Domain Generalization using Causal Matching

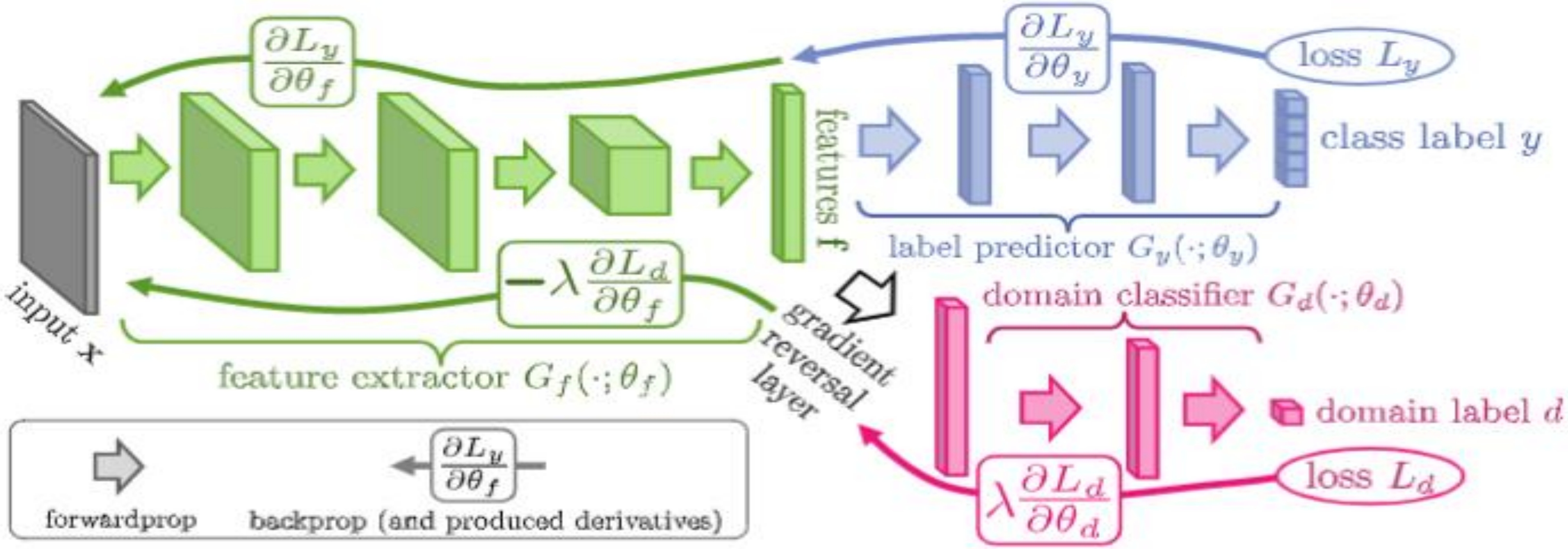
Divyat Mahajan, Shruti Tople, Amit Sharma

Microsoft Research

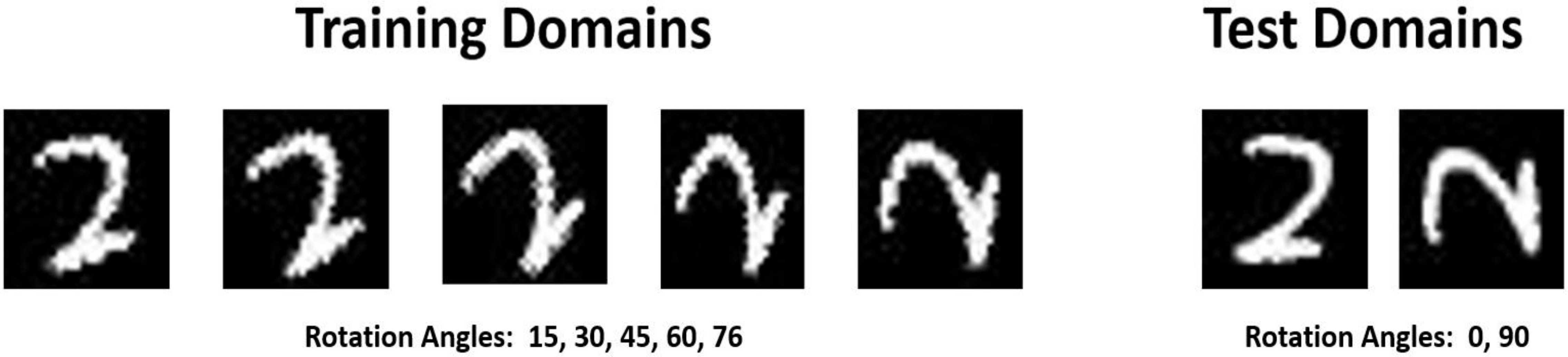
Feature Distribution Matching

- Domain Invariant Representations
  - Bad if labels and domain are correlated (Class Imbalance)
- Class Conditional Version
  - But does the distribution of invariant features need to be the same across domains?
  - Variance in the distribution due to different noise levels across domains

Domain Adversarial Training (Ganin et al.)



Perfect Match

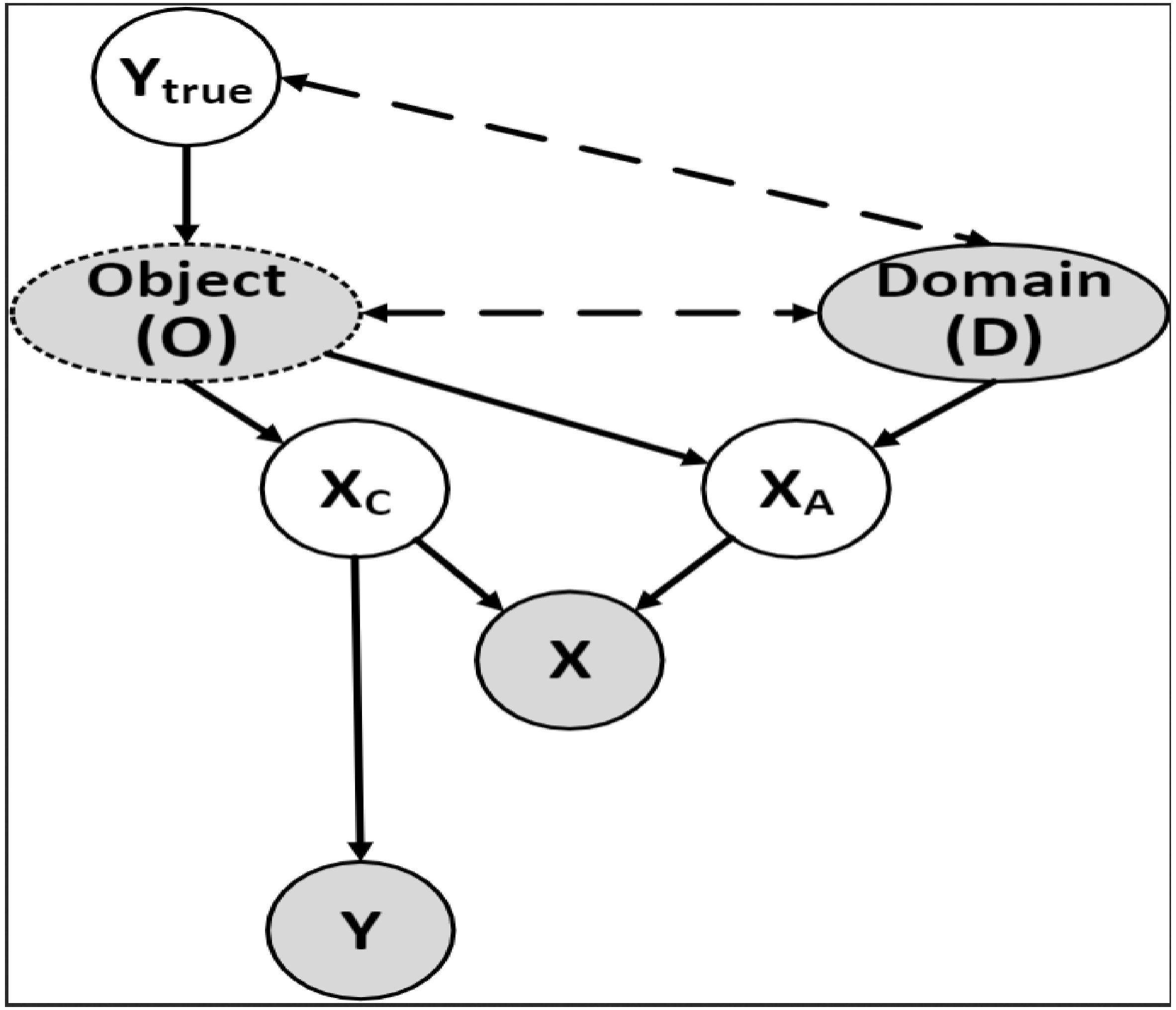


- Perfect Match:
    - Same data point rotated by different angle across domains shares the same invariant feature
    - Match feature representations for the “counterfactuals” of each data point across domains

Causal View of Domain Generalization

- Object (0) 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 area set of counterfactual inputs $x_i^{d\prime}$ where $d \neq d'$, but both correspond to the (possibly unobserved) same object (O)



Invariance Condition from SCM

- Invariance Condition: $X_C \perp\!\!\!\perp D \mid O$
- Perfect Match:

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

- Prior work incorrectness:
  - Domain-invariant representations: $X_C \perp\!\!\!\perp D$
  - Class-conditional domain-invariant: $X_C \perp\!\!\!\perp D \mid Y_{true}$
  - Both incorrect due to backdoor path via Object O

Observational Data

- Latent base object not known generally in observational data (PACS, VLCS)
  - Perfect Match still applicable using self augmentations

- Class Conditional Approximation:
  - Data points with the same class label are likely to closer under causal features as compared to point with different class labels

- Inferring latent base objects / match function ($\Omega : X \times X \rightarrow \{0,1\}$)
  - Contrastive Loss: $Dist(Anchor, Positive\ Match) - Dist(Anchor, Negative\ Match)$

- Iterative Contrastive Learning:
  - Initialize Ω with Random Match across domains with same class label
  - Using Ω to infer Positive Match given anchor and minimize contrastive loss
  - Update Ω based on nearest same-class pairs in the representation space

MatchDG

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

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

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.

Chest X Ray Dataset

Details: [Link]

- Source Domains (NIH, ChexPert)
  - Images with class label 0 are translated vertically downwards
- Target Domains (Kaggle)
  - No spurious correlation

	NIH (Source)	Chex (Source)	RSNA (Target)
ERM	78.9 (0.34)	**84.3** (3.52)	55.2 (2.27)
IRM	**79.1** (1.01)	83.4 (2.42)	56.6 (2.04)
CSD	73.2 (3.35)	83.3 (2.03)	60.5 (0.82)
RandMatch	75.3 (1.87)	83.6 (1.84)	57.4 (1.76)
MatchDG	74.7 (0.66)	82.2 (0.68)	58.4 (0.62)
MDGHybrid	74.3 (0.91)	82.4 (1.03)	**62.6** (0.72)

Evaluation Issues with DG

- OOD accuracy evaluated on few test domains (PACS, VLCS)
  - No guarantees regarding performance on a large set of unseen domains
  - Evaluation metrics to capture the extent to which DG algorithm learnt stable features

- Membership Inference (MI) Attacks
  - Utilize overfitting of ML models to predict train vs test dataset samples
  - Stable features $\rightarrow$ Better Generalization $\rightarrow$ Good Defense against MI attacks

- Connections between DG and Privacy Attacks
  - Theoretical: Causal models (stable feature learning) leads to better defense on MI attacks
  - Empirical: Use MI attacks to evaluate DG algorithms
  - Software: Toolkit to support DG algorithms and evaluate them on various privacy attacks