A PRINCIPLED APPROACH TO MODEL VALIDATION IN DOMAIN GENERALIZATION

Boyang Lyu[†⋆] *Thuan Nguyen*[¶⋆] *Matthias Scheutz*[¶] *Prakash Ishwar*[‡] *Shuchin Aeron*[†]

[¶] Department of Computer Science, Tufts University, Medford, MA 02155
[†] Department of Electrical and Computer Engineering, Tufts University, Medford, MA 02155
[‡]Department of Electrical and Computer Engineering, Boston University, Boston, MA 02215

ABSTRACT

Domain generalization aims to learn a model with good generalization ability, that is, the learned model should not only perform well on several seen domains but also generalize well on other unseen domains with different data distributions. The state-of-the-art domain generalization methods usually train a representation function followed by a classifier to minimize both the classification risk and the domain discrepancy. However, during the model selection process, most of these methods follow the traditional validation routines by only selecting the models with the lowest classification risk on the validation set. In this paper, we theoretically demonstrate a trade-off between minimizing classification risk and mitigating domain discrepancy, *i.e.,* it is impossible to achieve the minimum of these two objectives simultaneously. Motivated by this theoretical result, we revisit the current model selection (validation) methods for the domain generalization problem and suggest that the validation process must account for both the classification risk and the domain discrepancy. Finally, we numerically verify this argument on several domain generalization datasets.

Index Terms— Domain generalization, model selection, validation method, domain discrepancy.

1. INTRODUCTION

The success of traditional machine learning methods relies on an important assumption that the training and the test data are independent and identically distributed (*i.i.d*). However, in many real-world scenarios, the distributions of data in the training set and test set are not identical due to the "distribution-shift" phenomenon. Mitigating the problem caused by the distribution shift is the primary goal of the Domain Generalization (DG) problem, where a model is trained using data from several seen domains but later needs to be applied to an unseen (unknown but related) domain with different data distributions.

To address DG problem, a large number of methods considers training a representation function that can learn domain-invariant features[1] by minimizing the domain discrepancy in the representation space [1–6]. Though the domain discrepancy has been accounted for at the training step, few works considered it for model selection at the validation step [7]. Indeed, following traditional machine learning settings, most of the state-of-the-art DG methods form a validation set using a small portion of data from all seen domains and select the model that achieves the lowest classification risk or highest classification accuracy on it. However, unlike the traditional machine learning settings where a model with low classification risk on the validation set is likely to perform well on the test set, we theoretically show that for DG problem, where the *i.i.d* assumption does not hold, selecting the model with minimum classification risk may enlarge the domain discrepancy, subsequently leading to a non-optimal model on the unseen domain. We thus argue that one needs to consider both the classification risk and the domain discrepancy for selecting good models on unseen domains.

We summarize our contributions as follows:

1. We theoretically show that there is a trade-off between minimizing classification risk and domain discrepancy. This trade-off leads to the conclusion that if one only targets a model with the lowest classification risk on the validation set, it may encourage a huge mismatch in distributions between domains (enlarging domain discrepancy), and the learned model may lose its generalization ability.

2. Motivated by our theoretical result and the fact that only a few DG works focus on designing DG-specific validation processes, we suggest an easily implemented validation/model selection method combining both the classification risk and the domain discrepancy as validation criteria.

3. We demonstrate the efficiency of the proposed validation method on several DG benchmark datasets. Though the proposed method is able to achieve comparable or slightly better results compared to the state-of-the-art methods, we point out some limitations of our work, leaving them as open problems for future research.

[*]These authors contributed equally to this work.

[1]Domain-invariant features are the features having distributions that are unchanged and stable across domains.

2. RELATED WORK

The trade-off between minimizing the classification risk and domain discrepancy has been mentioned in the literature [8,9][2]. Shai *et al.* [8] constructed an upper bound on the risk of the target domain, composed of the risk from the source domain and the discrepancy between the target and source domains. The authors suggested that there must be a trade-off between minimizing the domain discrepancy and minimizing the seen domain's risk. However, Shai *et al.* [8] did not propose any further details on how this trade-off is determined and characterized. Zhao *et al.* [9] showed that the sum of the risks from source and target domains is lower bounded by the distribution discrepancy between domains. If the discrepancy between domains is large, one can not simultaneously achieve small risks in both domains. Though sharing some similarities, our theoretical result differs from [9] since Zhao *et al.* considered the trade-off between minimizing the risks of different domains rather than the trade-off between optimizing the classification risk and the domain discrepancy.

On the other hand, the practical model selection methods of most DG works follow the traditional machine learning settings, *i.e.*, a validation set is first formed by combining small portions of data from all seen domains then a model that produces the lowest classification risk or highest classification accuracy on the validation set is selected. Though leave-one-domain-out validation method [10] is also considered by some works [10] as an alternative way to form the validation set, it is unsuitable for datasets with three domains like Colored-MNIST [1] and computationally expensive, thus is out of the scope of this paper. To the best of our knowledge, there are only a few works that explore different model selection methods for DG [11–14]. The most related work of this study is [14], where the authors mentioned that they use the training loss (including both classification risk and adversarial domain discrepancy loss) on the validation set for model selection. However, it is not clear from both their paper and their released code[3] how the classification risk and the adversarial domain discrepancy loss are used to validate the model and how these two terms are balanced. In contrast, we propose an alternative approach for combining the classification risk and the domain discrepancy loss in a meaningful way in light of our theoretical results.

3. PROBLEM FORMULATION

3.1. Notations

Let $\mathcal{X}$, $\mathcal{Z}$, $\mathcal{Y}$ denote the input space, the representation space, and the label space, respectively. Let $\mathcal{D}^{(s)}$ denote the seen domain and $\mathcal{D}^{(u)}$ denote the unseen domain, respectively. Based on the theme of representation learning, DG tasks aim to learn a representation function $f : \mathcal{X} \to \mathcal{Z}$ followed by a classifier $g : \mathcal{Z} \to \mathcal{Y}$ that is trained on seen domain $\mathcal{D}^{(s)}$ but generalizes well on unseen domain $\mathcal{D}^{(u)}$. In practice, several seen domains are available at the training time.

Let X denote the input random variable, Z denote the extracted feature random variable, and Y denote the label random variable in input space, representation space, and label space, respectively. Let $p^{(s)}(X, Z)$, $p^{(s)}(Y, Z)$ and $p^{(u)}(X, Z)$, $p^{(u)}(Y, Z)$ denote the joint distribution between X and Z and the joint distribution between Y and Z in seen and unseen domains, respectively.

We use $p^{(s)}(\boldsymbol{x})$, $p^{(s)}(\boldsymbol{z})$, $p^{(s)}(\boldsymbol{x}, \boldsymbol{z})$ and $p^{(u)}(\boldsymbol{x})$, $p^{(u)}(\boldsymbol{z})$, $p^{(u)}(\boldsymbol{x}, \boldsymbol{z})$ to denote the distribution of input sample $\boldsymbol{x}$, the distribution of feature sample $\boldsymbol{z} = f(\boldsymbol{x})$, and their joint distribution in seen and unseen domains, respectively. Finally, we use $y^{(s)}(\boldsymbol{x})$ and $y^{(u)}(\boldsymbol{x})$ to denote the label of input sample $\boldsymbol{x}$ in seen and unseen domain, respectively.

3.2. Problem formulation

For a given representation function f and a labeling function g, the classification risk induced by f and g on seen domain is defined by:

$$\begin{aligned} C^{(s)}(f, g) &= \int_{\boldsymbol{x}\in\mathcal{X}} p^{(s)}(\boldsymbol{x})\ell(g(f(\boldsymbol{x})), y^{(s)}(\boldsymbol{x}))d\boldsymbol{x} \\ &= \int_{\boldsymbol{x}\in\mathcal{X}}\int_{\boldsymbol{z}\in\mathcal{Z}} p^{(s)}(\boldsymbol{x}, \boldsymbol{z})\ell(g(\boldsymbol{z}), y^{(s)}(\boldsymbol{x}))d\boldsymbol{x}d\boldsymbol{z} \quad (1) \end{aligned}$$

where $\ell(\cdot, \cdot)$ is a distance measure which quantifies the mismatch between the label outputted by classifier g and its true label.

For a given representation function f, the distribution discrepancy between seen and unseen domain induced by f is defined by:

$$D(f) = d(p^{(u)}(Y, Z)||p^{(s)}(Y, Z)) \quad (2)$$

where $d(\cdot||\cdot)$ is a divergence measure between two distributions. Indeed, to deal with the "distribution-shift" settings, one usually looks for a mapping f such that the discrepancy between distributions of seen and unseen domains $D(f)$ is small [15, 16].

Recent works on DG train a model on seen domains to minimize its classification risk $C^{(s)}(f, g)$ as well as the discrepancy between domains $D(f)$ [1–6]. Note that while $C^{(s)}(f, g)$ can be directly minimized, one usually approximately/heuristically optimizes $D(f)$ by optimizing the distribution discrepancy between several seen domains. Since both the theoretical work and empirical work on minimizing the classification risk and domain discrepancy are well done in the literature, in this paper, our goal is to show that there is a trade-off between minimizing the classification risk and optimizing the domain discrepancy (Sec 4), which motivates a crucial fact that one

[2]The works in [8, 9] are for domain adaptation, not domain generalization. However, one may derive a similar conclusion by replacing the "source domain" with seen domain and the "target domain" with unseen domain.

[3]https://github.com/belaalb/G2DM/tree/master/pacs-ours

should account for both classification risk and domain discrepancy for model selection method to achieve better performance on the unseen domain (Sec. 5).

4. TRADE-OFF BETWEEN CLASSIFICATION RISK AND DOMAIN DISCREPANCY

We first begin with a definition.

Definition 1 (Classification risk-domain discrepancy function)**.** *For any representation function f and classifier g, define:*

$$\begin{aligned} &T(\Delta) = \min_{f:\mathcal{X}\to\mathcal{Z}} D(f) = \min_{f:\mathcal{X}\to\mathcal{Z}} d(p^{(u)}(Y,Z)||p^{(s)}(Y,Z)) \\ &s.t. \quad C^{(s)}(f,g) = \int_{\boldsymbol{x}\in\mathcal{X}} p^{(s)}(\boldsymbol{x})\ell(g(f(\boldsymbol{x})), y^{(s)}(\boldsymbol{x}))d\boldsymbol{x} \le \Delta \end{aligned} \tag{3}$$

where Δ is a positive number, $\ell(\cdot,\cdot)$ is a distance measure, and $d(\cdot||\cdot)$ is a divergence measure.

$T(\Delta)$ is the minimal discrepancy between the joint distribution of unseen domain and seen domain if the classification risk on seen domain $C^{(s)}(f,g)$ does not exceed a positive threshold Δ. Next, we formally show that there is a trade-off between minimizing the distribution discrepancy $D(f)$ and minimizing the classification risk $C^{(s)}(f,g)$.

Theorem 1 (Main result)**.** *If the divergence measure $d(a||b)$ is convex (in both a and b), for a fixed classifier g, $T(\Delta)$ defined in (3) is:*

1. *monotonically non-increasing, and*
2. *convex.*

Proof. The proof of this theorem is mainly based on the proposed approach in Rate-Distortion theory [17]. Particularly, consider two positive numbers Δ_1 and Δ_2, and assume that $\Delta_1 \le \Delta_2$. For a given classifier g, we use $\mathcal{F}_{\Delta_1}$ and $\mathcal{F}_{\Delta_2}$ to denote the sets of mappings f such that $C^{(s)}(f,g) \le \Delta_1$ and $C^{(s)}(f,g) \le \Delta_2$, respectively.

First, we show that $T(\Delta)$ is a monotonically non-increasing function. Indeed, from $\Delta_1 \le \Delta_2$, we have $\mathcal{F}_{\Delta_1} \subset \mathcal{F}_{\Delta_2}$. Thus,

$$\begin{aligned} T(\Delta_1) &= \min_{f\in\mathcal{F}_{\Delta_1}} d(p^{(u)}(Y,Z)||p^{(s)}(Y,Z)) \\ &\ge \min_{f\in\mathcal{F}_{\Delta_2}} d(p^{(u)}(Y,Z)||p^{(s)}(Y,Z)) = T(\Delta_2). \end{aligned}$$

By definition, $T(\Delta)$ is a monotonically non-increasing function.

Second, to prove the convexity of $T(\Delta)$, we show that:

$$\lambda T(\Delta_1) + (1-\lambda)T(\Delta_2) \ge T(\lambda\Delta_1 + (1-\lambda)\Delta_2), \tag{4}$$

for any $\lambda \in [0,1]$.

To prove (4), we need some additional notations. Let:

$$f_1 = \arg\min_{f:\mathcal{X}\to\mathcal{Z}} D(f) \quad \text{s.t.} \qquad C^{(s)}(f,g) \le \Delta_1, \tag{5}$$

$$f_2 = \arg\min_{f:\mathcal{X}\to\mathcal{Z}} D(f) \quad \text{s.t.} \qquad C^{(s)}(f,g) \le \Delta_2. \tag{6}$$

Note that for any mapping f, $Y \to X \to Z$ forms a Markov chain, then:

$$p^{(u)}(Y,Z) = p^{(u)}(Y|X)\, p^{(u)}(X,Z), \tag{7}$$

and

$$p^{(s)}(Y,Z) = p^{(s)}(Y|X)\, p^{(s)}(X,Z), \tag{8}$$

where $p^{(u)}(Y|X)$ and $p^{(s)}(Y|X)$ are independent of f and only depend on the conditional distributions of label and data on seen and unseen domains.

Let $p_1^{(u)}(Y,Z)$, $p_1^{(s)}(Y,Z)$ be the joint distributions of Y and Z on unseen and seen domain produced by f_1. Let $p_2^{(u)}(Y,Z)$, $p_2^{(s)}(Y,Z)$ be the joint distributions of Y and Z on unseen and seen domain produced by f_2. Let $p_1^{(u)}(X,Z)$, $p_1^{(s)}(X,Z)$ be the joint distributions of X and Z on unseen and seen domain produced by f_1, and $p_2^{(u)}(X,Z)$, $p_2^{(s)}(X,Z)$ be the joint distributions of X and Z on unseen and seen domain produced by f_2.

Define:

$$p_\lambda^{(u)}(X,Z) = \lambda p_1^{(u)}(X,Z) + (1-\lambda)p_2^{(u)}(X,Z), \tag{9}$$

$$p_\lambda^{(s)}(X,Z) = \lambda p_1^{(s)}(X,Z) + (1-\lambda)p_2^{(s)}(X,Z). \tag{10}$$

By definition, the left hand side of (4) can be rewritten by:

$$\begin{aligned} & \lambda T(\Delta_1) + (1-\lambda)T(\Delta_2) \\ = \;& \lambda d(p_1^{(u)}(Y,Z)\,||\,p_1^{(s)}(Y,Z)) \\ + \;& (1-\lambda) d(p_2^{(u)}(Y,Z)\,||\,p_2^{(s)}(Y,Z)) \\ = \;& \lambda d(p^{(u)}(Y|X)p_1^{(u)}(X,Z)||p^{(s)}(Y|X)p_1^{(s)}(X,Z)) && (11) \\ + \;& (1-\lambda) d(p^{(u)}(Y|X)p_2^{(u)}(X,Z)||p^{(s)}(Y|X)p_2^{(s)}(X,Z)) && (12) \\ \ge \;& d(p^{(u)}(Y|X)p_\lambda^{(u)}(X,Z)||p^{(s)}(Y|X)p_\lambda^{(s)}(X,Z)) && (13) \end{aligned}$$

where (11) and (12) due to for any mapping f, $p^{(u)}(Y,Z) = p^{(u)}(Y|X)\, p^{(u)}(X,Z)$ and $p^{(s)}(Y,Z) = p^{(s)}(Y|X)\, p^{(s)}(X,Z)$ which are already mentioned in (7) and (8); (13) due to (9), (10), and the convexity of $d(\cdot||\cdot)$.

Let f_λ is the corresponding function that induces the joint distribution $p_\lambda^{(u)}(X,Z)$ and $p_\lambda^{(s)}(X,Z)$. Let:

$$\Delta_\lambda = \int_{\boldsymbol{x}\in\mathcal{X}} \int_{\boldsymbol{z}\in\mathcal{Z}} p_\lambda^{(s)}(\boldsymbol{x},\boldsymbol{z})\ell(g(\boldsymbol{z}), y^{(s)}(\boldsymbol{x}))\, d\boldsymbol{x}d\boldsymbol{z}. \tag{14}$$

By Definition of $T(\Delta)$ function in 1, we have:

$$d(p^{(u)}(Y|X)\, p_\lambda^{(u)}(X,Z)||p^{(s)}(Y|X)\, p_\lambda^{(s)}(X,Z)) \ge T(\Delta_\lambda). \tag{15}$$

Combine (13) and (15):

$$\lambda T(\Delta_1) + (1-\lambda)T(\Delta_2) \ge T(\Delta_\lambda). \tag{16}$$

That said, the left hand side of (4) is larger or at least equal to $T(\Delta_\lambda)$. We, therefore, need to show that $T(\Delta_\lambda)$ is at least as large as the right hand side of (4). Formally, we want to show that:

$$T(\Delta_\lambda) \geq T(\lambda\Delta_1 + (1-\lambda)\Delta_2). \tag{17}$$

Since $T(\Delta)$ is a monotonically non-increasing, (17) is equivalent to:

$$\Delta_\lambda \leq \lambda\Delta_1 + (1-\lambda)\Delta_2. \tag{18}$$

Indeed, we have:

$$\Delta_\lambda = \int_{\boldsymbol{x}}\int_{\boldsymbol{z}} p_\lambda^{(s)}(\boldsymbol{x},\boldsymbol{z})\ell(g(\boldsymbol{z}), y^{(s)}(\boldsymbol{x}))d\boldsymbol{x}d\boldsymbol{z} \tag{19}$$

$$= \lambda\int_{\boldsymbol{x}}\int_{\boldsymbol{z}} p_1^{(u)}(\boldsymbol{x},\boldsymbol{z})\ell(g(\boldsymbol{z}), y^{(s)}(\boldsymbol{x}))d\boldsymbol{x}d\boldsymbol{z} \tag{20}$$

$$+ \quad (1-\lambda)\int_{\boldsymbol{x}}\int_{\boldsymbol{z}} p_2^{(u)}(\boldsymbol{x},\boldsymbol{z})\ell(g(\boldsymbol{z}), y^{(s)}(\boldsymbol{x}))d\boldsymbol{x}d\boldsymbol{z} \tag{21}$$

$$\leq \quad \lambda\Delta_1 + (1-\lambda)\Delta_2 \tag{22}$$

with (19) due to (14), (20) and (21) due to (9), (22) due to (5) and (6), respectively.

From (18) and (22), (17) follows. Finally, from (16) and (17), (4) follows. The proof is complete. □

It is worth noting that the convexity of $d(\cdot||\cdot)$ is not a restricted condition, indeed, most of the divergence functions, for example, the Kullback-Leibler (KL) divergence is convex. Fig. 1 illustrates the plot of $T(\Delta)$.

Theorem 1 shows that if one focuses on enforcing a small distribution discrepancy between domains, then it will increase the classification risk and vice-versa. One may argue that if there is a feature having full information about the label (then one can design a classifier with a low risk on it) and at the same time, the distribution of this feature unchanged from domain to domain, is Theorem 1 still true? Based on our theoretical result, we provide a positive answer for this question. Indeed, suppose that the distribution-shift is light and one possible achieves a small value of $T(\Delta)$ for a given small value of Δ, it is still true that $T(\Delta)$ is monotonically non-increasing, and convex.

5. A NEW VALIDATION METHOD

Based on Theorem 1, we argue that to achieve a good model for unseen domains, one must account for both the classification risk and the domain discrepancy not only in the training process but also in the validation process. Note that state-of-the-art model evaluation methods for DG are mainly based on the classification risk or, equivalently, the classification accuracy [7] [10] on the validation set to select the models. Given this fact, in this paper, we want to select a model that minimizes the following objective function on the validation set:

$$L_{\text{Validation loss}} = \beta(1-\alpha)L_{\text{Classification risk}} + \alpha L_{\text{Domain-discrepancy loss}} \tag{23}$$

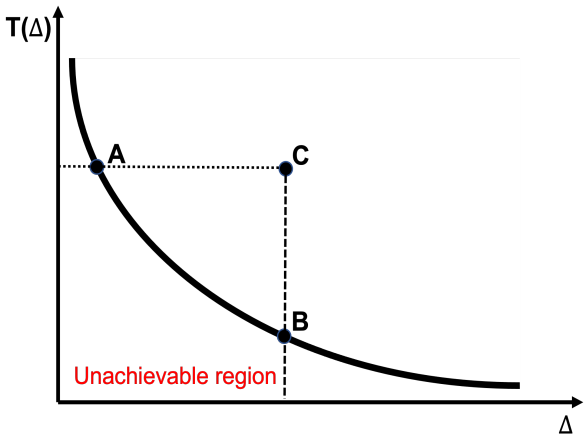

Fig. 1. $T(\Delta)$ is monotonically non-increasing, and convex. The region below $T(\Delta)$ curve is an unachievable region.

where α is the convex combination hyper-parameter and β is the scale hyper-parameter that supports the combination of objectives with different scales.

It is pretty clear that the cross-entropy loss is a good representation of classification risk, however, it is hard to quantify the domain-discrepancy loss. Indeed, there exist various definitions of domain discrepancy. Several works characterize the domain-discrepancy via the difference in the marginal distributions [5, 6], other works measure the domain-discrepancy by the mismatch in conditional distribution [1]. We believe that finding a good measure for domain discrepancy is still an open problem. Therefore, in this short paper, we decide to use the widely accepted Maximum Mean Discrepancy (MMD) loss [6] in the feature space to quantify the domain discrepancy. We acknowledge that this choice might not be the best for quantifying the domain discrepancy and encourage the readers to come up with other measures.

In practice, we found that MMD loss is at the same scale as the cross-entropy loss when the training process is stable, we thus choose β as 1. For α, we consider the classification performance and heuristically choose α as 0.2. From our experiments, we found that the performance of our validation method is robust to small values of α within the range of $[0.1, 0.3]$. One more insight from Fig. 1 is that one may want to avoid reaching the extreme point regions for both $T(\Delta)$ and Δ, to balance the model's ability of generalization and prediction. Thus, for each hyper-parameter configuration, we sort the validation cross-entropy loss in ascending order and only pick the models that produce 5% to 50% percentile of the validation cross-entropy loss as a subset of candidates for model selection.

Table 1. Results on PACS and C-MNIST datasets.

Algorithm	**Fish** [18]	**IRM** [1]	**GDRO** [13]	**Mixup** [19]	**CORAL** [5]	**MMD** [6]	**DANN** [20]	**CDANN** [21]	**MTL** [22]	**VREx** [23]	**RSC** [24]	**SagNet** [25]	**Wins**
PACS (Traditional)	**84.6**	84.9	84.2	83.3	**85.1**	83.6	84.6	**86.4**	83.0	**84.5**	**85.2**	83.7	
PACS (Ours)	82.0	**85.3**	**84.3**	**85.3**	84.9	**85.0**	**84.9**	82.0	**84.2**	84.2	81.3	**85.1**	7/12
CMNIST (Traditional)	**10.0**	10.0	10.2	**10.4**	9.7	**10.4**	10.0	9.9	10.5	10.2	10.2	10.4	
CMNIST (Ours)	9.7	**10.9**	**12.6**	10.3	**11.2**	9.9	**11.1**	**10.2**	**11.5**	**15.6**	**13.8**	**10.5**	9/12

6. NUMERICAL RESULTS

We compare our proposed model selection method with the Training-domain validation method[4] described in [10] on two datasets: PACS [26], Colored-MNIST (C-MNIST) [1] using DomainBed [10] package and 12 different DG algorithms provided there. For the PACS dataset, we report the average test accuracy over 4 different tasks with each time leaving one domain as the unseen domain. For the C-MNIST dataset, we only focus on the most difficult domain, where the correlation between the label and the color of the unseen domain is completely different from the seen domains and no algorithm can achieve more than 10.5% points accuracy [10].

The validation set is formed using 20% data from each seen domain, denoted as the training-domain validation set in [10]. For the training step, we follow exactly the same settings and training routine used in DomainBed and conduct 20 trials of random search over a joint distribution of hyperparameters for each task per algorithm. For the MMD loss implementation, we directly use the code provided in DomainBed package. We train each model for 5000 steps and record the validation cross-entropy loss, MMD loss, and validation accuracy every 300 steps.

From experiments, we heuristically select $\beta = 1$ and $\alpha = 0.2$. The performance of each algorithm under different validation methods on PACS and Colored-MNIST datasets is shown in Table 1. We refer to the Training-domain validation method as "Traditional" and the proposed method as "Ours". For the PACS dataset, the proposed validation method can select slightly better models for seven out of twelve DG algorithms. For the remaining five DG algorithms, the proposed method can also achieve comparable performance with the "Traditional" method on CORAL [5] and VREx [23]. However, for Fish [18], CDANN [21] and RSC [24], we observe a performance deterioration. For the C-MNIST dataset, the proposed validation method consistently selects models with better performance compared with the "Traditional" validation method. As shown in Table 1, there is a consistent increase in the accuracy for nine out of twelve tested algorithms using the proposed validation method, with the most significant improvement for VREx method by 5.4%. Due to the limit of space, our source code, the proof of Theorem 1 as well as additional numerical results on other datasets are uploaded at this link[5].

7. DISCUSSIONS

Note that by varying the value of α and β, calculating (23) approximately leads to different points in the curve of $T(\Delta)$. Unfortunately, from our theoretical result, all points belonging to $T(\Delta)$ are optimal. For example, in Fig. 1, one can not say a model induced point "A" is better than a model induced point "B". However, one can say the models induced "A" and "B" are better than a model induced "C". In addition, since the unseen domain is unknown, if the unseen domain is very similar to seen domains, we may want to select a small value of α while if the seen and unseen domains are very different, a large value of α might be required. Moreover, due to the domain discrepancies being different between datasets, one may not have the same optimal values of α and β for all datasets. Given this fact, theoretically determining the "optimal" values of α and β must be a hard problem. We, therefore, leave the questions of determining good values of α and β as well as finding the best measure for quantifying the domain discrepancy $L_{\text{Domain-discrepancy loss}}$ as open problems for future works.

8. CONCLUSIONS

In this paper, motivated by the trade-off between minimizing classification risk and domain discrepancy, and the fact that state-of-the-art model selection methods for DG are mainly based on minimizing the classification risk, we propose a validation method considering the classification risk and the domain discrepancy together. The proposed method demonstrates its effect on several datasets. Though our proposed method still has some limitations, we believe this approach provides some intuition and initial results for exploring new model selection methods in DG.

[4]Recall that the Training-domain validation method chooses the model that produces the highest validation accuracy while our method select the model that minimizes the objective function in (23).

[5]https://github.com/thuan2412/A-principled-approach-for-model-validation-for-domain-generalization

9. REFERENCES

[1] Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz, "Invariant risk minimization," *arXiv preprint arXiv:1907.02893*, 2019.

[2] Boyang Lyu, Thuan Nguyen, Prakash Ishwar, Matthias Scheutz, and Shuchin Aeron, "Barycentric-alignment and invertibility for domain generalization," *arXiv preprint arXiv:2109.01902*, 2021.

[3] Thuan Nguyen, Boyang Lyu, Prakash Ishwar, Matthias Scheutz, and Shuchin Aeron, "Conditional entropy minimization principle for learning domain invariant representation features," *arXiv preprint arXiv:2201.10460*, 2022.

[4] Bo Li, Yifei Shen, Yezhen Wang, Wenzhen Zhu, Dongsheng Li, Kurt Keutzer, and Han Zhao, "Invariant information bottleneck for domain generalization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022, vol. 36, pp. 7399–7407.

[5] Baochen Sun and Kate Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *European conference on computer vision*. Springer, 2016, pp. 443–450.

[6] Haoliang Li, Sinno Jialin Pan, Shiqi Wang, and Alex C Kot, "Domain generalization with adversarial feature learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5400–5409.

[7] Kaiyang Zhou, Ziwei Liu, Yu Qiao, Tao Xiang, and Chen Change Loy, "Domain generalization: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–20, 2022.

[8] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira, "Analysis of representations for domain adaptation," *Advances in neural information processing systems*, vol. 19, 2006.

[9] Han Zhao, Remi Tachet Des Combes, Kun Zhang, and Geoffrey Gordon, "On learning invariant representations for domain adaptation," in *International Conference on Machine Learning*. PMLR, 2019, pp. 7523–7532.

[10] Ishaan Gulrajani and David Lopez-Paz, "In search of lost domain generalization," in *International Conference on Learning Representations*, 2020.

[11] Yoav Wald, Amir Feder, Daniel Greenfeld, and Uri Shalit, "On calibration and out-of-domain generalization," *Advances in neural information processing systems*, vol. 34, pp. 2215–2227, 2021.

[12] Haotian Ye, Chuanlong Xie, Tianle Cai, Ruichen Li, Zhenguo Li, and Liwei Wang, "Towards a theoretical framework of out-of-distribution generalization," *Advances in Neural Information Processing Systems*, vol. 34, pp. 23519–23531, 2021.

[13] Shiori Sagawa*, Pang Wei Koh*, Tatsunori B. Hashimoto, and Percy Liang, "Distributionally robust neural networks," in *International Conference on Learning Representations*, 2020.

[14] Isabela Albuquerque, João Monteiro, Mohammad Darvishi, Tiago H Falk, and Ioannis Mitliagkas, "Generalizing to unseen domains via distribution matching," *arXiv preprint arXiv:1911.00804*, 2019.

[15] Thuan Nguyen, Boyang Lyu, Prakash Ishwar, Matthias Scheutz, and Shuchin Aeron, "Joint covariate-alignment and concept-alignment: a framework for domain generalization," *arXiv preprint arXiv:2208.00898*, 2022.

[16] Shai Ben David, Tyler Lu, Teresa Luu, and Dávid Pál, "Impossibility theorems for domain adaptation," in *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*. JMLR Workshop and Conference Proceedings, 2010, pp. 129–136.

[17] Thomas M Cover, *Elements of information theory*, John Wiley & Sons, 1999.

[18] Yuge Shi, Jeffrey Seely, Philip Torr, Siddharth N, Awni Hannun, Nicolas Usunier, and Gabriel Synnaeve, "Gradient matching for domain generalization," in *International Conference on Learning Representations*, 2022.

[19] Minghao Xu, Jian Zhang, Bingbing Ni, Teng Li, Chengjie Wang, Qi Tian, and Wenjun Zhang, "Adversarial domain adaptation with domain mixup," in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020, vol. 34, pp. 6502–6509.

[20] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky, "Domain-adversarial training of neural networks," *The journal of machine learning research*, vol. 17, no. 1, pp. 2096–2030, 2016.

[21] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao, "Deep domain generalization via conditional invariant adversarial networks," in *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018, pp. 624–639.

[22] Gilles Blanchard, Aniket Anand Deshmukh, Ürun Dogan, Gyemin Lee, and Clayton Scott, "Domain generalization by marginal transfer learning," *The Journal of Machine Learning Research*, vol. 22, no. 1, pp. 46–100, 2021.

[23] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville, "Out-of-distribution generalization via risk extrapolation (rex)," in *International Conference on Machine Learning*. PMLR, 2021, pp. 5815–5826.

[24] Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang, "Self-challenging improves cross-domain generalization," in *European Conference on Computer Vision*. Springer, 2020, pp. 124–140.

[25] Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo, "Reducing domain gap by reducing style bias," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8690–8699.

[26] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales, "Deeper, broader and artier domain generalization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 5542–5550.