Neural Networks 167 (2023) 233–243

Contents lists available at ScienceDirect

Neural Networks

journal homepage: www.elsevier.com/locate/neunet

Learning domain invariant representations by joint Wasserstein distance minimization

Léo Andéol [a,c], Yusei Kawakami [b,d], Yuichiro Wada [d,e], Takafumi Kanamori [b,e,*], Klaus-Robert Müller [a,c,f,g,h,*], Grégoire Montavon [i,a,c,**]

[a] Machine Learning group, Technische Universität Berlin, 10587 Berlin, Germany
[b] Tokyo Institute of Technology, Tokyo, Japan
[c] Berlin Institute for the Foundations of Learning and Data – BIFOLD, 10587 Berlin, Germany
[d] Fujitsu Laboratories Ltd., Japan
[e] RIKEN AIP, Japan
[f] Max Planck Institute for Informatics, Stuhlsatzenhausweg 4, 66123 Saarbrücken, Germany
[g] Department of Artificial Intelligence, Korea University, Seoul 136-713, South Korea
[h] Google Deepmind, Berlin, Germany
[i] Department of Mathematics and Computer Science, Freie Universität Berlin, 14195 Berlin, Germany

ARTICLE INFO

Article history:
Received 6 May 2022
Received in revised form 15 May 2023
Accepted 17 July 2023
Available online 31 July 2023

Keywords:
Domain invariance
Subpopulation shift
Joint distribution matching
Wasserstein distance
Neural networks
Supervised learning

ABSTRACT

Domain shifts in the training data are common in practical applications of machine learning; they occur for instance when the data is coming from different sources. Ideally, a ML model should work well independently of these shifts, for example, by learning a domain-invariant representation. However, common ML losses do not give strong guarantees on how consistently the ML model performs for different domains, in particular, whether the model performs well on a domain at the expense of its performance on another domain. In this paper, we build new theoretical foundations for this problem, by contributing a set of mathematical relations between classical losses for supervised ML and the Wasserstein distance in joint space (i.e. representation and output space). We show that classification or regression losses, when combined with a GAN-type discriminator between domains, form an upper-bound to the true Wasserstein distance between domains. This implies a more invariant representation and also more stable prediction performance across domains. Theoretical results are corroborated empirically on several image datasets. Our proposed approach systematically produces the highest minimum classification accuracy across domains, and the most invariant representation.

1. Introduction

Learning from data that originates from different provenances representing the same physical observations occurs rather commonly, but it is nevertheless a highly challenging endeavor. These multiple data sources may e.g. originate from different users, acquisition devices, geographical locations, they may encompass batch effects in biology, or they may come from the same measurement devices that each are calibrated differently. Because the source of the data itself is typically not task-relevant, a learned model is therefore required to be *invariant across domains*. A valid strategy for achieving this is to learn an invariant intermediate representation (illustrated in Fig. 2). Furthermore, in certain applications, privacy requirements such as anonymity dictate that the source should not be recoverable from the representation. Hence, building a domain invariant representation can also be a desideratum *by itself*.

Domain invariance, in some contexts referred to as subpopulation shift (Koh et al., 2021) or distributional shifts (Amodei et al., 2016; Goel, Gu, Li, & Ré, 2021), can be contrasted to two related and well-researched areas that are *domain adaptation* (DA) (Shimodaira, 2000; Sugiyama, Krauledat, & Müller, 2007) and *domain generalization* (DG) (Dou, de Castro, Kamnitsas, & Glocker, 2019; Zhou, Jiang, Shui, Wang and Chaib-draa, 2021). Domain adaptation is mainly concerned with the model performance on the (unlabeled) target domain, often at the expense of incurring more errors on the (labeled) source domain. Domain generalization, on the other hand, aims to build a ML model that generalizes across *all* domains, including unseen ones. This generality imposes additional constraints on the solution, that can hamper the careful enforcement of invariance w.r.t. the domains

* Corresponding authors.
** Corresponding author at: Department of Mathematics and Computer Science, Freie Universität Berlin, 14195 Berlin, Germany.
E-mail addresses: leo@andeol.eu (L. Andéol), kawakami-yusei@fujitsu.com (Y. Kawakami), wada.yuichiro@fujitsu.com (Y. Wada), kanamori@c.titech.ac.jp (T. Kanamori), klaus-robert.mueller@tu-berlin.de (K.-R. Müller), gregoire.montavon@fu-berlin.de (G. Montavon).

https://doi.org/10.1016/j.neunet.2023.07.028

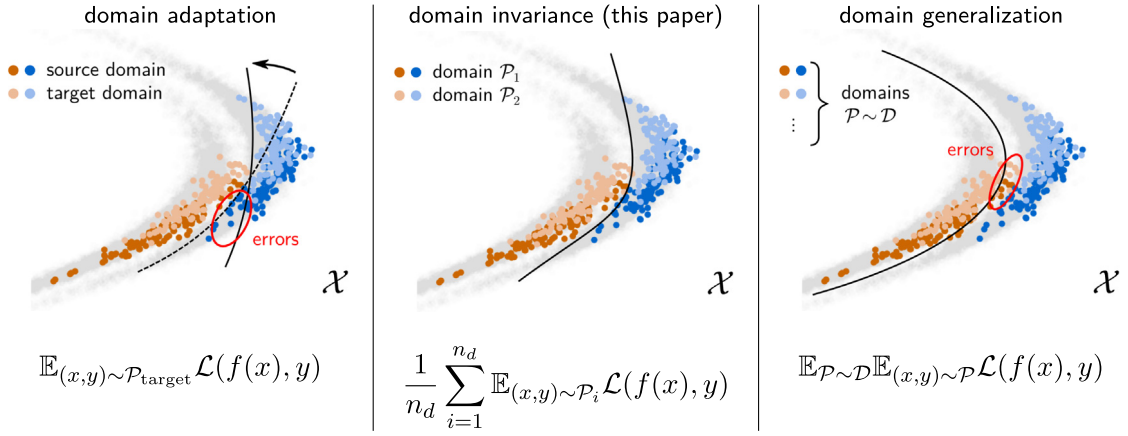

Fig. 1. Visual overview of the differences between domain adaptation, domain invariance and domain generalization in the context of classification. $\mathcal{X}$ denotes the input domain, and $\mathcal{P}$ denotes the various probability distributions. *Domain adaptation* learns a classifier that matches the target domain ($\mathcal{P}_{\text{target}}$) using information from the source domain, irrespective of its performance on it (errors as circled in red). *Domain invariance* treats each of the n_d domains equally and aims to build domain invariant representations and therefore a predictor that works equivalently well on each of them. *Domain generalization* addresses the more complex task of building a classifier that performs well on any domain drawn from some distribution $\mathcal{D}$ (including unseen ones, here depicted in gray). This is done potentially at the cost of giving up some accuracy on the few given domains (errors on the two domains of interest are circled in red).

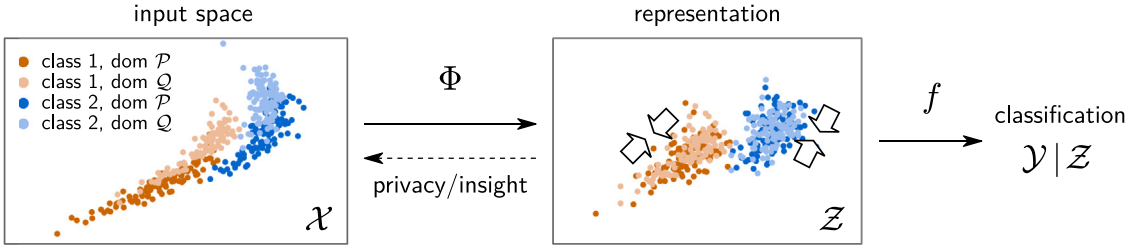

Fig. 2. Illustration of the problem of domain invariance in the case of classification. We would like to learn a function Φ that maps the data to a representation where the domains cannot be differentiated, and from which a domain-invariant classifier f can be built. The invariant representation induced by this model can serve further purposes such as domain privacy or extraction of domain-related insights. $\mathcal{X}$, $\mathcal{Z}$, $\mathcal{Y}$ correspond to the input, representation and target (label) space respectively.

at hand. In comparison, *domain invariance* (DI), our focus in this paper, considers that the ML model is trained and applied on a finite and given set of domains, and each domain is treated equally. The objective is to learn a model whose performance is well-balanced over the multiple given domains. The differences are highlighted graphically and with equations in Fig. 1. Hence, we address a singular and important problem, which has so far received little attention, especially in the context of deep learning models.

In order to address domain invariance, we consider in the present work the *Wasserstein distance* (Peyré, Cuturi, et al., 2019; Villani, 2008) as it characterizes the weak convergence of measures and displays several advantages over e.g. the more common Kullback–Leibler divergence, as discussed in Arjovsky, Chintala, and Bottou (2017) and Montavon, Müller, and Cuturi (2016). We contribute several bounds relating the Wasserstein distance between the joint distributions of two or more domains, and the objective function of practical supervised neural networks. This theoretical basis supports the rigorous learning of domain-invariant classifiers through the incorporation of a GAN-type discriminator between domains (or domain critic) as an auxiliary task.[1] With the proposed theoretical grounding, one can show that (1) the Wasserstein distance between the different domains is systematically reduced as an effect of training, and (2) the prediction performance gap between domains is also reduced as a result.

[1] Anecdotally, the use of a domain critic makes our method relate to works on domain adaptation such as DANN (Ganin et al., 2016) and WDGRL (Shen, Qu, Zhang, & Yu, 2018).

Furthermore, a significant part of the novelty of our work lies in contributing a formalism, which makes our theory applicable to *partially labeled* distributions. This allows us in particular to cover both supervised and semi-supervised learning scenarios. While a few other works also addressed the scenario where domains are partially labeled, they focus on the related but distinct problems of domain adaptation (Cheng & Pan, 2014; He, Liu, Fan, & You, 2020; López-Paz, Hernández-Lobato, & Schölkopf, 2012) and domain generalization (Sharifi-Noghabi, Asghari, Mehrasa, & Ester, 2020).

Our proposed approach is tested empirically on three domain invariance benchmarks: MNIST vs. SVHN, and the multi-domain Office-Caltech and PACS datasets. Results confirm our theoretical analysis, in particular, we find that our approach yields *highly invariant* representations, and that the latter support predictions that are accurate on *all* domains, including the most difficult ones. Lastly, we inspect the learned invariant representation using UMAP embeddings (McInnes, Healy, Saul, & Großberger, 2018) and 'explainable AI' (cf. Ribeiro, Singh, & Guestrin, 2016; Samek, Montavon, Lapuschkin, Anders, & Müller, 2021). This allows us to visually highlight how the data distributions associated to each domain merge into a single distribution under the effect of the training objective. It also allows us to explore which input features are used to map the data into the desired invariant representation (Liu, Long, Wang, & Jordan, 2019). Interestingly, we find that recognizing and exploiting *domain-specific* features remains in fact an integral part of the neural network strategy to arrive ultimately at the desired invariant representation.

2. Related work

Significant research in machine learning and statistics has been dedicated to the question of distributional shifts (between training and/or test distributions) (Amodei et al., 2016; Delage & Ye, 2010). This has resulted in a variety of machine learning formulations that can be broadly categorized into domain adaptation, domain generalization, and domain invariance.

2.1. Domain Adaptation

Domain Adaptation (Ben-David et al., 2010) has been studied due to the fact that in real-world situations, when the source and target distributions differ, for instance by a covariate shift (Shimodaira, 2000; Sugiyama et al., 2007), models trained on the source distribution perform significantly worse on the target. Domain adaptation has two major well-studied settings: Unsupervised Domain Adaptation and Semi-Supervised Domain Adaptation.

Unsupervised Domain Adaptation considers the situation where the source domain has labels but the target domain has not (cf. Zhang & Gao, 2022 for a review). Using the theoretical framework of Ben-David et al. (2010) where the target error is upper-bounded by the error on the source domain, the divergence between marginal distribution of the two domains and a constant term, Ganin et al. (2016) and Shen et al. (2018) propose a domain adaptation technique based on the minimization of this upper-bound. The joint distribution optimal transportation (JDOT) method of Courty, Flamary, Habrard, and Rakotomamonjy (2017) is similar to Shen et al. (2018), but it aims to minimize the Wasserstein distance between *joint* distributions. Note that Courty et al. (2017) does not use adversarial learning and instead solves the primal form of the optimal transport problem, and relies on a single-domain classifier to learn on the target domain with transported source-domain labels. The approach of Ganin et al. (2016) has been extended to a number of subsequent domain adaptation methods (Shu, Bui, Narui, & Ermon, 2018; Xu et al., 2020; Zhang, Ouyang, Li, & Xu, 2018). Other metrics than the Wasserstein distance can be used to align domains including the MMD, in works such as (Zhang, Li, & Teng, 2021) that use pseudo labels for unlabeled data, with a manifold regularization; or the Bures–Wasserstein distance (a specific case of the Wasserstein distance on normal distributions) such as in Liu, Ren, Xu, and Huang (2022). The method of Xiao and Zhang (2021), inspired by Saito, Watanabe, Ushiku, and Harada (2018), considers the alignment between domains and the class discriminability simultaneously, and proposes to weight these two terms in the objective in a dynamic manner. Although departing from existing theoretical frameworks, it achieves state-of-the-art empirical performance.

Semi-Supervised Domain Adaptation assumes a setting where there are a well-labeled source domain and a partially-labeled target domain. Observing that a few target labels can greatly improve task performance in applications such as object detection and image recognition, Semi-Supervised Domain Adaptation has recently attracted attention. The methods in Qin et al. (2021) and Saito, Kim, Sclaroff, Darrell, and Saenko (2019) correct the classifier's predictions that are biased to the large amount of labeled data in the source domain by using conditional entropy computed from its predictions. In Jiang et al. (2020), Kim and Kim (2020) and Li, Liu, Zhao, Zhang and Fu (2021), the input data is perturbed by a powerful data augmentation (e.g. Cubuk, Zoph, Shlens, & Le, 2020; DeVries & Taylor, 2017) or adversarial method (e.g. Miyato, Maeda, Koyama and Ishii, 2018), and then the model is trained so that the predictions for the original input and the perturbed input are consistent. Ref. Yang et al. (2021) proposes an efficient method for training a model by assigning pseudo one-hot labels to unlabeled target data predicted with high confidence during training. The methods presented above achieve good results in numerical experiments, but do not provide a rigorous theoretical discussion of the generalization error. In particular, Saito et al. (2019) minimizes an upper-bound of the target error, but that upper-bound contains the joint minimum error that cannot be optimized, and therefore it is not guaranteed that the target error will necessarily be small after training. One such exception is JDIP (Chen, Harandi, Jin, & Yang, 2020), which conducts a theoretical study with some similarities to ours, however in the case of domain adaptation, and with the classical L2 distance instead of a metric on distributions (the Wasserstein distance, with its advantages). This method builds on linear transformations and kernels models, whereas our approach works alongside more powerful and flexible neural networks.

Note that on both domain adaptation settings, the main goal is to improve generalization performance in the target domain, often at the expense of performance in the source domain. In our work, we would like to have high performance in *all* given domains, and this task is better addressed using domain invariance.

2.2. Domain Generalization

Domain Generalization (Blanchard, Lee, & Scott, 2011; Muandet, Balduzzi, & Schölkopf, 2013; Zhou, Liu, Qiao, Xiang, & Loy, 2022) is a very challenging problem that aims to achieve high performance on unseen target domains by learning models from multiple fully-labeled source domains. Domain generalization has received significant attention recently. Ref. Li, Pan, Wang and Kot (2018) combines an adversarial loss with a maximum mean discrepancy regularizer in order to extract a representation where domains are aligned. The method of Li, Tian et al. (2018) uses two adversarial losses to take advantage of label information in fully-labeled domains. The first loss matches the latent representation for each class, and the second loss reduces the negative effects of differences in class distributions across domains. The method of Dou et al. (2019) uses meta-learning in order to extract features that are consistent across domains. Ref. Zhou, Jiang et al. (2021) starts from an adversarial approach and incorporates a metric learning loss into the classifier in order to improve classification boundaries. Ref. Meng et al. (2022) introduces a new *attention diversification* framework, based on attention maps, where the latter are trained to produce diversified responses for task-related features and to remove domain specific features. The approach of Zhou, Yang, Qiao and Xiang (2021) mixes instance-level feature statistics across source domains. Mixing styles of training data has the effect of creating pseudo-new domains, resulting in increased diversity of training domains and improved generalization capability to unseen domains. The method of Li, Du et al. (2021) can address unsupervised domain adaptation and model adaptation (or source-free unsupervised domain adaptation) as well as domain generalization. The method generates adversarial attacks to the extent that semantic information of original data is retained, and then learns to reduce the classification loss for those adversarial examples.

A few works have focused on theoretical aspects of Domain Generalization. Ref. Li et al. (2020) develops theoretical arguments based on a strong assumption that the distribution of latent variables in all domains is represented by a linear combination of other domains. Ref. Albuquerque, Monteiro, Darvishi, Falk, and Mitliagkas (2019) shows an upper-bound theorem indicating that minimizing the divergence between the source marginal distributions like (Ganin et al., 2016; Shen et al., 2018) can minimize the unseen target error when the target distribution exists in the neighborhood of the convex hull of source distributions.

However, it is also known that minimizing the divergence to an extremely small value increases the divergence between the target distribution and the convex hull, which leads to an increase in the upper-bound. Sicilia, Zhao, and Hwang (2023) derives a tighter upper-bound of the target error than Albuquerque et al. (2019). Note that the negative effect resulting from minimizing the divergence remains unresolved.

While domain generalization is a challenging and interesting topic on its own, it differs from the setting we consider in the present paper by requiring the model to generalize well to *any* new domain. Not only this makes the theoretical analysis significantly harder than for the finite domain setting, performance gains on new unseen domains are often obtained at the expense of the existing domains, which are in our case the domains of interest.

2.3. Domain invariance

In contrast to domain adaptation and domain generalization, domain invariance is a comparatively less explored setting. Domain invariance shares some technical similarities with works in distributionally robust optimization (Duchi, Hashimoto, & Namkoong, 2023; Duchi & Namkoong, 2021; Rahimian & Mehrotra, 2019). These works however focus on the optimization problem and its theoretical properties rather than the problem of generalization between different groups or domains. Another related area is subpopulation shifts, which addresses the question of generalization across predefined subgroups (e.g. Goel et al., 2021; Koh et al., 2021; Sagawa, Koh, Hashimoto, & Liang, 2019). Unlike domain invariance, works on subpopulation shifts focus on building invariance to subgroups of the same domain, often numerous with few samples and small discrepancies, rather than producing invariance to qualitatively different domains. Furthermore, works on subpopulation shifts typically consider that the different subgroups are fully labeled, whereas the domain invariance framework we introduce in our work enables learning with domains that are partially labeled. Due to the limited previous works and the lack of reference methods for domain invariance, our experiments section will resort to ablation studies for comparison.

3. Domain invariance and optimal transport

Domain invariance can be described as the property of a representation to be indistinguishable with regards to its original domain, in particular, the multiple data distributions projected in representation space should look the same (i.e. have low distance). A recently popular way of measuring the distance between two distributions is the Wasserstein distance. The latter can be interpreted as the cost of transporting the probability mass of one distribution to the other if we follow the optimal transport plan, and it can be formally defined as follows:

Definition 1. Let $\mathcal{P} \in \mathcal{M}_+^1(\mathcal{A})$, $\mathcal{Q} \in \mathcal{M}_+^1(\mathcal{B})$ be two arbitrary probability distributions defined over two measurable metric spaces $\mathcal{A}$ and $\mathcal{B}$. Let c be a cost function. Their Wasserstein distance is:

$$W(\mathcal{P}, \mathcal{Q}) \stackrel{\text{def}}{=} \inf_{\pi \in \Pi(\mathcal{P}, \mathcal{Q})} \int_{\mathcal{A}\times\mathcal{B}} c(a, b) d\pi(a, b) \tag{1}$$

with $\Pi(\mathcal{P}, \mathcal{Q}) \stackrel{\text{def}}{=} \{\pi \in \mathcal{M}_+^1(\mathcal{A} \times \mathcal{B}) : P_{\mathcal{A}\#}\pi = \mathcal{P} \text{ and } P_{\mathcal{B}\#}\pi = \mathcal{Q}\}$, where $\mathbf{P}_{\mathcal{A}\#}$ and $\mathbf{P}_{\mathcal{B}\#}$ are push-forwards of the projection of $P_{\mathcal{A}}(a, b) = a$ and $P_{\mathcal{B}}(a, b) = b$. This can be loosely interpreted as $\Pi(\mathcal{P}, \mathcal{Q})$ being the set of joint distributions that have marginals $\mathcal{P}$ and $\mathcal{Q}$.

Hence, we measure the invariance of representations by how low the Wasserstein distance is between the distributions $\mathcal{P}$ and $\mathcal{Q}$ associated to the two domains. The $\mathcal{P}$ and $\mathcal{Q}$ distributions respectively are the $\mathcal{P}_1$ and $\mathcal{P}_2$ distribution of Fig. 1. The Wasserstein distance being scale-dependent, we assume that representations of both domains have fixed scale. In comparison to other common alternatives such as the Kullback–Leibler divergence, the Jensen–Shannon divergence, or the Total Variation distance, the Wasserstein distance has the advantage of taking into account the metric of the representation space (via the cost function $c(a, b)$), instead of looking at pure distributional overlap, and this typically leads to better ML models (Arjovsky et al., 2017; Montavon et al., 2016). Computing the Wasserstein distance with Eq. (1) is expensive. Luckily, if we use the metric of our space as a cost function, such as the Euclidean distance $c(a, b) = \|a - b\|_2$, we can derive a dual formulation of the 1-Wasserstein distance as follows:

$$W(\mathcal{P}, \mathcal{Q}) = \sup_{\|\varphi\|_{Lip} \leq 1} \mathbb{E}_{\mathcal{P}}[\varphi] - \mathbb{E}_{\mathcal{Q}}[\varphi] \tag{2}$$

where $\mathbb{E}_{\mathcal{P}}[\varphi]$ is the expected value of function φ on the distribution $\mathcal{P}$. This formulation replaces an explicit computation of a transport plan, by a function to estimate, a task particularly appropriate for neural networks. Recently several methods have used this approach to learn distributions (Montavon et al., 2016) specifically in the context of Generative Adversarial Networks (Arjovsky et al., 2017; Shen et al., 2018). The main constraint lies in the necessity of the function φ, which we will call the discriminator (or critic), to be 1-Lipschitz. A few approaches were proposed to tackle this problem, such as gradient clipping (Arjovsky et al., 2017), gradient penalty (Gulrajani, Ahmed, Arjovsky, Dumoulin, & Courville, 2017) and more recently the Spectral Normalization (Miyato, Kataoka, Koyama and Yoshida, 2018). It is however important to note that in practice the set of possible discriminators will be a subset of 1-Lipschitz continuous functions.

4. Relating Wasserstein distance to supervised losses

We would like to align the predicting behavior of a ML model on multiple domains following the approach illustrated in Fig. 2, i.e. by learning a *domain-invariant representation*. Specifically, we aim for a representation of data where the distributions associated to the two domains have minimum Wasserstein distance and therefore cannot be distinguished. At the same time, the representation should contain the features that are necessary to solve the given prediction task, e.g. using common supervised loss functions. We focus here on the two-domain case, and refer to Supplementary Note D for the case of three or more domains.

Let us start with some formalities: We denote by $\mathcal{X}$ the input space, by $\mathcal{Z} \subset \mathbb{R}^d$ our representation or feature space, and by $\mathcal{Y}$ the label or target space. We further denote by $\Phi : \mathcal{X} \to \mathcal{Z}$ the feature extractor, and by $f : \mathcal{Z} \to \mathcal{Y}$ the prediction function (e.g. regression; classification). We assume $\mathcal{Z}$ and $\mathcal{Y}$ to be compact measurable spaces, and we denote by $\mathcal{M}_+^1(\mathcal{Z}\times\mathcal{Y})$ the set of probability distributions defined on their product space. Let $\mathcal{P}^t, \mathcal{Q}^t \in \mathcal{M}_+^1(\mathcal{Z} \times \mathcal{Y})$ be the *true* probability distributions formed by the two domains we would like to align. When necessary, we add a subscript to these distributions to specify their support.

Similarly to previous works, domain alignment will be measured as the Wasserstein distance $W(\mathcal{P}^t, \mathcal{Q}^t)$ of samples embedded in *feature space*, but also including their labels. We contribute by showing that the Wasserstein distance $W(\mathcal{P}^t, \mathcal{Q}^t)$, can be related formally to common loss functions used in classification or regression, via mathematical inequalities. With these inequalities one can design practical learning objectives fairly easily, whose minimization not only solves the task at hand, but also implies as a side effect the minimization of the Wasserstein distance

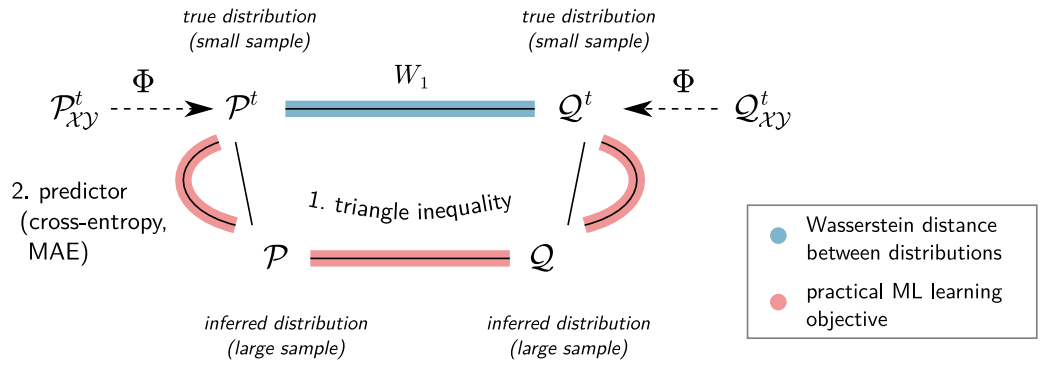

Fig. 3. Visual overview of our theoretical framework. It relates the Wasserstein distance between the joint distributions $\mathcal{P}^t$, $\mathcal{Q}^t$ of each domain (blue) to components of a practical ML objective (red) in two steps. Step 1: The true joint distributions (of which only a small sample is observable) can be related via the triangle inequality to inferred distributions $\mathcal{P}$, $\mathcal{Q}$ where missing labels are *predicted* from the features. Step 2: The expanded terms can be further upper-bounded by common terms of a ML objective (cross-entropy, mean absolute error (MAE), etc.).

between distributions of the two domains, thereby achieving domain invariance.

From a technical standpoint, our novel approach draws some inspiration from Courty et al. (2017) and is based on measure theory, in order to formalize partially labeled distributions and therefore our problem of aligning multiple joint distributions. The individual steps are presented in Sections 4.1–4.2, and we provide an overview of our novel theoretical framework in Fig. 3.

4.1. Incorporating semi-supervised data

Computation of the true Wasserstein distance $W(\mathcal{P}^t, \mathcal{Q}^t)$ would require knowledge of the true distributions $\mathcal{P}^t$ and $\mathcal{Q}^t$. In practice, we only have a finite sample of these distributions, and the quality with which the Wasserstein distance can be approximated largely depends on the amount of labeled data available. (For high-dimensional tasks, the necessary amount of labels would be overwhelming.) However, in practice, it is common that *unlabeled data* is available in much larger quantity than the labeled data. We consider this semi-labeled scenario where only a fraction of data are obtained from $\mathcal{P}^t$, $\mathcal{Q}^t$ (i.e. labeled). The remaining data are unlabeled and obtained from the marginals $\mathcal{P}^t_{\mathcal{Z}}$, $\mathcal{Q}^t_{\mathcal{Z}}$.

By learning an appropriate function $f : \mathcal{Z} \to \mathcal{Y}$, say, a neural network classifier, that infers labels from features, one can obtain an approximation $\mathcal{P}^f = (z, f(z))_{z\sim\mathcal{P}^t_{\mathcal{Z}}}$ of the true joint distribution $\mathcal{P}^t$. This implies that the distribution we effectively have access to (and draw from) in practice is the mixture:

$$\mathcal{P} = \alpha\mathcal{P}^t + (1-\alpha)\mathcal{P}^f.$$

where $\alpha \in (0, 1)$ is the fraction of labeled data. We will refer to $\mathcal{P}$ as the *inferred* distribution. Identically for the second domain, we construct an appropriate function $g : \mathcal{Z} \to \mathcal{Y}$ from which one can predict labels, and which results in another inferred distribution:

$$\mathcal{Q} = \beta\mathcal{Q}^t + (1-\beta)\mathcal{Q}^g.$$

Note that the functions f and g need not be identical. Also, β may differ from α, which addresses the case where different domains have different proportions of labeled data. Let the Wasserstein distance's cost function c be the metric on the space $\mathcal{Z} \times \mathcal{Y}$. By application of the triangle inequality, the following relation can be extracted:

$$W(\mathcal{P}^t, \mathcal{Q}^t) \le W(\mathcal{P}^t, \mathcal{P}) + W(\mathcal{P}, \mathcal{Q}) + W(\mathcal{Q}, \mathcal{Q}^t). \tag{3}$$

In other words, the distance between inferred distributions $W(\mathcal{P}, \mathcal{Q})$ to which we add the inference 'errors', i.e. the distance between true and inferred distributions on each domain, form an upper-bound to the true distance between distributions. Let us now analyze the error term $W(\mathcal{P}^t, \mathcal{P})$. We consider the case of $\mathcal{P}$ (analogously so for $\mathcal{Q}$).

Lemma 1. *Let $\mathcal{K} \in \mathcal{M}^1_+(\mathcal{Z} \times \mathcal{Y})$ be an arbitrary probability distribution, we then have $W(\mathcal{P}, \mathcal{K}) \le \alpha W(\mathcal{P}^t, \mathcal{K}) + (1-\alpha)W(\mathcal{P}^f, \mathcal{K})$, and for the special case where $\mathcal{K} = \mathcal{P}^t$, we get*

$$W(\mathcal{P}, \mathcal{P}^t) = (1-\alpha)W(\mathcal{P}^f, \mathcal{P}^t). \tag{4}$$

A proof can be found in Supplementary Note A. The proof proceeds by first decomposing $\mathcal{P}$ into its mixture components, and then applying Jensen's inequality on the Wasserstein dual's supremum. For the special case of Eq. (4), the equality is due to $\mathcal{K}$ being an element of the mixture $\mathcal{P}$ and the Wasserstein distance of $\mathcal{K}$ to that mixture element being consequently zero.

Finally, combining the results above, that is, by applying the triangle inequality, Lemma 1, and using the symmetry property of the Wasserstein distance, one can obtain another bound on the true Wasserstein distance, where unlike Eq. (3), some mixture components now appear explicitly:

Theorem 1. *Given the Wasserstein distance's cost function c is the metric on the product space $\mathcal{Z} \times \mathcal{Y}$, we get*

$$W(\mathcal{P}^t, \mathcal{Q}^t) \le (1-\alpha)W(\mathcal{P}^t, \mathcal{P}^f) + W(\mathcal{P}, \mathcal{Q}) + (1-\beta)W(\mathcal{Q}^t, \mathcal{Q}^f). \tag{5}$$

This final formulation will let us relate in Section 4.2 some of the expanded terms, specifically, the distances $W(\mathcal{P}^t, \mathcal{P}^f)$ and $W(\mathcal{Q}^t, \mathcal{Q}^f)$ to common loss functions used in supervised machine learning.

4.2. Connection to supervised ML losses

Various loss functions have been proposed for supervised learning. They address the diversity of output types (e.g. class labels; regression targets) and statistical properties of the data (e.g. margin between classes; presence/absence of outliers). Ideally, one would be able to achieve domain invariance while retaining the ability to optimize the most suitable loss function for a given problem.

Let us start with Eq. (5) in Theorem 1, in particular, the distance $W(\mathcal{P}^t, \mathcal{P}^f)$. The latter essentially measures the level of error with which the function f predicts the true labels y. It therefore plays a similar role to common loss functions used for supervised machine learning. Both can also be mathematically related. First, the Mean Absolute Error (MAE) commonly used in robust regression can be related to $W(\mathcal{P}^t, \mathcal{P}^f)$ as follows:

Lemma 2.

$$W(\mathcal{P}^t, \mathcal{P}^f) \leq \mathbb{E}_{(z,y)\sim\mathcal{P}^t}\big[|y - f(z)|\big] \tag{6}$$

The proof is given in Supplementary Note A. In the case of classification, a similar result can be provided for the Kullback–Leibler (KL) divergence, which is equivalent to the cross-entropy loss when $\mathcal{P}_{\mathcal{Y}|z}$ is deterministic:

Lemma 3. *Assuming that $\mathcal{P}^f$ and $\mathcal{P}^t$ admit densities, we then obtain*

$$W(\mathcal{P}^t, \mathcal{P}^f) \leq \operatorname{diam}(\mathcal{Z}\times\mathcal{Y})\sqrt{\frac{1}{2}\mathbb{E}_{z\sim\mathcal{P}_\mathcal{Z}}\Big[KL(\mathcal{P}^t_{\mathcal{Y}|z} \parallel \mathcal{P}^f_{\mathcal{Y}|z})\Big]}, \tag{7}$$

where $KL(\mathcal{P} \parallel \mathcal{Q}) = -\int_{\mathcal{Z}\times\mathcal{Y}} \log \frac{d\mathcal{Q}}{d\mathcal{P}} d\mathcal{P}$ is the Kullback–Leibler divergence; and where $\operatorname{diam}(\cdot)$ *is the diameter of the space received as input, i.e. the largest distance obtainable in that space.*

For a proof, see Supplementary Note A. Lemmas 2 and 3 now relate the Wasserstein distance formulation to loss functions occurring in regression and classification tasks that are easily computable, and with the desired statistical properties. Together with the relation shown in Section 4.1, we can now propose a ML formulation that both addresses the prediction task, and enforces domain invariance.

5. Learning a domain invariant neural network

Consider the data available consists of examples sampled from both domains, specifically, from distributions $\mathcal{P}$ and $\mathcal{Q}$. Under these distributions, part of the data comes with the true labels. For the rest of the data, labels are inferred via the functions f and g respectively. We denote by (X^p, Y^p) the dataset of n examples drawn from the first domain $\mathcal{P}$ and by (X^q, Y^q) the dataset of m examples drawn from the second domain $\mathcal{Q}$. Based on Theorem 1 and Lemmas 2–3, one can define a learning procedure that consists of simultaneously minimizing a supervised loss function $\mathcal{L}$ on each domain and the Wasserstein distance $W(\mathcal{P}, \mathcal{Q})$ aligning distributions of the two domains. In the classification setting, the supervised loss for the first domain is defined as

$$\mathcal{L}(Z^p, Y^p) = \frac{1}{n}\sum_{i=1}^{n} KL(f(Z_i^p) \parallel Y_i^p), \tag{8}$$

where Y_i^p and $f(Z_i^p)$ are vectors containing probabilities of each class. A similar loss function can be built for the second domain. Note that the loss is only effective on the examples that come with a true label, because when the label is inferred, we have $KL(f(Z_i^p) \parallel Y_i^p) = KL(f(Z_i^p) \parallel f(Z_i^p)) = 0$. In a similar fashion, for the regression setting, we define $\mathcal{L}(Z^p, Y^p) = \frac{1}{n}\sum_{i=1}^{n}|f(Z_i^p) - Y_i^p|$. For the minimization of the Wasserstein distance $W(\mathcal{P}, \mathcal{Q})$ we consider the dual form provided in Eq. (2), specifically, an empirical estimate of it:

$$W(\mathcal{P}, \mathcal{Q}) \approx \max_{\varphi:\, \|\varphi\|_{Lip}\leq 1} \underbrace{\Big\{\frac{1}{n}\sum_{i=1}^{n}\varphi(Z_i^p, Y_i^p) - \frac{1}{m}\sum_{i=1}^{m}\varphi(Z_i^q, Y_i^q)\Big\}}_{\Delta(Z^p, Y^p, Z^q, Y^q)} \tag{9}$$

and this forms our domain critic. Finally, one can sum the supervised terms and the domain critic, i.e. Eqs. (8) and (9), and optimize the resulting objective w.r.t. the functions f, Φ, φ (more precisely, the parameters of the neural networks implementing these functions). This can be formulated as the GAN-like optimization problem:

$$\min_{\Phi, f}\max_{\varphi} \Big\{ \Delta(\Phi(X^p), Y^p, \Phi(X^q), Y^q) + \lambda_p \mathcal{L}(f(\Phi(X^p)), Y^p) + \lambda_q \mathcal{L}(f(\Phi(X^q)), Y^q) \Big\} \quad \text{s.t. } \|\varphi\|_{Lip} \leq 1. \tag{10}$$

where the classifier terms and the domain critic are in competition. The hyperparameters λ_p and λ_q can either be fixed to $1-\alpha$ and $1-\beta$ respectively (and for classification multiplied by the domain's diameter) in order to match the theory; or they can be selected heuristically or based on some validation procedure. The Lipschitzness constraint on φ is practically enforced by using one of the regularization techniques mentioned at the end of Section 3. Additionally, a constraint on the scale of the representation or the Lipschitzness of the classifier f can be added in order to prevent an arbitrary downscaling of the representation which may cause the Wasserstein distance to artificially go to zero. Lastly, supplementary regularization terms, such as EntMin Grandvalet, Bengio, et al. (2005), Virtual Adversarial Training (Miyato, Maeda et al., 2018), and Virtual Mixup (Mao, Ma, Yang, Chen, & Li, 2019) can be added to the objective, in order to take further advantage of the unlabeled examples. A visual representation of our model is given in Fig. 4. The steps needed to train a domain invariant networks are summarized in Algorithm 1.

Data: Semi-supervised datasets for both domains: $(X^p, Y^p), (X^q, Y^q)$
Input: Untrained Φ, f with parameters θ, hyperparameters λ_p, λ_q
Result: Trained Φ, f
for epochs **do**
  for batch $(x^p, y^p), (x^q, y^q) \in (X^p, Y^p), (X^q, Y^q)$ **do**
    `/* Compute features */`
    $z^p \leftarrow \Phi(x^p)$
    $z^q \leftarrow \Phi(x^q)$
    `/* Impute instances with missing labels in the batch */`
    $y_i^p \leftarrow f(z_i^p)\ \ \forall$ unlabeled i
    $y_j^q \leftarrow f(z_j^q)\ \ \forall$ unlabeled j
    `/* Reverse gradient of features for domain discriminator */`
    $z^p_{\text{rev}}, y^p_{\text{rev}} \leftarrow \text{Rev\_Grad}(z^p), \text{Rev\_Grad}(y^p)$
    $z^q_{\text{rev}}, y^q_{\text{rev}} \leftarrow \text{Rev\_Grad}(z^q), \text{Rev\_Grad}(y^q)$
    `/* Compute losses */`
    $L_{\text{disc}} \leftarrow \Delta(z^p_{\text{rev}}, y^p_{\text{rev}}, z^q_{\text{rev}}, y^q_{\text{rev}})$
    $L_{\text{classif}} \leftarrow \lambda_p \mathcal{L}(f(z^p)), y^p) + \lambda_q \mathcal{L}(f(z^q)), y^q)$
    $L_{\text{total}} \leftarrow L_{\text{disc}} + L_{\text{classif}}$
    `/* Perform a gradient descent step */`
    $\theta \leftarrow \theta - \gamma \nabla L_{\text{total}}$
  end
end

Algorithm 1: Algorithm for training our proposed domain invariant network. The function 'Rev_Grad' denotes a gradient reversal layer, which leaves the forward pass unchanged but multiplies the gradient by -1 in the backward pass (see e.g. Ganin et al. (2016)).

We now outline some specific condition on the data distribution which provides statistical consistency of the estimator.

Remark 1. Let us consider the condition on which Eq. (10) is statistically consistent when sufficiently many labeled examples

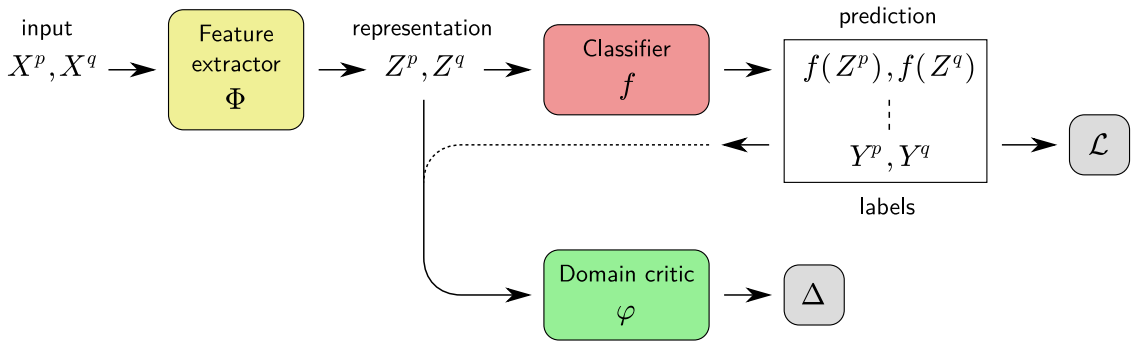

Fig. 4. Diagram of the proposed machine learning model, that induces a domain-invariant representation through a domain critic.

are observed. For the data distributions $\mathcal{P}^t_{\mathcal{X}\mathcal{Y}} = \mathcal{P}^t_{\mathcal{Y}|\mathcal{X}}\mathcal{P}^t_{\mathcal{X}}$ and $\mathcal{Q}^t_{\mathcal{X}\mathcal{Y}} = \mathcal{Q}^t_{\mathcal{Y}|\mathcal{X}}\mathcal{Q}^t_{\mathcal{X}}$, suppose that there exists a function $z = \Phi^o(x)$ such that

(i) the conditional probabilities satisfy $\mathcal{P}^t_{\mathcal{Y}|\mathcal{X}}(y|(\Phi^o)^{-1}(A)) = \mathcal{Q}^t_{\mathcal{Y}|\mathcal{X}}(y|(\Phi^o)^{-1}(A))$ for any measurable subset $A \subset \mathcal{Z}$, and
(ii) $\Phi^o_* P_{\mathcal{X}} = \Phi^o_* Q_{\mathcal{X}}$,

where Φ^o_* is the push-forward with the function $x \mapsto \Phi^o(x)$. Under the above assumptions, we see that $\Phi^o_* \mathcal{P}^t_{\mathcal{X}\mathcal{Y}} = \Phi^o_* \mathcal{Q}^t_{\mathcal{X}\mathcal{Y}}$ holds. Hence, the Wasserstein distance estimator Δ under the population distribution becomes zero. Due to the assumption on the conditional distribution, the optimal classifier f on $\mathcal{Z}$ is common for both distributions. Therefore, the optimal classifier with domain-invariant features is obtained by Eq. (10).

To put it more simply, by assuming there exists a feature map where marginal distributions are aligned, and that the conditional distributions are equal almost everywhere for the image of $\Phi^o(x)$, optimizing our objective function leads to the optimal classifier.

5.1. Generalization bounds

Interestingly, the Wasserstein distance between the true distributions of the two domains (that we have upper-bounded in Theorem 1) can also be related to the risks of the classifier on the two domains. Let $\mathcal{R}_{\mathcal{P}^t}(f) = \mathbb{E}_{z,y\sim\mathcal{P}^t}[\mathcal{L}(f(z), y)]$ be the risk or error of a classifier f. We here develop a result using the joint Wasserstein distance, similar to previous result obtained by Redko, Habrard, and Sebban (2017) on the distance between marginals.

Theorem 2. *Let $\mathcal{Z}, \mathcal{Y}$ be two compact measurable metric spaces whose product space has dimension d. Let $\mathcal{P}^t, \mathcal{Q}^t \in \mathcal{M}^1_+(\mathcal{Z}\times\mathcal{Y})$ two joint distributions associated to the two domains, and $\widehat{\mathcal{P}}^t, \widehat{\mathcal{Q}}^t$ their empirical counterparts. Let the transport cost function c associated to the optimal transport problem be $c(z_1, y_1; z_2, y_2) = \| \binom{z_1}{y_1} - \binom{z_2}{y_2} \|_2$, the Euclidean distance as the metric on $\mathcal{Z}\times\mathcal{Y}$ and $\mathcal{L} : \mathcal{Y}\times\mathcal{Y} \to \mathbb{R}^+$ a symmetric κ-Lipschitz loss function. Then for any $d' > d$ and $\psi' < \sqrt{2}$ there exists some constant N_0 depending on d' such that for any $\delta > 0$ and $\min(N_P, N_Q) \geq N_0 \max(\delta^{-(d'+2)}, 1)$ with probability at least $1-\delta$ for all λ-Lipschitz f the following holds:*

$$|\mathcal{R}_{\mathcal{Q}^t}(f) - \mathcal{R}_{\mathcal{P}^t}(f)| \leq \kappa\sqrt{(\lambda^2+1)} \times \left[W_1(\widehat{\mathcal{P}}^t, \widehat{\mathcal{Q}}^t) + \sqrt{\frac{2}{\psi'}\log\left(\frac{1}{\delta}\right)}\left(\sqrt{\frac{1}{N_P}} + \sqrt{\frac{1}{N_Q}}\right)\right]. \quad (11)$$

(A proof is given in Supplementary Note A.) In other words, the empirical Wasserstein distance between the two domains upper-bounds the prediction performance gap between the two domains. In practice, we can therefore expect the optimization of the objective in Eq. (10) to not only reduce the Wasserstein distance between domains (as we have shown in the previous sections), but also to produce a more uniform classification accuracy across domains and therefore a higher minimum accuracy.

We may also want to compare the joint discriminator $W(\mathcal{P}, \mathcal{Q})$ to the more common marginal discriminator $W(\mathcal{P}_{\mathcal{Z}}, \mathcal{Q}_{\mathcal{Z}})$. Indeed, it seems that in many cases, such as when the conditional distribution is identical between the two domains, the solution obtained appears to be equivalent. This is true due to theoretical reasons we will explore here. Let us first recall a bound on the distance between errors with a marginal Wasserstein distance.

Theorem 3 (*From Shen et al., 2018 (Adapted)*). *Let $\mathcal{P}^t_{\mathcal{Z}}, \mathcal{Q}^t_{\mathcal{Z}} \in \mathcal{M}^1_+(\mathcal{Z})$ be two probability measures. Assume the functions $f \in H$ are all λ-Lipschitz continuous for some λ. Then for every $f \in H$ the following holds*

$$|\mathcal{R}_{\mathcal{P}^t}(f) - \mathcal{R}_{\mathcal{Q}^t}(f)| \leq 2\lambda \cdot W_1\left(\mathcal{P}^t_{\mathcal{Z}}, \mathcal{Q}^t_{\mathcal{Z}}\right) + \mathcal{E}$$

where $\mathcal{E}$ is the combined error of the ideal f^* that minimizes the combined error $\mathcal{R}_{\mathcal{P}^t}(f) + \mathcal{R}_{\mathcal{Q}^t}(f)$.

The main difference compared with our bound is the presence of $\mathcal{E}$, the combined error of the hypothesis on both domains. Indeed, when the features of the two domains are properly aligned, the bound obtained with a joint or marginal Wasserstein distances are similar. However, when the domains are not properly aligned, usually due to the transformation between the two domains being large, and to the lack of labeled samples, we can have a large $\mathcal{E}$ such as $\mathcal{E} = 1$, which renders the bound very large. Such a case arises when both domains have entirely identical samples but with opposite labels, for instance. The bound with the joint Wasserstein distance can lead to features more aligned even with large transformations between domains. We expect to observe similar performances between marginal and joint discriminators on experiments with simple transformations, and larger discrepancies as the transformations get larger.

6. Experiments

To test whether our proposed approach truly achieves an invariant representation and reduces the performance gap between domains (as predicted by Theorems 1 and 2 respectively), we conduct experiments on three common image classification problems. First, a handwritten digits recognition task where the digit images come from two popular datasets: MNIST (LeCun, 1998) and SVHN (Netzer et al., 2011), each of them constituting one domain. Then, we consider the Office-Caltech classification dataset (Gong, Shi, Sha, & Grauman, 2012), which consists of four domains. Finally, we consider the recent and more complex PACS multi-domain image recognition dataset (Li, Yang, Song

and Hospedales, 2017), which also consists of four domains. We describe below these multi-domain tasks, and the training procedure for our models. More details are provided in Supplementary Note B.

6.1. Data and models

MNIST and SVHN are two common digit recognition datasets composed of 60 000 and 73 257 training examples respectively. While MNIST digits are black&white, SVHN digits are colored and have more complex appearances, making them harder to predict. In our MNIST-SVHN two-domain scenario, we simulate partly labeled data by only providing labels for a random subset of examples (1000 for each domain for the experiments of Table 1, and 3000 per domain for experiments of Table 2). The remaining examples are given unlabeled. MNIST images are brought to the SVHN format by scaling and setting each RGB component to the MNIST grayscale value. For experiments in Tables 1 and 2, the function Φ is implemented by the Conv-Large model from Miyato, Maeda et al. (2018). The model takes as input images of size $32 \times 32 \times 3$. We use small random translations of 2 pixels as well as color jittering as data augmentation.

Importantly, for the purpose of evaluating the domain invariance of representations, we would like to stabilize the scale of representations learned by the different models. Specifically, we add for the experiments of Table 1 a further penalty to the objective: the Wasserstein distance between the distribution of distances in representation space (the histogram of distances of the union distribution of $\mathcal{P}_\mathcal{Z}$ and $\mathcal{Q}_\mathcal{Z}$) and a predefined Gaussian mixture, which we set to be a univariate mixture of Gaussians $\frac{1}{10}\mathcal{N}(5, 2)+\frac{9}{10}\mathcal{N}(15, 3)$. The two modes model distances between data points of same class and of different classes respectively. Since the distribution of distances is a 1-dimension histogram, the Wasserstein distance can be computed analytically (Peyré et al., 2019). This added constraint ensures a similar scale for the representation extracted by our model and the different baselines. In particular, it ensures that a reduction of Wasserstein distance in representation space can be reliably interpreted as an increase of domain invariance, and not as a simple scaling of the representation. We have experimented with several new metrics and constraints for settling for this one, although it comes with some side effects. Indeed, by its very definition, it implies that there should be 10 equidistant and equally sized clusters, which is an assumption that is not verified for all datasets (for instance, SVHN). Moreover, it gives a stronger advantage (in the form of a prior) to the bare methods, without any discriminator, acting indirectly as one.

Our second scenario is based on the Office-Caltech dataset. It is composed of four domains (Amazon, Caltech, DSLR, Webcam) containing pictures of objects present in offices (such as monitors) from different sources, such as pictures from a real office, or ones with white backgrounds from an e-commerce website. There are 10 classes, and a varying number of samples depending on the domain (between 150 and 1100). We use the Decaf6 (Donahue et al., 2014) features with 4096 dimensions. We use the Resnet-18 architecture (He, Zhang, Ren, & Sun, 2016). We train a model on each possible bi-domain task (6 tasks) and average the resulting accuracies per domain.

Our third and last scenario is based on the PACS dataset, which consists of 10000 examples, with 4 domains (Photo, Art, Cartoon, Sketch) and 7 classes. We simulate semi-labeled data by providing labels for only 500 randomly sampled images from each domain, and giving remaining images unlabeled. The classes and domains are imbalanced, i.e. contain a different number of examples. The images are resized to $224 \times 224 \times 3$, and a pipeline of data augmentation is applied based on RandAugment (Cubuk et al., 2020). We again use the Resnet-18 architecture. On this dataset, we test domain invariance in a 'one vs. rest' setting.

Table 1
Effect of the domain critic on the classification accuracy and the Wasserstein distance between the two domains in representation space. We use 1000 labels per domain. Best performance is shown in bold. For indicative purpose, we report in the first two rows the classification accuracy on individual domains.

Model	Accuracy				W dist.
	MNIST	*SVHN*	Avg	Min	
No critic	*98.9*	*90.2*	94.55	90.2	3.92
Marginal critic (Shen et al., 2018)	*97.5*	*91.5*	94.5	**91.5**	3.43
Joint critic (Ours)	*97.5*	*91.5*	**94.6**	**91.5**	**3.36**

In all our experiments, the classifier f is a simple 2-layer MLP, and the discriminator φ a 3-layer MLP with spectral normalized weights (Miyato, Kataoka et al., 2018). (On the multi-domain PACS, we use a discriminator for each domain, computed in a one-vs-rest manner.) The weights (hyperparameters) for each loss term λ_p and λ_q are set to one, as well as the discriminator's. Unless mentioned otherwise, the networks are trained for 20 to 50 epochs using the Adam (Kingma & Ba, 2015) optimizer.

6.2. Results and analysis

As a first experiment, we study the effect of the domain critic we have proposed in Section 5 on the accuracy of the model, and on the Wasserstein distance between the two domains. We consider two baselines for comparison: (1) a simple supervised neural network without domain critic, (2) a supervised network where the critic φ is based only on *marginal* distributions (such as proposed in Shen et al. (2018)). These two baselines can be interpreted as an ablation study of our method, where instead of applying the Wasserstein distance to the joint input-label distribution, we apply it first only to the input variables (marginal critic), and then to no variables at all (no critic). For this experiment we do not use any additional losses/regularizers, and simply optimize the classification and domain alignment terms. We report the Wasserstein distance between the two domains' joint distributions, and the minimum classification accuracy for the two domains. These are two properties that our domain-invariant network is expected to fulfill (Theorems 1 and 2 respectively). Results are shown in Table 1.

Results corroborate our theory. In particular, we observe that the Wasserstein distance significantly decreases under the effect of adding a domain critic, specifically a joint domain critic that puts more focus on $\mathcal{Y}$, and the minimal accuracy over the two domains increases. Furthermore, we observe in this experiment that the use of a joint critic also leads to the highest average accuracy across domains.

Independently of the question of domain invariance, unsupervised data has already been routinely leveraged by classical semi-supervised learning approaches. These approaches have shown powerful on data with manifold structure (e.g. Li, Xu, Zhu, Zhang, 2017; Rasmus, Berglund, Honkala, Valpola, & Raiko, 2015). In our next experiment, we test the benefit of domain alignment techniques on models that are already equipped with semi-supervised learning mechanisms. Specifically, we consider a combination of two common semi-supervised techniques: conditional entropy minimization (EntMin) (Grandvalet et al., 2005) and virtual adversarial training (VAT) (Miyato, Maeda et al., 2018), which have shown strong empirical performance on numerous tasks. Results are given in Table 2.

We observe that semi-supervised learning on both domains, achieved by a combination of VAT and EntMin, leads to a strong baseline. In particular, it achieves the highest performance on

Table 2
Evaluation of our method in combination with classical semi-supervised learning regularizers (VAT + EntMin). We use 3000 labels per domain. Best results are in bold.

Model	Accuracy			
	MNIST	*SVHN*	Avg	Min
No critic + VAT/EntMin (only MNIST)	*99.14*	·	·	·
No critic + VAT/EntMin (only SVHN)	·	*94.79*	·	·
No critic	*98.76*	*87.33*	93.05	87.33
No critic + VAT/EntMin	*99.29*	*91.86*	95.58	91.86
Joint critic (Ours) + VAT/EntMin	*99.26*	*92.75*	96.01	92.75
Joint critic (Ours) + VAT/EntMin + Fine-tuning	*99.09*	*94.33*	**96.71**	**94.33**

Table 3
Comparison of our method to a classic marginal domain critic and an absence of critic, on the Office-Caltech dataset. We use 200 labels per domain except for DSLR which uses 100. Accuracy is reported as averages over of all bi-domain tasks. Best results overall are in bold.

	Amazon	*Caltech*	*DSLR*	*Webcam*	Avg	Min
No critic	*90.63*	*84.27*	*93.75*	*98.31*	91.74	84.27
Marginal critic	*91.32*	*85.46*	*92.71*	*96.07*	91.39	85.46
Joint critic (Ours)	*91.67*	*86.05*	*93.75*	*97.00*	**92.12**	**86.05**

Table 4
Comparison of our method to a classic marginal domain critic and an absence of critic, on the PACS dataset. We use 500 labels per domain. Accuracy is reported as Domain vs. Rest. Best results overall are in bold.

	Art vs. R	*Cartoon vs. R*	*Photo vs. R*	*Sketch vs. R*	Avg	Min
No critic	*84.03*	*85.62*	*78.74*	*60.45*	77.21	60.45
Marginal critic	*84.08*	*87.07*	*78.26*	*64.44*	78.46	64.44
Joint critic (Ours)	*77.15*	*88.61*	*83.41*	*71.52*	**80.18**	**71.52**

MNIST. Our domain invariant approach, combined with the same techniques, further improves over this strong baseline, by reducing the accuracy gap between the two domains and arriving at a higher accuracy on the most difficult domain (and also on average). Results are further improved by applying a final supervised fine-tuning step to our model without discriminator, and with classification loss re-weighted depending on the domain classification error. Note that this fine-tuning step, while improving classification, hampers the domain alignment and therefore the reusability of features for alternative tasks as well as the domain privacy. More details in Supplementary Note C.1.

Table 3 displays the average results of our bi-domain experiments on the Office-Caltech dataset. We observe that the joint critic (ours) is always better than the marginal one. We also observe that the joint critic, compared to the lack thereof, leads to more uniform results across domains (and therefore higher minimum accuracy), as well as higher average. These observations are consistent with our theoretical results.

Finally, Table 4 shows prediction performance on the more complex PACS dataset. We test our model on this data in a one-vs-rest setting, so that the model must learn to be invariant between one domain and the three remaining domains.

Again, we find that our model produces the best minimum and average accuracy in each scenario. We found that a trade-off may exist between Art and other domains. Although our method performs worse than competitors on this domain, we observe that it leads to domain accuracies more concentrated around the mean, and therefore a higher minimum accuracy. Additionally, we note that the average accuracy has also increased.

Lastly, we would like to reiterate that the problem of domain invariance has received considerably less attention in the context of deep neural networks than the tasks of domain adaptation and domain generalization. Our quantitative results as well as the multiple baseline results aim to provide useful reference values for future work on domain invariance.

6.3. Visual insights on learned representations

While results in the section above have verified quantitatively the performance of our proposed domain invariant network, we would like to also present some qualitative insights.

As a first experiment, we visualize how the representation of the Conv-Large model trained with our proposed approach becomes more task-specific and less domain-dependent throughout training. For this, we take samples from $\mathcal{P}_{\mathcal{Z}}$ and $\mathcal{Q}_{\mathcal{Z}}$, join them, and perform a low-dimensional embedding of the resulting distribution via UMAP (McInnes et al., 2018). Plots before and after training are shown in Fig. 5 (left). The visualization suggests that the two domains are strongly separated initially, but under the influence of domain invariant training, they collapse to the same regions in representation space. As expected, the learned representation also better resolves the different classes after training (here roughly given by the cluster structure).

As a second experiment, we present SVHN-like synthetic examples to our domain invariant network and vary the digit and the colors. Using the Layer-wise Relevance Propagation (LRP) explanation method (Bach et al., 2015), we then compute for each prediction the local response of the model. The LRP method identifies the contribution of each input pixel to the prediction. These pixel-wise contributions can also be seen as the summands of a linear model, and the latter forms a local interpretable surrogate for the original model. We refer to weights of this linear model as the 'LRP response map' (details on LRP and how to generate response maps are given in Supplementary Note C.2).

A selection of examples and associated LRP response maps is shown in Fig. 5 (right), featuring two digit classes and SVHN-like color variations. Although we would expect that style and color play a marginal role in representation space (our objective has enforced invariance between the colored SVHN and the black&white handwritten MNIST domains), recognizing such style and color variations remains an integral part of the neural network prediction strategy. We indeed observe that the model precisely adapts to the input digit by providing *domain-specific* response maps of corresponding colors. This strategy is therefore instrumental in the process of building the domain invariant representation.

7. Conclusion

Real-world data is often heterogeneous, subject to sub-population shifts, or coming from multiple domains. In this work, we have for the first time studied the problem of learning domain-invariant representations as measured by the joint Wasserstein distance. We have created a theoretical framework for semi-supervised domain invariance and have contributed several upper-bounds to the Wasserstein distance of joint distributions that links domain invariance to practical learning objectives.

In our benchmark experiments, we find that optimizing the resulting objective leads to high prediction accuracy on both domains while simultaneously achieving high domain invariance, which we also observe qualitatively on low-dimensional embedding visualizations. We have further observed, somewhat counterintuitively, that domain adversarial training can still result in a model that makes use of domain-specific features in order to arrive at the domain-invariant representations.

Our work allows for several future extensions. For example, it would be interesting to obtain a theoretical connection to other representation learning methods, in particular, contrastive learning, that may be integrated to our framework. Furthermore, an extension of our theory to domain generalization could enable

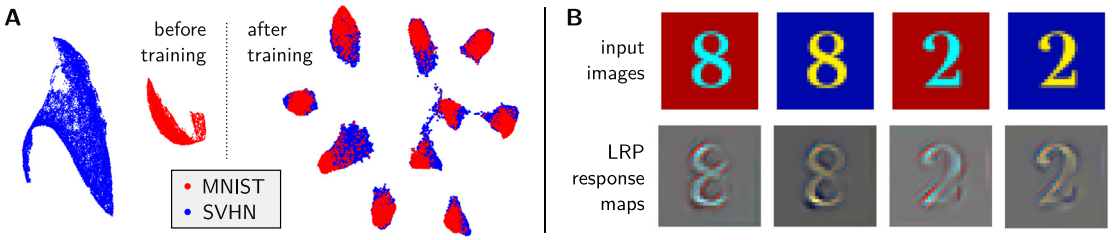

Fig. 5. Left: UMAP visualization of the extracted representation before and after training. Right: Response (extracted using LRP) of the model to various input digits with different style (color).

further applications and increase our understanding of domain generalization itself.

Overall, our work on domain invariance provides new theoretical insights as well as quantitative competitive results for a number of scenarios and baselines. We believe it thereby constitutes a useful first basis for further research on domain-invariant ML models and applications thereof.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

KRM was partly supported by the Institute of Information & Communications Technology Planning & Evaluation (IITP) grants funded by the Korea government(MSIT) (No. 2019-0-00079, Artificial Intelligence Graduate School Program, Korea University and No. 2022-0-00984, Development of Artificial Intelligence Technology for Personalized Plug-and-Play Explanation and Verification of Explanation). This work was supported in part by the German Ministry for Education and Research (BMBF), Germany under Grants 01IS14013A-E, 01GQ1115, 01GQ0850, 01IS18025A and 01IS18037A; and by JSPS KAKENHI, Japan Grant Number 20H00576, and 19H04071. Correspondence to TK, KRM and GM.

Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.neunet.2023.07.028.

References

Albuquerque, I., Monteiro, J., Darvishi, M., Falk, T. H., & Mitliagkas, I. (2019). Generalizing to unseen domains via distribution matching. arXiv preprint arXiv:1911.00804.

Amodei, D., Olah, C., Steinhardt, J., Christiano, P. F., Schulman, J., & Mané, D. (2016). Concrete problems in AI safety. arXiv preprint arXiv:1606.06565.

Arjovsky, M., Chintala, S., & Bottou, L. (2017). Wasserstein generative adversarial networks. In *International conference on machine learning* (pp. 214–223). PMLR.

Bach, S., Binder, A., Montavon, G., Klauschen, F., Müller, K.-R., & Samek, W. (2015). On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, *10*(7), Article e0130140.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., & Vaughan, J. W. (2010). A theory of learning from different domains. *Machine Learning*, *79*(1), 151–175.

Blanchard, G., Lee, G., & Scott, C. (2011). Generalizing from several related classification tasks to a new unlabeled sample. In *Advances in neural information processing systems, Vol. 24* (pp. 2178–2186).

Chen, S., Harandi, M., Jin, X., & Yang, X. (2020). Domain adaptation by joint distribution invariant projections. *IEEE Transactions on Image Processing*, *29*, 8264–8277.

Cheng, L., & Pan, S. J. (2014). Semi-supervised domain adaptation on manifolds. *IEEE Transactions on Neural Networks and Learning Systems*, *25*(12), 2240–2249.

Courty, N., Flamary, R., Habrard, A., & Rakotomamonjy, A. (2017). Joint distribution optimal transportation for domain adaptation. In *NIPS 2017*.

Cubuk, E. D., Zoph, B., Shlens, J., & Le, Q. V. (2020). RandAugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops* (pp. 702–703).

Delage, E., & Ye, Y. (2010). Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Operations Research*, *58*(3), 595–612.

DeVries, T., & Taylor, G. W. (2017). Improved regularization of convolutional neural networks with cutout. arXiv preprint arXiv:1708.04552.

Donahue, J., Jia, Y., Vinyals, O., Hoffman, J., Zhang, N., Tzeng, E., et al. (2014). Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning* (pp. 647–655). PMLR.

Dou, Q., de Castro, D. C., Kamnitsas, K., & Glocker, B. (2019). Domain generalization via model-agnostic learning of semantic features. In *NeurIPS* (pp. 6447–6458).

Duchi, J. C., Hashimoto, T., & Namkoong, H. (2023). Distributionally robust losses for latent covariate mixtures. *Oper. Res.*, *71*(2), 649–664.

Duchi, J. C., & Namkoong, H. (2021). Learning models with uniform performance via distributionally robust optimization. *The Annals of Statistics*, *49*(3), 1378–1406.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., et al. (2016). Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, *17*(59), 1–35.

Goel, K., Gu, A., Li, Y., & Ré, C. (2021). Model patching: Closing the subgroup performance gap with data augmentation. In *ICLR*. OpenReview.net.

Gong, B., Shi, Y., Sha, F., & Grauman, K. (2012). Geodesic flow kernel for unsupervised domain adaptation. In *2012 IEEE conference on computer vision and pattern recognition* (pp. 2066–2073). IEEE.

Grandvalet, Y., Bengio, Y., et al. (2005). Semi-supervised learning by entropy minimization. In *CAP* (pp. 281–296).

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. C. (2017). Improved training of wasserstein GANs. In *NIPS* (pp. 5767–5777).

He, G., Liu, X., Fan, F., & You, J. (2020). Classification-aware semi-supervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops* (pp. 964–965).

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 770–778).

Jiang, P., Wu, A., Han, Y., Shao, Y., Qi, M., & Li, B. (2020). Bidirectional adversarial training for semi-supervised domain adaptation.. In *IJCAI* (pp. 934–940).

Kim, T., & Kim, C. (2020). Attract, perturb, and explore: Learning a feature alignment network for semi-supervised domain adaptation. In *European conference on computer vision* (pp. 591–607). Springer.

Kingma, D. P., & Ba, J. (2015). Adam: A method for stochastic optimization. In Y. Bengio, & Y. LeCun (Eds.), *3rd international conference on learning representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, conference track proceedings*.

Koh, P. W., Sagawa, S., Marklund, H., Xie, S. M., Zhang, M., Balsubramani, A., et al. (2021). WILDS: a benchmark of in-the-wild distribution shifts. In *Proceedings of machine learning research*: *vol. 139*, *ICML* (pp. 5637–5664). PMLR.

LeCun, Y. (1998). The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/.

Li, J., Du, Z., Zhu, L., Ding, Z., Lu, K., & Shen, H. T. (2021). Divergence-agnostic unsupervised domain adaptation by adversarial attacks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Li, K., Liu, C., Zhao, H., Zhang, Y., & Fu, Y. (2021). ECACL: A holistic framework for semi-supervised domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 8578–8587).

Li, H., Pan, S. J., Wang, S., & Kot, A. C. (2018). Domain generalization with adversarial feature learning. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 5400–5409).

Li, Y., Tian, X., Gong, M., Liu, Y., Liu, T., Zhang, K., et al. (2018). Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the european conference on computer vision* (pp. 624–639).

Li, H., Wang, Y., Wan, R., Wang, S., Li, T.-Q., & Kot, A. (2020). Domain generalization for medical imaging classification with linear-dependency regularization. *Advances in Neural Information Processing Systems*, *33*, 3118–3129.

Li, C., Xu, T., Zhu, J., & Zhang, B. (2017). Triple generative adversarial nets. In *NIPS* (pp. 4088–4098).

Li, D., Yang, Y., Song, Y.-Z., & Hospedales, T. M. (2017). Deeper, broader and artier domain generalization. In *Proceedings of the IEEE international conference on computer vision* (pp. 5542–5550).

Liu, H., Long, M., Wang, J., & Jordan, M. (2019). Transferable adversarial training: A general approach to adapting deep classifiers. In *International conference on machine learning* (pp. 4013–4022). PMLR.

Liu, Y.-H., Ren, C.-X., Xu, X.-L., & Huang, K.-K. (2022). Bures joint distribution alignment with dynamic margin for unsupervised domain adaptation. arXiv preprint arXiv:2203.06836.

López-Paz, D., Hernández-Lobato, J. M., & Schölkopf, B. (2012). Semi-supervised domain adaptation with non-parametric copulas. In *NIPS* (pp. 674–682).

Mao, X., Ma, Y., Yang, Z., Chen, Y., & Li, Q. (2019). Virtual mixup training for unsupervised domain adaptation. arXiv preprint arXiv:1905.04215.

McInnes, L., Healy, J., Saul, N., & Großberger, L. (2018). UMAP: Uniform manifold approximation and projection. *Journal of Open Source Software*, *3*(29), 861.

Meng, R., Li, X., Chen, W., Yang, S., Song, J., Wang, X., et al. (2022). Attention diversification for domain generalization. In *Computer vision – ECCV 2022* (pp. 322–340).

Miyato, T., Kataoka, T., Koyama, M., & Yoshida, Y. (2018). Spectral normalization for generative adversarial networks. In *International conference on learning representations*.

Miyato, T., Maeda, S.-i., Koyama, M., & Ishii, S. (2018). Virtual adversarial training: a regularization method for supervised and semi-supervised learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *41*(8), 1979–1993.

Montavon, G., Müller, K.-R., & Cuturi, M. (2016). Wasserstein training of restricted Boltzmann machines. In *NIPS* (pp. 3711–3719).

Muandet, K., Balduzzi, D., & Schölkopf, B. (2013). Domain generalization via invariant feature representation. In *International conference on machine learning* (pp. 10–18). PMLR.

Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., & Ng, A. Y. (2011). Reading digits in natural images with unsupervised feature learning. In *NIPS workshop on deep learning and unsupervised feature learning 2011*.

Peyré, G., Cuturi, M., et al. (2019). Computational optimal transport: With applications to data science. *Foundations and Trends® in Machine Learning*, *11*(5–6), 355–607.

Qin, C., Wang, L., Ma, Q., Yin, Y., Wang, H., & Fu, Y. (2021). Contradictory structure learning for semi-supervised domain adaptation. In *Proceedings of the 2021 SIAM international conference on data mining* (pp. 576–584). SIAM.

Rahimian, H., & Mehrotra, S. (2019). Distributionally robust optimization: A review. arXiv preprint arXiv:1908.05659.

Rasmus, A., Berglund, M., Honkala, M., Valpola, H., & Raiko, T. (2015). Semi-supervised learning with ladder networks. In *NIPS* (pp. 3546–3554).

Redko, I., Habrard, A., & Sebban, M. (2017). Theoretical analysis of domain adaptation with optimal transport. In *Joint european conference on machine learning and knowledge discovery in databases* (pp. 737–753). Springer.

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016). "Why should I trust you?": Explaining the predictions of any classifier. In *KDD* (pp. 1135–1144). ACM.

Sagawa, S., Koh, P. W., Hashimoto, T. B., & Liang, P. (2019). Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. arXiv preprint arXiv:1911.08731.

Saito, K., Kim, D., Sclaroff, S., Darrell, T., & Saenko, K. (2019). Semi-supervised domain adaptation via minimax entropy. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 8050–8058).

Saito, K., Watanabe, K., Ushiku, Y., & Harada, T. (2018). Maximum classifier discrepancy for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3723–3732).

Samek, W., Montavon, G., Lapuschkin, S., Anders, C. J., & Müller, K.-R. (2021). Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE*, *109*(3), 247–278.

Sharifi-Noghabi, H., Asghari, H., Mehrasa, N., & Ester, M. (2020). Domain generalization via semi-supervised meta learning. arXiv preprint arXiv:2009.12658.

Shen, J., Qu, Y., Zhang, W., & Yu, Y. (2018). Wasserstein distance guided representation learning for domain adaptation. In *AAAI* (pp. 4058–4065). AAAI Press.

Shimodaira, H. (2000). Improving predictive inference under covariate shift by weighting the log-likelihood function. *Journal of Statistical Planning and Inference*, *90*(2), 227–244.

Shu, R., Bui, H. H., Narui, H., & Ermon, S. (2018). A DIRT-T approach to unsupervised domain adaptation. In *6th International Conference on Learning Representations, Conference Track Proceedings*. OpenReview.net.

Sicilia, A., Zhao, X., & Hwang, S. J. (2023). Domain adversarial neural networks for domain generalization: when it works and how to improve. *Mach. Learn.*, *112*(7), 2685–2721.

Sugiyama, M., Krauledat, M., & Müller, K.-R. (2007). Covariate shift adaptation by importance weighted cross validation. *Journal of Machine Learning Research*, *8*(5), 985–1005.

Villani, C. (2008). *Optimal transport: Old and new, Vol. 338*. Springer Science & Business Media.

Xiao, N., & Zhang, L. (2021). Dynamic weighted learning for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition* (pp. 15242–15251).

Xu, M., Zhang, J., Ni, B., Li, T., Wang, C., Tian, Q., et al. (2020). Adversarial domain adaptation with domain mixup. In *Proceedings of the AAAI conference on artificial intelligence, Vol. 34* (pp. 6502–6509).

Yang, L., Wang, Y., Gao, M., Shrivastava, A., Weinberger, K. Q., Chao, W.-L., et al. (2021). Deep co-training with task decomposition for semi-supervised domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision* (pp. 8906–8916).

Zhang, L., & Gao, X. (2022). Transfer adaptation learning: A decade survey. *IEEE Transactions on Neural Networks and Learning Systems*.

Zhang, W., Li, C., & Teng, S. (2021). Joint discriminative distribution adaptation and manifold regularization for unsupervised domain adaptation. In *2021 IEEE 24th international conference on computer supported cooperative work in design* (pp. 226–231). IEEE.

Zhang, W., Ouyang, W., Li, W., & Xu, D. (2018). Collaborative and adversarial network for unsupervised domain adaptation. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 3801–3809).

Zhou, F., Jiang, Z., Shui, C., Wang, B., & Chaib-draa, B. (2021). Domain generalization via optimal transport with metric similarity learning. *Neurocomputing*, *456*, 469–480.

Zhou, K., Liu, Z., Qiao, Y., Xiang, T., & Loy, C. C. (2022). Domain generalization: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.

Zhou, K., Yang, Y., Qiao, Y., & Xiang, T. (2021). Domain generalization with MixStyle. In *ICLR*. OpenReview.net.