Domain Generalization using Causal Matching

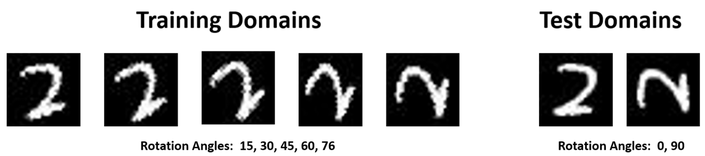
Divyat Mahajan [1] Shruti Tople [2] Amit Sharma [1]

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

Paper Github

Domain Generalization: Introduction

- Domain Generalization (DG) aims to learn a single classifier that generalizes well to data from unseen domains/ distributions
- **Training data**: $\{(d_i, \mathbf{x}_i, y_i)\}_{i=1}^n \sim (D_m, \mathcal{X}, \mathcal{Y})^n$ where $d_i \in D_m$ and $D_m \subset \mathcal{D}$ is a set of m domains
- **Covariate Shift Assumption**: $p_i(y|x) = p_j(y|x)$ for any two domains d_i, d_j
- We identify the right conditions to learn invariant representations for DG and propose a novel approach to satisfy them in practical scenarios



Causal View of Domain Generalization

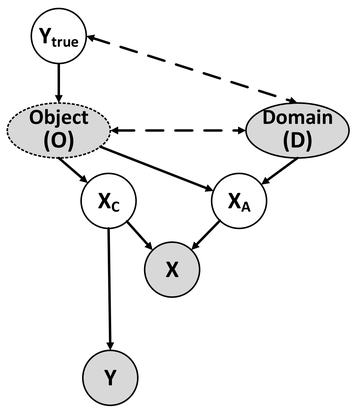


Figure 1:General SCM for DG task. Observed variables are shaded; dashed arrows denote correlated nodes. Object has a dotted outline since it may not be observed.

- Object (O) can be interpreted as the base person where the Domain (D) corresponds to different views that lead to creation of an image (X) for that person (O)
- Domains can be interpreted as interventions: For each observed x_i^d, there are a set of counterfactual inputs $x_i^{d'}$ where $d \neq d$, but both correspond to the (possibly unobserved) same object (O)

Invariance Condition from SCM

Constraints using d-separation on the SCM for DG

- $X_C \perp\!\!\!\perp D|O$ and $X_C \not\perp\!\!\!\perp O$ (Invariance Condition)
- $Y \perp\!\!\!\perp D|X_C$ (Generalizable Classifier based on Invariant Representation)

Identification of invariant representation X_C : `ERM-PerfMatch`

- In general X_C is unobserved and there could be multiple choices for X_C that might lead to sub optimal classifier (Eg: X_C as constant function)
- We propose the following loss function (`ERM-PerfMatch`) to recover optimal X_C (*Proof in the paper: Theorem 1*)

$$f_{\texttt{perfectmatch}} = \arg\min_{h,\Phi} \sum_{d=1}^{m} L_d(h(\Phi(X)), Y) + \lambda \sum_{\Omega(j,k)=1; d \neq d'} \text{dist}(\Phi(\mathbf{x}_j^{(d)}), \Phi(\mathbf{x}_k^{(d')})) \quad (1)$$

Comparison with prior works on invariant representation

- Domain Invariant Representation proposes $X_C \perp\!\!\!\perp D$
- Class Conditional Domain Invariant Representation proposes $X_C \perp\!\!\!\perp D|Y_{true}$

Both the conditions are incorrect due to the path through object (O) between X_C and D in the SCM for DG (*Proof in the paper: Corollary 1*)

Satisfying Invariance Condition with Observational Data

- Equation (1) relies on the object information (O) which is known in self-collected datasets only and a perfect counterfactual might not always exist in real-life datasets
- Goal is to learn a matching $\Omega : \mathcal{X} \times \mathcal{X} \to \{0, 1\}$ such that pairs having $\Omega(\mathbf{x}, \mathbf{x}') = 1$ have low difference in $\mathbf{x}_c$ and $\mathbf{x}'_c$.

Class Conditional Approximation: `ERM-RandMatch`

- Randomly match pairs across domains from the same class Ω_Y (*Proof in the paper: Theorem 2*)

$$f_{\texttt{randommatch}} = \arg\min_{h,\Phi} \sum_{d=1}^{m} L_d(h(\Phi(X)), Y) + \lambda \sum_{\Omega_Y(j,k)=1; d \neq d'} \text{dist}(\Phi(\mathbf{x}_j^{(d)}), \Phi(\mathbf{x}_k^{(d')})) \quad (2)$$

Contrastive Learning: `MatchDG`

- Optimise the following contrastive learning loss with positive matches as same class, different domains pairs and negative matches as different class pairs

$$l(\mathbf{x}_j, \mathbf{x}_k) = -\log \frac{\exp\left(\text{sim}(\Phi(\mathbf{x}_j), \Phi(\mathbf{x}_k))/\tau\right)}{\exp\left(\text{sim}(\Phi(\mathbf{x}_j), \Phi(\mathbf{x}_k))/\tau\right) + \sum_{i=0, y_i \neq y_j}^{b} \exp\left(\text{sim}(\Phi(\mathbf{x}_j), \Phi(\mathbf{x}_i))/\tau\right)} \quad (3)$$

`MatchDG`: Proposed Algorithm

- Learn the matching function Ω by minimizing loss in Eq (3) in Phase 1 and then minimize loss in Eq (2]) with the learn matching function
- In Phase 1, the positive matches initialized with random match (Ω_Y) and updated periodically based on nearest same-class pairs in the representation space

Algorithm 1: MatchDG

Input: Dataset $(d_i, x_i, y_i)_{i=1}^n$ from m domains, τ, t

Output: Function $f : \mathcal{X} \to \mathcal{Y}$

Create random match pairs Ω_Y.
Build a $n * m$ data matrix $\mathcal{M}$.
Phase I. while *notconverged* do
  for *batch* $\sim \mathcal{M}$ do
    Minimize contrastive loss (3).
  if *epoch* % t ==0 then
    Update match pairs using Φ_{epoch}.
Phase 2. Compute matching based on Φ. Minimize the loss (2) to obtain f.

Results

Table 1:Accuracy for Rotated MNIST & Fashion-MNIST datasets on target domains of 0° and 90°.

Dataset	Source	ERM	MASF	CSD	`ERM-RandMatch`	`MatchDG`	`ERM-PerfMatch`
Rotated MNIST	15, 30, 45, 60, 75	96.5 (0.15)	93 (0.2)	94.7 (0.2)	97.5 (0.17)	**97.5 (0.36)**	98.5 (0.08)
	30, 45, 60	80.6 (2.9)	69.4 (1.32)	**89.1 (0.004)**	82.8 (2.3)	88.9 (2.01)	93.6 (0.53)
	30, 45	64.0 (2.28)	60.8 (1.53)	77.2 (0.04)	69.7 (2.93)	**79.3 (4.2)**	84.2 (2.33)
Rotated Fashion MNIST	15, 30, 45, 60, 75	78.5 (1.15)	72.4 (2.9)	78.9 (0.7)	80.5 (0.97)	**83.5 (1.16)**	85.1 (0.97)
	30, 45, 60	33.9 (1.04)	25.7 (1.73)	27.8 (0.01)	35.5 (1.07)	**51.7 (2.08)**	61.04 (1.33)
	30, 45	21.85 (0.93)	20.8 (1.26)	20.2 (0.01)	23.9 (0.93)	**36.6 (2.17)**	42.0 (2.42)

Table 2:Accuracy results on the PACS dataset trained with Resnet-18.

Test Domain	ERM	MASF	JiGen	CSD	`ERM-RandMatch`	`MatchDG`
Photo	95.37 (0.37)	94.99 (0.09)	**96.03**	95.45	95.59 (0.07)	95.7 (0.52)
Art Painting	75.79 (1.38)	**80.29 (0.18)**	79.42	79.79	78.58 (0.56)	77.9 (0.61)
Cartoon	77.82 (0.39)	77.17 (0.08)	75.25	75.04	**80.02 (1.01)**	78.8 (0.29)
Sketch	69.75 (0.58)	71.69 (0.22)	71.35	72.46	76.03 (1.34)	**76.3 (0.92)**
Average	79.69	81.04	80.41	80.69	**82.56**	82.03