Exploiting Causal Chains for Domain Generalization

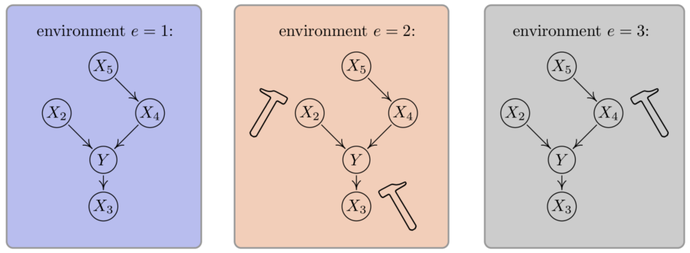
Olawale Salaudeen, Oluwasanmi Koyejo
Department of Computer Science, College of Engineering, University of Illinois at Urbana-Champaign

Background

We often encounter distribution shifts from train to test time; how can one learn a predictor that generalizes well to test distributions in this setting?

Invariant Causal Predictors

Assume that datasets belong to distinct domains identified by distinct interventions on the same shared causal mechanisms, resulting in different distributions



Invariant Mechanisms: Across domains, $\mu(Y \mid \text{parents}(Y))$ does not change, though the data distribution can be arbitrarily different. Consequently, identifying parents(Y) is sufficient to learn a predictor that is robust to distribution changes.

Representations that induce Invariant Predictors

When features are latent, a similarly strategy is to learn an embedding function Φ that induces invariance across training distributions, i.e.,

$$\mu(Y \mid \Phi(X)).$$

Invariant Risk Minimization (IRM): Learn a representation Φ such that the optimal predictor w, across domains $\mathcal{E}_{tr}$, is the same:

$$\min_{\substack{\Phi:\mathcal{X}\to\mathcal{H} \\ w:\mathcal{H}\to\mathcal{Y}}} \sum_{e\in\mathcal{E}_{\text{tr}}} R^e(w\circ\Phi)$$
$$\text{subject to} \quad w \in \arg\min_{\bar{w}:\mathcal{H}\to\mathcal{Y}} R^e(\bar{w}\circ\Phi), \text{ for all } e \in \mathcal{E}_{\text{tr}}$$

Let w^* be the optimal invariant predictor that we aim to identify.

The Risks of Invariant Risk Minimization: Let E be the number of distinct training domains and d_e the dimensionality of non-invariant features):

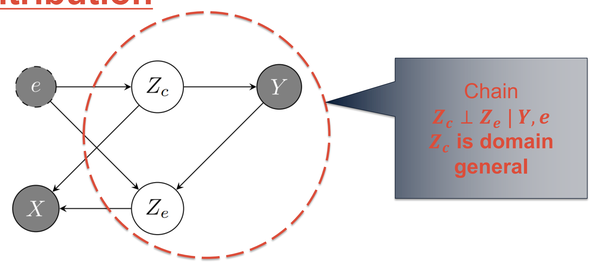
Linear case

- If $E \leq d_e$: There exists a feasible linear Φ which depends on non-invariant features and obtains a lower training risk than the optimal invariant predictor w^*
- If $E > d_e$: The optimal invariant predictor w^* achieves the lowest training risk

Nonlinear case

- IRM behaves just like ERM at test time

Contribution



Shaded nodes are observed while others are latent. e indicates the domain and X a function of the two latents Z_c, Z_e.

Under this generative process in the graph above, we propose to enforce a different Markov property than ICP, namely

$$Z_c \perp Z_e \mid Y, e. (TCRI)$$

$\perp$ means indicates independence.

We empirically show that under the chain generative model, the target conditioned representation invariance (TCRI) constraint yields a predictor that generalizes better than ERM and IRM to test distributions.

Target Conditioned Representation Invariance (TCRI)

Define two feature embedding functions Φ, Ψ – domain general and domain specific, respectively. These two embeddings are related by TCRI:

$$\Phi(X) \perp \Psi(X) \mid Y, e.$$

$$\min_{\Phi,\Psi,\theta_c,\theta_1,\theta_2,\ldots,\theta_E} \sum_{e\in\mathcal{E}} \Big[\underbrace{\mathcal{R}^e(\theta_c\circ\Phi)}_{(1)} + \beta\underbrace{\mathcal{R}^e(\theta_e\circ\Psi)}_{(2)} + \rho\underbrace{\widehat{TCRI}(\Phi^e,\Psi^e)}_{(3)}\Big]$$

(1) θ_c – optimal linear predictor on Φ across environments.

(2) $\theta_1, \theta_2, \ldots, \theta_E$ – optimal linear predictors on Ψ for each environment.

(3) The TCRI captures the constraint $\Phi(X) \perp \Psi(X) \mid Y \; \forall \, e$.

One option is the V-statistic-based Hilbert-Schmidt Independence Criterion (HSIC) estimate:

$$\widehat{HSIC}(X,Y) = \frac{1}{n^2}\text{trace}(\boldsymbol{K}_{XX'}\boldsymbol{H}_n\boldsymbol{K}_{YY'}\boldsymbol{H}_n),$$

where $\boldsymbol{K}_{XX'}, \boldsymbol{K}_{YY'} \in R^{n\times n}$ are Gram matrices, $\boldsymbol{K}^{\boldsymbol{i,j}}_{XX'} = \phi(X_i, X_j)$, $\boldsymbol{K}^{\boldsymbol{i,j}}_{YY'} = \phi(X_i, X_j)$, $\boldsymbol{H}_n = \frac{1}{n}\boldsymbol{I}_n\boldsymbol{I}_n'$ is a centering matrix, $\boldsymbol{I}_n$ is the $n\times$n dimensional identity matrix.

Another is the norm of the conditional cross-covariance:

$$\Sigma_{X_\Phi X_\Psi \mid Y} = \Sigma_{X_\Phi X_\Psi} - \Sigma_{X_\Phi Y}\Sigma_{YY}^{-1}\Sigma_{YX_\Psi}.$$

Experiments

We evaluate the following linear-Gaussian structural equation model (SEM):

$$\mathcal{SCM}(e) = \begin{cases} z_c^e \sim \mathcal{N}\left(0, (\sigma_c^2)^e I_{d_c}\right) \\ y^e = z_c^e\alpha + \varepsilon \\ z_e^e = y^e\gamma + \eta \end{cases},$$

where d_c, d_e are the dimensions of z_c, z_e, respectively, $\varepsilon \sim \mathcal{N}(0, (\sigma_\varepsilon^2)^e)$, and $\eta \sim \mathcal{N}\left(0, \left(\sigma_\eta^2\right)^e I_{d_e}\right)$.

$$\Phi, \Psi\colon \mathbb{R}^{d_c+d_e} \to \mathbb{R}, \qquad \theta_i\colon \mathbb{R} \to \mathbb{R} \; \forall \, i.$$

- Loss function: Mean Squared Error; TCRI: HSIC
- $\theta_c = 1.0$, is a *dummy* predictor; x^e = concatenation of z_c^e, z_e^e.

Results

Below are relative mean squared errors, $\frac{\text{Algorithm}}{\text{ERM}}$. Errors are computed for each domain independently – average is across all test environment errors and worst case is the worst error achieved on a distinct test domain.

Algorithm	Average		Worst Case	
	Train	Test	Train	Test
ERM	baseline			
IRM	1.08	1.24	1.92	2.16
TCRI (ours)	1.38	1.20	0.16	0.11
Causal (Oracle)	1.29	1.13	0.11	0.06

Summary of Results

- ERM performs best on average across train and test domains
  - Does not consider how large a distinct domain's error can be
  - Utilizing non-general features yields lower errors during training
- IRM is worse than ERM in this setting
  - We observe that IRM relies on domain-specific features more than ERM
- TCRI outperforms ERM and IRM in the worst case

Ongoing Work

- Experiments on real-world datasets using TCRI
  - Comparison to more SOA models on benchmark datasets

ILLINOIS