j}$ examples $\boldsymbol{x}\in\boldsymbol{X}_{\tilde{y}=i}$ with max margin $\hat{p}_{\boldsymbol{x},\tilde{y}=j}-\hat{p}_{\boldsymbol{x},\tilde{y}=i}$.

Method: C+NR. Combine the previous two methods via element-wise '*and*', i.e. set intersection.

Which CL method to use? CL requires no hyper-parameters, but five methods are presented to clean data. By default, we use CL: PBNR because it most closely matches the conditions of Thm. 2 by *pruning* for each off-diagonal in $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$. This choice is justified experimentally in Table 2. Once label errors are found, we observe ordering label errors by the normalized margin: $\hat{p}(\tilde{y}=i;\boldsymbol{x},\boldsymbol{\theta}) - \max_{j\neq i}\hat{p}(\tilde{y}=j;\boldsymbol{x},\boldsymbol{\theta})$ (Wei et al., 2018) works well. To train with errors removed, we use Co-Teaching with standard settings and re-weight the loss by $\frac{1}{\hat{p}(\tilde{y}=i|y^*=i)}=\frac{\hat{\boldsymbol{Q}}_{y^*}[i]}{\hat{\boldsymbol{Q}}_{\tilde{y},y^*}[i][i]}$ for each class $i\in 1..m$.

4. Theory

In this section, we examine sufficient conditions when (1) the confident joint exactly finds label errors and (2) $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$ is a consistent estimator for $\boldsymbol{Q}_{\tilde{y},y^*}$. We first analyze CL for noiseless $\hat{p}_{\boldsymbol{x},\tilde{y}=j}$, then evaluate more realistic conditions, culminating in Thm. 2 where we prove (1) and (2) with noise in predicted probabilities for every example. Proofs are in the Appendix (see Sec. B).

In the statement of the theorems, we use $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}\approxeq\boldsymbol{Q}_{\tilde{y},y^*}$, i.e. *approximately equals*, to account for precision error of using discrete count-based $\boldsymbol{C}_{\tilde{y},y^*}$ to estimate real-valued $\boldsymbol{Q}_{\tilde{y},y^*}$. For example, if a noise rate is 0.39, but the dataset has only 5 examples in that class, the nearest possible estimate by removing errors is $2/5=0.4\approxeq 0.39$. Otherwise, all equalities are exact. Throughout, we assume $\boldsymbol{X}$ includes at last one example from every class.

4.1. Noiseless Predicted Probabilities

We start with the *ideal* condition and a non-obvious lemma that yields a closed-form expression for t_j when $\hat{p}_{\boldsymbol{x},\tilde{y}=j}$ is ideal. Without some condition on $\hat{p}_{\boldsymbol{x},\tilde{y}=j}$, one cannot disambiguate label noise from model noise.

Condition (Ideal). The predicted probs $\hat{p}(\tilde{y};\boldsymbol{x},\boldsymbol{\theta})$ for a model θ are *ideal* if $\forall\boldsymbol{x}_k\in\boldsymbol{X}_{y^*=j}, i\in 1..m, j\in 1..m$, $\hat{p}(\tilde{y}=i;\boldsymbol{x}_k\in\boldsymbol{X}_{y^*=j},\boldsymbol{\theta}) = p^*(\tilde{y}=i|y^*=y_k^*) = p^*(\tilde{y}=i|y^*=j)$, where the last equality follows from the CNP assumption. The *ideal* condition implies error-free predicted probabilities: they match the noise rates of the y^* label corresponding to $\boldsymbol{x}$. We use $p^*_{\boldsymbol{x},\tilde{y}=i}$ as shorthand.

Lemma 1 (Ideal Thresholds). *For noisy dataset $\boldsymbol{X}\coloneqq(\boldsymbol{x},\tilde{y})^n\in(\mathbb{R}^d,\mathbb{N}_{>0})^n$ and model $\boldsymbol{\theta}$, if $\hat{p}(\tilde{y};\boldsymbol{x},\boldsymbol{\theta})$ is* ideal, *then* $\forall i\in 1..m, t_i=\sum_{j\in 1..m}p(\tilde{y}=i|y^*=j)p(y^*=j|\tilde{y}=i)$.

This form of the threshold is intuitively reasonable: the contributions to the sum when $i=j$ represents the probabilities of correct labeling, whereas when $i\neq j$, the terms give the probabilities of mislabeling $p(\tilde{y}=i|y^*=j)$, weighted by the probability $p(y^*=j|\tilde{y}=i)$ that the mislabeling is corrected. Using Lemma 1 under the ideal condition we prove in Thm. 1 confident learning exactly finds label errors and $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$ is a consistent estimator for $\boldsymbol{Q}_{\tilde{y},y^*}$ when each diagonal entry of $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximizes its row and column. The proof hinges on the fact that the construction of $\boldsymbol{C}_{\tilde{y},y^*}$

eliminates collisions.

Theorem 1 (Exact Label Errors)**.** *For a noisy dataset* $\boldsymbol{X} \coloneqq (\boldsymbol{x}, \tilde{y})^n \in (\mathbb{R}^d, \mathbb{N}_{>0})^n$ *and model* $\boldsymbol{\theta}{:}\boldsymbol{x}\to\hat{p}(\tilde{y})$*, if* $\hat{p}(\tilde{y}; \boldsymbol{x}, \boldsymbol{\theta})$ *is ideal and each diagonal entry of* $\boldsymbol{Q}_{\tilde{y}|y^*}$ *maximizes its row and column, then* $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$ *and* $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} \approxeq \boldsymbol{Q}_{\tilde{y},y^*}$ *(consistent estimator for* $\boldsymbol{Q}_{\tilde{y},y^*}$*).*

While Thm. 1 is a reasonable sanity check, observe that $y^* \leftarrow \arg\max_j \hat{p}(\tilde{y}{=}i|\tilde{y}^*{=}i; \boldsymbol{x})$, used by $\boldsymbol{C}_{\text{confusion}}$, trivially satisfies Thm. 1 under the assumption that the diagonal of $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximizes its row and column. We next consider conditions motivated by real-world settings where this is no longer the case.

4.2. Noisy Predicted Probabilities

Motivated by the importance of addressing class imbalance, we consider linear combinations of noise per-class.

Condition (Per-Class Diffracted). $\hat{p}_{\boldsymbol{x},\tilde{y}=i}$ is *per-class diffracted* if there exist linear combinations of class-conditional error in the predicted probabilities s.t. $\hat{p}_{\boldsymbol{x},\tilde{y}=i} = \epsilon_i^{(1)} p^*_{\boldsymbol{x},\tilde{y}=i} + \epsilon_i^{(2)}$ where $\epsilon_j^{(1)}, \epsilon_j^{(2)} \in \mathcal{R}$ and ϵ_j can be any distribution. This relaxes the ideal condition with noise relevant for neural networks, known to be class-conditionally overly confident (Guo et al., 2017).

Corollary 1.1 (Per-Class Robustness)**.** *For a noisy dataset* $\boldsymbol{X} \coloneqq (\boldsymbol{x}, \tilde{y})^n \in (\mathbb{R}^d, \mathbb{N}_{>0})^n$ *and model* $\boldsymbol{\theta}{:}\boldsymbol{x}\to\hat{p}(\tilde{y})$*, if* $\hat{p}_{\boldsymbol{x},\tilde{y}=i}$ *is* ***per-class diffracted*** *without label collisions and each diagonal entry of* $\boldsymbol{Q}_{\tilde{y}|y^*}$ *maximizes its row, then* $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$ *and* $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} \approxeq \boldsymbol{Q}_{\tilde{y},y^*}$.

Cor. 1.1 shows us that $\boldsymbol{C}_{\tilde{y},y^*}$ in confident learning is robust to any linear combination of per-class error in probabilities. Observe that $\boldsymbol{C}_{\text{confusion}}$ does not satisfy Cor. 1.1 because the theorem no longer requires that the diagonal of $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximize its column. By not using thresholds, $\boldsymbol{C}_{\text{confusion}}$ implicitly assumes similar distributions of probabilities for each class, whereas $\boldsymbol{C}_{\tilde{y},y^*}$ satisfies Cor. 1.1 using thresholds for robustness to distributional shift and class-imbalance.

Cor. 1.1 only allows for m alterations in the probabilities and there are only m^2 unique probabilities under the ideal condition, whereas in real-world conditions, an error-prone model could potentially output nm unique probabilities. Next, in Thm. 2, we examine a reasonable sufficient condition where CL is robust to erroneous probabilities for every example and class.

Condition (Per-Example Diffracted). $\hat{p}_{\boldsymbol{x},\tilde{y}=i}$ is *per-example diffracted* if $\forall j \in 1..m, \forall \boldsymbol{x} \in \boldsymbol{X}$, we have error as $\hat{p}_{\boldsymbol{x},\tilde{y}=j} = p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_{\boldsymbol{x},\tilde{y}=j}$ where $\epsilon_j = \mathbb{E}_{\boldsymbol{x}\in\boldsymbol{X}}\, \epsilon_{\boldsymbol{x},\tilde{y}=j}$ and

$$\epsilon_{\boldsymbol{x},\tilde{y}=j} \sim \begin{cases} \mathcal{U}(\epsilon_j + t_j - p^*_{\boldsymbol{x},\tilde{y}=j}\,,\ \epsilon_j - t_j + p^*_{\boldsymbol{x},\tilde{y}=j}] & p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j \\ \mathcal{U}[\epsilon_j - t_j + p^*_{\boldsymbol{x},\tilde{y}=j}\,,\ \epsilon_j + t_j - p^*_{\boldsymbol{x},\tilde{y}=j}) & p^*_{\boldsymbol{x},\tilde{y}=j} < t_j \end{cases} \tag{5}$$

where $\mathcal{U}$ denotes a uniform distribution (a more general case is discussed in the Appendix).

Theorem 2 (General Per-Example Robustness)**.** *For a noisy dataset* $\boldsymbol{X} \coloneqq (\boldsymbol{x}, \tilde{y})^n \in (\mathbb{R}^d, \mathbb{N}_{>0})^n$ *and model* $\boldsymbol{\theta}{:}\boldsymbol{x}\to\hat{p}(\tilde{y})$*, if* $\hat{p}_{\boldsymbol{x},\tilde{y}=i}$ *is* ***per-example diffracted*** *without label collisions and each diagonal entry of* $\boldsymbol{Q}_{\tilde{y}|y^*}$ *maximizes its row, then* $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$ *and* $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} \approxeq \boldsymbol{Q}_{\tilde{y},y^*}$.

In Thm. 2, we observe that if each example's predicted probability resides within the residual range of the ideal probability and the threshold, then CL exactly identifies label errors and consistently estimates $\boldsymbol{Q}_{\tilde{y},y^*}$. Intuitively, if $\hat{p}_{\boldsymbol{x},\tilde{y}=j} \geq t_j$ whenever $p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j$ and $\hat{p}_{\boldsymbol{x},\tilde{y}=j} < t_j$ whenever $p^*_{\boldsymbol{x},\tilde{y}=j} < t_j$, then regardless of error in $\hat{p}_{\boldsymbol{x},\tilde{y}=j}$, CL exactly finds label errors. As an example, consider an image $\boldsymbol{x}_k$ that is mislabeled as *fox*, but is actually a *dog* where $t_{fox} = 0.6$, $p^*(\tilde{y}{=}fox; \boldsymbol{x} \in \boldsymbol{X}_{y^*=dog}, \boldsymbol{\theta}) = 0.2$, $t_{dog} = 0.8$, and $p^*(\tilde{y}{=}dog; \boldsymbol{x} \in \boldsymbol{X}_{y^*=dog}, \boldsymbol{\theta}) = 0.9$. Then as long as $-0.4 \leq \epsilon_{\boldsymbol{x},fox} < 0.4$ and $-0.1 < \epsilon_{\boldsymbol{x},dog} \leq 0.1$, CL will surmise $y^*_k = dog$, not fox, even though $\tilde{y}_k = fox$ is given.

While $\boldsymbol{Q}_{\tilde{y},y^*}$ is a statistic to characterize aleatoric uncertainty from latent label noise, Thm. 2 addresses epistemic uncertainty in the case of erroneous predicted probabilities.

5. Experiments

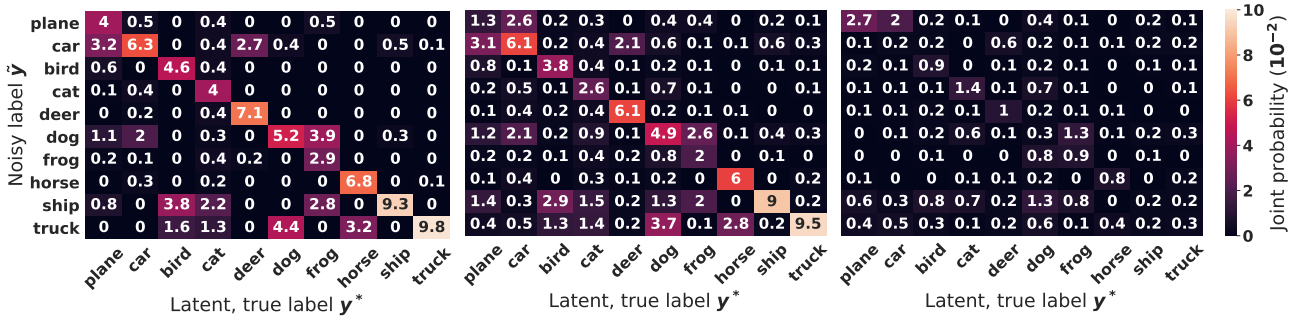
This section empirically validates CL on CIFAR (Krizhevsky & Hinton, 2009) and ImageNet (Russakovsky et al., 2015) benchmarks. MNIST is in the Appendix (see Sec F). Sec. 5.1 presents CL performance on noisy examples in CIFAR where true labels are known. Sec. 5.2 shows real-world noise identification using ImageNet, and the performance gain when training with CL. We compute out-of-sample predicted probabilities $\hat{\boldsymbol{P}}_{k,j}$ using four-fold cross validation with ResNet architectures.

5.1. Non-uniform Label Noise on CIFAR

We evaluate CL on three criteria: (a) joint estimation (Fig. 2), (b) accuracy finding label errors (Table 2), and (c) accuracy learning with noisy labels (Table 1).

Noise Generation. Following prior work (Sukhbaatar et al., 2015; Goldberger & Ben-Reuven, 2017), we study CL performance on non-uniform, asymmetric label noise for its resemblance to real-world noise. We generate noisy data from clean data by randomly switching some labels of training examples to different classes non-uniformly according to a randomly generated $\boldsymbol{Q}_{\tilde{y}|y^*}$ noise transition matrix. We generate $\boldsymbol{Q}_{\tilde{y}|y^*}$ matrices with different traces to run experiments for different noise levels. The noise matrices we use in our experiments are in the Appendix (see Fig. 7).

We generate noise in the CIFAR-10 training dataset across varying *sparsities*, the fraction of off-diagonals in $\boldsymbol{Q}_{\tilde{y},y^*}$



(a) True joint (unknown to CL) $\boldsymbol{Q}_{\tilde{y},y^*}$ (b) CL estimated joint $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$ (c) Absolute difference $|\boldsymbol{Q}_{\tilde{y},y^*} - \hat{\boldsymbol{Q}}_{\tilde{y},y^*}|$

Figure 2. Our estimation of the joint distribution of noisy labels and true labels for CIFAR with 40% label noise and 60% sparsity. Observe the similarity of values between (a) and (b) and the low absolute error in every entry in (c). Probabilities are scaled up by 100.

Table 1. Comparison of confident learning versus seven recent methods for learning with noisy labels in CIFAR-10. Highlighted cells show CL robustness to sparsity. The five CL methods estimate label errors, remove them, then train on the cleaned data using Co-Teaching.

Noise		0.2				0.4				0.7			
Sparsity	**AVG**	0	0.2	0.4	0.6	0	0.2	0.4	0.6	0	0.2	0.4	0.6
CL: $\boldsymbol{C}_{\text{CONFUSION}}$	0.674	0.875	0.877	0.886	0.882	0.830	0.825	0.764	0.711	0.376	0.370	0.350	0.337
CL: PBC	0.697	0.885	0.885	0.890	0.892	0.848	0.849	0.844	0.855	**0.386**	0.334	**0.364**	0.330
CL: C+NR	0.703	0.891	0.894	**0.897**	0.897	**0.868**	0.863	**0.861**	0.857	0.384	0.336	0.350	0.344
CL: PBNR	0.705	0.892	0.889	0.895	0.893	0.863	**0.864**	**0.861**	0.864	0.363	0.369	0.362	**0.349**
CL: $\boldsymbol{C}_{\tilde{y},y^*}$	**0.710**	**0.896**	**0.896**	**0.897**	**0.901**	0.859	0.861	**0.861**	**0.865**	0.380	**0.407**	0.349	0.344
INCV	0.661	0.878	0.886	0.896	0.892	0.844	0.766	0.854	0.736	0.283	0.253	0.348	0.297
MIXUP	0.622	0.856	0.868	0.870	0.843	0.761	0.754	0.686	0.598	0.322	0.313	0.323	0.269
SCE-LOSS	0.615	0.872	0.875	0.888	0.844	0.763	0.741	0.649	0.583	0.330	0.287	0.309	0.240
MENTORNET	0.590	0.849	0.851	0.832	0.834	0.644	0.642	0.624	0.615	0.300	0.316	0.293	0.279
CO-TEACHING	0.569	0.812	0.813	0.814	0.806	0.629	0.616	0.609	0.581	0.305	0.302	0.277	0.260
S-MODEL	0.556	0.800	0.800	0.797	0.791	0.586	0.612	0.591	0.575	0.284	0.285	0.279	0.273
REED	0.560	0.781	0.789	0.808	0.793	0.605	0.604	0.612	0.586	0.290	0.294	0.291	0.268
BASELINE	0.554	0.784	0.792	0.790	0.782	0.602	0.608	0.596	0.573	0.270	0.297	0.282	0.268

that are zero, and fractions of incorrect labels. We avoid symmetric label noise because its rarely occurs naturally. All models are evaluated on the unaltered test set.

In Table 1, we compare CL performance versus seven recent state-of-the-art approaches and a vanilla baseline for multiclass learning with noisy labels on CIFAR-10, including *INCV* (Chen et al., 2019) which finds clean data with multiple iterations of cross-validation then trains on the clean set, *SCE-loss* (symmetric cross entropy) (Wang et al., 2019) which adds a reverse cross entropy term for loss-correction, *Mixup* (Zhang et al., 2018) which linearly combines examples and labels to augment data, *MentorNet* (Jiang et al., 2018) which uses curriculum learning to avoid noisy data in training, *Co-Teaching* (Han et al., 2018) which trains two models in tandem to learn from clean data, *S-Model* (Goldberger & Ben-Reuven, 2017) which uses an extra softmax layer to model noise during training, and *Reed* (Reed et al., 2015) which uses loss-reweighting; and a *Baseline* model that denotes a vanilla training with the noisy labels.

All models, including ours, are trained using ResNet50 with settings: learning rate 0.1 for epoch [0,150), 0.01 for epoch [150,250), 0.001 for epoch [250,350); momentum 0.9; and weight decay 0.0001, except *INCV*, *SCE-loss*, and *Co-Teaching* which are trained using their official GitHub code. We report the highest score across hyper-parameters $\alpha \in \{1, 2, 4, 8\}$ for *Mixup* and $p \in \{0.7, 0.8, 0.9\}$ for *MentorNet*. For fair comparison with *INCV* which trains with *Co-Teaching*, we also train with *Co-Teaching*. Training information and an extensive comparison of *INCV* and CL is in the Appendix (see Sec. E). Exactly the same noisy labels are used for training all models for each column of Table 1.

Robustness to Sparsity and State-of-the-Art. Table 1 reports CIFAR test accuracy for learning with noisy labels across noise amount and sparsity, where the first five rows report our CL approaches. As shown, CL consistently outperforms other prior art across all noise and sparsity settings, exhibiting over 30% improvement over competitive approaches like MentorNet. We observe significant improvement in high-noise regimes and moderate improvement performance in low-noise regimes. Unlike the other methods, CL does not decrease in accuracy for realistic, high sparsity noise. The simplest CL method $CL: \boldsymbol{C}_{\text{confusion}}$

Table 2. Accuracy, F1, precision, and recall measures for finding label errors in CIFAR-10.

Measure	Accuracy				F1				Precision				Recall			
Noise	0.2		0.4		0.2		0.4		0.2		0.4		0.2		0.4	
Sparsity	0.0	0.6	0.0	0.6	0.0	0.6	0.0	0.6	0.0	0.6	0.0	0.6	0.0	0.6	0.0	0.6
CL: $C_{\text{CONFUSION}}$	0.84	0.85	0.85	0.81	0.71	0.72	0.84	0.79	0.56	0.58	0.74	0.70	**0.98**	**0.97**	**0.97**	**0.90**
CL: $C_{\tilde{y},y^*}$	0.89	**0.90**	0.86	**0.84**	0.75	0.78	0.84	**0.80**	**0.67**	**0.70**	0.78	0.77	0.86	0.88	0.91	0.84
CL: PBC	0.88	0.88	0.86	0.82	0.76	0.76	0.84	0.79	0.64	0.65	0.76	0.74	0.96	0.93	0.94	0.85
CL: PBNR	0.89	**0.90**	**0.88**	**0.84**	0.77	**0.79**	**0.85**	**0.80**	0.65	0.68	**0.82**	**0.79**	0.93	0.94	0.88	0.82
CL: C+NR	**0.90**	**0.90**	0.87	0.83	**0.78**	0.78	0.84	0.78	**0.67**	0.69	**0.82**	**0.79**	0.93	0.90	0.87	0.78

performs similarly to *INCV* and comparably to prior art with best performance by $\boldsymbol{C}_{\tilde{y},y^*}$ across all noise and sparsity settings. The results validate the benefit of directly modeling the joint noise distribution and show our method is competitive compared with state-of-the-art robust learning methods.

To understand why CL performs well, we evaluate CL joint estimation across noise and sparsity with RMSE in Table 5 in the Appendix and estimated $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$ in Fig. 5 in the Appendix. For the 20% and 40% noise settings, on average, CL achieves an RMSE of .004 relative to the true joint $\boldsymbol{Q}_{\tilde{y},y^*}$ across all sparsities. The simplest CL variant, $\boldsymbol{C}_{\text{confusion}}$ normalized via Eqn. (4) to obtain $\hat{\boldsymbol{Q}}_{confusion}$, achieves a slightly worse RMSE of .006.

In Fig. 2, we visualize the quality of CL joint estimation in a challenging high-noise (40%), high-sparsity (60%) regime on CIFAR. Sub-figure (a) demonstrates high sparsity in the latent true joint $\boldsymbol{Q}_{\tilde{y},y^*}$, with over half the noise in just six noise rates. Yet, as can be seen in sub-figures (b) and (c), CL still estimates over 80% of the entries of $\boldsymbol{Q}_{\tilde{y},y^*}$ within an absolute difference of .005. The results empirically substantiate the theoretical bounds of Section 4.

We also evaluate CL's accuracy in finding label errors. In Table 2, we compare five variants of CL methods across noise and sparsity and report their precision, recall, and F1 in recovering the true label. The results show that CL is able to find the label errors with high recall and reasonable F1.

5.2. Real-world Noise with ImageNet

Russakovsky et al. (2015) suggest label errors exist in ImageNet due to human error, but to our knowledge, no attempt has been made to find label errors in the ILSVRC 2012 training set, characterize them, or re-train without them. Here, we consider each application (see Appendix Sec. F for MNIST analogues). We use ResNet18 and ResNet50 architectures with standard settings: 0.1 initial learning rate, 90 training epochs with 0.9 momentum.

Ontological discovery via label noise characterization. Because ImageNet is a single-class dataset, classes are required to be mutually exclusive. We observe auto-discovery of ontological issues in datasets in Table 3 by listing the 10 largest non-diagonal entries in $\boldsymbol{C}_{\tilde{y},y^*}$. For example, the class *maillot* appears twice, the existence of *is-a* relationships like *bathtub is a tub*, misnomers like *projectile* and *missile*, and unanticipated issues caused by words with multiple definitions like *corn* and *ear*. We include $\boldsymbol{C}_{\text{confusion}}$ to show that while it counts fewer, it still ranks similarly.

Table 3. Ten largest non-diagonal entries in the confident joint $\boldsymbol{C}_{\tilde{y},y^*}$ for ImageNet train set used for ontological issue discovery.

$\boldsymbol{C}_{\tilde{y},y^*}$	$\tilde{y}$ name	y^* name	$\boldsymbol{C}_{\text{confusion}}$	$\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$
645	projectile	missile	494	0.00050
539	tub	bathtub	400	0.00042
476	breastplate	cuirass	398	0.00037
437	green_lizard	chameleon	369	0.00034
435	chameleon	green_lizard	362	0.00034
433	missile	projectile	362	0.00034
417	maillot	maillot	338	0.00033
416	horned_viper	sidewinder	336	0.00033
410	corn	ear	333	0.00032
407	keyboard	space_bar	293	0.00032

Finding label issues. Fig. 3 depicts the top 16 label issues found using **CL: PBNR** with ResNet50 ordered by the normalized margin. We use the term *issue* versus *error* because examples found by CL consist of a mixture of multi-label images, ontological issues, and actual label errors. Examples of each are indicated by colored borders in the figure. To evaluate CL in the absence of true labels, we conducted a small-scale human validation on a random sample of 500 **CL: PBNR** errors and found 58% were either multi-labeled, ontological issues, or errors. ImageNet data are often presumed clean yet ours is the first attempt to identify label errors automatically in ImageNet training images.

Training ResNet on ImageNet with label issues removed. To understand the performance differences, we train both ResNet50 (Fig. 4b) and ResNet18 (Fig. 4a) by progressively removing identified *noisy examples* in the ImageNet training set. Fig. 4 shows the top-1 accuracy on the ILSVRC validation set when removing label errors estimated by CL methods versus removing random examples. We do not compare with other baselines because they may not identify label errors. For each CL method, we plot the accuracy of training with 20%, 40%,..., 100% of estimated label errors removed, omitting points beyond 200k.

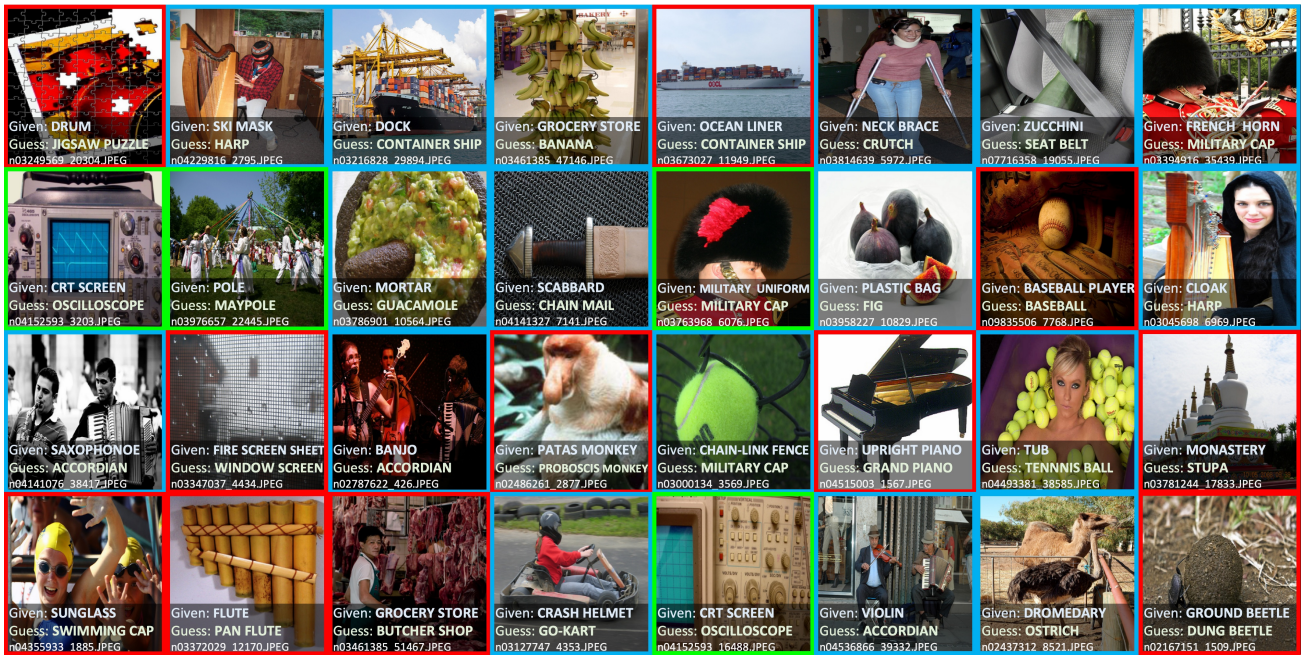


Figure 3. Top 32 identified label issues in the 2012 ILSVRC ImageNet train set using CL: PBNR. Errors are boxed in red. Ontological issues are boxed in green. Multi-label images are boxed in blue.

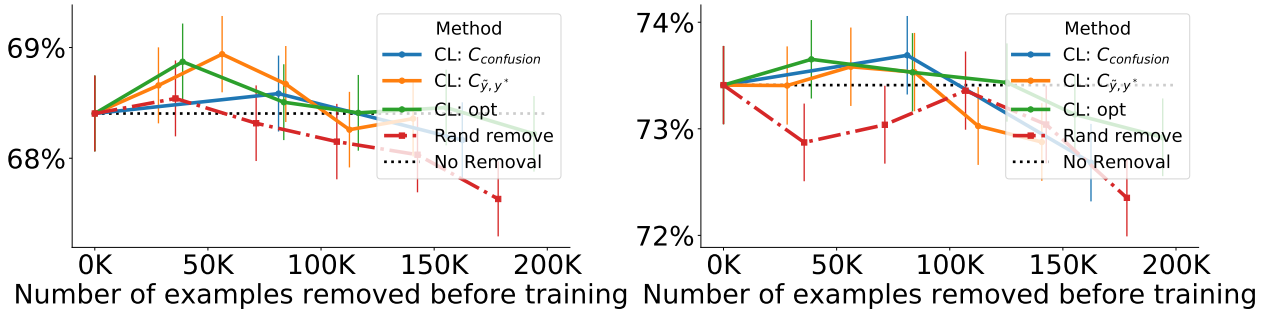
We find CL methods may even improve the standard ImageNet training on clean training data by filtering out a subset of training examples. The result is significant as ImageNet training images are often assumed to have correct labels. These results suggest CL is able to identify the label noise in the real-world dataset and improve the training over unnoticed label errors in the ImageNet train set. Once more than 100K many examples are removed, CL may not improve the standard training. However, CL methods still significantly outperform the random removal baseline. We provide additional comparison of CL: PBNR versus random pruning in the Appendix in Figures 9 and 8.

6. Related work

We first discuss prior work on confident learning, then review how CL relates to noise estimation and robust learning.

Confident learning. Our results build on a large body of work termed "confident learning". Elkan (2001) and Forman (2005) pioneered counting approaches to estimate false positive and false negative rates for binary classification. We extend counting principles to multi-class setting. To increase robustness against epistemic error in predicted probabilities and class imbalance, Elkan & Noto (2008) introduced thresholding, but required uncorrupted positive labels. CL generalizes the use of thresholds to multi-class noisy labels. CL also re-weights the loss during training to adjust priors for the data removed. This choice builds on formative works (Natarajan et al., 2013; Van Rooyen et al., 2015) which used loss re-weighting to prove equivalent empirical risk minimization for learning with noisy labels. More recently, Han et al. (2019) proposed an empirical deep self-supervised learning approach to avoid probabilities by using embedding layers of a neural network. In comparison, CL is non-iterative and theoretically grounded. Like CL, Lipton et al. (2018) and Chen et al. (2019) estimate label noise using approaches based on confusion matrices and cross-validation. However, unlike CL the former assumes a specific form of label shift and the latter, *INCV*, uses an iterative pruning approach comparable to the $\boldsymbol{C}_{\text{confusion}}$ baseline. An extensive comparison of *INCV* and CL is available in the Appendix (see Sec. E).

Label noise estimation. A number of formative works developed solutions to estimate noise rates using convergence criterion (Scott, 2015), positive-unlabeled learning (Elkan & Noto, 2008), and predicted probability ratios (Northcutt et al., 2017), but are limited to binary classification. Others prove equivalent empirical risk for *binary* learning with noisy labels (Natarajan et al., 2013; Liu & Tao, 2015; Sugiyama et al., 2012) assuming noise rates are known which is rarely true in practice. Unlike these binary approaches, CL estimates label uncertainty in the multiclass setting, where prior work often falls into five categories: (1) theoretical contributions (Katz-Samuels et al., 2019), (2) loss modification for label noise robustness (Patrini et al., 2016; 2017; Sukhbaatar et al., 2015; Van Rooyen et al., 2015), (3) deep learning and model-specific approaches (Sukhbaatar et al., 2015; Patrini et al.,



(a) ResNet18 Validation Accuracy

(b) ResNet50 Validation Accuracy

Figure 4. Increased ResNet validation accuracy using CL methods on ImageNet with original labels (no synthetic noise added). Each point on the line for each method, from left to right, depicts the accuracy of training with 20%, 40%..., 100% of estimated label errors removed. Error bars are computed with Clopper-Pearson 95% confidence intervals. The red dash-dotted baseline captures when examples are removed uniformly randomly. The black dotted line depicts accuracy when training with all examples.

2016; Jindal et al., 2016), (4) crowd-sourced labels via multiple workers (Zhang et al., 2017b; Dawid & Skene, 1979; Ratner et al., 2016), (5) factorization, distillation (Li et al., 2017), and imputation (Amjad et al., 2018) methods, among other (Sáez et al., 2014). Unlike these approaches, CL provides a consistent estimator to estimate the joint distribution of noisy and true labels directly.

Noise-robust learning. Beyond the above noise estimation approaches, extensive studies have investigated training models on noisy datasets, e.g. (Beigman & Klebanov, 2009; Brodley & Friedl, 1999). Noise-robust learning is important for deep learning because modern neural networks trained on noisy labels generalize poorly on clean validation data (Zhang et al., 2017a). A notable recent rend in noise robust learning is benchmarking with symmetric label noise in which labels are uniformly flipped, e.g. (Goldberger & Ben-Reuven, 2017; Arazo et al., 2019). However, noise in real-world datasets is highly non-uniform and often sparse. For example in ImageNet (Russakovsky et al., 2015), *missile* is likely to be mislabeled as *projectile*, but has a near-zero probability of being mislabeled as most other classes like *wool*, *ox*, or *wine*. To approximate real-world noise, an increasing number of studies examined asymmetric noise using, e.g. loss or label correction (Patrini et al., 2017; Reed et al., 2015; Goldberger & Ben-Reuven, 2017), per-example loss reweighting (Jiang et al., 2018; Shu et al., 2019), Co-Teaching (Han et al., 2018), semi-supervised learning (Hendrycks et al., 2018; Li et al., 2017; Vahdat, 2017), and symmetric cross entropy (Wang et al., 2019), among others. These approaches work by introducing novel new models or insightful modifications to the loss function during training. CL takes a different approach, instead focusing on generating clean data for training by directly estimating of the joint distribution of noisy and true labels.

7. Conclusion

These findings emphasize the practical nature of confident learning, identifying numerous label issues in ImageNet and CIFAR, and improving standard ResNet performance by training on a cleaned dataset. Confident learning motivates the need for further understanding of dataset uncertainty estimation, methods to clean training and test sets, and approaches to identify ontological and label issues in datasets.

Acknowledgements

We thank the following colleagues: Jonas Mueller assisted with notation. Anish Athayle suggested starting the proof in claim 1 of Theorem 1 with the identity. Tailin Wu contributed to Lemma 1. Niranjan Subrahmanya provided feedback on baselines for confident learning.

References

Amjad, M., Shah, D., and Shen, D. Robust synthetic control. *Journal of Machine Learning Research (JMLR)*, 19(1): 802–852, 2018.

Angluin, D. and Laird, P. Learning from noisy examples. *Machine Learning*, 2(4):343–370, 1988.

Arazo, E., Ortego, D., Albert, P., O'Connor, N. E., and McGuinness, K. Unsupervised label noise modeling and loss correction. In *International Conference on Machine Learning (ICML)*, 2019.

Beigman, E. and Klebanov, B. B. Learning with annotation noise. In *Annual Conference of the Association for Computational Linguistics (ACL)*, 2009.

Brodley, C. E. and Friedl, M. A. Identifying mislabeled training data. *Journal of Artificial Intelligence Research (JAIR)*, 11:131–167, 1999.

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 (ICML)*, 2019.

Dawid, A. P. and Skene, A. M. Maximum likelihood estimation of observer error-rates using the em algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1):20–28, 1979.

Elkan, C. The foundations of cost-sensitive learning. In *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 973–978, 2001. ISBN 1558608125.

Elkan, C. and Noto, K. Learning classifiers from only positive and unlabeled data. In *SIGKDD Conference on Knowledge Discovery and Data Mining (KDD)*, 2008.

Forman, G. Counting positives accurately despite inaccurate classification. In *European Conference on Computer Vision (ECCV)*, 2005.

Forman, G. Quantifying counts and costs via classification. *Data Mining and Knowledge Discovery*, 17(2):164–206, 2008.

Goldberger, J. and Ben-Reuven, E. Training deep neural-networks using a noise adaptation layer. In *International Conference on Learning Representations (ICLR)*, 2017.

Graepel, T. and Herbrich, R. The kernel gibbs sampler. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2001.

Guo, C., Pleiss, G., Sun, Y., and Weinberger, K. Q. On calibration of modern neural networks. In *International Conference on Machine Learning (ICML)*, 2017.

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 *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

Han, J., Luo, P., and Wang, X. Deep self-learning from noisy labels. In *International Conference on Computer Vision (ICCV)*, 2019.

Hendrycks, D., Mazeika, M., Wilson, D., and Gimpel, K. Using trusted data to train deep networks on labels corrupted by severe noise. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2018.

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 (ICML)*, 2018.

Jindal, I., Nokleby, M., and Chen, X. Learning deep networks from noisy labels with dropout regularization. In *International Conference on Data Mining (ICDM)*, 2016.

Katz-Samuels, J., Blanchard, G., and Scott, C. Decontamination of mutual contamination models. *Journal of Machine Learning Research (JMLR)*, 20(41):1–57, 2019.

Krizhevsky, A. and Hinton, G. Learning multiple layers of features from tiny images. *Master's thesis, Department of Computer Science, University of Toronto*, 2009.

Lawrence, N. D. and Schölkopf, B. Estimating a kernel fisher discriminant in the presence of label noise. In *International Conference on Machine Learning (ICML)*, 2001.

Li, Y., Yang, J., Song, Y., Cao, L., Luo, J., and Li, L.-J. Learning from noisy labels with distillation. In *International Conference on Computer Vision (ICCV)*, 2017.

Lipton, Z., Wang, Y.-X., and Smola, A. Detecting and correcting for label shift with black box predictors. In *International Conference on Machine Learning (ICML)*, 2018.

Liu, T. and Tao, D. Classification with noisy labels by importance reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 38(3):447–461, 2015.

Natarajan, N., Dhillon, I. S., Ravikumar, P. K., and Tewari, A. Learning with noisy labels. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2013.

Natarajan, N., Dhillon, I. S., Ravikumar, P., and Tewari, A. Cost-sensitive learning with noisy labels. *Journal of Machine Learning Research (JMLR)*, 18:155–1, 2017.

Northcutt, C. G., Wu, T., and Chuang, I. L. Learning with confident examples: Rank pruning for robust classification with noisy labels. In *Conference on Uncertainty in Artificial Intelligence (UAI)*, 2017.

Page, L., Brin, S., Motwani, R., and Winograd, T. Pagerank: Bringing order to the web. Technical report, Stanford Digital Libraries Working Paper, 1997.

Patrini, G., Nielsen, F., Nock, R., and Carioni, M. Loss factorization, weakly supervised learning and label noise robustness. In *International Conference on Machine Learning (ICML)*, pp. 708–717, 2016.

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 *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Ratner, A. J., De Sa, C. M., Wu, S., Selsam, D., and Ré, C. Data programming: Creating large training sets, quickly. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2016.

Reed, S. E., Lee, H., Anguelov, D., Szegedy, C., Erhan, D., and Rabinovich, A. Training deep neural networks on noisy labels with bootstrapping. In *International Conference on Learning Representations (ICLR)*, 2015.

Richard, M. D. and Lippmann, R. P. Neural network classifiers estimate bayesian a posteriori probabilities. *Neural computation*, 3(4):461–483, 1991.

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C., and Fei-Fei, L. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

Sáez, J. A., Galar, M., Luengo, J., and Herrera, F. Analyzing the presence of noise in multi-class problems: alleviating its influence with the one-vs-one decomposition. *Knowledge and Information Systems*, 38(1):179–206, 2014.

Scott, C. A rate of convergence for mixture proportion estimation, with application to learning from noisy labels. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2015.

Shu, J., Xie, Q., Yi, L., Zhao, Q., Zhou, S., Xu, Z., and Meng, D. Meta-weight-net: Learning an explicit mapping for sample weighting. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

Sugiyama, M., Suzuki, T., and Kanamori, T. *Density Ratio Estimation in ML*. Cambridge University Press, New York, NY, USA, 1st edition, 2012.

Sukhbaatar, S., Bruna, J., Paluri, M., Bourdev, L., and Fergus, R. Training convolutional networks with noisy labels. In *International Conference on Learning Representations (ICLR)*, 2015.

Vahdat, A. Toward robustness against label noise in training deep discriminative neural networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2017.

Van Rooyen, B., Menon, A., and Williamson, R. C. Learning with symmetric label noise: The importance of being unhinged. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2015.

Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., and Bailey, J. Symmetric cross entropy for robust learning with noisy labels. In *International Conference on Computer Vision (ICCV)*, 2019.

Wei, C., Lee, J. D., Liu, Q., and Ma, T. On the margin theory of feedforward neural networks. *arXiv preprint arXiv:1810.05369*, 2018.

Zhang, C., Bengio, S., Hardt, M., Recht, B., and Vinyals, O. Understanding deep learning requires rethinking generalization. In *International Conference on Learning Representations (ICLR)*, 2017a.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations (ICLR)*, 2018.

Zhang, J., Sheng, V. S., Li, T., and Wu, X. Improving crowdsourced label quality using noise correction. *IEEE Transactions on Neural Networks and Learning Systems*, 29(5):1675–1688, 2017b.

A. Notation

In this section of the Appendix, we summarize the notation used in confident learning in tabular form.

Table 4. Notation used in confident learning.

Notation	Defintion
m	The number of unique class labels
$\{1..m\}$	The set of m unique class labels
$\tilde{y}$	Discrete random variable $\tilde{y} \in \mathbb{N}_{>0}$ takes an observed, noisy label
y^*	Discrete random variable $y^* \in \mathbb{N}_{>0}$ takes the unknown, true, uncorrupted label
$\tilde{y}_k$	The observed, noisy label corresponding to $\boldsymbol{x}_k$
y_k^*	The unobserved, true label example of $\boldsymbol{x}_k$
$\boldsymbol{X}$	The dataset $(\boldsymbol{x}, \tilde{y})^n \in (\mathbb{R}^d, \mathbb{N}_{>0})^n$ of n examples $\boldsymbol{x} \in \mathbb{R}^d$ with noisy labels
n	The cardinality of $\boldsymbol{X} \coloneqq (\boldsymbol{x}, \tilde{y})^n$, i.e. the number of examples in the dataset
$\boldsymbol{\theta}$	Model parameters
$\boldsymbol{X}_{\tilde{y}=i}$	Subset of examples in $\boldsymbol{X}$ with noisy label i, *i.e.* $\boldsymbol{X}_{\tilde{y}=\text{cat}}$ is "examples labeled cat"
$\boldsymbol{X}_{\tilde{y}=i,y^*=j}$	Subset of examples in $\boldsymbol{X}$ with noisy label i and true label j
$\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$	Estimate of subset of examples in $\boldsymbol{X}$ with noisy label i and true label j
$p(\tilde{y}=i, y^*=j)$	Discrete joint probability of noisy label i and true label j.
$p(\tilde{y}=i\|y^*=j)$	Discrete conditional probability of true label flipping, called the noise rate
$p(y^*=j\|\tilde{y}=i)$	Discrete conditional probability of noisy label flipping, called the inverse noise rate
$\hat{p}(\cdot)$	Estimated or predicted probability (may replace $p(\cdot)$ in any context)
$\boldsymbol{Q}_{y^*}$	The prior of the latent labels
$\hat{\boldsymbol{Q}}_{y^*}$	Estimate of the prior of the latent labels
$\boldsymbol{Q}_{\tilde{y},y^*}$	The $m \times m$ joint distribution matrix for $p(\tilde{y}, y^*)$
$\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$	Estimate of the $m \times m$ joint distribution matrix for $p(\tilde{y}, y^*)$
$\boldsymbol{Q}_{\tilde{y}\|y^*}$	The $m \times m$ noise transition matrix (noisy channel) of flipping rates for $p(\tilde{y}\|y^*)$
$\hat{\boldsymbol{Q}}_{\tilde{y}\|y^*}$	Estimate of the $m \times m$ noise transition matrix of flipping rates for $p(\tilde{y}\|y^*)$
$\boldsymbol{Q}_{y^*\|\tilde{y}}$	The inverse noise matrix for $p(y^*\|\tilde{y})$
$\hat{\boldsymbol{Q}}_{y^*\|\tilde{y}}$	Estimate of the inverse noise matrix for $p(y^*\|\tilde{y})$
$\hat{p}(\tilde{y}=i; \boldsymbol{x}, \boldsymbol{\theta})$	Predicted probability of label $\tilde{y}=i$ for example $\boldsymbol{x}$ and model parameters $\boldsymbol{\theta}$
$\hat{p}_{\boldsymbol{x},\tilde{y}=i}$	Shorthand abbreviation for predicted probability $\hat{p}(\tilde{y}=i; \boldsymbol{x}, \boldsymbol{\theta})$
$\hat{p}(\tilde{y}=i; \boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i}, \boldsymbol{\theta})$	The *self-confidence* of example $\boldsymbol{x}$ belonging to its given label $\tilde{y}=i$
$\hat{\boldsymbol{P}}_{k,i}$	$n \times m$ matrix of out-of-sample predicted probabilities $\hat{p}(\tilde{y}=j; \boldsymbol{x}_k, \boldsymbol{\theta})$
$\boldsymbol{C}_{\tilde{y},y^*}$	The *confident joint* $\boldsymbol{C}_{\tilde{y},y^*} \in \mathbb{N}_{\geq 0}^{m \times m}$, an unnormalized estimate of $\boldsymbol{Q}_{\tilde{y},y^*}$
$\boldsymbol{C}_{\text{confusion}}$	Confusion matrix of given labels $\tilde{y}_k$ and predictions $\arg\max_{i \in 1..m} \hat{p}(\tilde{y}=i; \boldsymbol{x}_k, \boldsymbol{\theta})$
t_j	The expected (average) self-confidence for class j used as a threshold in $\boldsymbol{C}_{\tilde{y},y^*}$
$p^*(\tilde{y}=i\|y^*=y_k^*)$	*Ideal* probability for some example $\boldsymbol{x}_k$, equivalent to noise rate $p^*(\tilde{y}=i\|y^*=j)$
$p^*_{\boldsymbol{x},\tilde{y}=i}$	Shorthand abbreviation for ideal probability $p^*(\tilde{y}=i\|y^*=y_k^*)$

B. Theorems and proofs for confident learning

In this section, we restate the main theorems for confident learning and provide their proofs.

Lemma 1 (Ideal Thresholds)**.** *For a noisy dataset $\boldsymbol{X}$ of $(\boldsymbol{x}, \tilde{y})$ pairs and model $\boldsymbol{\theta}$, if $\hat{p}(\tilde{y}; \boldsymbol{x}, \boldsymbol{\theta})$ is* ideal*, then $\forall i \in 1..m, t_i = \sum_{j \in 1..m} p(\tilde{y}=i|y^*=j)p(y^*=j|\tilde{y}=i)$.*

Proof. We use t_i to denote the thresholds used to partition $\boldsymbol{X}$ into m bins, each estimating one of $\boldsymbol{X}_{y^*}$. By definition,

$$\forall i \in 1..m, t_i = \mathbb{E}_{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i}} \hat{p}(\tilde{y}=i; \boldsymbol{x}, \boldsymbol{\theta})$$

For any t_i, we show the following.

$$
\begin{aligned}
t_i &= \mathop{\mathbb{E}}_{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i}} \sum_{j \in 1..m} \hat{p}(\tilde{y}{=}i|y^*{=}j; \boldsymbol{x}, \boldsymbol{\theta}) \hat{p}(y^*{=}j; \boldsymbol{x}, \boldsymbol{\theta}) && \triangleright \text{Bayes Rule} \\
t_i &= \mathop{\mathbb{E}}_{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i}} \sum_{j \in 1..m} \hat{p}(\tilde{y}{=}i|y^*{=}j) \hat{p}(y^*{=}j; \boldsymbol{x}, \boldsymbol{\theta}) && \triangleright \text{CNP assumption} \\
t_i &= \sum_{j \in 1..m} \hat{p}(\tilde{y}{=}i|y^*{=}j) \mathop{\mathbb{E}}_{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i}} \hat{p}(y^*{=}j; \boldsymbol{x}, \boldsymbol{\theta}) \\
t_i &= \sum_{j \in 1..m} p(\tilde{y} = i|y^* = j) p(y^* = j|\tilde{y} = i) && \triangleright \text{Ideal Condition}
\end{aligned}
$$

This form of the threshold is intuitively reasonable: the contributions to the sum when $i = j$ represents the probabilities of correct labeling, whereas when $i \neq j$, the terms give the probabilities of mislabeling $p(\tilde{y} = i|y^* = j)$, weighted by the probability $p(y^* = j|\tilde{y} = i)$ that the mislabeling is corrected. □

Theorem 1 (Exact Label Errors)**.** *For a noisy dataset $\boldsymbol{X}$ of $(\boldsymbol{x}, \tilde{y})$ pairs and model $\boldsymbol{\theta}$:$\boldsymbol{x} \to \hat{p}(\tilde{y})$, if $\hat{p}(\tilde{y}; \boldsymbol{x}, \boldsymbol{\theta})$ is ideal and each diagonal entry of $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximizes its row and column, then $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$ and $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} = \boldsymbol{Q}_{\tilde{y},y^*}$.*

Proof. Alg. 1 defines the construction of the confident joint. We consider case 1: when there are collisions (trivial by construction of Alg. 1) and case 2: when there are no collisions (harder).

Case 1 (collisions):
When a collision occurs, by construction of the confident joint (Eqn. 3), a given example $\boldsymbol{x}_k$ gets assigned bijectively into bin

$$\boldsymbol{x}_k \in \hat{\boldsymbol{X}}_{\tilde{y},y^*}[\tilde{y}_k][\arg\max_{i \in 1..m} \hat{p}(\tilde{y} = i; \boldsymbol{x}, \boldsymbol{\theta})]$$

Because we have that $\hat{p}(\tilde{y}; \boldsymbol{x}, \boldsymbol{\theta})$ is ideal, we can rewrite this as

$$\boldsymbol{x}_k \in \hat{\boldsymbol{X}}_{\tilde{y},y^*}[\tilde{y}_k][\arg\max_{i \in 1..m} \hat{p}(\tilde{y} = i|y^*{=}y_k^*; \boldsymbol{x})]$$

And because by assumption each diagonal entry in $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximizes its column, we have

$$\boldsymbol{x}_k \in \hat{\boldsymbol{X}}_{\tilde{y},y^*}[\tilde{y}_k][y_k^*]$$

So any example $\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i,y^*=j}$ having a collision will be exactly assigned to $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$.

Case 2 (no collisions):

We want to show that $\forall i \in 1..m, j \in 1..m, \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$

We can partition $\boldsymbol{X}_{\tilde{y}=i}$ as

$$\boldsymbol{X}_{\tilde{y}=i} = \boldsymbol{X}_{\tilde{y}=i,y^*=j} \cup \boldsymbol{X}_{\tilde{y}=i,y^* \neq j}$$

We prove $\forall i \in 1..m, j \in 1..m, \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$ by proving two claims:

Claim 1: $\boldsymbol{X}_{\tilde{y}=i,y^*=j} \subseteq \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$

Claim 2: $\boldsymbol{X}_{\tilde{y}=i,y^* \neq j} \nsubseteq \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$

We don't need to show $\boldsymbol{X}_{\tilde{y} \neq i,y^*=j} \nsubseteq \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$ and $\boldsymbol{X}_{\tilde{y} \neq i,y^* \neq j} \nsubseteq \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$ because the noisy labels $\tilde{y}$ are given, thus the confident joint (Eqn. 3) will never place them in the wrong bin of $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$. Thus, claim 1 and claim 2 are sufficient to show that $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$.

***Proof (Claim 1) of Case 2*:** Inspecting Eqn (3) and Alg (1), by the construction of $\boldsymbol{C}_{\tilde{y},y^*}$, we have that $\forall \boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i}$, $\hat{p}(\tilde{y}=j|y^*{=}j;\boldsymbol{x},\boldsymbol{\theta}) \geq t_j \longrightarrow \boldsymbol{X}_{\tilde{y}=i,y^*=j} \subseteq \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$. In other words, when the left hand side is true, all examples with noisy label i and hidden, true label j are counted in $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$.

Thus, it is sufficient to prove

$$\forall \boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i}, \hat{p}(\tilde{y}=j|y^*{=}j;\boldsymbol{x},\boldsymbol{\theta}) \geq t_j \tag{6}$$

Because predicted probabilities satisfy the ideal condition, $\hat{p}(\tilde{y}=j|y^*{=}j,\boldsymbol{x}) = p(\tilde{y}=j|y^*{=}j), \forall \boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i}$. Note the change from predicted probability, $\hat{p}$, to an exact probability, p. Thus by the ideal condition, the inequality in (6) can be written as $p(\tilde{y}=j|y^*{=}j) \geq t_j$, which we prove below:

$$\begin{aligned}
p(\tilde{y}=j|y^*{=}j) &\geq p(\tilde{y}=j|y^*{=}j)\cdot 1 && \triangleright \text{Identity} \\
&\geq p(\tilde{y}=j|y^*{=}j)\cdot \sum_{i\in 1..m} p(y^*{=}i|\tilde{y}{=}j) && \\
&\geq \sum_{i\in 1..m} p(\tilde{y}=j|y^*{=}j)\cdot p(y^*{=}i|\tilde{y}{=}j) && \triangleright \text{move product into sum} \\
&\geq \sum_{i\in 1..m} p(\tilde{y}=j|y^*{=}i)\cdot p(y^*{=}i|\tilde{y}{=}j) && \triangleright \text{diagonal entry maximizes row} \\
&\geq t_j && \triangleright \text{Lemma 1, ideal condition}
\end{aligned}$$

***Proof (Claim 2) of Case 2*:** We prove $\boldsymbol{X}_{\tilde{y}=i,y^*\neq j} \nsubseteq \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$ by contradiction. Assume there exists some example $\boldsymbol{x}_k \in \boldsymbol{X}_{\tilde{y}=i,y^*=z}$ for $z \neq j$ such that $\boldsymbol{x}_k \in \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$. By claim 1, we have that $\boldsymbol{X}_{\tilde{y}=i,y^*=j} \subseteq \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$, therefore, $\boldsymbol{x}_k \in \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=z}$.

So, for some example $\boldsymbol{x}_k$, we have that $\boldsymbol{x}_k \in \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j}$ and also $\boldsymbol{x}_k \in \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=z}$.

But this is a collision and when a collision occurs, the confident joint will break the tie with $\arg\max$. Because each diagonal entry of $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximizes its row and column this will always be assign $\boldsymbol{x}_k \in \hat{\boldsymbol{X}}_{\tilde{y},y^*}[\tilde{y}_k][y_k^*]$ (the assignment from Claim 1).

This theorem also states $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} \approxeq \boldsymbol{Q}_{\tilde{y},y^*}$. This directly follows directly from the fact that $\forall i{\in}1..m, j{\in}1..m, \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$, i.e. the confident joint *exactly counts* the partitions $\boldsymbol{X}_{\tilde{y}=i,y^*=j}$ for all pairs $(i,j) \in 1..m \times M$, thus $\boldsymbol{C}_{\tilde{y},y^*} = n\boldsymbol{Q}_{\tilde{y},y^*}$ and $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} \approxeq \boldsymbol{Q}_{\tilde{y},y^*}$. The confident joint is a consistent estimator for $\boldsymbol{Q}_{\tilde{y},y^*}$. Equivalency is exact regardless of number of examples as long as the noise rates can be represented as fractions of the dataset size. For example, if a noise rate is 0.39, but the dataset has only 5 examples in that class, the best possible estimate by removing errors is $2/5 = 0.4 \approxeq 0.39$.

□

Corollary 1.0 (Exact Estimation)**.** *For a noisy dataset $(\boldsymbol{x},\tilde{y})^n \in (\mathbb{R}^d, \mathbb{N}_{>0})^n$ and $\boldsymbol{\theta}{:}\boldsymbol{x}{\to}\hat{p}(\tilde{y})$, if $\hat{p}(\tilde{y};\boldsymbol{x},\boldsymbol{\theta})$ is ideal and each diagonal entry of $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximizes its row and column, and if $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$, then $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} \approxeq \boldsymbol{Q}_{\tilde{y},y^*}$.*

Proof. The result follows directly from Thm. 1. Because the confident joint *exactly counts* the partitions $\boldsymbol{X}_{\tilde{y}=i,y^*=j}$ for all pairs $(i,j) \in 1..m \times M$ by Thm. 1, $\boldsymbol{C}_{\tilde{y},y^*} = n\boldsymbol{Q}_{\tilde{y},y^*}$, omitting discretization rounding errors. We name this corollary *consistent* estimation instead of *exact* estimation because the equivalency only holds *exactly* for infinite examples due to discretization rounding errors. □

In the main text, Theorem 1 includes Corollary 1.0 for brevity. We have separated out Corollary 1.0 here to make apparent that the primary contribution of Thm. 1 is to prove $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$, from which the result of Corollary 1.0, namely that as $n \to \infty$, $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} \approxeq \boldsymbol{Q}_{\tilde{y},y^*}$, naturally follows.

Corollary 1.1 (Per-class Robustness)**.** *For a noisy dataset $\boldsymbol{X} \coloneqq (\boldsymbol{x},\tilde{y})^n{\in}(\mathbb{R}^d, \mathbb{N}_{>0})^n$ and model $\boldsymbol{\theta}{:}\boldsymbol{x}{\to}\hat{p}(\tilde{y})$, if $\hat{p}_{\boldsymbol{x},\tilde{y}=i}$ is **per-class diffracted** without label collisions and each diagonal entry of $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximizes its row, then $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$ and as $n \to \infty$, $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} \approxeq \boldsymbol{Q}_{\tilde{y},y^*}$.*

Proof. Re-stating the meaning of **per-class diffracted**, we wish to show that if $\hat{p}(\tilde{y}; \boldsymbol{x}, \boldsymbol{\theta})$ is diffracted with class-conditional noise s.t. $\forall j \in 1..m, \hat{p}(\tilde{y}=j; \boldsymbol{x}, \boldsymbol{\theta}) = \epsilon_j^{(1)} \cdot p^*(\tilde{y}=j|y^*=y_k^*) + \epsilon_j^{(2)}$ where $\epsilon_j^{(1)} \in \mathcal{R}, \epsilon_j^{(2)} \in \mathcal{R}$ (for any distribution) without label collisions and each diagonal entry of $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximizes its row, then $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$ and $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} \approxeq \boldsymbol{Q}_{\tilde{y},y^*}$.

Firstly, note that combining linear combinations of real-valued $\epsilon_j^{(1)}$ and $\epsilon_j^{(2)}$ with the probabilities of class j for each example may result in some examples having $\hat{p}_{\boldsymbol{x},\tilde{y}=j} = \epsilon_j^{(1)} p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_j^{(2)} > 1$ or $\hat{p}_{\boldsymbol{x},\tilde{y}=j} = \epsilon_j^{(1)} p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_j^{(2)} < 0$. The proof makes no assumption about the validity of model outputs and therefore holds when this occurs. Furthermore, confident learning does not require valid probabilities when finding label errors because confident learning uses the *rank* principle, not probabilities.

When there are no label collisions, the bins created by the confident joint are:

$$\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} \coloneqq \{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i} : \hat{p}(\tilde{y}=j; \boldsymbol{x}, \boldsymbol{\theta}) \geq t_j\} \tag{7}$$

where

$$t_j = \mathop{\mathbb{E}}_{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=j}} \hat{p}_{\boldsymbol{x},\tilde{y}=j}$$

WLOG: we re-formulate the error $\epsilon_j^{(1)} p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_j^{(2)}$ as $\epsilon_j^{(1)}(p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_j^{(2)})$.

Now, for diffracted (non-ideal) probabilities, we re-write how the threshold t_j changes for a given $\epsilon_j^{(1)}, \epsilon_j^{(2)}$:

$$\begin{aligned}
t_j^{\epsilon_j} &= \mathop{\mathbb{E}}_{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=j}} \epsilon_j^{(1)}(p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_j^{(2)}) \\
t_j^{\epsilon_j} &= \epsilon_j^{(1)} \left(\mathop{\mathbb{E}}_{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=j}} p^*_{\boldsymbol{x},\tilde{y}=j} + \mathop{\mathbb{E}}_{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=j}} \epsilon_j^{(2)} \right) \\
t_j^{\epsilon_j} &= \epsilon_j^{(1)} \left(t_j^* + \epsilon_j^{(2)} \cdot \mathop{\mathbb{E}}_{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=j}} 1 \right) \\
t_j^{\epsilon_j} &= \epsilon_j^{(1)}(t_j^* + \epsilon_j^{(2)})
\end{aligned}$$

Thus, for per-class diffracted (non-ideal) probabilities, Eqn (7) becomes

$$\begin{aligned}
\hat{\boldsymbol{X}}^{\epsilon_j}_{\tilde{y}=i,y^*=j} &= \{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i} : \epsilon_j^{(1)}(p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_j^{(2)}) \geq \epsilon_j^{(1)}(t_j^* + \epsilon_j^{(2)})\} \\
&= \{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i} : p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j^*\} \\
&= \boldsymbol{X}_{\tilde{y}=i,y^*=j} \qquad \triangleright \text{ by Thm. (1)}
\end{aligned}$$

In the second to last step, we see that the formulation of the label errors is the formulation of $\boldsymbol{C}_{\tilde{y},y^*}$ for *ideal* probabilities, which we proved yields exact label errors and consistent estimation of $\boldsymbol{Q}_{\tilde{y},y^*}$ in Theorem 1, which concludes the proof. Note that we eliminate the need for the assumption that each diagonal entry of $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximizes its column because this assumption is only used in the proofs of Theorem 1 when collisions occur, but here we only consider the case when there are no collisions.

□

Theorem 2 (General Per-Example Robustness)**.** *For a noisy dataset $\boldsymbol{X} \coloneqq (\boldsymbol{x}, \tilde{y})^n \in (\mathbb{R}^d, \mathbb{N}_{>0})^n$ and model $\boldsymbol{\theta}{:}\boldsymbol{x} \to \hat{p}(\tilde{y})$, if $\hat{p}_{\boldsymbol{x},\tilde{y}=i}$ is **per-example diffracted** without label collisions and each diagonal entry of $\boldsymbol{Q}_{\tilde{y}|y^*}$ maximizes its row, then $\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} = \boldsymbol{X}_{\tilde{y}=i,y^*=j}$ and as $n \to \infty$, $\hat{\boldsymbol{Q}}_{\tilde{y},y^*} \approxeq \boldsymbol{Q}_{\tilde{y},y^*}$ (consistent).*

Proof. We consider the non-trivial, real-world setting when a learning model $\boldsymbol{\theta}{:}\boldsymbol{x} \to \hat{p}(\tilde{y})$ outputs erroneous, non-ideal predicted probabilities with an error term added for every example, across every class, such that $\forall \boldsymbol{x} \in \boldsymbol{X}, \forall j \in 1..m,\ \hat{p}_{\boldsymbol{x},\tilde{y}=j} = p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_{\boldsymbol{x},\tilde{y}=j}$. As a notation reminder $p^*_{\boldsymbol{x},\tilde{y}=j}$ is shorthand for the ideal probabilities $p^*(\tilde{y}=j|y^*=y_k^*) + \epsilon_{\boldsymbol{x},\tilde{y}=j}$ and $\hat{p}_{\boldsymbol{x},\tilde{y}=j}$ is shorthand for the predicted probabilities $\hat{p}(\tilde{y}=j; \boldsymbol{x}, \boldsymbol{\theta})$.

The predicted probability error $\epsilon_{\boldsymbol{x},\tilde{y}=j}$ is distributed uniformly with no other constraints. We use $\epsilon_j \in \mathcal{R}$ to represent the mean of $\epsilon_{\boldsymbol{x},\tilde{y}=j}$ per class, i.e. $\epsilon_j = \mathbb{E}_{\boldsymbol{x}\in\boldsymbol{X}}\, \epsilon_{\boldsymbol{x},\tilde{y}=j}$, which can be seen by looking at the form of the uniform distribution in Eqn. (5). If we wanted, we could add the constraint that $\epsilon_j = 0, \forall j \in 1..m$ which would simplify the theorem and the proof, but is not as general and we prove exact label error and joint estimation without this constraint.

We re-iterate the form of the error in Eqn. (5) here ($\mathcal{U}$ denotes a uniform distribution):

$$\epsilon_{\boldsymbol{x},\tilde{y}=j} \sim \begin{cases} U(\epsilon_j + t_j - p^*_{\boldsymbol{x},\tilde{y}=j}\,,\, \epsilon_j - t_j + p^*_{\boldsymbol{x},\tilde{y}=j}] & p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j \\ \mathcal{U}[\epsilon_j - t_j + p^*_{\boldsymbol{x},\tilde{y}=j}\,,\, \epsilon_j + t_j - p^*_{\boldsymbol{x},\tilde{y}=j}) & p^*_{\boldsymbol{x},\tilde{y}=j} < t_j \end{cases}$$

When there are no label collisions, the bins created by the confident joint are:

$$\hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} \coloneqq \{\boldsymbol{x} \in \boldsymbol{X}_{\tilde{y}=i} : \hat{p}_{\boldsymbol{x},\tilde{y}=j} \geq t_j\} \tag{8}$$

where

$$t_j = \frac{1}{|X_{\tilde{y}=j}|} \sum_{\boldsymbol{x}\in\boldsymbol{X}_{\tilde{y}=j}} \hat{p}_{\boldsymbol{x},\tilde{y}=j}$$

Rewriting the threshold t_j to include the error terms $\epsilon_{\boldsymbol{x},\tilde{y}=j}$ and ϵ_j, we have

$$\begin{aligned} t_j^{\epsilon_j} &= \frac{1}{|X_{\tilde{y}=j}|} \sum_{\boldsymbol{x}\in\boldsymbol{X}_{\tilde{y}=j}} p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_{\boldsymbol{x},\tilde{y}=j} \\ t_j^{\epsilon_j} &= \mathop{\mathbb{E}}_{\boldsymbol{x}\in\boldsymbol{X}_{\tilde{y}=j}} p^*_{\boldsymbol{x},\tilde{y}=j} + \mathop{\mathbb{E}}_{\boldsymbol{x}\in\boldsymbol{X}_{\tilde{y}=j}} \epsilon_{\boldsymbol{x},\tilde{y}=j} \\ &= t_j + \epsilon_j \end{aligned}$$

where the last step uses the fact that $\epsilon_{\boldsymbol{x},\tilde{y}=j}$ is uniformly distributed and $n \to \infty$ so that $\mathbb{E}_{\boldsymbol{x}\in\boldsymbol{X}_{\tilde{y}=j}}\, \epsilon_{\boldsymbol{x},\tilde{y}=j} = \mathbb{E}_{\boldsymbol{x}\in\boldsymbol{X}}\, \epsilon_{\boldsymbol{x},\tilde{y}=j}$. We now complete the proof by showing that

$$p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_{\boldsymbol{x},\tilde{y}=j} \geq t_j + \epsilon_j \iff p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j$$

If this statement is true then the subsets created by the confident joint in Eqn. 8 are unaltered and therefore $\hat{\boldsymbol{X}}^{\epsilon_{\boldsymbol{x},\tilde{y}=j}}_{\tilde{y}=i,y^*=j} = \hat{\boldsymbol{X}}_{\tilde{y}=i,y^*=j} \overset{Thm.1}{=} \boldsymbol{X}_{\tilde{y}=i,y^*=j}$, where $\hat{\boldsymbol{X}}^{\epsilon_{\boldsymbol{x},\tilde{y}=j}}_{\tilde{y}=i,y^*=j}$ denotes the confident joint subsets for $\epsilon_{\boldsymbol{x},\tilde{y}=j}$ predicted probabilities.

Now we complete the proof. From Eqn. 5 (the distribution for $\epsilon_{\boldsymbol{x},\tilde{y}=j}$) , we have that

$$\begin{aligned} p^*_{\boldsymbol{x},\tilde{y}=j} < t_j &\implies \epsilon_{\boldsymbol{x},\tilde{y}=j} < \epsilon_j + t_j - p^*_{\boldsymbol{x},\tilde{y}=j} \\ p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j &\implies \epsilon_{\boldsymbol{x},\tilde{y}=j} \geq \epsilon_j + t_j - p^*_{\boldsymbol{x},\tilde{y}=j} \end{aligned}$$

Re-arranging

$$\begin{aligned} p^*_{\boldsymbol{x},\tilde{y}=j} < t_j &\implies p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_{\boldsymbol{x},\tilde{y}=j} < t_j + \epsilon_j \\ p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j &\implies p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_{\boldsymbol{x},\tilde{y}=j} \geq t_j + \epsilon_j \end{aligned}$$

Using the contrapositive, we have

$$\begin{aligned} p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_{\boldsymbol{x},\tilde{y}=j} \geq t_j + \epsilon_j &\implies p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j \\ p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j &\implies p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_{\boldsymbol{x},\tilde{y}=j} \geq t_j + \epsilon_j \end{aligned}$$

Combining, we have

$$p^*_{\boldsymbol{x},\tilde{y}=j} + \epsilon_{\boldsymbol{x},\tilde{y}=j} \geq t_j + \epsilon_j \iff p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j$$

Therefore,

$$\hat{\boldsymbol{X}}^{\epsilon_{\boldsymbol{x},\tilde{y}=j}}_{\tilde{y}=i,y^*=j} \overset{Thm.1}{=} \boldsymbol{X}_{\tilde{y}=i,y^*=j}$$

The last line follows from the fact that we've reduced $\hat{\boldsymbol{X}}^{\epsilon_{\boldsymbol{x}},\tilde{y}=j}_{\tilde{y}=i,y^*=j}$ to counting the same condition ($p^*_{\boldsymbol{x},\tilde{y}=j} \geq t_j$) as the confident joint counts under ideal probabilities in Thm (1). Thus, we maintain exact label errors and also consistent estimation (Corollary 1.1) holds under no label collisions. While we assume there are infinite examples in $\boldsymbol{X}$, the proof applies for finite datasets if you omit discretization error.

Note that while we use a uniform distribution in Eqn. 5, any bounded symmetric distribution with mode $\epsilon_j = \mathbb{E}_{\boldsymbol{x} \in \boldsymbol{X}}\, \epsilon_{\boldsymbol{x},j}$ is sufficient. Observe that the bounds of the distribution are non-vacuous (they do not collapse to a single value e_j) because $t_j \neq p^*_{\boldsymbol{x},\tilde{y}=j}$ by Lemma 1.

□

C. The confident joint and joint algorithms

The confident joint can be expressed succinctly in equation Eqn 3 with the thresholds expressed in Eqn 2. For clarity, we provide these equations in algorithm-form below.

Algorithm 1 The Confident Joint Algorithm for class-conditional label noise characterization.

input $\hat{\boldsymbol{P}}$ an $n \times m$ matrix of out-of-sample predicted probabilities $\hat{\boldsymbol{P}}[i][j] \coloneqq \hat{p}(\tilde{y} = j; x, \boldsymbol{\theta})$
input $\tilde{\boldsymbol{y}} \in {\mathbb{N}_{\geq 0}}^n$, an $n \times 1$ array of noisy labels
procedure CONFIDENTJOINT($\hat{\boldsymbol{P}}$, $\tilde{\boldsymbol{y}}$):
PART 1 (COMPUTE THRESHOLDS)
for $j \leftarrow 1, m$ **do**
  for $i \leftarrow 1, n$ **do**
    $l \leftarrow$ new empty list []
    if $\tilde{\boldsymbol{y}}[i] = j$ **then**
      append $\hat{\boldsymbol{P}}[i][j]$ to l
  $\boldsymbol{t}[j] \leftarrow$ average(l)
PART 2 (COMPUTE CONFIDENT JOINT)
$\boldsymbol{C} \leftarrow m \times m$ matrix of zeros
for $i \leftarrow 1, m$ **do**
  $cnt \leftarrow 0$
  for $j \leftarrow 1, m$ **do**
    if $\hat{\boldsymbol{P}}[i][j] \geq \boldsymbol{t}[j]$ **then**
      $cnt \leftarrow cnt + 1$
      $y^* \leftarrow j$ ▷ guess of true label
  $\tilde{y} \leftarrow \tilde{\boldsymbol{y}}[i]$
  if $cnt > 1$ **then** ▷ if label collision
    $y^* \leftarrow \arg\max \hat{\boldsymbol{P}}[i]$
  if $cnt > 0$ **then**
    $\boldsymbol{C}[\tilde{y}][y^*] \leftarrow \boldsymbol{C}[\tilde{y}][y^*] + 1$
output $\boldsymbol{C}$ $_{m \times m}$ unnormalized counts matrix

The confident joint algorithm (Alg. 1) is an $\mathcal{O}(m^2 + nm)$ step procedure to compute $\boldsymbol{C}_{\tilde{y},y^*}$. The algorithm takes two inputs: (1) $\hat{\boldsymbol{P}}$ an $n \times m$ matrix of out-of-sample predicted probabilities $\hat{\boldsymbol{P}}[i][j] \coloneqq \hat{p}(\tilde{y} = j; x_i, \boldsymbol{\theta})$ and (2) the associated array of noisy labels. We typically use cross-validation to compute $\hat{\boldsymbol{P}}$ for train sets and a model trained on the train set and fine-tuned with cross-validation on the test set to compute $\hat{\boldsymbol{P}}$ for a test set. Any method works as long $\hat{p}(\tilde{y} = j; \boldsymbol{x}, \boldsymbol{\theta})$ are out-of-sample, holdout, predicted probabilities.

Computation time. Finding label errors in ImageNet takes 3 minutes on an i7 CPU. All tables are seeded and reproducible via open-sourced `cleanlab` package.

Note that Alg. 1 embodies Eqn. 3, and Alg. 2 realizes Eqn. 4.

Algorithm 2 The Joint Algorithm calibrates the confident joint to estimate the latent, true distribution of class-conditional label noise

input $\boldsymbol{C}_{\tilde{y},y^*}[i][j]$, $m \times m$ unnormalized counts
input $\tilde{\boldsymbol{y}}$ an $n \times 1$ array of noisy integer labels
procedure JOINTESTIMATION($\boldsymbol{C}$, $\tilde{\boldsymbol{y}}$):

$$\tilde{\boldsymbol{C}}_{\tilde{y}=i,y^*=j} \leftarrow \frac{\boldsymbol{C}_{\tilde{y}=i,y^*=j}}{\sum_{j \in 1..m} \boldsymbol{C}_{\tilde{y}=i,y^*=j}} \cdot |\boldsymbol{X}_{\tilde{y}=i}| \qquad \triangleright \text{ calibrate marginals}$$

$$\hat{\boldsymbol{Q}}_{\tilde{y}=i,y^*=j} \leftarrow \frac{\tilde{\boldsymbol{C}}_{\tilde{y}=i,y^*=j}}{\sum\limits_{i \in 1..m, j \in 1..m} \tilde{\boldsymbol{C}}_{\tilde{y}=i,y^*=j}} \qquad \triangleright \text{ joint sums to 1}$$

output $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$ joint dist. matrix $\sim p(\tilde{y}, y^*)$

D. Extended Comparison of Confident Learning Methods on CIFAR-10

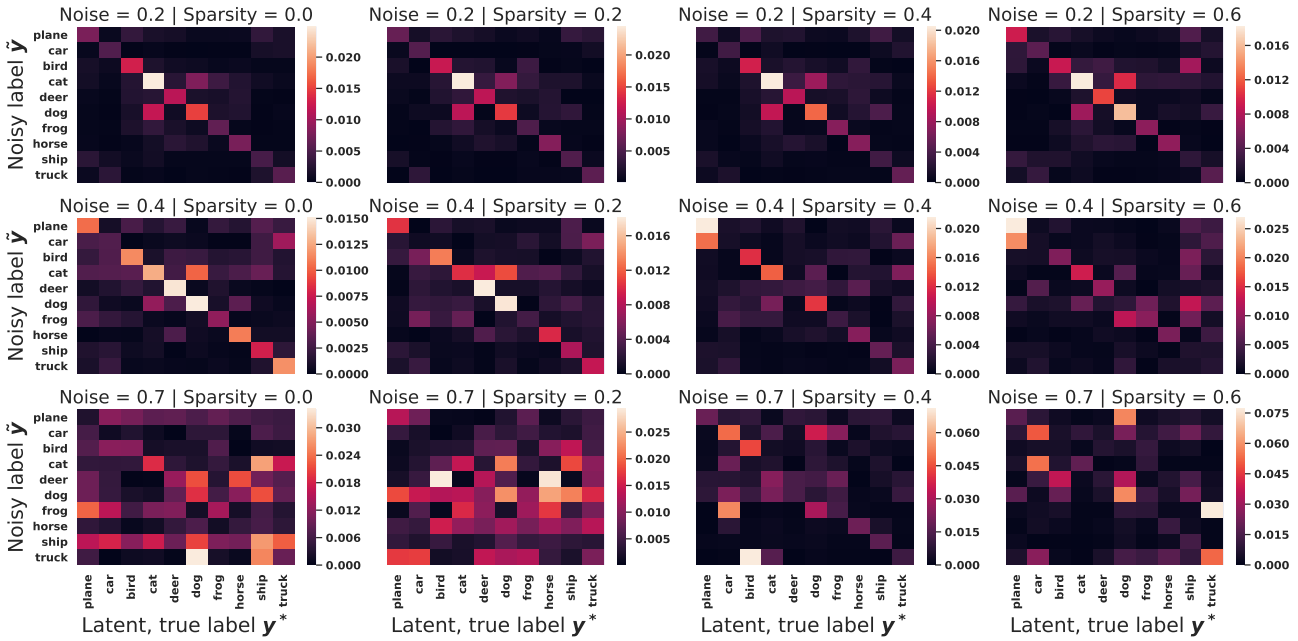


Figure 5. Absolute difference of the true joint $\boldsymbol{Q}_{\tilde{y},y^*}$ and the joint distribution estimated using confident learning $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$ on CIFAR-10, for 20%, 40%, and 70% label noise, 20%, 40%, and 60% sparsity, for all pairs of classes in the joint distribution of label noise.

Fig. 5 shows the absolute difference ofbsolute difference of the true joint $\boldsymbol{Q}_{\tilde{y},y^*}$ and the joint distribution estimated using confident learning $\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$ on CIFAR-10, for 20%, 40%, and 70% label noise, 20%, 40%, and 60% sparsity, for all pairs of classes in the joint distribution of label noise. Observe that in moderate noise regimes between 20% and 40% noise, confident learning accurately estimates nearly every entry in the joint distribution of label noise. This figure serves to provide evidence for how confident learning is able to identify the label errors with high accuracy as shown in Table 1 as well as support our theoretical contributions that confident learning is a consistent estimator for the joint distribution.

To our knowledge, for the first time we train both the ResNet50 (9 in Appendix) and ResNet18 (Fig. 8 in Appendix) models on an automatically cleaned ImageNet training set. In Figure 9 we demonstrate how validation accuracy of ResNet50 improves when removing label errors estimated with confident learning versus random examples. In (a) we see the validation accuracy when random pruning is used versus confident learning. Sub-figure (b) shows how that discrepancy increases as we look at the 20 noisiest classes identified by confident learning. The bottom two figures show a consistent increase in accuracy on the class identified as most noisy by confident learning (c) and a moderately noisy class (d). For (d) the spike improvement is in the middle because confident learning ranks examples and this class is not ranked as the noisiest, so label errors in that class did not get removed until 40%-60% of label errors are removed.

Table 5. RMSE error of joint distribution estimation on CIFAR-10 using CL ($\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$) vs. ($\hat{\boldsymbol{Q}}_{argmax}$).

NOISE	0.2				0.4				0.7			
SPARSITY	0	0.2	0.4	0.6	0	0.2	0.4	0.6	0	0.2	0.4	0.6
$\|\hat{\boldsymbol{Q}}_{\tilde{y},y^*} - \boldsymbol{Q}_{\tilde{y},y^*}\|_2$	0.004	0.004	0.004	0.004	0.004	0.004	0.004	0.005	0.011	0.010	0.015	0.017
$\|\hat{\boldsymbol{Q}}_{argmax} - \boldsymbol{Q}_{\tilde{y},y^*}\|_2$	0.006	0.006	0.005	0.005	0.005	0.005	0.005	0.007	0.011	0.011	0.015	0.019

Note we did not remove label errors from the validation set, so by training on a clean train set, we may have induced distributional shift, making the moderate increase accuracy more satisfying.

In Table 5 in the Appendix, we estimate the $\boldsymbol{Q}_{\tilde{y},y^*}$ using using the confusion-matrix $\boldsymbol{C}_{\text{confusion}}$ approach normalized via Eqn. (4) to obtain $\hat{\boldsymbol{Q}}_{argmax}$ and compare this with CL ($\hat{\boldsymbol{Q}}_{\tilde{y},y^*}$) for various amounts of noise and sparsity in $\boldsymbol{Q}_{\tilde{y},y^*}$. We show significant improvement using CL over $\boldsymbol{C}_{\text{confusion}}$, low RMSE scores, and robustness to sparsity in moderate-noise regimes.

E. Comparison of INCV Method and Confident Learning

The INCV algorithm (Chen et al., 2019) and confident learning both estimate clean data. Both use cross-validation and both uses aspects of confusion matrices for finding label errors. For learning with noisy labels, both us Co-Teaching (Han et al., 2018). However, confident learning has three key advantages over INCV.

First, INCV finds label errors by iteratively re-training on clean data and data it is unsure of, called the *candidate set*. Whereas INCV modifies Co-Teaching to first train with the clean set, then subsequently train with the combined clean and candidate set, confident learning just uses the standard unmodified Co-Teaching method with good results (see Table 1. This makes confident learning simple to use: find the label errors, remove them from the dataset, apply standard Co-Training.

Second, in each INCV training iteration, 2-fold cross-validation is performed. As we show in Table 6, this makes training slow and uses less data during training. Unlike INCV, confident learning is not iterative. In confident learning, cross-validated probabilities are computed only once beforehand from which the joint distribution of noisy and true labels is directly estimated which is used to identify all label errors with re-training.

Third, INCV errors are found similarly to the $\boldsymbol{C}_{\text{confusion}}$ confident learning baseline: any example with a different given label then its argmax prediction is considered a label error. This approach, while effective (see Table 1), fails to properly count errors for class imbalance or when a model is more confident (larger or smaller probabilities on average) for certain class than others, as discussed in Section 4. To account for this class-level bias in predicted probabilities and enable robustness, confident learning uses theoretically-supported thresholds (Elkan, 2001; Richard & Lippmann, 1991) while estimating the confident joint. CL further shows this procedure yields a consistent estimator for the true joint distribution of noisy labels and true labels under realistic conditions.

E.1. Benchmarking INCV

We benchmarked INCV using the official Github code[2] on a machine with 128 GB of RAM and 4 RTX 2080 ti GPUs. Due to memory leak issues in the implementation, training frequently stopped due to out-of-memory errors. For fair comparison, we restarted INCV training until all models trained for at least 90 epochs. The total time require for training, accuracies, and epochs completed, for each training is presented in Table 6. As shown in the table, INCV out-performed every other method except for confident learning, but training could take over 20 hours. For comparison CL takes less than three hours on the same machine: an hour for cross-validation, less than a minute to find errors, an hour to re-train.

Table 6. Information about INCV benchmark including accuracy, time, and epochs trained for the various noise and sparsity settings test.

NOISE	0.2				0.4				0.7			
SPARSITY	0	0.2	0.4	0.6	0	0.2	0.4	0.6	0	0.2	0.4	0.6
ACCURACY	0.878	0.886	0.896	0.892	0.844	0.766	0.854	0.736	0.283	0.253	0.348	0.297
TIME (HOURS)	9.120	11.350	10.420	7.220	7.580	11.720	20.420	6.180	16.230	17.250	16.880	18.300
EPOCHS TRAINED	91	91	200	157	91	200	200	139	92	92	118	200

[2] https://github.com/chenpf1025/noisy_label_understanding_utilizing

F. Case Study: MNIST

In the past two decades, the machine learning and vision communities have increasingly used the MNIST and ImageNet (Russakovsky et al., 2015) datasets as benchmarks to measure state-of-the-art progress and evaluate theoretical findings. The trustworthiness of these benchmarks rely on an implicit, and as we show fallacious, assumption that these datasets have noise-free labels. In this section, we unveil the existence of numerous label errors in both datasets and thereby demonstrate the efficacy of confident learning as a general tool for automatically detecting label errors in massive, crowd-sourced datasets. For MNIST, comprised of black-and-white handwritten digits, we estimate 48 (among 60,000) train label errors and 8 (among 10,000) test label errors, and for the 2012 ImageNet validation set, comprised of 50 color images for each of 1000 distinct classes, we estimate 5,000 (among 50,000) validation labels are erroneous or confounded (the image has more than one valid label).

MNIST train set We used confident learning to automatically identify label errors without human intervention in the original, unperturbed MNIST train set. The key computational step is computing the $n \times m$ matrix of predicted probabilities. We first pre-trained a PyTorch MNIST CNN (architecture in Supplementary Materials Fig. 10) for 50 epochs, then used five-fold cross-validation to obtain $\hat{p}(\tilde{y} = k; x)$, the out-of-sample, holdout, transductive predicted probabilities for the train set. The 24 least confident examples in X_{err}, i.e. *likely label errors*, identified by confident learning are shown in Fig. 6. Errors are ordered left-right, top-down by increasing self-confidence, $\hat{p}(\tilde{y} = k; x \in X_{\tilde{y}=k})$, denoted as *conf* in teal with the MNIST-provided label. The green boxes depict the $\arg\max \hat{p}(\tilde{y} = k; \boldsymbol{x} \in \boldsymbol{X})$ predicted label and confidence . The results are compelling, with the obvious errors enclosed in red. Unlike most approaches, ours is transductive, thus no information from other datasets is needed. For verification, the indices of the train label errors in Fig. 6 are shown in grey.

Figure 6. Label errors of the original MNIST train dataset identified algorithmically with CL: PBNR. Depicts the 24 least confident labels, ordered left-right, top-down by increasing self-confidence, denoted *conf* in teal. $\arg\max \hat{p}(\tilde{y} = k; x)$ label is in green. Overt errors are in red.

The confidences (in green) in Fig. 6 tend to extreme values. This *overconfidence* is typical of deep architectures. Because confident learning is classifier agnostic, we also tried the default scikit-learn implementation of logistic regression with isometric calibration (Fig. 11 in the Supplementary Materials). Although the confidence values are less extreme than a convnet, the results are perceptibly less accurate. The extremity of a convnet's confidence values can be improved using modern calibration techniques (Guo et al., 2017). However, this is unnecessary because calibration is a monotonic operation and confident learning depends only on the *rank* of the confidence values, hence the name. Additionally, the confident joint estimation step decides its thresholds based on an average of the probabilities, for which monotonic adjustment is vacuous. These properties, along with the quality of label error identification in Fig. 6 follow from *removal by rank* principles of confident learning.

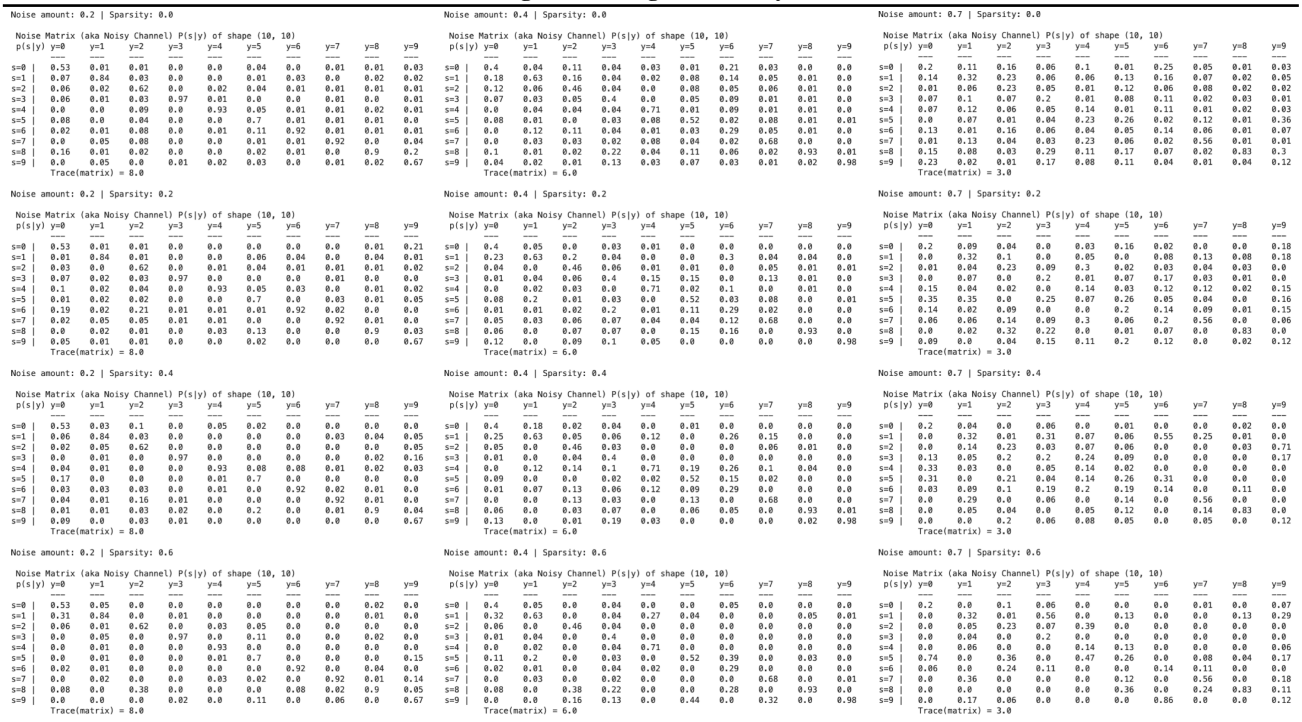


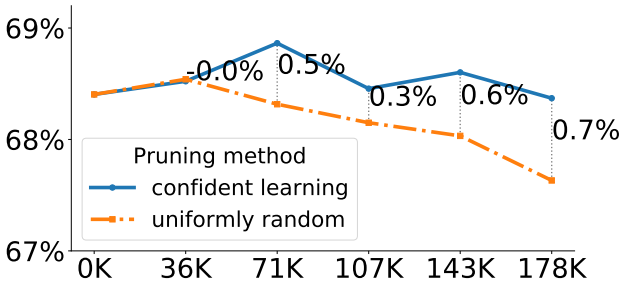
Figure 7. The CIFAR-10 noise transition matrices used to create the synthetic label errors. In the code base, s is used in place of $\tilde{y}$ to notate the noisy unobserved labels and y is used in place of y^* to notate the latent uncorrupted labels.

G. Additional Figures

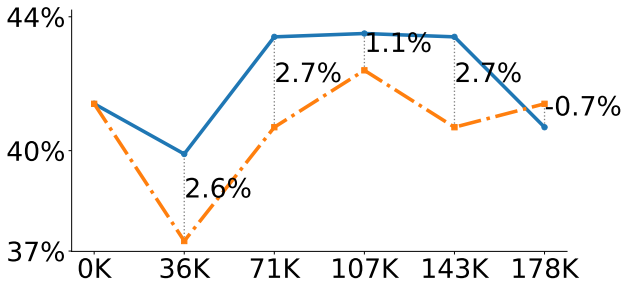
In this section, we include additional figures that support the main manuscript.

The noise matrices shown in Fig. 7 were used to generate the synthetic noisy labels for the results in Tables 2 and 1 .

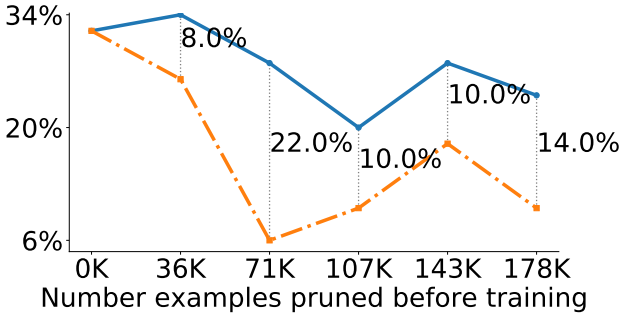
Fig. 8 replicates the experiment from Fig. 9 using ResNet18 instead of ResNet50. Results are reasonably similar.



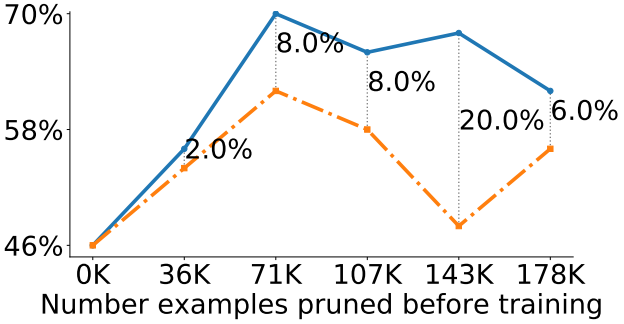
(a) Accuracy of Validation Set



(b) Accuracy of top 20 noisiest classes

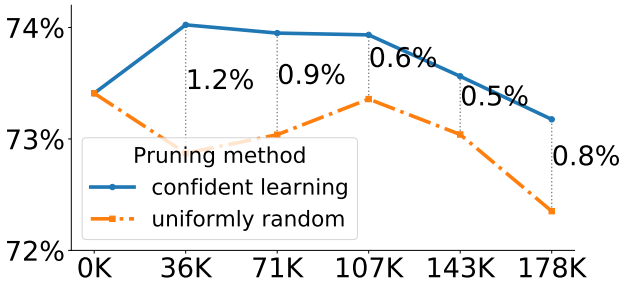


(c) Accuracy of the noisiest class: foxhound

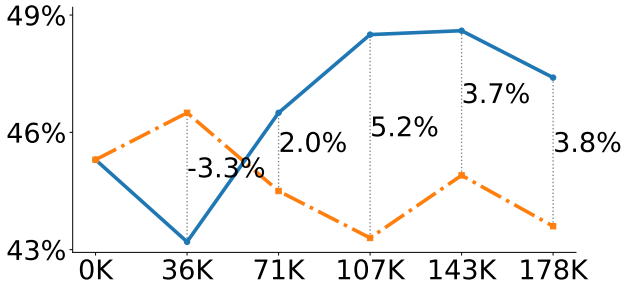


(d) Accuracy of a moderately noisy class: bighorn

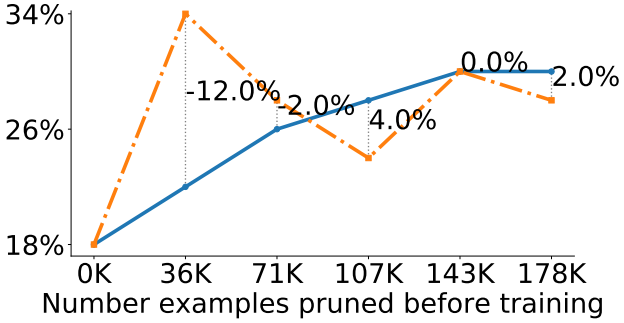
Figure 8. ResNet18 Val Accuracy on ImageNet when 20%, 40%,...,100% of the label errors found automatically with confident learning are removed prior to training from scratch. Vertical bars depict the improvement or reduction when removing examples with confident learning vs random examples.



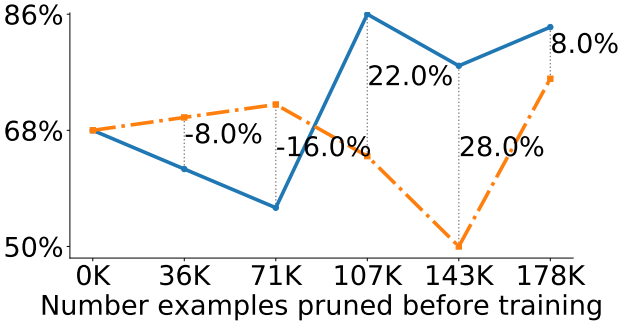
(a) Accuracy of Validation Set



(b) Accuracy of top 20 noisiest classes



(c) Accuracy of the noisiest class: foxhound



(d) Accuracy of a moderately noisy class: bighorn

Figure 9. ResNet50 Val Accuracy on ImageNet when 20%, 40%,...,100% of the label errors found automatically with confident learning are removed prior to training from scratch. Vertical bars depict the improvement or reduction when removing examples with confident learning vs random examples.

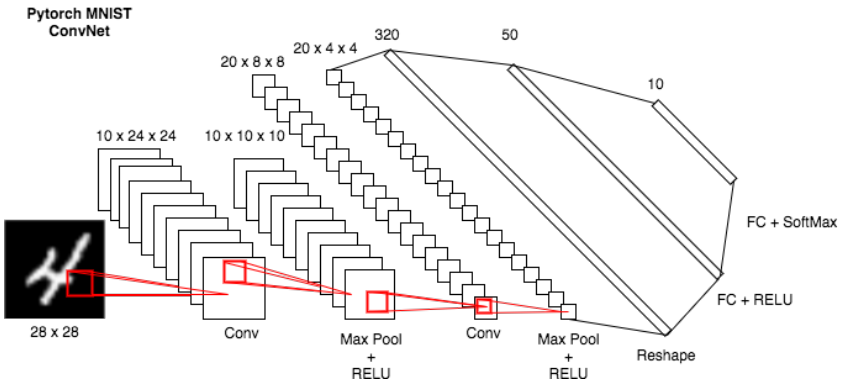


Figure 10. Architecture used for training and probability estimation in our MNIST experiments and figures. This figure was adapted from https://github.com/floydhub/mnist (2018).

Figure 11. Label errors of the original MNIST train dataset identified algorithmically with CL: PBNR using the default scikit-learn implementation of **logistic regression with isometric calibration**. Depicts the 24 least confident labels, ordered left-right, top-down by increasing self-confidence $\hat{p}(\tilde{y} = k|X_{\tilde{y}=k})$, denoted *conf* in teal. The $\arg\max \hat{p}(\tilde{y} = k|x)$ label is in green. Overt errors are in red. Note the confidence values are less extreme than when using a convolutional network, however, the results are perceptibly less accurate.

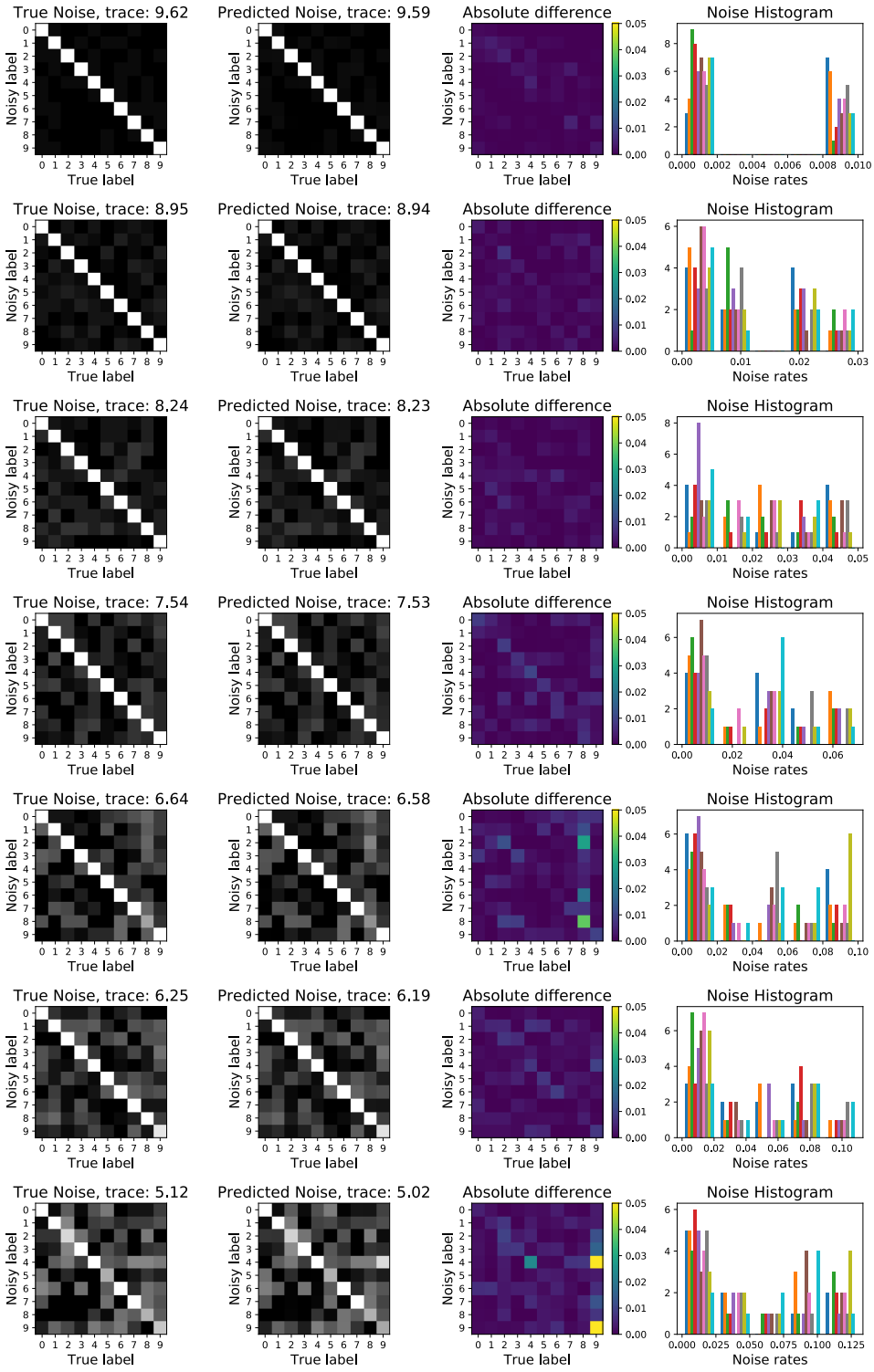


Figure 12. Evidence of nearly perfect noise rate matrix $\hat{\boldsymbol{Q}}_{\tilde{y}|y^*}$ estimation by confident learning. With each row, the label noise increases from top to bottom. The histograms of individual noise rate values in $\hat{\boldsymbol{Q}}_{\tilde{y}|y^*}$ capture the challenging asymmetry. Noise rates were uniformly randomly generated between 0 and 0.2 . To illustrate the severity of noise in the lowest row, note that a trace of 5.12 implies $\frac{1}{10}\sum_{i=1}^{m=10} p(\tilde{y}=i|y=i) = 0.512$.