Learning to Pivot with Adversarial Networks #105

Gilles Louppe, Michael Kagan and Kyle Cranmer

✉ g.louppe@uliege.be, makagan@slac.stanford.edu, kyle.cranmer@nyu.edu

https://github.com/glouppe/paper-learning-to-pivot/

Abstract

Several techniques for domain adaptation have been proposed to account for differences in the distribution of the data used for training and testing. The majority of this work focuses on a binary domain label. Similar problems occur in a scientific context where there may be a continuous family of plausible data generation processes associated to the presence of systematic uncertainties. Robust inference is possible if it is based on a pivot – a quantity whose distribution does not depend on the unknown values of the nuisance parameters that parametrize this family of data generation processes. In this work, we introduce and derive theoretical results for a training procedure based on adversarial networks for enforcing the pivotal property (or, equivalently, fairness with respect to continuous attributes) on a predictive model. The method includes a hyperparameter to control the trade-off between accuracy and robustness. We demonstrate the effectiveness of this approach with a toy example and examples from particle physics.

Problem statement

Data generating process $p(X, Y, Z)$
$x \in \mathcal{X}$ are the data
$y \in \mathcal{Y}$ are the target labels
$z \in \mathcal{Z}$ are the nuisance parameters
continuous or categorical

Goal : Learn a classification/regression function f with parameters θ_f and minimizing a loss $\mathcal{L}_f(\theta_f)$ (e.g., the cross-entropy) such that

$$p(f(X;\theta_f) = s|z) = p(f(X;\theta_f) = s|z').$$

That is such that $f(X;\theta_f)$ and Z are independent variables.

Method

Let $r := p_{\theta_r}(z|f(X;\theta_f) = s)$ be an adversary model with parameters θ_r and loss $\mathcal{L}_r(\theta_f, \theta_r)$. As with generative adversarial networks, f and r are trained simultaneously, which is carried out by considering the value function

$$E(\theta_f, \theta_r) = \mathcal{L}_f(\theta_f) - \mathcal{L}_r(\theta_f, \theta_r) \tag{1}$$

that we optimize by finding the minimax solution

$$\hat{\theta}_f, \hat{\theta}_r = \arg \min_{\theta_f} \max_{\theta_r} E(\theta_f, \theta_r). \tag{2}$$

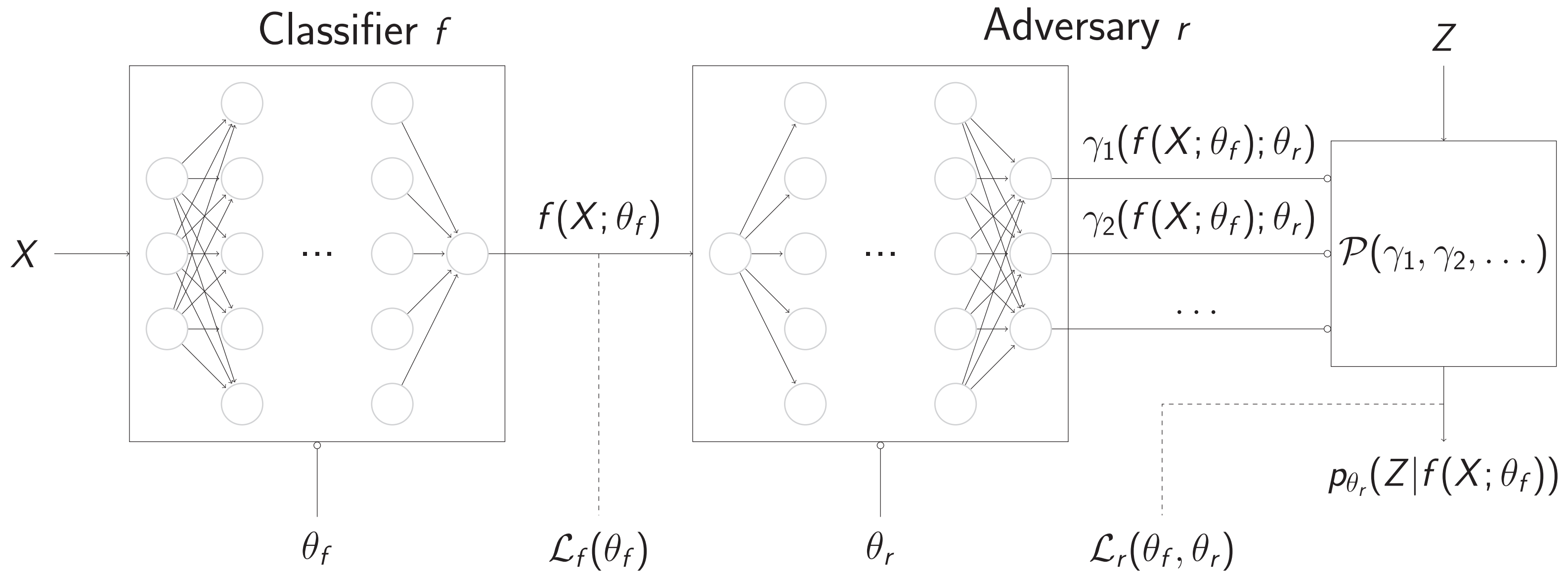
Classifier f
Adversary r
Z
X
$\cdots$
$f(X;\theta_f)$
$\cdots$
$\gamma_1(f(X;\theta_f);\theta_r)$
$\gamma_2(f(X;\theta_f);\theta_r)$
$\dots$
$\mathcal{P}(\gamma_1, \gamma_2, \dots)$
$p_{\theta_r}(Z|f(X;\theta_f))$
θ_f
$\mathcal{L}_f(\theta_f)$
θ_r
$\mathcal{L}_r(\theta_f, \theta_r)$

Figure 1. Architecture for the adversarial training of a binary classifier f against a nuisance parameters Z. The adversary r models the distribution $p(z|f(X;\theta_f) = s)$ of the nuisance parameters as observed only through the output $f(X;\theta_f)$ of the classifier. By maximizing the antagonistic objective $\mathcal{L}_r(\theta_f, \theta_r)$, the classifier f forces $p(z|f(X;\theta_f) = s)$ towards the prior $p(z)$, which happens when $f(X;\theta_f)$ is independent of the nuisance parameter Z and therefore pivotal.

Theoretical motivation

Assume that $\mathcal{L}_f$ and $\mathcal{L}_r$ are respectively set to the expected value of the negative log-likelihood of $Y|X$ under f and of $Z|f(X;\theta_f)$ under r :

$$\mathcal{L}_f(\theta_f) = \mathbb{E}_{x\sim X}\mathbb{E}_{y\sim Y|x}[-\log p_{\theta_f}(y|x)] \tag{3}$$

$$\mathcal{L}_r(\theta_f, \theta_r) = \mathbb{E}_{s\sim f(X;\theta_f)}\mathbb{E}_{z\sim Z|s}[-\log p_{\theta_r}(z|s)] \tag{4}$$

Proposition 1. *If there exists a minimax solution $(\hat{\theta}_f, \hat{\theta}_r)$ for Eqn. (2) such that $E(\hat{\theta}_f, \hat{\theta}_r) = H(Y|X) - H(Z)$, then $f(\cdot;\hat{\theta}_f)$ is both an optimal classifier and a pivotal quantity.*

Proof. (*sketch*)

(i) For fixed θ_f, adversary r is optimal at $\hat{\hat{\theta}}_r = \arg\max_{\theta_r} E(\theta_f, \theta_r) = \arg\min_{\theta_r} \mathcal{L}_r(\theta_f, \theta_r)$, in which case $p_{\hat{\hat{\theta}}_r}(z|f(X;\theta_f) = s) = p(z|f(X;\theta_f) = s)\ \forall z, s$, and $\mathcal{L}_r \rightarrow \mathbb{E}_{s\sim f(X;\theta_f)}[H(Z|f(X;\theta_f) = s)]$;

(ii) Value function E can be restated as a function of θ_f only : $E'(\theta_f) = \mathcal{L}_f(\theta_f) - H(Z|f(X;\theta_f))$;

(iii) We have the lower bound $H(Y|X) - H(Z) \leq \mathcal{L}_f(\theta_f) - H(Z|f(X;\theta_f))$ where the equality holds at $\hat{\theta}_f = \arg\min_{\theta_f} E'(\theta_f)$ when :
• $\hat{\theta}_f$ minimizes the negative log-likelihood of $Y|X$ under f, which happens when $\hat{\theta}_f$ are the parameters of an optimal classifier. Then, $\mathcal{L}_f$ reduces to minimum value $H(Y|X)$.
• $\hat{\theta}_f$ maximizes the conditional entropy $H(Z|f(X;\theta_f))$, since $H(Z|f(X;\theta)) \leq H(Z)$.

⇒ By assumption, the bound is active, hence $H(Z|f(X;\theta_f)) = H(Z)$ because of the second condition, which happens when Z and $f(X;\theta_f)$ are independent variables. □

In practice

The assumption of existence of an optimal and pivotal classifier may not hold because the nuisance parameter directly shapes the decision boundary. In this situation, the value function E can be rewritten as

$$E_\lambda(\theta_f, \theta_r) = \mathcal{L}_f(\theta_f) - \lambda \mathcal{L}_r(\theta_f, \theta_r) \tag{5}$$

where $\lambda \geq 0$ is a hyper-parameter controlling the trade-off between the performance of f and its independence with respect to the nuisance parameter.
- Setting λ to a large value preferably enforces f to be pivotal.
- Setting λ close to 0 constrains f to be optimal.

Toy example

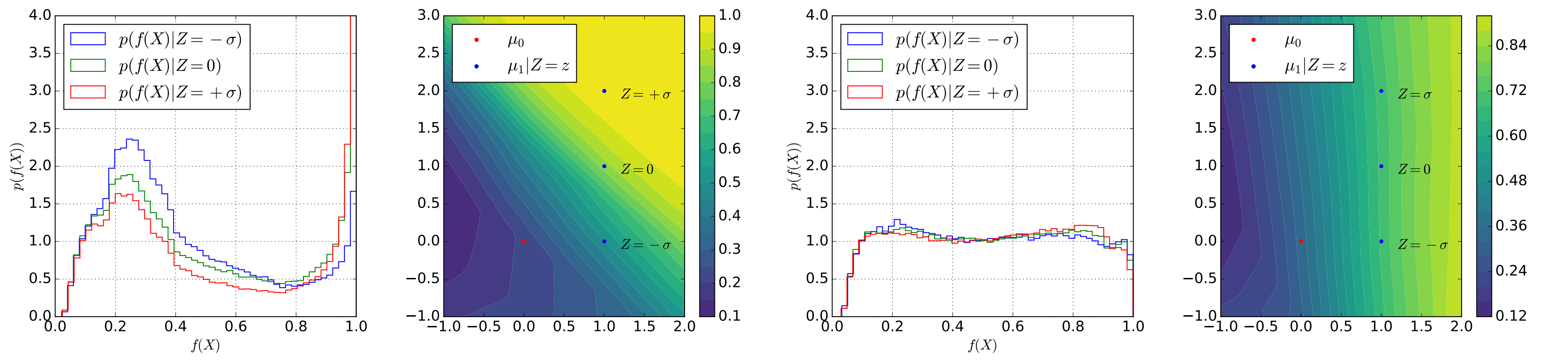

Figure 2. (Left) Decision scores without adversarial training, showing clear dependence on Z (Middle left) Decision surface. Samples are easier to classify for values of Z above σ. (Middle right) Decision scores with adversarial training. The resulting densities are now almost identical to each other, indicating only a small dependency on Z. (Right) Adversarial training reshapes the decision function vertically to erase the dependency on Z.

Examples

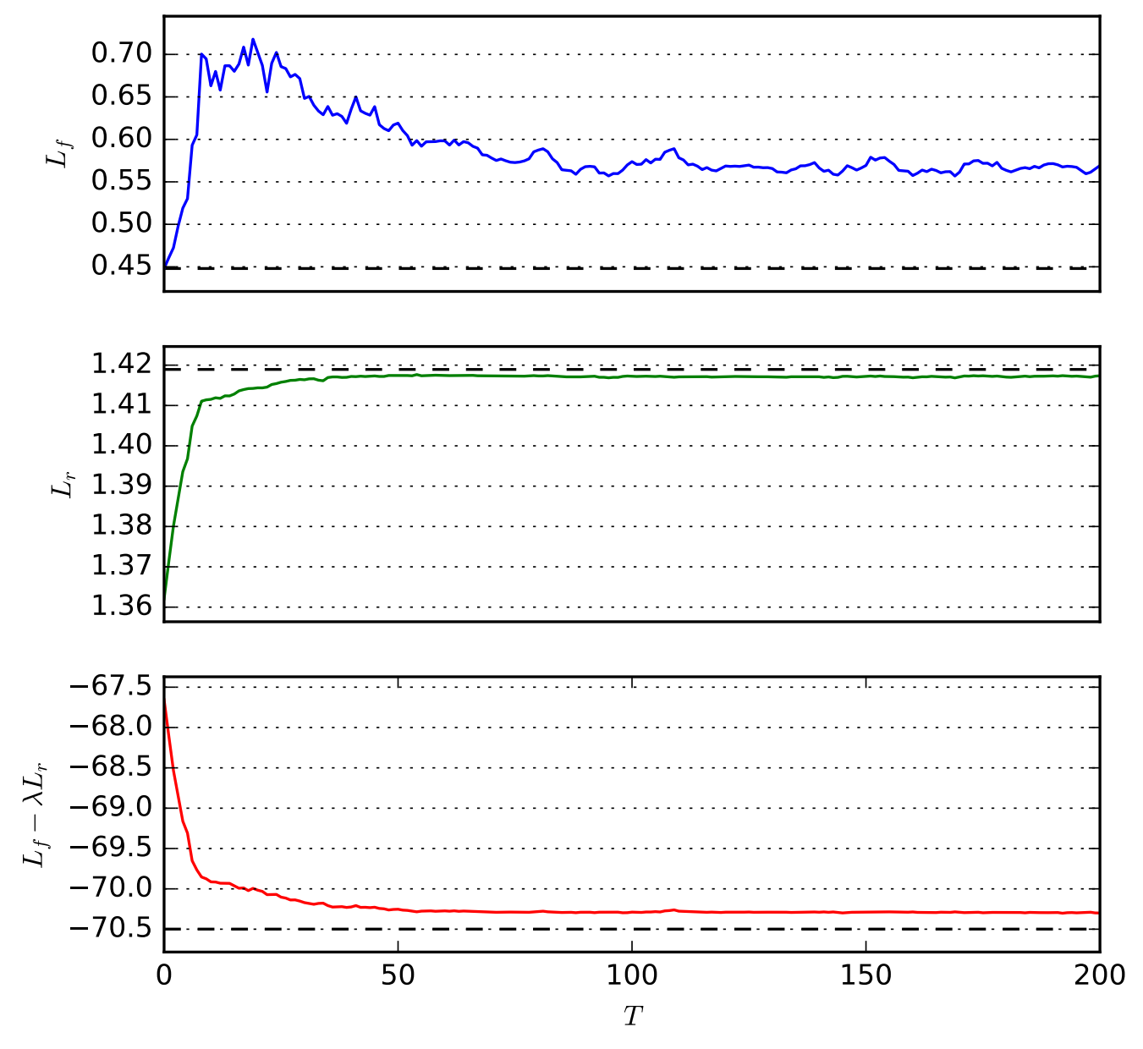

Figure 3. Toy example. Training curves for $\mathcal{L}_f(\theta_f)$, $\mathcal{L}_r(\theta_f, \theta_r)$ and $\mathcal{L}_f(\theta_f) - \lambda\mathcal{L}_r(\theta_f, \theta_r)$. Initialized with a pre-trained classifier f, adversarial training was performed for 200 iterations, mini-batches of size $M = 128$, $K = 500$ and $\lambda = 50$

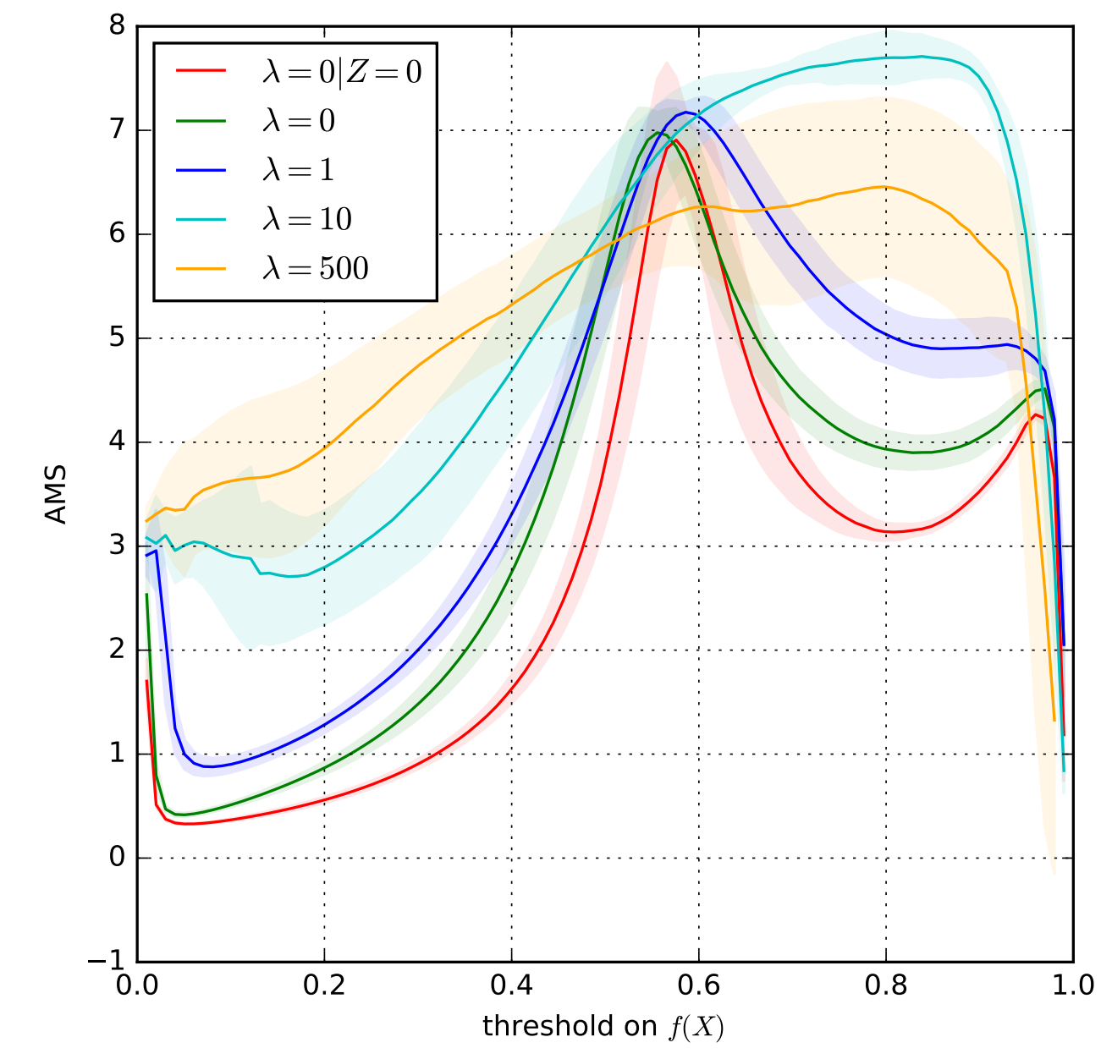

Figure 4. Physics example. Approximate median significance as a function of the decision threshold on the output of f. At $\lambda = 10$, trading accuracy for independence to pileup results in a net benefit in terms of statistical significance.

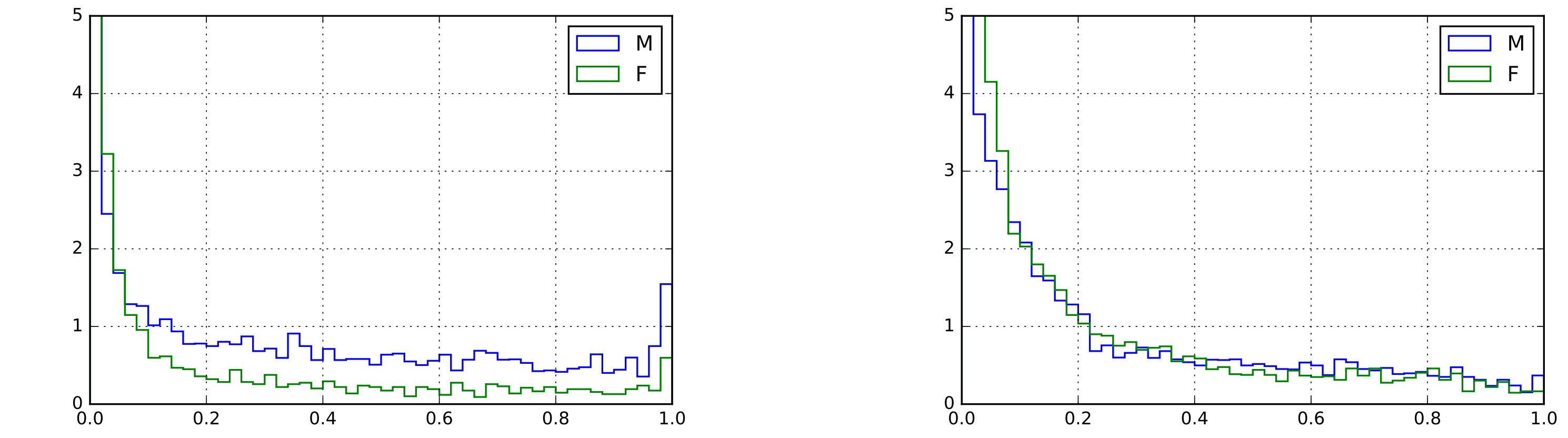

Figure 5. Building a fair classifier independent of gender. (Left) Without adversarial training, decision scores depend on gender. (Right) With adversarial training.

Conclusions

✓ Proposed a flexible learning procedure for building a predictive model that is independent of continuous or categorical nuisance parameters by jointly training two neural networks in an adversarial fashion.

✓ Motivated the algorithm by showing that the minimax value of its value function corresponds to a predictive model that is both optimal and pivotal (if that models exists) or for which one can tune the trade-off between power and robustness.

□ Proposed solution can be used in any situation where the training data may not be representative of the real data the predictive model will be applied to in practice.

□ In the scientific context, the presence of systematic uncertainty can be incorporated by considering a family of data generation processes.

□ The approach extends to cases where independence of the predictive model with respect to observed random variables is desired, as in fairness for classification.

KC and GL are both supported through NSF ACI-1450310, additionally KC is supported through PHY-1505463 and PHY-1205376 and GL through the NRB Big Data Chair. MK is supported by the US Department of Energy (DOE) under grant DE-AC02-76SF00515 and by the SLAC Panofsky Fellowship.