A Constrained Bayesian Approach to Out-of-Distribution Prediction

Ziyu Wang*, Binjie Yuan*, Jiaxun Lu, Bowen Ding, Yunfeng Shao, Qibin Wu, Jun Zhu#

OOD prediction and its hardness

Given data from different **environments**:

$$\mathcal{D}_{tr} := \{\{(x_i^{inv}, x_i^{spu}, y_i) \sim P_e\}_{i=1}^{n} : e \in \mathcal{E}_{tr}\}$$

find a predictor for a test env $e_* \notin \mathcal{E}_{tr}$

- x^{inv} induces invariant $p(y \mid x^{inv})$ across envs; defines an *invariant predictor*
- x^{spu} induces spurious correlations and hinders generalization of ERM

With a sufficiently large $m = |\mathcal{E}_{tr}|$ we can learn the best invariant predictor using e.g., Group DRO:

$$\hat{h}_{GDRO} := \arg\min_{h \in \mathcal{H}} \arg\max_{e \in \mathcal{E}_{tr}} \hat{R}_n(P_e, h)$$

With a **smaller** m, GDRO, IRM, etc. may all fail

- E.g., $m \lesssim \dim \mathbf{x}$ for some linear problems

Adaptation using labeled test samples can be necessary

A constrained posterior for adaptation

Assume known[1] lower bound of invariant predictor performance (e.g., accuracy $\geq \rho = 0.95$); define

$$P_{CB}(d\theta \mid \mathcal{D}_{tr}, \mathcal{D}_*) \propto \pi(d\theta) \underbrace{\mathbf{1}\{\max_{e \in \mathcal{E}_{tr}} \hat{R}_n(P_e, h_\theta) \geq \rho\}}_{\textit{relaxed}\text{ GDRO constraints}} \underbrace{p(\mathcal{D}_* \mid \theta)}_{\text{test likelihood / general exp. loss}}$$

Approx. inference with **LMC + line search**

Avoids a pathology of "standard"/scaled posterior,

$$\tilde{P}_\alpha(\theta \mid \mathcal{D}_{tr}, \mathcal{D}_*) \propto \pi(d\theta) p^\alpha(\mathcal{D}_{tr} \mid \theta) p(\mathcal{D}_* \mid \theta)$$

$\alpha \ll 1 \Rightarrow \mathcal{D}_{tr}$ not efficiently utilized; $\alpha = 1 \Rightarrow \tilde{P}$ can fail just like ERM

Ex. training-time $R_e(h_{inv}) \equiv 1\%$, $R_e(h_{spu}) \equiv 0\%$, $n = 10^5$; test $R_*(h_{inv}) = 0\%$, $R_*(h_{spu}) = 100\% \Rightarrow \tilde{P}_1$ requires $n_* \gtrsim 10^3$ samples to switch to h_{inv} from h_{spu}; P_{CB} only requires $n_* = O(1)$

[1]: without such knowledge we can still set ρ based on ERM to trade-off between in-dist and OOD performance; see paper for discussion and PACS results in this scenario [2]: this includes methods tested in the DomainBed benchmark (for PACS and ColorMNIST), and IRM, DGRO and DANN for the real-world problem [3]: defined as the 20% percentile of accuracy across replications, for the worst train/test env split

With **only a few** training environments, don't use them to learn an invariant predictor.

Adapt to environment shift by using them to define **constraints.**

Paper

Code

Analysis: improved convergence of *a* constrained estimator

Setup. linear-Gaussian model

$$\bar{\beta}^e_{spu} \sim \mathcal{N}(0, d_{spu}^{-1} I), \; x_i^e = M \begin{bmatrix} x^e_{inv,i} \\ x^e_{spu,i} \end{bmatrix} \sim \mathcal{N}(0, I),$$
$$y_i^e \sim \mathcal{N}(\bar{\beta}_{inv}^\top x^e_{inv,i} + (\bar{\beta}^e_{spu})^\top x^e_{spu,i}, \sigma_y^2)$$

- $\bar{\beta}_{inv}$ arbitrary & fixed vector with norm $O(1)$; test-time $\bar{\beta}$ arbitrary & fixed
- Nontrivial problem: there exists $\theta_{non-inv}$ s.t. $R_e(\theta_{inv}) - R_e(\theta_{non-inv}) \gtrsim m^{-1} \; \forall e \in \mathcal{E}_{tr}$
- Analyzes a constrained *point estimator*: $\hat{\theta} := \arg\max_{\theta \in \mathcal{C}_{tr}} p(\mathcal{D}_* \mid \theta)$ where $\mathcal{C}_{tr}$ is the constraint set in P_{CB}

Takeaway. $\tilde{\theta}$ outperforms both ERM/GDRO and the unconstrained posterior ($\tilde{P}_0$) when $n_* \asymp d_{spu}, 1 \ll m \ll d_{spu}$

(See paper for full results and discussion.)

Experiments

Setup. synthetic, benchmark (modified ColorMNIST, PACS) and real-world classification tasks; $m \in \{3, 4\}$, $n \in [10^3, 10^5]$

Baselines. ERM (no adaptation); unconstrained/scaled/"standard" posterior $\tilde{P}_\alpha$; DivDis (Lee et al, 2023)

Results.

- Test-time adaptation significantly improves over ERM on datasets where strong domain generalization baselines[2] do not
- Our method is the only one that consistently achieves near-the-top performance

PACS: avg. accuracy / perf. estimate for unfavorable conditions[3]

<table>
<tr><th>n_*</th><th>0 (ERM)</th><th>16</th><th>256</th></tr>
<tr><td>$\tilde{P}_0$</td><td rowspan="4">83.2/72.6</td><td>83.8/70.1</td><td>89.4/80.6</td></tr>
<tr><td>$\tilde{P}_1$</td><td>85.0/76.1</td><td>87.1/77.2</td></tr>
<tr><td>DivDis</td><td>85.0/**77.6**</td><td>85.0/76.9</td></tr>
<tr><td>Ours</td><td>**86.4/77.6**</td><td>**90.3/83.7**</td></tr>
</table>

Real-world task: avg./unfavorable accuracy. *: α selected using additional test data.

<table>
<tr><th>n_*</th><th>0 (ERM)</th><th>20</th><th>80</th></tr>
<tr><td>$\tilde{P}_0$</td><td rowspan="3">85.0/81.9</td><td>87.3/82.4</td><td>92.0/90.0</td></tr>
<tr><td>$\tilde{P}_\alpha^*$</td><td>**89.3/85.4**</td><td>92.7/90.5</td></tr>
<tr><td>Ours</td><td>89.0/85.1</td><td>**92.8/91.3**</td></tr>
</table>

(See paper for full results and additional experiments.)