A principled approach to model validation in domain generalization

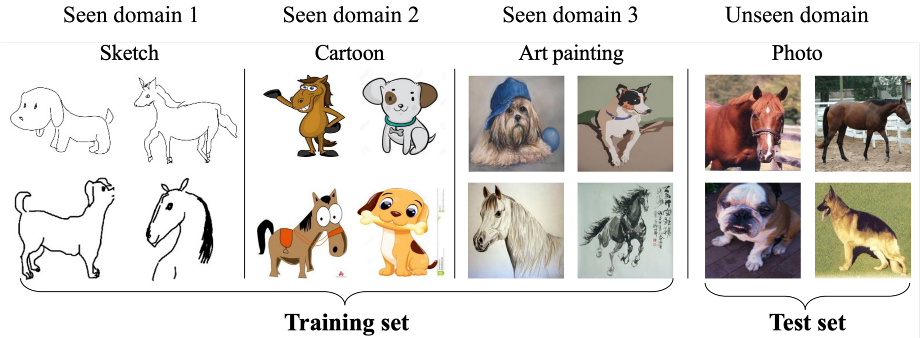
Boyang Lyu*, Thuan Nguyen*, Matthias Scheutz*, Prakash Ishwar**, and Shuchin Aeron*

*Tufts University, **Boston University

May 7, 2023

Domain Generalization (DG)

- **Given:** labeled samples from several "*seen*" domains
- **Goal:** learn a classifier that can generalize well to "*unseen*" domains
- **Challenge:** *distribution-shift*

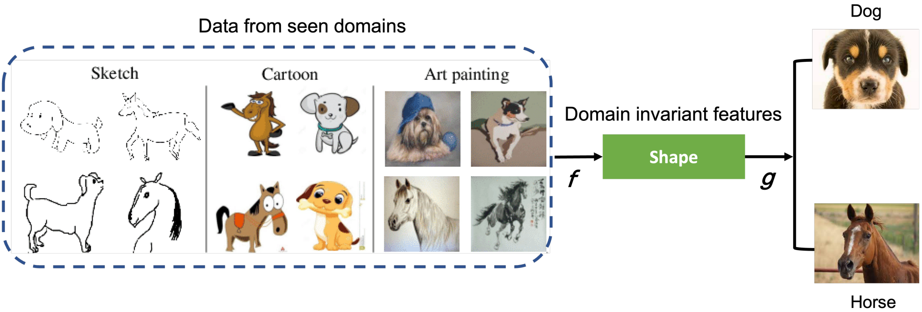


DG Methodology

Key idea:

- learn *domain-invariant* features via a representation function: $f : \mathcal{X} \to \mathcal{Z}$
- jointly train a classifier $g : \mathcal{Z} \to \mathcal{Y}$ to minimize DG loss[1]:

$$\mathrm{L}_{\text{Training}}(f, g) = \beta \, \mathrm{L}_{\text{Classification}}(f, g) + \mathrm{L}_{\text{Domain-discrepancy}}(f) \tag{1}$$



[1] Ben-David, Shai, et al. "A theory of learning from different domains," in Machine learning 79 (2010): 151-175.

DG Methodology

1. Classification risk:

$$\mathrm{L}_{\text{Classification}}(f, g) = \mathbb{E}_{(\boldsymbol{x},y)\sim p^{(s)}(\boldsymbol{x},y)}\left[\ell(g(f(\boldsymbol{x})), y)\right] \tag{2}$$

- $p^{(s)}(\boldsymbol{x}, y)$: seen-domain joint distribution in *input space*
- $\ell(\cdot, \cdot)$: classification loss function

2. Domain discrepancy:

$$\mathrm{L}_{\text{Domain-discrepancy}}(f) = d\left(p^{(u)}(f(\boldsymbol{x}), y)||p^{(s)}(f(\boldsymbol{x}), y)\right) \tag{3}$$

- $p^{(u)}(f(\boldsymbol{x}), y)$, $p^{(s)}(f(\boldsymbol{x}), y)$: joint distributions in *representation-space*
- $d(\cdot||\cdot)$: discrepancy measure

Fundamental Trade-off between Classification Risk and Domain Discrepancy

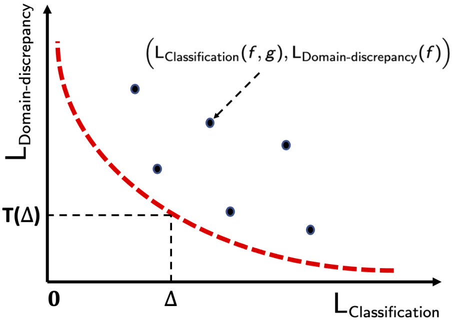
Theorem

Let

$$T(\Delta) := \min_{f:\mathcal{X}\to\mathcal{Z}} \mathsf{L}_{\text{Domain-discrepancy}}(f) \qquad (4)$$
$$s.t. \quad \mathsf{L}_{\text{Classification}}(f, g) \leq \Delta$$

If $d(a||b)$ is a convex function of (a, b), then for any classifier g, $T(\Delta)$ is

1. *non-increasing and*
2. *convex*



Training, Validation, and Testing in DG

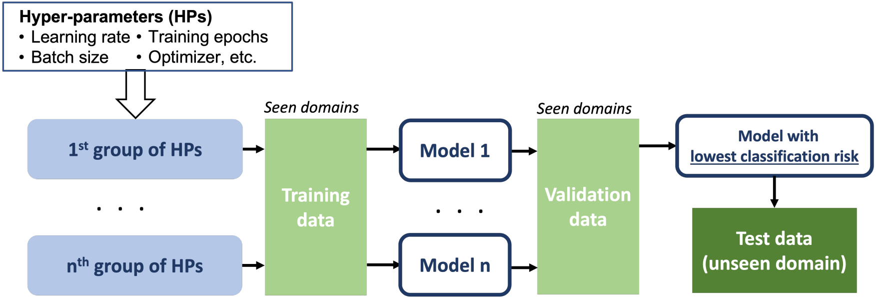
- **Training:** minimize training loss over the training set:

$$\mathrm{L}_{\text{Training}}(f, g) = \beta\, \mathrm{L}_{\text{Classification}}(f, g) + \mathrm{L}_{\text{Domain-discrepancy}}(f), \tag{5}$$

  - β controls the trade-off between classification risk and domain discrepancy

- **Validation:** select hyper-parameters (models) that *only* minimize classification risk on validation set (domain discrepancy ignored)

$$\mathrm{L}_{\text{Validation}}(f, g) = \mathrm{L}_{\text{Classification}}(f, g) \tag{6}$$



A New Model Validation Method in DG

1. *New* validation loss:

$$\mathsf{L}_{\text{Validation}} = \beta(1-\alpha)\mathsf{L}_{\text{Classification}} + \alpha \underbrace{\mathsf{L}_{\text{Domain-discrepancy}}}_{\text{New criterion}} \tag{7}$$

   - α: convex combination hyper-parameter
   - β: scale hyper-parameter for combining objectives with different scales

2. Practical implementation:
   - Cross-entropy to approximate $\mathsf{L}_{\text{Classification}}$
   - Maximum Mean Discrepancy loss[2] to approximate $\mathsf{L}_{\text{Domain-discrepancy}}$
   - $\beta = 1$, and $\alpha = 0.2$

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

Experimental Results

1. Datasets: PACS, VLCS, and C-MNIST[3]
2. Algorithms: 12 SOTA DG algorithms in DomainBed package[4]
3. Performance metric: Classification accuracy (percentage).

Datasets	**Algorithms**												**Wins**
	Fish	**IRM**	**GDRO**	**Mixup**	**CORAL**	**MMD**	**DANN**	**CDANN**	**MTL**	**VREx**	**RSC**	**SagNet**	
PACS (Traditional)	**84.6**	84.9	84.2	83.3	**85.1**	83.6	84.6	**86.4**	83.0	**84.5**	**85.2**	83.7	
PACS (Ours)	82.0	**85.3**	**84.3**	**85.3**	84.9	**85.0**	**84.9**	82.0	**84.2**	84.2	81.3	**85.1**	7/12
VLCS (Traditional)	**79.4**	76.0	78.1	77.4	76.8	**78.5**	77.8	79.2	77.3	76.4	**78.6**	**80.5**	
VLCS (Ours)	77.5	**79.2**	**79.6**	**77.6**	**78.8**	78.0	**78.5**	**80.3**	**78.2**	**78.6**	76.1	79.3	8/12
CMNIST (Traditional)	**10.0**	10.0	10.2	**10.4**	9.7	**10.4**	10.0	9.9	10.5	10.2	10.2	10.4	
CMNIST (Ours)	9.7	**10.9**	**12.6**	10.3	**11.2**	9.9	**11.1**	**10.2**	**11.5**	**15.6**	**13.8**	**10.5**	9/12

[3] https://github.com/facebookresearch/DomainBed

[4] Gulrajani, Ishaan, and David Lopez-Paz. "In search of lost domain generalization," in International Conference on Learning Representations (ICLR), 2021.

Thank you for your attention!

?