Self-Relaxed Joint Training: Sample Selection for Severity Estimation with Ordinal Noisy Labels

Shumpei Takezaki[1] Kiyohito Tanaka[2] Seiichi Uchida[1]
[1]Kyushu University, Fukuoka, Japan [2]Kyoto Second Red Cross Hospital, Kyoto, Japan
shumpei.takezaki@human.ait.kyushu-u.ac.jp

Abstract

Severity level estimation is a crucial task in medical image diagnosis. However, accurately assigning severity class labels to individual images is very costly and challenging. Consequently, the attached labels tend to be noisy. In this paper, we propose a new framework for training with "ordinal" noisy labels. Since severity levels have an ordinal relationship, we can leverage this to train a classifier while mitigating the negative effects of noisy labels. Our framework uses two techniques: clean sample selection and dual-network architecture. A technical highlight of our approach is the use of soft labels derived from noisy hard labels. By appropriately using the soft and hard labels in the two techniques, we achieve more accurate sample selection and robust network training. The proposed method outperforms various state-of-the-art methods in experiments using two endoscopic ulcerative colitis (UC) datasets and a retinal Diabetic Retinopathy (DR) dataset. Our codes are available at https://github.com/shumpei-takezaki/Self-Relaxed-Joint-Training.

1. Introduction

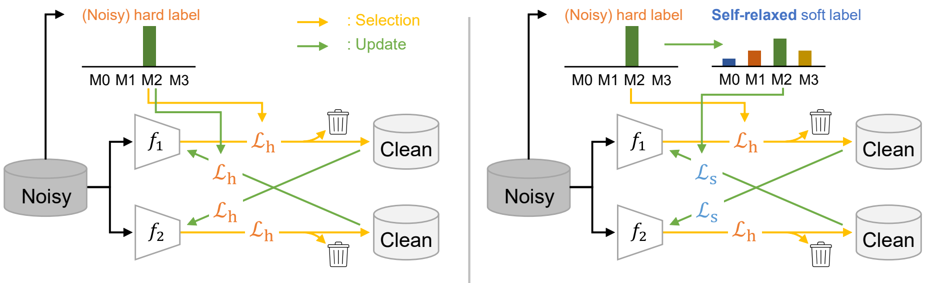
Estimating severity levels is one of the important tasks in medical image diagnosis. Traditionally, medical experts evaluate severity levels as discrete *ordinal* labels. For example, the severity levels of endoscopic ulcerative colitis (UC) images are evaluated as four-level Mayo scores [29]; Mayo 0 is a normal or inactive disease, and Mayo 3 is a severe disease.

A practical and serious problem in estimating severity levels by machine-learning models is that ordinal labels attached to individual images tend to be noisy. This is because the severity is continuous, and many ambiguous cases exist. For example, for an image with a severity level of around 1.5, one expert might label it 1 and another 2. As a possible remedy, several public datasets, such as LIMUC [26] for UC, employ multiple medical experts to attach reliable labels. This remedy, however, requires enormous efforts from multiple experts. Furthermore, it is impractical for diagnostic applications in rare diseases where only a few experts exist.

A more efficient remedy to the noisy-label problem is to utilize some learning models with noisy labels [2, 17, 31]. Fig. 1(a) shows a traditional joint-training framework for noisy labels. This framework employs two techniques and has experimentally demonstrated its high performance. First, the joint-training framework employs sample selection for discarding images with suspicious labels. Specifically, an image with a class label c is discarded when a loss by c shows a high value. Second, the framework employs a dual-network architecture. Roughly speaking, two networks f_1 and f_2 are trained jointly in a complementary manner by referring to each other's loss values. This joint-training framework is useful in canceling the negative effect of noisy labels. Note that in the traditional framework, both techniques use the same criterion $\mathcal{L}_\mathrm{h}$, which is a loss value evaluated by (noisy) hard labels (i.e., one-hot teacher vectors).

The traditional joint-training framework of Fig. 1(a) still has two issues to be solved for ordinal noisy labels. The first issue is that sample selection by the loss $\mathcal{L}_\mathrm{h}$ might be imperfect. In fact, the past attempts, such as Co-teaching [8], JoCor [34], and CoDis [36], experimentally proved that their label precision (i.e., the ratio of samples with correct labels in the selected samples) is far below 100%. If we cannot discard samples with incorrect labels, they harm updating the network models because they force the model to output the incorrect class.

The second issue is that the traditional framework assumes non-specific noisy labels; in other words, it does not assume the ordinal characteristics of the severity estimation tasks. Most incorrect labels for ordinal labels likely occur between neighboring labels (such as Mayo 1 and 2) and unlikely between distant labels (such as Mayo 0 and 3). More formally, letting y denote the true label, we can assume that the attached noisy label $\tilde{y}$ follows $\tilde{y} \sim \lfloor \mathcal{N}(y, \sigma^2) \rceil$, where $\lfloor \cdot \rceil$ is a round operation and σ^2 controls the noisiness of la-



(a) Traditional sample selection for noisy labels

(b) Proposed

Figure 1. (a) Traditional joint-training framework with dual-network model with sample selection for noisy labels. (b) The proposed *self-relaxed joint training* framework for learning with ordinal noisy labels. "M0" stands for "Mayo 0."

bels.

To solve these two issues, we propose a novel framework called *self-relaxed joint training*, shown in Fig. 1(b). Its key idea is that we have $y \sim \lfloor \mathcal{N}(\tilde{y}, \sigma^2) \rceil$ from the above assumption; that is, given a noisy label $\tilde{y}$, we can estimate the distribution of the true label y. We can use this distribution as a *soft label* "relaxed" from the hard label $\tilde{y}$. Since this soft label is derived from the provided hard label, we refer to it as the "*self*-relaxed" soft label.

Our new framework can efficiently solve the two issues. Specifically, it solves the second issue because it uses the soft labels derived under the above assumption about ordinal noisy labels. It is also possible to solve the first issue by using the loss from the soft label, $\mathcal{L}_\mathrm{s}$, instead of the loss by the hard label, $\mathcal{L}_\mathrm{h}$, for updating the models f_1 and f_2. Roughly speaking, using $\mathcal{L}_\mathrm{s}$ successfully weakens the negative impact of the selected samples with incorrect labels. Note that we still use $\mathcal{L}_\mathrm{h}$ for the criterion of the sample selection; our experimental results clearly show that this synergetic use of $\mathcal{L}_\mathrm{s}$ and $\mathcal{L}_\mathrm{h}$ is significantly effective.

We apply the proposed framework to the state-of-the-art methods in the traditional framework, that is, Co-teaching, CoDis, and JoCor, respectively, and evaluate how their performance is improved through the severity level classification task on two UC endoscopic image datasets and the retinal Diabetic Retinopathy (DR) dataset. The experimental results show significant improvements. Moreover, our methods (i.e., "Co-teaching+Ours," "CoDis+Ours," and "JoCor+Ours") also outperform state-of-the-art methods in various frameworks.

The main contributions of this study are as follows:

- We propose a novel framework called self-relaxed joint training for learning with ordinal noisy labels.
- The proposed framework introduces soft labels for representing ordinal noisy labels and a combination of hard and soft labels for sample selection.
- Our framework is versatile and thus can improve arbitrary methods in the traditional join-training framework, such as [8, 34, 36].
- Experimental evaluations on two UC datasets and the DR dataset with ordinal noisy labels show performance superiority over various state-of-the-art methods.

2. Related Work

2.1. Training a classifier with noisy labels

As noted in Section 1 and shown by Fig. 1(a), the traditional joint-training framework to train a classifier with samples with noisy labels employ two techniques, sample selection and dual networks [31]. The former selects small loss samples as samples with clean labels and discards large loss samples. The latter uses two networks in a complementary manner. For example, Decoupling [22] updates the networks only using samples where the predictions of two different networks differ. Co-teaching [8] aims to utilize the complementary performance of two networks by exchanging their clean samples. Co-teaching+ [40] combines Co-teaching and Decoupling for robust training. CoDis [36] uses possibly clean samples that have high discrepancy prediction probabilities between two networks. On the other hand, JoCor [34] uses the same clean sample set for the two networks and introduces co-regularization to reduce divergence between the networks.

As shown by Fig. 1(b), the proposed framework also uses sample selection and dual networks. Therefore, we can apply the proposed framework to the above methods, such as Co-teaching [8], CoDis [36], and JoCor [34]. The later experiments will show that our methods (e.g., "Co-teaching + Ours") can improve their performance by employing the proposed framework with the self-relaxed soft labels.

In addition to the above methods in the traditional joint-training framework, various strategies have been proposed for handling noisy labels. For example, designing loss functions robust to noisy labels [21, 39, 42] is a popular strategy. Learning with label noise transition matrix uses the probability of label errors for loss functions [11, 25, 37]. Learning to reweight examples uses importance weights for clean samples [6, 18, 28]. Introducing various regularization strategies is also useful to suppress overfitting to the samples with incorrect labels [9,14,38]. In the later experiment, we compare our methods with the most recent methods in the above strategies and prove that ours outperforms them.

2.2. Ordinal regression with noisy labels

Ordinal regression (, or ordinal classification) deals with classes with ordinal relationships and is well-studied in [5, 16, 19, 24, 30]. However, to the authors' best knowledge, only a few papers [1, 7, 12] address ordinal regression with noisy labels. Among them, two papers [1, 12] rely on a strong assumption that multiple labels from multiple annotators are available for each sample. This strong assumption significantly narrows the scope of its applicability. In fact, as noted in Section 1, finding multiple experts is often difficult, especially for rare diseases.

Consequently, only Garg et al. [7] deals with ordinal regression with noisy labels under the same assumption as ours; only a single (noisy) label is attached to each sample. Their method decomposes a C-class ordinal regression problem into $(C-1)$ binary classification problems. Each binary problem aims to classify an input sample as belonging to a class higher or lower than class c. Each classifier uses a loss function that is robust to noisy labels. Consequently, Garg et al. [7] uses a totally different methodology from ours. Moreover, the later experiment shows that ours largely outperforms their method.

2.3. Label smoothing for classification task

Label smoothing (, or soft labeling) has been introduced to improve accuracy across many tasks [4, 27, 32, 33, 43]. In these studies, label smoothing is mainly used for regularization during training [23]. Moreover, previous studies have applied label smoothing to learning with noisy labels, demonstrating improvements in classification performance [20, 35]. Additionally, for ordinal regression tasks, label-smoothing has been used [5].

Our method is a novel framework to combine hard and soft labels for training with ordinal noisy labels. Furthermore, experimental results show that this combination improves classification accuracy compared to using soft labels alone.

3. Self-Relaxed Joint Training for Ordinal Noisy Labels

3.1. Overview

We propose a self-relaxed joint-training framework to train a classifier with ordinal noisy labels. As shown in Fig. 1(b), for learning with ordinal noisy labels, the proposed framework uses $\mathcal{L}_\mathrm{h}$ (the loss by hard labels) for the clean sample selection like the traditional methods, such as Co-teaching [8], CoDis [36], and JoCor [34]. One main difference between our framework and the traditional framework is that ours also uses soft labels, which are relaxed versions of the hard labels, to update dual networks, f_1 and f_2. More specifically, we calculate $\mathcal{L}_\mathrm{s}$ (the loss by the soft label) and use it to update f_1 and f_2. Suppose a sample $\boldsymbol{x}$ is incorrectly labeled as 1 while its correct label is 2. If this sample passes the selection, f_1 and f_2 are updated to classify $\boldsymbol{x}$ as class 1 according to $\mathcal{L}_\mathrm{h}$. In contrast, using $\mathcal{L}_\mathrm{s}$ mitigates the negative impact of incorrectly labeled samples.

Algorithm 1 Proposed framework with Co-teaching

```
1: **Input:** Dataset $\tilde{\mathcal{D}}$, two networks $f_1$ and $f_2$ with initialized weights $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$, learning rate $\eta$, noise rate $\epsilon$, epoch $T'$ and $T_\mathrm{max}$, iteration $t_\mathrm{max}$, temperature $\tau$;
for $T = 1, 2, \ldots, T_\mathrm{max}$ do
    2: Shuffle training set $\tilde{\mathcal{D}}$;
    for $t = 1, \ldots, t_\mathrm{max}$ do
        3: Fetch mini-batch $\tilde{\mathcal{B}}$ from $\tilde{\mathcal{D}}$;
        4: Select clean samples from $\tilde{\mathcal{B}}$ by $\mathcal{L}_\mathrm{h}$ (with $\tau$);
            $\mathcal{B}_1 \leftarrow \arg\min_{\mathcal{B}':|\mathcal{B}'|\geq R(T)|\tilde{\mathcal{B}}|} \mathcal{L}_\mathrm{h}(f_1, \mathcal{B}')$;
            $\mathcal{B}_2 \leftarrow \arg\min_{\mathcal{B}':|\mathcal{B}'|\geq R(T)|\tilde{\mathcal{B}}|} \mathcal{L}_\mathrm{h}(f_2, \mathcal{B}')$;
        5: Derive soft labels $\boldsymbol{l}_\mathrm{s}$ from $\boldsymbol{l}_\mathrm{h}$ for $\mathcal{B}_1, \mathcal{B}_2$ by Eq.(3);
        6: Update networks;
            $\boldsymbol{\theta}_1 \leftarrow \boldsymbol{\theta}_1 - \eta\nabla\mathcal{L}_\mathrm{s}(f_1, \mathcal{B}_2)$;
            $\boldsymbol{\theta}_2 \leftarrow \boldsymbol{\theta}_2 - \eta\nabla\mathcal{L}_\mathrm{s}(f_2, \mathcal{B}_1)$;
    end
    7: Update $R(T) \leftarrow 1 - \min\left\{\frac{T}{T'}\epsilon, \epsilon\right\}$;
end
8: Output: two trained networks with $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$.
```

As indicated by the similarity between Figs. 1(a) and (b), the proposed framework can be applied to various methods with sample selection and dual networks. This application can be done by replacing $\mathcal{L}_\mathrm{h}$ for updating the models by $\mathcal{L}_\mathrm{s}$. This simple replacement, however, has a significant benefit on performance. In this section, we employ Co-teaching [8] as a backbone method because it is the most standard method with sample selection and dual networks. Algorithm 1 shows the entire training procedure of "Co-teaching + Ours," namely, the proposed framework applied to Co-teaching. The details of the algorithm will be explained below.

3.2. Problem formulation

We assume an image dataset $\tilde{\mathcal{D}} = \{\boldsymbol{x}_i, \tilde{y}_i\}_{i=1}^N$, where $\boldsymbol{x}_i$ is the i-th image and $\tilde{y}_i \in \{1, 2, \ldots, C\}$ is its (noisy) class label and the C classes have an ordinal relationship, $1 \prec 2 \prec \ldots \prec C$. For the case of four-level Mayo scores (Mayo 0, 1, 2, and 3) of UC images, the label $\tilde{y}_i = c$ means a UC image $\boldsymbol{x}_i$ is annotated with Mayo $(c-1)$.

Hereafter, we refer to the following one-hot vector $\boldsymbol{l}_\mathrm{h}(\tilde{y}_i)$

as a *hard label*:

$$\boldsymbol{l}_{\mathrm{h}}(\tilde{y}_i) = (0, \ldots, \overset{\tilde{y}_i}{1} \ldots, 0). \tag{1}$$

Note again that $\tilde{y}_i$ is noisy due to the ambiguity of discrete severity levels; consequently, this hard label is also noisy.

We use two deep neural networks $f_1(\boldsymbol{x}_i; \boldsymbol{\theta}_1)$ and $f_2(\boldsymbol{x}_i; \boldsymbol{\theta}_2)$, where $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$ are weight parameters. The networks f_1 and f_2 are trained with soft labels in a complementary manner. We denote the outputs from f_n ($n \in \{1, 2\}$) as $\boldsymbol{p}_n(\boldsymbol{x}_i) = (p_n^1(\boldsymbol{x}_i), \ldots, p_n^C(\boldsymbol{x}_i))$, where the element $p_n^c(\boldsymbol{x}_i)$ is the probability of class c predicted at the softmax layer of f_n.

3.3. Soft labels from hard labels

As noted in Section 1, we can have the distribution of the clean class label y_i from noisy class label $\tilde{y}_i$ by using the ordinal characteristics of the noisy labels. Specifically, if we can assume

$$\tilde{y}_i \sim \lfloor \mathcal{N}(y_i, \sigma^2) \rceil, \tag{2}$$

we have $y_i \sim \lfloor \mathcal{N}(\tilde{y}_i, \sigma^2) \rceil$. Considering this relationship and following [5], we define the *soft label* as $\boldsymbol{l}_{\mathrm{s}}(\tilde{y}_i) = (l_{\mathrm{s}}^1(\tilde{y}_i), \ldots, l_{\mathrm{s}}^C(\tilde{y}_i))$, where

$$l_{\mathrm{s}}^c(\tilde{y}_i) = \frac{\exp(-|c - \tilde{y}_i|)}{\sum_{c'=1}^{C} \exp(-|c' - \tilde{y}_i|)}. \tag{3}$$

The soft label $\boldsymbol{l}_{\mathrm{s}}(\tilde{y}_i)$ has the largest value at $c = \tilde{y}_i$ and becomes smaller as the difference from $\tilde{y}_i$ becomes larger. Therefore, $\boldsymbol{l}_{\mathrm{s}}(\tilde{y}_i)$ is a relaxed version of the hard label $\boldsymbol{l}_{\mathrm{h}}(\tilde{y}_i)$ of Eq.(1)

3.4. Sample selection by hard labels

A sample $\boldsymbol{x}_i$ is selected as clean when its $\mathcal{L}_{\mathrm{h}}$ is small:

$$\begin{aligned} \mathcal{L}_{\mathrm{h}}(f_n, \tilde{\mathcal{B}}) &= -\sum_{\{\boldsymbol{x}_i, \tilde{y}_i\} \in \tilde{\mathcal{B}}} \sum_{c=1}^{C} l_{\mathrm{h}}^c(\tilde{y}_i) \log p_n^c(\boldsymbol{x}_i) \\ &= -\sum_{\{\boldsymbol{x}_i, \tilde{y}_i\} \in \tilde{\mathcal{B}}} \log p_n^{\tilde{y}_i}(\boldsymbol{x}_i). \end{aligned} \tag{4}$$

Specifically, among the samples in a mini-batch $\tilde{\mathcal{B}}$, the $R(T)|\tilde{\mathcal{B}}|$ samples with the smallest loss value are selected at the Tth epoch, where $R(T) \in (0, 1]$ is the selection rate.

Precisely speaking, the softmax layer with a temperature $\tau \in (0, 1)$ is used to make $\boldsymbol{p}_n(\boldsymbol{x}_i)$ more peaky for sample selection. (This is a practical remedy for the over-smoothed output probabilities by the network trained with soft labels.) Moreover, following the traditional setting [8, 34, 36], we used $R(T) = 1 - \min\{\frac{T}{T'}\epsilon, \epsilon\}$, where T' is a hyperparameter and ϵ is the noise rate (i.e., the ratio of incorrect labels) of the dataset. Under this setting, more samples than $(1 - \epsilon)|\tilde{\mathcal{B}}|$ are selected at earlier epochs $T < T'$; this treatment is helpful to avoid a phenomenon that networks tend to overfit (very) clean samples. At later epochs $T \geq T'$, $(1 - \epsilon)|\tilde{\mathcal{B}}|$ samples are selected.

3.5. Updating the network parameters by soft labels

For updating the networks, we use the loss function $\mathcal{L}_{\mathrm{s}}$ evaluated with soft labels $\boldsymbol{l}_{\mathrm{s}}$. In addition, each network uses the samples selected by the other network for its updating, following [8, 36]. Consequently, the network f_1 is updated with the loss function:

$$\mathcal{L}_{\mathrm{s}}(f_1, \mathcal{B}_2) = -\sum_{\{x_i, \tilde{y}_i\} \in \mathcal{B}_2} \sum_{c=1}^{C} l_{\mathrm{s}}^c(\tilde{y}_i) \log p_1^c(\boldsymbol{x}_i), \tag{5}$$

where $\mathcal{B}_2$ is a mini-batch with the samples selected by f_2. For f_2, we use $\mathcal{L}_{\mathrm{s}}(f_2, \mathcal{B}_1)$.

3.6. How to apply our framework to other methods

As noted in Section 3.1, our framework can be applied to any method in the joint-training framework of Fig. 1(a). The application is done with two modifications of the traditional framework. First, as shown in Algorithm 1 of "Co-teaching + Ours," ours employs $\mathcal{L}_{\mathrm{s}}$ for updating the networks, whereas the original Co-teaching [8] uses $\mathcal{L}_{\mathrm{h}}$. Second, the temperature τ is introduced to the softmax layer of the networks. Similar modifications of CoDis [36] and JoCor [34] give "CoDis+Ours" and "JoCor+Ours." (See the supplementary file for their details.) These modifications seem simple but result in a significant improvement in sample selection and classification.

4. Experimental Evaluation

4.1. Experimental Setup

Datasets with clean labels. We used two datasets of UC endoscopic images. One is the publicly available dataset, LIMUC [26], which contains 11,276 images from 564 patients. The other is our private dataset from 388 patients at Kyoto Second Red Cross Hospital. For both datasets, three medical experts attached the Mayo scores to ensure clean labels at the cost of annotation effort. The distribution of labels (Mayo 0, 1, 2, and 3) is 6,105, 3,052, 1,254, and 865 images on LIMUC and 6,678, 1,995, 1,395, and 197 images on the private dataset, respectively. All images in the two datasets were resized to 256 × 256 pixels.

Datasets with noisy labels. We created noisy-labeled datasets from the clean-labeled datasets with label perturbations for the quantitative evaluation. The perturbations are specified by the label transition matrix $P = [P_{ij}]$, where the $P_{ij} = \Pr(\tilde{y} = j | y = i)$ (i.e., probability of mistaking class i to class j). We employed two types of P. The first type is **Quasi-Gaussian**, which is an approximated version

Table 1. Classification results on LIMUC with Quasi-Gaussian noise. Following tradition, the test accuracy (Acc.), mean absolute error (MAE), and macro F1 (mF1)are averaged over the last ten epochs. The mean and standard deviations of five-fold cross-validation are shown. The best and second-best results are highlighted in red and blue, respectively. For plugin settings, improved results are shown by **bold**.

Method	Noise rate: $\epsilon = 0.2$			Noise rate: $\epsilon = 0.4$		
	Acc.↑	MAE↓	mF1↑	Acc.↑	MAE↓	mF1↑
Standard	0.673±0.013	0.383±0.020	0.572±0.011	0.556±0.017	0.564±0.025	0.463±0.019
Sord [5]	0.701±0.018	0.320±0.020	0.621±0.020	0.582±0.017	0.469±0.016	0.518±0.021
Label-smooth [23]	0.684±0.021	0.358±0.029	0.582±0.023	0.600±0.008	0.477±0.009	0.498±0.012
F-correction [25]	0.666±0.020	0.389±0.024	0.562±0.012	0.562±0.016	0.562±0.027	0.469±0.018
Reweight [18]	0.670±0.012	0.387±0.016	0.567±0.009	0.556±0.013	0.560±0.017	0.465±0.010
Mixup [9]	0.676±0.016	0.374±0.026	0.578±0.015	0.596±0.012	0.494±0.012	0.486±0.012
CDR [38]	0.664±0.016	0.399±0.024	0.560±0.013	0.550±0.005	0.585±0.009	0.454±0.010
Garg [7]	0.681±0.012	0.377±0.015	0.497±0.018	0.600±0.019	0.579±0.026	0.381±0.010
Co-teaching [8]	0.699±0.015	0.336±0.016	0.602±0.010	0.647±0.009	0.399±0.009	0.550±0.018
Co-teaching + Ours	**0.726±0.010**	**0.292±0.012**	**0.640±0.011**	**0.689±0.012**	**0.344±0.016**	**0.575±0.023**
JoCor [34]	0.718±0.010	0.310±0.011	0.624±0.005	0.679±0.016	0.363±0.019	0.536±0.077
JoCor + Ours	**0.733±0.007**	**0.286±0.008**	**0.644±0.010**	**0.702±0.007**	**0.328±0.012**	**0.595±0.023**
CoDis [36]	0.695±0.012	0.342±0.015	0.600±0.009	0.619±0.009	0.440±0.011	0.526±0.014
CoDis + Ours	**0.727±0.008**	**0.290±0.009**	**0.642±0.010**	**0.673±0.005**	**0.360±0.005**	**0.573±0.012**

Table 2. Classification results on our private UC dataset with Quasi-Gaussian noise. See the caption of Table 1 for details.

Method	Noise rate: $\epsilon = 0.2$			Noise rate: $\epsilon = 0.4$		
	Acc.↑	MAE↓	mF1↑	Acc.↑	MAE↓	mF1↑
Standard	0.756±0.009	0.291±0.008	0.530±0.015	0.640±0.022	0.474±0.023	0.436±0.016
Sord [5]	0.792±0.007	0.226±0.007	0.579±0.033	0.678±0.018	0.358±0.023	0.513±0.022
Label-smooth [23]	0.783±0.018	0.250±0.020	0.564±0.038	0.694±0.019	0.367±0.021	0.485±0.022
F-correction [25]	0.755±0.014	0.293±0.017	0.545±0.033	0.635±0.021	0.472±0.032	0.447±0.019
Reweight [18]	0.762±0.016	0.286±0.021	0.549±0.009	0.634±0.023	0.484±0.043	0.434±0.017
Mixup [9]	0.765±0.011	0.281±0.015	0.530±0.026	0.673±0.014	0.409±0.013	0.450±0.014
CDR [38]	0.752±0.004	0.295±0.007	0.535±0.006	0.631±0.017	0.488±0.024	0.443±0.019
Garg [7]	0.734±0.020	0.311±0.022	0.467±0.023	0.664±0.031	0.472±0.051	0.342±0.015
Co-teaching [8]	0.790±0.009	0.237±0.008	0.594±0.014	0.719±0.015	0.324±0.016	0.512±0.036
Co-teaching + Ours	**0.818±0.008**	**0.200±0.009**	**0.599±0.031**	**0.776±0.015**	**0.256±0.019**	**0.555±0.031**
JoCor [34]	0.812±0.005	0.209±0.008	0.601±0.018	0.787±0.006	0.242±0.005	0.547±0.047
JoCor + Ours	**0.819±0.006**	**0.197±0.005**	0.585±0.031	0.785±0.011	0.242±0.015	**0.558±0.040**
CoDis [36]	0.784±0.010	0.246±0.010	0.585±0.015	0.699±0.022	0.350±0.025	0.508±0.027
CoDis + Ours	**0.808±0.007**	**0.209±0.007**	**0.598±0.019**	**0.756±0.016**	**0.271±0.015**	**0.549±0.034**

of the Gaussian perturbation of Eq. (2). Specifically, by following Garg et al. [7], we set $P_{ij} = \frac{\rho}{|i-j|}, \forall i \neq j$ and $P_{ii} = 1 - \sum_{j \backslash i} P_{ij}$, $\forall i$. Here, ρ is a parameter for noise strength. The second type is **Truncated-Gaussian**, which mimics the case that experts do not make severe mistakes. Specifically, $P_{ij} = 1 - \rho$ for $|i-j| = 1$ and $P_{ij} = 0$ for $|i-j| > 1$. This means, for example, that a UC image with Mayo 1 is only mislabeled as Mayo 0 or 2 and not 3. The diagonal element P_{ii} is calculated by the same equation as Quasi-Gaussian.

Noise rate ϵ. The noise rate ϵ indicates the ratio of samples with incorrect labels in the dataset. According to the past attempts with noisy labels [8, 34, 36, 40], we use two rates, i.e., $\epsilon = 0.2$ (moderately noisy) and 0.4 (extremely noisy). For setting ϵ at 0.2 and 0.4, we specify ρ to appropriate values. For Quasi-Gaussian, we set ρ at 0.1 and 0.2 to have $\epsilon = 0.2$ and 0.4, respectively. Similarly, we set ρ at 0.15 and 0.3 for Truncated-Gaussian, respectively.

Real-world noisy dataset. We used the Diabetic Retinopathy (DR) dataset[1] as a real-world noisy dataset, where noisy labels naturally present. This dataset includes 35,108 retinal

[1] https://kaggle.com/competitions/diabetic-retinopathy-detection

Table 3. Classification results on our private UC dataset with Truncated-Gaussian noise. See the caption of Table 1 for details.

Method	Noise rate: $\epsilon = 0.2$			Noise rate: $\epsilon = 0.4$		
	Acc.↑	MAE↓	mF1↑	Acc.↑	MAE↓	mF1↑
Standard	0.760±0.010	0.270±0.010	0.559±0.012	0.629±0.028	0.418±0.027	0.423±0.038
Sord [5]	0.792±0.009	0.225±0.008	0.594±0.026	0.680±0.011	0.340±0.009	0.485±0.025
Label-smooth [23]	0.786±0.008	0.243±0.008	0.590±0.040	0.665±0.022	0.375±0.018	0.448±0.019
F-correction [25]	0.763±0.008	0.263±0.007	0.563±0.035	0.655±0.023	0.379±0.024	0.456±0.030
Reweight [18]	0.761±0.019	0.271±0.018	0.559±0.022	0.610±0.017	0.436±0.016	0.431±0.011
Mixup [9]	0.757±0.016	0.273±0.013	0.551±0.019	0.646±0.022	0.398±0.020	0.445±0.022
CDR [38]	0.758±0.014	0.272±0.012	0.574±0.015	0.620±0.017	0.430±0.012	0.412±0.019
Garg [7]	0.737±0.019	0.303±0.017	0.481±0.013	0.610±0.036	0.625±0.061	0.233±0.010
Co-teaching [8]	0.788±0.009	0.236±0.008	0.599±0.031	0.702±0.022	0.328±0.020	0.490±0.036
Co-teaching + Ours	**0.815±0.010**	**0.202±0.008**	**0.621±0.035**	**0.748±0.031**	**0.282±0.033**	**0.491±0.032**
JoCor [34]	0.812±0.011	0.209±0.011	0.621±0.032	0.744±0.032	0.286±0.031	0.474±0.028
JoCor + Ours	**0.817±0.007**	**0.199±0.007**	0.607±0.040	**0.760±0.027**	**0.266±0.025**	**0.493±0.014**
CoDis [36]	0.780±0.011	0.246±0.010	0.585±0.023	0.675±0.015	0.357±0.013	0.483±0.025
CoDis + Ours	**0.809±0.009**	**0.207±0.007**	**0.611±0.038**	**0.749±0.024**	**0.272±0.023**	**0.531±0.028**

images, classified into five ordinal levels of DR severity: No DR (lowest), Mild DR, Moderate DR, Severe DR, and Proliferative DR (highest). The sample sizes for these levels are 25,802, 2,438, 5,288, 872, and 708, respectively. For training the proposed method, we set the noise rate $\epsilon = 0.3$, as the DR dataset is estimated to contain approximately 30% noisy ordinal labels due to through annotation [12]. All images were resized to 256 × 256 pixels.

Comparative methods. We compare the proposed method with the traditional and state-of-the-art methods for learning with noisy labels: Co-teaching [8], JoCor [34], and CoDis [36]. In addition, we employ the recent methods that use robust loss functions, reweighting examples, and regularizations: F-correction [25], Reweight [18], Mixup [9], and CDR [38]. Moreover, we use Garg et al. [7], the only ordinal regression method with noisy labels. Furthermore, we employ a label smoothing method (Label-smooth) [23, 32]. As two simple baselines, we use Standard and Sord [5]. Both are just a single CNN without any sample selection, and the former is trained with the hard labels, whereas the latter is trained with the soft labels.

As noted in Section 3.6, we applied our framework to Co-teachig [8], JoCor [34], and CoDis [36]. We refer to the resulting method as “Co-teaching + Ours,” “JoCor + Ours,” and “CoDis + Ours.”

Evaluation metrics. We used accuracy and mean absolute error (MAE) for the test set to evaluate the classification performance. MAE is computed as the mean of the absolute differences between the predicted class labels and the true (and clean) class label. In addition, macro-F1 was also evaluated for evaluation metrics because of class imbalance in the two UC datasets and the DR dataset. Following the tradition of the papers on learning with noisy labels [8, 34, 36, 40], all metrics were calculated as averages over the last ten epochs.

We used five-fold cross-validation in all evaluations. Each clean-label dataset was split into training, validation, and test sets with 60, 20, and 20%, respectively, by random patient-disjoint sampling. Then, for two UC datasets, as noted above, we made the training and validation sets noisy by the label transition matrix P to have $\epsilon = 0.2$ or 0.4. This means we use the noisy validation set to simulate a practical situation where no clean dataset is available. Only the test set is clean for the meaningful evaluation. The methods with a dual-network architecture have two networks with slightly different accuracies. In the following results, we show the average of the performance of two networks.

Implementation details. We used ResNet-18 pretrained on ImageNet for all methods. We also used Adam as the optimizer and set the batch size to 64 for two UC datasets and 128 for the DR dataset. Training epochs were 150 in total, and an initial learning rate was 1×10^{-4}. The softmax temperature τ was set at 0.1. Following the traditional methods [8, 34, 36], T' was set at 5. For fair comparisons, L2 regularization of parameters and a multi-step learning rate scheduler were used to avoid overfitting noisy labels. We used random horizontal flipping and random cropping (224 × 224) for data augmentation in training. (Note that center-cropped 224 × 224 images were fed to the networks in the evaluation step.)

4.2. Quantitative Evaluation Results

Tables 1 and 2 show the quantitative evaluation results for LIMUC and our private UC dataset, respectively, where Quasi-Gaussian was used to make the datasets noisy. The proposed methods (“$*$ + Ours”) achieve the best or second-

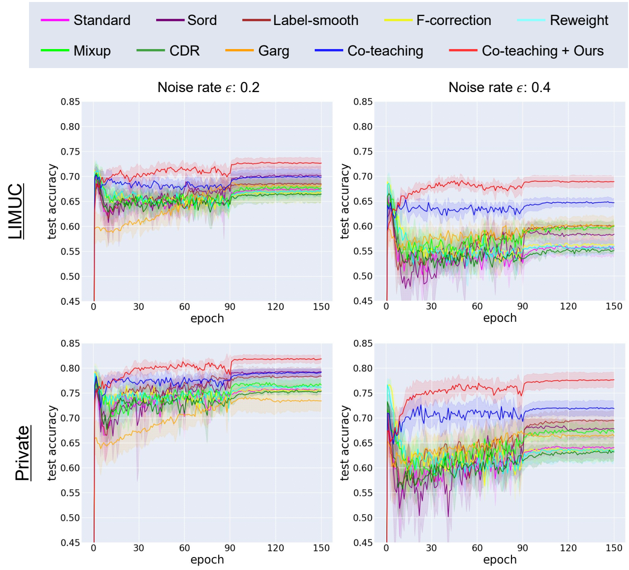


Figure 2. Test accuracy curves. The width of the shading indicates the standard deviation in cross-validation.

best results across all metrics in both datasets and at both noise rates. We can also confirm that our self-relaxed joint-training framework boosts the performance of traditional methods, i.e., Co-teaching, JoCor, and CoDis. Moreover, ours outperforms Garg et al. [7]; one possible reason is they do not assume Gaussian-like characteristics of ordinal noisy labels. From those results, we can conclude that introducing soft labels representing the distribution of true labels is useful for dealing with ordinal noisy labels.

Table 3 shows the results on our private dataset under Truncated-Gaussian noise. Our methods ("$*$+ Ours") still outperform the other methods like Table 2. Compared to Table 2, the individual accuracies in Table 3 are slightly lower. Under the same noise rate ϵ, Truncated-Gaussian shows slightly inferior results than Quasi-Gaussian. This is because Quasi-Gaussian contains "obviously incorrect" labels (such as the "Mayo 0" label for a "Mayo 2" sample) to some extent, and they are easily discarded. In contrast, Truncated-Gaussian contains only ambiguous samples with only one-level errors, and therefore, it was difficult for all methods to discard the samples.

4.3. Test accuracy curves

Fig. 2 shows the test accuracy curves for the individual methods on two UC datasets with Quasi-Gaussian noise. The comparative methods show a sharp increase in their test accuracy in very early epochs (< 5) by using a limited number of "very" clean samples to train their model. Then, the comparative models often start "memorizing" the samples with incorrect labels. The training accuracy is further improved through memorization, but the test accuracy degrades. This phenomenon is known as the memorization effect [3, 41] in learning with noisy labels. In contrast, our method ("Co-teaching + Ours") could avoid the memorization effect, and its test accuracy is roughly improved with the training steps. As shown in Section 4.4, our method could select clean samples much more than Co-teaching, even under the higher noise rate condition, and therefore, avoid the memorization effect more successfully than the other methods.

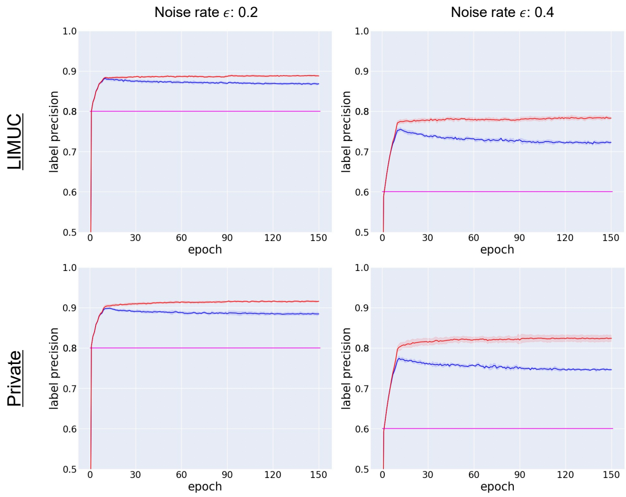


Figure 3. Label precision curves. The blue and red curves show the label precisions by "Co-teaching" and "Co-teaching + Ours," respectively. The pink horizontal line shows $(1 - \epsilon)$.

4.4. Label precision curve

Fig. 3 shows, on UC datasets with Quasi-Gaussian noise, the change in label precision, which evaluates the ratio of clean samples among all the selected samples. The backbone method is Co-teaching. The pink horizontal lines $(1 - \epsilon)$ indicate the label precision under random sample selection. The proposed method ("Co-teaching + Ours," the red curve) shows far better label precisions than random selection. In addition, we can see that the selection performance is stable and thus does not degrade with epochs. In contrast, Co-teaching (the blue curve) shows the degradation with epochs because the network updated by hard labels is not reliable enough for sample selection.

4.5. How good is our combination of hard and soft labels?

As shown in Fig. 1(b), our method uses the hard labels $\boldsymbol{l}_{\mathrm{h}}$ and their corresponding loss $\mathcal{L}_{\mathrm{h}}$ for sample selection and the soft labels $\boldsymbol{l}_{\mathrm{s}}$ and their loss $\mathcal{L}_{\mathrm{s}}$ for updating the model parameters. Tables 4 and 5 show how this combination is appropriate for learning with ordinal (Quasi-Gaussian) noisy labels of LIMUC and the private dataset, respectively. In these experiments, Co-teaching is used as the backbone method; therefore, the upper-most "hard & hard" case cor-

Table 4. Results of LIMUC dataset with Quasi-Gaussian noise under different loss usages for sample selection and updating. The best and second-best results are highlighted in red and blue, respectively.

Selection	Updating	Noise rate: $\epsilon = 0.2$			Noise rate: $\epsilon = 0.4$		
		Acc.↑	MAE↓	mF1↑	Acc.↑	MAE↓	mF1↑
hard	hard	0.699±0.015	0.336±0.016	0.602±0.010	0.647±0.009	0.399±0.009	0.550±0.018
soft	soft	0.726±0.010	0.293±0.011	0.633±0.014	0.677±0.011	0.367±0.013	0.530±0.009
hard	soft	0.726±0.010	0.292±0.012	0.640±0.011	0.689±0.012	0.344±0.016	0.575±0.023

Table 5. Results of private UC dataset with Quasi-Gaussian noise under different loss usages for sample selection and updating.

Selection	Updating	Noise rate: $\epsilon = 0.2$			Noise rate: $\epsilon = 0.4$		
		Acc.↑	MAE↓	mF1↑	Acc.↑	MAE↓	mF1↑
hard	hard	0.790±0.009	0.237±0.008	0.594±0.014	0.719±0.015	0.324±0.016	0.512±0.036
soft	soft	0.812±0.011	0.204±0.012	0.602±0.017	0.757±0.023	0.276±0.025	0.529±0.036
hard	soft	0.818±0.008	0.200±0.009	0.599±0.031	0.776±0.015	0.256±0.019	0.555±0.036

responds to the original Co-teaching [8] and the lower-most "hard & soft" to "Co-teaching + Ours."

The most important comparison with these tables is "hard & soft" (i.e., ours) vs "soft & soft." One might think that the consistent " soft & soft" is more reasonable than our inconsistent "hard & soft." However, this is not true. Using soft labels for sample selection is less suitable than using hard labels, especially for the private dataset.

4.6. Effectiveness in performance on a real noisy dataset

Table 6 shows the classification results on the DR dataset, a real-world noisy dataset. Our methods ("$*$ + Ours") achieve either the best or second-best results across all metrics. Additionally, the accuracy and MAE of traditional joint-training methods (i.e., Co-teaching, JoCor, and CoDis) are significantly worse than those of comparison methods. This is because traditional methods were designed for standard noisy labels. In contrast, our self-relaxed joint training framework boosts performance as our framework introduces modifications that allow these methods to handle ordinal noisy labels. From these results, we can conclude that the proposed method is effective for practical medical diagnosis tasks that contain ordinal noisy labels.

5. Conclusion

We proposed a self-relaxed joint-training framework to train a classifier with ordinal noisy labels. The proposed framework is based on the traditional joint-training framework with clean sample selection and dual-network architecture [8, 34, 36] and improved for learning with ordinal noisy labels. Specifically, our framework trains dual networks by soft labels, representing the distribution of true labels. In contrast, ours adheres to using original hard labels for sample selection. This combination of hard and soft labels was essential to improve sample selection accuracy and final classification accuracy. Using three medical image datasets, we confirmed that applying our framework to several state-of-the-art methods within the joint-training framework improved their performance and outperformed other methods for learning with noisy labels.

Table 6. Classification results on the DR dataset. The mean of five-fold cross-validation is shown. The best and second-best results are highlighted in red and blue, respectively. For plugin settings, improved results are shown by **bold**.

Method	Acc.↑	MAE↓	mF1↑
Standard	0.731	0.463	0.356
Sord [5]	0.744	0.428	0.373
Label-smooth [23]	0.742	0.445	0.356
F-correction [25]	0.729	0.457	0.366
Reweight [18]	0.726	0.465	0.369
Mixup [9]	0.723	0.477	0.377
CDR [38]	0.733	0.454	0.363
Garg [7]	0.729	0.498	0.247
Co-teaching [8]	0.536	0.746	0.362
Co-teaching + Ours	**0.737**	**0.434**	**0.388**
JoCor [34]	0.496	0.749	0.361
JoCor + Ours	**0.642**	**0.546**	**0.393**
CoDis [36]	0.690	0.518	0.367
CoDis + Ours	**0.747**	**0.421**	**0.384**

Future work will focus on the *class imbalance* problems on ordinal noisy labels. Medical image datasets tend to be class-imbalanced. Unfortunately, the methods for learning with noisy labels suffer from class imbalance [10, 13, 15]. Therefore, it will be helpful for our methods to integrate some techniques to mitigate class imbalance.

Acknowledgements. This work was supported by JSPS KAKENHI Grant Number JP22H05173, JP22H05180, JP24KJ1805.

References

[1] Xianze Ai, Zehui Liao, and Yong Xia. URL: Combating Label Noise for Lung Nodule Malignancy Grading. In *Data Augmentation, Labelling, and Imperfections*, pages 1–11, 2024. 3

[2] Görkem Algan and Ilkay Ulusoy. Image Classification with Deep Learning in the Presence of Noisy Labels: A Survey. *Knowledge-Based Systems*, 215:106771, 2021. 1

[3] Devansh Arpit, Stanisław Jastrzkebski, Nicolas Ballas, David Krueger, Emmanuel Bengio, Maxinder S. Kanwal, Tegan Maharaj, Asja Fischer, Aaron Courville, Yoshua Bengio, and Simon Lacoste-Julien. A Closer Look at Memorization in Deep Networks. In *International Conference on Machine Learning*, pages 233–242, 2017. 7

[4] Jan Chorowski and Navdeep Jaitly. Towards Better Decoding and Language Model Integration in Sequence to Sequence Models. *arXiv preprint arXiv:1612.02695*, 2016. 3

[5] Raul Diaz and Amit Marathe. Soft Labels for Ordinal Regression. In *Computer Vision and Pattern Recognition*, pages 4738–4747, 2019. 3, 4, 5, 6, 8

[6] Tongtong Fang, Nan Lu, Gang Niu, and Masashi Sugiyama. Rethinking Importance Weighting for Deep Learning under Distribution Shift. In *Advances in Neural Information Processing Systems*, volume 33, pages 11996–12007, 2020. 2

[7] Bhanu Garg and Naresh Manwani. Robust Deep Ordinal Regression under Label Noise. In *Asian Conference on Machine Learning*, pages 782–796, 2020. 3, 5, 6, 7, 8

[8] Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust Training of Deep Neural Networks with Extremely Noisy Labels. In *Advances in Neural Information Processing Systems*, pages 8535–8545, 2018. 1, 2, 3, 4, 5, 6, 8

[9] Hongyi Zhang and Moustapha Cisse and Yann N. Dauphin and David Lopez-Paz. Mixup: Beyond Empirical Risk Minimization. In *International Conference on Learning Representations*, 2018. 3, 5, 6, 8

[10] Yingsong Huang, Bing Bai, Shengwei Zhao, Kun Bai, and Fei Wang. Uncertainty-Aware Learning against Label Noise on Imbalanced Datasets. *AAAI Conference on Artificial Intelligence*, 36(6):6960–6969, 2022. 8

[11] Zhimeng Jiang, Kaixiong Zhou, Zirui Liu, Li Li, Rui Chen, Soo-Hyun Choi, and Xia Hu. An Information Fusion Approach to Learning with Instance-Dependent Label Noise. In *International Conference on Learning Representations*, 2022. 2

[12] Lie Ju, Xin Wang, Lin Wang, Dwarikanath Mahapatra, Xin Zhao, Quan Zhou, Tongliang Liu, and Zongyuan Ge. Improving Medical Images Classification With Label Noise Using Dual-Uncertainty Estimation. *IEEE Transactions on Medical Imaging*, 41(6):1533–1546, 2022. 3, 6

[13] Shyamgopal Karthik, Jérome Revaud, and Boris Chidlovskii. Learning from Long-Tailed Data with Noisy Labels. *arXiv preprint arXiv:2108.11096*, 2021. 8

[14] Youngdong Kim, Junho Yim, Juseung Yun, and Junmo Kim. NLNL: Negative Learning for Noisy Labels. In *International Conference on Computer Vision*, 2019. 3

[15] Jinpeng Li, Hanqun Cao, Jiaze Wang, Furui Liu, Qi Dou, Guangyong Chen, and Pheng-Ann Heng. Learning Robust Classifier for Imbalanced Medical Image Dataset with Noisy Labels by Minimizing Invariant Risk. In *Medical Image Computing and Computer Assisted Intervention*, pages 306–316, 2023. 8

[16] Wanhua Li, Xiaoke Huang, Jiwen Lu, Jianjiang Feng, and Jie Zhou. Learning Probabilistic Ordinal Embeddings for Uncertainty-Aware Regression. In *Computer Vision and Pattern Recognition*, pages 13896–13905, 2021. 3

[17] Xuefeng Liang, Xingyu Liu, and Longshan Yao. Review–A Survey of Learning from Noisy Labels. *ECS Sensors Plus*, 1(2):021401, 2022. 1

[18] Tongliang Liu and Dacheng Tao. Classification with Noisy Labels by Importance Reweighting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(3):447–461, 2016. 2, 5, 6, 8

[19] Xiaofeng Liu, Yang Zou, Yuhang Song, Chao Yang, Jane You, and B. V. K Vijaya Kumar. Ordinal Regression with Neuron Stick-breaking for Medical Diagnosis. In *European Conference on Computer Vision Workshops*, pages 0–0, 2018. 3

[20] Michal Lukasik, Srinadh Bhojanapalli, Aditya Menon, and Sanjiv Kumar. Does Label Smoothing Mitigate Label Noise ? In *International Conference on Machine Learning*, volume 119, pages 6448–6458, 2020. 3

[21] Xingjun Ma, Hanxun Huang, Yisen Wang, Simone Romano, Sarah Erfani, and James Bailey. Normalized Loss Functions for Deep Learning with Noisy Labels. In *International Conference on Machine Learning*, volume 119, pages 6543–6553, 2020. 2

[22] Eran Malach and Shai Shalev-Shwartz. Decoupling "When to Update" from "How to Update". In *Advances in Neural Information Processing Systems*, page 961–971, 2017. 2

[23] Rafael Müller, Simon Kornblith, and Geoffrey E Hinton. When Does Label Smoothing Help? In *Advances in Neural Information Processing Systems*, volume 32, 2019. 3, 5, 6, 8

[24] Zhenxing Niu, Mo Zhou, Le Wang, Xinbo Gao, and Gang Hua. Ordinal Regression With Multiple Output CNN for Age Estimation. In *Computer Vision and Pattern Recognition*, pages 4920–4928, 2016. 3

[25] Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making Deep Neural Networks Robust to Label Noise: A Loss Correction Approach. In *Computer Vision and Pattern Recognition*, pages 1944–1952, 2017. 2, 5, 6, 8

[26] Gorkem Polat, Haluk Tarik Kani, Ilkay Ergenc, Yesim Ozen Alahdab, Alptekin Temizel, and Ozlen Atug. Improving the Computer-Aided Estimation of Ulcerative Colitis Severity According to Mayo Endoscopic Score by Using Regression-Based Deep Learning. *Inflammatory Bowel Diseases*, 29(9):1431–1439, 2022. 1, 4

[27] Esteban Real, Alok Aggarwal, Yanping Huang, and Quoc V. Le. Regularized Evolution for Image Classifier Architecture Search. *AAAI Conference on Artificial Intelligence*, 33(01):4780–4789, 2019. 3

[28] Mengye Ren, Wenyuan Zeng, Bin Yang, and Raquel Urtasun. Learning to Reweight Examples for Robust Deep Learning. In *International Conference on Machine Learning*, volume 80, pages 4334–4343, 2018. 2

[29] Kenneth W Schroeder, William J Tremaine, and Duane M Ilstrup. Coated Oral 5-Aminosalicylic Acid Therapy for Mildly to Moderately Active Ulcerative Colitis. *New England Journal of Medicine*, 317(26):1625–1629, 1987. 1

[30] Nyeong-Ho Shin, Seon-Ho Lee, and Chang-Su Kim. Moving Window Regression: A Novel Approach to Ordinal Regression. In *Computer Vision and Pattern Recognition*, pages 18760–18769, 2022. 3

[31] Hwanjun Song, Minseok Kim, Dongmin Park, Yooju Shin, and Jae-Gil Lee. Learning From Noisy Labels With Deep Neural Networks: A Survey. *IEEE Transactions on Neural Networks and Learning Systems*, 34(11):8135–8153, 2023. 1, 2

[32] Christian Szegedy, Vincent Vanhoucke, Sergey Ioffe, Jon Shlens, and Zbigniew Wojna. Rethinking the Inception Architecture for Computer Vision. In *Computer Vision and Pattern Recognition*, pages 2818–2826, 2016. 3, 6

[33] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is All you Need. In *Advances in Neural Information Processing Systems*, volume 30, 2017. 3

[34] Hongxin Wei, Lei Feng, Xiangyu Chen, and Bo An. Combating Noisy Labels by Agreement: A Joint Training Method with Co-Regularization. In *Computer Vision and Pattern Recognition*, pages 13726–13735, 2020. 1, 2, 3, 4, 5, 6, 8

[35] Jiaheng Wei, Hangyu Liu, Tongliang Liu, Gang Niu, Masashi Sugiyama, and Yang Liu. To Smooth or Not? When Label Smoothing Meets Noisy Labels. In *International Conference on Machine Learning*, volume 162, pages 23589–23614, 2022. 3

[36] Xiaobo Xia, Bo Han, Yibing Zhan, Jun Yu, Mingming Gong, Chen Gong, and Tongliang Liu. Combating Noisy Labels with Sample Selection by Mining High-Discrepancy Examples. In *International Conference on Computer Vision*, pages 1833–1843, 2023. 1, 2, 3, 4, 5, 6, 8

[37] Xiaobo Xia, Tongliang Liu, Bo Han, Nannan Wang, Mingming Gong, Haifeng Liu, Gang Niu, Dacheng Tao, and Masashi Sugiyama. Part-dependent Label Noise: Towards Instance-dependent Label Noise. In *Advances in Neural Information Processing Systems*, volume 33, pages 7597–7610. Curran Associates, Inc., 2020. 2

[38] Xiaobo Xia and Tongliang Liu and Bo Han and Chen Gong and Nannan Wang and Zongyuan Ge and Yi Chang. Robust Early-learning: Hindering the Memorization of Noisy Labels. In *International Conference on Learning Representations*, 2021. 3, 5, 6, 8

[39] Yilun Xu, Peng Cao, Yuqing Kong, and Yizhou Wang. L_DMI: A Novel Information-theoretic Loss Function for Training Deep Nets Robust to Label Noise. In *Advances in Neural Information Processing Systems*, volume 32, 2019. 2

[40] Xingrui Yu, Bo Han, Jiangchao Yao, Gang Niu, Ivor Tsang, and Masashi Sugiyama. How Does Disagreement Help Generalization against Label Corruption? In *International Conference on Machine Learning*, pages 7164–7173, 2019. 2, 5, 6

[41] Chiyuan Zhang, Samy Bengio, Moritz Hardt, Benjamin Recht, and Oriol Vinyals. Understanding Deep Learning Requires Rethinking Generalization. In *International Conference on Learning Representations*, 2017. 7

[42] Zhilu Zhang and Mert Sabuncu. Generalized Cross Entropy Loss for Training Deep Neural Networks with Noisy Labels. In *Advances in Neural Information Processing Systems*, volume 31, 2018. 2

[43] Barret Zoph, Vijay Vasudevan, Jonathon Shlens, and Quoc V. Le. Learning Transferable Architectures for Scalable Image Recognition. In *Conference on Computer Vision and Pattern Recognition*, pages 8697–8710, 2018. 3