DNA: Domain Generalization with Diversified Neural Averaging

Xu Chu [*1] **Yujie Jin** [*2] **Wenwu Zhu** [1] **Yasha Wang** [2] **Xin Wang** [1] **Shanghang Zhang** [2] **Hong Mei** [2]

Abstract

The inaccessibility of the target domain data causes domain generalization (DG) methods prone to forget target discriminative features, and challenges the pervasive theme in existing literature in pursuing a single classifier with an ideal joint risk. In contrast, this paper investigates model misspecification and attempts to bridge DG with classifier ensemble theoretically and methodologically. By introducing a pruned Jensen-Shannon (PJS) loss, we show that the target square-root risk w.r.t. the PJS loss of the ρ-ensemble (the averaged classifier weighted by a quasi-posterior ρ) is bounded by the averaged source square-root risk of the Gibbs classifiers. We derive a tighter bound by enforcing a positive principled diversity measure of the classifiers. We give a PAC-Bayes upper bound on the target square-root risk of the ρ-ensemble. Methodologically, we propose a diversified neural averaging (DNA) method for DG, which optimizes the proposed PAC-Bayes bound approximately. The DNA method samples Gibbs classifiers transversely and longitudinally by simultaneously considering the dropout variational family and optimization trajectory. The ρ-ensemble is approximated by averaging the longitudinal weights in a single run with dropout shut down, ensuring a fast ensemble with low computational overhead. Empirically, the proposed DNA method achieves the state-of-the-art classification performance on standard DG benchmark datasets.

1. Introduction

In real-world challenges, the identically distributed condition between the training (*source domain*) data and the testing (*target domain*) data is likely to be violated. For example, the patient population in the medical image processing task (Li et al., 2020), the weather condition in the autonomous driving recognition task (Michaelis et al., 2019), and the visual resolution in multi-modality data analysis (Zhu et al., 2015) could be varied. Traditional machine learning algorithms based on the empirical risk minimization (ERM) (Vapnik, 1999) are error-prone to such distributional shifts, whence the empirical source risk does not necessarily converge to the population target risk. Deep learning algorithms are acknowledged as being especially sensitive to distributional shifts (Su et al., 2019; Recht et al., 2019). Such concerns on deploying machine learning algorithms in critical systems urge increasing attention to domain generalization (DG).

DG aims at generalizing a classifier trained by the source domain(s) data to an unseen target domain in the presence of distributional shift (Blanchard et al., 2011; Muandet et al., 2013). A significant challenge of DG manifests as the inaccessibility of the target data, differing DG from domain adaptation (DA) where comparing covariates of the source domain and the target domain is possible during training (Ben-David et al., 2007). Many principles have been proposed to tackle DG over the last decade (Muandet et al., 2013; Volpi et al., 2018; Piratla et al., 2020). Nevertheless, a predominant theme of existing DG literature is setting the model misspecification (whence the hypothesis space's support may not contain an ideal classifier David et al. (2010)) at negligible, and assuming the support of the deep hypothesis space during training contains an ideal classifier, whose joint risks over the source domain and the target domain are ideal (Muandet et al., 2013; Zhang et al., 2021a). Such an ideal joint risk assumption is challenged by the information bottleneck principle of the deep neural networks (DNNs) (Tishby et al., 2000; Tishby & Zaslavsky, 2015): When training on the source data, a deep classifier tends to remember only discriminative features of seen training data and forget any other information, including those that might be target-discriminative. The inaccessibility of the target data during training implies that the deep classifiers' hypothesis space tends to support on the subspace of low source risk, not necessarily on the subspace of low target risk. In a word, the ideal classifier is possibly depart from the training stage hypothesis space's effective support.

[*]Equal contribution [1]Tsinghua University [2]Peking University. Correspondence to: Wenwu Zhu <wwzhu@tsinghua.edu.cn>, Yasha Wang <wangyasha@pku.edu.cn>, Xin Wang <xin_wang@tsinghua.edu.cn>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

A practical approach to handle model misspecification is classifier ensemble (model/classifier averages) (Masegosa, 2020), which works by enriching the support set of the hypothesis space (Domingos, 1997). Empirical studies have found ensembles outperform thereof Bayesian rivals under distributional shift (Fort et al., 2019). There is literature discussing ensembles for DA (Germain et al., 2013; 2016b). However, there is no adequate research on bridging DG and ensembles from the lens of model misspecification, possibly due to the hiatus of a theoretically convenient loss function.

This paper attempts to bridge DG and classifier ensemble both theoretically and methodologically. We connect a domain's generalization risk for a given classifier with a pruned Jensen-Shannon (PJS) divergence by proposing a novel pruned Jensen-Shannon (PJS) loss function. The risk with respect to (w.r.t.) the PJS loss function has two intriguing properties: Firstly, the square-root risk inherits the satisfaction of triangle inequality from the original JS divergence (Endres & Schindelin, 2003). Secondly, the square-root risk enjoys convexity w.r.t. the classifiers, a property that the original JS divergence does not possess. With the assurance of the convexity and satisfaction of triangle inequality, we show that the target square-root risk of the $\boldsymbol{\rho}$-ensemble (the averaged classifier weighted by a quasi-posterior ρ) is bounded by the averaged source square-root risk of the Gibbs classifiers. We prove that the upper bound on the target square-root risk can be tighter by enforcing a positive diversity measure of classifiers in an ensemble. The diversity measure can be seen as a principled extension of the measure proposed recently (Masegosa, 2020; Ortega et al., 2021) concerning distributional shift. To enable a principled algorithm, we give a PAC-Bayes bound on the square-root target risk in terms of empirical source risks.

Grounded on the proposed PAC-Bayes bound on the square-root target risk, we propose a practical fast-ensemble method for DG with low computation overhead, dubbed the **D**iversified **N**eural **A**veraging (**DNA**). The DNA method samples Gibbs classifiers transversely and longitudinally by simultaneously considering the dropout variational family and the optimization trajectory. For dropout realizations, the proposed DNA encourages a positive diversity measure explicitly. The diversity for trajectory realizations is induced implicitly during the training. In order to reduce the computational overhead, inspired by the recent advance in the fast neural ensemble (Izmailov et al., 2018), the proposed DNA method approximates the $\boldsymbol{\rho}$-ensemble by DNN weight averaging in a single run, without saving checkpoints.

We highlight the contributions of this paper:
1. This paper theoretically and methodologically bridges DG and classifier ensemble from the lens of model misspecification, and this paper firstly discusses a principled diversity measure in the DG scenario.
2. This paper proposes a novel PJS loss, connecting the generalization risk and the PJS divergence. Thence it allows an upper bound on the square-root target risk of the $\boldsymbol{\rho}$-ensemble. This paper also proposes a PAC-Bayes upper bound in terms of empirical source risks.
3. This paper proposes a principled DNA method for DG. The DNA method is a fast-ensemble method with low computational overhead and a competitive method for DG with high performance. The proposed DNA method achieves competitive classification performance on five DG benchmarks (PACS (Li et al., 2017), VLCS (Fang et al., 2013), OfficeHome (Venkateswara et al., 2017), TerraIncognita (Beery et al., 2018) and DomainNet (Peng et al., 2019)) in extensive experiments.

2. Related Work

Over the past decades, there has been a long line of DG literature. Many efforts are devoted to mitigating the domain gap (Muandet et al., 2013) or the risk gap (Arjovsky et al., 2019) in a learned latent space: (a) Directly optimizing statistical distance (Sun & Saenko, 2016) such as Wasserstein distance (Ganin et al., 2016; Li et al., 2018d) and maximum mean discrepancy (MMD) (Li et al., 2018b; Blanchard et al., 2021). (b) Indirectly reducing domain gap by methods such as data augmentation (Wang et al., 2020b) or normalization (Nam & Kim, 2018). c) Adjusting target distribution at test-time, such as meta-learning methods (Li et al., 2018a; 2019; Zhang et al., 2020) and uncertainty minimization methods (Wang et al., 2020a; Iwasawa & Matsuo, 2021). Another line of work tries to learn stable/causal features across domains by learning disentangled features (Mahajan et al., 2021; Zhang et al., 2021b; Li et al., 2021).

A more relevant line of DG methods is increasing the model robustness. The class of input-robust methods learns a robust classifier against input perturbations (Sinha et al., 2018; Volpi et al., 2018; Sagawa et al., 2020; Yi et al., 2021; Krueger et al., 2021) by optimizing the worst-case risk over a set of perturbations on the inputs. On the other hand, SWAD (Cha et al., 2021) and *Transfer* (Zhang et al., 2021a) consider perturbations in the hypothesis (parameter) space. Although SWAD and our DNA employ stochastic weight averaging (SWA) (Izmailov et al., 2018) from the methodological perspective, the motivations differ. SWAD adopts SWA to seek a perturbation-resistant parameter subspace. In contrast, the proposed DNA adopts SWA to approximate the diversified $\boldsymbol{\rho}$-ensemble to reduce computational overhead.

The Bayesian model averaging (BMA) (Xiao et al., 2021) is a close class of methods to the ensemble. BMA reduces the uncertainty to distinguish the best single model with limited data (Hoeting et al., 1999). Classifier ensemble works by enriching the hypothesis space (Domingos, 1997). There are a few works proposing heuristic-based multi-expert meth-

ods (Seo et al., 2020; Zhou et al., 2021; Zhang et al., 2021c), which aim to exploits the complementary information in various source domains. The domains labels are essential for those multi-expert methods. Whereas the domains labels is not necessary for our method. There is a recent work (Thomas et al., 2021) assumes an environment $\mathcal{B}$ and proposes an upper bound for the averaged-case target risk $\mathbb{E}_{\mathcal{Q}\sim\mathcal{B}}\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{Q}}[\ell]$. In contrast, we propose a stronger result, an upper bound for the worst-case target risk, $\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{Q}}[\ell]$ for any $\mathcal{Q}$. A contemporary work (Arpit et al., 2021) proposes an EoA method which also adopts ensemble to tackle DG. We highlight three inherent differences: a) DNA is more theoretically grounded by optimizing the proposed PAC-Bayes target risk bound w.r.t. the PJS loss function. b) DNA promotes a principled measure of diversity and our theoretical results can be viewed as a justification for EoA. c) DNA is a fast-ensemble method that approximates a diversified classifier averaging in a single run, instead of averaging classifiers out of multiple runs with multiplied computational cost. A more detailed discussion on the theoretical connections with existing literature is in Sect. 3.3.

3. Theory

In this section[1], we first formulate the DG problem. Then we introduce the definition and properties of the proposed PJS divergence and PJS loss function, where we set up a link between the generalization risk and the proposed divergence. After that, we introduce theoretical results on the ρ-ensembles, demonstrating the benefits of considering a diversified classifier ensemble for DG. Lastly, we discuss the connections with existing DG theories.

Problem Formulation Denote the input space by $\mathbb{X} \subseteq \mathbb{R}^n$ and the output space by the $C-1$ simplex $\mathbb{Y} = \Delta^{C-1}$. Denote by $M_+(\mathbb{X}\times\mathbb{Y})$ the space of all probability measures on the sample space $\mathbb{X}\times\mathbb{Y}$ (provided with a σ-algebra Σ). The common setting of domain generalization (DG) focuses on the C-class single label classification tasks. The source domain(s)[2] refers to a probability measure $\mathcal{P} \in M_+(\mathbb{X}\times\mathbb{I})$ that is available for sampling, where $\mathbb{I} = \{\mathbf{y} \in \{0,1\}^C : \sum_{j=1}^C y_j = 1\} \subseteq \mathbb{Y}$ is the space of one-hot label vectors. In contrast, the target domain refers to a related measure $\mathcal{Q} \in M_+(\mathbb{X}\times\mathbb{I})$ that is invisible during training. Given a sample $D^n = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$ independently identically distributed (i.i.d.) drawn from $\mathcal{P}^n$, a hypothesis space $\mathbb{H}$ of stochastic classifiers, and a loss function $\ell : \mathbb{X}\times\mathbb{I}\times\mathbb{H} \to \mathbb{R}^+$, the goal of DG is to learn a classifier $\hat{\mathbf{h}} \in \mathbb{H}$ with a minimal generalization risk on the target domain $\mathcal{Q}$,

$$\hat{\mathbf{h}} = \arg\min_{\mathbf{h}\in\mathbb{H}} R^\ell_{\mathcal{Q}}(\mathbf{h}) = \arg\min_{\mathbf{h}\in\mathbb{H}} \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{Q}}\left[\ell(\mathbf{x},\mathbf{y},\mathbf{h})\right]. \quad (1)$$

[1]Please refer to the Appendix A for the proofs of this section.

[2]We consider a single source domain and a single target domain for easiness and fairness concerns (Hashimoto et al., 2018). The extension for multiple source domains is straightforward.

3.1. The PJS Divergence and PJS Loss

Definition 3.1 (pruned Jensen-Shannon divergence)**.** Consider a subset of $M_+(\mathbb{X}\times\mathbb{Y})$ induced by a dominating measure λ, $M_+^\lambda = \{\mu \in M_+(\mathbb{X}\times\mathbb{Y}) : \mu \ll \lambda\}$, such that each measure in M_+^λ is absolutely continuous (w.r.t.) the the dominating measure λ. For measures $\Upsilon, \mathcal{P} \in M_+^\lambda$ and $\mathcal{P} \ll \Upsilon$, let $p = \frac{\mathrm{d}\mathcal{P}}{\mathrm{d}\lambda}$ and $\upsilon = \frac{\mathrm{d}\Upsilon}{\mathrm{d}\lambda}$ denote the Radon–Nikodym derivatives accordingly. The pruned Jensen-Shannon (PJS) divergence $D_{PJS}(\mathcal{P}\|\Upsilon) : M_+^\lambda \times M_+^\lambda \to \mathbb{R}$ from a measure Υ to a measure $\mathcal{P}$ is defined by an integral over the support set $\mathbb{A}_\mathcal{P}$ of measure $\mathcal{P}$, that is

$$D_{PJS}(\mathcal{P}\|\Upsilon) = \int_{\mathbb{A}_\mathcal{P}} p\log\frac{2p}{p+\upsilon} + \upsilon\log\frac{2\upsilon}{p+\upsilon}\mathrm{d}\lambda. \quad (2)$$

The integral over the support set $\mathbb{A}_\mathcal{P}$ of measure $\mathcal{P}$ implies that $D_{PJS}(\mathcal{P}\|\Upsilon)$ only describes the behavior of measures $\mathcal{P}$ and Υ on $\mathbb{A}_\mathcal{P}$, but not the behavior on the different set of the supports $\mathbb{A}_\Upsilon - \mathbb{A}_\mathcal{P}$. Whereas the original Jensen-Shannon (JS) divergence (Endres & Schindelin, 2003) is defined by the intergal of the same function (differs by a factor of $\frac{1}{2}$) over the union of the supports. The subtle difference makes the proposed PJS divergence violate the identity of indiscernibles ($D_{PJS}(\mathcal{P}\|\Upsilon) = 0 \nRightarrow \mathcal{P} = \Upsilon$) so that it is not a statistical divergence. However, as we will see in Thm. 3.2, the PJS divergence inherits the ideal property of satisfying the triangle inequality from the JS divergence. Moreover, the proposed PJS divergence is more compatible with a diversified ensemble learning than the JS divergence.

Theorem 3.2 (properties of PJS divergence)**.** *Consider measures $\mathcal{Q}, \mathcal{P}, \Upsilon \in M_+^\lambda$. Suppose that $\mathcal{Q} \ll \Upsilon$, $\mathcal{P} \ll \Upsilon$ and $\mathbb{A}_\mathcal{P} = \mathbb{A}_\mathcal{Q}$, then the PJS divergence satisfies*
(a) $D_{PJS}(\mathcal{P}\|\mathcal{Q}) = 2D_{JS}(\mathcal{P}\|\mathcal{Q})$, where D_{JS} is the vanilla JS divergence.
(b) $D_{PJS}(\mathcal{P}\|\Upsilon) \geq 0$.
(c) $\sqrt{D_{PJS}(\mathcal{Q}\|\Upsilon)} \leq \sqrt{D_{PJS}(\mathcal{P}\|\Upsilon)} + \sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}$.

Next, we introduce the PJS loss ℓ_{PJS} that connects the generalization risk $R^{\ell_{PJS}}$ and the PJS divergence $D_{PJS}(\cdot\|\cdot)$.

Definition 3.3 (pruned Jensen-Shannon loss)**.** Let $\mathbb{A}$ be the support of the underlying (empirical) distribution where the realization $(\mathbf{x},\mathbf{y})$ is drawn from. The pruned Jensen-Shannon (PJS) loss $\ell_{PJS} : \mathbb{X}\times\mathbb{I}\times\mathbb{H} \to \mathbb{R}^+$ is

$$\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h}) = \left(\log\frac{2}{h^*+1} + h^*\log\frac{2h^*}{h^*+1}\right)\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}\}}, \quad (3)$$

where $h^* \in (0,1)$ is the vector component of $\mathbf{h}$ that corresponds to the correct class, and $\mathbb{1}_{\{\cdot\}}$ is the indicator function.

Provided a domain $\mathcal{P} \in M_+(\mathbb{X}\times\mathbb{I})$, assume the existence of the density function over the input space $\mathbb{X}$, denoted by $p^\mathcal{P}(\mathbf{x})$. Then $p^\mathcal{P}(\mathbf{x})\mathbf{h}$ induces a (set of) measure(s) $\Upsilon^\mathcal{P} \in M_+(\mathbb{X}\times\mathbb{Y})$ whose density over $\mathbb{X}\times\mathbb{Y}$ is $p^\mathcal{P}(\mathbf{x})\mathbf{h}$.

Proposition 3.4. *(connecting risk and PJS divergence) For a classifier $\boldsymbol{h}$ and a domain $\mathcal{P} \in M_+(\mathbb{X} \times \mathbb{I})$. Let $\Upsilon^{\mathcal{P}}$ be the induced measure. Assuming $p^{\mathcal{P}}(\boldsymbol{y}|\boldsymbol{x}) \in \{0,1\}$[3], then*

$$R^{\ell_{PJS}}_{\mathcal{P}}(\boldsymbol{h}) = \mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{P}}[\ell_{PJS}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{h})] = D_{PJS}(\mathcal{P}\|\Upsilon^{\mathcal{P}}) \quad (4)$$

Founding on the connection of the risk and the PJS divergence in the Prop. 3.4, we have bounds of the target risk by invoking the triangle inequality (c) in Thm 3.2.

Theorem 3.5 (bounds of the target risk)**.** *Let $\mathcal{Q}$ and $\mathcal{P}$ be the (fixed) target domain and the source domain in $M_+(\mathbb{X} \times \mathbb{I})$, with $p^{\mathcal{Q}}(\boldsymbol{x})$ and $p^{\mathcal{P}}(\boldsymbol{x})$ being their density functions over the input space $\mathbb{X}$, respectively. Suppose the supports of $\mathcal{Q}$ and $\mathcal{P}$ are identical, i.e., $\mathbb{A}_{\mathcal{P}} = \mathbb{A}_{\mathcal{Q}}$, with $p^{\mathcal{P}}(\boldsymbol{y}|\boldsymbol{x}), p^{\mathcal{Q}}(\boldsymbol{y}|\boldsymbol{x}) \in \{0,1\}$. Let $\mathcal{H}$ be a hypothesis space, such that for any $\boldsymbol{h} \in \mathbb{H}$, the density $p^{\mathcal{P}}(\boldsymbol{x})\boldsymbol{h}$ induced measure $\Upsilon^{\mathcal{P}}$ dominates*[4] *$\mathcal{P}$, and $\Upsilon^{\mathcal{Q}}$ dominates $\mathcal{Q}$, i.e., $\mathcal{P} \ll \Upsilon^{\mathcal{P}}$ and $\mathcal{Q} \ll \Upsilon^{\mathcal{Q}}$. Then for any $\boldsymbol{h} \in \mathbb{H}$,*

$$\begin{cases} \sqrt{R^{\ell_{PJS}}_{\mathcal{Q}}(\boldsymbol{h})} \le \sqrt{R^{\ell_{PJS}}_{\mathcal{P}}(\boldsymbol{h})} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}. & (5) \\ \sqrt{R^{\ell_{PJS}}_{\mathcal{Q}}(\boldsymbol{h})} \ge \sqrt{R^{\ell_{PJS}}_{\mathcal{P}}(\boldsymbol{h})} - 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}. & (6) \end{cases}$$

3.2. The ρ-ensemble

One could arrive at similar bounds within the framework of the original JS divergence. Whereas in this paper we aim at (diversified) ensembles. The pruned JS divergence enables a guaranteed training towards the ensemble quasi-posterior ρ. Formally, we introduce the ρ-ensemble.

Definition 3.6 (ρ-ensemble)**.** Suppose that ρ is a measure on a hypothesis space $\mathbb{H}$. The ρ-ensemble $\boldsymbol{\rho}$ is the ρ-weighted averaged classifier, $\boldsymbol{\rho} = \mathbb{E}_{\mathbf{h}\sim\rho}(\mathbf{h})$.

Directly training the averaged classifier $\boldsymbol{\rho}$ is often impracticable in modern deep architectures. A practical solution is sampling classifiers (Gibbs classifier) from $\mathbb{H}$ according to ρ first, and then training the Gibbs classifiers. However, the sampling-training paradigm relies on a crucial assumption: The generalization risk of the $\boldsymbol{\rho}$ is no larger than the averaged risk of the Gibbs classifiers, i.e., $R(\boldsymbol{\rho}) \le \mathbb{E}_\rho[R(\mathbf{h})]$. The proposed PJS divergence satisfies this requirement.

Theorem 3.7 (inequalities related to the ρ-ensemble)**.** *Let $\mathbb{H}$ be a hypothesis space as stated in the Thm. 3.5. For any $\rho \in M_+(\mathbb{H})$, any measure $\mathcal{P}$, take $\mathbb{D}_{\mathcal{P}} = \mathbb{E}_{\mathcal{P}}[Var_\rho(\sqrt{\ell_{PJS}})]$,*

$$\sqrt{R^{\ell_{PJS}}_{\mathcal{P}}(\boldsymbol{\rho})} \le \sqrt{\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{P}}(\boldsymbol{h})] - \mathbb{D}_{\mathcal{P}}} \le \mathbb{E}_\rho[\sqrt{R^{\ell_{PJS}}_{\mathcal{P}}(\boldsymbol{h})}]. \quad (7)$$

Moreover, $\mathbb{D}_{\mathcal{P}} > 0$ is a necessary condition for the second inequality becoming strict. If $\frac{\ell_{PJS}(\boldsymbol{x}_1,\boldsymbol{y},\boldsymbol{h})}{\ell_{PJS}(\boldsymbol{x}_2,\boldsymbol{y},\boldsymbol{h})}$ is varying on (a non-zero measured subset of) the set $\{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{h}) : Var_\rho(\sqrt{\ell_{PJS}}) > 0\}$ for $\boldsymbol{x}_1 \neq \boldsymbol{x}_2$, then $\mathbb{D}_{\mathcal{P}} > 0$ is a sufficient condition for the second inequality being strict.

The Thm. 3.7 says that in order to obtain a ρ-ensemble with an ideal source risk, we may sample Gibbs classifiers first and optimize the (square-root) risk of each Gibbs classifier. There are at least two options according to the two inequalities in Eq. (7). The first inequality implies a diversity-promoting strategy: Minimizing $\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}}$ prefers a quasi-posterior ρ supported on diversified classifiers to produce a large $\mathbb{D}_{\mathcal{P}} = \mathbb{E}_{\mathcal{P}}[Var_\rho(\sqrt{\ell_{PJS}})]$. The second option is minimizing $\mathbb{E}_\rho[\sqrt{R^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})}]$ without the explicit emphasis on diversity. When $\mathbb{D}_{\mathcal{P}} > 0$ and $\frac{\ell_{PJS}(\mathbf{x}_1,\mathbf{y},\mathbf{h})}{\ell_{PJS}(\mathbf{x}_2,\mathbf{y},\mathbf{h})}$ is non-constant on a non-zero measured set, the former diversity promoting strategy will lead to a strictly tighter upper bound on the ρ-ensemble source risk than the latter one.

Invoking the Thm. 3.5 gives an upper bound on the square-root target risk of the $\boldsymbol{\rho}$-ensemble.

Corollary 3.8 (target risk upper bound of the ensembles)**.** *Given a fixed source domain $\mathcal{P}$ and a target domain $\mathcal{Q}$, for any measure $\rho \in M_+(\mathbb{H})$ on hypothesis space $\mathbb{H}$,*

$$\sqrt{R^{\ell_{PJS}}_{\mathcal{Q}}(\boldsymbol{\rho})} \le \sqrt{\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{P}}(\boldsymbol{h})] - \mathbb{D}_{\mathcal{P}}} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}. \quad (8)$$

Note that the Thm. 3.7 holds for any measure. Therefore we may apply Eq. (7) to the target measure and derive the following corollary on the joint risk.

Corollary 3.9 (joint risk upper bound of the ensembles)**.** *Given a fixed source domain $\mathcal{P}$ and a target domain $\mathcal{Q}$, for any measure $\rho \in M_+(\mathbb{H})$ on the hypothesis space $\mathbb{H}$,*

$$\begin{aligned} &\sqrt{R^{\ell_{PJS}}_{\mathcal{P}}(\boldsymbol{\rho})} + \sqrt{R^{\ell_{PJS}}_{\mathcal{Q}}(\boldsymbol{\rho})} \\ &\le \sqrt{\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{P}}(\boldsymbol{h})] - \mathbb{D}_{\mathcal{P}}} + \sqrt{\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{Q}}(\boldsymbol{h})] - \mathbb{D}_{\mathcal{Q}}} \quad (9) \\ &\le \mathbb{E}_\rho[\sqrt{R^{\ell_{PJS}}_{\mathcal{P}}(\boldsymbol{h})}] + \mathbb{E}_\rho[\sqrt{R^{\ell_{PJS}}_{\mathcal{Q}}(\boldsymbol{h})}]. \end{aligned}$$

Cor. 3.9 says the joint risk of the $\boldsymbol{\rho}$-ensemble is no more significant than that of the Gibbs classifier, validating the motivation of employing classifier ensemble to reduce the undue influence of model misspecification.

We conclude this section by introducing a PAC-Bayes generalization upper bound on the ρ-ensemble target risk in terms of empirical estimates (denoted by $\hat{R}(\cdot)$ or $\hat{\mathbb{D}}(\cdot)$).

Theorem 3.10 (PAC-Bayesian generalization upper bound)**.** *For a fixed source domain $\mathcal{P}$ and a fixed target domain $\mathcal{Q}$, let $\mathbb{H}$ be a hypothesis space as stated in the Thm. 3.5. Suppose that π is a prior over $\mathbb{H}$, which is independent of*

[3] It says a real-world input $\mathbf{x}$ cannot in two categories, combined with $\mathbb{A}_{\mathcal{P}} = \mathbb{A}_{\mathcal{Q}}$ would imply a covariate shift assumption with deterministic conditional $p(\mathbf{y}|\mathbf{x})$. Since invariance is not the focus of this paper, we left the relaxation of invariance for future work.

[4] This is not a strong requirement as long as the component of the correct class in vector $\mathbf{h}$ is non-zero.

draws of source realizations $D^n = \{(\boldsymbol{x}_i, \boldsymbol{y}_i)\}_{i=1}^n \overset{i.i.d.}{\sim} \mathcal{P}^n$. *Then for any* $c > 0$, $\rho \in M_+(\mathbb{H})$, *and any* $\delta \in (0, 1)$, *with probability over* $1 - \delta$

$$\sqrt{R_{\mathcal{Q}}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})} + \sqrt{\mathbb{E}_\rho[\hat{R}_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{h})] - \hat{\mathbb{D}}_{\mathcal{P}} + \frac{2D_{KL}(\rho\|\pi) + \log\frac{1}{\delta} + \Psi_{\mathcal{P},\pi}^{\ell_{PJS}}(c,n)}{cn}}, \tag{10}$$

where $\log \mathbb{E}_{\pi^2}\mathbb{E}_{\mathcal{P}^n}[e^{cn\mathbb{E}_{\mathcal{P}}\sqrt{\ell(\boldsymbol{h}')}\mathbb{E}_{\mathcal{P}}\sqrt{\ell(\boldsymbol{h})} - \hat{\mathbb{E}}_{\mathcal{P}}\sqrt{\ell(\boldsymbol{h}')}\hat{\mathbb{E}}_{\mathcal{P}}\sqrt{\ell(\boldsymbol{h})}}]$ $= \Psi_{\mathcal{P},\pi}^{\ell}(c,n)$ *is constant w.r.t.* ρ *for fixed* c, n, π, ℓ, *and* δ.

3.3. Connection with Existing DG Theory

Bridging DG with classifier ensemble is a complement rather than a substitute for existing DG literature. (a) From the *transferability* perspective: Existing literature assume the optimal Bayes classifier for various domains is in the hypothesis space (Blanchard et al., 2011; Muandet et al., 2013), or the excess risk of the target domain is comparable to those of the source domains in a hypothesis subspace (Zhang et al., 2021a), thence assuring a classifier with an ideal joint risk. We argue that this requirement might be too restrictive for the hypothesis space of deep classifiers without access to the target data, and we may ease the dilemma by enriching the hypothesis with classifier ensembles. (b) From the *invariance* perspective: We assume the identical support of the target domain and the source domain with a deterministic conditional density $p(\mathbf{y}|\mathbf{x})$, which is equivalent to the Dirac delta posterior and the covariate shift assumption in the existing literature (Blanchard et al., 2011; Muandet et al., 2013). (c) From the *discrepancy* perspective: The upper bounds on the target risks usually include a discrepancy measure between the target domain and the source domain. The term discrepancy measure is related to the underlying loss function. For example, our PJS loss leads to the JS divergence. The 0-1 loss usually leads to the total variation divergence (Zhao et al., 2019; Cha et al., 2021). The JS divergence is tighter than the variation divergence (Polyanskiy & Wu, 2014), implying narrow risk gaps. (d) Last but not least: we discuss classifier ensemble in the DG scenario without target access, which is different from the domain adaptation literature discussing classifier ensemble w.r.t. the 0-1 loss (Germain et al., 2013; 2016b).

4. Method

In this section, we propose a Diversified Neural Averaging (DNA) method for DG, whose principle is to optimize the PAC-Bayesian generalization upper bound introduced in Thm. 3.10. We start by quoting the necessary preliminaries from existing deep learning literature before diving into the detailed algorithm.

Dropout variational family (Gal & Ghahramani, 2016). Suppose that $\boldsymbol{\theta} = \{\mathbf{A}_i\}_{i=1}^L \in \Theta$ are NN weight matrices in each layer. Let $\boldsymbol{\epsilon}_i$ be a Bernoulli random vector. The well-known dropout regularization (Srivastava et al., 2014) on activations of i-th layer is equivalent to multiplying $\mathbf{A}_i$ by $\boldsymbol{\epsilon}_i$, resulting in a stochastic weight matrix $\mathbf{W}_i = \boldsymbol{\epsilon}_i\mathbf{A}_i$. The resulting $\boldsymbol{\omega} = \{\mathbf{W}_i\}_{i=1}^L$ can be viewed as draws from a quasi-posterior distribution $\tilde{\rho}_{\boldsymbol{\theta}}$ parametrized by $\boldsymbol{\theta} = \{\mathbf{A}_i\}_{i=1}^L$. The transformation from $\boldsymbol{\epsilon} = \{\boldsymbol{\epsilon}_i\}_{i=1}^L$ to $\boldsymbol{\omega}$, $\boldsymbol{\omega} = r(\boldsymbol{\epsilon}, \boldsymbol{\theta})$, is referred to as the reparameterization trick.

The JS divergence between the source domain and the target domain in this bound is inestimable and often treated as a relatively small constant in DG literature, which is implicitly comprised in the underlying assumption of DG that the target domain is similar to the source domain. Thus, by discarding constant terms of the bound w.r.t. ρ, the minimization problem can be written as,

$$\underset{\rho\in M_+(\mathbb{H})}{minimize}\ \mathbb{E}_\rho[\hat{R}_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})] - \hat{\mathbb{D}}_{\mathcal{P}}(\rho) + \frac{2D_{KL}(\rho\|\pi)}{cn} \tag{11}$$

To optimize this objective, we need to sample Gibbs classifiers from the quasi-posterior ρ. The proposed DNA method samples Gibbs classifiers *transversely* and *longitudinally* by simultaneously considering two aspects. On the one hand, the optimization trajectory of DNNs produces a natural sampling distribution of parameters. On the other hand, we adopt the dropout variational family as a practical choice because it provides sampling and computing convenience.

Formally, we focus on the situation when the hypothesis space $\mathbb{H}$ is the space of parametric functions induced by a DNN. Suppose that $\mathbb{H} = \{h^{\boldsymbol{\omega}}(\cdot)|\boldsymbol{\omega} \in \boldsymbol{\Omega}\}$, where $h(\cdot)$ is the functional form determined by fixed network architecture and $\boldsymbol{\omega}$ parameterizes the function $h^{\boldsymbol{\omega}}(\cdot)$. $\boldsymbol{\Omega}$ is the parameter space that parameters $\boldsymbol{\omega}$ live in. As mentioned above, $\boldsymbol{\omega}$ is reparameterized by the DNN weight matrices parameters $\boldsymbol{\theta}$ and the Bernoulli vector $\boldsymbol{\epsilon}$, where $\boldsymbol{\theta}$ is sampled from an optimization trajectory distribution $\mathcal{T} \in M_+(\Theta)$ and $\boldsymbol{\epsilon}$ is sampled from a Bernoulli random vector B with each dimension being 1 with probability 0.5. For the first term in the Eq. (11), we optimize the empirical risk of each sampled Gibbs classifier,

$$\mathbb{E}_{\rho|\boldsymbol{\theta}}[\hat{R}_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})] = \frac{1}{nm}\sum_{i=1}^n\sum_{\boldsymbol{\epsilon}\sim B}^m \ell_{PJS}(\mathbf{x}_i, \mathbf{y}_i, h^{r(\boldsymbol{\epsilon},\boldsymbol{\theta})}). \tag{12}$$

For the diversity measure $\hat{\mathbb{D}}_{\mathcal{P}}(\rho)$ in the Eq. (11), it is difficult to handle this term among the Gibbs classifiers sampled along the optimization trajectory simultaneously. Therefore, we only explicitly model and promote the diversity measure for each fixed $\boldsymbol{\theta}$ on the trajectory with different dropout realizations,

$$\hat{\mathbb{D}}_{\mathcal{P}}(\rho|\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^n Var_{\boldsymbol{\epsilon}|\boldsymbol{\theta}}(\sqrt{\ell_{PJS}(\mathbf{x}_i, \mathbf{y}_i, h^{r(\boldsymbol{\epsilon},\boldsymbol{\theta})})}), \tag{13}$$

where $Var_{\boldsymbol{\epsilon}|\boldsymbol{\theta}}(\cdot)$ is the variance calculated on the m samples of $\boldsymbol{\epsilon} \sim B$ for a fixed $\boldsymbol{\theta}$. The usage of the KL term $D_{KL}(\rho\|\pi)$ in Eq. (11) is regularizing the quasi-posterior ρ to be not far from the prior π over the hypothesis space. The $D_{KL}(\rho\|\pi)$ is introduced into the upper bound by applying the Donsker and Varadhan's variational formula (cf. the proof for Thm. 3.10 in Appendix A for more details). Assured by the *KL condition* (Gal, 2016; Gal & Ghahramani, 2016), when the prior is approximately Gaussian, $D_{KL}(\rho\|\pi)$ can be approximated by the weight decay for the Bernoulli (and Gaussian) variational family. Moreover, there have been literature (Loshchilov & Hutter, 2018; Maddox et al., 2019) suggesting the approximate Gaussian prior of modern deep learning.

Thus, the optimization objective of the proposed DNA is

$$\underset{\boldsymbol{\theta}\in\Theta}{minimize}\ \mathbb{E}_{\rho|\boldsymbol{\theta}}[\hat{R}_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})] - \eta\hat{\mathbb{D}}_{\mathcal{P}}(\rho|\boldsymbol{\theta}), \qquad (14)$$

where $\eta > 0$ is a hyperparameter controlling the trade-off between minimizing the source risk and promoting the diversity, and the weight decay regularization is used in DNA as a default setting.

In DNA, the target samples are classified by the $\boldsymbol{\rho}$-ensemble. $\boldsymbol{\rho}$-ensemble is the averaged classifier over the quasi-posterior ρ. The classifier averaging over ρ, i.e., $\boldsymbol{\rho} = \mathbb{E}_{\boldsymbol{\theta}\sim\mathcal{T}}\mathbb{E}_{\boldsymbol{\epsilon}\sim B}(h^{r(\boldsymbol{\epsilon},\boldsymbol{\theta})})$, comprises two steps. In the first step, classifier averaging is performed across different dropout realizations for a fixed $\boldsymbol{\theta}$, which is approximated well by shutting down the dropout (Srivastava et al., 2014). In the second step, classifier averaging is performed along the optimization trajectory. In order to reduce computational overhead, we do not save every checkpoint and ensemble these classifiers. Considering the strong connections between the classifiers and the parameter space of DNN (Fort et al., 2019), we approximate classifier averaging by adopting a dense stochastic weight averaging (SWA) (Izmailov et al., 2018; Cha et al., 2021) over the parameter space in a single run, making our DNA a fast-ensemble method. To be specific, we use SWAD (Cha et al., 2021) for weight averaging, which is a modified version of SWA (Izmailov et al., 2018) with a dense and overfit-aware stochastic weight sampling strategy. The entire procedure of DNA method is given in Algorithm 1[5].

4.1. Complexity analysis.

Compared to the baseline (Gulrajani & Lopez-Paz, 2021), the additonal time complexity of DNA incurs from higher evaluation frequency and dropout sampling. We use the same technique as in (Cha et al., 2021) to analyze the overhead: let the total number of source domain samples n, training-validation split ratio r_s, and evaluation frequency f. For one epoch, let t_f be the forward time, t_b be the backward time, r_t be the ratio of forward (backward) time cost of the fully connected layer to that of the convolutional backbone in a DNN. For simplicity, we assume that $t_f = t_b = t$. For one epoch, the training time is $2tnr_s(1 + mr_t)/(r_s + 1)(r_t + 1)$, the evaluation time is $ftn/(r_s + 1)$. Thus, the total time of DNA is $\frac{tn}{r_s+1}[f + 2r_s(1 + mr_t)/(r_t + 1)]$, while the total time of ERM is $\frac{tn}{r_s+1}[f_d + 2r_s]$, where f_d is the default evaluation frequency in DomainBed (Gulrajani & Lopez-Paz, 2021). The final time overhead ratio is $[f + 2r_s(1 + mr_t)/(r_t + 1)]/(f_d + 2r_s)$. Since $f \sim 3f_d$ (in our experimental settings), $f_d < 1$, $r_s = 4$ and $r_t \ll 1$, the ratio indicates a low computational overhead.

Algorithm 1 DNA: DG with Diversified Neural Averaging.

Input: source dataset $D = \{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^n$, dropout model h with weight matrices $\boldsymbol{\theta}$, batch size b, total iterations for training I, trade-off hyperparameter η, number of dropout samples m
Output: learned ρ-ensemble $\boldsymbol{\rho}$
$iter \leftarrow 0$, `initialize` $\overline{\boldsymbol{\theta}}$, $\boldsymbol{\theta}$
while $iter < I$ **do**
  $\mathcal{L} \leftarrow 0$
  $\{(\mathbf{x}_i, \mathbf{y}_i)\}_{i=1}^b \leftarrow$randomly sample b instances from D
  $\{\boldsymbol{\epsilon}_j\}_{j=1}^m \leftarrow$ randomly sample m dropout realizations
  $\mathcal{L} \leftarrow \mathcal{L} + \frac{1}{bm}\sum_{i=1}^b\sum_{j=1}^m \ell_{PJS}(\mathbf{x}_i, \mathbf{y}_i, h^{r(\boldsymbol{\epsilon}_j,\boldsymbol{\theta})})$
  $\mathcal{L} \leftarrow \mathcal{L} - \frac{\eta}{b}\sum_{i=1}^b Var(\sqrt{\ell_{PJS}(\mathbf{x}_i, \mathbf{y}_i, h^{r(\boldsymbol{\epsilon}_j,\boldsymbol{\theta})})})$
  update $\boldsymbol{\theta}$ using $\nabla_{\boldsymbol{\theta}}\mathcal{L}$
  $\overline{\boldsymbol{\theta}} \leftarrow (iter \cdot \overline{\boldsymbol{\theta}} + \boldsymbol{\theta})/(iter + 1)$
  $iter = iter + 1$
end while
$\boldsymbol{\rho} \leftarrow h^{\overline{\boldsymbol{\theta}}}$ with dropout shutdown

5. Experiment

5.1. Experimental Settings

Datasets. Following Gulrajani & Lopez-Paz (2021), we exhaustively conduct experiments on various benchmark datasets to validate the proposed DNA: **PACS** (Li et al., 2017) comprises four domains $d \in$ {photo, art, cartoon, sketch}, containing 9991 images of 7 categories. **VLCS** (Fang et al., 2013) comprises four photographic domains $d \in$ {VOC2007, LabelMe, Caltech101, SUN09}, with 10729 samples of 5 classes. **Office-Home** (Venkateswara et al., 2017) has four domains $d \in$ {art, clipart, product, real}, containing 15500 images with a larger label sets of 65 categories. **TerraIncognita** (Beery et al., 2018) comprises photos of wild animals taken by cameras at different locations. Following (Gulrajani & Lopez-Paz, 2021), we use domains of $d \in$ {L100, L38, L43, L46}, which include 24778 samples and 10 classes. **DomainNet** (Peng et al., 2019) is a large scale dataset containing 596006 images and 345 classes, over six domains $d \in$ {clipart, infograph, painting, quickdraw, real, sketch}.

Evaluation protocol. For a fair comparison, we follow the

[5]Codes are avalable at https://github.com/JinYujie99/DNA

Table 1. **Benchmark Comparisons.** Out-of-domain classification accuracies(%) on PACS, VLCS, OfficeHome, TerraIncognita, and DomainNet are shown. The results of mDSDI and SWAD are from the original literature, and the other numbers are from DomainBed (Gulrajani & Lopez-Paz, 2021). Each experiment is repeated 3 times and we highlight the **best results**.

Method	PACS	VLCS	OfficeHome	TerraIncognita	DomainNet	Avg
ERM (Vapnik, 1999)	85.5 ± 0.2	77.5 ± 0.4	66.5 ± 0.3	46.1 ± 1.8	40.9 ± 0.1	63.3
IRM (Arjovsky et al., 2019)	83.5 ± 0.8	78.5 ± 0.5	64.3 ± 2.2	47.6 ± 0.8	33.9 ± 2.8	61.6
DRO (Sagawa et al., 2020)	84.4 ± 0.8	76.7 ± 0.6	66.0 ± 0.7	43.2 ± 1.1	33.3 ± 0.2	60.7
MMD (Li et al., 2018c)	84.6 ± 0.5	77.5 ± 0.9	66.3 ± 0.1	42.2 ± 1.6	23.4 ± 9.5	58.8
DANN (Ganin et al., 2016)	83.6 ± 0.4	78.6 ± 0.4	65.9 ± 0.6	46.7 ± 0.5	38.3 ± 0.1	62.6
CDANN (Li et al., 2018d)	82.6 ± 0.9	77.5 ± 0.1	65.8 ± 1.3	45.8 ± 1.6	38.3 ± 0.3	62.0
MTL (Blanchard et al., 2021)	84.6 ± 0.5	77.2 ± 0.4	66.4 ± 0.5	45.6 ± 1.2	40.6 ± 0.1	62.9
ARM (Zhang et al., 2020)	85.1 ± 0.4	77.6 ± 0.3	64.8 ± 0.3	45.5 ± 0.3	35.5 ± 0.2	61.7
VREx (Krueger et al., 2021)	84.9 ± 0.6	78.3 ± 0.2	66.4 ± 0.6	46.4 ± 0.6	33.6 ± 2.9	61.9
RSC (Huang et al., 2020)	85.2 ± 0.9	77.1 ± 0.5	65.5 ± 0.9	46.6 ± 1.0	38.9 ± 0.5	62.7
Mixup (Wang et al., 2020b)	84.6 ± 0.6	77.4 ± 0.6	68.1 ± 0.3	47.9 ± 0.8	39.2 ± 0.1	63.4
MLDG (Li et al., 2018a)	84.9 ± 1.0	77.2 ± 0.4	66.8 ± 0.6	47.7 ± 0.9	41.2 ± 0.1	63.6
SagNet (Nam et al., 2021)	86.3 ± 0.2	77.8 ± 0.5	68.1 ± 0.1	48.6 ± 1.0	40.3 ± 0.1	64.2
CORAL (Sun & Saenko, 2016)	86.2 ± 0.3	78.8 ± 0.6	68.7 ± 0.3	47.6 ± 1.0	41.5 ± 0.1	64.5
mDSDI (Bui et al., 2021)	86.2 ± 0.2	79.0 ± 0.3	69.2 ± 0.4	48.1 ± 1.4	42.8 ± 0.1	65.1
SWAD (Cha et al., 2021)	88.1 ± 0.4	**79.1** ± 0.4	70.6 ± 0.3	50.0 ± 0.4	46.5 ± 0.2	66.9
DNA (ours)	**88.4** ± 0.1	79.0 ± 0.1	**71.2** ± 0.1	**52.2** ± 0.4	**47.2** ± 0.1	**67.6**

training and evaluation protocol in DomainBed (Gulrajani & Lopez-Paz, 2021). We choose a domain as the target domain for testing and use the remaining domains as source domains for training. We split each source domain into 8:2 training/validation splits and integrate the validation subsets of each source domain to create an overall validation set, which is used for validation and model selection. The chosen model is tested on the unseen target domain, and we report the mean and standard deviation of out-of-domain classification accuracies from three different runs with different training-validation splits.

Implementation details. We use ResNet-50 (He et al., 2016) pre-trained on ImageNet (Deng et al., 2009) as the backbone network for all datasets. All the batch normalization (BN) layers are frozen during training. We replace the last FC layer of the ResNet-50 with a 2-layer classifier with 1024 hidden units and employ the dropout regularization on the 1024-dimensional output (2048 hidden units are used for DomainNet, adjusting for the larger label space). The network is trained using the Adam (Kingma & Ba, 2015) optimizer. The number of dropout samples m is set to 5. For weight averaging, we use the densely and overfit-aware sampling strategy in (Cha et al., 2021). We follow the hyperparameter search protocol in DomainBed (Gulrajani & Lopez-Paz, 2021) and follow Cha et al. (2021) to use a reduced search space for computational efficiency. We search the trade-off hyperparameter $\eta \in \{0.01, 0.1, 1.0\}$ and set $\eta = 0.1$ by model selection on the validation sets[6].

[6]More implementation details are given in Appendix B.2.

5.2. Results

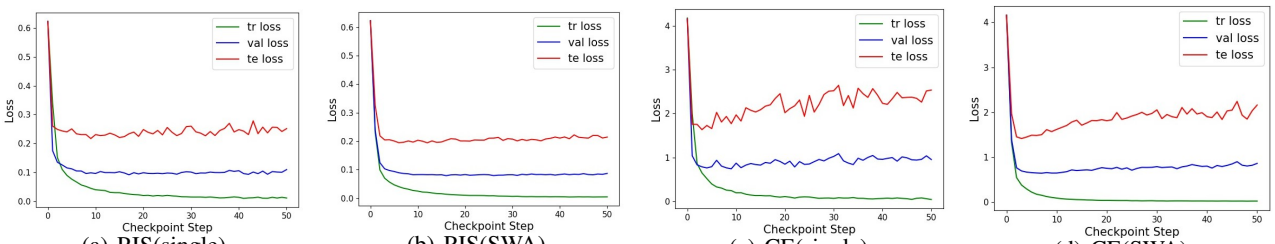
We compare our proposed DNA with various domain generalization methods under the same evaluation protocol and report the out-of-domain classification accuracies[7] in Tab. 1. For all baseline methods, the ResNet-50 is used as the backbone network. It is shown that our DNA method achieves state-of-the-art performance on four of the five benchmarks, including PACS, OfficeHome, TerraIncognita, and DomainNet. While on VLCS, the performance of our method (79.0%) is on par with the previous best results (79.1%). Thus our DNA method achieves a significantly better average accuracy of 67.6% against the baselines. Moreover, the smaller standard errors of DNA manifest that our method is relatively more stable and robust.

5.3. Ablation Studies

To verify the effectiveness of the design of our DNA method, we do ablation studies with regard to the *weight averaging*, the *diversity enforcement* $\mathbb{D}_{\mathcal{P}}$, the *dropout variational family* and the *PJS loss*. In all experiments, the DNN architecture of different DNA variants is kept consistent.

Effects of weight averaging. To inspect the effect of weight averaging, we use only the PJS loss (no dropout or any diversity enforcement term $\mathbb{D}_{\mathcal{P}}$ in the optimization objective) to train the DNN, with (w/) and without (w/o) SWA on OfficeHome, TerraIncognita and DomainNet. The results are

[7]Baseline descriptions and full results per dataset per domain are detailed in Appendix B.1 and Appendix C.



(a) PJS(single) (b) PJS(SWA) (c) CE(single) (d) CE(SWA)

Figure 1. **Losses visualization.** "single" and "SWA" indicate the single model and the SWA (from the beginning of the optimization trajectory) ensemble model. "tr loss", "val loss" and "te loss" denote the loss on the training set, validation set, and test set, respectively. Settings: OfficeHome dataset, the target domain is art while source domains include clipart, product, and real.

Table 2. Out-of-domain accuracies(%) of the PJS loss with (w/) or without (w/o) SWA. Each experiment is repeated 3 times.

	OfficeHome	**TerraIncognita**	**DomainNet**
w/ SWA	70.5 ± 0.1	51.4 ± 0.4	46.5 ± 0.1
w/o SWA	66.5 ± 0.2	49.4 ± 0.3	43.7 ± 0.1

Table 3. Out-of-domain accuracies(%) with (w/) or without (w/o) $\mathbb{D}_{\mathcal{P}}$ for optimization. Each experiment is repeated 3 times.

	PACS	**VLCS**	**DomainNet**
w/ $\mathbb{D}_{\mathcal{P}}$	88.2 ± 0.2	78.7 ± 0.1	46.8 ± 0.1
w/o $\mathbb{D}_{\mathcal{P}}$	88.4 ± 0.1	79.0 ± 0.1	47.2 ± 0.1

shown in Tab. 2, instantiating the advantage of ensembles and result of Thm. 7[8].

To validate the reasonableness of approximating the ρ-ensemble by DNN weight averaging in a single run, we do an ablation study to compare the high-cost classifier averaging with the low-cost weight averaging. The results are shown in Tab. 4. Specifically, we save checkpoints at every evaluation steps in every single run. Then we uniformly sample these models and ensemble them for prediction. The results show that the performance of DNA can be approached by classifier averaging as the number of models for ensemble increases. While DNA is a fast-ensemble method without saving checkpoints, it greatly reduces the computational overhead.

Effects of explicitly enforcing diversity and the choice of variational family. To validate the efficacy of explicitly enforcing diversity of classifiers for ensemble, i.e., encouraging a positive $\mathbb{D}_{\mathcal{P}}$ in the optimization objective, we compare the (w/) and without (w/o) $\mathbb{D}_{\mathcal{P}}$ variants of DNA on all the five benchmarks. The results are shown in Tab. 3 (PACS, VLCS, and DomainNet) and Tab. 5 (OfficeHome and TerraIncognita according to the 1st column). The performance of explicitly enforcing diversity of classifiers consistently outperforms, which substantiates the first and the second inequalities in Eq.(7) of Thm. 3.7.

We also compare DNA with variants regarding the Gaussian dropout variational family. The Bernoulli (the default configuration for DNA) and Gaussian variational families are different choices for optimizing the PAC-Bayes bound approximately. The Bernoulli family is more appropriate for multi-modality distributions. The results of two variational family are reported in Tab. 5 (according to the 2nd column). We can find that the performances of DNA with the Bernoulli dropout variational family are better than that of the Gaussian variational family. We interpret the superiority as a result of better multi-modality posterior tolerance.

Effects of the PJS loss. We compare the models trained with the proposed pruned Jensen-Shannon (PJS) loss function with the models trained with cross-entropy (CE) loss function. The results of optimizing the CE loss and PJS loss with SWA and with explicitly enforcing diversity $\mathbb{D}_{\mathcal{P}}$ are reported in Tab. 6. Note that the diversity measure proposed recently in Masegosa (2020) is used to adapt the CE loss for the CE scenario. From the results, we can find that each PJS model outperforms its CE variant, indicating the effectiveness of the proposed PJS loss and the diversity measure in the presence of distributional shift. The similar accuracies of PJS and CE on OfficeHome (15,500 images of 65 classes) are more of the consequence of the dataset limitation than the limitation of the method. When the dataset is easy or small, the DNN tends to give a high-confidence output, i.e., a $h^* = prob(y_{correct}|\mathbf{x})$ close to 1. From the formulation of the PJS loss (cf. Eq. (3)) and the CE loss, the PJS loss and CE loss will behave similarly when h^* is close to 1. On the other hand, when the dataset is hard or large, two losses behave dissimilarly, and the target risk optimization w.r.t. the PJS loss is theoretically guaranteed by our theory, leading to better accuracy. Additionally, comparing the *with*

[8]Tab. 2 rediscovers the advantage of applying SWA for DG (Cha et al., 2021). In contrast to SWAD (Cha et al., 2021), the proposed DNA is motivated from the lens of fast classifier ensemble, and the results are developed by optimizing the PJS loss.

Table 4. Out-of-domain accuracies(%) of ensemble of different number of models are shown. "K" denotes the number of sampled checkpoints for averaging. Each experiment is repeated 3 times.

	OfficeHome					**TerraIncognita**				
	A	C	P	R	Avg	L100	L38	L43	L46	Avg
K=5	64.6 ± 0.3	56.1 ± 0.4	77.8 ± 0.2	79.5 ± 0.3	69.5	53.0 ± 1.4	42.0 ± 0.2	58.2 ± 0.5	42.6 ± 1.6	49.0
K=10	65.9 ± 0.3	56.6 ± 0.1	78.5 ± 0.2	80.0 ± 0.1	70.3	55.9 ± 1.3	43.9 ± 1.2	60.0 ± 0.4	42.8 ± 1.0	50.6
K=20	66.4 ± 0.1	57.0 ± 0.2	78.8 ± 0.1	**80.5** ± 0.1	70.7	**57.1** ± 1.2	43.7 ± 0.9	60.0 ± 0.4	43.1 ± 1.0	51.0
DNA	**67.7** ± 0.2	**57.7** ± 0.3	**78.9** ± 0.2	**80.5** ± **0.2**	**71.2**	56.8 ± 1.2	**47.0** ± 0.9	**61.0** ± 0.5	**44.0** ± 1.0	**52.2**

Table 5. Out-of-domain accuracies (%) of different DNA variants are shown. "$\mathbb{D}_{\mathcal{P}}$" indicates whether the diversity measure is explicitly enforced. "Bernoulli" and "Gaussian" denote the type of dropout variational family. Each experiment is repeated 3 times.

$\mathbb{D}_{\mathcal{P}}$	Variational Family	**OfficeHome**					**TerraIncognita**				
		A	C	P	R	Avg	L100	L38	L43	L46	Avg
×	Gaussian	67.1 ± 0.1	57.4 ± 0.1	78.6 ± 0.1	80.2 ± 0.4	70.8	54.8 ± 1.6	**48.2** ± 0.4	60.2 ± 1.0	43.2 ± 0.9	51.6
✓	Gaussian	67.2 ± 0.2	57.5 ± 0.1	78.7 ± 0.3	80.3 ± 0.4	70.9	56.0 ± 0.9	47.2 ± 1.3	60.9 ± 0.7	43.4 ± 0.7	51.9
×	Bernoulli	67.1 ± 0.3	57.6 ± 0.1	78.7 ± 0.2	80.4 ± 0.2	70.8	55.6 ± 1.8	47.9 ± 1.0	60.5 ± 0.4	43.2 ± 0.7	51.8
✓	Bernoulli	**67.7** ± 0.2	**57.7** ± 0.3	78.9 ± 0.2	**80.5** ± 0.2	**71.2**	**56.8** ± 1.2	47.0 ± 0.9	**61.0** ± 0.5	**44.0** ± 1.0	**52.2**

Table 6. Out-of-domain accuracies(%) by optimizing the cross-entropy (CE) loss and pruned Jensen-Shannon (PJS) loss with SWA and with explicitly enforcing diversity $\mathbb{D}_{\mathcal{P}}$. Each experiment is repeated 3 times.

loss	**OfficeHome**	**TerraIncognita**	**DomainNet**
CE	71.1 ± 0.1	51.9 ± 0.5	46.9 ± 0.1
PJS	71.2 ± 0.1	52.2 ± 0.4	47.2 ± 0.1

SWA results in Tab. 2 to the *ERM* results in Tab. 1 demonstrates the individual influence of loss functions: similar accuracies of two losses on OfficeHome, accuracies of PJS on TerraIncognita and DomainNet outperform.

We also inspect the optimization process by observing the loss functions during the training time between the model trained with the PJS loss and the model trained with the CE loss. Specifically, we visualize the empirical estimate of PJS/CE loss on the training set, validation set and test set for the single model and the SWA ensemble model along the optimization trajectory. The results are shown in Fig. 1. We observe that in the averaging and non-averaging scenario, our proposed PJS loss suffers a lower level of overfitting, i.e., smaller gaps between training losses and testing/validation losses. Whereas the CE loss suffers a relatively higher level of overfitting, in the sense that the test loss goes up in late iterations. The second intriguing aspect of our PJS loss is the smoothness over the validation and test data, indicating that our method suffers less influence of model selection. For both loss functions, the averaging version exhibits a lower level of overfitting and a smoother plot of loss values in late iterations, substantiating our motivation of considering classifier ensemble.

6. Conclusion and Future Works

This paper bridges DG and classifier ensemble both theoretically and methodologically. We propose a novel PJS divergence and a novel PJS loss, upon which we derive generalization bounds for the risk of the target domain. Grounded on the theoretical results, we propose a fast-ensemble method for DG, DNA. The DNA method is a highly competitive method with a low computational overhead. Extensive experiments validate the effectiveness of the proposed DNA. There are many possible directions for future works, to name a few: (a) For fairness without demographics (Hashimoto et al., 2018). The proposed DNA does not utilize the domain labels, highlighting its potential in protecting private features of the minority subgroups. (b) For learning under label noise. The proposed PJS loss is a bounded and symmetric loss. The boundedness and symmetry of a loss are empirically beneficial for robustness against label noise (Wang et al., 2019; Englesson & Azizpour, 2021). (c) For more competitive DG methods. DNA is a drop-in framework that tackles DG from the model misspecification perspective and is potentially compatible with existing DG principles for higher empirical performance.

Acknowledgements

This work is supported by the National Key Research and Development Program of China No. 2020AAA0106300 and National Natural Science Foundation of China No. 62102222.

References

Alquier, P., Ridgway, J., and Chopin, N. On the properties of variational approximations of gibbs posteriors. *The*

Journal of Machine Learning Research, 17(1):8374–8414, 2016.

Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Arpit, D., Wang, H., Zhou, Y., and Xiong, C. Ensemble of averages: Improving model selection and boosting performance in domain generalization. *arXiv preprint arXiv:2110.10832*, 2021.

Beery, S., Van Horn, G., and Perona, P. Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision*, pp. 456–473, 2018.

Ben-David, S., Blitzer, J., Crammer, K., Pereira, F., et al. Analysis of representations for domain adaptation. In *Advances in Neural Information Processing Systems*, 2007.

Blanchard, G., Lee, G., and Scott, C. Generalizing from several related classification tasks to a new unlabeled sample. In *Advances in Neural Information Processing Systems*, volume 24, pp. 2178–2186, 2011.

Blanchard, G., Deshmukh, A. A., Dogan, U., Lee, G., and Scott, C. Domain generalization by marginal transfer learning. *Journal of Machine Learning Research*, 2021.

Bui, M.-H., Tran, T., Tran, A., and Phung, D. Exploiting domain-specific features to enhance domain generalization. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Cha, J., Chun, S., Lee, K., Cho, H.-C., Park, S., Lee, Y., and Park, S. Swad: Domain generalization by seeking flat minima. In *Advances in Neural Information Processing Systems*, 2021.

David, S. B., Lu, T., Luu, T., and Pál, D. Impossibility theorems for domain adaptation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pp. 129–136. JMLR Workshop and Conference Proceedings, 2010.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., and Fei-Fei, L. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2009.

Domingos, P. M. Why does bagging work? a bayesian account and its implications. In *KDD*, pp. 155–158. Citeseer, 1997.

Donsker, M. D. and Varadhan, S. R. S. Asymptotic evaluation of certain markov process expectations for large time. iv. *Communications on Pure and Applied Mathematics*, 36(2):183–212, 1983.

Endres, D. M. and Schindelin, J. E. A new metric for probability distributions. *IEEE Transactions on Information Theory*, 49(7):1858–1860, 2003.

Englesson, E. and Azizpour, H. Generalized jensen-shannon divergence loss for learning with noisy labels. In *Advances in Neural Information Processing Systems*, 2021.

Fang, C., Xu, Y., and Rockmore, D. N. Unbiased metric learning: On the utilization of multiple datasets and web images for softening bias. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1657–1664, 2013.

Fort, S., Hu, H., and Lakshminarayanan, B. Deep ensembles: A loss landscape perspective. *arXiv preprint arXiv:1912.02757*, 2019.

Gal, Y. *Computational Complexity of Machine Learning*. PhD thesis, Department of Engineering, University of Cambridge, 2016.

Gal, Y. and Ghahramani, Z. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *International Conference on Machine Learning*, pp. 1050–1059. PMLR, 2016.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1), 2016.

Germain, P., Habrard, A., Laviolette, F., and Morvant, E. A pac-bayesian approach for domain adaptation with specialization to linear classifiers. In *International Conference on Machine Learning*, pp. 738–746. PMLR, 2013.

Germain, P., Bach, F., Lacoste, A., and Lacoste-Julien, S. Pac-bayesian theory meets bayesian inference. In *Advances in Neural Information Processing Systems*, pp. 1876–1884, 2016a.

Germain, P., Habrard, A., Laviolette, F., and Morvant, E. A new pac-bayesian perspective on domain adaptation. In *International Conference on Machine Learning*, pp. 859–868. PMLR, 2016b.

Gulrajani, I. and Lopez-Paz, D. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

Hardy, G. H., Littlewood, J. E., Pólya, G., Pólya, G., et al. *Inequalities*. Cambridge University Press, 1952.

Hashimoto, T., Srivastava, M., Namkoong, H., and Liang, P. Fairness without demographics in repeated loss minimization. In *International Conference on Machine Learning*, pp. 1929–1938. PMLR, 2018.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2016.

Hoeting, J. A., Madigan, D., Raftery, A. E., and Volinsky, C. T. Bayesian model averaging: a tutorial (with comments by m. clyde, david draper and ei george, and a rejoinder by the authors. *Statistical science*, 14(4):382–417, 1999.

Huang, Z., Wang, H., Xing, E. P., and Huang, D. Self-challenging improves cross-domain generalization. In *Proceedings of the European Conference on Computer Vision*, 2020.

Iwasawa, Y. and Matsuo, Y. Test-time classifier adjustment module for model-agnostic domain generalization. In *Advances in Neural Information Processing Systems*, 2021.

Izmailov, P., Podoprikhin, D., Garipov, T., Vetrov, D., and Wilson, A. G. Averaging weights leads to wider optima and better generalization. In *Conference on Uncertainty in Artificial Intelligence*, 2018.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Krueger, D., Caballero, E., Jacobsen, J.-H., Zhang, A., Binas, J., Zhang, D., Le Priol, R., and Courville, A. Out-of-distribution generalization via risk extrapolation. In *International Conference on Machine Learning*, 2021.

Li, B., Shen, Y., Wang, Y., Zhu, W., Reed, C. J., Zhang, J., Li, D., Keutzer, K., and Zhao, H. Invariant information bottleneck for domain generalization. *arXiv preprint arXiv:2106.06333*, 2021.

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

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. M. Learning to generalize: Meta-learning for domain generalization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018a.

Li, H., Pan, S. J., Wang, S., and Kot, A. C. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5400–5409, 2018b.

Li, H., Pan, S. J., Wang, S., and Kot, A. C. Domain generalization with adversarial feature learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2018c.

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

Li, Y., Gong, M., Tian, X., Liu, T., and Tao, D. Domain generalization via conditional invariant representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2018d.

Li, Y., Yang, Y., Zhou, W., and Hospedales, T. Feature-critic networks for heterogeneous domain generalization. In *International Conference on Machine Learning*, pp. 3915–3924. PMLR, 2019.

Loshchilov, I. and Hutter, F. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2018.

Maddox, W. J., Izmailov, P., Garipov, T., Vetrov, D. P., and Wilson, A. G. A simple baseline for bayesian uncertainty in deep learning. *Advances in Neural Information Processing Systems*, 32, 2019.

Mahajan, D., Tople, S., and Sharma, A. Domain generalization using causal matching. In *International Conference on Machine Learning*, pp. 7313–7324. PMLR, 2021.

Masegosa, A. Learning under model misspecification: Applications to variational and ensemble methods. In *Advances in Neural Information Processing Systems*, volume 33, pp. 5479–5491, 2020.

Michaelis, C., Mitzkus, B., Geirhos, R., Rusak, E., Bringmann, O., Ecker, A. S., Bethge, M., and Brendel, W. Benchmarking robustness in object detection: Autonomous driving when winter is coming. *arXiv preprint arXiv:1907.07484*, 2019.

Muandet, K., Balduzzi, D., and Schölkopf, B. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pp. 10–18. PMLR, 2013.

Nam, H. and Kim, H.-E. Batch-instance normalization for adaptively style-invariant neural networks. In *Advances in Neural Information Processing Systems*, 2018.

Nam, H., Lee, H., Park, J., Yoon, W., and Yoo, D. Reducing domain gap by reducing style bias. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021.

Ortega, L. A., Cabañas, R., and Masegosa, A. R. Diversity and generalization in neural network ensembles. *arXiv preprint arXiv:2110.13786*, 2021.

Peng, X., Bai, Q., Xia, X., Huang, Z., Saenko, K., and Wang, B. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1406–1415, 2019.

Piratla, V., Netrapalli, P., and Sarawagi, S. Efficient domain generalization via common-specific low-rank decomposition. In *International Conference on Machine Learning*, pp. 7728–7738. PMLR, 2020.

Polyanskiy, Y. and Wu, Y. Lecture notes on information theory. *Lecture Notes for ECE563 (UIUC) and*, 6(2012-2016):7, 2014.

Recht, B., Roelofs, R., Schmidt, L., and Shankar, V. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning*, pp. 5389–5400. PMLR, 2019.

Royden, H. L. and Fitzpatrick, P. *Real analysis*, volume 32. Macmillan New York, 1988.

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. Distributionally robust neural networks for group shifts: On the importance of regularization for worst-case generalization. In *International Conference on Learning Representations*, 2020.

Seo, S., Suh, Y., Kim, D., Kim, G., Han, J., and Han, B. Learning to optimize domain specific normalization for domain generalization. In *European Conference on Computer Vision*, pp. 68–83. Springer, 2020.

Sinha, A., Namkoong, H., and Duchi, J. Certifying some distributional robustness with principled adversarial training. In *International Conference on Learning Representations*, 2018.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The Journal of Machine Learning Research*, 15(1):1929–1958, 2014.

Su, J., Vargas, D. V., and Sakurai, K. One pixel attack for fooling deep neural networks. *IEEE Transactions on Evolutionary Computation*, 23(5):828–841, 2019.

Sun, B. and Saenko, K. Deep coral: Correlation alignment for deep domain adaptation. In *Proceedings of the European Conference on Computer Vision*, 2016.

Thomas, X., Mahajan, D., Pentland, A., and Dubey, A. Adaptive methods for aggregated domain generalization. *arXiv preprint arXiv:2112.04766*, 2021.

Tishby, N. and Zaslavsky, N. Deep learning and the information bottleneck principle. In *2015 IEEE Information Theory Workshop (ITW)*, pp. 1–5. IEEE, 2015.

Tishby, N., Pereira, F. C., and Bialek, W. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

Vapnik, V. N. An overview of statistical learning theory. *IEEE Transactions on Neural Networks*, 10(5), 1999.

Venkateswara, H., Eusebio, J., Chakraborty, S., and Panchanathan, S. Deep hashing network for unsupervised domain adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2017.

Volpi, R., Namkoong, H., Sener, O., Duchi, J. C., Murino, V., and Savarese, S. Generalizing to unseen domains via adversarial data augmentation. In Bengio, S., Wallach, H., Larochelle, H., Grauman, K., Cesa-Bianchi, N., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems*, volume 31, 2018.

Wang, D., Shelhamer, E., Liu, S., Olshausen, B., and Darrell, T. Tent: Fully test-time adaptation by entropy minimization. In *International Conference on Learning Representations*, 2020a.

Wang, Y., Ma, X., Chen, Z., Luo, Y., Yi, J., and Bailey, J. Symmetric cross entropy for robust learning with noisy labels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 322–330, 2019.

Wang, Y., Li, H., and Kot, A. C. Heterogeneous domain generalization via domain mixup. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3622–3626, 2020b.

Xiao, Z., Shen, J., Zhen, X., Shao, L., and Snoek, C. G. A bit more bayesian: Domain-invariant learning with uncertainty. In *International Conference on Machine Learning*, 2021.

Yi, M., Hou, L., Sun, J., Shang, L., Jiang, X., Liu, Q., and Ma, Z. Improved ood generalization via adversarial training and pretraing. In *International Conference on Machine Learning*, 2021.

Zhang, G., Zhao, H., Yu, Y., and Poupart, P. Quantifying and improving transferability in domain generalization. In *Advances in Neural Information Processing Systems*, 2021a.

Zhang, H., Cisse, M., Dauphin, Y. N., and Lopez-Paz, D. mixup: Beyond empirical risk minimization. In *International Conference on Learning Representations*, 2018.

Zhang, H., Zhang, Y.-F., Liu, W., Weller, A., Schölkopf, B., and Xing, E. P. Towards principled disentanglement for domain generalization. *arXiv preprint arXiv:2111.13839*, 2021b.

Zhang, J., Qi, L., Shi, Y., and Gao, Y. More is better: A novel multi-view framework for domain generalization. *arXiv preprint arXiv:2112.12329*, 2021c.

Zhang, M., Marklund, H., Dhawan, N., Gupta, A., Levine, S., and Finn, C. Adaptive risk minimization: A meta-learning approach for tackling group distribution shift. *arXiv*, 2020.

Zhao, H., Des Combes, R. T., Zhang, K., and Gordon, G. On learning invariant representations for domain adaptation. In *International Conference on Machine Learning*, 2019.

Zhou, K., Yang, Y., Qiao, Y., and Xiang, T. Domain adaptive ensemble learning. *IEEE Transactions on Image Processing*, 30:8008–8018, 2021.

Zhu, W., Cui, P., Wang, Z., and Hua, G. Multimedia big data computing. *IEEE multimedia*, 22(3):96–c3, 2015.

A. Proofs for Section 3

Definition A.1 (pruned Jensen-Shannon divergence)**.** Consider a subset of $M_+(\mathbb{X}\times\mathbb{Y})$ induced by a dominating measure λ, $M_+^\lambda=\{\mu\in M_+(\mathbb{X}\times\mathbb{Y}):\mu\ll\lambda\}$, such that each measure in M_+^λ is absolutely continuous (w.r.t.) the the dominating measure λ. For measures $\Upsilon,\mathcal{P}\in M_+^\lambda$ and $\mathcal{P}\ll\Upsilon$, let $p=\frac{\mathrm{d}\mathcal{P}}{\mathrm{d}\lambda}$ and $\upsilon=\frac{\mathrm{d}\Upsilon}{\mathrm{d}\lambda}$ denote the Radon–Nikodym derivatives accordingly. The pruned Jensen-Shannon (PJS) divergence $D_{PJS}(\mathcal{P}\|\Upsilon):M_+^\lambda\times M_+^\lambda\to\mathbb{R}$ from a measure Υ to a measure $\mathcal{P}$ is defined by an integral over the support set $\mathbb{A}_\mathcal{P}$ of measure $\mathcal{P}$, that is

$$D_{PJS}(\mathcal{P}\|\Upsilon)=\int_{\mathbb{A}_\mathcal{P}}p\log\frac{2p}{p+\upsilon}+\upsilon\log\frac{2\upsilon}{p+\upsilon}\mathrm{d}\lambda. \tag{1}$$

Theorem A.2 (properties of PJS divergence)**.** *Consider measures $\mathcal{Q},\mathcal{P},\Upsilon\in M_+^\lambda$. Suppose that $\mathcal{Q}\ll\Upsilon$, $\mathcal{P}\ll\Upsilon$ and $\mathbb{A}_\mathcal{P}=\mathbb{A}_\mathcal{Q}$, then the PJS divergence satisfies*
(a) $D_{PJS}(\mathcal{P}\|\mathcal{Q})=2D_{JS}(\mathcal{P}\|\mathcal{Q})$, where D_{JS} is the vanilla JS divergence.
(b) $D_{PJS}(\mathcal{P}\|\Upsilon)\geq 0$.
(c) $\sqrt{D_{PJS}(\mathcal{Q}\|\Upsilon)}\leq\sqrt{D_{PJS}(\mathcal{P}\|\Upsilon)}+\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}$.

Proof. Let $p=\frac{\mathrm{d}\mathcal{P}}{\mathrm{d}\lambda}$, $q=\frac{\mathrm{d}\mathcal{Q}}{\mathrm{d}\lambda}$ and $\upsilon=\frac{\mathrm{d}\Upsilon}{\mathrm{d}\lambda}$ denote the Radon–Nikodym derivatives accordingly.
(a) The proof is straightforward by recalling the definition of the original Jensen-Shannon divergence (Endres & Schindelin, 2003). Taking $\mathbb{A}=\mathbb{A}_\mathcal{P}=\mathbb{A}_\mathcal{Q}$,

$$\begin{aligned}D_{JS}(\mathcal{P}\|\mathcal{Q})&=\int_{\mathbb{A}}\frac{1}{2}p\log\frac{2p}{p+q}+\frac{1}{2}q\log\frac{2q}{p+q}\mathrm{d}\lambda\\&=\frac{1}{2}\int_{\mathbb{A}_\mathcal{P}}p\log\frac{2p}{p+q}+q\log\frac{2q}{p+q}\mathrm{d}\lambda=\frac{1}{2}D_{PJS}(\mathcal{P}\|\mathcal{Q}).\end{aligned} \tag{2}$$

(b) We show equivalently that $-D_{PJS}(\mathcal{P}\|\Upsilon)\leq 0$,

$$\begin{aligned}-D_{PJS}(\mathcal{P}\|\Upsilon)&=-\int_{\mathbb{A}_\mathcal{P}}p\log\frac{2p}{p+\upsilon}+\upsilon\log\frac{2\upsilon}{p+\upsilon}\mathrm{d}\lambda\\&=-\int_{\mathbb{A}_\mathcal{P}}\frac{\mathrm{d}\mathcal{P}}{\mathrm{d}\lambda}\log\frac{2\frac{\mathrm{d}\mathcal{P}}{\mathrm{d}\lambda}}{\frac{\mathrm{d}\mathcal{P}}{\mathrm{d}\lambda}+\frac{\mathrm{d}\Upsilon}{\mathrm{d}\lambda}}+\frac{\mathrm{d}\Upsilon}{\mathrm{d}\lambda}\log\frac{2\frac{\mathrm{d}\Upsilon}{\mathrm{d}\lambda}}{\frac{\mathrm{d}\mathcal{P}}{\mathrm{d}\lambda}+\frac{\mathrm{d}\Upsilon}{\mathrm{d}\lambda}}\mathrm{d}\lambda\\&=-\int_{\mathbb{A}_\mathcal{P}}\mathrm{d}\mathcal{P}\log\frac{2\mathrm{d}\mathcal{P}}{\mathrm{d}\mathcal{P}+\mathrm{d}\Upsilon}+\mathrm{d}\Upsilon\log\frac{2\mathrm{d}\Upsilon}{\mathrm{d}\mathcal{P}+\mathrm{d}\Upsilon}\\&=\int_{\mathbb{A}_\mathcal{P}}\mathrm{d}\mathcal{P}\log\frac{\mathrm{d}\mathcal{P}+\mathrm{d}\Upsilon}{2\mathrm{d}\mathcal{P}}+\mathrm{d}\Upsilon\log\frac{\mathrm{d}\mathcal{P}+\mathrm{d}\Upsilon}{2\mathrm{d}\Upsilon}\\&\overset{(i)}{\leq}\int_{\mathbb{A}_\mathcal{P}}\mathrm{d}\mathcal{P}(\frac{\mathrm{d}\mathcal{P}+\mathrm{d}\Upsilon}{2\mathrm{d}\mathcal{P}}-1)+\mathrm{d}\Upsilon(\frac{\mathrm{d}\mathcal{P}+\mathrm{d}\Upsilon}{2\mathrm{d}\Upsilon}-1)\\&=\int_{\mathbb{A}_\mathcal{P}}(\frac{\mathrm{d}\mathcal{P}+\mathrm{d}\Upsilon}{2}-\mathrm{d}\mathcal{P})+(\frac{\mathrm{d}\mathcal{P}+\mathrm{d}\Upsilon}{2}-\mathrm{d}\Upsilon)=0\end{aligned} \tag{3}$$

the inequality (i) uses the fact that $\log x\leq x-1$ for $x>0$.

(c) The triangle inequality simply relies on the lemma given in (Endres & Schindelin, 2003),

Lemma A.3 (Lemma 2 in Endres & Schindelin (2003))**.** *Let $p,q,\upsilon\in\mathbb{R}^+$ and let $L(p,q)\triangleq p\log\frac{2p}{p+q}+q\log\frac{2q}{p+q}$. Then*

$$\sqrt{L(q,\upsilon)}\leq\sqrt{L(p,\upsilon)}+\sqrt{L(p,q)} \tag{4}$$

We skip the proof for this lemma, please refer to the Lemma 2 in Endres & Schindelin (2003) for the detailed proof.

By definition of $D_{PJS}(\mathcal{P}\|\Upsilon)$ and take $\mathbb{A}=\mathbb{A}_{\mathcal{P}}=\mathbb{A}_{\mathcal{Q}}$, there is

$$\begin{aligned}\sqrt{D_{PJS}(\mathcal{Q}\|\Upsilon)} &= \sqrt{\int_{\mathbb{A}} q\log\frac{2q}{q+\upsilon}+\upsilon\log\frac{2\upsilon}{p+\upsilon}\mathrm{d}\lambda}\\ &= \sqrt{\int_{\mathbb{A}}(\sqrt{L(q,\upsilon)})^2\mathrm{d}\lambda}\\ &\overset{(i)}{\leq} \sqrt{\int_{\mathbb{A}}(\sqrt{L(p,\upsilon)}+\sqrt{L(p,q)})^2\mathrm{d}\lambda}\\ &\overset{(ii)}{\leq} \sqrt{\int_{\mathbb{A}}(\sqrt{L(p,\upsilon)})^2\mathrm{d}\lambda}+\sqrt{\int_{\mathbb{A}}(\sqrt{L(p,q)})^2\mathrm{d}\lambda}\\ &= \sqrt{D_{PJS}(\mathcal{P}\|\Upsilon)}+\sqrt{D_{PJS}(\mathcal{P}\|\mathcal{Q})}=\sqrt{D_{PJS}(\mathcal{P}\|\Upsilon)}+\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})},\end{aligned} \tag{5}$$

where the inequality (i) invokes the Lem. A.3 and the inequality (ii) invokes the Minkowski's inequality (Royden & Fitzpatrick (1988),Sect.7.2). □

Definition A.4 (pruned Jensen-Shannon loss)**.** Let $\mathbb{A}$ be the support of the underlying (empirical) distribution where the realization $(\mathbf{x},\mathbf{y})$ is drawn from. The pruned Jensen-Shannon (PJS) loss $\ell_{PJS}:\mathbb{X}\times\mathbb{I}\times\mathbb{H}\to\mathbb{R}^+$ is

$$\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})=(\log\frac{2}{h^*+1}+h^*\log\frac{2h^*}{h^*+1})\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}\}}, \tag{6}$$

where $h^*\in[0,1]$ is the vector component of $\mathbf{h}$ that corresponds to the correct class, and $\mathbb{1}_{\{\cdot\}}$ is the indicator function.

Proposition A.5 (connecting risk and PJS divergence)**.** *For a classifier $\boldsymbol{h}$ and a domain $\mathcal{P}\in M_+(\mathbb{X}\times\mathbb{I})$. Let $\Upsilon^{\mathcal{P}}$ be the induced measure. Assuming $p^{\mathcal{P}}(\boldsymbol{y}|\boldsymbol{x})\in\{0,1\}$, denoting $\upsilon^{\mathcal{P}}(\boldsymbol{x},\boldsymbol{y})=p(\boldsymbol{x})\boldsymbol{h}(\boldsymbol{y}|\boldsymbol{x})$ as the induced density of $\Upsilon^{\mathcal{P}}$, then*

$$R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{h})=\mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{y})\sim\mathcal{P}}[\ell_{PJS}(\boldsymbol{x},\boldsymbol{y},\boldsymbol{h})]=D_{PJS}(\mathcal{P}\|\Upsilon^{\mathcal{P}}) \tag{7}$$

Proof. For ease of notations, we omit the superscript referring $\mathcal{P}$ of $p^{\mathcal{P}}(\mathbf{x})$ and $\Upsilon^{\mathcal{P}}$, by using $p(\mathbf{x})$ and Υ instead. By definition of $D_{PJS}(\mathcal{P}\|\Upsilon)$ and the facts: (a) $p(y^*|\mathbf{x})=1$ for the correct class y^* and $p(y'|\mathbf{x})=0$ for the wrong class y' (Uniqueness: one-hot labels),(b)$1=\sum_{j=1}^{C}p(y_j|\mathbf{x},\mathbf{y})$. We have ($\mathrm{d}\mathbf{y}$ referring to the counting measure)

$$\begin{aligned}&D_{PJS}(\mathcal{P}\|\Upsilon)=\int_{\mathbb{A}_{\mathcal{P}}}p(\mathbf{x},\mathbf{y})\log\frac{2p(\mathbf{x},\mathbf{y})}{p(\mathbf{x},\mathbf{y})+\upsilon(\mathbf{x},\mathbf{y})}+\upsilon(\mathbf{x},\mathbf{y})\frac{2\upsilon(\mathbf{x},\mathbf{y})}{p(\mathbf{x},\mathbf{y})+\upsilon(\mathbf{x},\mathbf{y})}\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{y}\\
&=\int_{\mathbb{A}_{\mathcal{P}}}p(\mathbf{x})p(\mathbf{y}|\mathbf{x})\log\frac{2p(\mathbf{x},\mathbf{y})}{p(\mathbf{x},\mathbf{y})+\upsilon(\mathbf{x},\mathbf{y})}+p(\mathbf{x})h(\mathbf{y}|\mathbf{x})\log\frac{2\upsilon(\mathbf{x},\mathbf{y})}{p(\mathbf{x},\mathbf{y})+\upsilon(\mathbf{x},\mathbf{y})}\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{y}\\
&=\int_{\mathbb{A}_{\mathcal{P}}}p(\mathbf{x})p(\mathbf{y}|\mathbf{x})\left[\log\frac{2p(\mathbf{x},\mathbf{y})}{p(\mathbf{x},\mathbf{y})+\upsilon(\mathbf{x},\mathbf{y})}+\frac{h(\mathbf{y}|\mathbf{x})}{p(\mathbf{y}|\mathbf{x})}\log\frac{2\upsilon(\mathbf{x},\mathbf{y})}{p(\mathbf{x},\mathbf{y})+\upsilon(\mathbf{x},\mathbf{y})}\right]\mathrm{d}\mathbf{x}\mathrm{d}\mathbf{y}\\
&=\int_{\{\mathbf{x}:p(\mathbf{x})>0\}}p(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\sum_{j=1}^{C}p(y_j|\mathbf{x})\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_{\mathcal{P}}\}}[\frac{2p(\mathbf{x})p(y_j|\mathbf{x})}{p(\mathbf{x})p(y_j|\mathbf{x})+p(\mathbf{x})h_j}+\frac{h_j}{p(y_j|\mathbf{x})}\log\frac{2p(\mathbf{x})h_j}{p(\mathbf{x})p(y_j|\mathbf{x})+p(\mathbf{x})h_j}]\mathrm{d}\mathbf{x}\\
&\overset{invoking(a)}{=}\int_{\{\mathbf{x}:p(\mathbf{x})>0\}}p(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_{\mathcal{P}}\}}\left[\log\frac{2p(\mathbf{x})}{p(\mathbf{x})+p(\mathbf{x})h^*}+h^*\log\frac{2p(\mathbf{x})h^*}{p(\mathbf{x})+p(\mathbf{x})h^*}\right]\mathrm{d}\mathbf{x}\\
&=\int_{\{\mathbf{x}:p(\mathbf{x})>0\}}p(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_{\mathcal{P}}\}}\left[\log\frac{2}{1+h^*}+h^*\log\frac{2h^*}{1+h^*}\right]\mathrm{d}\mathbf{x}\\
&\overset{invoking(b)}{=}\int_{\{\mathbf{x}:p(\mathbf{x})>0\}}p(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_{\mathcal{P}}\}}\sum_{j=1}^{C}p(y_j|\mathbf{x})\left[\log\frac{2}{1+h^*}+h^*\log\frac{2h^*}{1+h^*}\right]\mathrm{d}\mathbf{x}=R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h}).\end{aligned} \tag{8}$$

□

Theorem A.6 (bounds of the target risk)**.** *Let $\mathcal{Q}$ and $\mathcal{P}$ be the (fixed) target domain and the source domain in $M_+(\mathbb{X}\times\mathbb{I})$, with $p^{\mathcal{Q}}(\boldsymbol{x})$ and $p^{\mathcal{P}}(\boldsymbol{x})$ being their density functions over the input space $\mathbb{X}$, respectively. Suppose the supports of $\mathcal{Q}$ and $\mathcal{P}$ are identical, i.e., $\mathbb{A}_{\mathcal{P}}=\mathbb{A}_{\mathcal{Q}}$, with $p^{\mathcal{P}}(\boldsymbol{y}|\boldsymbol{x}), p^{\mathcal{Q}}(\boldsymbol{y}|\boldsymbol{x})\in\{0,1\}$. Let $\mathcal{H}$ be a hypothesis space, such that for any $\boldsymbol{h}\in\mathbb{H}$, the density $p^{\mathcal{P}}(\boldsymbol{x})\boldsymbol{h}$ induced measure $\Upsilon^{\mathcal{P}}$ dominates[9] $\mathcal{P}$, and $\Upsilon^{\mathcal{Q}}$ dominates $\mathcal{Q}$, i.e., $\mathcal{P}\ll\Upsilon^{\mathcal{P}}$ and $\mathcal{Q}\ll\Upsilon^{\mathcal{Q}}$. Then for any $\boldsymbol{h}\in\mathbb{H}$,*

$$\begin{cases}\sqrt{R_{\mathcal{Q}}^{\ell_{PJS}}(\boldsymbol{h})}\le\sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{h})}+2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}. & (9)\\ \sqrt{R_{\mathcal{Q}}^{\ell_{PJS}}(\boldsymbol{h})}\ge\sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{h})}-2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}. & (10)\end{cases}$$

Proof. We use the following lemma to prove the first inquality.

Lemma A.7. *Given $P,Q\in M_+(\mathbb{X}\times\mathbb{I})$, suppose that Υ^Q is a measure induced by a classifier $\boldsymbol{h}$ and the marginal density function $p^Q(\boldsymbol{x})$ of Q, i.e., the density function υ^Q of Υ^Q is $\upsilon^Q=p^Q(\boldsymbol{x})\boldsymbol{h}$. Similarly, let Υ^P be a measure induced by $\boldsymbol{h}$ and the marginal density function $p^P(\boldsymbol{x})$ of P, with $\upsilon^P=p^P(\boldsymbol{x})\boldsymbol{h}$. Then if $p^{\mathcal{P}}(\boldsymbol{y}|\boldsymbol{x}), p^{\mathcal{Q}}(\boldsymbol{y}|\boldsymbol{x})\in\{0,1\}$, $P\ll Q$, $P\ll\Upsilon^P$ and $Q\ll\Upsilon^Q\ll,\lambda$, then the following holds,*

$$D_{PJS}(\Upsilon^P\|\Upsilon^Q)\le D_{PJS}(P\|Q) \tag{11}$$

proof of lemma: by definition of the PJS divergence, we have

$$\begin{aligned}
&D_{PJS}(\Upsilon^P\|\Upsilon^Q)=\int_{\mathbb{A}_P}\upsilon^P\log\frac{2\upsilon^P}{\upsilon^P+\upsilon^Q}+\upsilon^Q\log\frac{2\upsilon^Q}{\upsilon^P+\upsilon^Q}\mathrm{d}\lambda\\
&=\int_{\mathbb{A}_P}p^P(\mathbf{x})\mathbf{h}\log\frac{2p^P(\mathbf{x})\mathbf{h}}{p^P(\mathbf{x})\mathbf{h}+p^Q(\mathbf{x})\mathbf{h}}+p^Q(\mathbf{x})\mathbf{h}\log\frac{2p^Q(\mathbf{x})\mathbf{h}}{p^P(\mathbf{x})\mathbf{h}+p^Q(\mathbf{x})\mathbf{h}}\mathrm{d}\lambda\\
&\le\int_{\{\mathbf{x}:p^P(\mathbf{x})>0\}}p^P(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_P\}}\sum_{j=1}^{C}h(y_j|\mathbf{x})\log\frac{2p^P(\mathbf{x})h_j}{p^P(\mathbf{x})h_j+p^Q(\mathbf{x})h_j}\\
&\quad+p^Q(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_P\}}\sum_{j=1}^{C}h(y_j|\mathbf{x})\log\frac{2p^Q(\mathbf{x})h_j}{p^P(\mathbf{x})h_j+p^Q(\mathbf{x})h_j}\mathrm{d}\mathbf{x}\\
&=\int_{\{\mathbf{x}:p^P(\mathbf{x})>0\}}p^P(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_P\}}\log\frac{2p^P(\mathbf{x})}{p^P(\mathbf{x})+p^Q(\mathbf{x})}\sum_{j=1}^{C}h(y_j|\mathbf{x})\\
&\quad+p^Q(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_P\}}\log\frac{2p^Q(\mathbf{x})}{p^P(\mathbf{x})+p^Q(\mathbf{x})}\sum_{j=1}^{C}h(y_j|\mathbf{x})\mathrm{d}\mathbf{x}\\
&=\int_{\{\mathbf{x}:p^P(\mathbf{x})>0\}}p^P(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_P\}}\log\frac{2p^P(\mathbf{x})}{p^P(\mathbf{x})+p^Q(\mathbf{x})}+p^Q(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_P\}}\log\frac{2p^Q(\mathbf{x})}{p^P(\mathbf{x})+p^Q(\mathbf{x})}\mathrm{d}\mathbf{x}\\
&\overset{(i)}{=}\int_{\{\mathbf{x}:p^P(\mathbf{x})>0\}}p^P(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_P\}}p^P(y^*|\mathbf{x})\log\frac{2p^P(\mathbf{x})p^P(y^*|\mathbf{x})}{p^P(\mathbf{x})p^P(y^*|\mathbf{x})+p^Q(\mathbf{x})p^Q(y^*|\mathbf{x})}\\
&\quad+p^Q(\mathbf{x})\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_P\}}p^Q(y^*|\mathbf{x})\log\frac{2p^Q(\mathbf{x})p^Q(y^*|\mathbf{x})}{p^P(\mathbf{x})p^P(y^*|\mathbf{x})+p^Q(\mathbf{x})p^Q(y^*|\mathbf{x})}\mathrm{d}\mathbf{x}\\
&\le\int_{\{\mathbf{x}:p^P(\mathbf{x})>0\}}\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_P\}}p^P(\mathbf{x},\mathbf{y})\log\frac{2p^P(\mathbf{x},\mathbf{y})}{p^P(\mathbf{x},\mathbf{y})+p^Q(\mathbf{x},\mathbf{y})}+\sum_{\mathbf{y}\in\mathbb{I}}\mathbb{1}_{\{(\mathbf{x},\mathbf{y})\in\mathbb{A}_P\}}p^Q(\mathbf{x},\mathbf{y})\log\frac{2p^Q(\mathbf{x},\mathbf{y})}{p^P(\mathbf{x},\mathbf{y})+p^Q(\mathbf{x},\mathbf{y})}\mathrm{d}\mathbf{x}\\
&=D_{PJS}(P\|Q),
\end{aligned}\tag{12}$$

where equality (i) invokes the facts that $p^P(y^*|\mathbf{x})=1, p^P(y^*|\mathbf{x})=1, p^Q(y^*|\mathbf{x})=1$ and $P\ll Q$, since the labels are one-hot vectors and the probability is 1 for being in the correct class. **Q.E.D.** of the lemma A.7.

[9]This is not a strong requirement as long as the component of the correct class in vector $\mathbf{h}$ is non-zero.

We then apply the triangle inequality in Thm. A.2(c) to the measures $\mathcal{P}$,$\mathcal{Q}$,$\Upsilon^{\mathcal{P}}$ and $\Upsilon^{\mathcal{Q}}$, leading to

$$\begin{aligned}\sqrt{D_{PJS}(\mathcal{Q}\|\Upsilon^{\mathcal{Q}})} &\leq \sqrt{D_{PJS}(\mathcal{Q}\|\Upsilon^{\mathcal{P}})} + \sqrt{D_{PJS}(\Upsilon^{\mathcal{Q}}\|\Upsilon^{\mathcal{P}})} \\ &\leq \sqrt{D_{PJS}(\mathcal{P}\|\Upsilon^{\mathcal{P}})} + \sqrt{D_{PJS}(\mathcal{P}\|\mathcal{Q})} + \sqrt{D_{PJS}(\Upsilon^{\mathcal{Q}}\|\Upsilon^{\mathcal{P}})} \overset{(ii)}{\leq} \sqrt{D_{PJS}(\mathcal{P}\|\Upsilon^{\mathcal{P}})} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})},\end{aligned} \tag{13}$$

where Lem. A.7 and Thm. A.2(a) are applied to the inequality (ii).
Invoking Prop. A.5, there are

$$\begin{cases} R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h}) = D_{PJS}(\mathcal{P}\|\Upsilon^{\mathcal{P}}), & (14) \\ R_{\mathcal{Q}}^{\ell_{PJS}}(\mathbf{h}) = D_{PJS}(\mathcal{Q}\|\Upsilon^{\mathcal{Q}}), & (15) \end{cases}$$

thereof combined with Eq. (13) would lead to the desired inequality

$$\sqrt{R_{\mathcal{Q}}^{\ell_{PJS}}(\mathbf{h})} \leq \sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}. \tag{16}$$

Similarly, by switching argument $\mathcal{Q}$ and $\mathcal{P}$, we have

$$\begin{aligned}\sqrt{D_{PJS}(\mathcal{P}\|\Upsilon^{\mathcal{P}})} &\leq \sqrt{D_{PJS}(\mathcal{P}\|\Upsilon^{\mathcal{Q}})} + \sqrt{D_{PJS}(\Upsilon^{\mathcal{P}}\|\Upsilon^{\mathcal{Q}})} \\ &\leq \sqrt{D_{PJS}(\mathcal{Q}\|\Upsilon^{\mathcal{Q}})} + \sqrt{D_{PJS}(\mathcal{P}\|\mathcal{Q})} + \sqrt{D_{PJS}(\Upsilon^{\mathcal{P}}\|\Upsilon^{\mathcal{Q}})} \leq \sqrt{D_{PJS}(\mathcal{Q}\|\Upsilon^{\mathcal{Q}})} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})},\end{aligned} \tag{17}$$

Therefore invoking Prop. A.5 again would result in the second inequality

$$\sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})} \leq \sqrt{R_{\mathcal{Q}}^{\ell_{PJS}}(\mathbf{h})} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})} \Longleftrightarrow \sqrt{R_{\mathcal{Q}}^{\ell_{PJS}}(\mathbf{h})} \geq \sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})} - 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}. \tag{18}$$

□

Definition A.8 (ρ-ensemble)**.** Suppose that ρ is a measure on a hypothesis space $\mathbb{H}$. The ρ-ensemble $\boldsymbol{\rho}$ is the ρ-weighted averaged classifier,

$$\boldsymbol{\rho} = \mathbb{E}_{\mathbf{h}\sim\rho}(\mathbf{h}). \tag{19}$$

Theorem A.9 (inequalities related to the ρ-ensemble)**.** *Let $\mathbb{H}$ be a hypothesis space as stated in the Thm. A.6. For any $\rho \in M_+(\mathbb{H})$, any measure $\mathcal{P}$, take $\mathbb{D}_{\mathcal{P}} = \mathbb{E}_{\mathcal{P}}[Var_\rho(\sqrt{\ell_{PJS}})]$,*

$$\sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq \sqrt{\mathbb{E}_\rho[R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{h})] - \mathbb{D}_{\mathcal{P}}} \leq \mathbb{E}_\rho[\sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{h})}]. \tag{20}$$

Moreover, $\mathbb{D}_{\mathcal{P}} > 0$ is a necessary condition for the second inequality becoming strict. If $\frac{\ell_{PJS}(\boldsymbol{x}_1,\boldsymbol{y},\boldsymbol{h})}{\ell_{PJS}(\boldsymbol{x}_2,\boldsymbol{y},\boldsymbol{h})}$ is varying on (a non-zero measured subset of) the set $\{(\boldsymbol{x},\boldsymbol{y},\boldsymbol{h}) : Var_\rho(\sqrt{\ell_{PJS}}) > 0\}$ for $\boldsymbol{x}_1 \neq \boldsymbol{x}_2$, then $\mathbb{D}_{\mathcal{P}} > 0$ is a sufficient condition for the second inequality being strict.

Proof. The inequalities are relying on the convexity induced by the design of the PJS loss. We prove the inequalities based on the following lemma.

Lemma A.10. *The function $g(t)$ is convex on $t \in (0, 1)$*

$$g(t) = \sqrt{\log\frac{2}{t+1} + t\log\frac{2t}{t+1}}. \tag{21}$$

proof of lemma: we check the sign of the second derivative of $g(t)$.

$$\frac{\mathrm{d}g(t)}{\mathrm{d}t} = \frac{\log\frac{2t}{t+1}}{2g(t)}. \tag{22}$$

$$\frac{\mathrm{d}^2 g(t)}{\mathrm{d}t^2} = \frac{2g^2(t) - t(t+1)\log^2(\frac{2t}{t+1})}{4t(t+1)g^3(t)} = \frac{2\log\frac{2}{t+1} + 2t\log\frac{2t}{t+1} - t(t+1)\log^2(\frac{2t}{t+1})}{4t(t+1)g^3(t)}. \tag{23}$$

It suffices to check the sign of numerator of Eq. (23) since the denominator $4t(1+t)g^3(t)$ is positive on the interval $t \in (0,1)$. Let $G(t) = 2\log\frac{2}{t+1} + 2t\log\frac{2t}{t+1} - t(t+1)\log^2(\frac{2t}{t+1})$,

$$\frac{\mathrm{d}G(t)}{\mathrm{d}t} = -(2t+1)\log^2(\frac{2t}{t+1}) < 0, \ for \ \ t \in (0,1). \tag{24}$$

Thence $G(t)$ is decreasing on $t \in (0,1)$ and we have

$$G(t) > G(1) = 2\log\frac{2}{1+1} + 2\log\frac{2}{1+1} - 2\log^2(\frac{2}{1+1}) = 0. \tag{25}$$

Therefore $\frac{\mathrm{d}^2 g(t)}{\mathrm{d}t^2} > 0$ on the interval $t \in (0,1)$, which implies $g(t)$ is convex on $t \in (0,1)$. **Q.E.D.** of the lemma A.10.

Remark A.11. *T*he inequalities in the Thm. 3.7, Cor. 3.8, Cor. 3.9 and Thm. 3.10 rely on the design of the PJS divergence. Specifically, the key is the convexity of the function $g(t) = \sqrt{\log\frac{2}{t+1} + t\log\frac{2t}{t+1}}$ on $t \in (0,1)$. Whereas the function $g^*(t) = \sqrt{\log\frac{2}{t+1} + t\log\frac{2t}{t+1} + (1-t)log2}$ is not convex nor concave on $t \in (0,1)$. Thence within the framework of the original JS divergence, the corresponding inequalities do not necessarily hold.

Next we prove the first inequality $\sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq \sqrt{\mathbb{E}_\rho[R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}}}$ with $\mathbb{D}_{\mathcal{P}} = \mathbb{E}_{\mathcal{P}}[Var_\rho(\sqrt{\ell_{PJS}})]$.
By Lem. A.10 and Jensen's inequality (Royden & Fitzpatrick (1988),Sect.6.6),

$$g(\mathbb{E}_\rho(h^*)) \leq \mathbb{E}_\rho[g(h^*)]. \tag{26}$$

Since $g(t)$ is non-negative, then

$$g^2(\mathbb{E}_\rho(h^*)) \leq (\mathbb{E}_\rho[g(h^*)])^2. \tag{27}$$

Taking expectation over $\mathcal{P}$ on both side of Eq. (27) leads to

$$\mathbb{E}_{\mathcal{P}}[g^2(\mathbb{E}_\rho(h^*))] \leq \mathbb{E}_{\mathcal{P}}[\mathbb{E}_\rho g(h^*)]^2. \tag{28}$$

It suffices to show the left hand side (L.H.S.) of Eq. (28) is $R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{\rho})$ and the right hand side (R.H.S.) of Eq. (28) is $\mathbb{E}_\rho[R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}}$ with $\mathbb{D}_{\mathcal{P}} = \mathbb{E}_{\mathcal{P}}[Var_\rho(\sqrt{\ell_{PJS}})]$, which are straightforward by the definitions,

$$\begin{aligned} L.H.S : g^2(\mathbb{E}_\rho(h^*)) &= \log\frac{2}{\mathbb{E}_\rho(h^*)} + \mathbb{E}_\rho(h^*)\log\frac{\mathbb{E}_\rho(h^*)}{\mathbb{E}_\rho(h^*)+1} \\ &\Longrightarrow \mathbb{E}_{\mathcal{P}}[g^2(\mathbb{E}_\rho(h^*))] = \mathbb{E}_{\mathcal{P}}[\ell_{PJS}(\mathbf{x},\mathbf{y},\boldsymbol{\rho})] = R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{\rho}). \end{aligned} \tag{29}$$

$$\begin{aligned} R.H.S : \mathbb{E}_{\mathcal{P}}[\mathbb{E}_\rho g(h^*)]^2 &= \mathbb{E}_{\mathcal{P}}[\mathbb{E}_\rho g^2(h^*) - Var_\rho(g(h^*))] = \mathbb{E}_{\mathcal{P}}[\mathbb{E}_\rho g^2(h^*)] - \mathbb{E}_{\mathcal{P}}[Var_\rho(g(h^*))] \\ &= \mathbb{E}_{\mathcal{P}}[\mathbb{E}_\rho \ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})] - \mathbb{E}_{\mathcal{P}}[Var_\rho(\sqrt{\ell_{PJS}})] \overset{(i)}{=} \mathbb{E}_\rho[\mathbb{E}_{\mathcal{P}}\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})] - \mathbb{D}_{\mathcal{P}} \\ &= \mathbb{E}_\rho[R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}}, \end{aligned} \tag{30}$$

where the equality (i) invokes the Fubini's theorem (Royden & Fitzpatrick (1988),Sect.20.1). Thus we have proved the first inequality

$$R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{\rho}) \leq= \mathbb{E}_\rho[R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}} \Longrightarrow \sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq \sqrt{\mathbb{E}_\rho[R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}}}. \tag{31}$$

Next we prove the second inequality $\sqrt{\mathbb{E}_\rho[R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}}} \leq \mathbb{E}_\rho[\sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})}]$. By Eq. (30), it suffices to show $\sqrt{\mathbb{E}_{\mathcal{P}}[\mathbb{E}_\rho g(h^*)]^2} \leq \mathbb{E}_\rho[\sqrt{R_{\mathcal{P}}^{\ell_{PJS}}(\mathbf{h})}]$. This is equivalent to show

$$\sqrt{\mathbb{E}_{\mathcal{P}}[\mathbb{E}_\rho g(h^*)]^2} \leq \mathbb{E}_\rho[\sqrt{\mathbb{E}_{\mathcal{P}} g^2(h^*)}] \Longleftrightarrow \quad \left(\int_{\mathbb{A}_{\mathcal{P}}}[\int_{\mathbb{A}_\rho} g(h^*)\mathrm{d}\rho]^2\mathrm{d}\mathcal{P}\right)^{\frac{1}{2}} \leq \int_{\mathbb{A}_\rho}[\int_{\mathbb{A}_{\mathcal{P}}} g^2(h^*)\mathrm{d}\mathcal{P}]^{\frac{1}{2}}\mathrm{d}\rho, \tag{32}$$

which is guaranteed to be true by the Minkowski's integral inequality (Hardy et al. (1952),Thm.202).

The inequality in Eq. (32) becomes equality if and only if $\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h}) = g^2(h^*)\mathbb{1}_{\{\mathbf{x},\mathbf{y}\in\mathbb{A}_\mathcal{P}\}} = \psi(\mathbf{x},\mathbf{y})\phi(\mathbf{h})$ for some fixed measurable functions $\psi(\cdot)$ and $\phi(\cdot)$ almost everywhere.

When there is a non-zero measured set $B \subseteq \{\mathbb{A}_\mathcal{P} \cap \mathbb{A}_\rho\}$ such that $\frac{\ell_{PJS}(\mathbf{x}_1,\mathbf{y},\mathbf{h})}{\ell_{PJS}(\mathbf{x}_2,\mathbf{y},\mathbf{h})}$ is varying, the existence of the non-zero measured set B would result in a contradiction of the existence of function $\psi(\cdot)$ and $\phi(\cdot)$.

Assuming such set B, functions $\psi(\cdot)$ and $\phi(\cdot)$ exist at the same time, then on B, when $\mathbf{h}_1 \neq \mathbf{h}_2$, $\mathbf{x}_1 \neq \mathbf{x}_2$,

$$\frac{\ell_{PJS}(\mathbf{x}_1,\mathbf{y},\mathbf{h}_1)}{\ell_{PJS}(\mathbf{x}_2,\mathbf{y},\mathbf{h}_1)} \neq \frac{\ell_{PJS}(\mathbf{x}_1,\mathbf{y},\mathbf{h}_2)}{\ell_{PJS}(\mathbf{x}_2,\mathbf{y},\mathbf{h}_2)} \Longleftrightarrow \frac{\psi(\mathbf{x}_1,\mathbf{y})\phi(\mathbf{h}_1)}{\psi(\mathbf{x}_2,\mathbf{y})\phi(\mathbf{h}_1)} \neq \frac{\psi(\mathbf{x}_1,\mathbf{y})\phi(\mathbf{h}_2)}{\psi(\mathbf{x}_2,\mathbf{y})\phi(\mathbf{h}_2)} \Longrightarrow \frac{\psi(\mathbf{x}_1,\mathbf{y})}{\psi(\mathbf{x}_2,\mathbf{y})} \neq \frac{\psi(\mathbf{x}_1,\mathbf{y})}{\psi(\mathbf{x}_2,\mathbf{y})}, \tag{33}$$

which reduces to absurdity. Therefore when such a set B exists, the second inequaility of the theorem is strict.

The necessity condition of $\mathbb{D}_\mathcal{P} > 0$:
If $\mathbb{D}_\mathcal{P} = 0$, then $\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})$ is constant on $\mathbb{A}_\mathcal{P} \cap \mathbb{A}_\rho$, then there are infinitely many solutions for funcitons $\psi(\cdot)$ and $\phi(\cdot)$ being exist, thus the second inequality of the theorem would be equality. Therefore if the inequality is strict, $\mathbb{D}_\mathcal{P}$ can not be 0, which proves the necessity condition. □

Corollary A.12 (target risk upper bound of the ensembles)**.** *Given a fixed source domain $\mathcal{P}$ and a target domain $\mathcal{Q}$, for any measure $\rho \in M_+(\mathbb{H})$ on hypothesis space $\mathbb{H}$,*

$$\sqrt{R_\mathcal{Q}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq \sqrt{\mathbb{E}_\rho[R_\mathcal{P}^{\ell_{PJS}}(\boldsymbol{h})] - \mathbb{D}_\mathcal{P}} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}. \tag{34}$$

Proof. The proof is straightforward by combining Thm. A.6 and Thm. A.9.

$$\sqrt{R_\mathcal{Q}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq \sqrt{R_\mathcal{P}^{\ell_{PJS}}(\mathbf{h})} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})} \leq \sqrt{\mathbb{E}_\rho[R_\mathcal{P}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_\mathcal{P}} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})} \tag{35}$$

□

Corollary A.13 (joint risk upper bound of the ensembles)**.** *Given a fixed source domain $\mathcal{P}$ and a target domain $\mathcal{Q}$, for any measure $\rho \in M_+(\mathbb{H})$ on hypothesis space $\mathbb{H}$,*

$$\sqrt{R_\mathcal{P}^{\ell_{PJS}}(\boldsymbol{\rho})} + \sqrt{R_\mathcal{Q}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq \sqrt{\mathbb{E}_\rho[R_\mathcal{P}^{\ell_{PJS}}(\boldsymbol{h})] - \mathbb{D}_\mathcal{P}} + \sqrt{\mathbb{E}_\rho[R_\mathcal{Q}^{\ell_{PJS}}(\boldsymbol{h})] - \mathbb{D}_\mathcal{Q}} \leq \mathbb{E}_\rho[\sqrt{R_\mathcal{P}^{\ell_{PJS}}(\boldsymbol{h})}] + \mathbb{E}_\rho[\sqrt{R_\mathcal{Q}^{\ell_{PJS}}(\boldsymbol{h})}]. \tag{36}$$

Proof. It's easy to check the requirement of applying Thm. A.9 is satisfied by both source domain $\mathcal{P}$ and the target domain $\mathcal{Q}$. The proof is straightforward by invoking Thm. A.9 twice w.r.t. $\mathcal{P}$ and $\mathcal{Q}$.

$$\begin{cases} \sqrt{R_\mathcal{P}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq \sqrt{\mathbb{E}_\rho[R_\mathcal{P}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_\mathcal{P}} \leq \mathbb{E}_\rho[\sqrt{R_\mathcal{P}^{\ell_{PJS}}(\mathbf{h})}]. & (37) \\ \sqrt{R_\mathcal{Q}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq \sqrt{\mathbb{E}_\rho[R_\mathcal{Q}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_\mathcal{Q}} \leq \mathbb{E}_\rho[\sqrt{R_\mathcal{Q}^{\ell_{PJS}}(\mathbf{h})}]. & (38) \end{cases}$$

Therefore we have

$$\sqrt{R_\mathcal{P}^{\ell_{PJS}}(\boldsymbol{\rho})} + \sqrt{R_\mathcal{Q}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq \sqrt{\mathbb{E}_\rho[R_\mathcal{P}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_\mathcal{P}} + \sqrt{\mathbb{E}_\rho[R_\mathcal{Q}^{\ell_{PJS}}(\mathbf{h})] - \mathbb{D}_\mathcal{Q}} \leq \mathbb{E}_\rho[\sqrt{R_\mathcal{P}^{\ell_{PJS}}(\mathbf{h})}] + \mathbb{E}_\rho[\sqrt{R_\mathcal{Q}^{\ell_{PJS}}(\mathbf{h})}]. \tag{39}$$

□

Theorem A.14 (PAC-Bayesian generalization upper bound)**.** *For a fixed source domain $\mathcal{P}$ and a fixed target domain $\mathcal{Q}$, let $\mathbb{H}$ be a hypothesis space as stated in the Thm. A.6. Suppose that π is a prior over $\mathbb{H}$, which is independent of draws of source realizations $D^n = \{(\boldsymbol{x}_i,\boldsymbol{y}_i)\}_{i=1}^n \overset{i.i.d.}{\sim} \mathcal{P}^n$. Then for any $c > 0$, $\rho \in M_+(\mathbb{H})$, and any $\delta \in (0,1)$, with probability over $1-\delta$*

$$\sqrt{R_\mathcal{Q}^{\ell_{PJS}}(\boldsymbol{\rho})} \leq 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})} + \sqrt{\mathbb{E}_\rho[\hat{R}_\mathcal{P}^{\ell_{PJS}}(\boldsymbol{h})] - \hat{\mathbb{D}}_\mathcal{P} + \frac{2D_{KL}(\rho\|\pi) + \log\frac{1}{\delta} + \Psi_{\mathcal{P},\pi}^{\ell_{PJS}}(c,n)}{cn}}, \tag{40}$$

where $\Psi_{\mathcal{P},\pi}^{\ell}(c,n) = \log \mathbb{E}_{\pi^2}\mathbb{E}_{\mathcal{P}^n}[e^{cn\mathbb{E}_\mathcal{P}\sqrt{\ell(\boldsymbol{h}')}\mathbb{E}_\mathcal{P}\sqrt{\ell(\boldsymbol{h})} - \hat{\mathbb{E}}_\mathcal{P}\sqrt{\ell(\boldsymbol{h}')}\hat{\mathbb{E}}_\mathcal{P}\sqrt{\ell(\boldsymbol{h})}}]$ is constant w.r.t. ρ for fixed c, n, π, ℓ, and δ.

Proof. The main part of the proof of this theorem employs the same technique as in the proof of Thm.3 in Masegosa (2020), which relies on a lemma (Thm.3 in Germain et al. (2016a)) derived from the acknowledged Donsker and Varadhan's variational formula (Donsker & Varadhan, 1983).

Lemma A.15 (Alquier et al. (2016),Germain et al. (2016a)). *Given a distribution $\mathcal{P} \in M_+(\mathbb{X} \times \mathbb{Y})$, a hypothesis space $\mathbb{H}$, a loss function $\ell : \mathbb{X} \times \mathbb{Y} \times \mathbb{H} \longrightarrow \mathbb{R}$, a prior distribution $\pi \in M_+(\mathbb{H})$. Given a $\delta \in (0, 1)$ and a real number $\zeta > 0$, with probability at least $1 - \delta$ over the choice $\mathcal{P}^n$, for any $\rho \in M_+(\mathbb{H})$,*

$$\mathbb{E}_{\boldsymbol{h}\sim\rho}[L^{\ell}_{\mathcal{P}}(\boldsymbol{h})] \leq \mathbb{E}_{\boldsymbol{h}\sim\rho}[\hat{L}^{\ell}_{\mathcal{P}}(\boldsymbol{h})] + \frac{1}{\zeta}[D_{KL}(\rho\|\pi) + \log\frac{1}{\delta} + \Psi_{\mathcal{P},\pi}(\zeta, n)], \tag{41}$$

where $\Psi_{\mathcal{P},\pi}(\zeta, n) = \log \mathbb{E}_{\boldsymbol{h}\sim\pi}\mathbb{E}_{\mathcal{P}^n}[e^{\zeta(L^{\ell}_{\mathcal{P}}(\boldsymbol{h}) - \hat{L}^{\ell}_{\mathcal{P}}(\boldsymbol{h}))}]$.

We skip the proof for this lemma, please refer to the Thm. 3 in Germain et al. (2016a) for the detailed proof. Recalling the Eq. 30 in the proof of Thm. A.9, where we got $\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}} = \mathbb{E}_{\mathcal{P}}[\mathbb{E}_\rho g(h^*)]^2$, then

$$\begin{aligned}
\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}} &= \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{P}}[\mathbb{E}_{\mathbf{h}\sim\rho}\sqrt{\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})}]^2 \\
&= \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{P}}[\mathbb{E}_{\mathbf{h}\sim\rho}\sqrt{\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})}\mathbb{E}_{\mathbf{h}\sim\rho}\sqrt{\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})}] \\
&= \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{P}}[\mathbb{E}_{\mathbf{h}'\sim\rho}\sqrt{\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h}')}\mathbb{E}_{\mathbf{h}\sim\rho}\sqrt{\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})}] \\
&= \mathbb{E}_{(\mathbf{h}',\mathbf{h})\sim\rho^2}[\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{P}}\sqrt{\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h}')}\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{P}}\sqrt{\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})}]
\end{aligned} \tag{42}$$

Let $L^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h}', \mathbf{h}) = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{P}}\sqrt{\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h}')}\mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{P}}\sqrt{\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})}$. Regarding $L^{\ell}_{\mathcal{P}}(\cdot)$ in Lem. A.15 as $L^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h}', \mathbf{h})$, and invoking the fact $D_{KL}(\rho^2\|\pi^2) = 2D_{KL}(\rho\|\pi)$, from Lem. A.15, there is

$$\begin{aligned}
\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}} &= \mathbb{E}_{(\mathbf{h}',\mathbf{h})\sim\rho^2}[L^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h}', \mathbf{h})] \\
&\leq \mathbb{E}_{(\mathbf{h}',\mathbf{h})\sim\rho^2}[L^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h}', \mathbf{h})] + \frac{1}{cn}[2D_{KL}(\rho\|\pi) + \log\frac{1}{\delta} + \Psi_{\mathcal{P},\pi}(c, n)] \\
&= \mathbb{E}_\rho[\hat{R}^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})] - \hat{\mathbb{D}}_{\mathcal{P}} + \frac{1}{cn}[2D_{KL}(\rho\|\pi) + \log\frac{1}{\delta} + \Psi_{\mathcal{P},\pi}(c, n)],
\end{aligned} \tag{43}$$

where we take $\zeta = cn$. Note that $\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}} = \mathbb{E}_{(\mathbf{x},\mathbf{y})\sim\mathcal{P}}[\mathbb{E}_{\mathbf{h}\sim\rho}\sqrt{\ell_{PJS}(\mathbf{x},\mathbf{y},\mathbf{h})}]^2 > 0$, thus we have

$$\sqrt{\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}}} \leq \sqrt{\mathbb{E}_\rho[\hat{R}^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})] - \hat{\mathbb{D}}_{\mathcal{P}} + \frac{1}{cn}[2D_{KL}(\rho\|\pi) + \log\frac{1}{\delta} + \Psi_{\mathcal{P},\pi}(c, n)]}. \tag{44}$$

Combining Cor. A.12 and Eq. (44) completes the proof,

$$\begin{aligned}
\sqrt{R^{\ell_{PJS}}_{\mathcal{Q}}(\boldsymbol{\rho})} &\leq \sqrt{R^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})} \leq \sqrt{\mathbb{E}_\rho[R^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})] - \mathbb{D}_{\mathcal{P}}} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})} \\
&\leq \sqrt{\mathbb{E}_\rho[\hat{R}^{\ell_{PJS}}_{\mathcal{P}}(\mathbf{h})] - \hat{\mathbb{D}}_{\mathcal{P}} + \frac{1}{cn}[2D_{KL}(\rho\|\pi) + \log\frac{1}{\delta} + \Psi_{\mathcal{P},\pi}(c, n)]} + 2\sqrt{2D_{JS}(\mathcal{P}\|\mathcal{Q})}.
\end{aligned} \tag{45}$$

□

B. Experimental details.

B.1. Baseline details

This appendix provides a detailed description of the 14 baseline methods used for benchmark comparisons.

- Empirical Risk Minimization (**ERM**, (Vapnik, 1999)) aggregates the data from all source domains together, and minimizes the cross entropy loss for classification.
- Invariant Risk Minimization (**IRM**, (Arjovsky et al., 2019)) learns a feature mapping such that the optimal linear classifier on top of that representation matches across source domains.
- Group Distributionally Robust Optimization (**DRO**, (Sagawa et al., 2020)) performs ERM while increasing the importance of domains with larger error by re-weighting minibatches.
- Inter-domain Mixup (**Mixup**, (Wang et al., 2020b)) employs mixup (Zhang et al., 2018) technique across multiple domains and performs ERM on the augmented heterogeneous mixup distribution.
- Meta-Learning for Domain Generalization (**MLDG**, (Li et al., 2018a)) divides the source domains into meta-train-domains and meta-test-domain to simulate domain shift, and regulate the model trained on meta-train-domains to perform well on meta-test-domain.
- Deep CORrelation ALignment (**CORAL**, (Sun & Saenko, 2016)) matches the first-order (mean) and the second-order (covariance) statistics of feature distributions across source domains.
- Maximum Mean Discrepancy (**MMD**, (Li et al., 2018c)) achieves distribution alignment in the latent space of an autoencoder by using adversarial learning and the maximum mean discrepancy criteria.
- Domain Adversarial Neural Network (**DANN**, (Ganin et al., 2016)) employs a domain discriminator to align feature distributions across domains using adversarial learning.
- Class-conditional Domain Adversarial Neural Network (**CDANN**, (Li et al., 2018d)) matches conditional feature distributions across domains, enabling alignment of multimodal distributions for all class labels.
- Marginal Transfer Learning (**MTL**, (Blanchard et al., 2021)) estimates a kernel mean embedding per domain, passed as a second argument to the classifier. Then, these embeddings are estimated using single test examples at test time.
- Style Agnostic Networks (**SagNet**, (Nam et al., 2021)) disentangle style encodings from class categories to prevent style biased predictions and focus more on the contents.
- Adaptive Risk Minimization (**ARM**, (Zhang et al., 2020) is an extension of MLDG and introduces an additional module to compute domain embeddings, which are used by the prediction module to infer information about the input distribution.
- Variance Risk Extrapolation (**VREx**, (Krueger et al., 2021)) is a form of robust optimization over a perturbation set of extrapolated domains and minimizes the variance of training risks across domains.
- Representation Self-Challenging (**RSC**, (Huang et al., 2020)) iteratively discards the dominant features activated on the training data, and forces the CNN to activate remaining features that correlates with labels.
- Meta-Domain Specific-Domain Invariant (**mDSDI**, (Bui et al., 2021)) disentangles features in the latent space while jointly learning the domain-invariant and domain-specific features in a unified meta-learning framework. The domain-specific information is utilized to enhance predictions.
- Stochastic Weight Averaging Densely (**SWAD**, (Cha et al., 2021)) finds flatter minima and suffers less from overfitting by modifying the vanilla SWA with a dense and overfit-aware sampling strategy.

B.2. Implementation details.

Training and Evaluation protocol. We follow the training and evaluation protocol described in (Gulrajani & Lopez-Paz, 2021) and our implementation is built upon the released SWAD[10] library. For training, we choose a domain as the target domain and use the remaining domains as source domains. We split each source domain into 8:2 training/validation splits and integrate the validation subsets of each source domain to create an overall validation set, which is used for validation and model selection.

Model Architectures. We use the ResNet-50 pre-trained on ImageNet as the backbone network and all the BN layers are frozen during training. We replace the last FC layer of the ResNet-50 with a 2-layer classifier with 1024 hidden units and employ the dropout regularization on the 1024-dimensional output (except for DomainNet, 2048 hidden units are used since its label set is much larger).

Hyperparameters. We use the hyperparameter search protocol in (Gulrajani & Lopez-Paz, 2021) with a slight modification following (Cha et al., 2021): we do not search independently for each test domain, use hyperparameters searched in the first random data split to the other splits and search algorithm-specific hyperparameters independently from the universal ones. We use a smaller search space (shown in Table 1) for computational efficiency. Batch size and ResNet dropout rate are fixed as 32 and 0. All the SWAD-specific hyperparameters are not searched and the default values are used. For DNA hyperparameters, we search the FC dropout rate and η. Here, the FC dropout rate is searched in $[0.1, 0.3, 0.5]$ depending on datasets, hence 0.1 is used in DomainNet and 0.5 is used in the other datasets. η is searched in $[0.01, 0.1, 1.0]$ on PACS and the searched value 0.1 is used for all experiments. The total number of training iterations is 20000 for DomainNet and 5000 for the other datasets, which are enough numbers for our method to be converged. The evaluation frequency is: 500 for DomainNet, 50 for VLCS and 100 for the others.

Table 1. **Hyperparameter search space comparisons.** "Uni" and list denote uniform distribution and random choice, respectively.

Hyper-Parameter	Default Value	DomainBed	Ours
Batch size	32	$2^{\mathrm{Uni}(3,5.5)}$	32
Learning rate	5e-5	$10^{\mathrm{Uni}(-5,-3.5)}$	[3e-5, 4e-5, 5e-5]
ResNet dropout	0	[0, 0.1, 0.5]	0
Weight decay	0	$10^{\mathrm{Uni}(-6,-2)}$	[1e-6,1e-4]

B.3. Experimental environments.

Hardware environments. We perform our experiments on three machines: two with 8 Nvidia RTX3090s and Xeon E5-2680, and one with 4 Nvidia V100 and Xeon Platinum 8163.

Software environments. Our experiments are conducted with Python 3.7.9, and the following packages are used: PyTorch 1.7.1, torchvision 0.8.2 and NumPy 1.19.4.

[10] https://github.com/khanrc/swad

C. Full results.

This section provides full results of Table 1 for each domain of each dataset.

C.1. PACS

Table 2. Out-of-domain accuracies(%) on PACS.

Method	A	C	P	S	Avg
ERM (Vapnik, 1999)	84.7 ± 0.4	80.8 ± 0.6	97.2 ± 0.3	79.3 ± 1.0	85.5
IRM (Arjovsky et al., 2019)	84.8 ± 1.3	76.4 ± 1.1	96.7 ± 0.6	76.1 ± 1.0	83.5
DRO (Sagawa et al., 2020)	83.5 ± 0.9	79.1 ± 0.6	96.7 ± 0.3	78.3 ± 2.0	84.4
Mixup (Wang et al., 2020b)	86.1 ± 0.5	78.9 ± 0.8	97.6 ± 0.1	75.8 ± 1.8	84.6
MLDG (Li et al., 2018a)	85.5 ± 1.4	80.1 ± 1.7	97.4 ± 0.3	76.6 ± 1.1	84.9
CORAL (Sun & Saenko, 2016)	88.3 ± 0.2	80.0 ± 0.5	97.5 ± 0.3	78.8 ± 1.3	86.2
MMD (Li et al., 2018c)	86.1 ± 1.4	79.4 ± 0.9	96.6 ± 0.2	76.5 ± 0.5	84.6
DANN (Ganin et al., 2016)	86.4 ± 0.8	77.4 ± 0.8	97.3 ± 0.4	73.5 ± 2.3	83.6
CDANN (Li et al., 2018d)	84.6 ± 1.8	75.5 ± 0.9	96.8 ± 0.3	73.5 ± 0.6	82.6
MTL (Blanchard et al., 2021)	87.5 ± 0.8	77.1 ± 0.5	96.4 ± 0.8	77.3 ± 1.8	84.6
SagNet (Nam et al., 2021)	87.4 ± 1.0	80.7 ± 0.6	97.1 ± 0.1	80.0 ± 0.4	86.3
ARM (Zhang et al., 2020)	86.8 ± 0.6	76.8 ± 0.5	97.4 ± 0.3	79.3 ± 1.2	85.1
VREx (Krueger et al., 2021)	86.0 ± 1.6	79.1 ± 0.6	96.9 ± 0.5	77.7 ± 1.7	84.9
RSC (Huang et al., 2020)	85.4 ± 0.8	79.7 ± 1.8	97.6 ± 0.3	78.2 ± 1.2	85.2
mDSDI (Bui et al., 2021)	87.7 ± 0.4	80.4 ± 0.7	**98.1** ± 0.3	78.4 ± 1.2	86.2
SWAD (Cha et al., 2021)	89.3 ± 0.2	**83.4** ± 0.6	97.3 ± 0.3	82.5 ± 0.5	88.1
DNA (ours)	**89.8** ± 0.2	**83.4** ± 0.4	97.7 ± 0.1	**82.6** ± 0.2	**88.4**

C.2. VLCS

Table 3. Out-of-domain accuracies(%) on VLCS.

Method	C	L	S	V	Avg
ERM (Vapnik, 1999)	97.7 ± 0.4	64.3 ± 0.9	73.4 ± 0.5	74.6 ± 1.3	77.5
IRM (Arjovsky et al., 2019)	98.6 ± 0.1	64.9 ± 0.9	73.4 ± 0.6	77.3 ± 0.9	78.5
DRO (Sagawa et al., 2020)	97.3 ± 0.3	63.4 ± 0.9	69.5 ± 0.8	76.7 ± 0.7	76.7
Mixup (Wang et al., 2020b)	98.3 ± 0.6	64.8 ± 1.0	72.1 ± 0.5	74.3 ± 0.8	77.4
MLDG (Li et al., 2018a)	97.4 ± 0.2	65.2 ± 0.7	71.0 ± 1.4	75.3 ± 1.0	77.2
CORAL (Sun & Saenko, 2016)	98.3 ± 0.1	66.1 ± 1.2	73.4 ± 0.3	77.5 ± 1.2	78.8
MMD (Li et al., 2018c)	97.7 ± 0.1	64.0 ± 1.1	72.8 ± 0.2	75.3 ± 3.3	77.5
DANN (Ganin et al., 2016)	**99.0** ± 0.3	65.1 ± 1.4	73.1 ± 0.3	77.2 ± 0.6	78.6
CDANN (Li et al., 2018d)	97.1 ± 0.3	65.1 ± 1.2	70.7 ± 0.8	77.1 ± 1.5	77.5
MTL (Blanchard et al., 2021)	97.8 ± 0.4	64.3 ± 0.3	71.5 ± 0.7	75.3 ± 1.7	77.2
SagNet (Nam et al., 2021)	97.9 ± 0.4	64.5 ± 0.5	71.4 ± 1.3	77.5 ± 0.5	77.8
ARM (Zhang et al., 2020)	98.7 ± 0.2	63.6 ± 0.7	71.3 ± 1.2	76.7 ± 0.6	77.6
VREx (Krueger et al., 2021)	98.4 ± 0.3	64.4 ± 1.4	74.1 ± 0.4	76.2 ± 1.3	78.3
RSC (Huang et al., 2020)	97.9 ± 0.1	62.5 ± 0.7	72.3 ± 1.2	75.6 ± 0.8	77.1
mDSDI (Bui et al., 2021)	97.6 ± 0.1	**66.4** ± 0.4	74.0 ± 0.6	77.8 ± 0.7	79.0
SWAD (Cha et al., 2021)	98.8 ± 0.1	63.3 ± 0.3	**75.3** ± 0.5	79.2 ± 0.6	**79.1**
DNA (ours)	98.8 ± 0.1	63.6 ± 0.2	74.1 ± 0.1	**79.5** ± 0.4	79.0

C.3. OfficeHome

Table 4. Out-of-domain accuracies(%) on OfficeHome.

Method	A	C	P	R	Avg
ERM (Vapnik, 1999)	61.3 ± 0.7	52.4 ± 0.3	75.8 ± 0.1	76.6 ± 0.3	66.5
IRM (Arjovsky et al., 2019)	58.9 ± 2.3	52.2 ± 1.6	72.1 ± 2.9	74.0 ± 2.5	64.3
DRO (Sagawa et al., 2020)	60.4 ± 0.7	52.7 ± 1.0	75.0 ± 0.7	76.0 ± 0.7	66.0
Mixup (Wang et al., 2020b)	62.4 ± 0.8	54.8 ± 0.6	76.9 ± 0.3	78.3 ± 0.2	68.1
MLDG (Li et al., 2018a)	61.5 ± 0.9	53.2 ± 0.6	75.0 ± 1.2	77.5 ± 0.4	66.8
CORAL (Sun & Saenko, 2016)	65.3 ± 0.4	54.4 ± 0.5	76.5 ± 0.1	78.4 ± 0.5	68.7
MMD (Li et al., 2018c)	60.4 ± 0.2	53.3 ± 0.3	74.3 ± 0.1	77.4 ± 0.6	66.3
DANN (Ganin et al., 2016)	59.9 ± 1.3	53.0 ± 0.3	73.6 ± 0.7	76.9 ± 0.5	65.9
CDANN (Li et al., 2018d)	61.5 ± 1.4	50.4 ± 2.4	74.4 ± 0.9	76.6 ± 0.8	65.8
MTL (Blanchard et al., 2021)	61.5 ± 0.7	52.4 ± 0.6	74.9 ± 0.4	76.8 ± 0.4	66.4
SagNet (Nam et al., 2021)	63.4 ± 0.2	54.8 ± 0.4	75.8 ± 0.4	78.3 ± 0.3	68.1
ARM (Zhang et al., 2020)	58.9 ± 0.8	51.0 ± 0.5	74.1 ± 0.1	75.2 ± 0.3	64.8
VREx (Krueger et al., 2021)	60.7 ± 0.9	53.0 ± 0.9	75.3 ± 0.1	76.6 ± 0.5	66.4
RSC (Huang et al., 2020)	60.7 ± 1.4	51.4 ± 0.3	74.8 ± 1.1	75.1 ± 1.3	65.5
mDSDI (Bui et al., 2021)	**68.1** ± 0.3	52.1 ± 0.4	76.0 ± 0.2	80.4 ± 0.2	69.2
SWAD (Cha et al., 2021)	66.1 ± 0.4	**57.7** ± 0.4	78.4 ± 0.1	80.2 ± 0.2	70.6
DNA (ours)	67.7 ± 0.2	**57.7** ± 0.3	**78.9** ± 0.2	**80.5** ± 0.2	**71.2**

C.4. TerraIncognita

Table 5. Out-of-domain accuracies(%) on TerraIncognita.

Method	L100	L38	L43	L46	Avg
ERM (Vapnik, 1999)	49.8 ± 4.4	42.1 ± 1.4	56.9 ± 1.8	35.7 ± 3.9	46.1
IRM (Arjovsky et al., 2019)	54.6 ± 1.3	39.8 ± 1.9	56.2 ± 1.8	39.6 ± 0.8	47.6
DRO (Sagawa et al., 2020)	41.2 ± 0.7	38.6 ± 2.1	56.7 ± 0.9	36.4 ± 2.1	43.2
Mixup (Wang et al., 2020b)	**59.6** ± **2.0**	42.2 ± 1.4	55.9 ± 0.8	33.9 ± 1.4	47.9
MLDG (Li et al., 2018a)	54.2 ± 3.0	44.3 ± 1.1	55.6 ± 0.3	36.9 ± 2.2	47.7
CORAL (Sun & Saenko, 2016)	51.6 ± 2.4	42.2 ± 1.0	57.0 ± 1.0	39.8 ± 2.9	47.6
MMD (Li et al., 2018c)	41.9 ± 3.0	34.8 ± 1.0	57.0 ± 1.9	35.2 ± 1.8	42.2
DANN (Ganin et al., 2016)	51.1 ± 3.5	40.6 ± 0.6	57.4 ± 0.5	37.7 ± 1.8	46.7
CDANN (Li et al., 2018d)	47.0 ± 1.9	41.3 ± 4.8	54.9 ± 1.7	39.8 ± 2.3	45.8
MTL (Blanchard et al., 2021)	49.3 ± 1.2	39.6 ± 6.3	55.6 ± 1.1	37.8 ± 0.8	45.6
SagNet (Nam et al., 2021)	53.0 ± 2.9	43.0 ± 2.5	57.9 ± 0.6	40.4 ± 1.3	48.6
ARM (Zhang et al., 2020)	49.3 ± 0.7	38.3 ± 2.4	55.8 ± 0.8	38.7 ± 1.3	45.5
VREx (Krueger et al., 2021)	48.2 ± 4.3	41.7 ± 1.3	56.8 ± 0.8	38.7 ± 3.1	46.4
RSC (Huang et al., 2020)	50.2 ± 2.2	39.2 ± 1.4	56.3 ± 1.4	40.8 ± 0.6	46.6
mDSDI (Bui et al., 2021)	53.2 ± 3.0	43.3 ± 1.0	56.7 ± 0.5	39.2 ± 1.3	48.1
SWAD (Cha et al., 2021)	55.4 ± 0.0	44.9 ± 1.1	59.7 ± 0.4	39.9 ± 0.2	50.0
DNA (ours)	56.8 ± 1.2	**47.0** ± **0.9**	**61.0** ± 0.5	**44.0** ± 1.0	**52.2**

C.5. DomainNet

Table 6. Out-of-domain accuracies(%) on DomainNet.

Method	C	I	P	Q	R	S	Avg
ERM (Vapnik, 1999)	58.1 ± 0.3	18.8 ± 0.3	46.7 ± 0.3	12.2 ± 0.4	59.6 ± 0.1	49.8 ± 0.4	40.9
IRM (Arjovsky et al., 2019)	48.5 ± 2.8	15.0 ± 1.5	38.3 ± 4.3	10.9 ± 0.5	48.2 ± 5.2	42.3 ± 3.1	33.9
DRO (Sagawa et al., 2020)	47.2 ± 0.5	17.5 ± 0.4	33.8 ± 0.5	9.3 ± 0.3	51.6 ± 0.4	40.1 ± 0.6	33.3
Mixup (Wang et al., 2020b)	55.7 ± 0.3	18.5 ± 0.5	44.3 ± 0.5	12.5 ± 0.4	55.8 ± 0.3	48.2 ± 0.5	39.2
MLDG (Li et al., 2018a)	59.1 ± 0.2	19.1 ± 0.3	45.8 ± 0.7	13.4 ± 0.3	59.6 ± 0.2	50.2 ± 0.4	41.2
CORAL (Sun & Saenko, 2016)	59.2 ± 0.1	19.7 ± 0.2	46.6 ± 0.3	13.4 ± 0.4	59.8 ± 0.2	50.1 ± 0.6	41.5
MMD (Li et al., 2018c)	32.1 ± 13.3	11.0 ± 4.6	26.8 ± 11.3	8.7 ± 2.1	32.7 ± 13.8	28.9 ± 11.9	23.4
DANN (Ganin et al., 2016)	53.1 ± 0.2	18.3 ± 0.1	44.2 ± 0.7	11.8 ± 0.1	55.5 ± 0.4	46.8 ± 0.6	38.3
CDANN (Li et al., 2018d)	54.6 ± 0.4	17.3 ± 0.1	43.7 ± 0.9	12.1 ± 0.7	56.2 ± 0.4	45.9 ± 0.5	38.3
MTL (Blanchard et al., 2021)	57.9 ± 0.5	18.5 ± 0.4	46.0 ± 0.1	12.5 ± 0.1	59.5 ± 0.3	49.2 ± 0.1	40.6
SagNet (Nam et al., 2021)	57.7 ± 0.3	19.0 ± 0.2	45.3 ± 0.3	12.7 ± 0.5	58.1 ± 0.5	48.8 ± 0.2	40.3
ARM (Zhang et al., 2020)	49.7 ± 0.3	16.3 ± 0.5	40.9 ± 1.1	9.4 ± 0.1	53.4 ± 0.4	43.5 ± 0.4	35.5
VREx (Krueger et al., 2021)	47.3 ± 3.5	16.0 ± 1.5	35.8 ± 4.6	10.9 ± 0.3	49.6 ± 4.9	42.0 ± 3.0	33.6
RSC (Huang et al., 2020)	55.0 ± 1.2	18.3 ± 0.5	44.4 ± 0.6	12.2 ± 0.2	55.7 ± 0.7	47.8 ± 0.9	38.9
mDSDI (Bui et al., 2021)	62.1 ± 0.3	19.1 ± 0.4	49.4 ± 0.4	12.8 ± 0.7	62.9 ± 0.3	50.4 ± 0.4	42.8
SWAD (Cha et al., 2021)	66.0 ± 0.1	22.4 ± 0.3	53.5 ± 0.1	16.1 ± 0.2	**65.8** ± 0.4	55.5 ± 0.3	46.5
DNA (ours)	**66.1** ± 0.2	**23.0** ± 0.1	**54.6** ± 0.1	**16.7** ± 0.1	**65.8** ± 0.2	**56.8** ± 0.1	**47.2**