Re-labeling Domains Improves Multi-Domain Generalization

Kowshik Thopalli[1], **Pavan Turaga**[1], **Jayaraman J. Thiagarajan**[2]
[1]Arizona State University, [2] Lawrence Livermore National Laboratory
kthopall@asu.edu, pturaga@asu.edu, jjayaram@llnl.gov

Abstract

In multi-source zero shot domain generalization, the goal is to train using data from multiple source domains and generalize to any arbitrary target domain. Surprisingly, it has been found that standard empirical risk minimization (ERM) is a highly competitive baseline for this problem. More importantly, even sophisticated approaches that explicitly optimize for invariance across different domains do not necessarily provide non-trivial gains over ERM. We hypothesize that this behavior arises due to the poor definition of the domain splits itself. In this paper, we make a first attempt to understand the role pre-defined domain labels play in the success of domain-aware DG methods. We conduct analysis on two standard benchmarks PACS and VLCS and demonstrate the benefit of re-categorizing samples into new domain groups on DG performance.

1 Introduction

When the commonly made *i.i.d.* (independent and identically distributed) assumption between train and test data is violated, the performance of machine learned models can suffer [16]. In this paper, we consider the problem of generalizing to any arbitrary target domain, when data from multiple source domains are used to train models. Commonly referred to as zero-shot, multi-domain generalization (MDG), this does not assume any *a priori* knowledge about the target domain. The simplest, yet highly effective, solution to this problem is a naïve empirical risk minimization (ERM) [7, 17] approach that minimizes an average loss computed on data pooled together from all available source domains. Interestingly, even sophisticated approaches that explicitly optimize for invariance across different domains do not always provide non-trivial gains over ERM.

Theoretical results of average risk estimation error bound for MDG in the binary classification setting were provided in [2] and they have been extended to muti-class classification in [12]. We reproduce Theorem 3 from [18] which provides an upper bound on average risk estimation error for MDG using binary classifiers. Here M is the number of domains, n denotes number of samples in each domain, h denotes the hypothesis and $\mathcal{E}(h)$ represents average risk over all possible target domains by h.

Theorem 1 (Average risk estimation error bound for binary classification [2]). Assume that the loss function ℓ is L_ℓ-Lipschitz in its first argument and is bounded by B_ℓ. Assume also that the kernels k_X, k'_X and κ are bounded by $B_k^2, B_{k'}^2 \geq 1$ and B_κ^2, respectively, and the canonical feature map $\Phi_\kappa : v \in \mathcal{H}_{k'_X} \mapsto \kappa(v, \cdot) \in \mathcal{H}_\kappa$ of κ is L_κ-Hölder of order $\alpha \in (0, 1]$ on closed ball $\mathcal{B}_{\mathcal{H}_{k'_X}}(B_{k'})$ [1].

[1]This means that for any $u, v \in \mathcal{B}_{\mathcal{H}_{k'_X}}(\mathcal{B}_{\mathcal{H}_{k'}})$, it holds that $\|\Phi_\kappa(u) - \Phi_\kappa(v)\| \leq L_\kappa \|u - v\|^\alpha$, where the norms are of the respective RKHSs.

Workshop on Distribution Shifts, 35th Conference on Neural Information Processing Systems (NeurIPS 2021).

Then for any $r > 0$ and $\delta \in (0, 1)$, with probability at least $1 - \delta$, it holds that:

$$\sup_{h \in \mathcal{B}_{\mathcal{H}_{\tilde{k}}}(r)} \left| \hat{\mathcal{E}}(h) - \mathcal{E}(h) \right| \leq C \Bigg(B_\ell \sqrt{-M^{-1} \log \delta} + r B_k L_\ell \Big(B_{k'} L_\kappa \big(n^{-1} \log(M/\delta) \big)^{\alpha/2} + B_\kappa / \sqrt{M} \Big) \Bigg), \tag{1}$$

where C is a constant.

Note that this average risk upper bound becomes larger in general if (M, n) is replaced with (1, Mn) thus indicating that using domain-wise datasets is better than pooling them into one single dataset [18]. Under the light of this result, the success of ERM over methods that use domain-specific datasets is very surprising and warrants attention. We hypothesize the reason for the poor performance of domain-aware DG solutions to be the poor quality of domain labels. Through this paper, we make a first attempt at understanding the important link between domain-aware MDG solutions and domain labels. We are especially interested in re-grouping data so that they can be meaningfully leveraged for improving the worst-case performance. To this end, we propose a new MDG solution that includes an iterative unsupervised clustering step to infer domain labels which are then used by a domain-aware algorithm to build generalizable models. For implementing the domain-specific MDG algorithm, we consider GroupDRO a distributional robust optimization algorithm, which improves upon standard ERM by maintaining an adaptive weight for every group so that worst-case performance is maximized. Furthermore, we propose a novel regularization into GroupDRO to better guide the unsupervised learning step. We conduct experiments on two popular MDG benchmarks, PACS [11] and VLCS [5], and show that our approach improves over ERM and GroupDRO, which uses the domain labels that come with the dataset. Our results clearly validate our hypothesis of needing better domain labels for achieving non-trivial improvements over ERM.

2 Multi-domain Generalization

Problem Setup. Given access to M labeled source domains $\{\mathcal{D}_1, \ldots, \mathcal{D}_M\}$ where $\mathcal{D}_m = \{(x_i, y_i)\}_{i=1}^{N_m} \sim P^{(m)}(X, Y)$ is the m^{th} domain comprising N_m samples-label pairs, the goal is to generalize to a novel test domain $\mathcal{D}^\dagger$, without requiring any labeled or unlabeled examples. In this paper, we consider the homogeneous MDG setting *i.e.*, observed and unobserved domains share the same label space. Existing solutions for MDG can be grouped into three non-mutually exclusive categories [18] as: (i) Data manipulation which encompasses data augmentation and data generation ; (ii) Domain invariant representation learning; and (iii) methods that use advanced learning strategies, *e.g.*, meta-learning, self-supervised learning. We refer readers to surverys [18] for a detailed exposition. More importantly, MDG methods can also be broadly seen as as domain-aware i.e., methods that use pre-defined domain labels like [4, 10, 14, 15] and domain-agnostic methods such as ERM [17], IRM [1] etc. which ignore those labels. In this paper, we study the relationship between pre-defined domain-labels and domain-aware MDG algorithms.

ERM Baseline. A straight-forward and a simple MDG method is to train a network on data pooled from all source domains. Formally, With $\ell \colon \mathcal{X} \times \mathcal{Y} \to \mathbb{R}^+$ as the loss function that measures predictive error such as cross-entropy loss, we aim to learn a function $f : x \to y$ that maps samples x to labels y by minimizing the empirical risk. This simple baseline method does not leverage the inherent discrepancies between the source domains, and as a result one might expect this to be ineffective in practice. However, surprisingly, this solution has been shown to produce competitive performance to state-of-the-art methods on standard DG benchmarks through the use of appropriate model selection strategies [7] and hyper-parameter search.

3 Approach

Improving ERM by maximizing worst-case performance. A major drawback of ERM formulation is that it treats all samples from all domains/groups equally and thus decreases the loss in an average sense. An insight from distributional robust optimization literature is that increasing the worst-group performance can lead to better generalization. Hence, a natural extension of ERM is to decrease a weighted mean of the group-level loss with an adaptive weight per-domain such that large weights

are given to groups that have poor performance. Formally, let $g_i = k \in \{1, \cdots, K\}$ denote the group to which a sample (x_i, y_i) belongs to, where K is the total number of groups, and q_k is the weight for a particular group. The risk now becomes

$$\frac{1}{|\mathcal{D}|} \sum_{(x_i, y_i, g_i) \in \mathcal{D}} \ell\left(q_{g_i} f\left(x_i\right), y_i\right); \quad \mathcal{D} \coloneqq \bigcup_m \mathcal{D}_m, \tag{2}$$

A similar formulation was adopted in [14] for MDG, wherein an update rule for q_g across iterations was also proposed, such that a group with larger error will get a higher weight. Note that this approach while has shown to be effective for certain datasets, the improvements in performance compared to ERM are not consistent across benchmarks. As GroupDRO requires explicit domain labels g_i for each sample to maximize the worst-case performance, appropriate domain labels are critical for its success. We argue that this is the case for all MDG methods that explicitly leverage domain labels in the dataset like [6, 10, 14, 15, 19] etc. We make thus, a first attempt at studying the important link between choosing the appropriate group labels and generalization performance.

ClusterDG. To this end, we ignore the domain-labels that come with standard MDG benchmarks such as (photos, sketch etc in PACS [11] dataset) and attempt to re-categorize the data during training. This re-grouping is performed such that domain-aware MDG approaches can better leverage the inter-group and intra-group variations so as to produce better generalizable solutions. Our proposed solution ClusterDG alternatively perform unsupervised clustering to re-group samples and minimize the objective in Eq (2). We also introduce a novel regularization strategy to make latent space amenable for meaningful clustering.

Algorithm 1: ClusterDG-Reg

Input: Set of training domains $\mathcal{D} \coloneqq \{\mathcal{D}_1 \ldots \mathcal{D}_m\}$, hyper-parameters $K, \lambda, \gamma, \eta_q$
Output: Trained model f_θ
Initialization: Model parameters $\theta^{(0)}$, group labels for each sample $\{G\}$ and q^0 ;
for *iter* ***in*** n_{iter} **do**
  for t ***in*** $1 \cdots T$ **do**
    //run GroupDRO
    for g *in* $1 \cdots K$ **do**
      $q' \leftarrow q^{(t-1)}$
      //update group weight
      $q'_g \leftarrow q'_g \exp(\eta_q \ell(\theta^{(t-1)}; (x, y)))$
    end
    //Renormalize q
    $q^{(t)} \leftarrow q' / \sum_g q'_{g'}$;
    //GroupDRO loss
    $\mathcal{L} = \sum_g q_g^{(t)} \ell_g\left(\theta^{(t-1)}; (x, y)\right)$;
    //q_i is group-weight for sample i
    $\mathcal{L}_{reg} = \sum_i^N \ell_i q_i^\gamma$;
    //update θ $\theta^{(t)} \leftarrow \theta^{(t-1)} - \alpha \nabla_\theta (\mathcal{L} + \lambda \mathcal{L}_{reg})$
  end
  //Cluster the samples//
  $\mathcal{Z} \coloneqq \cup_i f_\theta(x_i)$;
  $G \leftarrow k$-means($\mathcal{Z}$)
end

Figure 1: An outline of the proposed approach.

Unsupervised Clustering. While there are many algorithms to cluster data, our design choice is to use a simple algorithm such as k-means. DeepCluster [3] is a popular deep learning based clustering solution that iteratively does (i) cluster latent representations via k-means and (ii) use the cluster-labels as pseudo-labels to train the network with a classification loss. Our pipeline is inspired from DeepCluster with a key difference in the notion of clusters- i.e., our clusters could contain samples from many classes rather than samples from only one category like in DeepCluster.

Approach. We begin by first extracting the latent representations of data from all source domains via a pre-trained network. We represent it by $\mathcal{Z}$: $\{z_1, z_2 \ldots . z_N\}$ with $z_i = f_\theta(x_i)$ where f is the model parameterized by θ and N denotes total number of samples pooled from all training domains. Using k-means, $\mathcal{Z}$ is clustered to form K groups and each sample can now be represented as a tuple (x_i, y_i, g_i). Model f_θ is trained with a GroupDRO style optimization i.e., with adaptive weights for each group for certain iterations following which, we freeze the model to obtain the latent representations of data and re-cluster. We repeat this process until convergence.

ClusterDG-reg. In our approach, since the clustering algorithm is disconnected from the training except for utilizing the latent vectors, it thus becomes important to regularize the training so that suitable domain groups can be created. Note that, ERM on one end aggregates losses at a micro scale i.e., sample-level, while GroupDRO operates at macro scale i.e, at group level, there is need to have a finer control between these two strategies. To illustrate this intuitively, consider these non-desirable cases: (i) An individual sample can have a high loss while its group has been assigned a smaller weight. In this case, the update to f_θ via GroupDRO will not have the desired effect as the weight is low; (ii) On the other hand, a sample having a low loss value when its group has a larger weight. Due to this wrong group association, that sample would still be updated during SGD.

To combat these issues, we introduce a novel regularization term that balances both the group-level weighting and sample-level weighting: $\mathcal{L}_{reg} = \ell_i q_{g_i}^\gamma$ where ℓ_i is sample level mis-classification error, q_{g_i} is the weight of the group to which the sample belongs to, and $0 < \gamma < 1$ controls the sharpness

Table 1: **PACS dataset**. By re-labeling the domains, ClusterDG-Reg provides significant improvements over ERM and GroupDRO. Ablation experiments GroupDRO-random and ClusterDG further demonstrate the efficacy of ClusterDG-Reg.

	A	**C**	**P**	**S**	**Average**
ERM	83.2 ± 0.3	76.8 ± 1.1	97.2 ± 0.5	74.8 ± 0.8	83.0
GroupDRO	83.5 ± 0.9	79.1 ± 0.6	96.7 ± 0.3	78.3 ± 2.0	84.4
GroupDRO-random	**86.1 ± 2.1**	**81.8± 2.08**	95.16 ± 0.82	78.97 ± 4.35	85.54
ClusterDG	86.03 ± 0.5	77.72 ± 0.3	96.4 ± 0.4	79.01 ± 1.0	84.79
ClusterDG-Reg	84.99 ± 0.2	82.78 ± 0.4	**97.2± 0.3**	**81.2± 0.6**	**86.6**

Table 2: **VLCS dataset**. This result also clearly demonstrates the benefit of domain re-labeling.

	C	**L**	**S**	**V**	**Average**
ERM	97.7 ± 0.4	64.3 ± 0.9	73.4 ± 0.5	74.6 ± 1.3	77.5
GroupDRO	97.3 ± 0.3	63.4 ± 0.9	69.5 ± 0.8	76.7 ± 0.7	76.7
ClusterDG-Reg	**98.41 ± 0.5**	**67.34± 0.8**	**75.7± 0.2**	**77.79 ± 0.4**	**79.81**

of regularization. Thus the final objective is given by $\mathcal{L} + \lambda\mathcal{L}_{reg}$ where $\mathcal{L}, \mathcal{L}_{reg}$ and λ denotes the hyper-parameter that weights the regularization term when attempting to maximize worst-case performance. Our algorithm is explained in Figure 1.

4 Experiments

We evaluate ClusterDG using two standard visual multi-domain generalization benchmarks (i) PACS [11] and (ii) VLCS [5] dataset. Following standard practice in MDG, for every dataset, we run experiments by leaving out one of M domains for testing while using the $M-1$ domains for training. To enable a fair comparison with the state-of-the-art, we implement ClusterDG in publicly available DomainBed[2] and use ResNet-50 [8], pre-trained on ImageNet [13] as the backbone feature extractor. Across experiments we use the following hyper-parameters: (i) batch size of 32 per domain; (ii) learning rate α as $5e-5$; (iii) Adam optimizer [9], (iv) number of training iterations n_{iter} set to 5000 and T as 300 (v) number of groups K is set to 4 as we didnot see any significant benefit going beyond 4 and (vi) λ, γ, η_q are set to 0.1, 0.3 and 0.2 respectively. To cluster we use k-means implementation from [3]. All experiments are run thrice and mean and standard deviation are reported.

In Tables 1 and 2, we present results on PACS and VLCS datasets. In PACS, we first note that GroupDRO a domain-aware method does not give strong benefits compared to ERM whereas our proposed method ClusterDG-Reg which re-labels domains gives an improvement of around 2.2% over vanilla GroupDRO and 3.6% over ERM . Similarly in the case of VLCS, we see a boost of 3.1% compared to GroupDRO and around 2% compared to ERM. It is evident from both the cases that ClusterDG-Reg provides non-trivial gains over GroupDRO, a domain aware approach. These experiments provide empirical evidence to our hypothesis that re-grouping domains along with better regularization leads to better generalizable solutions. For comparison, we include GroupDRO-random which creates K random groups (*i.e.*, domains) from data and trains GroupDRO model.

5 Conclusion

We explored the important link between the role pre-defined domain labels play in the success of domain-aware MDG methods. We proposed ClusterDG-Reg which includes a domain re-labeling step along with training a domain-aware approach such as GroupDRO. We conducted experiments on two popular MDG benchmarks and our results show improvements over both a domain agnostic approach like ERM and an domain-wise approach such as GroupDRO. Our results support our hypothesis that domain-aware approaches benefit by re-labeling the data rather than using pre-defined human-annotated domain labels.

[2] `https://github.com/facebookresearch/DomainBed`

[3] `https://scikit-learn.org/`

Acknowledgements

This work was performed under the auspices of the U.S. Department of Energy by the Lawrence Livermore National Laboratory under Contract No. DE-AC52-07NA27344, Lawrence Livermore National Security, LLC.

References

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

[2] Gilles Blanchard, Gyemin Lee, and Clayton Scott. Generalizing from several related classification tasks to a new unlabeled sample. *Advances in neural information processing systems*, 24:2178–2186, 2011.

[3] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 132–149, 2018.

[4] Prithvijit Chattopadhyay, Yogesh Balaji, and Judy Hoffman. Learning to balance specificity and invariance for in and out of domain generalization. In *European Conference in Computer Vision (ECCV)*, 2020.

[5] Chen Fang, Ye Xu, and Daniel N Rockmore. 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*, pages 1657–1664, 2013.

[6] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The journal of machine learning research*, 17(1):2096–2030, 2016.

[7] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[9] Diederik P Kingma and Jimmy Ba. Adam: A methodfor stochastic optimization. In *International Conference onLearning Representations (ICLR)*, 2015.

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

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

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

[13] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3):211–252, 2015.

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

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

[16] Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *CVPR 2011*, pages 1521–1528. IEEE, 2011.

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

[18] Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, and Tao Qin. Generalizing to unseen domains: A survey on domain generalization. *CoRR*, abs/2103.03097, 2021.

[19] Kaiyang Zhou, Yongxin Yang, Yu Qiao, and Tao Xiang. Domain generalization with mixstyle. In *International Conference on Learning Representations*, 2021.