DISCREPANCY-OPTIMAL META-LEARNING FOR DOMAIN GENERALIZATION

Anonymous authors
Paper under double-blind review

ABSTRACT

This work attempts to tackle the problem of domain generalization (DG) via learning to reduce domain shift with an episodic training procedure. In particular, we measure the domain shift with $\mathcal{Y}$-discrepancy and learn to optimize $\mathcal{Y}$-discrepancy between the unseen target domain and source domains only using source-domain samples. Theoretically, we give a PAC-style generalization bound for discrepancy-optimal meta-learning and further make comparisons with other DG bounds including ERM and domain-invariant learning. The theoretical analyses show that there is a tradeoff between classification performance and computational complexity for discrepancy-optimal meta-learning. The theoretical results also shed light on a bilevel optimization algorithm for DG. Empirically, we evaluate the algorithm with DomainBed and achieves state-of-the-art results on two DG benchmarks.

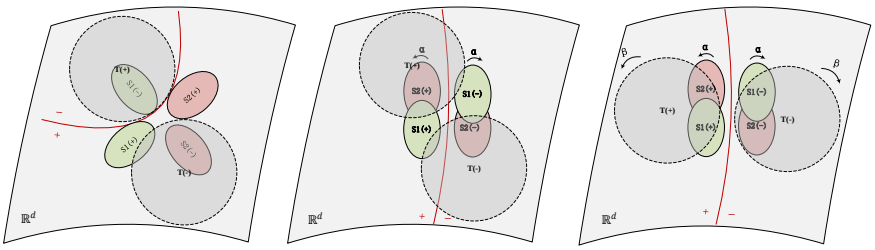
1 INTRODUCTION

Deep learning has achieved highly competitive performance on test data drawn from the same distribution as large training data. However, in practice there are many circumstances where access to target data is impossible. Research on *domain adaptation* (DA) uses unlabeled target data to transfer labeled source information to a specific target domain (Pan & Yang, 2009; Mansour et al., 2009). In recent years, *domain generalization* (DG) has gained increasing attention. Different from DA, DG aims to solve a more practical and challenging problem, where the target domain is invisible and thus explicit training on the target is impossible (Blanchard et al., 2011; Muandet et al., 2013).

To tackle the invisibility of target domain for DG, a naive method is empirical risk minimization (ERM) (Vapnik, 1999), which relays on the diversity of source data to achieve better generalization performance for unseen target domains (Shankar et al., 2018; Volpi et al., 2018; Gulrajani & Lopez-Paz, 2020). However, it is almost impossible in practice to acquire training data from enough domains to achieve promising DG performance. Another route is to better utilize the feature distributions of source domains, some previous work learn domain-invariant representation across source domains (Albuquerque et al., 2020; Xiao et al., 2021). As shown in Figure 1 (b), the domain-invariant representation can regularize the feature space to improve classification performance across source domains (α-arrows), but the discrepancy between the unseen target domain and source domains still limits the generalization performance.

In order to effectively utilize the available source data to improve generalization performance, episodic training process (Finn et al., 2017) is recently applied to DG (Li et al., 2018a; Balaji et al., 2018; Li et al., 2019b), i.e., randomly extracting a meta-target sample and meta-source samples from available source data to simulate domain shift in each iteration. This work attempts to train a discrepancy-optimal meta-learner that gains experience during the episodic training process so as to minimize the domain discrepancy between any unseen target domain and the available source domains. We theoretically show the effectiveness of discrepancy-optimal meta-learning for DG with a PAC-style learning bound and further make comparisons with ERM and domain-invariant learning for DG. The theoretical analyses mainly show that there is a tradeoff between generalization performance and computational complexity for the proposed discrepancy-optimal meta-learning framework.

In order to make the theoretical idea into practice, we design an bilevel optimization problem for discrepancy-optimal meta-learning, where the inner-loop objective is to minimize $\mathcal{Y}$-discrepancy across meta-source samples while the outer-loop objective is to minimize $\mathcal{Y}$-discrepancy between meta-target and meta-source samples. The effectivenesses of inner-loop and outer-loop optimization



(a) Baseline (for comparison). (b) Domain-invariant Learning. (c) Discrepancy-optimal Meta-learning.

Figure 1: Motivations for discrepancy-optimal meta-learning. In the feature space $\mathbb{R}^d$, T denotes the range of target domains and S1, S2 denote the ranges of two source domains. $(+)$ and $(-)$ denote positive examples and negative examples, respectively. A real-world case is shown in Appendix F

.

is shown as the α-arrows and β-arrows in Figure 1 (c), respectively. A theoretical analysis from a geometric perspective shows that the collaborative effectiveness of the inner-loop and outer-loop optimization is to minimize $\mathcal{Y}$-discrepancy between target domain and the convex hull of source domains. Empirically, we conduct experiments on DomainBed (Gulrajani & Lopez-Paz, 2020) and evaluate on two DG benchmarks. Results show that our method is highly effective and achieves state-of-the-art performances. The code will be released at `https://anonymous.com`.

2 Related Work

Domain Generalization. A promising solution for DG is to learn domain-invariant features across source domains (Muandet et al., 2013; Deshmukh et al., 2019; Li et al., 2017; 2018b;c; Albuquerque et al., 2020; Deng et al., 2020). Early work perform kernel-based optimization to learn an invariant feature mapping to RKHS (Muandet et al., 2013; Deshmukh et al., 2019). Neural-network methods have achieved promising results in recent years (Li et al., 2017). Li et al. (2018b) employ maximum mean discrepancy (MMD) constraints via an Auto-Encoder. Recently, adversarial training strategies have shown highly effectiveness for DG via reducing covariant shift (Li et al., 2018c; Albuquerque et al., 2020; Deng et al., 2020) or reducing both covariant shift and conditional shift (Li et al., 2018c). Another method utilizes Variational Bayes for domain-invariant learning (Xiao et al., 2021). Instead of constructing invariant feature spaces, some methods use data augmentation (Shankar et al., 2018; Volpi et al., 2018; Qiao et al., 2020; Zhou et al., 2020) to enrich source diversity, which also shows usefulness for DG. We borrow the idea of extracting domain-invariant feature representations from the previous work, but our algorithm also optimizes the discrepancy between the expected target domain and source domains via meta-learning.

Meta-learning. Meta-learning provides a framework to gain learning experiences for future tasks over multiple training episodes, which often cover a distribution of related tasks (Thrun & Pratt, 1998; Baxter, 2000). For neural models, gradient-based meta-learning methods (Finn et al., 2017; Grant et al., 2018; Gordon et al., 2018) are successfully applied to few-shot learning. The episodic training procedure is also introduced to address DG (Li et al., 2018a; Balaji et al., 2018; Li et al., 2019a;b; Dou et al., 2019; Du et al., 2020) via simulating domain shift between source domains and unseen target domains. To alleviate domain shift during episodic training, MLDG (Li et al., 2018a) follows the update rule of MAML (Finn et al., 2017), minimizing the risk on meta-target data upon the optimized parameters by meta-source data. A limitation of using the source and target task objectives directly for inner-loop and outer-loop optimization might be sub-optimal, since it is highly abstracted from feature representation (Dou et al., 2019). MetaReg (Balaji et al., 2018) adds a classifier's weights regularization term to the objective of source task for inner-loop optmization to prouduce a more general classifier. Li et al. (2019b) improve MetaReg by training a feature-critic network to obtain a more general feature extractor. Dou et al. (2019) keeps source task risk as the inner-loop objective but replace the outer-loop objective by global class alignment and local sample clustering objectives to explictly regularize the feature space. Our method performs episodic training procedure based on MAML (Finn et al., 2017), but different from the previous work in that we optimize the discrepancy between target and source domains while the inner objective is to minimize the dicrepancy accorss source domains. We also show that our method has strong theoretical foundations.

3 PRELIMINARIES

3.1 SET-UP

Notations. Let $\mathcal{X}$ be the input space and $\mathcal{Y} = \{0, 1\}$ be the output space. Following Blanchard et al. (2011) and Muandet et al. (2013), we define a domain as a joint distribution $\mathbb{P}_{XY}$ ($\mathbb{P}$ for brevity) on $\mathcal{X} \times \mathcal{Y}$ and let $\mathfrak{P}_{\mathcal{X}\times\mathcal{Y}}$ ($\mathfrak{P}$ for brevity) denote the set of all domains. We assume that all domains are drawn *i.i.d.* according to $\mathscr{P}$ on $\mathfrak{P}_{\mathcal{X}\times\mathcal{Y}}$.

We denote a set of N source domains as $\mathcal{S} = \{\mathbb{S}^i\}_{1\le i\le N}$, whose elements are drawn *i.i.d.* according to a distribution $\mathscr{P}$. Empirically, there exist training samples $\hat{\mathcal{S}} = \{\hat{\mathbb{S}}^i\}_{1\le i\le N}$ of N source domains. We assume that the number of source domains $N \in \{1, \ldots, N_m\}$ obeys a distribution $p(N)$ on the set $\{1, \ldots, N_m\}$, where N_m is the maximum number of source domains. For each $\mathbb{S}^i$ ($i \in \{1, \ldots, N\}$), training sample $\hat{\mathbb{S}}^i = \{(x_k^i, y_k^i)\}_{1\le k\le n_i}$ are drawn *i.i.d.* according to $\mathbb{S}^i$. For brevity, we assume all domains have the same number of training samples, i.e., $n_T = n_1 = \ldots = n_N = n$.

A *decision function* $h \in \mathcal{H} : \mathcal{X} \to \mathcal{Y}$ maps the input space into the output space and a *loss function* $\ell \circ h : \mathcal{X} \times \mathcal{Y} \to \mathbb{R}_+$ is a real-valued function that maps from a data point $(x, y) \in \mathcal{Z}$ to a non-negative real number, e.g., $\ell(h(x), y) = |h(x) - y|$ $(x, y) \in \mathcal{Z}$ (Ben-David et al., 2010). The *expected error* $\epsilon_{\mathbb{D}}(h)$ and *empirical error* $\hat{\epsilon}_{\hat{\mathbb{D}}}(h)$ on a domain $\mathbb{D}$ w.r.t. a decision function h are represented as follows:

$$\epsilon_{\mathbb{D}}(h) := \mathbb{E}_{(x,y)\sim\mathbb{D}}[\ell(h(x), y)]; \qquad \hat{\epsilon}_{\hat{\mathbb{D}}}(h) := \frac{1}{|\hat{\mathbb{D}}|} \sum_{(x,y)\in\hat{\mathbb{D}}} \ell(h(x), y) \tag{1}$$

In deep learning, a *neural decision function* $h = g \circ f$ can be regarded as a composition function of a *feature extractor* $f \in \mathcal{F} : \mathcal{X} \to \mathbb{R}^d$ and a *classifier* $g \in \mathcal{G} : \mathbb{R}^d \to \{0, 1\}$ and the corresponding *expected error* and *empirical error* can be represented as $\epsilon_{\mathbb{D}_f}(g) := \epsilon_{\mathbb{D}}(g \circ f)$ and $\hat{\epsilon}_{f(\hat{\mathbb{D}})}(g) := \hat{\epsilon}_{\hat{\mathbb{D}}}(g \circ f)$, respectively. We denote $\mathbb{D}_f$ as a distribution on $\mathbb{R}^d$, where $\mathbb{D}_f(f(x)) = \mathbb{D}(x), x \in \mathcal{X}$.

Learning algorithm & Objective. A *learning algorithm* $A : \{\mathcal{Z}^{N\times n} : N \in [N_m]\} \to \mathcal{H}$ for DG maps from source data $\hat{\mathcal{S}} \in \mathcal{Z}^{N\times n}$ $(N \in [N_m])$ to a decision function $h \in \mathcal{H}$ and the set of all learning algorithms is represented as $\mathcal{A}(\mathcal{H}, \mathcal{Z}^{N\times n})$ or $\mathcal{A}$ for brevity. We assume that $\{A(\hat{\mathcal{S}}) : A \in \mathcal{A}, \hat{\mathcal{S}} \in \mathcal{Z}^{N\times n}, N \in [N_m]\} \subseteq \mathcal{H}$. Following Maurer & Jaakkola (2005), a nautral measure for the performance of a given algorithm A for DG is represented as:

$$\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)} \left[\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]\right] \tag{2}$$

For a neural decision function $g \circ f$, we denote the partial effect of A on the feature extractor as an algorithm, $A^f : \{\mathcal{Z}^{N\times n} : N \in [N_m]\} \to \mathcal{F}$ and similarly, $A^g : \{\mathcal{Z}^{N\times n} : N \in [N_m]\} \to \mathcal{G}$.

A *meta-algorithm* (or *meta-learner*) $\mathbf{A} : \{\hat{\mathcal{D}}\} \to \mathcal{A}(\mathcal{F}, \mathcal{Z}^{N\times n})$ maps from a meta-sample $\hat{\mathcal{D}}$ to an algorithm $A \in \mathcal{A}(\mathcal{F}, \mathcal{Z}^{N\times n})$. A *meta-sample* $\hat{\mathcal{D}}$ denotes M pairs of meta-source and meta-target samples $\hat{\mathcal{D}} = \{(\hat{\mathcal{S}}_i, \hat{\mathbb{T}}_i)\}_{1\le i\le M}$, where the size of meta-sample $M \approx N_m \cdot 2^{N_m - 1}$ in that we consider all possible numbers of meta-source domains in the episodic training process. In meta-learning for DG, a meta-algorithm $\mathbf{A}$ learns an algorithm $A \leftarrow \mathbf{A}(\hat{\mathcal{D}})$ from the meta-sample $\hat{\mathcal{D}}$, which can learn such decision function $h \leftarrow A(\hat{\mathcal{S}})$ from source-domain data $\hat{\mathcal{S}}$.

3.2 DOMAIN DISCREPANCY

We will present $\mathcal{Y}$-discrepancy in this section, which has been widely used in DA (Zhang et al., 2012). In deep learning, we follow the previous DA work to optimize the feature space to minimize the domain discrepancy (Ganin et al., 2016; Zhao et al., 2018), thus we also give the definition of $\mathcal{Y}$-discrepancy based on the learning algorithm for feature extractor A^f.

Definition 1. **($\mathcal{Y}$-discrepancy):** *Let $\mathcal{H}$ be a decision function class on $\mathcal{X}$ with finite VC-dimension. The $\mathcal{Y}$-discrepancy* $\mathrm{disc}_{\mathcal{Y}}(\mathbb{S}, \mathbb{T})$ *between two domains $\mathbb{S}$ and $\mathbb{T}$ and its empirical version* $\hat{\mathrm{disc}}_{\mathcal{Y}}(\hat{\mathbb{S}}, \hat{\mathbb{T}})$ *are defined as:*

$$\mathrm{disc}_{\mathcal{Y}}(\mathbb{S}, \mathbb{T}) := \sup_{h\in\mathcal{H}} \left|\epsilon_{\mathbb{T}}(h) - \epsilon_{\mathbb{S}}(h)\right|; \quad \hat{\mathrm{disc}}_{\mathcal{Y}}(\hat{\mathbb{S}}, \hat{\mathbb{T}}) := \sup_{h\in\mathcal{H}} \left|\hat{\epsilon}_{\hat{\mathbb{T}}}(h) - \hat{\epsilon}_{\hat{\mathbb{S}}}(h)\right| \tag{3}$$

In neural models, $\mathcal{Y}$-discrepancy between domains is computed in the feature space, i.e., output space of the feature extractor $A^f(\hat{\mathcal{S}})$, with the classification function $g \in \mathcal{G}$,

$$\begin{aligned} \mathrm{disc}_{\mathcal{Y}}\left(A^f(\hat{\mathcal{S}})(\mathbb{S},\mathbb{T})\right) &:= \sup_{g\in\mathcal{G}} \left| \epsilon_{\mathbb{T}}\big(g\circ A^f(\hat{\mathcal{S}})\big) - \epsilon_{\mathbb{S}}\big(g\circ A^f(\hat{\mathcal{S}})\big)\right|; \\ \hat{\mathrm{disc}}_{\mathcal{Y}}\left(A^f(\hat{\mathcal{S}})(\hat{\mathbb{S}},\hat{\mathbb{T}})\right) &:= \sup_{g\in\mathcal{G}} \left| \hat{\epsilon}_{\hat{\mathbb{T}}}\big(g\circ A^f(\hat{\mathcal{S}})\big) - \hat{\epsilon}_{\hat{\mathbb{S}}}\big(g\circ A^f(\hat{\mathcal{S}})\big)\right| \end{aligned} \tag{4}$$

It is clear that $\mathcal{Y}$-discrepancy defines a pseudo distance between a pair of domains in that it satisfies symmetry and the triangle inequality but not satisfies identity of indiscernibles since $\mathrm{disc}_{\mathcal{Y}}(\mathbb{S},\mathbb{T}) = 0 \nRightarrow \mathbb{S} = \mathbb{T}$.

$\mathcal{Y}$-discrepancy can measure not only covariant shift between domains, but also conditional shift between domains (Zhang et al., 2012). Therefore, we choose $\mathcal{Y}$-discrepancy as a measurement for domain discrepancy in following derivations of DG theory.

4 Learning Guarantees

In this section, we first give a PAC-style learning bound for DG under the discrepancy-optimal meta-learning framework and then make comparisons with ERM and domain-invariant learning for DG.

4.1 Discrepancy-based DG Bounds

We will bound the measure of DG performance w.r.t. an algorithm as defined in Eq. (2), under the discrepancy-optimal meta-learning framework given a meta-sample $\hat{\mathcal{D}} = \{(\hat{\mathcal{S}}_i, \hat{\mathbb{T}}_i)\}_{1\le i\le M}$.

Lemma 1. (Discrepancy-based DG Bound) *Given any $\xi, \xi' > 0$, for any $n \ge \frac{8\mathcal{B}^2}{\xi^2}$ and any $M \ge \frac{8\mathcal{B}^2}{(\xi')^2}$, we have for any $\delta > 0$, with probability at least $1-2\delta$,*

$$\begin{aligned} &\left| \mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)}\big[\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]\big] - \frac{1}{M}\sum_{k:\hat{\mathcal{S}}_k\in\hat{\mathcal{D}}} \frac{1}{|\hat{\mathcal{S}}_k|} \sum_{i:\hat{\mathbb{S}}_k^i\in\hat{\mathcal{S}}_k} \hat{\epsilon}_{\hat{\mathbb{S}}_k^i}(A(\hat{\mathcal{S}}_k)) \right| \\ \le &\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)} \mathbb{E}_{\mathbb{T}\sim\mathscr{P}}\left[\,\mathrm{disc}_{\mathcal{Y}}\left(A^f(\hat{\mathcal{S}})(\widetilde{\mathcal{S}},\mathbb{T})\right)\right] + \Xi(\xi,\delta,\hat{\mathcal{S}}) + \Pi(\xi',\delta,\hat{\mathcal{D}}) + \mathcal{B}\delta \end{aligned} \tag{5}$$

where $\widetilde{\mathcal{S}} = \frac{1}{N}\sum_{i:\mathbb{S}^i\in\hat{\mathcal{S}}} \mathbb{S}^i$ and the complexity terms corresponding to the data size are $\Xi(\xi,\delta,\hat{\mathcal{S}}) \propto \mathcal{O}(1/\sqrt{n})$ and $\Pi(\xi',\delta,\hat{\mathcal{D}}) \propto \mathcal{O}(1/\sqrt{M})$. Proof can be found in Appendix B.1.

Lemma 1 bounds the difference between algorithmic performance of DG and empirical classification error of meta-source samples with the expectation of $\mathcal{Y}$-discrepancy between target and source domains over random meta-samples. Next, we will bound the expected $\mathcal{Y}$-discrepancy over random meta-samples in the following lemma.

Lemma 2. (Meta-optimizing $\mathcal{Y}$-discrepancy (Eq. (5))) *Given any $\varepsilon, \varepsilon' > 0$, for any $n \ge \frac{8\mathcal{B}^2}{\varepsilon^2}$ and any $M \ge \frac{8\mathcal{B}^2}{(\varepsilon')^2}$, we have for any $\delta > 0$, with probability at least $1-3\delta$,*

$$\begin{aligned} &\left| \mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)} \mathbb{E}_{\mathbb{T}\sim\mathscr{P}}\left[\,\mathrm{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\widetilde{\mathcal{S}},\mathbb{T})\right)\right] - \frac{1}{M}\sum_{i=1}^{M} \hat{\mathrm{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}}_i)(\hat{\mathcal{S}}_i,\hat{\mathbb{T}}_i)\big) \right| \\ \le &\Xi(\varepsilon,\delta,\hat{\mathcal{S}}) + \Psi(\varepsilon',\delta,\hat{\mathcal{D}}) + 2\mathcal{B}\delta \end{aligned} \tag{6}$$

where the complexity terms corresponding to data size are $\Xi(\varepsilon,\delta,\hat{\mathcal{S}}) \propto \mathcal{O}(1/\sqrt{n})$ and $\Psi(\varepsilon',\delta,\hat{\mathcal{D}}) \propto \mathcal{O}(1/\sqrt{M})$. Proof can be found in Appendix B.2.

Finally, we combine the above two lemmas and obtain the generalization bound for DG via discrepancy-optimal meta-learning in the following theorem.

Table 1: Comparison with related DG and DA learning bounds.

	Objective (LHS)	Upper bound (RHS)		
		Empirical Term	Complexity Term	Other Term
Ours	$\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}[\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]]$	Classification error emp. $\mathcal{Y}-$discrepancy	$\mathcal{O}(1/\sqrt{n})+\mathcal{O}(1/\sqrt{M})$	$\mathcal{O}(\delta)$
ERM (DG) B.4	$\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}[\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]]$	Classification error	$\mathcal{O}(1/\sqrt{n})+\mathcal{O}(1/\sqrt{N_m})$	$\mathcal{O}(\delta)$
Domain-invariant (DA) B.5	$\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))$	Classification error	$\mathcal{O}(1/\sqrt{n})$	-
Domain-invariant (DG) B.6	$\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]$	Classification error	$\mathcal{O}(1/\sqrt{n})$	γ

where $M \approx N_m \cdot 2^{N_m-1}$ in episodic training and $\gamma := \mathbb{E}_{\mathbb{T}\sim\mathscr{P}}\big[\operatorname{disc}_{\mathcal{Y}}\big(A^f(\hat{\mathcal{S}})(\overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})},\mathbb{T})\big)\big], \overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})} = \arg\min_{\overline{\mathbb{T}}\in\operatorname{conv}(\mathcal{S})}\operatorname{disc}_{\mathcal{Y}}(A^f(\hat{\mathcal{S}})(\overline{\mathbb{T}},\mathbb{T}))$

Theorem 1. (DG Bound based on Discrepancy-optimal Meta-learning) *Given any $\varepsilon,\varepsilon',\varepsilon''>0$, for any $n \geq \frac{8\mathcal{B}^2}{\varepsilon^2}$ and any $M \geq \max\big\{\frac{8\mathcal{B}^2}{(\varepsilon')^2},\frac{8\mathcal{B}^2}{(\varepsilon'')^2}\big\}$, we have for any $\delta>0$, with probability at least $1-5\delta$,*

$$\begin{aligned}&\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}[\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]]-\frac{1}{M}\sum_{k:\hat{\mathcal{S}}_k\in\hat{\mathcal{D}}}\frac{1}{|\hat{\mathcal{S}}_k|}\sum_{i:\hat{\mathbb{S}}^i_k\in\hat{\mathcal{S}}_k}\hat{\epsilon}_{\hat{\mathbb{S}}^i_k}(A(\hat{\mathcal{S}}_k))\right|\\ \leq&\frac{1}{M}\sum_{i=1}^{M}\hat{\operatorname{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}}_i)(\hat{\mathcal{S}}_i,\hat{\mathbb{T}}_i)\big)+\Xi(\varepsilon,\delta,\hat{\mathcal{S}})+\Pi(\varepsilon',\delta,\hat{\mathcal{D}})+\Psi(\varepsilon'',\delta,\hat{\mathcal{D}})+3\mathcal{B}\delta\end{aligned} \tag{7}$$

where the complexity terms corresponding to data size are $\Xi(\varepsilon,\delta,\hat{\mathcal{S}})\propto\mathcal{O}(1/\sqrt{n})$, $\Pi(\varepsilon',\delta,\hat{\mathcal{D}})\propto\mathcal{O}(1/\sqrt{M})$, $\Psi(\varepsilon'',\delta,\hat{\mathcal{D}})\propto\mathcal{O}(1/\sqrt{M})$. Proof can be found in Appendix B.3.

Theorem 1 shows that when applying the discrepancy-optimal meta-learning on a meta-sample $\hat{\mathcal{D}}=\{(\hat{\mathcal{S}}_i,\hat{\mathbb{T}}_i)\}_{1\leq i\leq M}$, the difference between algorithmic performance of DG and empirical classification error of meta-source samples can be bounded mainly by the empirical $\mathcal{Y}$-discrepancy between meta-source and meta-target samples as well as the complexity term $\mathcal{O}(1/\sqrt{n})+\mathcal{O}(1/\sqrt{M})$.

4.2 Comparison with Related Bounds

We first compare with the learning bound of ERM for DG and explain the tradeoff between classification performance and computational complexity under the discrepancy-optimal meta-learning framework for DG. We also compare with domain-invariant learning for DA (Zhang et al., 2012) & DG (Albuquerque et al., 2020) and analyze the motivation of meta-learning strategy for optimizing the domain discrepancy for DG. The theoretical results are listed in Table 1.

Comparison with ERM. As shown in Table 1, the complexity term of ERM bound for DG relates to the number of training examples in each domain $\mathcal{O}(1/\sqrt{n})$ and the maximum number of source domains $\mathcal{O}(1/\sqrt{N_m})$. We can find that the complexity term of the proposed discrepancy-optimal meta-learning is better than the complexity terms of ERM since $\mathcal{O}(1/\sqrt{M})\approx\mathcal{O}(1/\sqrt{N_m\cdot 2^{N_m-1}})\ll\mathcal{O}(1/\sqrt{N_m})$. This means that the discrepancy-optimal meta-learning bound for DG has the potential to be better than the ERM when the empirical discrepancy term can be sufficiently minimized. Intuitively, this shows that meta-learning with episodic training process can utilize the source domains to optimize the domain discrepancy for DG, and there is a tradeoff between the performance and the computational complexity. We show the empirical results of the tradeoff in Appendix G.

Comparison with domain-invariant learning. We can find that DA and DG based on domain-invariant learning are tend to minimize $\mathcal{Y}$-discrepancy between target and source domains and $\mathcal{Y}$-discrepancy across source domains, respectively. In particular, the DG bound via domain-invariant learning consists of an approximating term γ, which is defines as the smallest expected $\mathcal{Y}$-discrepancy between any random target domain and the convex hull of source domains. However, γ can not be optimized in domain-invariant learning, thus it is unable to be estimated the tightness of the domain-invariant bound for DG and the induced algorithm may not has sufficient capacity to reduce the domain shift for DG. While the proposed discrepancy-optimal meta-learning for DG optimizes the $\mathcal{Y}$-discrepancy between target and source domains directly with a meta-sample and the corresponding bound can be optimized. Besides, Corollary 1 will show that a bilevel optimization algorithm under the discrepancy-optimal meta-learning framework can effectively optimize the term γ.

5 ALGORITHM

We will present a meta-learning algorithm via bilevel optimization to utilize the meta-sample for optimizing $\mathcal{Y}$-discrepancy between target and source domains in Section 5.1 and then show a practical meta-learning procedure in Section 5.2.

5.1 LEARNING TO MINIMIZE DOMAIN DISCREPANCY

Given a meta-sample $\hat{\mathcal{D}} = \{(\hat{\mathcal{D}}^i_{tr}, \hat{\mathbb{D}}^i_{te})\}_{1\le i\le M}$, the empirical term in Theorem 1 guides us to design an algorithm that takes the meta-source samples $\hat{\mathcal{D}}^i_{tr} \subset \hat{\mathcal{S}}$ as input and outputs a feature extractor (hypothesis) that minimizes $\mathcal{Y}$-discrepancy between the meta-target sample $\hat{\mathbb{D}}^i_{te} \in \hat{\mathcal{S}}$ and meta-source samples $\hat{\mathcal{D}}^i_{tr}$. To this end, the meta-training and meta-test problems are non-symmetric in the episodic training process (Finn et al., 2017). It is a natural idea to use bilevel optimization to tackle such a non-symmetric optimization problem. In particular, we specify the meta-learner $\mathbf{A}$ in Theorem 1 to optimize the initialized parameters ψ_0 so that an algorithm initialized with ψ_0 minimizes $\mathcal{Y}$-discrepancy across source domains (Eq. (10)) can also minimize $\mathcal{Y}$-discrepancy between target and source domains (Eq. (9)). Corollary 1 shows the affect of collaborative effectiveness of inner-loop and outer-loop optimization in bilevel optimization from a geometry perspective.

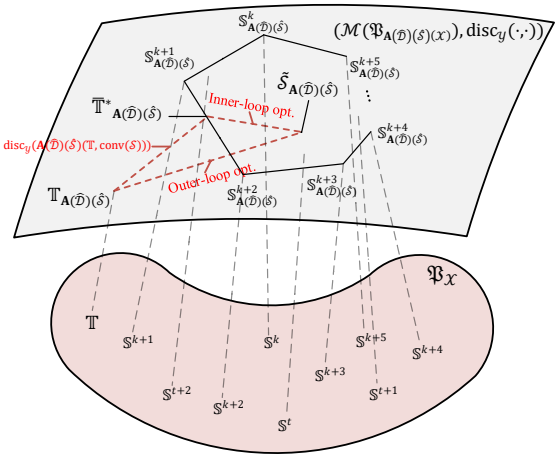


Figure 2: Geometric understanding. $\mathfrak{P}_{\mathcal{X}}$ denotes the space of domains (distributions) on input space and $\big(\mathcal{M}(\mathfrak{P}_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\mathcal{X})}), \mathrm{disc}_{\mathcal{Y}}(\cdot,\cdot)\big)$ denotes a pseudo-metric space of domains (distributions) on feature space.

Definition 2. (Bilevel Optimization) *We denote the the outer-loop and inner-loop objectives w.r.t. the feature extractor f_ψ as $\mathcal{L}_{out}$ and $\mathcal{L}_{in}$, respectively. ψ and ψ_0 denote the parameters of feature extractor in inner-loop optimization and outer-loop optimization, respectively. The bilevel optimization problem is defined as:*

$$\min_{\psi_0\in\Psi} \sum_{i\in[M]} \mathcal{L}_{out}(\psi_i^*, \psi_0; (\hat{\mathbb{D}}^i_{te}, \hat{\mathcal{D}}^i_{tr})), \quad \text{subject to } \psi_i^* \in \operatorname*{arg\,min}_{\psi\in\Phi(\psi_0)} \mathcal{L}_{in}(\psi, \psi_0; \hat{\mathcal{D}}^i_{tr}) \tag{8}$$

where $\Phi(\cdot)$ denotes a parameter constraint brought by parameter initialization. The outer-loop and inner-loop objectives are defined as follows:

$$\mathcal{L}_{out}(\psi_i^*, \psi_0; (\hat{\mathbb{D}}^i_{te}, \hat{\mathcal{D}}^i_{tr})) := \sum_{\hat{\mathbb{S}}^k\in\hat{\mathcal{D}}^i_{tr}} \alpha_i \hat{\mathrm{disc}}_{\mathcal{Y}}\big(f_{\psi_i^*}(\hat{\mathbb{D}}^i_{te}), f_{\psi_i^*}(\hat{\mathbb{S}}^k)\big) \tag{9}$$

$$\mathcal{L}_{in}(\psi, \psi_0; \hat{\mathcal{D}}^i_{tr}) := \sum_{\hat{\mathbb{S}}^k\in\hat{\mathcal{D}}^i_{tr}} \alpha_i \sum_{\hat{\mathbb{S}}^t\in\hat{\mathcal{D}}^i_{tr}, k\neq t} \hat{\mathrm{disc}}_{\mathcal{Y}}\big(f_{\psi \overset{\text{Init}}{\leftarrow} \psi_0}(\hat{\mathbb{S}}^k), f_{\psi \overset{\text{Init}}{\leftarrow} \psi_0}(\hat{\mathbb{S}}^t)\big) \tag{10}$$

where $\overset{\text{Init}}{\leftarrow}$ denotes parameter initialization. Definition 2 specifies the meta-algorithm $\mathbf{A}(\cdot)$ as an outer-loop optimization problem, $\mathbf{A} : \hat{\mathcal{D}} \mapsto \arg\min_{\psi_0} \sum_i \mathcal{L}_{out}(\hat{\mathbb{D}}^i_{te}, \hat{\mathcal{D}}^i_{tr})$, which optimizes the initialized parameters of feature extractor. The output algorithm of the meta-algorithm $A_{\psi_0^*}(\cdot)$, with the optimized $\psi_0^* := \arg\min_{\psi_0} \sum_i \mathcal{L}_{out}(\hat{\mathbb{D}}^i_{te}, \hat{\mathcal{D}}^i_{tr})$, is specified accordingly as the inner-loop optimization problem, $A_{\psi_0^*} : \hat{\mathcal{D}}_{tr} \mapsto \arg\min_{\psi\in\Phi(\psi_0^*)} \mathcal{L}_{in}(\hat{\mathcal{D}}_{tr})$, which optimizes the feature extractor using the initialized parameters ψ_0^*.

Corollary 1. (Geometric Understanding of the Bilevel Optimization) *Given the meta-sample $\hat{\mathcal{D}}$ and the source samples $\hat{\mathcal{S}}$, we consider a pseudo-metric space $\big(\mathcal{M}(\mathfrak{P}_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\mathcal{X})}), \mathrm{disc}_{\mathcal{Y}}(\cdot,\cdot)\big)$,*

Algorithm 1 Meta-training Procedure.

Input data: N_m source-domain training samples, hyperparameters: η, β
Parameters: Feature extractor F_ψ, task classifier T_θ
Output: Meta-trained model parameters $\{\psi^{tr}, \theta^{tr}\}$

while Stopping condition is not met **do**
  Sample minibatch of meta-target sample $\hat{\mathbb{D}}_{te}$ and minibatch of meta-source samples $\hat{\mathcal{D}}_{tr}$
  Evaluate inner-loop objective, $\mathcal{L}_{in} \leftarrow \sum_{\hat{\mathbb{S}}^k, \hat{\mathbb{S}}^t \in \hat{\mathcal{D}}_{tr}} \hat{\text{disc}}_{\mathcal{Y}}(F_\psi(\hat{\mathbb{S}}^k, \hat{\mathbb{S}}^t))$
  Update the feature extractor with gradient descent, $\psi' \leftarrow \psi - \eta \nabla_\psi \mathcal{L}_{in}$
  Evaluate outer-loop objective w.r.t. the updated ψ', $\mathcal{L}_{out} \leftarrow \sum_{\hat{\mathbb{S}}^k \in \hat{\mathcal{D}}_{tr}} \hat{\text{disc}}_{\mathcal{Y}}(F_{\psi'}(\hat{\mathbb{S}}^k, \hat{\mathbb{D}}_{te}))$
  Update the feature extractor w.r.t. the initial parameters ψ, $\psi \leftarrow \psi - \eta\beta \nabla_\psi \mathcal{L}_{out}$
  Evaluate task objective, $\mathcal{L}_{task} \leftarrow \hat{\epsilon}_{\hat{\mathcal{D}}_{tr}}(T_\theta \circ F_\psi)$
  Update the task classifier and feature extractor $[\theta, \psi] \leftarrow [\theta, \psi] - \eta \nabla_{\theta,\psi} \mathcal{L}_{task}$
end while

Algorithm 2 Meta-test Procedure.

Input data: N_m source-domain training samples, hyperparameters: η, β
Output: Feature extractor F_ψ, task classifier T_θ

Initialize with the meta-trained parameters, $\psi \leftarrow \psi^{tr}$, $\theta \leftarrow \theta^{tr}$
while Stopping condition is not met **do**
  Sample minibatch of all source-domain samples $\hat{\mathcal{D}}$
  Evaluate inner-loop objective, $\mathcal{L}_{in} \leftarrow \sum_{\hat{\mathbb{S}}^k, \hat{\mathbb{S}}^t \in \hat{\mathcal{D}}} \hat{\text{disc}}_{\mathcal{Y}}(F_\psi(\hat{\mathbb{S}}^k, \hat{\mathbb{S}}^t))$
  Update the feature extractor $\psi \leftarrow \psi - \eta\beta \nabla_\psi \mathcal{L}_{in}$
  Evaluate task objective, $\mathcal{L}_{task} \leftarrow \hat{\epsilon}_{\hat{\mathcal{D}}_{tr}}(T_\theta \circ F_\psi)$
  Update the task classifier and feature extractor $[\theta, \psi] \leftarrow [\theta, \psi] - \eta \nabla_{\theta,\psi} \mathcal{L}_{task}$
end while

defined as the set of domains $\mathfrak{P}_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\mathcal{X})}$ *in the feature space* $\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\mathcal{X})$ *equipped with a pseudo-metric* $\text{disc}_{\mathcal{Y}}(\cdot,\cdot) : \mathbb{R}^d \times \mathbb{R}^d \to \mathbb{R}_+$. *Where the distance between a target domain* $\mathbb{T}$ *and the convex hull of source domains* $\text{conv}(\mathcal{S})$ *is defined as* $\text{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\mathbb{T}, \text{conv}(\mathcal{S}))\right) = \text{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\overline{\mathbb{T}}^*_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})}, \mathbb{T})\right)$, *where* $\overline{\mathbb{T}}^*_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})} = \overline{\mathbb{T}}^*_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})} = \arg\min_{\overline{\mathbb{T}} \in \text{conv}(\mathcal{S})} \text{disc}_{\mathcal{Y}}(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}}))(\overline{\mathbb{T}}, \mathbb{T}))$ *denotes the nearest point to the target domain in* $\text{conv}(\mathcal{S})$. *Then we have:*

$$\begin{aligned}
&\text{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\mathbb{T}, \text{conv}(\mathcal{S}))\right) = \text{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\overline{\mathbb{T}}^*_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})}, \mathbb{T})\right) \\
&\leq \underbrace{\frac{1}{|\hat{\mathcal{S}}|} \sum_{i:\hat{\mathbb{S}}^i \in \hat{\mathcal{S}}} \hat{\text{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\hat{\mathbb{T}}, \hat{\mathbb{S}}^i)\big)}_{\text{Outer-loop Objective } (\mathcal{L}_{out})} + \underbrace{\frac{2}{|\hat{\mathcal{S}}|} \sum_{i<j}^{|\hat{\mathcal{S}}|} \hat{\text{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\hat{\mathbb{S}}^i, \hat{\mathbb{S}}^j)\big)}_{\text{Inner-loop Objective } (\mathcal{L}_{in})} + \Xi(\varepsilon, \delta, \hat{\mathcal{S}}) \qquad (11)
\end{aligned}$$

where the complexity term according to data size is $\Xi(\xi, \delta, \hat{\mathcal{S}}) \propto \mathcal{O}(1/\sqrt{n})$. *Proof can be found in Appendix B.7.*

As shown in Figure 2, The triangle relationship in the pseudo-metric feature distribution space (in the red dotted line) among the target domain $\mathbb{T}_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})}$, the combination of source domains $\widetilde{\mathcal{S}}_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})}$ and the nearest point to the target domain in the convex hull of source domains $\overline{\mathbb{T}}^*_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})}$ demonstrates that the collaborative effectiveness of inner-loop and outer-loop optimization is to minimize the $\mathcal{Y}$-discrepancy between the target domain and the convex hull of source domains.

5.2 Meta-learning Procedure

Based on the meta-target/meta-source splits in episodic training process, we use first-order approximation for gradient descent to update the meta-parameters in each iteration. The procedures of meta-training and meta-test are briefly shown in Algorithm 1 and Algorithm 2, respectively. More complete procedures with the adversarial training strategy to estimate $\mathcal{Y}$-discrepancy are shown in Appendix C.

Table 2: Accuracy (%) on PACS and DomainNet using pretrained ResNet-50 backbone. † indicates statistical significance compared to other methods with low FPR ($\alpha < 0.1$) by the Mann-Whitney test (McKnight & Najab, 2010).

Method	PACS					DomainNet						
	paintings	*cartoon*	*photo*	*sketch*	**Avg.**	*clipart*	*infograph*	*painting*	*qickdraw*	*real*	*sketch*	**Avg.**
ERM	$84.7_{\pm 0.4}$	$80.8_{\pm 0.6}$	$97.2_{\pm 0.3}$	$79.3_{\pm 1.0}$	85.5	$58.1_{\pm 0.3}$	$18.8_{\pm 0.3}$	$46.7_{\pm 0.3}$	$12.2_{\pm 0.4}$	$59.6_{\pm 0.1}$	$49.8_{\pm 0.4}$	40.9
CORAL	$\mathbf{88.3}_{\pm 0.2}$	$80.0_{\pm 0.5}$	$97.5_{\pm 0.3}$	$78.8_{\pm 1.3}$	86.2	$59.2_{\pm 0.1}$	$19.7_{\pm 0.2}$	$46.6_{\pm 0.3}$	$13.4_{\pm 0.4}$	$59.8_{\pm 0.2}$	$50.1_{\pm 0.6}$	41.5
MLDG	$85.5_{\pm 1.4}$	$80.1_{\pm 1.7}$	$97.4_{\pm 0.3}$	$76.6_{\pm 1.1}$	84.9	$59.1_{\pm 0.2}$	$19.1_{\pm 0.3}$	$45.8_{\pm 0.7}$	$13.4_{\pm 0.3}$	$59.6_{\pm 0.2}$	$50.2_{\pm 0.4}$	41.2
SAGNET	$87.4_{\pm 1.0}$	$80.7_{\pm 0.6}$	$97.1_{\pm 0.1}$	$80.0_{\pm 0.4}$	86.3	$57.7_{\pm 0.3}$	$19.0_{\pm 0.2}$	$45.3_{\pm 0.3}$	$12.7_{\pm 0.5}$	$58.1_{\pm 0.5}$	$48.8_{\pm 0.2}$	40.3
Ours	$88.2_{\pm 0.8}$	$\mathbf{81.7}_{\pm 1.2}$	$\mathbf{97.8}_{\pm 0.3}$	$\mathbf{79.6}_{\pm 0.7}$	**86.8**†	$\mathbf{60.7}_{\pm 0.8}$	$\mathbf{23.3}_{\pm 0.5}$	$\mathbf{51.2}_{\pm 0.2}$	$\mathbf{14.7}_{\pm 0.4}$	$\mathbf{63.6}_{\pm 0.4}$	$\mathbf{51.8}_{\pm 0.8}$	**44.2**†

Meta-training. As shown in Algorithm 1, in each iteration, one task is to update the feature extractor w.r.t. the outer-loop objective computed by the updated parameters w.r.t. the inner-loop optimization (line 3-6). The other task is to update the task classifier and feature extractor w.r.t. the classification objective (line 7-8).

Meta-test. As shown in Algorithm 2, in each iteration, one task is to update the feature extractor w.r.t. the inner-loop objective (line 4-5), the other task is to update the task classifier and feature extractor w.r.t. the classification objective (line 6-7).

6 EXPERIMENTS

6.1 EXPERIMENTAL SETTINGS

Datasets. The dataset **PACS** (Li et al., 2017) includes images of seven categories and four domains, including *photo*, *art paintings*, *cartoon* and *sketches*. We use the recommended training and validation split ratio of $90\%/10\%$ (Li et al., 2017), and use the overall validation set aggregated by the validation sets of each training domain for model selection (Gulrajani & Lopez-Paz, 2020). The dataset **DomainNet** (Peng et al., 2019) consists of 345 categories and six distinct domains, including *sketch*, *real*, *quickdraw*, *painting*, *infograph* and *clipart*. We use the same training/validation split ratio of $70\%/30\%$ as previous work Peng et al. (2019), and use the same way as the PACS benchmark for model selection.

Training details. We build our model and conduct experiments based on DomainBed (Gulrajani & Lopez-Paz, 2020), a testbed for DG. In particular, we use an ImageNet pretrained ResNet-50 (Gulrajani & Lopez-Paz, 2020) as the feature extractor for all experiments on the two benchmarks. We use Adam (Kingma & Ba, 2014) for both inner-loop and outer-loop optimization with five gradient steps in the inner-loop optimization during the meta-training period, and 15 gradient steps during the test period. The choices of number of gradient steps are based on performance on the evaluation sets. For hyperparameter search, each hyperparameter is assigned a default value and tuned via random search (Bergstra & Bengio, 2012) over a range near the default value. All hyperparameters were tuned jointly according to their respective random search distributions with a maximum number of 20 trials. Appendix D lists the details of hyperparameter search.

6.2 RESULTS

Table 2 and 3 show the results on two benchmarks. Each reported result is the average of three independent repetitions of the entire study with different hyperparameters, different network parameter initialization and different dataset splits.

State-of-the-art comparison. We list the results of ERM and other methods that are better than ERM on either PACS and DomainNet in Table 2. Our method outperforms ERM on both two datasets, which is in consistent with the theoretical analysis that discrepancy-optimal meta-learning has the potential to be better than ERM when $\mathcal{Y}$-discrepancy can be promisingly optimized. CORAL (Sun & Saenko, 2016) learn domain-invariant features across source domains via optimizing mean and covariance of distributions. Compared with this method, the strength of our meta-learning method is the to ability to optimize $\mathcal{Y}$-discrepancy between unseen target and source domains, which is

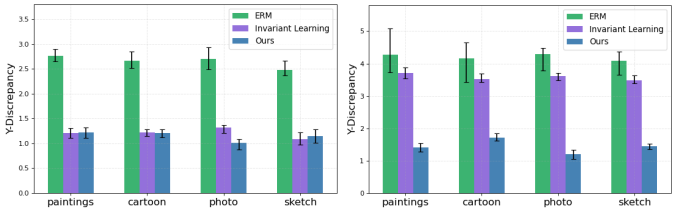
Table 3: Ablation study on PACS. † indicates statistical significance compared to other baselines with low FPR ($\alpha < 0.1$) by the Mann-Whitney test (McKnight & Najab, 2010).

Training Objectives		**PACS**				
Inner Objective	Outer Objective	*paintings*	*cartoon*	*photo*	*sketch*	**Avg.**
DISC	None	$87.1_{\pm 0.7}$	$77.3_{\pm 0.8}$	$96.3_{\pm 0.5}$	$74.2_{\pm 1.3}$	83.7
TASK	TASK	$84.8_{\pm 0.8}$	$80.3_{\pm 1.4}$	$96.7_{\pm 0.4}$	$77.1_{\pm 1.2}$	84.7
TASK	DISC	$86.1_{\pm 0.7}$	$80.7_{\pm 1.5}$	$97.7_{\pm 0.1}$	$77.4_{\pm 0.7}$	85.5
DISC+TASK	TASK	$87.6_{\pm 1.2}$	$81.2_{\pm 1.0}$	$97.1_{\pm 0.2}$	$77.7_{\pm 0.8}$	85.9
DISC	DISC	$\mathbf{88.2}_{\pm 0.8}$	$\mathbf{81.7}_{\pm 1.2}$	$\mathbf{97.8}_{\pm 0.3}$	$\mathbf{79.6}_{\pm 0.7}$	**86.8**†

also in consistent with the theoretical analysis. A state-of-the-art meta-learning method for DG MLDG (Li et al., 2018a) directly uses classification objectives of target and source domains as the inner-loop and outer-loop objectives in bilevel optimization. Our method focus on optimizing the domain discrepancy to reduce domain shift, which has reasoning theoretical guarantees and achieves better results. Besides, our method also outperforms a recent state-of-the-art method SAGNET (Nam et al., 2019), which uses style-agnostic networks to reduce intrinsic style bias of CNN.

Ablation study. As shown in Table 3, we compare with a range of variations of choosing the inner-loop or outer-loop objectives between classification objective and $\mathcal{Y}$-discrepancy. Compared with the first line of results, which is equivalent to domain-invariant learning that only optimizes $\mathcal{Y}$-discrepancy between source-domain samples, all of the other meta-learning methods achieve improvements, which shows the effectiveness of meta-learning with episodic training process for DG. In addition, compared with other meta-learning methods, our method achieves the best results. This shows the potential of optimizing domain discrepancy to reduce domains shift for DG.

Domain discrepancy. As shown in Figure 3, we compare $\mathcal{Y}$-discrepancy (Zhang et al., 2012) between our method and other baselines on PACS. Figure 3 (a) shows that both our method and domain-invariant learning can better reduce $\mathcal{Y}$-discrepancy between source domains compared with ERM. This is because these two methods have a training objective to reduce $\mathcal{Y}$-discrepancy across source-domain samples. In addition, Figure 3 (b) shows that $\mathcal{Y}$-discrepancy between the unseen target and source domains of our method is much lower than both ERM and domain-invariant learning, which shows the effectiveness of our discrepancy-optimal meta-learning to reduce domain shift when faced with unseen target domains.



(a) Y-discepancy between sources. (a) Y-discepancy between target and sources.

Figure 3: $\mathcal{Y}$-discrepancy on PACS.

7 CONCLUSION

This work investigates discrepancy-optimal meta-learning for DG from both theoretical and empirical perspectives. The theoretical analysis shows that training a meta-learner to optimize $\mathcal{Y}$-discrepancy between unseen target and source domains is effective for DG. This guides a meta-learning algorithm with episodic training process via bilevel optimization, where the inner-loop objective is to minimize $\mathcal{Y}$-discrepancy across source-domain samples and the outer-loop objective is to minimize $\mathcal{Y}$-discrepancy between ,eta-target and meta-source samples. Empirically, our method achieves state-of-the-art results on two DG benchmarks.

REFERENCES

Isabela Albuquerque, Joao Monteiro, Mohammad Darvishi, Tiago H Falk, and Ioannis Mitliagkas. Generalizing to unseen domains via distribution matching. *arXiv preprint arXiv:1911.00804*, 2020.

Yogesh Balaji, Swami Sankaranarayanan, and Rama Chellappa. Metareg: Towards domain generalization using meta-regularization. *Advances in Neural Information Processing Systems*, 31: 998–1008, 2018.

Jonathan Baxter. A model of inductive bias learning. *Journal of artificial intelligence research*, 12: 149–198, 2000.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine Learning*, 79(1-2):151–175, 2010.

James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of machine learning research*, 13(2), 2012.

Gilles Blanchard, Gyemin Lee, and Clayton Scott. Generalizing from several related classification tasks to a new unlabeled sample. *Advances in Neural Information Processing Systems*, 24:2178–2186, 2011.

Gilles Blanchard, Aniket Anand Deshmukh, Urun Dogan, Gyemin Lee, and Clayton Scott. Domain generalization by marginal transfer learning. *Journal of Machine Learning Research*, 22(2):1–55, 2021.

Zhun Deng, Frances Ding, Cynthia Dwork, Rachel Hong, Giovanni Parmigiani, Prasad Patil, and Pragya Sur. Representation via representations: Domain generalization via adversarially learned invariant representations. *arXiv preprint arXiv:2006.11478*, 2020.

Aniket Anand Deshmukh, Yunwen Lei, Srinagesh Sharma, Urun Dogan, James W Cutler, and Clayton Scott. A generalization error bound for multi-class domain generalization. *arXiv preprint arXiv:1905.10392*, 2019.

Qi Dou, Daniel C Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. *arXiv preprint arXiv:1910.13580*, 2019.

Yingjun Du, Jun Xu, Huan Xiong, Qiang Qiu, Xiantong Zhen, Cees GM Snoek, and Ling Shao. Learning to learn with variational information bottleneck for domain generalization. In *European Conference on Computer Vision*, pp. 200–216. Springer, 2020.

Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *International Conference on Machine Learning*, pp. 1126–1135. PMLR, 2017.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in Neural Information Processing Systems*, 27:2672–2680, 2014.

Jonathan Gordon, John Bronskill, Matthias Bauer, Sebastian Nowozin, and Richard E Turner. Meta-learning probabilistic inference for prediction. *arXiv preprint arXiv:1805.09921*, 2018.

Erin Grant, Chelsea Finn, Sergey Levine, Trevor Darrell, and Thomas Griffiths. Recasting gradient-based meta-learning as hierarchical bayes. *arXiv preprint arXiv:1801.08930*, 2018.

Arthur Gretton, Karsten M Borgwardt, Malte J Rasch, Bernhard Schölkopf, and Alexander Smola. A kernel two-sample test. *The Journal of Machine Learning Research*, 13(1):723–773, 2012.

Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*, 2020.

Zeyi Huang, Haohan Wang, Eric P Xing, and Dong Huang. Self-challenging improves cross-domain generalization. *arXiv preprint arXiv:2007.02454*, 2, 2020.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5542–5550, 2017.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018a.

Da Li, Jianshu Zhang, Yongxin Yang, Cong Liu, Yi-Zhe Song, and Timothy M Hospedales. Episodic training for domain generalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1446–1455, 2019a.

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

Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision*, pp. 624–639, 2018c.

Yiying Li, Yongxin Yang, Wei Zhou, and Timothy Hospedales. Feature-critic networks for heterogeneous domain generalization. In *International Conference on Machine Learning*, pp. 3915–3924. PMLR, 2019b.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9:2579–2605, 2008.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*, 2009.

Andreas Maurer and Tommi Jaakkola. Algorithmic stability and meta-learning. *Journal of Machine Learning Research*, 6(6), 2005.

Patrick E McKnight and Julius Najab. Mann-whitney u test. *The Corsini encyclopedia of psychology*, pp. 1–1, 2010.

Shahar Mendelson. A few notes on statistical learning theory. In *Advanced lectures on machine learning*, pp. 1–40. Springer, 2003.

Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pp. 10–18, 2013.

Hyeonseob Nam, HyunJae Lee, Jongchan Park, Wonjun Yoon, and Donggeun Yoo. Reducing domain gap via style-agnostic networks. *arXiv preprint arXiv:1910.11645*, 2019.

Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2009.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1406–1415, 2019.

Fengchun Qiao, Long Zhao, and Xi Peng. Learning to learn single domain generalization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12556–12565, 2020.

Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. *arXiv preprint arXiv:1911.08731*, 2019.

Shiv Shankar, Vihari Piratla, Soumen Chakrabarti, Siddhartha Chaudhuri, Preethi Jyothi, and Sunita Sarawagi. Generalizing across domains via cross-gradient training. In *International Conference on Learning Representations*, 2018.

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

Sebastian Thrun and Lorien Pratt. Learning to learn: Introduction and overview. In *Learning to learn*, pp. 3–17. Springer, 1998.

Vladimir N Vapnik. An overview of statistical learning theory. *IEEE transactions on neural networks*, 10(5):988–999, 1999.

Riccardo Volpi, Hongseok Namkoong, Ozan Sener, John C Duchi, Vittorio Murino, and Silvio Savarese. Generalizing to unseen domains via adversarial data augmentation. In *Advances in Neural Information Processing Systems*, pp. 5334–5344, 2018.

Yufei Wang, Haoliang Li, and Alex C Kot. Heterogeneous domain generalization via domain mixup. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3622–3626. IEEE, 2020.

Zehao Xiao, Jiayi Shen, Xiantong Zhen, Ling Shao, and Cees GM Snoek. A bit more bayesian: Domain-invariant learning with uncertainty. *arXiv preprint arXiv:2105.04030*, 2021.

Chao Zhang, Lei Zhang, and Jieping Ye. Generalization bounds for domain adaptation. *Advances in neural information processing systems*, 4:3320, 2012.

Han Zhao, Shanghang Zhang, Guanhang Wu, José MF Moura, Joao P Costeira, and Geoffrey J Gordon. Adversarial multiple source domain adaptation. *Advances in Neural Information Processing Systems*, 31:8559–8570, 2018.

Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. Learning to generate novel domains for domain generalization. In *European Conference on Computer Vision*, pp. 561–578. Springer, 2020.

A TECHNICAL TOOLS

We first present the definition of uniform entropy numbers (Mendelson, 2003; Maurer & Jaakkola, 2005) and PAC-style generalization bound based on uniform entropy numbers (Mendelson, 2003), which are the preliminaries for the following proofs.

Definition 3. (Uniform Entropy Numbers) (Mendelson, 2003; Maurer & Jaakkola, 2005): *Let $\mathcal{F}$ be a function class and $\mathbf{Z}_n = \{z_i\}_{1\leq i\leq n},\ (z_i \in \mathcal{Z})$ be a set of examples with the size of n. For any $\varepsilon > 0$, the covering number of $\mathcal{F}$ at radius ε w.r.t. the metric d is denoted as $\mathcal{N}(\varepsilon, \mathcal{F}, d)$. We set the metric d as the ℓ_1-norm $\ell_1(\mathbf{Z}_n)$ corresponding to $\mathbf{Z}_n$. The uniform entropy number of $\mathcal{F}$ w.r.t. $\ell_1(\mathbf{Z}_n)$ is defined as:*

$$\ln \mathcal{N}_1(\varepsilon, \mathcal{F}, \ell_1) = \sup_{n\in\mathbb{N}_+} \sup_{\mathbf{Z}_n\in\mathcal{Z}^n} \ln \mathcal{N}(\varepsilon, \mathcal{F}, \ell_1(\mathbf{Z}_n))$$

where the ℓ_1-norm $\ell_1(\mathbf{Z}_n)$ w.r.t. two functions f and g is defined as $\frac{1}{n}\sum_{i=1}^n |f(z_i) - g(z_i)|,\ (z_i \in \mathbf{Z}_n)$.

Proposition 1. (Generalization Bound based on Uniform Entropy Numbers) (Mendelson, 2003): *Let $\mathcal{F}$ be a class of functions bounded by $\mathcal{B}$, and $\mathbf{Z}_n \in \mathcal{Z}^{N\times n}, (N \geq 1)$ be a set of examples with the size of n drawn i.i.d. from a combination of N distributions $\mathcal{P} = \{\mathbb{P}_i\}_{1\leq i\leq N}$. For every $\varepsilon > 0$ and any $n \geq \frac{8\mathcal{B}}{\varepsilon^2}$,*

$$\Pr\left\{\sup_{f\in\mathcal{F}}\left|\frac{1}{N}\sum_{i=1}^N\frac{1}{n}\sum_{j=1}^n f(z_{i,j}) - \frac{1}{N}\sum_{i=1}^N \mathbb{E}_{\mathbb{P}^i}[f]\right| > \varepsilon\right\} \leq 8\mathbb{E}\big[\mathcal{N}\big(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1(\mathbf{Z}_n)\big)\big]\exp\left(-\frac{nN\varepsilon^2}{128\mathcal{B}^2}\right)$$

Set the RHS to $delta$ and by the definition of uniform entropy numbers, then for any $\delta > 0$, with probability at least $1-\delta$,

$$\sup_{f\in\mathcal{F}}\left|\frac{1}{N}\sum_{i=1}^N\frac{1}{n}\sum_{j=1}^n f(z_{i,j}) - \frac{1}{N}\sum_{i=1}^N \mathbb{E}_{\mathbb{P}^i}[f]\right| \leq \left(128\mathcal{B}^2\frac{\ln\mathcal{N}_1(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1) - \ln\frac{\delta}{8}}{nN}\right)^{\frac{1}{2}}$$

$$= \mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1) - \ln\frac{\delta}{8}}{nN}\right)^{\frac{1}{2}}\right) \leq \mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1) - \ln\frac{\delta}{8}}{n}\right)^{\frac{1}{2}}\right)$$

Proof.

Step 1: (Symmetrization) Let us denote $\overline{\mathbb{E}}_{\mathcal{P}}[f] := \frac{1}{N}\sum_{i=1}^N \mathbb{E}_{\mathbb{P}_i}[f]$ and $\hat{\mathbb{E}}_{\mathbf{Z}_n}[f] := \frac{1}{N}\sum_{i=1}^N\frac{1}{n}\sum_{j=1}^n f(z_{i,j})$. For any $\varepsilon > 0$, such that $n \geq \frac{8\mathcal{B}^2}{\varepsilon^2}$, Let f_n be the function achieving the supremum w.r.t. the samples $\mathbf{Z}_n$,

$$\mathbb{1}_{|\overline{\mathbb{E}}_{\mathcal{P}}[f]-\hat{\mathbb{E}}_{\mathbf{Z}_n}[f]|>\varepsilon}\mathbb{1}_{|\overline{\mathbb{E}}_{\mathcal{P}}[f]-\hat{\mathbb{E}}_{\mathbf{Z}'_n}[f]|<\varepsilon/2} = \mathbb{1}_{|\overline{\mathbb{E}}_{\mathcal{P}}[f]-\hat{\mathbb{E}}_{\mathbf{Z}_n}[f]|>\varepsilon\wedge|\overline{\mathbb{E}}_{\mathcal{P}}[f]-\hat{\mathbb{E}}_{\mathbf{Z}'_n}[f]|<\varepsilon/2} \leq \mathbb{1}_{|\hat{\mathbb{E}}_{\mathbf{Z}'_n}[f]-\hat{\mathbb{E}}_{\mathbf{Z}_n}[f]|>\varepsilon/2}$$

Taking expectation w.r.t. the second sample gives:

$$\mathbb{1}_{|\overline{\mathbb{E}}_{\mathcal{P}}[f]-\hat{\mathbb{E}}_{\mathbf{Z}_n}[f]|>\varepsilon}\Pr'\left\{|\overline{\mathbb{E}}_{\mathcal{P}}[f] - \hat{\mathbb{E}}_{\mathbf{Z}'_n}[f]| < \varepsilon/2\right\} \leq \Pr'\left\{|\hat{\mathbb{E}}_{\mathbf{Z}'_n}[f] - \hat{\mathbb{E}}_{\mathbf{Z}_n}[f]| > \varepsilon/2\right\}$$

By Chebyshev's inequality, since $\mathbf{Z}'_n$ are n *i.i.d.* samples drawn from the multiple distributions $\mathcal{P} = \{\mathbb{P}_i\}_{1\leq i\leq N}$, we have:

$$\Pr'\left\{|\overline{\mathbb{E}}_{\mathcal{P}}[f] - \hat{\mathbb{E}}_{\mathbf{Z}'_n}[f]| \geq \varepsilon/2\right\} = \Pr\left\{\left|\mathbb{E}_{\mathbf{z}\sim\mathcal{P}}\left[\frac{1}{N}\sum_{i=1}^N[f(\mathbf{z}_i)]\right] - \frac{1}{n}\sum_{j=1}^n\frac{1}{N}\sum_{i=1}^N f(z_{i,j})\right| \geq \varepsilon/2\right\}$$

$$\leq \frac{4}{n\varepsilon^2}\operatorname{Var}\left(\frac{1}{N}\sum_{i=1}^N[f(\mathbf{z}_i)]\right) \leq \frac{4\mathcal{B}^2}{n\varepsilon^2}$$

Then, we have:

$$\mathbb{1}_{|\overline{\mathbb{E}}_{\mathcal{P}}[f]-\hat{\mathbb{E}}_{\mathbf{Z}_n}[f]|>\varepsilon}\left(1 - \frac{4\mathcal{B}^2}{n\varepsilon^2}\right) \leq \Pr'\left\{|\hat{\mathbb{E}}_{\mathbf{Z}'_n}[f] - \hat{\mathbb{E}}_{\mathbf{Z}_n}[f]| > \varepsilon/2\right\}$$

Taking expectation w.r.t. the first sample, if $n \geq \frac{8\mathcal{B}^2}{\varepsilon^2}$, we have:

$$\Pr\left\{\sup_{f\in\mathcal{F}}\left|\overline{\mathbb{E}}_{\mathcal{P}}[f]-\hat{\mathbb{E}}_{\mathbf{Z}_n}[f]\right|>\varepsilon\right\}\leq 2\Pr\left\{\sup_{f\in\mathcal{F}}\left|\hat{\mathbb{E}}_{\mathbf{Z}'_n}[f]-\hat{\mathbb{E}}_{\mathbf{Z}_n}[f]\right|>\frac{\varepsilon}{2}\right\} \tag{12}$$

Step 2:

Let $\varepsilon \in \{-1,1\}^{N\times n}\}$ are independent Rademacher random variables, Then,

$$\begin{aligned}&\Pr\left\{\sup_{f\in\mathcal{F}}\left|\hat{\mathbb{E}}_{\mathbf{Z}'_n}[f]-\hat{\mathbb{E}}_{\mathbf{Z}_n}[f]\right|>\frac{\varepsilon}{2}\right\}=\Pr\Pr_{\boldsymbol{\varepsilon}}\left\{\sup_{f\in\mathcal{F}}\left|\sum_{i,j}\varepsilon_{i,j}\big(f(z'_{i,j})-f(z_{i,j})\big)\right|>\frac{nN\varepsilon}{2}\right\}\\&\leq 2\Pr\Pr_{\boldsymbol{\varepsilon}}\left\{\sup_{f\in\mathcal{F}}\left|\sum_{i,j}\varepsilon_{i,j}f(z_{i,j})\right|>\frac{nN\varepsilon}{4}\right\}\end{aligned} \tag{13}$$

For a realization of $\mathbf{Z}_n$, set $\mathcal{N}(\frac{\varepsilon}{8},\mathcal{F},\ell_1(\mathbf{Z}_n))$ to be an $\frac{\varepsilon}{8}$ cover of $\mathcal{F}$ w.r.t. the $\ell_1(\mathbf{Z}_n)$ norm. Hence, there is some $g\in\mathcal{N}(\frac{\varepsilon}{8},\mathcal{F},\ell_1(\mathbf{Z}_n))$ which satisfies that $\frac{1}{nN}\sum_{i,j}|f(z_{i,j})-g(z_{i,j})|<\frac{\varepsilon}{8}$, by the triangle inequality, union bound and Hoeffding's inequality,

$$\begin{aligned}&\Pr_{\boldsymbol{\varepsilon}}\left\{\sup_{f\in\mathcal{F}}\left|\sum_{i,j}\varepsilon_{i,j}f(z_{i,j})\right|>\frac{nN\varepsilon}{4}\right\}\leq\Pr_{\boldsymbol{\varepsilon}}\left\{\sup_{g\in\mathcal{N}(\frac{\varepsilon}{8},\mathcal{F},\ell_1(\mathbf{Z}_n))}\left|\sum_{i,j}\varepsilon_{i,j}g(z_{i,j})\right|>\frac{nN\varepsilon}{8}\right\}\\&\leq 2\mathcal{N}(\frac{\varepsilon}{8},\mathcal{F},\ell_1(\mathbf{Z}_n))\Pr_{\boldsymbol{\varepsilon}}\left\{\left|\sum_{i,j}\varepsilon_{i,j}g(z_{i,j})\right|>\frac{nN\varepsilon}{8}\right\}\leq 2\mathcal{N}(\frac{\varepsilon}{8},\mathcal{F},\ell_1(\mathbf{Z}_n))\exp\left(-\frac{nN\varepsilon^2}{128\mathcal{B}^2}\right)\end{aligned} \tag{14}$$

Combining Eq.(12), Eq. (13) and Eq. (14), we obtain the first conclusion of Proposition 1.

□

Proposition 2. (Technical Lemma based on Union bound) *Let $\{E_l\}_{1\leq l\leq m}$ be a set of conclusions, which satisfy $p(E_l)\geq 1-\delta_l$, with $\delta_l\geq 0,(i\in[m])$. $\mathbf{E}$ is a conclusion derived using $\{E_l\}_{1\leq l\leq m}$, $\mathbf{E}\leftarrow E_1\wedge E_2\wedge\ldots\wedge E_m$. Then, $p(\mathbf{E})\geq 1-\sum_{l=1}^m\delta_l$.*

Proof. Using Morgan's laws, we have:

$$p(\bigwedge_{l=1}^m E_l)=1-p(\bigvee_{l=1}^m\neg E_l)$$

Using the union bound, we have:

$$p(\bigvee_{l=1}^m\neg E_l)\leq\sum_{l=1}^m p(\neg E_l)=\sum_{l=1}^m(1-p(E_l))$$

Then, we have:

$$p(\mathbf{E})\geq p(\bigwedge_{l=1}^m E_l)\geq 1-\sum_{l=1}^m(1-p(E_l))\geq 1-\sum_{l=1}^m(1-(1-\delta_l))=1-\sum_{l=1}^m\delta_l$$

□

B PROOF OF MAIN RESULTS

B.1 PROOF OF LEMMA 1

proof.

By triangle inequality,

$$
\begin{aligned}
&\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}[\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]]-\frac{1}{M}\sum_{k:\hat{\mathcal{S}}_k\in\hat{\mathcal{D}}}\frac{1}{|\hat{\mathcal{S}}_k|}\sum_{i:\hat{\mathbb{S}}_k^i\in\hat{\mathcal{S}}_k}\hat{\epsilon}_{\hat{\mathbb{S}}_k^i}(A(\hat{\mathcal{S}}_k))\right| \\
\leq&\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}[\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]]-\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\left[\frac{1}{N}\sum_{i:\mathbb{S}^i\in\mathcal{S}}\epsilon_{\mathbb{S}^i}(A(\hat{\mathcal{S}}))\right]\right| \\
&+\sup_{A\in\mathcal{A}}\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\left[\frac{1}{N}\sum_{i:\mathbb{S}^i\in\mathcal{S}}\epsilon_{\mathbb{S}^i}(A(\hat{\mathcal{S}}))\right]-\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\left[\frac{1}{N}\sum_{i:\hat{\mathbb{S}}^i\in\hat{\mathcal{S}}}\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathcal{S}}))\right]\right| \\
&+\sup_{A\in\mathcal{A}}\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\left[\frac{1}{N}\sum_{i:\hat{\mathbb{S}}^i\in\hat{\mathcal{S}}}\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathcal{S}}))\right]-\frac{1}{M}\sum_{k:\mathcal{S}_k\in\hat{\mathcal{D}}}\frac{1}{|\hat{\mathcal{S}}_k|}\sum_{i:\hat{\mathbb{S}}_k^i\in\hat{\mathcal{S}}_k}\hat{\epsilon}_{\hat{\mathbb{S}}_k^i}(A(\hat{\mathcal{S}}_k))\right| \\
&=(\mathbf{I})+(\mathbf{II})+(\mathbf{III})
\end{aligned}
$$

Splitting A as a combination of A^f and A^g and by the definition of $\mathcal{Y}$-discrepancy,

$$
\begin{aligned}
(\mathbf{I})\leq&\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}\left[\sup_{g\in\mathcal{G}}\left|\epsilon_{\mathbb{T}}(g\circ A^f(\hat{\mathcal{S}}))-\epsilon_{\widetilde{\mathcal{S}}}(g\circ A^f(\hat{\mathcal{S}}))\right|\right]\ s.t.\ \widetilde{\mathcal{S}}:=\frac{1}{N}\sum_{i=1}^N\mathbb{S}^i \\
=&\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}\left[\operatorname{disc}_{\mathcal{Y}}\left(A^f(\hat{\mathcal{S}})(\widetilde{\mathcal{S}},\mathbb{T})\right)\right]
\end{aligned}
$$

By Jensen's inequality,

$$
\begin{aligned}
(\mathbf{II})\leq&\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\left[\sup_{A\in\mathcal{A}}\left|\frac{1}{N}\sum_{i:\mathbb{S}^i\in\mathcal{S}}\epsilon_{\mathbb{S}^i}(A(\hat{\mathcal{S}}))-\frac{1}{N}\sum_{i:\hat{\mathbb{S}}^i\in\hat{\mathcal{S}}}\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathcal{S}}))\right|\right] \\
\leq&\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\left[\sup_{h\in\mathcal{H}}\left|\frac{1}{N}\sum_{i:\mathbb{S}^i\in\mathcal{S}}\epsilon_{\mathbb{S}^i}(h)-\frac{1}{N}\sum_{i:\hat{\mathbb{S}}^i\in\hat{\mathcal{S}}}\hat{\epsilon}_{\hat{\mathbb{S}}^i}(h)\right|\right]+\mathcal{B}\delta \\
\overset{\text{(with prob. at least }1-\delta)}{\leq}&\mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\xi}{8},\mathcal{F},\ell_1)-\ln\frac{\delta}{8}}{n}\right)^{\frac{1}{2}}\right)+\mathcal{B}\delta
\end{aligned}
$$

where $\mathcal{F}=\{z=(x,y)\mapsto\ell(h(x),y):h\in\mathcal{H}\}$ represents the loss function class w.r.t. the decision function class $\mathcal{H}$. The inequality (according to the complexity term) is derived by Proposition 1 for any $\xi>0$, any $n\geq\frac{8\mathcal{B}^2}{\xi^2}$ and any $\delta>0$.

By applying the Proposition 1 via regarding the meta-sources data $\{\hat{\mathcal{S}}_i\}_{1\leq i\leq M}$ as a sample, we have for any $\xi'>0$, any $M\geq\frac{8\mathcal{B}^2}{(\xi')^2}$ and any $\delta>0$,

$$
(\mathbf{III})\overset{\text{(with prob. at least }1-\delta)}{\leq}\mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\xi'}{8},\widetilde{\mathcal{F}},\ell_1)-\ln\frac{\delta}{8}}{M}\right)^{\frac{1}{2}}\right)
$$

where $\widetilde{\mathcal{F}}=\{\hat{\mathcal{S}}\mapsto\frac{1}{|\hat{\mathcal{S}}|}\sum_i\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathcal{S}})):A\in\mathcal{A}\}$ represents a class of empirical error functions w.r.t. the algorithm class $\mathcal{A}$.

The final conclusion can be obtained by applying Proposition 2 to the controls of **(I)**, **(II)** and **(III)**. □

B.2 PROOF OF LEMMA 2

proof.

By triangle inequality,

$$
\begin{aligned}
&\left|R_{\mathrm{disc}_{\mathcal{Y}}}-\frac{1}{M}\sum_{i=1}^{M}\hat{\mathrm{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}}_i)(\hat{\mathcal{S}}_i,\hat{\mathbb{T}}_i)\big)\right| \\
=&\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}\big[\,\mathrm{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\widetilde{\mathcal{S}},\mathbb{T})\right)\big]-\frac{1}{M}\sum_{i=1}^{M}\hat{\mathrm{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}}_i)(\hat{\mathcal{S}}_i,\hat{\mathbb{T}}_i)\big)\right| \\
\le&\Big|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}\big[\,\mathrm{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\widetilde{\mathcal{S}},\mathbb{T})\right)\big] \\
&\quad-\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\mathbb{E}_{\hat{\mathbb{T}}\sim\mathbb{T}^n,\mathbb{T}\sim\mathscr{P}}\big[\hat{\mathrm{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\hat{\mathcal{S}},\hat{\mathbb{T}})\big)\big]\Big| \\
&+\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\mathbb{E}_{\hat{\mathbb{T}}\sim\mathbb{T}^n,\mathbb{T}\sim\mathscr{P}}\big[\hat{\mathrm{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\hat{\mathcal{S}},\hat{\mathbb{T}})\big)\big]-\frac{1}{M}\sum_{i=1}^{M}\hat{\mathrm{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}}_i)(\hat{\mathcal{S}}_i,\hat{\mathbb{T}}_i)\big)\right| \\
=&(\mathbf{I})+(\mathbf{II})
\end{aligned}
$$

By triangle inequality and additivity of the supremum function,

$$
\begin{aligned}
(\mathbf{I})\le&\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\mathbb{E}_{\hat{\mathbb{T}}\sim\mathbb{T}^n,\mathbb{T}\sim\mathscr{P}}\big[\big|\,\mathrm{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\widetilde{\mathcal{S}},\mathbb{T})\right)-\hat{\mathrm{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\hat{\mathcal{S}},\hat{\mathbb{T}})\big)\big|\big] \\
\le&\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\mathbb{E}_{\hat{\mathbb{T}}\sim\mathbb{T}^n,\mathbb{T}\sim\mathscr{P}}\left[\sup_{g\in\mathcal{G}}\left|\epsilon_{\widetilde{\mathcal{S}}}\big(g\circ\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})\big)-\epsilon_{\mathbb{T}}\big(g\circ\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})\big)\right.\right. \\
&\qquad\qquad\left.\left.-\hat{\epsilon}_{\hat{\mathcal{S}}}\big(g\circ\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})\big)+\hat{\epsilon}_{\hat{\mathbb{T}}}\big(g\circ\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})\big)\right|\right] \\
\le&\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\left[\sup_{g\in\mathcal{G}}\left|\epsilon_{\widetilde{\mathcal{S}}}\big(g\circ\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})\big)-\hat{\epsilon}_{\hat{\mathcal{S}}}\big(g\circ\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})\big)\right|\right] \\
&+\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\mathbb{E}_{\hat{\mathbb{T}}\sim\mathbb{T}^n,\mathbb{T}\sim\mathscr{P}}\left[\sup_{g\in\mathcal{G}}\left|\epsilon_{\mathbb{T}}\big(g\circ\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})\big)-\hat{\epsilon}_{\hat{\mathbb{T}}}\big(g\circ\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})\big)\right|\right] \\
\le&\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n,\mathcal{S}\sim\mathscr{P}^N,N\sim p(N)}\left[\sup_{h\in\mathcal{H}}\left|\epsilon_{\widetilde{\mathcal{S}}}(h)-\hat{\epsilon}_{\hat{\mathcal{S}}}(h)\right|\right]+\mathbb{E}_{\hat{\mathbb{T}}\sim\mathbb{T}^n,\mathbb{T}\sim\mathscr{P}}\left[\sup_{h\in\mathcal{H}}\left|\epsilon_{\mathbb{T}}(h)-\hat{\epsilon}_{\hat{\mathbb{T}}}(h)\right|\right] \\
\overset{\substack{\text{(with prob.}\\ \text{at least } 1-2\delta)}}{\le}&\mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\varepsilon}{8},\mathcal{F},\ell_1)-\ln\frac{\delta}{8}}{n}\right)^{\frac{1}{2}}\right)+2\mathcal{B}\delta
\end{aligned}
$$

where $\mathcal{F}=\{z=(x,y)\mapsto\ell(h(x),y):h\in\mathcal{H}\}$ represents the loss function class. The last inequality according to the complexity terms is derived by applying Proposition 1 twice, and the conclusion can be obtained for any $\varepsilon>0$, any $n\ge\frac{8\mathcal{B}^2}{\varepsilon^2}$ and any $\delta>0$.

By Proposition 1 via treating a combination of meta-source and meta-target samples, we have for any $\varepsilon'>0$, any $M\ge\frac{8\mathcal{B}^2}{(\varepsilon')^2}$ and any $\delta>0$,

$$
(\mathbf{II})\overset{\substack{\text{(with prob.}\\ \text{at least } 1-\delta)}}{\le}\mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\varepsilon'}{8},\mathcal{R},\ell_1)-\ln\frac{\delta}{8}}{M}\right)^{\frac{1}{2}}\right)
$$

where $\mathcal{R}:=\{\{\hat{\mathcal{S}},\hat{\mathbb{T}}\}\mapsto\hat{\mathrm{disc}}_{\mathcal{Y}}(A(\hat{\mathcal{S}})(\hat{\widetilde{\mathcal{S}}},\hat{\mathbb{T}})):A\in\mathcal{A}\}$ represents a class of functions w.r.t. the empirical $\mathcal{Y}$-discrepancy $\hat{\mathrm{disc}}_{\mathcal{Y}}(A^f(\cdot)(\cdot,\cdot))$ between a combination of source domains and target domain.

The final conclusion can be reached by applying Proposition 2 to the control of **(I)** and **(II)**. □

B.3 Proof of Theorem 2

proof.

By applying Proposition 2 to Lemma 1 and Lemma 2, for any $\varepsilon, \varepsilon', \varepsilon'' > 0$, any $n \geq \frac{8\mathcal{B}^2}{\varepsilon^2}$ and any $M \geq \max\{\frac{8\mathcal{B}^2}{(\varepsilon')^2}, \frac{8\mathcal{B}^2}{(\varepsilon'')^2}\}$, we have for any $\delta > 0$, with probability at least $1 - 5\delta$,

$$
\begin{aligned}
&\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)}[\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]] - \frac{1}{M}\sum_{k:\hat{\mathcal{S}}_k\in\hat{\mathcal{D}}}\frac{1}{|\hat{\mathcal{S}}_k|}\sum_{i:\hat{\mathbb{S}}_k^i\in\hat{\mathcal{S}}_k}\hat{\epsilon}_{\hat{\mathbb{S}}_k^i}(A(\hat{\mathcal{S}}_k))\right| \\
\leq &\frac{1}{M}\sum_{i=1}^{M}\hat{\mathrm{disc}}_{\mathcal{Y}}\big(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}}_i)(\hat{\mathcal{S}}_i, \hat{\mathbb{T}}_i)\big) + 3\mathcal{B}\delta + \mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1) - \ln\frac{\delta}{8}}{n}\right)^{\frac{1}{2}}\right) \\
&+ \mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\varepsilon'}{8}, \widetilde{\mathcal{F}}, \ell_1) - \ln\frac{\delta}{8}}{M}\right)^{\frac{1}{2}}\right) + \mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\varepsilon''}{8}, \mathcal{R}, \ell_1) - \ln\frac{\delta}{8}}{M}\right)^{\frac{1}{2}}\right)
\end{aligned}
$$

□

B.4 ERM-BASED DG BOUND

Theorem 2. (ERM-based DG Bound): *Given any $\varepsilon, \varepsilon' > 0$,for any $n \geq \frac{8\mathcal{B}^2}{\varepsilon^2}$ and any $M_m \geq \frac{8\mathcal{B}^2}{(\varepsilon')^2}$, we have for any $\delta > 0$, with probability at least $1 - 2\delta$,*

$$
\begin{aligned}
&\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)}[\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]] - \frac{1}{N}\sum_{i=1}^{N}\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\{\hat{\mathbb{S}}^i\}_{1\leq i\leq N_m}))\right| \\
\leq &\Xi(\varepsilon, \delta, \hat{\mathcal{S}}) + \Omega(\varepsilon', \delta, \hat{\mathcal{S}}) + \mathcal{B}\delta
\end{aligned}
$$

where the complexity terms according to the data size are $\Xi(\varepsilon, \delta, \hat{\mathcal{S}}) \propto \mathcal{O}(1/\sqrt{n})$ and $\Omega(\varepsilon', \delta, \hat{\mathcal{S}}) \propto \mathcal{O}(1/\sqrt{N_m})$.

Notes. As shown in Theorem 2, the complexity terms of ERM-based DG bound relates to the number of training examples in each domain $\mathcal{O}(1/\sqrt{n})$ and the maximum number of source domains $\mathcal{O}(1/\sqrt{N_m})$. The bound sheds light on some DG methods based on data augmentation (Shankar et al., 2018; Volpi et al., 2018; Qiao et al., 2020; Zhou et al., 2020), which produce more training examples in each domain to reduce $\Xi(\varepsilon, \delta, \hat{\mathcal{S}})$ or produce more source domains to reduce $\Omega(\varepsilon', \delta, \hat{\mathcal{S}})$.
proof.

By triangle inequality, Jensen's inequality and Fubini's theorem,

$$
\begin{aligned}
&\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)}[\mathbb{E}_{\mathbb{T}\sim\mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))]] - \frac{1}{N}\sum_{i=1}^{N}\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\{\hat{\mathbb{S}}^i\}_{1\leq i\leq N_m}))\right| \\
\leq &\sup_{A\in\mathcal{A}}\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)}\big[\mathbb{E}_{\mathbb{P}\sim\mathscr{P}}[\epsilon_{\mathbb{P}}(A(\hat{\mathcal{S}}))] - \mathbb{E}_{\mathbb{P}\sim\mathscr{P}}\mathbb{E}_{\hat{\mathbb{P}}\sim\mathbb{P}^n}[\hat{\epsilon}_{\hat{\mathbb{P}}}(A(\hat{\mathcal{S}}))]\big]\right| \\
&+ \sup_{A\in\mathcal{A}}\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)}\mathbb{E}_{\mathbb{P}\sim\mathscr{P}}\mathbb{E}_{\hat{\mathbb{P}}\sim\mathbb{P}^n}[\hat{\epsilon}_{\hat{\mathbb{P}}}(A(\hat{\mathcal{S}}))] - \sum_{i=1}^{N}\alpha_i\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathbb{S}}^i))\right| \\
\leq &\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)}\mathbb{E}_{\mathbb{P}\sim\mathscr{P}}\mathbb{E}_{\hat{\mathbb{P}}\sim\mathbb{P}^n}\left[\sup_{A\in\mathcal{A}}\left|\epsilon_{\mathbb{P}}(A(\hat{\mathcal{S}})) - \hat{\epsilon}_{\hat{\mathbb{P}}}(A(\hat{\mathcal{S}}))\right|\right] \\
&+ \sup_{A\in\mathcal{A}}\left|\mathbb{E}_{\hat{\mathcal{S}}\sim\mathcal{S}^n, \mathcal{S}\sim\mathscr{P}^N, N\sim p(N)}\mathbb{E}_{\hat{\mathbb{P}}\sim\mathcal{P}^n}[\hat{\epsilon}_{\hat{\mathbb{P}}}(A(\hat{\mathcal{S}}))] - \sum_{i=1}^{N}\alpha_i\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\{\hat{\mathbb{S}}^i\}_{1\leq i\leq N_m}))\right| \; s.t. \mathcal{P} := \mathbb{E}_{\mathscr{P}}[\mathbb{P}] \\
&= (\mathbf{I}) + (\mathbf{II})
\end{aligned}
$$

By applying Proposition 1, for any $\varepsilon > 0$ and any $n \geq \frac{8\mathcal{B}^2}{\varepsilon^2}$, we have for any $\delta > 0$,

$$
\begin{aligned}
(\mathbf{I}) &\leq \mathbb{E}_{\mathbb{P}\sim\mathscr{P}}\mathbb{E}_{\hat{\mathbb{P}}\sim\mathbb{P}^n}\left[\sup_{N\in[\mathbb{N}_m]}\sup_{\hat{\mathcal{S}}\in\mathcal{Z}^{N\times n}}\sup_{A\in\mathcal{A}}\left|\epsilon_{\mathbb{P}}(A(\hat{\mathcal{S}})) - \hat{\epsilon}_{\hat{\mathbb{P}}}(A(\hat{\mathcal{S}}))\right|\right] \\
&\leq \mathbb{E}_{\mathbb{P}\sim\mathscr{P}}\mathbb{E}_{\hat{\mathbb{P}}\sim\mathbb{P}^n}\left[\sup_{h\in\mathcal{H}}\left|\epsilon_{\mathbb{P}}(h) - \hat{\epsilon}_{\hat{\mathbb{P}}}(h)\right|\right] \overset{\substack{\text{(with prob.}\\ \text{at least } 1-\delta)}}{\leq} \mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\varepsilon}{8},\mathcal{F},\ell_1) - \ln\frac{\delta}{8}}{n}\right)^{\frac{1}{2}}\right) + \mathcal{B}\delta
\end{aligned}
$$

where $\mathcal{F} = \{z = (x, y) \mapsto \ell(h(x), y) : h \in \mathcal{H}\}$ represents the loss function class w.r.t. the decision function class $\mathcal{H}$.

By applying Proposition 1 via treating $\{(\hat{\mathcal{S}}_i, \hat{\mathbb{D}}_i)\}_{1\leq i\leq N_m} = \{(\{\hat{\mathbb{S}}^i\}_{1\leq i\leq N_m}, \hat{\mathbb{S}}^i)\}_{1\leq i\leq N_m}$ as the sample, whose element is drawn *i.i.d.* according to a joint distribution ($\mathcal{S}^n$ ($\mathcal{S} \sim \mathscr{P}^N, N \sim p(N)$), $\mathscr{P}$), for any $\varepsilon' > 0$ and any $N_m \geq \frac{8\mathcal{B}^2}{(\varepsilon')^2}$, we have for any $\delta > 0$,

$$
(\mathbf{II}) \overset{\substack{\text{(with prob.}\\ \text{at least } 1-\delta)}}{\leq} \mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\varepsilon'}{8},\widetilde{\mathcal{F}},\ell_1) - \ln\frac{\delta}{8}}{N_m}\right)^{\frac{1}{2}}\right)
$$

where $\widetilde{\mathcal{F}} = \{(\hat{\mathcal{S}}, \hat{\mathbb{D}}) \mapsto \hat{\epsilon}_{\hat{\mathbb{D}}}(A(\hat{\mathcal{S}})) : A \in \mathcal{A}\}$ denotes a function class of empirical error w.r.t. the algorithm class $\mathcal{A}$.

□

B.5 Domain-invariant Learning for DA

Theorem 3. (Domain-invariant Learning for DA) (Zhang et al., 2012) *Given any $\varepsilon > 0$, we have that for any $n \geq \frac{8\mathcal{B}^2}{\varepsilon^2}$ and any $\delta > 0$, with probability at least $1 - \delta$,*

$$
\left|\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}})) - \frac{1}{N_m}\sum_{i=1}^{N_m}\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathcal{S}}))\right| \leq \frac{1}{N_m}\sum_{i=1}^{N_m}\hat{\mathrm{disc}}_{\mathcal{Y}}\big(A(\hat{\mathcal{S}})\big(\hat{\mathbb{S}}^i, \hat{\mathbb{T}}\big)\big) + \Xi(\varepsilon, \delta, \hat{\mathcal{S}})
$$

where the complexity term according to data size is $\Xi(\varepsilon, \delta, \hat{\mathcal{S}}) \propto \mathcal{O}(1/\sqrt{n})$.

proof.

By triangle inequality,

$$
\begin{aligned}
&\left|\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}})) - \frac{1}{N_m}\sum_{i=1}^{N_m}\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathcal{S}}))\right| \\
\leq&\left|\epsilon_{\mathbb{T}}\big(A^g(\hat{\mathcal{S}})\circ A^f(\hat{\mathcal{S}})\big) - \frac{1}{N_m}\sum_{i=1}^{N_m}\epsilon_{\mathbb{S}^i}\big(A^g(\hat{\mathcal{S}})\circ A^f(\hat{\mathcal{S}})\big)\right| + \left|\frac{1}{N_m}\sum_{i=1}^{N_m}\epsilon_{\mathbb{S}^i}(A(\hat{\mathcal{S}})) - \frac{1}{N_m}\sum_{i=1}^{N_m}\hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathcal{S}}))\right| \\
=&(\mathbf{I}) + (\mathbf{II})
\end{aligned}
$$

By Proposition 1, given any $\xi > 0$, for any $n \geq \frac{8\mathcal{B}^2}{\xi^2}$, we have for any $\delta > 0$,

$$
(\mathbf{II}) \leq \sup_{h\in\mathcal{H}}\left|\frac{1}{N_m}\sum_{i=1}^{N_m}\epsilon_{\mathbb{S}^i}(h) - \frac{1}{N_m}\sum_{i=1}^{N_m}\hat{\epsilon}_{\hat{\mathbb{S}}^i}(h)\right| \overset{\substack{\text{(with prob.}\\ \text{at least } 1-\delta)}}{\leq} \mathcal{O}\left(\left(\frac{\ln\mathcal{N}_1(\frac{\xi}{8},\mathcal{F},\ell_1) - \ln\frac{\delta}{8}}{n}\right)^{\frac{1}{2}}\right)
$$

where $\mathcal{F} = \{z = (x, y) \mapsto \ell(h(x), y) : h \in \mathcal{H}\}$ represents the loss function class.

$$
\begin{aligned}
(\mathbf{I}) \leq & \sup_{g \in \mathcal{G}} \left| \epsilon_{\mathbb{T}}\big(g \circ A^f(\hat{\mathcal{S}})\big) - \frac{1}{N_m} \sum_{i=1}^{N_m} \epsilon_{\mathbb{S}^i}\big(g \circ A^f(\hat{\mathcal{S}})\big) \right| = \operatorname{disc}_{\mathcal{Y}}\left(A^f(\hat{\mathcal{S}})(\mathbb{T}, \widetilde{\mathcal{S}})\right) \; s.t. \; \widetilde{\mathcal{S}} = \frac{1}{N_m} \sum_{i \in [N_m]} \mathbb{S}^i \\
\leq & \frac{1}{N_m} \sum_{i \in [N_m]} \sup_{g \in \mathcal{G}} \left| \epsilon_{\mathbb{T}}\big(g \circ A^f(\hat{\mathcal{S}})\big) - \epsilon_{\mathbb{S}^i}\big(g \circ A^f(\hat{\mathcal{S}})\big) \right| = \frac{1}{N_m} \sum_{i=1}^{N_m} \operatorname{disc}_{\mathcal{Y}}(A^f(\hat{\mathcal{S}})\big(\mathbb{T}, \mathbb{S}^i)\big) \\
\overset{(\textbf{Step 1}^*)}{\leq} & \frac{1}{N_m} \sum_{i=1}^{N_m} \hat{\operatorname{disc}}_{\mathcal{Y}}(A(\hat{\mathcal{S}})\big(\hat{\mathbb{S}}^i, \hat{\mathbb{T}}\big)) + \mathcal{O}\left(\left(\frac{\ln \mathcal{N}_1(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1) - \ln \frac{\delta}{8}}{n}\right)^{\frac{1}{2}}\right)
\end{aligned}
$$

***Control of Step 1**

First we use triangle inequality and additivity of the supermum function, next we use Proposition 1, given any ε, for any $n \geq \frac{8\mathcal{B}^2}{\varepsilon^2}$, we have for any $\delta > 0$,

$$
\begin{aligned}
& \operatorname{disc}_{\mathcal{Y}}\left(A^f(\hat{\mathcal{S}})(\mathbb{T}, \mathbb{S}^i)\right) - \hat{\operatorname{disc}}_{\mathcal{Y}}\big(A^f(\hat{\mathcal{S}})(\hat{\mathbb{T}}, \hat{\mathbb{S}}^i)\big) \\
= & \sup_{g \in \mathcal{G}} \left| \epsilon_{\mathbb{T}}(g \circ A^f(\hat{\mathcal{S}})) - \epsilon_{\mathbb{S}^i}(g \circ A^f(\hat{\mathcal{S}})) \right| - \sup_{g \in \mathcal{G}} \left| \hat{\epsilon}_{\hat{\mathbb{T}}}(g \circ A^f(\hat{\mathcal{S}})) - \hat{\epsilon}_{\hat{\mathbb{S}}^i}(g \circ A^f(\hat{\mathcal{S}})) \right| \\
\leq & \sup_{g \in \mathcal{G}} \left| \epsilon_{\mathbb{T}}(g \circ A^f(\hat{\mathcal{S}})) - \epsilon_{\mathbb{S}^i}(g \circ A^f(\hat{\mathcal{S}})) - \hat{\epsilon}_{\hat{\mathbb{T}}}(g \circ A^f(\hat{\mathcal{S}})) + \hat{\epsilon}_{\hat{\mathbb{S}}^i}(g \circ A^f(\hat{\mathcal{S}})) \right| \\
\leq & \sup_{g \in \mathcal{G}} \left| \epsilon_{\mathbb{T}}(g \circ A^f(\hat{\mathcal{S}})) - \hat{\epsilon}_{\hat{\mathbb{T}}}(g \circ A^f(\hat{\mathcal{S}})) \right| + \sup_{g \in \mathcal{G}} \left| \epsilon_{\mathbb{S}^i}(g \circ A^f(\hat{\mathcal{S}})) - \hat{\epsilon}_{\hat{\mathbb{S}}^i}(g \circ A^f(\hat{\mathcal{S}})) \right| \\
\leq & \sup_{h \in \mathcal{H}} \left| \epsilon_{\mathbb{T}}(h) - \hat{\epsilon}_{\hat{\mathbb{T}}}(h) \right| + \sup_{h \in \mathcal{H}} \left| \epsilon_{\mathbb{S}^i}(h) - \hat{\epsilon}_{\hat{\mathbb{S}}^i}(h) \right| \overset{\substack{\text{(with prob.} \\ \text{at least } 1-2\delta)}}{\leq} \mathcal{O}\left(\left(\frac{\ln \mathcal{N}_1(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1) - \ln \frac{\delta}{8}}{n}\right)^{\frac{1}{2}}\right)
\end{aligned}
$$

where $\mathcal{F} = \{z = (x, y) \mapsto \ell(h(x), y) : h \in \mathcal{H}\}$ denotes the loss function class w.r.t. the decision function class $\mathcal{H}$.

Using Proposition 2 to combine the controls of **(I)** and **(II)**, we can obtain the conclusion. □

B.6 Domain-invariant Learning for DG

Theorem 4. (Domain-invariant Learning for DG) (Albuquerque et al., 2020) *Given any $\varepsilon > 0$, we have that for any $n \geq \frac{8\mathcal{B}^2}{\varepsilon^2}$ and any $\delta > 0$, with probability at least $1 - \delta$,*

$$
\left| \mathbb{E}_{\mathbb{T} \sim \mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))] - \frac{1}{N_m} \sum_{i=1}^{N_m} \hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathcal{S}})) \right| \leq \gamma + \frac{2}{N_m} \sum_{i<j}^{N_m} \hat{\operatorname{disc}}_{\mathcal{Y}}\big(A^f(\hat{\mathcal{S}})(\hat{\mathbb{S}}^i, \hat{\mathbb{S}}^j)\big) + \Xi(\varepsilon, \delta, \hat{\mathcal{S}}) \tag{15}
$$

where $\gamma := \mathbb{E}_{\mathbb{T} \sim \mathscr{P}}\big[\operatorname{disc}_{\mathcal{Y}}\left(A^f(\hat{\mathcal{S}})(\overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})}, \mathbb{T})\right)\big], \overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})} = \arg\min_{\overline{\mathbb{T}} \in \operatorname{conv}(\mathcal{S})} \operatorname{disc}_{\mathcal{Y}}(A^f(\hat{\mathcal{S}})(\overline{\mathbb{T}}, \mathbb{T}))$ defines as the shortest $\mathcal{Y}$-discrepancy between target domain and the convex hull of source domains. The complexity term according to data size is $\Xi(\varepsilon, \delta, \hat{\mathcal{S}}) \propto \mathcal{O}(1/\sqrt{n})$.

proof.

By triangle inequality,

$$
\begin{aligned}
& \left| \mathbb{E}_{\mathbb{T} \sim \mathscr{P}}[\epsilon_{\mathbb{T}}(A(\hat{\mathcal{S}}))] - \frac{1}{N_m} \sum_{i=1}^{N_m} \hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathcal{S}})) \right| \\
\leq & \mathbb{E}_{\mathbb{T} \sim \mathscr{P}}\big[\operatorname{disc}_{\mathcal{Y}}\left(A^f(\hat{\mathcal{S}})(\widetilde{\mathcal{S}}, \mathbb{T})\right)\big] + \left| \epsilon_{\widetilde{\mathcal{S}}}(A(\hat{\mathcal{S}})) - \frac{1}{N_m} \sum_{i=1}^{N_m} \hat{\epsilon}_{\hat{\mathbb{S}}^i}(A(\hat{\mathcal{S}})) \right| \\
= & (\mathbf{I}) + (\mathbf{II})
\end{aligned}
$$

By Proposition 1, given any $\varepsilon > 0$, for any $n \geq \frac{8\mathcal{B}^2}{\varepsilon^2}$, we have for any $\delta > 0$,

$$(\mathbf{II}) \leq \sup_{h \in \mathcal{H}} \left| \epsilon_{\widetilde{\mathcal{S}}}(h) - \frac{1}{N_m} \sum_{i=1}^{N_m} \hat{\epsilon}_{\hat{\mathbb{S}}^i}(h) \right| \overset{\text{(with prob. at least } 1-\delta)}{\leq} \mathcal{O}\left(\left(\frac{\ln \mathcal{N}_1(\frac{\xi}{8}, \mathcal{F}, \ell_1) - \ln \frac{\delta}{8}}{n} \right)^{\frac{1}{2}} \right)$$

where $\mathcal{F} = \{z = (x, y) \mapsto \ell(h(x), y) : h \in \mathcal{H}\}$ denotes the loss function class w.r.t. the decision function class $\mathcal{H}$.

By triangle inequality,

$$\begin{aligned}
(\mathbf{I}) \leq & \mathbb{E}_{\mathbb{T} \sim \mathscr{P}} \left[\operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})}, \mathbb{T}) \right) + \operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})}, \widetilde{\mathcal{S}}) \right) \right] \\
\overset{(\textbf{Step 1}^*)}{\leq} & \mathbb{E}_{\mathbb{T} \sim \mathscr{P}} \left[\operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})}, \mathbb{T}) \right) \right] + \frac{2}{N_m} \sum_{i<j}^{N_m} \operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\mathbb{S}^i, \mathbb{S}^j) \right) \\
\overset{(\textbf{Step 2}^*)}{\leq} & \mathbb{E}_{\mathbb{T} \sim \mathscr{P}} \left[\operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})}, \mathbb{T}) \right) \right] + \frac{2}{N_m} \sum_{i<j}^{N_m} \hat{\operatorname{disc}}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\hat{\mathbb{S}}^i, \hat{\mathbb{S}}^j) \right) \\
& + \mathcal{O}\left(\left(\frac{\ln \mathcal{N}_1(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1) - \ln \frac{\delta}{8}}{n} \right)^{\frac{1}{2}} \right)
\end{aligned}$$

***Control of Step 1**

We define $\overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})} = \arg\min_{\overline{\mathbb{T}} \in \operatorname{conv}(\mathcal{S})} \operatorname{disc}_{\mathcal{Y}}(A^f(\hat{\mathcal{S}})(\overline{\mathbb{T}}, \mathbb{T}))$ as the nearest point to $\mathbb{T}$ in the convex hull of sources $\operatorname{conv}(\mathcal{S})$ w.r.t. the $\mathcal{Y}$-discrepancy $\operatorname{disc}_{\mathcal{Y}}(A^f(\hat{\mathcal{S}})(\cdot, \cdot))$. By Jensen's inequality,

$$\begin{aligned}
& \mathbb{E}_{\mathbb{T} \sim \mathscr{P}} \left[\operatorname{disc}_{\mathcal{Y}} \left(A(\hat{\mathcal{S}})(\overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})}, \widetilde{\mathcal{S}}) \right) \right] \\
= & \mathbb{E}_{\mathbb{T} \sim \mathscr{P}} \left[\sup_{g \in \mathcal{G}} \left| \sum_{j=1}^{N_m} \beta_j^{A(\hat{\mathcal{S}})} \epsilon_{\mathbb{S}^j}(g \circ A^f(\hat{\mathcal{S}})) - \frac{1}{N_m} \sum_{i=1}^{N_m} \epsilon_{\mathbb{S}^i}(g \circ A(\hat{\mathcal{S}})) \right| \right] \\
\leq & \mathbb{E}_{\mathscr{P}} \left[\frac{1}{N_m} \sum_{j=1}^{N_m} \sum_{i=1}^{N_m} \beta_j^{A(\hat{\mathcal{S}})} \sup_{g \in \mathcal{G}} \left| \epsilon_{\mathbb{S}^j}(g \circ A^f(\hat{\mathcal{S}})) - \epsilon_{\mathbb{S}^i}(g \circ A(\hat{\mathcal{S}})) \right| \right] \\
= & \mathbb{E}_{\mathscr{P}} \left[\frac{1}{N_m} \sum_{j=1}^{N_m} \sum_{i=1}^{N_m} \beta_j^{A(\hat{\mathcal{S}})} \operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\mathbb{S}^i, \mathbb{S}^j) \right) \right] \leq \frac{1}{N_m} \sum_{i=1}^{N_m} \max_{j \in [N_m]} \operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\mathbb{S}^i, \mathbb{S}^j) \right) \\
\leq & \frac{1}{N_m} \sum_{i=1}^{N_m} \sum_{j \in [N_m]} \operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\mathbb{S}^i, \mathbb{S}^j) \right) = \frac{2}{N_m} \sum_{i<j}^{N_m} \operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\mathbb{S}^i, \mathbb{S}^j) \right)
\end{aligned}$$

where $\boldsymbol{\beta}^{A(\hat{\mathcal{S}})} := \arg\min_{\boldsymbol{\beta}' \in \Lambda^N} \operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\sum_{i=1}^N \beta'_j \mathbb{S}^j, \mathbb{T}) \right)$ is the combination weights of $\overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})}$ w.r.t. the source domains $\{\mathbb{S}^i\}_{1 \leq i \leq N}$, represented as $\overline{\mathbb{T}}^*_{A(\hat{\mathcal{S}})} = \sum_{i=1}^N \beta_i^{A(\hat{\mathcal{S}})} \mathbb{S}^i$.

***Control of Step 2**

Using a similar derivation as the control of step 1 in B.5, given any $\varepsilon > 0$, for any $n > \frac{8\mathcal{B}}{\varepsilon^2}$, we have for any $\delta > 0$,

$$\operatorname{disc}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\mathbb{S}^i, \mathbb{S}^j) \right) - \hat{\operatorname{disc}}_{\mathcal{Y}} \left(A^f(\hat{\mathcal{S}})(\hat{\mathbb{S}}^i, \hat{\mathbb{S}}^j) \right) \overset{\text{(with prob. at least } 1-2\delta)}{\leq} \mathcal{O}\left(\left(\frac{\ln \mathcal{N}_1(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1) - \ln \frac{\delta}{8}}{n} \right)^{\frac{1}{2}} \right)$$

where $\mathcal{F} = \{z = (x, y) \mapsto \ell(h(x), y) : h \in \mathcal{H}\}$ denotes the loss function class w.r.t. the decision function class $\mathcal{H}$.

Using Proposition 2 to combine the controls of **(I)** and **(II)**, we can obtain the conclusion. □

B.7 PROOF OF COROLLARY 1

proof.

By the triangle inequality of $\mathcal{Y}$-discrepancy,

$$\begin{aligned}
&\operatorname{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\mathbb{T}, \operatorname{conv}(\mathcal{S}))\right) = \operatorname{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\overline{\mathbb{T}}^{*}_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})}, \mathbb{T})\right) \\
\leq &\operatorname{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\mathbb{T}, \widetilde{\mathcal{S}})\right) + \operatorname{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\overline{\mathbb{T}}^{*}_{\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})}, \widetilde{\mathcal{S}})\right) \\
= &(\mathbf{I}) + (\mathbf{II})
\end{aligned}$$

Using a similar derivation as the control of **(I)** in B.5, given any $\varepsilon > 0$, for any $n > \frac{8\mathcal{B}}{\varepsilon^2}$, we have for any $\delta > 0$,

$$(\mathbf{I}) \overset{\substack{\text{(with prob.at} \\ \text{least } 1-\delta)}}{\leq} \underbrace{\frac{1}{|\hat{\mathcal{S}}|} \sum_{i:\hat{\mathbb{S}}^i \in \hat{\mathcal{S}}} \hat{\operatorname{disc}}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\hat{\mathbb{T}}, \hat{\mathbb{S}}^i)\right)}_{\text{Outer-loop Objective } (\mathcal{L}_{out})} + \mathcal{O}\left(\left(\frac{\ln \mathcal{N}_1(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1) - \ln \frac{\delta}{8}}{n}\right)^{\frac{1}{2}}\right)$$

First using a similar derivation as the control of Step 1 in B.6, then using a similar derivation as the control of Step 1 in B.5, given any $\varepsilon > 0$, for any $n > \frac{8\mathcal{B}}{\varepsilon^2}$, we have for any $\delta > 0$,

$$\begin{aligned}
(\mathbf{II}) \leq &\frac{1}{|\hat{\mathcal{S}}|} \sum_{i:\mathbb{S}^i \in \mathcal{S}} \sum_{j:\mathbb{S}^j \in \mathcal{S}} \operatorname{disc}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\mathbb{S}^i, \mathbb{S}^j)\right) \\
\overset{\substack{\text{(with prob.at} \\ \text{least } 1-\delta)}}{\leq} &\underbrace{\frac{2}{|\hat{\mathcal{S}}|} \sum_{i<j}^{|\hat{\mathcal{S}}|} \hat{\operatorname{disc}}_{\mathcal{Y}}\left(\mathbf{A}(\hat{\mathcal{D}})(\hat{\mathcal{S}})(\hat{\mathbb{S}}^i, \hat{\mathbb{S}}^j)\right)}_{\text{Inner-loop Objective } (\mathcal{L}_{in})} + \mathcal{O}\left(\left(\frac{\ln \mathcal{N}_1(\frac{\varepsilon}{8}, \mathcal{F}, \ell_1) - \ln \frac{\delta}{8}}{n}\right)^{\frac{1}{2}}\right)
\end{aligned}$$

Using Proposition 2 to combine the controls of **(I)** and **(II)**, we can obtain the final conclusion. □

Algorithm 3 Meta-training Procedure.

Input data: N_m source-domain training samples, hyperparameters: η, β
Parameters: Feature extractor F_ψ, adversarial classifiers $\mathcal{T} = \{T_{\theta_{i,j}}\}_{1 \le i < j \le N_m}$, task classifier T_θ
Output: Meta-trained model parameters $\{\psi^{tr}, \{\theta^{tr}_{i,j}\}_{1 \le i < j \le N_m}, \theta^{tr}\}$

1: **while** Stopping condition is not met **do**
2: Sample minibatch of meta-target sample $\hat{\mathbb{D}}_{te}$ (the l-th domain in N_m) and minibatch of meta-source samples $\hat{\mathcal{D}}_{tr}$
3: **for** $\hat{\mathbb{S}}^k, \hat{\mathbb{S}}^t \in \hat{\mathcal{D}}_{tr}, k < t$ **do**
4: $\mathcal{L}^{k,t}_{in} \leftarrow \left| \hat{\epsilon}_{\hat{\mathbb{S}}^k}(T_{\theta_{k,t}} \circ F_\psi) - \hat{\epsilon}_{\hat{\mathbb{S}}^t}(T_{\theta_{k,t}} \circ F_\psi) \right| \quad T_{\theta_{k,t}} \in \mathcal{T}$ # inner-loop objectives
5: $\theta_{k,t} \leftarrow \theta_{k,t} + \eta \nabla_{\theta_{k,t}} \mathcal{L}^{k,t}_{in}$ # update adv. classifiers
6: **end for**
7: $\psi' \leftarrow \psi - \eta \nabla_\psi \sum_{k,t} \mathcal{L}^{k,t}_{in}$ # inner-loop update adv. classifiers
8: **for** $\mathbb{S}^k \in \hat{\mathcal{D}}_{tr}$ **do**
9: $\mathcal{L}^{k}_{out} \leftarrow \left| \hat{\epsilon}_{\hat{\mathbb{S}}^k}(T_{\theta_{k,l}} \circ F_{\psi'}) - \hat{\epsilon}_{\hat{\mathbb{D}}_{te}}(T_{\theta_{k,l}} \circ F_{\psi'}) \right| \quad T_{\theta_{k,l}} \in \mathcal{T}$ # outer-loop objectives
10: $\theta_{k,l} \leftarrow \theta_{k,l} + \eta \nabla_{\theta_{k,l}} \mathcal{L}^{k}_{out}$ # update adv. classifiers
11: **end for**
12: $\psi \leftarrow \psi - \eta\beta \nabla_\psi \sum_k \mathcal{L}^k_{out}$ # update feature extractor
13: $\mathcal{L}_{task} \leftarrow \hat{\epsilon}_{\hat{\mathcal{D}}_{tr}}(T_\theta \circ F_\psi)$ # task objectives
14: $[\theta, \psi] \leftarrow [\theta, \psi] - \eta \nabla_{\theta,\psi} \mathcal{L}_{task}$ # update task classifier and feature extractor
15: **end while**

Algorithm 4 Meta-test Procedure.

Input data: N_m source-domain training samples, hyperparameters: η, β
Output: Feature extractor F_ψ, task classifier T_θ

1: $\psi \leftarrow \psi^{tr}$, $\{\theta_{i,j}\}_{1 \le i < j \le N_m} \leftarrow \{\theta^{tr}_{i,j}\}_{1 \le i < j \le N_m}$, $\theta \leftarrow \theta^{tr}$ # Initialize with meta-trained parameters
2: **while** Stopping condition is not met **do**
3: Sample minibatch of all source-domain samples $\hat{\mathcal{D}}$
4: **for** $\hat{\mathbb{S}}^k, \hat{\mathbb{S}}^t \in \hat{\mathcal{D}}, 1 \le k < t \le N_m$ **do**
5: $\mathcal{L}^{k,t}_{in} \leftarrow \left| \hat{\epsilon}_{\hat{\mathbb{S}}^k}(T_{\theta_{k,t}} \circ F_\psi) - \hat{\epsilon}_{\hat{\mathbb{S}}^t}(T_{\theta_{k,t}} \circ F_\psi) \right| \quad T_{\theta_{k,t}} \in \mathcal{T}$ # inner-loop objectives
6: $\theta_{k,t} \leftarrow \theta_{k,t} + \eta \nabla_{\theta_{k,t}} \mathcal{L}_{in}$ # update adv. classifiers
7: **end for**
8: $\psi \leftarrow \psi - \eta\beta \nabla_\psi \sum_{k,t} \mathcal{L}^{k,t}_{in}$ # update feature extractor
9: $\mathcal{L}_{task} \leftarrow \hat{\epsilon}_{\hat{\mathcal{D}}}(T_\theta \circ F_\psi)$ # task objectives
10: $[\theta, \psi] \leftarrow [\theta, \psi] - \eta \nabla_{\theta,\psi} \mathcal{L}_{task}$ # update task classifier and feature extractor
11: **end while**

C TRAINING PROCEDURE

We use the first-order approximation for gradient descent to update the meta-parameters, ignoring the second derivative terms (Finn et al., 2017). In particular, the update of meta-parameters (with the outer-loop optimization) is only based on the descents computed w.r.t. the parameters updated after the last step of the inner-loop optimization. Besides, we use an adversarial training strategy (Goodfellow et al., 2014; Ganin et al., 2016) to optimize the inner-loop and outer-loop objectives ($\mathcal{L}_{in}$, $\mathcal{L}_{out}$ in Definition 2) in meta-training procedure. In particular, the adversarial training process is implemented via a gradient reverse component (Ganin et al., 2016). The whole meta-learning procedure is shown in Algorithm 3 & 4 and described as follows.

Meta-training. As shown in Algorithm 3, line $3-7$ show an adversarial training process to optimize the inner-loop objective $\mathcal{L}_{in}$, which can be seen as a two-player minimax game between adversarial classifiers and the feature extractor. Line $8-12$ show an similar way to optimize the outer-loop objective $\mathcal{L}_{out}$ via adversarial training. In addition, Line $13-14$ show the training process of the classification task w.r.t. the task classifier and feature extractor with the source domain labeled data.

Table 4: Hyperparameters, their default values and distributions for random search.

Hyperparameters	Default value	Random distribution
batch size (PACS)	32	$2^{\mathcal{U}(3,5.5)}$
batch size (DomainNet)	32	$2^{\mathcal{U}(3,5)}$
dropout	0	Random Choise from $\{0, 0.1, 0.5\}$
learning rate	$5e^{-5}$	$10^{\mathcal{U}(-5,-3.5)}$
generator learning rate	$5e^{-5}$	$10^{\mathcal{U}(-5,-3.5)}$
classifier learning rate	$5e^{-5}$	$10^{\mathcal{U}(-5,-3.5)}$
weight decay	0	$10^{\mathcal{U}(-6,-2)}$
generator weight decay	0	$10^{\mathcal{U}(-6,-2)}$
discriminator weight decay	0	$10^{\mathcal{U}(-6,-2)}$
adam β_1	0.5	Random Choise from $\{0, 0.5\}$
β	1	$10^{\mathcal{U}(-1,1)}$

$\mathcal{U}(a, b)$ denotes a random variable sampled according to the uniform distribution on $[a, b]$.

Meta-test. As shown in Algorithm 4, the learned feature extractor is further trained on all of the N source domains with the inner-loop objective in line $4 - 8$ and simultaneously, the classification task w.r.t. the task classifier and feature extractor is also trained with the source-domain labeled data in line $9 - 10$.

D HYPERPARAMETERS

We list the default value and random search distribution for each hyperparameter in Table 4.

E FULL RESULTS OF STATE-OF-THE-ART COMPARISON

The full results of state-of-the-art comparisons on PACS and DomainNet are listed in Table 5 and Table 6, respectively.

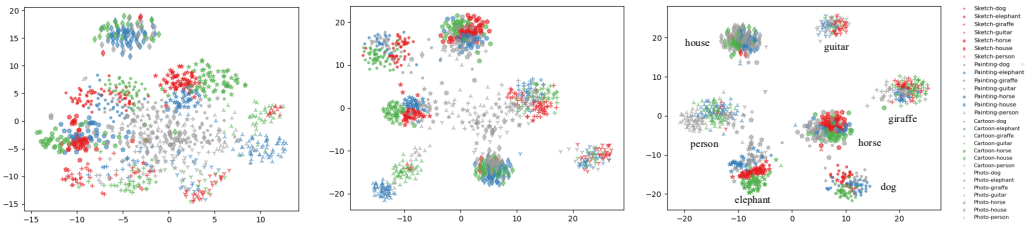
We first make comparisons with ERM methods: ERM-AUG (Vapnik, 1999) which improves ERM via data augmentation, DRO (Sagawa et al., 2019), which increases the importance of domains with larger risks when performing ERM and MIXUP (Wang et al., 2020), which is in consistent with the theoretical analysis that discrepancy-optimal meta-learning has the potential to be better than ERM when $\mathcal{Y}$-discrepancy can be promisingly optimized. CORAL (Sun & Saenko, 2016), MMD (Li et al., 2018b), DANN (Ganin et al., 2016) and C-DANN (Li et al., 2018c) learn domain-invariant features across sources through optimizing mean and covariance of distributions, maximum mean discrepancy (MMD) (Gretton et al., 2012) or $\mathcal{H}$-divergence. Compared with this method, the strength of our meta-learning method is the to ability to optimize $\mathcal{Y}$-discrepancy between unseen target and source domains, which is also in consistent with the theoretical analysis. A state-of-the-art meta-learning DG method MLDG (Li et al., 2018a) directly uses classification objectives of target and source domains as the inner-loop and outer-loop objectives in bilevel optimization. Our method focus on optimizing the domain discrepancy to reduce domain shift, which has reasoning theoretical guarantees and achieves better results. Besides, our method also outperforms recent state-of-the-art methods such as MTL (Blanchard et al., 2021), which uses marginal transfer learning based on kernel-based methods; SAGNET (Nam et al., 2019), which uses style-agnostic networks to reduce intrinsic style bias of CNN; and RSC (Huang et al., 2020), which uses a simple training heuristic algorithm to improve DG.

Table 5: Accuracy (%) on PACS using pretrained ResNet-50 backbone.

Method	*paintings*	*cartoon*	*photo*	*sketch*	**Avg.**
ERM (Vapnik, 1999)	84.7$_{\pm 0.4}$	80.8$_{\pm 0.6}$	97.2$_{\pm 0.3}$	79.3$_{\pm 1.0}$	85.5
DRO (Sagawa et al., 2019)	83.5$_{\pm 0.9}$	79.1$_{\pm 0.6}$	96.7$_{\pm 0.3}$	78.3$_{\pm 2.0}$	84.4
MIXUP (Wang et al., 2020)	86.1$_{\pm 0.5}$	78.9$_{\pm 0.8}$	97.6$\pm$0.1	75.8$_{\pm 1.8}$	84.6
CORAL (Sun & Saenko, 2016)	**88.3**$_{\pm 0.2}$	80.0$_{\pm 0.5}$	97.5$_{\pm 0.3}$	78.8$_{\pm 1.3}$	86.2
MMD (Li et al., 2018b)	86.1$_{\pm 1.4}$	79.4 $_{\pm 0.9}$	96.6$_{\pm 0.2}$	76.5$_{\pm 0.5}$	84.6
DANN (Ganin et al., 2016)	86.4$_{\pm 0.8}$	77.4$_{\pm 0.8}$	97.3$_{\pm 0.4}$	73.5$_{\pm 2.3}$	83.6
C-DANN (Li et al., 2018c)	84.6$_{\pm 1.8}$	75.5$_{\pm 0.9}$	96.8$_{\pm 0.3}$	73.5$_{\pm 0.6}$	82.6
MLDG (Li et al., 2018a)	85.5$_{\pm 1.4}$	80.1$_{\pm 1.7}$	97.4$_{\pm 0.3}$	76.6$_{\pm 1.1}$	84.9
MTL (Blanchard et al., 2021)	87.5$_{\pm 0.8}$	77.1$_{\pm 0.5}$	96.4$_{\pm 0.8}$	77.3$_{\pm 1.8}$	84.6
SAGNET (Nam et al., 2019)	87.4$_{\pm 1.0}$	80.7$_{\pm 0.6}$	97.1$_{\pm 0.1}$	80.0$_{\pm 0.4}$	86.3
RSC (Huang et al., 2020)	85.4$_{\pm 0.8}$	79.7$_{\pm 1.8}$	97.6$_{\pm 0.3}$	78.2$_{\pm 1.2}$	85.2
Ours	88.2$_{\pm 0.8}$	**81.7**$_{\pm 1.2}$	**97.8**$_{\pm 0.3}$	**79.6**$_{\pm 0.7}$	**86.8**

Table 6: Accuracy (%) on the DomainNet using pretrained ResNet-50 backbone.

Method	*clipart*	*infograph*	*painting*	*qickdraw*	*real*	*sketch*	**Avg.**
ERM (Vapnik, 1999)	58.1$_{\pm 0.3}$	18.8$_{\pm 0.3}$	46.7$_{\pm 0.3}$	12.2$_{\pm 0.4}$	59.6$_{\pm 0.1}$	49.8$_{\pm 0.4}$	40.9
DRO (Sagawa et al., 2019)	47.2$_{\pm 0.5}$	17.5$_{\pm 0.4}$	33.8$_{\pm 0.5}$	9.3$_{\pm 0.3}$	51.6$_{\pm 0.4}$	40.1$_{\pm 0.6}$	33.3
MIXUP (Wang et al., 2020)	55.7$_{\pm 0.3}$	18.5$_{\pm 0.5}$	44.3$_{\pm 0.5}$	12.5$_{\pm 0.4}$	55.8$_{\pm 0.3}$	48.2$_{\pm 0.5}$	39.2
CORAL (Sun & Saenko, 2016)	59.2$_{\pm 0.1}$	19.7$_{\pm 0.2}$	46.6$_{\pm 0.3}$	13.4$_{\pm 0.4}$	59.8$_{\pm 0.2}$	50.1$_{\pm 0.6}$	41.5
MMD (Li et al., 2018b)	32.1$_{\pm 13.3}$	11.0$_{\pm 4.6}$	26.8$_{\pm 11.3}$	8.7$_{\pm 2.1}$	32.7$_{\pm 13.8}$	28.9$_{\pm 11.9}$	23.4
DANN (Ganin et al., 2016)	53.1$_{\pm 0.2}$	18.3$_{\pm 0.1}$	44.2$_{\pm 0.7}$	11.8$_{\pm 0.1}$	55.5$_{\pm 0.4}$	46.8$_{\pm 0.6}$	38.3
C-DANN (Li et al., 2018c)	54.6$_{\pm 0.4}$	17.3$_{\pm 0.1}$	43.7$_{\pm 0.9}$	12.1$_{\pm 0.7}$	56.2$_{\pm 0.4}$	45.9$_{\pm 0.5}$	38.3
MLDG (Li et al., 2018a)	59.1$_{\pm 0.2}$	19.1$_{\pm 0.3}$	45.8$_{\pm 0.7}$	13.4$_{\pm 0.3}$	59.6$_{\pm 0.2}$	50.2$_{\pm 0.4}$	41.2
MTL (Blanchard et al., 2021)	57.9$_{\pm 0.5}$	18.5$_{\pm 0.4}$	46.0$_{\pm 0.1}$	12.5$_{\pm 0.1}$	59.5$_{\pm 0.3}$	49.2$_{\pm 0.1}$	40.6
SAGNET (Nam et al., 2019)	57.7$_{\pm 0.3}$	19.0$_{\pm 0.2}$	45.3$_{\pm 0.3}$	12.7$_{\pm 0.5}$	58.1$_{\pm 0.5}$	48.8$_{\pm 0.2}$	40.3
RSC (Huang et al., 2020)	55.0$_{\pm 1.2}$	18.3$_{\pm 0.5}$	44.4$_{\pm 0.6}$	12.2$_{\pm 0.2}$	55.7$_{\pm 0.7}$	47.8$_{\pm 0.9}$	38.9
Ours	**60.7**$_{\pm 0.8}$	**23.3**$_{\pm 0.5}$	**51.2**$_{\pm 0.2}$	**14.7**$_{\pm 0.4}$	**63.6**$_{\pm 0.4}$	**51.8**$_{\pm 0.8}$	**44.2**



(a) ERM (b) Domain-Invariant Learning (c) Ours

Figure 4: t-SNE visualization of feature representions on PACS when the target domain is *photo*. Different image categories are represented by different markers and different domains are represented in different colors where the instances of target domain are represented in gray.

F VISUALIZATION

We visualize the learned feature representations by t-SNE (Maaten & Hinton, 2008), which can be seen as a real-world case for Figure 1. We randomly select 250 test examples from each domain. As shown in Figure 4, compared with ERM, both domain-invariant learning and our method can match the feature distributions of source domains; Compared with the domain-invariant learning, our method can also well match the feature distributions of the target and source domains, which benefits from the outer-loop objective in bilevel optimization to improve the robustness to domain shift.

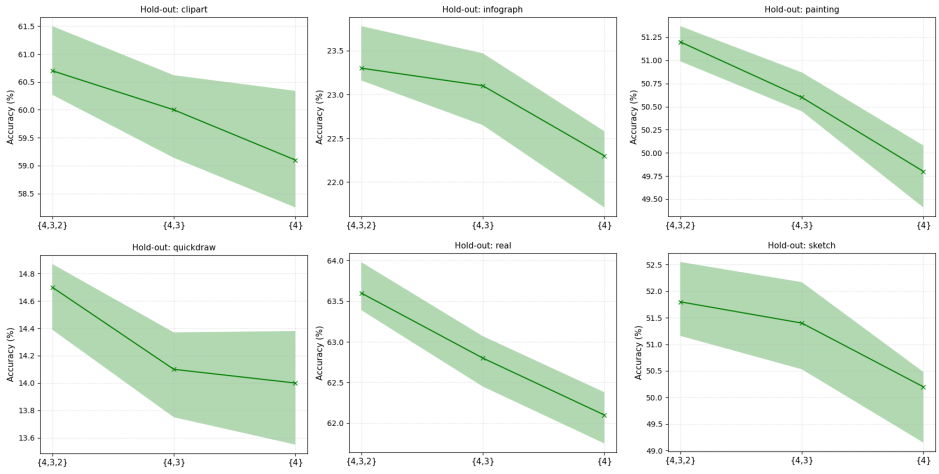


Figure 5: Number of meta-training domains on DomainNet.

G TRADEOFF BETWEEN PERFORMANCE AND COMPLEXITY

We study the possible number of meta-source domains, which has positive correlation with the size of meta-sample in episodic training process. As shown in Figure 5, we compare three sets of possible numbers of meta-source domains: $\{4, 3, 2\}$, $\{4, 3\}$, $\{4\}$, where 4 is the maximum number of meta-source domains on DomainNet. We find that although more possible numbers of meta-source domains can increase the compute complexity, it also improves the performances, because it improves the robustness of algorithm to different numbers of source domains. This is important for DG based on episodic training, because in test phase, the source domains are more than those in meta-training phase.