Domain Generalization using Causal Matching

Divyat Mahajan [1] Shruti Tople [2] Amit Sharma [1]

[1]Microsoft Research, India [2]Microsoft Research, UK

Paper: arxiv/2006.07500 **Code:** github/microsoft/robustdg

Domain Generalization: Introduction

- **Aim:** Learn a single classifier (f) with training data (X, Y) sampled from m domains that generalizes well to data from unseen domains/distributions
- **Assumption:** There exist stable (causal) features (X_c) whose relationship with outcome Y, $P(Y|X_c)$, is invariant across domains
- **Notation:**
  - Representation network: $\Phi : \mathcal{X} \to \mathcal{C}$; Classification network: $h : \mathcal{C} \to \mathcal{Y}$.
  - Ideal solutions $h^*, \Phi^* = \arg\min_{h,\Phi} \mathbb{E}_{(d,x,y)}[l(y, h(\Phi(x)))]$ satisfy $x_c = \Phi^*(x)$ and $f^* = h^*(\mathbf{x}_c)$
- **Contributions:**
  - Identify conditions for failure of class-conditional invariance objective [1, 2]
  - Propose object-invariant condition for domain generalization, along with a novel approach to satisfy it in practical scenarios

Training Domains **Test Domains**

Rotation Angles: 15, 30, 45, 60, 76 Rotation Angles: 0, 90

Why Class-Conditional Domain Invariance Fails?

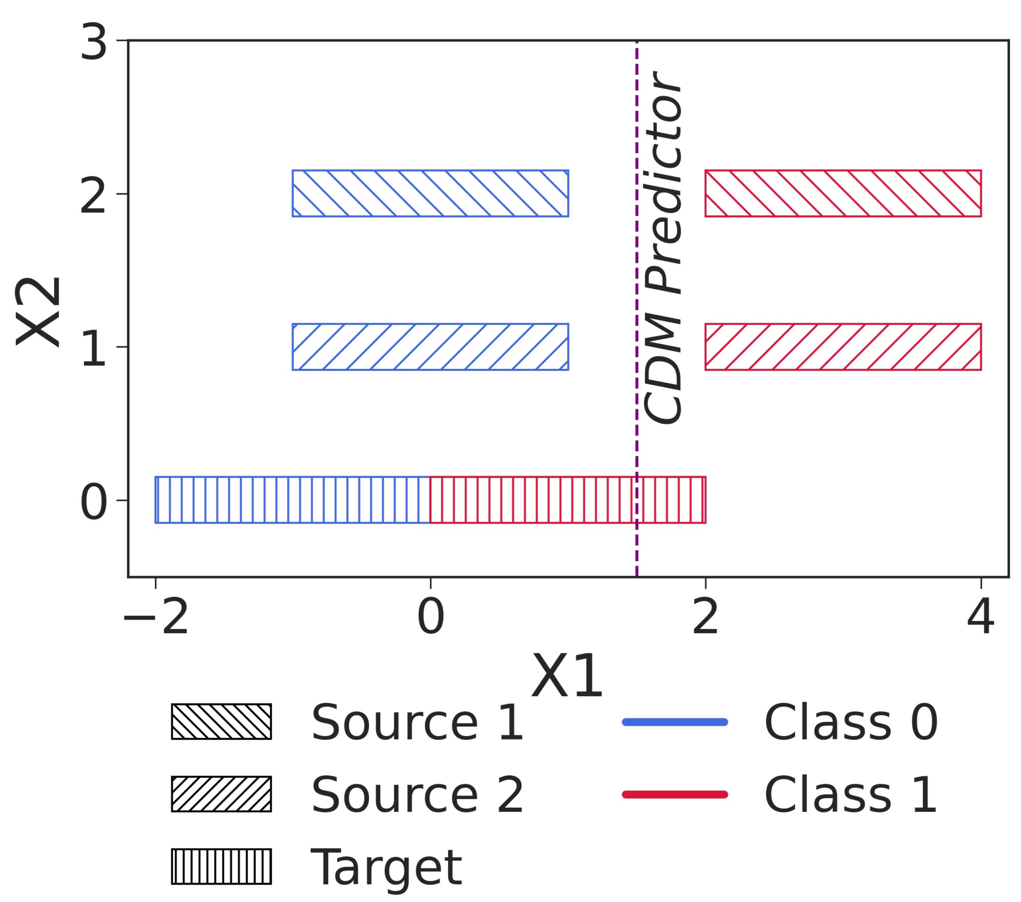
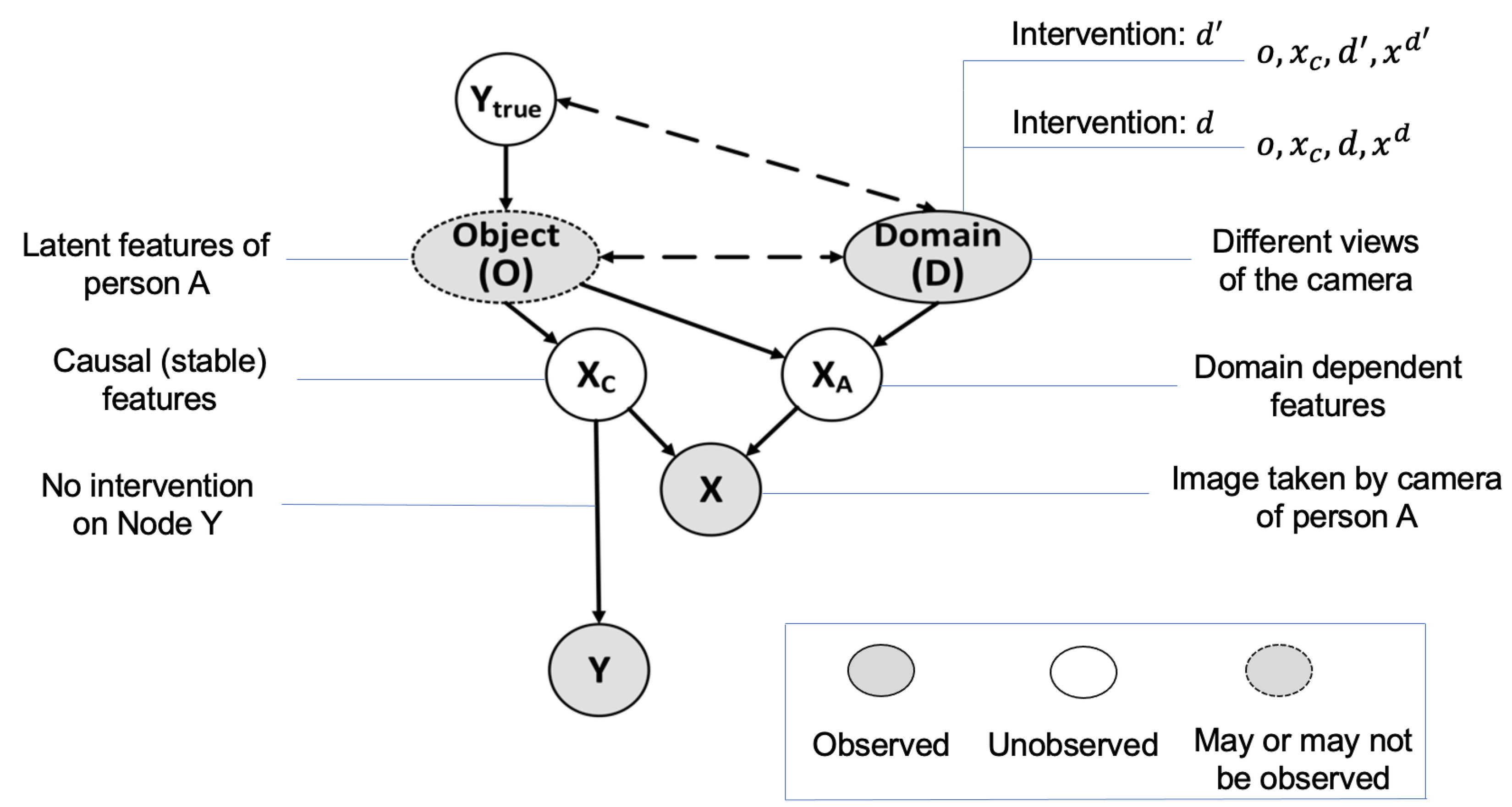


Figure: $x_1 = x_c + \alpha_d$, $x_2 = \alpha_d$ where x_c and α_d are unobserved

- **Class-Conditional Invariance**: [1, 2] The learnt representation $\phi(x)$ should satisfy $\phi(x) \perp\!\!\!\perp d|y$, which is satisfied by $\phi(x_1, x_2) = x_1$ for the example above
- **Invariant predictor not recovered:** The classifier built over $\phi(x_1, x_2) = x_1$ gets only 62.5 percent test accuracy
- **Intra-Class Variation:** The reason for failure of class-conditional invariance is due to varying conditional distribution of stable features, $p(x_c|y)$, across domains (refer to proposition 1 in paper for more details)
- In some datasets, class-conditional invariance can be satisfied by spurious features (refer to slab dataset in paper for more details)

Causal View of Domain Generalization



- **Domain as intervention:** For each observed x^d, there are a set of counterfactual inputs $x^{d'}$ where $d \neq d'$, but both have similar causal features x_c
- **Object-Invariant Condition:** $X_c \perp\!\!\!\perp D|O$
  - *Empirical:* $\sum_{\Omega(j,k)=1; d \neq d'} \text{dist}(\Phi(\mathbf{x}_j^{(d)}), \Phi(\mathbf{x}_k^{(d')})) = 0$; $\Omega = 1$ if $o_j^d = o_k^{d'}$, $\Omega = 0$ otherwise

Perfect Match: Proposed approach for known true objects

- **Loss:** Empirical Risk Minimization Loss + $\lambda \times$ (Object-Invariant Constraint)
- **Intuition:** Match counterfactuals (same base object pairs) instead of same class pairs to account for intra-class variability

MatchDG: Proposed approach for unkown true objects

- **Goal:** Learn a match function such that $\Omega(\mathbf{x}, \mathbf{x}') = 1$ when $Dist(\mathbf{x}_c, \mathbf{x}_c')$ is small
- **Assumption:** Same-class inputs are closer in true causal representation than different-classes inputs
- **Simple Baseline:** Use contrastive loss to learn a representation under which same-class inputs become close than different-class inputs
- **Our approach:** Contrastive Learning with iterative updates to positive matches to help in capturing intra-class variance across domains

Figure: Different line styles indicate different domains; different colors indicate different class labels; different shapes indicate different base objects

- **Execution of our approach:**
  - **Contrastive Loss:** With x^1 as anchor, Positive Match(x^1)= x^2 and Negative Match(x^1)= x^4, optimize: $\min_\phi\ Dist(\phi(x^1), \phi(x^2)) - Dist(\phi(x^1), \phi(x^4))$
  - **Iterative Update:** Update positive match for x^1: $min_i\ Dist(\phi(x^1, \phi(x^i)))\ \forall x^i \in d^2, y^1 = y^i$
  - **Updated Contrastive Loss:** Positive Match(x^1) updated to x^3 that shares the same base object as x^1; optimize contrastive loss with new positive matches

Results: OOD accuracy on DG benchmarks

Dataset	ERM	Best Prior Work	Rand Match	MatchDG	PerfMatch
Rot MNIST (5)	93.0	94.5	93.4	**95.1**	96.0
Rot MNIST (3)	76.2	77.7	78.3	**83.6**	89.7
Fashion MNIST (5)	77.9	78.7	77.0	**80.9**	81.6
Fashion MNIST (3)	36.1	37.8	38.4	**43.8**	54.0

Figure: Out of Domain Accuracy (OOD) Results: Brackets denote number of source domains for Rotated & Fashion MNIST

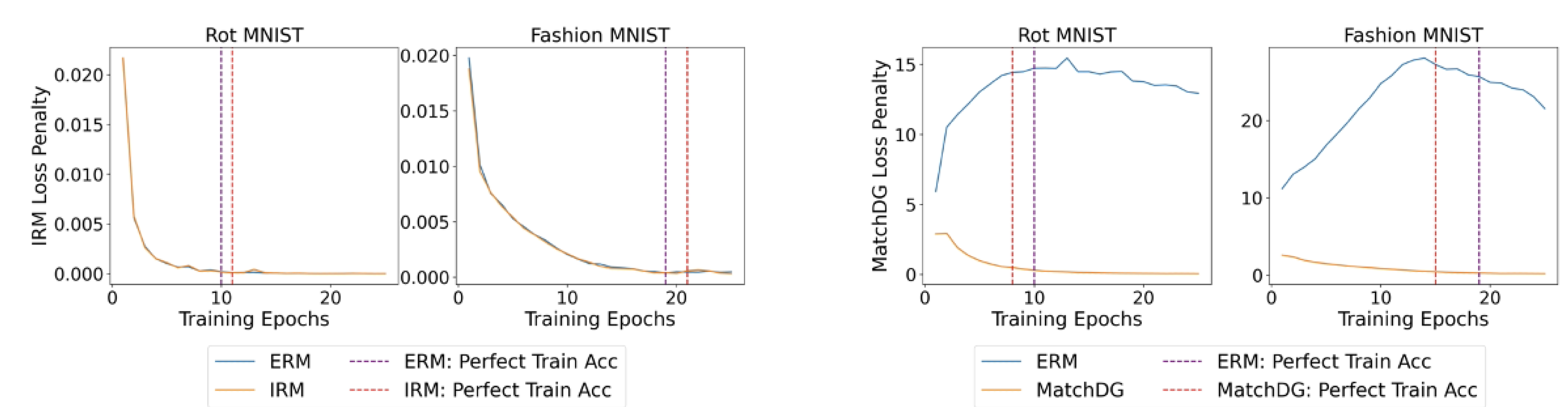
- MatchDG, Perfect Match obtain SOTA accuracy and improvement over baseline is highlighted in the case of fewer source domains
- MatchDG obtains comparable performance to SOTA approaches on more realistic benchmarks like PACS (refer to section 6.2 in paper for more details)

Does MatchDG learn the causal features?

Dataset	Method	Overlap (%)	Top 10 Overlap (%)	Mean Rank
Rotated MNIST	ERM	15.8	48.8	27.4
	MatchDG	**28.9**	**64.2**	**18.6**
Fashion MNIST	ERM	2.1	11.1	224.3
	MatchDG	**17.9**	**43.1**	**89.0**

Figure: Results for quality of match function using following metrics: Overlap of top-1 match with the true object match, Overlap of top-10 matches with the true object match, Mean rank of the true object match in the learnt representation (lower is better)

MatchDG works even under the zero training error regime



- Zero training error does not necessarily imply similar representations for each class, resulting in ERM unable to satisfy MatchDG penalty
- Methods based on comparing variation in loss across domains, like IRM [3], will be affected under zero training error

References

[1] Y. Li, X. Tian, M. Gong, Y. Liu, T. Liu, K. Zhang, and D. Tao, ``Deep domain generalization via conditional invariant adversarial networks,'' in *ECCV*, pp. 624--639, 2018.

[2] Y. Li, M. Gong, X. Tian, T. Liu, and D. Tao, ``Domain generalization via conditional invariant representations,'' in *AAAI*, 2018.

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