Regularization Penalty Optimization for Addressing Data Quality Variance in OoD Algorithms

Runpeng Yu,[1,*] Hong Zhu,[2,*] Kaican Li,[2] Lanqing Hong,[2] Rui Zhang,[3,†] Nanyang Ye,[4,†] Shao-Lun Huang,[1] Xiuqiang He[2]

[1] Tsinghua-Berkeley Shenzhen Institute, Tsinghua University
[2] Huawei Noah's Ark Lab
[3] www.ruizhang.info
[4] Shanghai Jiao Tong University
yrp19@mails.tsinghua.edu.cn, zhuhong8@huawei.com, mjust.lkc@gmail.com, honglanqin@huawei.com, rayteam@yeah.net, ynylincoln@sjtu.edu.cn, shaolun.huang@sz.tsinghua.edu.cn, hexiuqiang1@huawei.com

Abstract

Due to the poor generalization performance of traditional empirical risk minimization (ERM) in the case of distributional shift, Out-of-Distribution (OoD) generalization algorithms receive increasing attention. However, OoD generalization algorithms overlook the great variance in the quality of training data, which significantly compromises the accuracy of these methods. In this paper, we theoretically reveal the relationship between training data quality and algorithm performance and analyze the optimal regularization scheme for Lipschitz regularized invariant risk minimization. A novel algorithm is proposed based on the theoretical results to alleviate the influence of low-quality data at both the sample level and the domain level. The experiments on both the regression and classification benchmarks validate the effectiveness of our method with statistical significance.

1 Introduction

Traditional empirical risk minimization (Vapnik 1998) focuses on in-distribution generalization under the assumption that training and testing data are independent and identically distributed (i.i.d.). In practice, however, it is common that the distribution of the training and testing data differ greatly, which has motivated the research of Out-of-Distribution (OoD) generalization (Gulrajani and Lopez-Paz 2021; Arjovsky 2020; Li et al. 2018b; Yan et al. 2020; Arjovsky et al. 2019). In this work, we identify a common but often overlooked problem under the setting of OoD generalization – the varying quality of training data among samples (or instances) and across *domains*[1] – which could severely hurt model performance in some cases. We focus on two types of quality of training data: the density of data and the noise (or error[2]) rate of data. Training data in the ranges of low density or with high noise rate are deemed as low-quality data. The variance in data quality mainly exists at two levels, the sample level and the domain level, and we need to address the problem at both levels.

Our intuition to address the data quality problem is that, when applying any OoD generalization algorithm, low-quality data should be detected and regularized (penalized) differently from normal-quality data. The main challenge is how to properly regularize each sample and domain since the numbers of samples and domains are usually quite large. We base our method on one of the most influential OoD generalization paradigms: invariant risk minimization (IRM) (Arjovsky et al. 2019). To overcome the aforementioned challenge, we propose Lipschitz regularized IRM (LipIRM) loss with varied regularization coefficients on different samples and domains. We then theoretically reveal the links among algorithm performance, data quality (in terms of the density of the data and variance of the noise) and regularization. Based on the theoretical analysis, we propose a regularization scheme that is optimal in the sense that it minimizes the distance between the estimated model and the ground truth model (measured by Eq. 3).

Our contributions are summarized as follows: (i) We identify the performance degeneration of the OoD generalization algorithms when trained on data of varied quality, and we provide the first solution to this problem. (ii) We theoretically derive the optimal regularization scheme for IRM with the Lipschitz regularizer. (iii) Based on the theoretical analysis, we propose a novel algorithm named **R**egularization **P**enalty **O**ptimization (RPO) to address the data quality variance problem, which is via sample-wise and domain-wise regularization. (iv) We conduct extensive experiments on publicly available real-world datasets, and the results show that RPO outperforms the state-of-the-art algorithms.

2 Preliminaries

OoD Generalization. In the setting of OoD generalization, a predictor (e.g., neural network) is trained on multiple training datasets generated from different domains and is evaluated on unseen testing domains (Gulrajani and Lopez-Paz 2021). Let $\mathcal{E}_{all}$ and $\mathcal{E}_{tr} \subset \mathcal{E}_{all}$ denote the set of

[*] These authors contributed equally.
[†] Corresponding author.

[1] A domain is a data source or a data group via artificial data segmentation based on prior knowledge of data distribution (Sagawa et al. 2019; Arjovsky et al. 2019)

[2] In the rest of the paper, we omit "error" and error is implied when we say "noise".

all possible domains and the set of the domains generating the available training datasets, respectively. Let $D^e \triangleq \{(x_e^{(i)}, y_e^{(i)})\}_{i=1}^{N_e}$ denotes the dataset generated from the domain e, where $e \in \mathcal{E}_{all}$, $x_e^{(i)} \in \mathcal{X}$ is the features of the i-th sample, $y_e^{(i)} \in \mathcal{Y}$ is the corresponding label and N_e is the sample size of domain e. Given the training datasets $D_{tr} \triangleq \{D^e \mid e \in \mathcal{E}_{tr}\}$, the goal of the OoD problem is to minimize the maximum risk across all the domains, namely to minimize $\max_{e \in \mathcal{E}_{all}} R^e(f)$, where $f \in \mathcal{F} : \mathcal{X} \to \mathcal{Y}$ is the predictor, and $R^e(f)$ is the risk evaluated in domain e.

IRM. Under the framework of IRM, the predictor f is decomposed into a concatenation of a representation function $\Phi : \mathcal{X} \to \mathcal{H}$ and a classifier $w : \mathcal{H} \to \mathcal{Y}$, i.e. $f = w \circ \Phi$. To guarantee the existence of the unified optimal predictor over $\mathcal{E}_{all}$, IRM utilizes the invariant condition (Ahuja et al. 2021), which assumes that there exists a data representation Φ_* inducing an invariant predictor $w_* \circ \Phi_*$, where the classifier w_* is optimal over all training domains, simultaneously (i.e. $\forall e \in \mathcal{E}_{tr}, w_* \in \arg\min_{\bar{w}:\mathcal{H}\to\mathcal{Y}} R^e(\bar{w} \circ \Phi_*)$). It searches for such invariant predictor $w \circ \Phi$ by minimizing the empirical risk with an extra constraint on the unified optimality of w, that is

$$\min_{\substack{\Phi:\mathcal{X}\to\mathcal{H} \\ w:\mathcal{H}\to\mathcal{Y}}} \sum_{e\in\mathcal{E}_{tr}} R^e(w \circ \Phi),$$
$$s.t. \quad w \in \arg\min_{\bar{w}:\mathcal{H}\to\mathcal{Y}} R^e(\bar{w} \circ \Phi_*), \forall e \in \mathcal{E}_{tr}.$$

In order to solve the above bi-leveled optimization problem, IRM utilizes the augmented Lagrangian method to transfer the constraint to a gradient regularization on each domain e. At last, the object function of IRM is expressed as

$$\min_{\substack{\Phi:\mathcal{X}\to\mathcal{H} \\ w:\mathcal{H}\to\mathcal{Y}}} \sum_{e\in\mathcal{E}_{tr}} R^e(\Phi) + \eta \|\nabla_{w|w=1.0} R^e(w\Phi)\|_2^2,$$

where the classifier w reduces to a scalar and fixed "dummy" classifier; Φ becomes the entire invariant predictor; η is the unified IRM regularization coefficient, which, later in our setting, will vary across domains and serve as the tool of downweighting the domains with low-quality data.

Lipschitz regularization. Lipschitz regularization is first used in the statistical regression problems (Wang, Du, and Shen 2013) and recently proved to have a better generalization guarantee (Wei and Ma 2019, 2020) and be a sufficient condition for the smoothness of the representation function (Shui, Wang, and Gagné 2021) in the deep learning context. It penalizes the gradient of the output of the model corresponding to input features. Moreover, the strength of the penalties can be designed individually for each sample. So, we can assign large Lipschitz regularization penalties to the low-quality data to prevent the model from fitting on them. By doing this, the negative influence of the low-quality data is mitigated by the Lipschitz regularization. Comparatively, the techniques which punish uniformly on the entire model, such as l_n norm regularization, cannot filter out the low-quality data and thus lacks the robustness against the low-quality data.

Symbol	Description
$f_*(x)$	The ground true function projecting from $\mathcal{X}$ to $\mathcal{Y}$.
$\hat{f}(x)$	The estimation of the ground true function.
$\mathcal{R}(\hat{f})$	The expected MSE of the $\hat{f}$ used to evaluate the goodness of $\hat{f}$.
η_e	The penalty coefficient before the IRM regularization of domain e.
λ	The scale of the penalty coefficient before the Lipschitz regularization.
$\rho(x)$	The relative value of penalty coefficient before the Lipschitz regularization for input feature x.
$\hat{r}_e(x)$	The empirical probability density of input feature x in domain e.
$r_e(x)$	The true probability density of x in domain e.
$r(x)$	The summation of the true density of x over training domains.
$\sigma_e^2(x)$	The variance of the noise of input x in domain e.
$\hat{r}_{e,k}$	The groupwise mean of $\hat{r}_e(x)$ in the k-th group.
$\sigma_{e,k}^2$	The groupwise mean of $\sigma_e^2(x)$ in the k-th group.
N_e	The sample size of domain e.

Table 1: Frequently Used Symbols

3 Our Method

As discussed in Section 1, data quality variance exists at two levels, the sample-level and the domain-level, so we propose a Lipschitz regularized IRM (LipIRM) loss to realize the different regularization at each level. Our key insight is that the second term of the IRM objective function may be regarded as a domain-level regularization to the ERM loss, and Lipschitz regularization may further add different sample-level penalties to different samples. Formally, our proposed LipIRM loss is

$$L(f, X, Y) = \underbrace{\sum_{e\in\mathcal{E}_{tr}} R^e(\Phi)}_{\text{ERM}} + \underbrace{\sum_{e\in\mathcal{E}_{tr}} \eta_e \|\nabla_{w|w=1.0} R^e(w\Phi)\|_2^2}_{\text{IRM regularization}}$$
$$\underbrace{+ \lambda \int_{\mathcal{X}} \rho(x)[f'(x)]^2 dx}_{\text{Lipschitz regularization}}, \tag{1}$$

where λ is a hyperparameter for the Lipschitz regularization on the whole set of samples, η_e is a hyperparameter for the IRM regularization in domain e, and $\rho(x)$ is a hyperparameter for the Lipschitz regularization on each sample x. The LipIRM loss consists of three terms, the ERM term, the IRM regularization term (the above-mentioned domain-level regularization), and the Lipschitz regularization term (the above-mentioned sample-level regularization). We call these two regularization terms collectively the *LipIRM regularization*. Note that, the larger the η_e, the more the model fits the training samples in domain e; in contrast, the larger the $\rho(x)$, the less the model fits the sample x. Therefore, if a domain has low data quality, η_e should be made small, and if a sample has low quality, $\rho(x)$ should be made large. Unlike the usual approach of grid search to obtain the values for

these hyperparameters, we will theoretically derive the optimal values for them below. Table 1 summarizes frequently used symbols. When there is no ambiguity, we simplify the notation of function by removing its argument. For example, sometimes we may abbreviate $f(x)$ to f.

The estimated optimal model $\hat{f}$ (e.g. a well-trained neural network) is obtained by minimizing the loss in Eq. (1), i.e.,

$$\hat{f} = \arg\min_{f \in \mathcal{F}} L(f, X, Y). \tag{2}$$

We assume there is an underlying ground truth $f_* \in \mathcal{F}$, and for each sample $(x_e^{(i)}, y_e^{(i)})$ in domain e, label $y_e^{(i)}$ equals $f_*(x_e^{(i)})$ plus a relatively small noise $\epsilon_e(x_e^{(i)})$, i.e. $y_e^{(i)} = f_*(x_e^{(i)}) + \epsilon_e(x_e^{(i)})$, and the mean of the noise $\epsilon_e(x_e^{(i)})$ is zero. Denote the variance of $\epsilon_e(x_e^{(i)})$ by $\sigma_e^2(x_e^{(i)})$, which varies across different domains and samples. To measure the goodness of $\hat{f}$, the expectation of the mean square error (expected MSE for short) is used to calculate the distance between $\hat{f}$ and f_*, which is defined as

$$\mathcal{R}(\hat{f}) = \mathbb{E}\big[\int_{\mathcal{X}} [\hat{f}(x) - f_*(x)]^2 dx\big], \tag{3}$$

where the expectation is taken over the noise $\{\epsilon_e \mid e \in \mathcal{E}_{tr}\}$. Finding the optimal $\hat{f}$ is equivalent to minimizing $\mathcal{R}(\hat{f})$. As our analysis will show later, deriving the optimal solution for generic cases is difficult, because deriving the explicit expression of $\mathcal{R}(\hat{f})$ for high-dimensional function $\hat{f}$ involves solving an intractable system of partial differential equations. Therefore, we first focus our theoretical analysis on a simple case of one-dimensional regression problem, where $\mathcal{X} = [0, 1]$ and $R^e(f)$ is the mean square loss. Under this setting, with mild assumptions, the analytical form of $\mathcal{R}(\hat{f})$ can be obtained as given by Theorem 1 below. We discuss in the end of Section 3.2 how our method may be extended to more generic cases, and our extensive experimental study validates the effectiveness of our method in generic cases.

3.1 Closed-Form Relationship Among $\mathcal{R}(\hat{f})$, Data Quality and LipIRM Regularization

We show in Theorem 1 that in the simple case of one-dimensional regression problem, $\mathcal{R}(\hat{f})$ can be expressed as a closed-form formula of the quality of the training data and the LipIRM regularization. Since the training data are given and the statistics representing the data quality can be estimated from the training data, the theorem can guide us to adjust the regularization hyperparameters according to the estimated data quality so that $\mathcal{R}(\hat{f})$ is minimized.

Before presenting the theorem, we introduce some notations. Let $\delta(\cdot)$ denote the Dirac delta function. For each domain $e \in \mathcal{E}_{tr}$, let function $\hat{r}_e(x) \triangleq \frac{1}{N_e}\sum_{i=1}^{N_e} \delta(x - x_e^{(i)})$ represents the empirical density of x in domain e, and function $r_e(x)$ represent the true density of x in domain e. Let $r(x) \triangleq \sum_{e \in \mathcal{E}_{tr}} r_e(x)$ denote the summation of the true density at x.

Further, we make the following assumptions: (i) $\rho(x)$ is chosen to satisfy that $\forall x \in [0, 1]$, $\rho(x)$ is positive, and $\rho(x)$ is an analytic function; (ii) $\forall x \in [0, 1]$, $r(x)$ is positive and analytic; (iii) f_* is third-order differentiable, which usually holds in mainstream models; (iv) $\lambda \ll 1$, which holds since the regularization term usually has a weight less than 0.1; (v) the noise $\epsilon_e(x)$ is small compared with $f_*(x)$. That is, let us decompose $\epsilon_e(x)$ as $\epsilon_e(x) = \tau_e(x)\upsilon$, where υ is a random variable with $\mathbb{E}[\upsilon] = 0$ and $\mathrm{Var}[\upsilon] = 1$, $\tau_e(x)$ is a deterministic function. Then $\tau_e(x)$ should satisfy $\sup_{x\in[0,1]} \left|\frac{\tau_e(x)}{f_*(x)}\right| \ll 1$.

Theorem 1 (Evaluation of the expected MSE $\mathcal{R}(\hat{f})$)**.**

$$\mathcal{R}(\hat{f}) = \int_0^1 \big[\mathbb{E}^2[\hat{f}(t) - f_*(t)] + \mathrm{Var}[\hat{f}(t) - f_*(t)]\big] dt,$$

where

$$\mathbb{E}^2[\hat{f}(x) - f_*(x)] = \lambda^2 \left[\frac{[\rho(x) f_*'(x)]'}{r(x)} - \sum_{e \in \mathcal{E}_{tr}} \frac{\hat{r}_e(x) f_*(x)}{4\eta_e r(x) \int_0^1 \frac{\hat{r}_e^3(t)}{r^2(t)} f_*^2(t) dt}\right]^2,$$

$$\mathrm{Var}[\hat{f}(x) - f_*(x)] = \frac{1}{\sqrt{\lambda}} \sum_{e \in \mathcal{E}_{tr}} \frac{\hat{r}_e(x)\sigma_e^2(x)}{N_e r(x)\sqrt{r(x)\rho(x)}}.$$

The crux of proving Theorem 1 is the evaluation of $\hat{f}$, which is achieved as follows. Eq. (2) formulates an optimization problem on the functional space, without limiting the analysis to a specific parameterized model. Based on the functional analysis, we derive the sufficient and necessary condition for $\hat{f}$, which is an equation including high order derivative of $\hat{f}$. We then convert the evaluation of $\hat{f}$ to a boundary value problem, which largely involves solving complicated differential equations. To solve these complicated partial differential equations, we utilize the technique of Green functions (Stone and Goldbart 2009). After obtaining $\hat{f}$, substituting it into Eq. (3) completes the proof. See Appendix B.1-B.6 for the detailed derivation.

Based on Theorem 1, we derive the LipIRM regularization scheme that minimizes $\mathcal{R}(\hat{f})$, which we call *the optimal LipIRM regularization scheme*, in Section 3.2.

3.2 Optimal LipIRM Regularization

We derive the values of the hyperparameters λ, η_e, and ρ for the optimal LipIRM regularization. First, we find the optimal value for λ by computing the partial derivative of $\mathcal{R}(\hat{f})$ in Theorem 1 over λ. We obtain that $\lambda = (\sum_{e \in \mathcal{E}_{tr}} \frac{1}{N_e})^{\frac{2}{5}}$. See details in Appendix B.7.

Next, we find optimal values of ρ and η_e. Instead of finding the optimal continuous function ρ^*, we parameterize ρ as a group-wise constant function, then find the optimal group-wise ρ. More specifically, suppose the support of x can be divided into K small groups, $[x_0, x_1), [x_1, x_2), \cdots, [x_{K-1}, x_K)$. For the k-th group, $k = 1, \cdots, K$, we use the constant ρ_k as the Lipschitz regularization penalty for all of the x in this group. We further constrain that the data samples belonging to the same group are

in the same domain. It turns out that the optimal η_e and ρ are related to the variance of the noise $\sigma_e^2(x)$ and the empirical density $\hat{r}_e(x)$, for $x \in \mathcal{X}$ and $e \in \mathcal{E}_{tr}$. Instead of estimating $\sigma_e^2(x)$ and $\hat{r}_e(x)$ for each $x \in \mathcal{X}$ and $e \in \mathcal{E}_{tr}$, we also use the group-wise average to approximate them. For all of the samples in domain e and the k-th group, let $\sigma_{e,k}^2$ denote their average variance of the noise, and let $\hat{r}_{e,k}$ denote their average empirical density. The following proposition provides the estimation of the optimal η_e and ρ. The main idea of the proof is to let the derivative of $\mathcal{R}(\hat{f})$ be zero corresponding to η_e and ρ. See Appendix B.8 for the details.

Proposition 1. *For domain $e \in \mathcal{E}_{tr}$ and group $k = 1, \cdots, K$, the optimal ρ_k^* and optimal η_e^* are*

$$\rho_k^* = \frac{1}{4^{\frac{2}{5}}} \sum_{e \in \mathcal{E}_{tr}} \left(\frac{\sigma_{e,k}}{\hat{r}_{e,k}} \right)^{\frac{4}{5}} \mathbb{1}_{e,k},$$

$$\eta_e^* = \frac{N_e}{4^{\frac{7}{5}}} \left[\sum_{k=1}^{K} \left(\frac{\sigma_{e,k}}{\hat{r}_{e,k}} \right)^{\frac{4}{5}} \mathbb{1}_{e,k} \right]^{-1},$$

where the indicator $\mathbb{1}_{e,k} = 1$ if the k-th group contains the data samples in domain e, and $\mathbb{1}_{e,k} = 0$ otherwise.

The optimal ρ_k^* is proportional to $\sigma_{e,k}^{\frac{4}{5}}$ and $\hat{r}_{e,k}^{-\frac{4}{5}}$, which indicates that we should fit more on the data samples that have high density and less noise. The optimal η_e^* is proportional to N_e, which indicates that LipIRM should fit more on the domain with sufficient data samples. Another factor in η_e^* is the reciprocal of the weighted sum of $\hat{r}_{e,k}^{-\frac{4}{5}}$, whose weight is an exponent of the variance of the noise $\sigma_{e,k}^{\frac{4}{5}}$. Then we have the following observations for η_e^*: (i) In domain e, if the variance of the noise is a constant, the more density variance e has, the smaller η_e^* is; (ii) In domain e, the greater the variance of the noise is, the smaller η_e^* is; (iii) In domain e, if the groups of greater noise have smaller density, η_e^* will be small. In summary, we infer that LipIRM should emphasize fitting the domains that have small density variance and low noise rate.

Extension to Generic Cases. So far, our analysis and derivations have been based on the simple case of one-dimensional regression problem, where $\mathcal{X} = [0, 1]$ and $R^e(f)$ is the mean square loss. For more generic cases of high-dimensional data or classification tasks, we may not get nice closed-form expressions like Proposition 1, but the insights provided by Proposition 1 still hold, i.e., for training data of low quality, η_e should be set small and ρ_k should be set large. Thus the expressions in Proposition 1 may still be reasonable choices of the optimal penalties. In the experimental study, all the datasets are of high-dimensional data. We have used the method in Section 3.2 for setting the values of the hyperparameters, and our algorithm performs very well, outperforming state-of-the-art algorithms significantly. The experiments in Section 4.4 are for classification tasks, and our algorithm also performs very well. These validate the effectiveness of our method for generic cases.

Finally, the empirical LipIRM loss is written as

$$\begin{aligned} \hat{L}(f, D_{tr}) = & \sum_{e \in \mathcal{E}_{tr}} \sum_{i=1}^{N_e} l(f(x_e^{(i)}), y_e^{(i)}) \\ & + \sum_{e \in \mathcal{E}_{tr}} \eta_e^* \| \nabla_{w|w=1.0} [\sum_{i=1}^{N_e} l(wf(x_e^{(i)}), y_e^{(i)})] \|_2^2 \\ & + \lambda \sum_{e \in \mathcal{E}} \sum_{i=1}^{N_e} \rho^*(x_e^{(i)}) [f'(x_e^{(i)})]^2, \end{aligned} \tag{4}$$

where $l(\cdot, \cdot)$ is the loss function over a sample, ρ_k^* and optimal η_e^* are the optimal penalties obtained in Proposition 1, $\rho^*(x_e^{(i)}) = \rho_k^*$ if sample $(x_e^{(i)}, y_e^{(i)})$ is in group k.

Discussions on Implementation. There are two steps left before using the expressions in Proposition 1: (i) grouping the data and (ii) estimating the statistics ($\hat{r}_{e,k}$ and $\sigma_{e,k}^2$). Note that we should not group the data arbitrarily. For example, the naive way of evenly dividing after randomly shuffling the data will make all the groups have similar frequencies and noise rates, and hence fails to reflect the data quality variance in the original data. The right way to group the data should guarantee that the samples in the same group are close in the feature space and possess similar statistical properties. This kind of grouping can be treated as a clustering problem with constraints on the statistical similarity of the data in each cluster, and we can solve it by the axis parallel subspace clustering algorithm (Kriegel, Kröger, and Zimek 2009). Specifically, we first select several dominating features which can be regarded as the main causes of the statistical heterogeneity and then cluster the samples in the subspace of these features. The dominated features can be identified by human experts if the dataset has explicit features. In the task of image classification where there are no explicit features, the class label can be used as the dominated feature, e.g. the samples with the same digit in the Colored-MNIST or the same type of the animal (or vehicle) in NICO are grouped into the same group.

For estimating the statistics, (i) $\hat{r}_{e,k}$ can be estimated from the empirical density (i.e. $\hat{r}_{e,k} = \frac{N_{e,k}}{N_e}$, where $N_{e,k}$ is the number of samples in domain e and the k-th group). (ii) Because the variance can be rewritten as $\sigma_e^2(x) = \mathbb{E}\big[[\epsilon_e(x)]^2\big] = \mathbb{E}\big[[y_e(x) - f_*(x)]^2\big]$, where $y_e(x) = f_*(x) + \epsilon_e(x)$ is the observed label, and the expectation is taken over the random noise $\epsilon_e(x)$. $f_*(x)$ is indeed unknown, so we propose to replace it with the estimation $\tilde{f}(x)$ and approximate the variance via $\sigma_e^2(x) \approx \mathbb{E}\big[[y_e(x) - \tilde{f}(x)]^2\big]$. Therefore, we first train an extra model $\tilde{f}$ and then use the average prediction error of $\tilde{f}$ over the samples in the domain e and group k to approximate $\sigma_{e,k}^2$. Our algorithm is summarized in Algorithm 1.

4 Experiments

We experimentally evaluate the performance of our proposed algorithm RPO on four datasets, two (Cigar and Wage) for regression and two (Colored-MNIST and NICO)

Algorithm 1: Regularization Penalty Optimization (RPO)

Require: The training datasets $D_{tr} = \{D^e \mid e \in \mathcal{E}_{tr}\}$ and a parameterized model f_θ where θ is the parameters.

1: Randomly initialize the model parameters: $\theta \leftarrow \theta_0'$.
2: For all e and k, $\rho_k \leftarrow 1$, $\eta_e \leftarrow 1$
3: $f_{\theta_*'} \leftarrow$ SGD with LipIRM loss in Eq. (4) on D_{tr}
4: For all e and k, estimate $\hat{r}_{e,k}$, $\sigma^2_{e,k}$
5: For all e and k, compute ρ_k^* and η_e^* using Proposition 1
6: Randomly reinitialize the model parameters: $\theta \leftarrow \theta_0$.
7: $f_{\theta_*} \leftarrow$ SGD with LipIRM loss in Eq. (4) on D_{tr}

for classification. Following existing work on these tasks, all the results are averaged over multiple runs under the same setup with different random seeds. For each run, ρ_i and η_e in RPO are calculated according to Algorithm 1. Other hyperparameters in RPO such as learning rates and batch size and the hyperparameters in baselines are optimized by grid research on an independently divided validation set.

4.1 Baselines

We compared with a large number of baselines and listed blow. The details on them are provided in Appendix C.1. We evaluate all the 19 baselines for classification tasks. However, because most of the baselines are designed only for classification, we evaluate those that can also perform regression (6 of them) for regression tasks.

ERM. For completeness, we add the ERM algorithm into our baselines to indicate the rationality of the OoD method and the Lipschitz regularizer. We implement two types of ERM loss, one with a l_2-norm regularizer (ERM + l_2), and the other with a uniform Lipschitz regularizer (ERM + lip).

OoD generalization methods. We compare our methods with the following OoD generalization methods: ANDMask (Parascandolo et al. 2020), CDANN (Li et al. 2018b), CORAL (Sun and Saenko 2016), DANN (Ganin et al. 2016), GroupDRO (Sagawa et al. 2019), IGA (Koyama and Yamaguchi 2020), and the original IRM (Arjovsky et al. 2019) with either a uniform l_2-norm regularizer (IRM + l_2) or a uniform Lipschitz regularizer (IRM + lip), Mixup (Yan et al. 2020), MLDG (Li et al. 2018a), MTL (Blanchard et al. 2021), RSC (Huang et al. 2020), SagNet (Nam et al. 2019), and SD (Pezeshki et al. 2020).

Combination of IRM and methods for learning from imbalanced or noisy data. In the context of in-distribution generalization, many algorithms have been proposed to promote the performance of learning from the imbalanced (Cao et al. 2019; Liu et al. 2019) or noisy data (Li et al. 2019; Xu et al. 2019; Chen et al. 2019; Huang, Zhang, and Zhang 2020). We select three recently proposed algorithms from them and combine IRM with each of them to create three new baselines. We compare our methods with these three baselines to show that, when learning from imbalanced or noisy data, simply adopting the ideas in the algorithms built for in-distribution generalization does not work well for OoD problems. The algorithms we choose are: an imbalanced learning algorithm, LDAM-DRW (Cao et al. 2019), and a noisy label learning algorithm, Self-Adaptive Training (SAT) (Huang, Zhang, and Zhang 2020). We also compare with HAR (Cao et al. 2021) which addresses the imbalance and label noise in a uniform way by adaptive regularization technique.

Sample reweighting method for OoD problems. Since the idea of our algorithm is related to sample reweighting methods, we compare with CNBB (He, Shen, and Cui 2021) which is the state-of-the-art that uses sample reweighting to address OoD generalization problems.

	Cigar	Wage
ERM + l_2	29.85 ± 2.90***	64.33 ± 13.50***
ERM + lip	6.90 ± 0.95***	34.95 ± 7.09***
ANDMask	16.63 ± 10.52***	101.01 ± 73.38***
IRM + l_2	24.79 ± 1.76***	49.16 ± 7.94***
IRM + lip	6.80 ± 0.84***	34.93 ± 7.12***
MLDG	6.54 ± 3.22***	114.72 ± 85.14***
RPO (ours)	**4.35** ± **0.33**	**2.82** ± **1.43**

Table 2: Performances on regression datasets Cigar and Wage, measured by Mean Square Error (MSE) ± its standard deviation. "***", "**" and "*" indicate the significance of the t-test (see Section 4.2).

4.2 *t*-test

The t-test provides a more strict examination of the performance gap between two methods, so besides reporting the average performance metric (e.g. the mean square error or the mean accuracy), for each baseline, we carry out the Welch's t-test of the null hypothesis that the mean of certain performance metric of our method is equal to the mean of that metric of the baseline. The signs "***","**" and "*" in Table 2, Figure 1 and Figure 2 are used to indicate that the p-value of the t-test is less than 0.01, 0.05 and 0.1, respectively. The more "*" a baseline has, the more significant performance gain our method has compared to that baseline.

4.3 Regression Experiments

Cigar (Baltagi and Levin 1992; Baltagi 2021) and Wage (Wooldridge 2016) are two commonly used datasets on regression tasks in the study of fixed effect model (Wooldridge 2016; Rosner 2010). These datasets satisfy the assumption that the statistical differences in data are caused by only a few dominating features. We construct different domains by adding different confounders (or in other words, non-causal features) following the procedure in (Arjovsky et al. 2019), where the confounders are generated by different linear transformations of the label plus Gaussian noises. In the training domains, the confounders are closely related to the label (in the sense of Pearson Correlation Coefficient). In the testing domain, the variances of the Gaussian noises are increased to decorrelate the confounders with the label. This also makes the scale

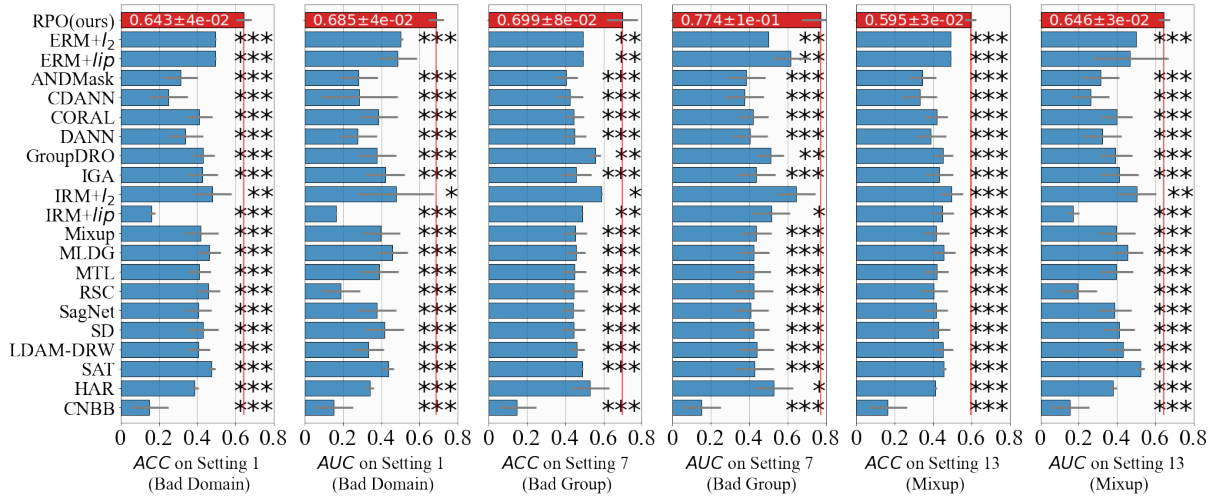

Figure 1: Performances on Colored-MNIST under three kinds of data setting (Bad Domain, Bad Group, Mixup). The horizontal bars indicate the accuracy (ACC) or the Area under the ROC Curve (AUC) of the method. The narrow gray bars indicate the standard deviation. "***", "**" and "*" indicate the significance of the t-test (see Section 4.2).

of the confounders in the testing domain much larger than that in the training domains. So, in the testing domain, the MSE of the predictor which utilizes the confounders to draw prediction should be much larger than the MSE of the predictor which filters out the confounders. We use this phenomenon to detect whether the OoD generalization algorithms can successfully learn the causal features. As shown by the results in Table 2, the MSE of our method is much smaller than other baselines, especially, on the Wage dataset. This indicates that our method has the best OoD performance. Some additional descriptions are shown in Appendix C.2 to C.4.

4.4 Classification Experiments

Colored-MNIST (Arjovsky et al. 2019) and NICO (He, Shen, and Cui 2021) are two commonly used datasets for OoD generalization problem on classification tasks.

Colored-MNIST endows the grayscale MNIST image with a binary color attribute to simulate the spurious correlation in the training set. We focus on a harder but more practical setting by introducing density variance and noise rate variance into Colored-MNIST.

We systematically create 14 experiment settings (see Appendix C.5) to simulate various kinds of distributions of the data with density variance and noise rate variance. Here, the results of setting 1 (Bad Domain[3]), setting 7 (Bad Group[4]), and setting 13 (Mixup[5]) are shown in Figure 1. An illustration of setting 1 and setting 7 are shown in Figure 3. It is worth noticing that our method achieves statistical significant performance gain over all baselines, except the AUC of ERM+lip on Setting 1, IRM+l_2 on Setting 7, and ERM+lip on Setting 13, and the ACC of HAR on Setting 7. We attribute these failure cases to the fact that the performance of these algorithms are unstable. The variances of the AUC (or ACC) of these baselines are large so that the t-test fails to distinguish our method from them. Despite lack of statistical significance in these three comparisons, our method is stabler (has smaller variance) and achieves better average performance than them.

NICO consists of real-world photos of animals and vehicles taken in a large variety of contexts, for example "on snow", "on beach" and "on grass". Our version of NICO is a binary classification problem where the class labels $Animal$ and $Vehicle$ are spuriously correlated to the photo context. Density variance and noise rate variance are stimulated following the same procedure as generating Setting 13 in our experiment of Colored-MNIST. As shown in Figure 2, RPO achieves state-of-the-art (SOTA) performance in terms of both ACC and AUC. The performance gains are statistically significant comparing to almost all the baselines. This demonstrate the utility of RPO in the complex real-world dataset. Additional descriptions are shown in Appendix C.6.

4.5 Further Discussions

In this subsection, further experimental results are provided.

Visualization of η_e^* and ρ_i^*. A graphical illustration of how we generate setting 1 and 7 together with the regularization penalty obtained by RPO are shown in Figure 3. As shown by the figure, our method successfully downweights the low-quality domains with smaller IRM penalties and penalizes the low-quality data groups with larger Lipschitz regularizations.

Ablation study. RPO (our method) uses the optimized penalty for both the IRM term and the Lipschitz regularization term. We compare RPO with *Ablation Study 1:* RPO-

[3]Bad Domain. There are certain low-quality domains in which a part of the data has high density variance and noise rate variance.

[4]Bad Group. Suppose the data can be divided into groups according to some prior knowledge. There are certain low-quality groups suffering from density variance and noise rate variance.

[5]Mixup. The Bad Group and Bad Domain occur together.

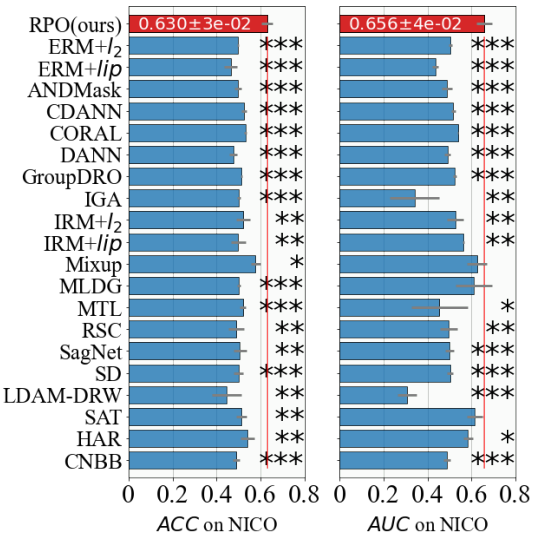

Figure 2: Performances on NICO. The horizontal bars indicate the accuracy (ACC) or the Area under the ROC Curve (AUC) of the method. The narrow gray bars indicate the standard deviation. "***", "**" and "*" indicate the significance of the t-test (see Section 4.2).

	Setting 1 (Bad domain)		Setting 7 (Bad Group)	
	ACC	AUC	ACC	AUC
RPO-Lip	0.398±1e-2	0.351±2e-3	0.679±3e-2	0.740±5e-2
RPO-Pen	0.616±5e-2	0.649±7e-2	0.544±1e-1	0.555±1e-1
RPO	**0.643±4e-2**	**0.685±4e-2**	**0.699±8e-2**	**0.774±1e-1**

Table 3: Ablation study on Colored-MNIST, measured by the accuracy (ACC) or the Area under the ROC Curve (AUC) ± corresponding standard deviation.

Pen, which uses optimized penalty for the IRM term and uniform regularization for the Lipschitz regularization term, and *Ablation Study 2:* RPO-Lip, which uses uniform regularization for the IRM term and optimized penalty for the Lipschitz regularization term. As shown in the Table 3, for the setting with bad domain, domain-wise reweighting is more effective, and RPO-Pen performs better than RPO-Lip. However, the setting with bad group benefits more from the sample-wise reweighting, and RPO-Lip performs better than RPO-Pen. This is consistent with the intuition that both domain-wise reweighting and sample-wise reweighting are necessary, though, they are suitable for different types of low-quality data distributions. Moreover, the combination of domain-wise reweighting and sample-wise reweighting leads to further improvements, outperforming RPO-Pen and RPO-Lip in both settings.

5 Related Work

Our work is in the field of OoD generalization. Approaches like Bayesian methods (Neal 2012), data augmentation (Zhao et al. 2019; Liu et al. 2020a), robust optimization (Lee and Raginsky 2018; Hoffman, Mohri, and Zhang 2018), and causal-based methods (Kuang et al. 2018; Rojas-Carulla et al. 2018; Subbaswamy, Schulam, and Saria 2019; Shen et al. 2020) have been proved successful in addressing the OoD generalization problem. In this paper, we take two types of data quality problems into consideration: the density variance and the noise rate variance. In the in-distribution generalization setting, the density variance (also named as imbalance or long-tailed distribution) of data can be alleviated through reweighting (Cao et al. 2019; Li, Liu, and Wang 2019), resampling (Byrd and Lipton 2019; Liu et al. 2020b), Bayesian learning (Tian et al. 2020) and self-supervise learning (Yang and Xu 2020). The noise rate variance is new for the deep learning context (Cao et al. 2021) and can be connected to the field of noisy data learning which can be handled by reweighting (Ren et al. 2018; Shu et al. 2019), self-adaptive learning (Huang, Zhang, and Zhang 2020), transition matrix estimating (Yao et al. 2020) and curriculum learning (Jiang et al. 2018). However, as shown by our experiments, a simple combination of the OoD generalization algorithms with the long-tailed data learning or noisy data learning methods cannot achieve the optimal performance. To recover the performance loss when training with low-quality data, OoD algorithms need special designs.

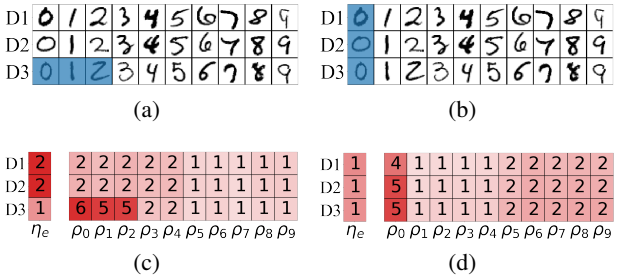

Figure 3: Subfigure 3(a) and 3(b) illustrate our setting 1 and 7, respectively. "Domain" is abbreviated to "D" in the figure. We divide the data in each domain into 10 groups according to the digit. We corrupt the data belonging to the groups in blue shadow by downsampling with probability 0.1 and switching their binary label with probability 0.3. Though only three groups out of thirty have density variance and noise rate variance, the performance of OoD generalization algorithms declines drastically. Subfigure 3(c) and 3(d) present the (relative values of) weights η_e and ρ_i obtained by our method under setting 1 and 7, respectively.

6 Conclusion

We have identified the problem of poor performance of OoD generalization algorithms caused by the varying quality of training data and provided the first solution to this problem. By theoretically deriving the optimal regularization scheme for IRM with Lipschitz regularizer, we have proposed a novel algorithm named RPO which regularizes via sample-wise and domain-wise penalties. We have conducted extensive experiments on publicly available real-world datasets and the results validate the effectiveness of our method.

Acknowledgments

Nanyang Ye was supported by National Natural Science Foundation of China under Grant 62106139. The work of Shao-Lun Huang was supported in part by the National Natural Science Foundation of China under Grant 61807021, in part by the Shenzhen Science and Technology Program under Grant KQTD20170810150821146.

References

Ahuja, K.; Wang, J.; Dhurandhar, A.; Shanmugam, K.; and Varshney, K. R. 2021. Empirical or Invariant Risk Minimization? A Sample Complexity Perspective. In *ICLR*.

Arjovsky, M. 2020. *Out of Distribution Generalization in Machine Learning*. Ph.D. thesis, New York University.

Arjovsky, M.; Bottou, L.; Gulrajani, I.; and Lopez-Paz, D. 2019. Invariant Risk Minimization. *arXiv/1907.02893*.

Baltagi, B. H. 2021. *Econometric Analysis of Panel Data*. SPRINGER.

Baltagi, B. H.; and Levin, D. 1992. Cigarette Taxation: Raising Revenues and Reducing Consumption. *Structural Change and Economic Dynamics*.

Blanchard, G.; Deshmukh, A. A.; Dogan, Ü.; Lee, G.; and Scott, C. 2021. Domain Generalization by Marginal Transfer Learning. *J. Mach. Learn. Res.*

Byrd, J.; and Lipton, Z. C. 2019. What is the Effect of Importance Weighting in Deep Learning? In *ICML*.

Cao, K.; Chen, Y.; Lu, J.; Arechiga, N.; Gaidon, A.; and Ma, T. 2021. Heteroskedastic and Imbalanced Deep Learning with Adaptive Regularization. In *ICLR*.

Cao, K.; Wei, C.; Gaidon, A.; Aréchiga, N.; and Ma, T. 2019. Learning Imbalanced Datasets with Label-Distribution-Aware Margin Loss. In *NeurIPS*.

Chen, P.; Liao, B.; Chen, G.; and Zhang, S. 2019. Understanding and Utilizing Deep Neural Networks Trained with Noisy Labels. In *ICML*.

Ganin, Y.; Ustinova, E.; Ajakan, H.; Germain, P.; Larochelle, H.; Laviolette, F.; Marchand, M.; and Lempitsky, V. S. 2016. Domain-Adversarial Training of Neural Networks. *J. Mach. Learn. Res.*

Gulrajani, I.; and Lopez-Paz, D. 2021. In Search of Lost Domain Generalization. In *ICLR*.

He, Y.; Shen, Z.; and Cui, P. 2021. Towards Non-I.I.D. image classification: A dataset and baselines. *Pattern Recognit.*

Hoffman, J.; Mohri, M.; and Zhang, N. 2018. Algorithms and Theory for Multiple-Source Adaptation. In *NeurIPS*.

Huang, L.; Zhang, C.; and Zhang, H. 2020. Self-Adaptive Training: beyond Empirical Risk Minimization. In *NeurIPS*.

Huang, Z.; Wang, H.; Xing, E. P.; and Huang, D. 2020. Self-challenging Improves Cross-Domain Generalization. In *ECCV*.

Jiang, L.; Zhou, Z.; Leung, T.; Li, L.; and Fei-Fei, L. 2018. MentorNet: Learning Data-Driven Curriculum for Very Deep Neural Networks on Corrupted Labels. In *ICML*.

Koyama, M.; and Yamaguchi, S. 2020. When is invariance useful in an Out-of-Distribution Generalization problem ? *arXiv/2008.01883*.

Kriegel, H.; Kröger, P.; and Zimek, A. 2009. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. *ACM Trans. Knowl. Discov. Data*.

Kuang, K.; Cui, P.; Athey, S.; Xiong, R.; and Li, B. 2018. Stable Prediction across Unknown Environments. In *KDD*.

Lee, J.; and Raginsky, M. 2018. Minimax Statistical Learning with Wasserstein distances. In *NeurIPS*.

Li, B.; Liu, Y.; and Wang, X. 2019. Gradient Harmonized Single-Stage Detector. In *AAAI*.

Li, D.; Yang, Y.; Song, Y.; and Hospedales, T. M. 2018a. Learning to Generalize: Meta-Learning for Domain Generalization. In *AAAI*.

Li, J.; Wong, Y.; Zhao, Q.; and Kankanhalli, M. S. 2019. Learning to Learn From Noisy Labeled Data. In *CVPR*.

Li, Y.; Tian, X.; Gong, M.; Liu, Y.; Liu, T.; Zhang, K.; and Tao, D. 2018b. Deep Domain Generalization via Conditional Invariant Adversarial Networks. In *ECCV*.

Liu, J.; Sun, Y.; Han, C.; Dou, Z.; and Li, W. 2020a. Deep Representation Learning on Long-Tailed Data: A Learnable Embedding Augmentation Perspective. In *CVPR*.

Liu, Z.; Miao, Z.; Zhan, X.; Wang, J.; Gong, B.; and Yu, S. X. 2019. Large-Scale Long-Tailed Recognition in an Open World. In *CVPR*.

Liu, Z.; Wei, P.; Jiang, J.; Cao, W.; Bian, J.; and Chang, Y. 2020b. MESA: Boost Ensemble Imbalanced Learning with MEta-SAmpler. In *NeurIPS*.

Nam, H.; Lee, H.; Park, J.; Yoon, W.; and Yoo, D. 2019. Reducing Domain Gap via Style-Agnostic Networks. *arXiv/1910.11645*.

Neal, R. M. 2012. *Bayesian Learning for Neural Networks*. Springer Science & Business Media.

Parascandolo, G.; Neitz, A.; Orvieto, A.; Gresele, L.; and Schölkopf, B. 2020. Learning explanations that are hard to vary. *arXiv/2009.00329*.

Pezeshki, M.; Kaba, S.; Bengio, Y.; Courville, A. C.; Precup, D.; and Lajoie, G. 2020. Gradient Starvation: A Learning Proclivity in Neural Networks. *arXiv/2011.09468*.

Ren, M.; Zeng, W.; Yang, B.; and Urtasun, R. 2018. Learning to Reweight Examples for Robust Deep Learning. In *ICML*.

Rojas-Carulla, M.; Schölkopf, B.; Turner, R. E.; and Peters, J. 2018. Invariant Models for Causal Transfer Learning. *J. Mach. Learn. Res.*

Rosner, B. 2010. *Fundamentals of Biostatistics*. Cengage Learning.

Sagawa, S.; Koh, P. W.; Hashimoto, T. B.; and Liang, P. 2019. Distributionally Robust Neural Networks for Group Shifts: On the Importance of Regularization for Worst-Case Generalization. *arXiv/1911.08731*.

Shen, Z.; Cui, P.; Zhang, T.; and Kuang, K. 2020. Stable Learning via Sample Reweighting. In *AAAI*.

Shu, J.; Xie, Q.; Yi, L.; Zhao, Q.; Zhou, S.; Xu, Z.; and Meng, D. 2019. Meta-Weight-Net: Learning an Explicit Mapping For Sample Weighting. In *NeurIPS*.

Shui, C.; Wang, B.; and Gagné, C. 2021. On the benefits of representation regularization in invariance based domain generalization. *arXiv/2105.14529*.

Stone, M.; and Goldbart, P. 2009. *Mathematics for Physics: A Guided Tour for Graduate Students*. Cambridge University Press.

Subbaswamy, A.; Schulam, P.; and Saria, S. 2019. Preventing Failures Due to Dataset Shift: Learning Predictive Models That Transport. In *AISTATS*.

Sun, B.; and Saenko, K. 2016. Deep CORAL: Correlation Alignment for Deep Domain Adaptation. In *ECCV*.

Tian, J.; Liu, Y.; Glaser, N.; Hsu, Y.; and Kira, Z. 2020. Posterior Re-calibration for Imbalanced Datasets. In *NeurIPS*.

Vapnik, V. 1998. *Statistical Learning Theory*. Wiley.

Wang, X.; Du, P.; and Shen, J. 2013. Smoothing splines with varying smoothing parameter. *Biometrika*.

Wei, C.; and Ma, T. 2019. Data-dependent Sample Complexity of Deep Neural Networks via Lipschitz Augmentation. In *NeurIPS*.

Wei, C.; and Ma, T. 2020. Improved Sample Complexities for Deep Neural Networks and Robust Classification via an All-Layer Margin. In *ICLR*.

Wooldridge, J. M. 2016. *Introductory Econometrics: A Modern Approach*. Nelson Education.

Xu, Y.; Cao, P.; Kong, Y.; and Wang, Y. 2019. L_DMI: A Novel Information-theoretic Loss Function for Training Deep Nets Robust to Label Noise. In *NeurIPS*.

Yan, S.; Song, H.; Li, N.; Zou, L.; and Ren, L. 2020. Improve Unsupervised Domain Adaptation with Mixup Training. *arXiv/2001.00677*.

Yang, Y.; and Xu, Z. 2020. Rethinking the Value of Labels for Improving Class-Imbalanced Learning. In *NeurIPS*.

Yao, Y.; Liu, T.; Han, B.; Gong, M.; Deng, J.; Niu, G.; and Sugiyama, M. 2020. Dual T: Reducing Estimation Error for Transition Matrix in Label-noise Learning. In *NeurIPS*.

Zhao, S.; Fard, M. M.; Narasimhan, H.; and Gupta, M. R. 2019. Metric-Optimized Example Weights. In *ICML*.