

Trade-off between reconstruction loss and feature alignment for domain generalization

Thuan Nguyen^{*†}, Boyang Lyu^{*}, Prakash Ishwar[†], Matthias Scheutz[‡] and Shuchin Aeron^{*}

^{*}Department of Electrical and Computer Engineering, Tufts University, Medford, MA 02155

[†]Department of Electrical and Computer Engineering, Boston University, Boston, MA 02215

[‡]Department of Computer Science, Tufts University, Medford, MA 02155

Email: Thuan.Nguyen@tufts.edu, Boyang.Lyu@tufts.edu, pi@bu.edu, Matthias.Scheutz@tufts.edu, Shuchin@ece.tufts.edu

Abstract—Domain generalization (DG) is a branch of transfer learning that aims to train the learning models on several seen domains and subsequently apply these pre-trained models to other unseen (unknown but related) domains. To deal with challenging settings in DG where both data and label of the unseen domain are not available at training time, the most common approach is to design the classifiers based on the domain-invariant representation features, *i.e.*, the latent representations that are unchanged and transferable between domains. Contrary to popular belief, we show that designing classifiers based on invariant representation features alone is necessary but insufficient in DG. Our analysis indicates the necessity of imposing a constraint on the reconstruction loss induced by representation functions to preserve most of the relevant information about the label in the latent space. More importantly, we point out the trade-off between minimizing the reconstruction loss and achieving domain alignment in DG. Our theoretical results motivate a new DG framework that jointly optimizes the reconstruction loss and the domain discrepancy. Both theoretical and numerical results are provided to justify our approach.

I. INTRODUCTION

Domain Generalization (DG) has been widely studied over the past decade [1], [2]. Similar to Domain Adaptation (DA), DG aims to design a classifier based on one or several seen (source) domains and then apply these pre-trained classifiers to the unseen (target) domains. While accessing the data from unseen (target) domains is allowed in DA, this is strictly prohibited in DG, leading to a more challenging problem.

In practice, without any knowledge about unseen domains, one of the most common methods in DG is first to look for domain-invariant features, *i.e.*, the features that are general and transferable between domains, then design pre-trained classifiers based on these features. This approach is known as domain-invariant or domain-alignment representation learning and is widely considered as one of the most promising and efficient approaches in DG (please see our short survey in Sec. II). Though domain-invariant representation learning is possible to learn invariant features, recent work [3] shows that this technique does not account for the information loss caused by non-invertible representation maps which consequently motivates the use of (nearly) invertible representation maps [3], [4].

In this paper, based on the theme of domain-invariant representation learning and the information-theoretic point of view, we indicate the necessity of imposing a constraint on the

reconstruction loss induced by representation mappings to retain the relevant information about the label in learned features¹. In addition, we show that there is a trade-off between minimizing the reconstruction loss and minimizing the discrepancy between domains.

In this paper, our contributions include:

- 1) We derive a lower bound on mutual information between the latent representation and their labels to demonstrate the necessity of imposing a constraint on reconstruction loss in DG [Proposition 1].
- 2) We characterize the trade-off between minimizing the reconstruction loss *vs.* minimizing the discrepancy of joint distributions between domains. In other words, we show that it is impossible to perfectly accomplish these two objectives at the same time [Proposition 2].
- 3) We propose a new DG learning framework that directly accounts for both the reconstruction loss and the discrepancy between domains and demonstrate the efficiency of our proposed framework on several datasets.

The remainder of this paper is structured as follows. In Section II, we provide a short survey of existing works on DG. In Section III, we formally describe the problem formulation and clarify the notations. Section IV provides some initial results which support to our main results in Section V. The practical approach is described in Section VI. Finally, we provide the numerical results in Section VII and conclude in Section VIII.

II. RELATED WORK

The most common approach in DG is domain-invariant representation learning which aims to extract domain-invariant features and then design a classifier based on these features [5]–[13]. Domain-invariant approach, however, requires two key assumptions: (a) the domain-invariant features must exist and be shared between domains, and (b) the domain-invariant features must be strongly correlated with labels [5], [14]–[16]. In addition, to precisely obtain the domain-invariant features, one usually requires the availability of a sufficiently large

¹Note that a smaller reconstruction loss between input data and its representation implies the representation function is nearly invertible. Thus, our results agree with the work in [3], [4] that using nearly invertible representation functions is necessary for DG.

number of seen domains at training time [16], [17]. Therefore, if (a) the invariant features are neither existent nor strongly correlated with the label, or (b) the number of observed (seen) domains is not large enough, domain-invariant methods may fail [18]–[21].

Domain-invariant representation learning can be categorized into two main branches: (a) marginal distribution-invariant methods (covariate-alignment), *i.e.*, learning the features such that their distributions are unchanged according to domains, and (b) conditional distribution-invariant methods (concept-alignment), *i.e.*, learning the features such that the conditional distributions of labels given features are stable from domain to domain. The first branch (covariate-alignment) includes the works in [22]–[30]. Particularly, in [24], [27], the authors employ deep neural networks to learn transformations such that the differences between variances of transformed features over seen domains are minimized. Similarly, Li *et al.* [25] propose a method called Maximum Mean Discrepancy (MMD) that aims to minimize the maximum of the mean discrepancy between marginal distributions in seen domains. Sun and Saenko [31] propose a method that not only matches the mean but also synchronizes the covariance of feature distributions over different domains. Shen *et al.* [30] minimize the Wasserstein distance between marginal distributions of representation variables from different seen domains in latent space to extract invariant features. In a new approach, Bui *et al.* [15] desire to learn domain-invariant features (with marginal distributions unchanged according to domains) together with domain-specific features to enhance the generalization performance. The second branch (concept-alignment) includes the works in [5]–[8], [13], [32]–[34]. Particularly, linear/non-linear Invariant Risk Minimization algorithms are proposed in [5], [6] to find a common optimal linear/non-linear classifier over all observed domains under a key assumption that a common optimal classifier exists if the conditional distributions of the learned features are stable from domain to domain. Li *et al.* [8] propose a method to extract domain-invariant features via minimizing the mutual information of label given extracted feature for a given domain. Wang *et al.* [32] minimize the Kullback–Leibler (KL) divergence between conditional distributions in each class to obtain domain-invariant features. There are also several works that aim to simultaneously achieve both covariate-alignment and concept-alignment by minimizing the divergence of joint distributions between domains [25], [35]–[37].

It is worth noting that domain-invariant methods may fail under some particular settings, for example, if the labels more strongly correlate with the spurious features than with the true invariant features [7], [16]. To prevent the failure of learning models in these particular scenarios, Ahuja *et al.* suggest adding a constraint on the entropy of extracted features to capture the true invariant features [7]. In a similar vein, Nguyen *et al.* employ the conditional entropy minimization principle to eliminate the spurious-invariant features [16].

III. PROBLEM FORMULATION

A. Notations

Let \mathcal{X} , \mathcal{Z} , \mathcal{Y} denote the input space, the representation space, and the label space, respectively. For a given family of domains \mathcal{D} , suppose that the data from S observed (seen) domains $\mathcal{D}^{(1)}, \mathcal{D}^{(2)}, \dots, \mathcal{D}^{(S)} \in \mathcal{D}$ is accessible, DG tasks aim to learn a representation function $f : \mathcal{X} \rightarrow \mathcal{Z}$ followed by a classifier $g : \mathcal{Z} \rightarrow \mathcal{Y}$ that generalizes well on an unseen domain $\mathcal{D}^{(u)} \in \mathcal{D}$, $u \neq 1, 2, \dots, S$.

Let X denote the input random variable, $Z = f(X)$ denote the extracted feature random variable, Y denote the label random variable in input space, representation space, and label space, respectively. We use superscription $^{(i)}$ to denote the variables and functions specified on domain $\mathcal{D}^{(i)}$. For example, we use $p^{(i)}(\mathbf{x})$, $p^{(i)}(\mathbf{z})$, $p^{(i)}(\mathbf{x}, \mathbf{z})$ to denote the distribution of input sample \mathbf{x} , the distribution of feature sample $\mathbf{z} = f(\mathbf{x})$, and their joint distribution on $\mathcal{D}^{(i)}$, respectively. We use $p^{(i)}(X, Z)$ and $p^{(i)}(Y, Z)$ to denote the joint distribution between input random variable X and its representation random variable Z and the joint distribution between label random variable Y and representation random variable Z in $\mathcal{D}^{(i)}$. Finally, we use $H(A|B)$ and $I(A; B)$ to denote the conditional entropy and mutual information between two random variables A and B , respectively.

B. Problem formulation

For given S seen domains, a DG task aims to find an optimal representation function f^* by solving the following optimization problem:

$$\min_{f: \mathcal{X} \rightarrow \mathcal{Z}} \mathbf{R}^{(u)}(g_f \circ f) \quad (1)$$

where $\mathbf{R}^{(u)}(g_f \circ f)$ denotes the risk (classification error) introduced by using a representation map f followed by an optimal classifier g_f on unseen domain $\mathcal{D}^{(u)}$. Note that for a given $f : \mathcal{X} \rightarrow \mathcal{Z}$, the optimal classifier $g_f : \mathcal{Z} \rightarrow \mathcal{Y}$ completely depends on f .

In this paper, under the information-theoretic point of view, we want to solve the following optimization problem:

$$\max_{f: \mathcal{X} \rightarrow \mathcal{Z}} I^{(u)}(Y; Z) \quad (2)$$

where $I^{(u)}(Y; Z)$ denotes mutual information between the labels and representation features on unseen domain $\mathcal{D}^{(u)}$. It is worth noting that a higher mutual information between representation features and its labels likely leads to a higher classification accuracy. Thus, solving (2) acts as a proxy for minimizing the classification risk on the unseen domain which is the ultimate goal of DG tasks in (1).

IV. PRELIMINARY

This section provides some definitions and preliminary results that support our main results in Sec. V.

A. Measure of domain discrepancy

Under DG settings, one only can access to the data from seen domains, therefore, the most common approach is first to learn domain-invariant features and then design a classifier based on these features [5]–[13]. For a given divergence measure $D(\cdot||\cdot)$ and a seen domain $\mathcal{D}^{(s)}$, previous works on DG usually aim to (a) enforce covariate-alignment, *i.e.*, minimizing $D(p^{(u)}(Z)||p^{(s)}(Z))$, or (b) enforce concept-alignment, *i.e.*, minimizing $D(p^{(u)}(Y|Z)||p^{(s)}(Y|Z))^2$. We, however, want to learn a mapping f to minimize the mismatch between joint distributions of seen and unseen domains³.

Definition 1 [Domain discrepancy induced by a representation function]. *For a representation function $f : \mathcal{X} \rightarrow \mathcal{Z}$, unseen domain $\mathcal{D}^{(u)}$, and seen domain $\mathcal{D}^{(s)}$, the domain-discrepancy between $\mathcal{D}^{(u)}$ and $\mathcal{D}^{(s)}$ induced by f is:*

$$K(f) = D(p^{(u)}(Y, Z)||p^{(s)}(Y, Z)) \quad (3)$$

where $D(\cdot||\cdot)$ is a divergence measure that quantifies the mismatch between two distributions.

If the mapping f induces $K(f) = 0$, the distributions between seen and unseen domains are perfectly aligned. In practice, one usually wants to enforce $K(f) \leq \epsilon$ where ϵ is a positive number.

Definition 2. *Let $W(\epsilon)$ denote the maximum discrepancy between mutual information of unseen domain $\mathcal{D}^{(u)}$ and seen domain $\mathcal{D}^{(s)}$ when the domain discrepancy $K(f)$ does not exceed a positive number ϵ . Formally,*

$$W(\epsilon) = \max_{\substack{f: \mathcal{X} \rightarrow \mathcal{Z}, \\ K(f) \leq \epsilon}} |I^{(u)}(Y; Z) - I^{(s)}(Y; Z)| \quad (4)$$

where $I^{(u)}(Y; Z)$ and $I^{(s)}(Y; Z)$ are mutual information between label Y and representation feature Z in unseen domain and seen domain, respectively.

If $\epsilon = 0$, $K(f) = 0$, $I^{(u)}(Y; Z) = I^{(s)}(Y; Z)$, and $W(0) = 0$. In addition, it is possible to verify that $W(\epsilon)$ is a monotonically increasing function of ϵ .

B. Measure of reconstruction loss

Note that by Data Processing Inequality [38] and the fact that $Y \rightarrow X \rightarrow Z$ forms a Markov chain, for any representation function f :

$$I^{(u)}(Y; X) \geq I^{(u)}(Y; Z) \quad (5)$$

where $I^{(u)}(Y; X)$ and $I^{(u)}(Y; Z)$ denote mutual information between label and input and mutual information between label and feature on unseen domain, respectively. The equality happens in (5) if f is invertible.

²Under DG settings, one cannot directly align the distribution from the unseen domain, therefore, aligning the distribution over all seen domains is usually used as a proxy to achieve this goal.

³Even though this condition is restricted, our examples in Appendix A show that achieving covariate-alignment or concept-alignment alone is not enough to guarantee a small classification risk on unseen domain.

It is worth noting that there may exist non-invertible representation functions that make the equality happens. Indeed, if the label information can be precisely preserved under mapping f , *i.e.*, using Z to predict Y is as good as using X to predict Y , for example, if $H^{(u)}(Y|X) = H^{(u)}(Y|Z)$, then $I^{(u)}(Y; X) = I^{(u)}(Y; Z)$. However, under DG settings, there is no information about the data, nor the label from unseen domains. Thus, it is impossible to design such non-invertible mappings that perfectly preserve the label information on unseen domain. On the other hand, (nearly) invertible mappings can be constructed regardless of domains which is a possible way to retain the useful information on unseen domains, *i.e.*, making $I^{(u)}(Y; Z)$ close to $I^{(u)}(Y; X)$. In practice, (nearly) invertible mappings can be enforced by minimizing the reconstruction loss between input data and its representation.

Definition 3 [Reconstruction loss]. *For a representation function $f : \mathcal{X} \rightarrow \mathcal{Z}$, and a function $\theta : \mathcal{Z} \rightarrow \mathcal{X}$, the reconstruction loss (on unseen domain) induced by f and θ is defined by:*

$$\begin{aligned} R(f, \theta) &= \int_{\mathbf{x} \in \mathcal{X}} p^{(u)}(\mathbf{x}) \ell(\mathbf{x}, \theta(f(\mathbf{x}))) d\mathbf{x} \\ &= \int_{\mathbf{x} \in \mathcal{X}} \int_{\mathbf{z} \in \mathcal{Z}} p^{(u)}(\mathbf{x}, \mathbf{z}) \ell(\mathbf{x}, \theta(\mathbf{z})) d\mathbf{x} d\mathbf{z} \end{aligned} \quad (6)$$

where $\ell(\cdot, \cdot)$ is a distortion function.

Usually, f is called *encoder* and θ is called *decoder*.

Definition 4. *Let $Q(\gamma)$ denote the maximum mutual information loss (on unseen domain) when the reconstruction loss induced by encoder f and decoder θ does not exceed a positive number γ . Formally,*

$$Q(\gamma) = \max_{\substack{f: \mathcal{X} \rightarrow \mathcal{Z}, \\ \theta: \mathcal{Z} \rightarrow \mathcal{X}, \\ R(f, \theta) \leq \gamma}} I^{(u)}(Y; X) - I^{(u)}(Y; Z). \quad (7)$$

Note that $\gamma = 0$ implies f is invertible, leading to $I^{(u)}(Y; X) = I^{(u)}(Y; Z)$. Therefore, $Q(0) = 0$. In addition, it is possible to show that $Q(\gamma)$ is a monotonic increasing function of γ .

V. MAIN RESULTS

Based on definitions and initial results in Sec. IV, we point out the necessity of employing the representation functions such that a small reconstruction loss is induced in order to solve the optimization problem in (2). More interestingly, we show that there is a trade-off between minimizing the reconstruction loss and aligning the joint distributions between domains.

Proposition 1 [Main result]. *For unseen domain $\mathcal{D}^{(u)}$, seen domain $\mathcal{D}^{(s)}$, and any encoder f and decoder θ :*

$$I^{(u)}(Y; Z) \geq \max \left[I^{(s)}(Y; Z) - W(K(f)); \right. \\ \left. I^{(u)}(Y; X) - Q(R(f, \theta)) \right].$$

Proof. Please see Appendix B. □

Proposition 1 points out a possible way to solve the optimization problem proposed in (2). Particularly, to maximize $I^{(u)}(Y; Z)$, one simultaneously needs to (a) maximize $I^{(s)}(Y; Z) - W(K(f))$, and (b) maximize $I^{(u)}(Y; X) - Q(R(f, \theta))$. Since $I^{(u)}(Y; X)$ is a constant, $W(\cdot)$ and $Q(\cdot)$ are monotonically increasing functions, we need to find an encoder f and a decoder θ to maximize the mutual information on seen domain $I^{(s)}(Y; Z)$, minimize the domain discrepancy $K(f)$, and minimize the reconstruction loss $R(f, \theta)$, at the same time.

In practice, if the invariant features exist, strongly correlate with the label, and can be precisely learned, there may exist a mapping f such that $I^{(s)}(Y; Z)$ is large and $K(f)$ is small which make the first lower bound $I^{(s)}(Y; Z) - W(K(f))$ is tighter than the second lower bound $I^{(u)}(Y; X) - Q(R(f, \theta))$. However, under some failure cases in literature [18]–[21], for example, if the invariant feature does not exist, or if the invariant feature is not strongly correlated with the label, then $K(f)$ is large and $I^{(s)}(Y; Z)$ is small, and the second lower bound $I^{(u)}(Y; X) - Q(R(f, \theta))$ might be the tighter one. Thus, the traditional approaches that simultaneously target to learn the invariant features (minimizing $K(f)$) and minimize the empirical risk (a proxy for maximizing the mutual information on seen domain $I^{(s)}(Y; Z)$) for optimizing the first lower bound, can fail.

Motivated by Proposition 1, one wants to design an encoder f and a decoder θ to simultaneously minimize both $K(f)$ and $R(f, \theta)$. Unfortunately, we show that it is impossible to optimize $K(f)$ and $R(f, \theta)$ at the same time.

Definition 5 [Reconstruction-alignment function]. *For unseen domain $\mathcal{D}^{(u)}$, seen domain $\mathcal{D}^{(s)}$, and a given decoder θ , the reconstruction-alignment function $T(\gamma)$ is defined by:*

$$\begin{aligned} T(\gamma) &= \min_{f: \mathcal{X} \rightarrow \mathcal{Z}} K(f) = \min_{f: \mathcal{X} \rightarrow \mathcal{Z}} D(p^{(u)}(Y, Z) \| p^{(s)}(Y, Z)) \\ \text{s.t. } R(f, \theta) &= \int_{\mathbf{x} \in \mathcal{X}} \int_{\mathbf{z} \in \mathcal{Z}} p^{(u)}(\mathbf{x}, \mathbf{z}) \ell(\mathbf{x}, \theta(\mathbf{z})) d\mathbf{x} d\mathbf{z} \leq \gamma \end{aligned} \quad (8)$$

where γ is a positive number, $\ell(\cdot, \cdot)$ is a distortion measure, and $D(\cdot \| \cdot)$ is a divergence measure.

The reconstruction-alignment function $T(\gamma)$ is the minimal discrepancy between the joint distributions of the unseen domain $\mathcal{D}^{(u)}$ and seen domain $\mathcal{D}^{(s)}$ that can be obtained if the reconstruction loss (on unseen domain) does not exceed a positive number γ . We formally characterize the trade-off between minimizing reconstruction loss and achieving domain alignment as below.

Proposition 2 [Main result]. *If the divergence measure $D(a \| b)$ is convex (in both variables a and b), then $T(\gamma)$ defined in (8) is (a) monotonically non-increasing, and (b) convex.*

Proof. Please see Appendix C. \square

Sharing some similarities with rate-distortion theory [38], Proposition 2 characterizes the trade-off between minimizing the domain discrepancy $K(f)$ and minimizing the reconstruction loss $R(f, \theta)$. Since Proposition 2 holds for any decoder θ ,

there is no encoder f and decoder θ that can perfectly minimize the domain discrepancy and the reconstruction loss.

In addition, though the proof of Proposition 2 is constructed by considering the reconstruction loss on unseen domain, a similar proof holds for seen domains, *i.e.*, there is a universal trade-off between minimizing the domain discrepancy and minimizing the reconstruction loss regardless of domain. Finally, it is worth noting that the assumption about the convexity of the divergence $D(\cdot \| \cdot)$ is not too restricted in practice. Indeed, most of the divergence measures, for example, the Kullback-Leibler divergence, are convex [38].

VI. PRACTICAL APPROACH

In this section, motivated by Proposition 1, we propose a framework that simultaneously optimizes the domain discrepancy, the reconstruction loss, and the empirical risk on seen domains⁴. Specifically, we want to minimize the following loss function:

$$\min_{f, g_f, \theta} \sum_{i=1}^M \mathbf{R}^{(i)}(g_f \circ f) + \alpha L_{\text{discrepancy}}(f) + \beta L_{\text{reconstruction}}(f, \theta), \quad (9)$$

where the first term is the empirical classification risk over M seen domains, the second term denotes the domain discrepancy, and the third term represents the reconstruction loss. α , and β are two positive hyper-parameters that control the trade-off between minimizing these three loss terms.

Compared to most of the existing works in DG, the main difference of our objective function in (9) comes from the reconstruction loss term which is added in light of Proposition 1 to preserve the information between the latent representation and its labels on unseen domain. Therefore, (9) can be practically optimized by adding a decoder (for optimizing the reconstruction loss) into the well-established existing DG models that already handle the empirical risk and the domain discrepancy terms. Practically, we employ the following DG methods: Invariant Risk Minimization (IRM) algorithm [5], Maximum Mean Discrepancy (MMD) algorithm [25], CORrelation ALignment (CORAL) algorithm [31], Invariant Risk Minimization-Maximum Mean Discrepancy (IRM-MMD) algorithm [37], Information Bottleneck-Invariant Risk Minimization (IB-IRM) algorithm [7], Empirical Risk Minimization (ERM) algorithm [39], and Conditional Entropy Minimization (CEM) algorithm [16] to minimize the first two terms in (9)⁵.

To minimize the reconstruction loss term in (9), we train an encoder $f: \mathcal{X} \rightarrow \mathcal{Z}$ together with a decoder $\theta: \mathcal{Z} \rightarrow \mathcal{X}$ to minimize:

$$L_{\text{reconstruction}}(f, \theta) = \sum_{i=1}^M \int_{\mathbf{x} \in \mathcal{X}} p^{(i)}(\mathbf{x}) \ell(\mathbf{x}, \theta(f(\mathbf{x}))) d\mathbf{x}, \quad (10)$$

⁴Here, minimizing the empirical risk (on seen domains) is considered as a proxy for maximizing the mutual information between label and representation features. Using mutual information as a direct objective function will be kept in our future work.

⁵Due to the limited time, we randomly select some algorithms from recent works on DG to add the reconstruction loss term. We encourage the reader to find the details of these algorithms in [5], [7], [16], [25], [31], [37].

where the squared-Euclidean distance is selected as the distortion measure, *i.e.*, $\ell(a, b) = (a - b)^2$, and $p^{(i)}(\mathbf{x})$ denotes the input distribution on domain $\mathcal{D}^{(i)}$, $i = 1, 2, \dots, M$.

By adding the reconstruction loss term into IRM, MMD, CORAL, IRM-MMD, IB-IRM, ERM, and CEM algorithms, the following new algorithms are constructed: IRM-Rec, MMD-Rec, CORAL-Rec, IRM-MMD-Rec, IB-IRM-Rec, ERM-Rec, and CEM-Rec, respectively. One of the advantages of employing multiple algorithms for dealing with the first two terms in (9) is that it allows us to evaluate the effectiveness of combining the reconstruction-loss term on a variety of DG methods. Indeed, our numerical results in the next section show that adding the reconstruction loss term leads to improvements in the accuracy of existing DG methods.

VII. EXPERIMENTS

A. Datasets

Colored-MNIST (CMNIST) [5]. The CMNIST dataset is a common DG dataset which was first proposed in [5]. The learning task is to classify a colored digit into two classes: the digit is less than or equal to four or the digit is strictly greater than four. There are three domains in CMNIST, two domains contain 25,000 images each and one domain contains 20,000 images. Here, the color is considered as a spurious feature which is added in a way such that the label is more correlated with the color than with the digit. Due to a strong spurious correlation between colors and labels, any algorithm simply aims to minimize the training error will tend to discover the color rather than the shape of the digit (on seen domains) and therefore fail in the test on unseen domains. More details about the CMNIST dataset can be found in [5].

Covariate-Shift-CMNIST (CS-CMNIST) [40]. The CS-CMNIST dataset is a dataset derived from CMNIST dataset which was first introduced in [40]. There are 10 classes in CS-CMNIST dataset where each class corresponds to a digit from zero to nine and each digit is associated with a single color. There are three domains in CS-CMNIST: two training domains and one testing domain, each containing 20,000 images. The color is considered the spurious feature and is added in a way such that the color is more correlated to digits on seen domains than on unseen domains. More detail about CS-CMNIST can be found in [7], [40].

B. Compared Methods

As previously discussed in Sec. VI, by adding the reconstruction loss term, we compare the proposed IRM-Rec, MMD-Rec, CORAL-Rec, IRM-MMD-Rec, IB-IRM-Rec, ERM-Rec, and CEM-Rec algorithms against their original IRM [5], MMD [25], CORAL [31], IRM-MMD [37], IB-IRM [7], ERM [39], and CEM [16] algorithms.

C. Implementation Details

For the CMNIST dataset, we utilize the excellent implementation in Domainbed [21] that employs the MNIST-ConvNet with four convolutional layers as the learning model. 20 trials corresponding to 20 pairs of hyper-parameters α and β are

randomly selected in $[10^{-1}, 10^4]$. For each trial, the learning rate is randomly picked in $[10^{-4.5}, 10^{-3.5}]$ while the batch size is randomly selected in $[2^3, 2^9]$.

Since the CS-CMNIST dataset is not available in Domainbed [21], we follow the implementation proposed in [7] where the learning model is composed of three convolutional layers with feature map dimensions of 256, 128, and 64, respectively. The last layer (linear layer) is used to classify the colored digit back to 10 classes corresponding to 10 digits from zero to nine. We use an SGD optimizer for training with a batch size fixed to 128, the learning rate fixed to 10^{-1} and decay every 600 steps with the total number of steps set to 2,000. In contrast to CMNIST, a grid search is performed in CS-CMNIST with $\alpha, \beta \in \{0.1, 1, 10, 10^2, 10^3, 10^4\}$.

The training-domain validation set procedure is used for model selection, *i.e.*, selecting the hyper-parameters (the models) that induce the highest validation accuracy on the validation set sampled from seen domains [7], [21].

We repeat the whole experiment three times for CMNIST and five times for CS-CMNIST via selecting different random seeds⁶. For each selected random seed, the whole process of hyper-parameters tuning and model selection is repeated. After the whole process is finished, only the average accuracy and its corresponding standard deviation are reported. Our code can be found at this link⁷.

D. Results and Discussion

Algorithm	IRM [5]	IB-IRM [7]	MMD-IRM [37]	CEM [16]
Accuracy	61.5 \pm 1.5	71.8 \pm 0.7	77.2 \pm 0.9	85.7 \pm 0.9
Algorithm	IRM-Rec	IB-IRM-Rec	MMD-IRM-Rec	CEM-Rec
Accuracy	71.0 \pm 0.8	75.6 \pm 1.1	79.7 \pm 0.6	87.1 \pm 1.3

TABLE I
Average accuracy (%) of compared methods on CS-CMNIST dataset.

Algorithm	IRM [5]	MMD [25]	ERM [39]	CORAL [31]
Accuracy	52.0 \pm 0.1	51.5 \pm 0.2	51.5 \pm 0.1	51.5 \pm 0.1
Algorithm	IRM-Rec	MMD-Rec	ERM-Rec	CORAL-Rec
Accuracy	51.7 \pm 0.2	51.7 \pm 0.1	51.8 \pm 0.1	52.0 \pm 0.1

TABLE II
Average accuracy (%) of compared methods on CMNIST dataset.

Table I and II provide the accuracy of compared methods on CS-CMNIST dataset and CMNIST dataset, respectively.

As seen, for the CS-CMNIST dataset, the accuracy of all four tested algorithms has been improved when the reconstruction loss term is added. Particularly, the lowest improvement is 1.4% observed from CEM algorithm [16], while the largest improvement appears in IRM algorithm [5] with the gain

⁶We follow the settings in [7], [21]. Particularly, in [21], the experiment is repeated three times while in [7], the experiment is repeated five times.

⁷https://github.com/thuan2412/tradeoff_between_domain_alignment_and_reconstruction_loss

of 9.5%. We believe that the difference in the improvement between the tested algorithms can be explained by Proposition 1. Indeed, it seems like the original CEM algorithm [16] already works well on CS-CMNIST, therefore, the first lower bound in Proposition 1 induced by CEM is pretty tight which leads to a small improvement in accuracy when the reconstruction loss term is added. On the other hand, since IRM algorithm [5] performs poorly on CS-CMNIST, we suspect that the first lower bound in Proposition 1 induced by IRM is the looser one, leading to a large improvement when the reconstruction loss term is added for optimizing the second bound.

Compared to CS-CMNIST, CMNIST is a more challenging dataset where all tested algorithms perform poorly. Indeed, because there exists a strong spurious correlation between the colors and the labels of digits in CMNIST, there is no algorithm that works well on CMNIST [21]. However, as observed from Table II, three out of four tested algorithms have been improved by adding the reconstruction loss term when tested on CMNIST dataset. Even though the improvement is not substantial with the largest margin being only 0.5% from the CORAL algorithm, this still demonstrates the usefulness of optimizing the reconstruction loss term in DG.

It is worth noting that the numerical results for IRM, MMD, CORAL, and ERM on the CMNIST dataset are collected from [21] while the numerical results for IRM, IB-IRM, and CEM on the CS-CMNIST dataset are collected from [16]. Since the source code of MMD-IRM [37] was not released, we implemented this algorithm by our-self in order to construct the MMD-IRM-Rec algorithm.

Finally, our future work will focus on integrating the reconstruction loss into other state-of-the-art DG algorithms. Of course, using mutual information as a direct objective function instead of empirical risk will also be considered as one of our future works.

VIII. CONCLUSIONS

In this paper, we showed that learning domain-invariant representation features is necessary in DG but insufficient to preserve the mutual information between label and representation features on unseen domains. This fact suggests for imposing a constraint on the reconstruction loss between the input and its latent representation in order to preserve most of the relevant information about labels. More importantly, we also point out the trade-off between minimizing the reconstruction loss and achieving domain alignment in DG. In other words, it is impossible to minimize both the reconstruction loss and the domain discrepancy at the same time. Our theoretical results motivate a new practical framework that jointly accounts for both the reconstruction loss and the domain discrepancy to learn the optimal representation features. In practice, our proposed algorithms can provide a comparable or slightly better performance compared to state-of-the-art DG methods.

REFERENCES

- [1] J. Wang, C. Lan, C. Liu, Y. Ouyang, T. Qin, W. Lu, Y. Chen, W. Zeng, and P. Yu, "Generalizing to unseen domains: A survey on domain generalization," *IEEE Transactions on Knowledge and Data Engineering*, 2022.
- [2] K. Zhou, Z. Liu, Y. Qiao, T. Xiang, and C. C. Loy, "Domain generalization: A survey," *arXiv preprint arXiv:2103.02503*, 2021.
- [3] F. D. Johansson, D. Sontag, and R. Ranganath, "Support and invertibility in domain-invariant representations," in *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, 2019, pp. 527–536.
- [4] B. Lyu, T. Nguyen, P. Ishwar, M. Scheutz, and S. Aeron, "Barycentric-alignment and invertibility for domain generalization," *arXiv preprint arXiv:2109.01902*, 2021.
- [5] M. Arjovsky, L. Bottou, I. Gulrajani, and D. Lopez-Paz, "Invariant risk minimization," *arXiv preprint arXiv:1907.02893*, 2019.
- [6] C. Lu, Y. Wu, J. M. Hernández-Lobato, and B. Schölkopf, "Nonlinear invariant risk minimization: A causal approach," *arXiv preprint arXiv:2102.12353*, 2021.
- [7] K. Ahuja, E. Caballero, D. Zhang, J.-C. Gagnon-Audet, Y. Bengio, I. Mitliagkas, and I. Rish, "Invariance principle meets information bottleneck for out-of-distribution generalization," *Advances in Neural Information Processing Systems*, vol. 34, pp. 3438–3450, 2021.
- [8] B. Li, Y. Shen, Y. Wang, W. Zhu, D. Li, K. Keutzer, and H. Zhao, "Invariant information bottleneck for domain generalization," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 7, 2022, pp. 7399–7407.
- [9] Y. Du, J. Xu, H. Xiong, Q. Qiu, X. Zhen, C. G. Snoek, and L. Shao, "Learning to learn with variational information bottleneck for domain generalization," in *European Conference on Computer Vision*. Springer, 2020, pp. 200–216.
- [10] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky, "Domain-adversarial training of neural networks," *The journal of machine learning research*, vol. 17, no. 1, pp. 2096–2030, 2016.
- [11] D. Mahajan, S. Tople, and A. Sharma, "Domain generalization using causal matching," in *International Conference on Machine Learning*. PMLR, 2021, pp. 7313–7324.
- [12] F. Zhou, Z. Jiang, C. Shui, B. Wang, and B. Chaib-draa, "Domain generalization with optimal transport and metric learning," *arXiv preprint arXiv:2007.10573*, 2020.
- [13] K. Ahuja, K. Shanmugam, K. Varshney, and A. Dhurandhar, "Invariant risk minimization games," in *International Conference on Machine Learning*. PMLR, 2020, pp. 145–155.
- [14] V. Nagarajan, A. Andreassen, and B. Neyshabur, "Understanding the failure modes of out-of-distribution generalization," *arXiv preprint arXiv:2010.15775*, 2020.
- [15] M.-H. Bui, T. Tran, A. Tran, and D. Phung, "Exploiting domain-specific features to enhance domain generalization," *Advances in Neural Information Processing Systems*, vol. 34, 2021.
- [16] T. Nguyen, B. Lyu, P. Ishwar, M. Scheutz, and S. Aeron, "Conditional entropy minimization principle for learning domain invariant representation features," *arXiv preprint arXiv:2201.10460*, 2022.
- [17] Y. Chen, E. Rosenfeld, M. Sellke, T. Ma, and A. Risteski, "Iterative feature matching: Toward provable domain generalization with logarithmic environments," *arXiv preprint arXiv:2106.09913*, 2021.
- [18] E. Rosenfeld, P. Ravikumar, and A. Risteski, "The risks of invariant risk minimization," in *International Conference on Learning Representations*, vol. 9, 2021.
- [19] B. Aubin, A. Slowik, M. Arjovsky, L. Bottou, and D. Lopez-Paz, "Linear unit-tests for invariance discovery," *arXiv preprint arXiv:2102.10867*, 2021.
- [20] P. Kamath, A. Tangella, D. Sutherland, and N. Srebro, "Does invariant risk minimization capture invariance?" in *International Conference on Artificial Intelligence and Statistics*. PMLR, 2021, pp. 4069–4077.
- [21] I. Gulrajani and D. Lopez-Paz, "In search of lost domain generalization," in *International Conference on Learning Representations*, 2020.
- [22] P. Russo, F. M. Carlucci, T. Tommasi, and B. Caputo, "From source to target and back: symmetric bi-directional adaptive gan," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 8099–8108.
- [23] L. Hu, M. Kan, S. Shan, and X. Chen, "Duplex generative adversarial network for unsupervised domain adaptation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 1498–1507.

- [24] K. Muandet, D. Balduzzi, and B. Schölkopf, "Domain generalization via invariant feature representation," in *International Conference on Machine Learning*. PMLR, 2013, pp. 10–18.
- [25] H. Li, S. J. Pan, S. Wang, and A. C. Kot, "Domain generalization with adversarial feature learning," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5400–5409.
- [26] M. Ghifary, D. Balduzzi, W. B. Kleijn, and M. Zhang, "Scatter component analysis: A unified framework for domain adaptation and domain generalization," *IEEE transactions on pattern analysis and machine intelligence*, vol. 39, no. 7, pp. 1414–1430, 2016.
- [27] S. Erfani, M. Baktashmotlagh, M. Moshtaghi, X. Nguyen, C. Leckie, J. Bailey, and R. Kotagiri, "Robust domain generalisation by enforcing distribution invariance," in *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence (IJCAI-16)*. AAAI Press, 2016, pp. 1455–1461.
- [28] X. Jin, C. Lan, W. Zeng, and Z. Chen, "Feature alignment and restoration for domain generalization and adaptation," *arXiv preprint arXiv:2006.12009*, 2020.
- [29] G. Blanchard, A. A. Deshmukh, U. Dogan, G. Lee, and C. Scott, "Domain generalization by marginal transfer learning," *Journal of Machine Learning Research*, vol. 22, pp. 1–55, 2021.
- [30] J. Shen, Y. Qu, W. Zhang, and Y. Yu, "Wasserstein distance guided representation learning for domain adaptation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.
- [31] B. Sun and K. Saenko, "Deep coral: Correlation alignment for deep domain adaptation," in *European conference on computer vision*. Springer, 2016, pp. 443–450.
- [32] Z. Wang, M. Loog, and J. van Gemert, "Respecting domain relations: Hypothesis invariance for domain generalization," in *2020 25th International Conference on Pattern Recognition (ICPR)*. IEEE, 2021, pp. 9756–9763.
- [33] S. Zhao, M. Gong, T. Liu, H. Fu, and D. Tao, "Domain generalization via entropy regularization," *Advances in Neural Information Processing Systems*, vol. 33, pp. 16 096–16 107, 2020.
- [34] O. E. Salaudeen and O. O. Koyejo, "Exploiting causal chains for domain generalization," in *NeurIPS 2021 Workshop on Distribution Shifts: Connecting Methods and Applications*, 2021.
- [35] T. Nguyen, B. Lyu, P. Ishwar, M. Scheutz, and S. Aeron, "Joint covariate-alignment and concept-alignment: a framework for domain generalization," *arXiv preprint arXiv:2208.00898*, 2022.
- [36] W. Yang, T. Ling, C. Yang, L. Wang, Y. Shi, L. Zhou, and M. Yang, "Class distribution alignment for adversarial domain adaptation," *arXiv preprint arXiv:2004.09403*, 2020.
- [37] R. Guo, P. Zhang, H. Liu, and E. Kiciman, "Out-of-distribution prediction with invariant risk minimization: The limitation and an effective fix," *arXiv preprint arXiv:2101.07732*, 2021.
- [38] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.
- [39] V. N. Vapnik, "An overview of statistical learning theory," *IEEE transactions on neural networks*, vol. 10, no. 5, pp. 988–999, 1999.
- [40] K. Ahuja, J. Wang, A. Dhurandhar, K. Shanmugam, and K. R. Varshney, "Empirical or invariant risk minimization? a sample complexity perspective," in *International Conference on Learning Representations*, 2021.

APPENDIX

A. Covariate-alignment or concept-alignment alone is not sufficient to achieve Domain Generalization

In this section, we provide two small examples to show that achieving covariate-alignment or concept-alignment alone is not enough to minimize the classification error on unseen domains.

Example A.1 (Covariate-alignment alone is not enough). Suppose that there exists a mapping $f : \mathcal{X} \rightarrow \mathcal{Z}$ such that the marginal distributions of seen and unseen domains in the latent space are perfectly aligned. Particularly, we assume that $\mathcal{Z} = \{0, 1\}$, and $p^{(s)}(Z = 0) = p^{(u)}(Z = 0) = p^{(s)}(Z = 1) = p^{(u)}(Z = 1) = 0.5$. Next, suppose that there is a mismatch between the conditional

distribution between two domains (inadequate concept-alignment), for example,

$$\begin{aligned} p^{(s)}(Y = 0|Z = 0) &= 0.9, \\ p^{(s)}(Y = 1|Z = 0) &= 0.1, \\ p^{(s)}(Y = 0|Z = 1) &= 0.1, \\ p^{(s)}(Y = 1|Z = 1) &= 0.9, \end{aligned}$$

and

$$\begin{aligned} p^{(u)}(Y = 0|Z = 0) &= 0.1, \\ p^{(u)}(Y = 1|Z = 0) &= 0.9, \\ p^{(u)}(Y = 0|Z = 1) &= 0.9, \\ p^{(u)}(Y = 1|Z = 1) &= 0.1. \end{aligned}$$

If one trains a maximum likelihood classifier $g : \mathcal{Z} \rightarrow \mathcal{Y}$ on seen domain, then $g(Z = 0) = 0$ and $g(Z = 1) = 1$. The classification error on the seen domain induced by f and g is:

$$\begin{aligned} R^{(s)}(g \circ f) &= p^{(s)}(Z = 0)[1 - p^{(s)}(Y = 0|Z = 0)] \\ &+ p^{(s)}(Z = 1)[1 - p^{(s)}(Y = 1|Z = 1)] \\ &= 0.1. \end{aligned}$$

Next, if one applies this pre-trained classifier g to the unseen domain, the classification error is:

$$\begin{aligned} R^{(u)}(g \circ f) &= p^{(u)}(Z = 0)[1 - p^{(u)}(Y = 0|Z = 0)] \\ &+ p^{(u)}(Z = 1)[1 - p^{(u)}(Y = 1|Z = 1)] \\ &= 0.9. \end{aligned}$$

Therefore, covariate-alignment alone is not sufficient to guarantee a low classification error on unseen domain.

Example A.2 (Concept-alignment alone is not enough). Suppose that there exists a mapping $f : \mathcal{X} \rightarrow \mathcal{Z}$ such that the conditional distributions of seen and unseen domains in the latent space are perfectly aligned. Particularly, we assume that $\mathcal{Z} = \{0, 1\}$, and

$$\begin{aligned} p^{(s)}(Y = 0|Z = 0) &= p^{(u)}(Y = 0|Z = 0) = 0.9, \\ p^{(s)}(Y = 1|Z = 0) &= p^{(u)}(Y = 1|Z = 0) = 0.1, \\ p^{(s)}(Y = 0|Z = 1) &= p^{(u)}(Y = 0|Z = 1) = 0.49, \\ p^{(s)}(Y = 1|Z = 1) &= p^{(u)}(Y = 1|Z = 1) = 0.51. \end{aligned}$$

Next, suppose that there is a mismatch between the marginal distribution of two domains (inadequate covariate-alignment), for example, $p^{(s)}(Z = 0) = 0.9$, $p^{(s)}(Z = 1) = 0.1$ while $p^{(u)}(Z = 0) = 0.1$, $p^{(u)}(Z = 1) = 0.9$.

If one trains a maximum likelihood classifier $g : \mathcal{Z} \rightarrow \mathcal{Y}$ on seen domain, then $g(Z = 0) = 0$ and $g(Z = 1) = 1$. The classification error on seen domain induced by f and g is:

$$\begin{aligned} R^{(s)}(g \circ f) &= p^{(s)}(Z = 0)[1 - p^{(s)}(Y = 0|Z = 0)] \\ &+ p^{(s)}(Z = 1)[1 - p^{(s)}(Y = 1|Z = 1)] \\ &= 0.139. \end{aligned}$$

Next, if one transfers this pre-trained classifier g to the unseen domain, the classification error is:

$$\begin{aligned} R^{(u)}(g \circ f) &= p^{(u)}(Z = 0)[1 - p^{(u)}(Y = 0|Z = 0)] \\ &+ p^{(u)}(Z = 1)[1 - p^{(u)}(Y = 1|Z = 1)] \\ &= 0.451. \end{aligned}$$

Therefore, achieving concept-alignment alone does not ensure a low classification error on unseen domain.

B. Proof of Proposition 1

First, from Definition 2, for a given f :

$$W(K(f)) \geq I^{(s)}(Y; Z) - I^{(u)}(Y; Z), \quad (11)$$

which is equivalent to:

$$I^{(u)}(Y; Z) \geq I^{(s)}(Y; Z) - W(K(f)). \quad (12)$$

Next, from Definition 4, for given f and θ :

$$Q(R(f, \theta)) \geq I^{(u)}(Y; X) - I^{(u)}(Y; Z) \quad (13)$$

which is equivalent to:

$$I^{(u)}(Y; Z) \geq I^{(u)}(Y; X) - Q(R(f, \theta)). \quad (14)$$

Combine (12) and (14), the proof follows.

C. Proof of Proposition 2

First, it is worth noting that our proof closely follows to the proof of rate-distortion theory in [38].

Particularly, consider two positive numbers γ_1 and γ_2 , and assume that $\gamma_1 \leq \gamma_2$. For a given decoder θ , let \mathcal{F}_{γ_1} and \mathcal{F}_{γ_2} denote the sets of representation functions f such that $R(f, \theta) \leq \gamma_1$ and $R(f, \theta) \leq \gamma_2$, respectively. From $\gamma_1 \leq \gamma_2$, $\mathcal{F}_{\gamma_1} \subset \mathcal{F}_{\gamma_2}$. Therefore:

$$T(\gamma_1) = \min_{f \in \mathcal{F}_{\gamma_1}} K(f) \geq \min_{f \in \mathcal{F}_{\gamma_2}} K(f) = T(\gamma_2). \quad (15)$$

Thus, $T(\gamma)$ is a monotonically non-increasing function of γ .

Next, let:

$$f_1 = \arg \min_{f: \mathcal{X} \rightarrow \mathcal{Z}} K(f) \quad \text{s.t.} \quad R(f, \theta) \leq \gamma_1, \quad (16)$$

$$f_2 = \arg \min_{f: \mathcal{X} \rightarrow \mathcal{Z}} K(f) \quad \text{s.t.} \quad R(f, \theta) \leq \gamma_2. \quad (17)$$

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

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

Note that for any representation function f , we have $p^{(u)}(Y, Z) = p^{(u)}(Y|X)p^{(u)}(X, Z)$ and $p^{(s)}(Y, Z) = p^{(s)}(Y|X)p^{(s)}(X, Z)$ where $p^{(u)}(Y|X)$ and $p^{(s)}(Y|X)$ denote the conditional distribution between label and input data in unseen and seen domain, respectively. Of course, $p^{(u)}(Y|X)$ and $p^{(s)}(Y|X)$ are independent with the mapping f . Thus,

$$p_1^{(u)}(Y, Z) = p^{(u)}(Y|X)p_1^{(u)}(X, Z), \quad (18)$$

$$p_2^{(u)}(Y, Z) = p^{(u)}(Y|X)p_2^{(u)}(X, Z), \quad (19)$$

and,

$$p_1^{(s)}(Y, Z) = p^{(s)}(Y|X)p_1^{(s)}(X, Z), \quad (20)$$

$$p_2^{(s)}(Y, Z) = p^{(s)}(Y|X)p_2^{(s)}(X, Z). \quad (21)$$

Next, to prove the convexity of $T(\gamma)$, we show that:

$$\lambda T(\gamma_1) + (1 - \lambda)T(\gamma_2) \geq T(\lambda\gamma_1 + (1 - \lambda)\gamma_2), \quad (22)$$

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

Now, let:

$$p_\lambda^{(u)}(X, Z) = \lambda p_1^{(u)}(X, Z) + (1 - \lambda)p_2^{(u)}(X, Z), \quad (23)$$

$$p_\lambda^{(s)}(X, Z) = \lambda p_1^{(s)}(X, Z) + (1 - \lambda)p_2^{(s)}(X, Z). \quad (24)$$

By definition, the left hand side of (22) can be rewritten by:

$$\begin{aligned} & \lambda T(\gamma_1) + (1 - \lambda)T(\gamma_2) \\ &= \lambda D(p_1^{(u)}(Y, Z) \| p_1^{(s)}(Y, Z)) \\ &+ (1 - \lambda)D(p_2^{(u)}(Y, Z) \| p_2^{(s)}(Y, Z)) \\ &= \lambda D(p^{(u)}(Y|X)p_1^{(u)}(X, Z) \| p^{(s)}(Y|X)p_1^{(s)}(X, Z)) \quad (25) \\ &+ (1 - \lambda)D(p^{(u)}(Y|X)p_2^{(u)}(X, Z) \| p^{(s)}(Y|X)p_2^{(s)}(X, Z)) \quad (26) \\ &\geq D(p^{(u)}(Y|X)p_\lambda^{(u)}(X, Z) \| p^{(s)}(Y|X)p_\lambda^{(s)}(X, Z)) \quad (27) \end{aligned}$$

where (25) and (26) due to (18), (19), (20), and (21); (27) due to (23), (24), 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 $p_\lambda^{(s)}(X, Z)$ ⁸, the reconstruction loss corresponding to f_λ is:

$$\gamma_\lambda = \int_{\mathbf{x} \in \mathcal{X}} \int_{\mathbf{z} \in \mathcal{Z}} p_\lambda^{(u)}(\mathbf{x}, \mathbf{z}) \ell(\mathbf{x}, \theta(\mathbf{z})) d\mathbf{x} d\mathbf{z}. \quad (28)$$

By Definition 5,

$$D(p^{(u)}(Y|X)p_\lambda^{(u)}(X, Z) \| p^{(s)}(Y|X)p_\lambda^{(s)}(X, Z)) \geq T(\gamma_\lambda). \quad (29)$$

Combine (27) and (29):

$$\lambda T(\gamma_1) + (1 - \lambda)T(\gamma_2) \geq T(\gamma_\lambda), \quad (30)$$

or the left hand side of (22) is larger or at least equal to $T(\gamma_\lambda)$. Next, we show that $T(\gamma_\lambda)$ is at least as large as the right hand side of (22). Particularly, we want to show:

$$T(\gamma_\lambda) \geq T(\lambda\gamma_1 + (1 - \lambda)\gamma_2). \quad (31)$$

Since $T(\gamma)$ is a monotonically non-increasing function, we want to show that:

$$\gamma_\lambda \leq \lambda\gamma_1 + (1 - \lambda)\gamma_2. \quad (32)$$

Indeed,

$$\gamma_\lambda = \int_{\mathbf{x}} \int_{\mathbf{z}} p_\lambda^{(u)}(\mathbf{x}, \mathbf{z}) \ell(\mathbf{x}, \theta(\mathbf{z})) d\mathbf{x} d\mathbf{z} \quad (33)$$

$$= \int_{\mathbf{x}} \int_{\mathbf{z}} (\lambda p_1^{(u)}(\mathbf{x}, \mathbf{z}) + (1 - \lambda)p_2^{(u)}(\mathbf{x}, \mathbf{z})) \ell(\mathbf{x}, \theta(\mathbf{z})) d\mathbf{x} d\mathbf{z} \quad (34)$$

$$= \lambda \int_{\mathbf{x}} \int_{\mathbf{z}} p_1^{(u)}(\mathbf{x}, \mathbf{z}) \ell(\mathbf{x}, \theta(\mathbf{z})) d\mathbf{x} d\mathbf{z} \quad (35)$$

$$+ (1 - \lambda) \int_{\mathbf{x}} \int_{\mathbf{z}} p_2^{(u)}(\mathbf{x}, \mathbf{z}) \ell(\mathbf{x}, \theta(\mathbf{z})) d\mathbf{x} d\mathbf{z} \quad (36)$$

$$\leq \lambda\gamma_1 + (1 - \lambda)\gamma_2 \quad (37)$$

with (33) due to (28); (34) due to (23); (35) and (36) due to a bit of algebra; (37) due to (16) and (17), respectively.

From (32) and (37), (31) follows. Finally, from (30) and (31), (22) follows. The proof is complete.

⁸Indeed, we always can construct f_λ that induces $p_\lambda^{(u)}(X, Z)$ and $p_\lambda^{(s)}(X, Z)$ by linear interpolating between f_1 and f_2 .