Linear unit tests for invariance discovery

Benjamin Aubin[†], Agnieszka Słowik[†], Martin Arjovsky[‡], Leon Bottou[†], David Lopez-Paz[†]

[†] Facebook AI Research

[‡] INRIA - PSL Research University

Abstract

There is an increasing interest in algorithms to learn **invariant correlations across training environments**. A big share of the current proposals finds theoretical support in the causality literature, but how useful are they in practice?

We propose **a benchmark of six linear unit tests** that can be used to evaluate the robustness to spurious correlations. Following initial experiments, none of the recently proposed invariant learning algorithms [1, 4, 3] pass all tests.

By providing the code to replicate our experiments, we hope that our unit tests become a standard stepping stone for researchers in out-of-distribution generalization.

`https://www.github.com/facebookresearch/InvarianceUnitTests`

Shared assumptions

We collect datasets $D_e = \{(x_i^e, y_i^e)\}_{i=1}^{n_e}$ containing n_e samples for n_{env} environments: $e \in \mathcal{E} = \{E_j\}_{j=0}^{n_{\text{env}}-1}$.

The input feature vector $x^e = (x^e_{\text{inv}}, x^e_{\text{spu}}) \in \mathbb{R}^d$ contains features $x^e_{\text{inv}} \in \mathbb{R}^{d_{\text{inv}}}$ that elicit invariant correlations and features $x^e_{\text{spu}} \in \mathbb{R}^{d_{\text{spu}}}$ that elicit spurious correlations such that $d = d_{\text{inv}} + d_{\text{spu}}$. The goal is to construct invariant predictors that estimate the target variable y^e by relying on x^e_{inv}, and ignoring x^e_{spu}.

To measure the extent to which an algorithm ignores the features x^e_{spu}, we sample a *train* split, a *validation* split, and a *test* split per problem and environment. In the test split, the features x^e_{spu} are shuffled at random across examples. This way, only those predictors ignoring x^e_{spu} will achieve minimal test error.

Ex1: Regression from causes and effects

A linear least-squares regression problem where features contain causes and effects of the target variable [1].

To construct the datasets D_e for every $e \in \mathcal{E}$ and $i = 1, \dots, n_e$, sample:

$$x^e_{\text{inv},i} \sim \mathcal{N}_{d_{\text{inv}}}(0, (\sigma^e)^2), \qquad x^e_i \leftarrow (x^e_{\text{inv},i}, x^e_{\text{spu},i}),$$
$$\tilde{y}^e_i \sim \mathcal{N}_{d_{\text{inv}}}(W_{yx}\, x^e_{\text{inv},i}, (\sigma^e)^2), \qquad y^e_i \leftarrow \frac{2}{d} \cdot 1^\top_{d_{\text{inv}}}\, \tilde{y}^e_i;$$
$$x^e_{\text{spu},i} \sim \mathcal{N}_{d_{\text{spu}}}(W_{xy}\, \tilde{y}^e_i, 1),$$

Ex2: Cows vs camels

In the spirit of [2, 1], we add a **binary classification problem to imitate the introductory example "most cows appear in grass and most camels appear in sand"**.

To construct the datasets D_e for every $e \in \mathcal{E}$ and $i = 1, \dots, n_e$, sample:

$$j^e_i \sim \text{Categorical}\left(p^e s^e, (1-p^e)s^e, p^e(1-s^e), (1-p^e)(1-s^e)\right);$$

$$x^e_{\text{inv},i} \sim \begin{cases} (\mathcal{N}_{d_{\text{inv}}}(0, 10^{-1}) + \mu_{\text{cow}}) \cdot \nu_{\text{animal}} & \text{if } j^e_i \in \{1,2\}, \\ (\mathcal{N}_{d_{\text{inv}}}(0, 10^{-1}) + \mu_{\text{camel}}) \cdot \nu_{\text{animal}} & \text{if } j^e_i \in \{3,4\}, \end{cases}$$

$$x^e_{\text{spu},i} \sim \begin{cases} (\mathcal{N}_{d_{\text{spu}}}(0, 10^{-1}) + \mu_{\text{grass}}) \cdot \nu_{\text{background}} & \text{if } j^e_i \in \{1,4\}, \\ (\mathcal{N}_{d_{\text{spu}}}(0, 10^{-1}) + \mu_{\text{sand}}) \cdot \nu_{\text{background}} & \text{if } j^e_i \in \{2,3\}, \end{cases}$$

$$x^e_i \leftarrow (x^e_{\text{inv},i}, x^e_{\text{spu},i}); \qquad y^e_i \leftarrow \begin{cases} 1 & \text{if } 1^\top_{d_{\text{inv}}} x^e_{i,\text{inv}} > 0, \\ 0 & \text{else}; \end{cases}$$

Ex3: Small invariant margin

Spiral binary classification: the first two dimensions offer an invariant small-margin decision boundary. The rest of the dimensions offer a changing large-margin decision boundary. Linear version of the spiral problem [4].

To construct the datasets D_e for every $e \in \mathcal{E}$ and $i = 1, \dots, n_e$, sample:

$$y^e_i \sim \text{Bernoulli}\left(\frac{1}{2}\right),$$

$$x^e_{\text{inv},i} \sim \begin{cases} \mathcal{N}_{d_{\text{inv}}}(+\gamma, 10^{-1}) & \text{if } y^e_i = 0, \\ \mathcal{N}_{d_{\text{inv}}}(-\gamma, 10^{-1}) & \text{if } y^e_i = 1; \end{cases} \qquad x^e_i \leftarrow (x^e_{\text{inv},i}, x^e_{\text{spu},i}).$$

$$x^e_{\text{spu},i} \sim \begin{cases} \mathcal{N}_{d_{\text{spu}}}(+\mu^e, 10^{-1}) & \text{if } y^e_i = 0, \\ \mathcal{N}_{d_{\text{spu}}}(-\mu^e, 10^{-1}) & \text{if } y^e_i = 1; \end{cases}$$

Please refer to our paper and codebase for a full list of parameters and their values.

Baseline results and analysis

We define three additional problems: **"scrambled" variations**. Scrambled variations build observed datasets $D^e = \{(S^\top x^e_i, y^e_i)\}_{i=1}^{n^e}$, where $S \in \mathbb{R}^{d \times d}$ is a random rotation matrix fixed for all environments $e \in \mathcal{E}$.

We evaluate ERM[5], IRM[1], IGA[3], AND-mask[4] on our six problems.

Oracle is a version of ERM where all data splits contain randomized x^e_{spu}, and therefore are trivial to ignore. The purpose of this method is to understand the achievable upper bound performance in our problems.

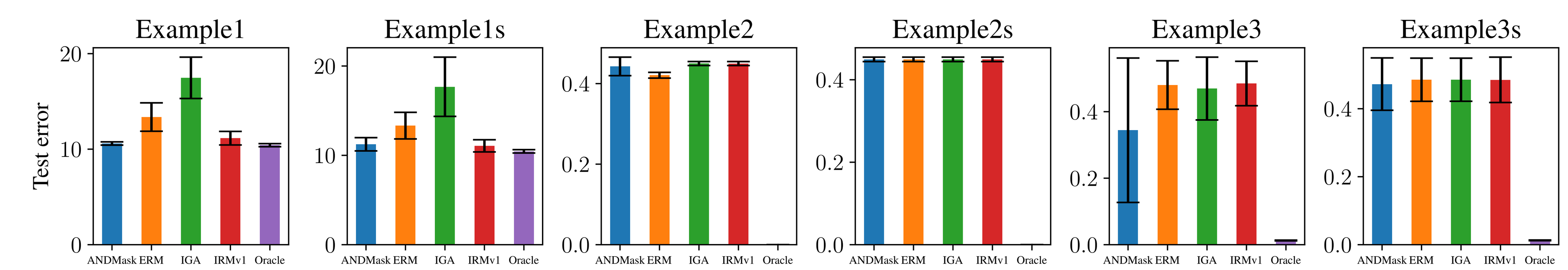

Fig. 1: Test error averaged across environments (E0, E1, E2) for $(d_{\text{inv}}, d_{\text{spu}}, n_{\text{env}}) = (5, 5, 3)$.

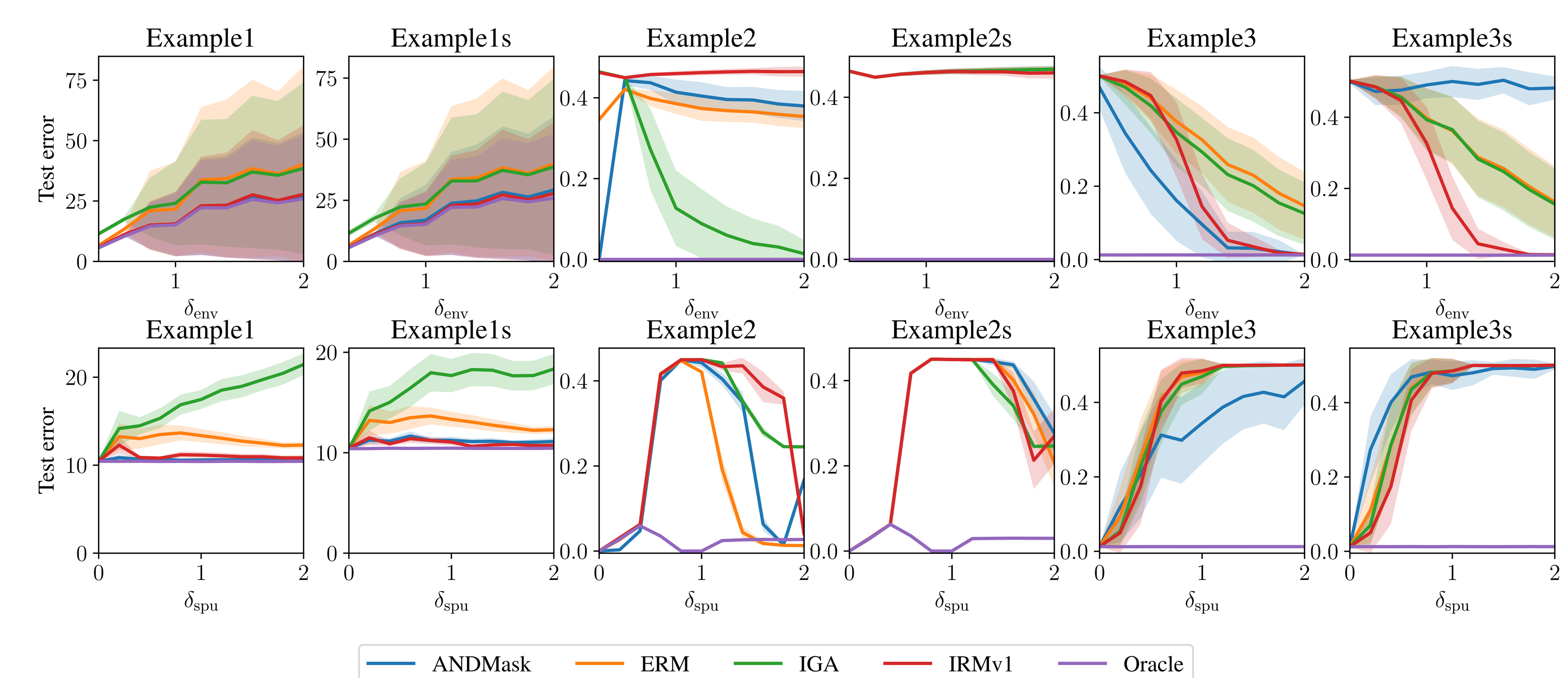

Fig. 2: Test error averaged across environments for ANDMask, ERM, IGA, IRMv1 and Oracle on the unit-tests as a function of the ratio $\delta_{\text{env}} = \frac{n_{\text{env}}}{d_{\text{spu}}}$ at fixed dimensions $(d_{\text{inv}}, d_{\text{spu}}) = (5, 5)$ (top) and as a function of $\delta_{\text{spu}} = \frac{d_{\text{spu}}}{d_{\text{inv}}}$ for $(d_{\text{inv}}, n_{\text{env}}) = (5, 3)$ (bottom).

References

[1] Martin Arjovsky et al. "Invariant risk minimization". In: *arXiv preprint arXiv:1907.02893* (2019).

[2] Sara Beery, Grant Van Horn, and Pietro Perona. "Recognition in terra incognita". In: *ECCV* (2018).

[3] Masanori Koyama and Shoichiro Yamaguchi. "Out-of-distribution generalization with maximal invariant predictor". In: *arXiv preprint arXiv:2008.01883* (2020).

[4] Giambattista Parascandolo et al. "Learning explanations that are hard to vary". In: *arXiv preprint arXiv:2009.00329* (2020).

[5] Vladimir Vapnik. "Statistical learning theory Wiley". In: *New York* (1998).