Modern Paradigms in Learning and Decision-Making
Lecture #9: An Online Learning Perspective on Performative Prediction*

Juan C. Perdomo†

Fall 2026

1 Recap

Last lecture, we introduced performative prediction [PZMDH20] and studied the following retraining dynamic:

$$\theta_{t+1} = G(\theta_t) := \operatorname*{argmin}_{\theta \in \Theta} \mathop{\mathbf{E}}_{z \sim \mathcal{D}(\theta_t)} \ell(z; \theta). \tag{1}$$

Here, $\mathcal{D}(\cdot)$ is the *distribution map*, a function from models $\theta \in \Theta \subseteq \mathbb{R}^d$ to distributions over data points $z \in \mathcal{Z}$, and $\ell(z; \theta)$ is the loss. A model θ_{PS} is *performatively stable* if it is a fixed point of G:

$$\theta_{\mathrm{PS}} = G(\theta_{\mathrm{PS}}) \iff \mathop{\mathbf{E}}_{z \sim \mathcal{D}(\theta_{\mathrm{PS}})} \ell(z; \theta_{\mathrm{PS}}) \leq \min_{\theta \in \Theta} \mathop{\mathbf{E}}_{z \sim \mathcal{D}(\theta_{\mathrm{PS}})} \ell(z; \theta).$$

The main result of the previous lecture was that retraining is a contraction, and therefore converges to a stable point, as per the following theorem.

Theorem 1.1 ([PZMDH20])**.** *Assume that $\ell(z; \theta)$ is γ-strongly convex in θ for every z, that $\nabla_\theta \ell(z; \theta)$ is β-Lipschitz in z for every θ, and that $\mathcal{D}(\cdot)$ is ε-sensitive. Then, with $\rho := \varepsilon\beta/\gamma$,*

$$\left\|G(\theta) - G(\theta')\right\|_2 \leq \rho \left\|\theta - \theta'\right\|_2 \qquad \text{for all } \theta, \theta' \in \Theta.$$

Notice that if $\rho < 1$, then G is a contraction. By Banach's fixed point theorem, G then has a unique fixed point θ_{PS}, and retraining converges to it at a linear rate,

$$\|\theta_t - \theta_{\mathrm{PS}}\|_2 \leq \rho^t \|\theta_0 - \theta_{\mathrm{PS}}\|_2 .$$

In the theorem, we used the assumption that $\mathcal{D}(\cdot)$ is ε-*sensitive*, meaning that for all $\theta, \theta' \in \Theta$,

$$\mathcal{W}_1\big(\mathcal{D}(\theta), \mathcal{D}(\theta')\big) \leq \varepsilon \left\|\theta - \theta'\right\|_2 ,$$

where $\mathcal{W}_1$ is the Wasserstein-1 distance. Intuitively, ε captures the strength of performativity: it bounds how much the induced distribution can move when the model moves. Recall from last

*These notes are a work in progress and have not been subjected to the usual scrutiny reserved for formal publications. Please email me if you see any mistakes or find something confusing.

†© 2026, Juan C. Perdomo. Not to be sold, published, or distributed without the authors' consent.

lecture that the proof of Theorem 1.1 plays this bound against the curvature of the loss, which is where the ratio $\rho = \varepsilon\beta/\gamma$ comes from. In particular, retraining is only guaranteed to converge when performativity is weak relative to the curvature of the loss, $\varepsilon < \gamma/\beta$. The next example shows that this condition is tight: retraining oscillates forever as soon as $\rho \geq 1$.

Example 1.2 (Retraining oscillates when $\rho \geq 1$)**.** *Let $\Theta = [0,1]$, $\ell(z;\theta) = \frac{1}{2}(z-\theta)^2$, so $\gamma = \beta = 1$, and let $\mathcal{D}(\theta)$ be a point mass at $z = 1-\varepsilon\theta$. The higher the prediction, the lower the outcome. Think of a high predicted risk triggering an intervention that reduces the risk. Since $\mathcal{W}_1(\mathcal{D}(\theta), \mathcal{D}(\theta')) = \varepsilon|\theta - \theta'|$, the map is ε-sensitive and $\rho = \varepsilon$. In this example, the retraining operator takes the following form:*

$$G(\theta) = \mathrm{Proj}_{[0,1]}(1 - \varepsilon\theta) = \max\{0,\, 1 - \varepsilon\theta\}.$$

The unique stable point is $\theta_{\mathrm{PS}} = 1/(1+\varepsilon)$.

- *If $\varepsilon < 1$, then $\theta_t \to \theta_{\mathrm{PS}}$ at rate ε^t.*
- *If $\varepsilon = 1$, then θ_t alternates between θ_0 and $1 - \theta_0$ forever.*
- *If $\varepsilon > 1$, the gap $|\theta_t - \theta_{\mathrm{PS}}|$ grows by a factor ε per round until the iterates hit the boundary, after which they alternate $0, 1, 0, 1, \ldots$ forever. For instance, from $\theta_0 = 0$ we get $G(0) = 1$ and $G(1) = 0$.*

1.1 Is this the limit?

Notice that Theorem 1.1 and Example 1.2 apply to a specific procedure, retraining. They leave open the question of whether other algorithms could find stable models when $\rho \geq 1$. We point out two barriers to any such alternative.

- **Stable models need not exist.** If $\mathcal{D}(\cdot)$ is discontinuous, G need not have a fixed point. Take $\ell(z;\theta) = (z-\theta)^2$ on $\Theta = [0,1]$ and let $\mathcal{D}(\theta)$ be a point mass at $z = 0$ if $\theta > 1/2$ and at $z = 1$ if $\theta \leq 1/2$. Then $G(\theta) = \mathbf{1}\{\theta \leq 1/2\}$ has no fixed point, and retraining alternates $0, 1, 0, 1, \ldots$
- **When they exist, they are hard to find.** If $\mathcal{D}(\cdot)$ is ε-sensitive and ℓ is strongly convex, then G is ρ-Lipschitz and hence continuous, so by Brouwer's theorem a stable model exists on any compact convex Θ. But Anagnostides et al. [ACP+26] show that computing one is PPAD-complete once $\rho \geq 1 + \Omega(1)$, even for strongly convex, smooth losses and ε-sensitive maps.[1] So under standard complexity assumptions, no efficient algorithm finds a single stable model outside the contractive regime.

1.2 Mixtures

Fortunately, there is a way around these impossibility results. Instead of predicting according to a single model, we allow ourselves to predict according to a mixture of models. A *mixture* is a

[1] PPAD is the class of problems polynomial-time equivalent to finding a Brouwer fixed point; computing a Nash equilibrium of a two-player game is PPAD-complete.

distribution $\mu \in \Delta(\Theta)$ over models. For example, a mixture supported on two points deploys θ_1 with probability p and θ_2 with probability $1-p$. More generally, μ can be supported on many points, and a single model is the special case of a point mass.

When the Learner deploys a mixture, it induces a data distribution according to the following process. We first draw a model from the mixture, and then people respond to the specific model deployed on them:

$$\theta \sim \mu, \qquad z \sim \mathcal{D}(\theta).$$

Intuitively, if a bank uses a prediction to decide who gets a loan, people respond to the specific prediction made on them, not to the bank's distribution over models. We now extend the definition of stability to accommodate mixtures.

Definition 1.3 (Stable mixture)**.** *A mixture $\mu \in \Delta(\Theta)$ is* ε-performatively stable *if*

$$\mathop{\mathbf{E}}_{\theta\sim\mu}\mathop{\mathbf{E}}_{z\sim\mathcal{D}(\theta)} \ell(z;\theta) \le \min_{\theta'\in\Theta} \mathop{\mathbf{E}}_{\theta\sim\mu}\mathop{\mathbf{E}}_{z\sim\mathcal{D}(\theta)} \ell(z;\theta') + \varepsilon. \tag{2}$$

Stable mixtures inherit the same motivation as stable models. From the perspective of the distribution induced by deploying μ, there is no incentive to deviate via retraining: no single model achieves lower loss on that data.

In the rest of the lecture, we will show that stable mixtures are always efficiently computable by establishing a reduction to online learning. In order to do so, we first need to provide a bit of background on martingales.

2 Background

A *stochastic process* is a sequence of random variables $X_0, X_1, X_2, \ldots$, indexed by time. Let $\mathcal{F}_0 \subseteq \mathcal{F}_1 \subseteq \cdots$ be an increasing sequence of histories, where $\mathcal{F}_n$ is the collection of random variables observed up to time n and $\mathcal{F}_0$ is empty.[2] We say X_n is *measurable with respect to* $\mathcal{F}_n$ if it is a deterministic function of the variables in $\mathcal{F}_n$.

We recall the following facts about conditional expectations. If X is measurable with respect to $\mathcal{F}$, then $\mathbf{E}[XY \mid \mathcal{F}] = X\,\mathbf{E}[Y \mid \mathcal{F}]$ (taking out what is known). For histories $\mathcal{F} \subseteq \mathcal{G}$, $\mathbf{E}[\mathbf{E}[X \mid \mathcal{G}] \mid \mathcal{F}] = \mathbf{E}[X \mid \mathcal{F}]$ (tower property), and in particular $\mathbf{E}[\mathbf{E}[X \mid \mathcal{G}]] = \mathbf{E}[X]$.

Definition 2.1 (Martingale)**.** *A process $X_0, X_1, \ldots$, with X_n measurable with respect to $\mathcal{F}_n$, is a* martingale *with respect to $\{\mathcal{F}_n\}$ if $\mathbf{E}\,|X_n| < \infty$ and*

$$\mathbf{E}[X_{n+1} \mid \mathcal{F}_n] = X_n \qquad \textit{for all } n \ge 0.$$

Given the past, the best guess for the next value is the current value. The fortune of a gambler in a fair game is the canonical example: if $\xi_1, \xi_2, \ldots$ are independent with $\mathbf{Pr}[\xi_i = \pm 1] = 1/2$, then $X_n = \sum_{i\le n} \xi_i$ is a martingale with respect to $\mathcal{F}_n = (\xi_1, \ldots, \xi_n)$. The increments need not be independent of the past; they only need to have mean zero conditional on it.

[2] Formally, each $\mathcal{F}_n$ is a σ-algebra and the sequence is a *filtration*.

Definition 2.2 (Martingale difference sequence)**.** *A sequence $Z_1, Z_2, \ldots,$ with Z_n measurable with respect to $\mathcal{F}_n$, is a* martingale difference sequence *with respect to $\{\mathcal{F}_n\}$ if $\mathbf{E}\,|Z_n| < \infty$ and*

$$\mathbf{E}[Z_n \mid \mathcal{F}_{n-1}] = 0 \qquad \text{for all } n \geq 1.$$

If X_n is a martingale, then $Z_n = X_n - X_{n-1}$ is a martingale difference sequence, and conversely the partial sums of a martingale difference sequence form a martingale.

Notice that any martingale difference sequence satisfies $\mathbf{E}[Z_n] = 0$ for every n, by the tower property. Consequently,

$$\mathbf{E}\Big[\sum_{n=1}^{N} Z_n\Big] = \sum_{n=1}^{N} \mathbf{E}\left[\,\mathbf{E}[Z_n \mid \mathcal{F}_{n-1}]\right] = 0. \tag{3}$$

This is not just true in expectation, but also with high probability. Sums of bounded martingale differences concentrate in the same way as sums of independent random variables.

Theorem 2.3 (Azuma-Hoeffding)**.** *Let $Z_1, \ldots, Z_N$ be a martingale difference sequence with $|Z_n| \leq B$ with probability one. Then, for all $s \geq 0$,*

$$\mathbf{Pr}\Big[\sum_{n=1}^{N} Z_n \geq s\Big] \leq \exp\Big(-\frac{s^2}{2B^2N}\Big).$$

We will use the fact in (3) in the next section as part of an online-to-batch conversion, to relate the guarantees of online algorithms to those of the batch problem.

3 No-Regret Algorithms are Performatively Stable

In this section, we show how, using tools from online learning and martingales, no-regret algorithms naturally converge to performatively stable mixtures.

We start by recalling the online learning protocol. At each round t, the Learner selects $\theta_t \in \Theta$, and then Nature, having seen θ_t, selects a loss $\ell_t : \Theta \to \mathbb{R}$. An algorithm is *no-regret* if it selects θ_t deterministically as a function of $\ell_1, \ldots, \ell_{t-1}$, and for every sequence of losses,

$$\sum_{t=1}^{T} \ell_t(\theta_t) - \min_{\theta \in \Theta} \sum_{t=1}^{T} \ell_t(\theta) \leq \mathrm{Regret}_T = o(T).$$

Notice that in this protocol, Nature can be adaptive: the loss ℓ_t can depend on θ_t. Furthermore, the guarantee is deterministic, since it holds for every sequence of losses $\ell_1, \ldots, \ell_T$.

Using these two facts, we can show that if we make predictions using any no-regret algorithm, as described in Algorithm 1, then the uniform distribution over the sequence of models it produces is approximately stable.

Algorithm 1: No-regret learning under performativity

Input: a no-regret algorithm $\mathcal{A}$ over Θ, horizon T.
For $t = 1, \dots, T$:

1. $\mathcal{A}$ outputs θ_t. The Learner deploys θ_t and observes $z_t \sim \mathcal{D}(\theta_t)$.
2. $\mathcal{A}$ receives the loss $\ell_t(\theta) := \ell(z_t; \theta)$.

Output: the uniform mixture μ over $\theta_1, \dots, \theta_T$.

Theorem 3.1 ([FP26]). *Assume ℓ is bounded. Let $\theta_1, \dots, \theta_T$ be the iterates of Algorithm 1, let $S = (z_1, \dots, z_T)$, and let μ be the uniform mixture over the iterates. Then*

$$\mathop{\mathbf{E}}_{S}\Big[\mathop{\mathbf{E}}_{\theta\sim\mu}\mathop{\mathbf{E}}_{z\sim\mathcal{D}(\theta)}\ell(z;\theta)\Big] \le \min_{\theta'\in\Theta}\mathop{\mathbf{E}}_{S}\Big[\mathop{\mathbf{E}}_{\theta\sim\mu}\mathop{\mathbf{E}}_{z\sim\mathcal{D}(\theta)}\ell(z;\theta')\Big] + \frac{\mathrm{Regret}_T}{T}.$$

That is, μ is Regret_T/T-performatively stable in expectation over the samples.

Proof. Fix $\theta_\star \in \Theta$. Since μ is uniform over the iterates,

$$\mathop{\mathbf{E}}_{\theta\sim\mu}\mathop{\mathbf{E}}_{z\sim\mathcal{D}(\theta)}\big[\ell(z;\theta) - \ell(z;\theta_\star)\big] = \frac{1}{T}\sum_{t=1}^{T}\mathop{\mathbf{E}}_{z\sim\mathcal{D}(\theta_t)}\big[\ell(z;\theta_t) - \ell(z;\theta_\star)\big].$$

Add and subtract the realized losses $\ell(z_t;\theta_t) - \ell(z_t;\theta_\star)$:

$$= \underbrace{\frac{1}{T}\sum_{t=1}^{T} V_t}_{\text{martingale term}} + \underbrace{\frac{1}{T}\sum_{t=1}^{T}\big[\ell(z_t;\theta_t) - \ell(z_t;\theta_\star)\big]}_{\le \mathrm{Regret}_T/T},$$

where

$$V_t := \mathop{\mathbf{E}}_{z\sim\mathcal{D}(\theta_t)}\big[\ell(z;\theta_t) - \ell(z;\theta_\star)\big] - \big[\ell(z_t;\theta_t) - \ell(z_t;\theta_\star)\big].$$

The second term is at most Regret_T/T by the regret guarantee, applied to the realized losses $\ell_t = \ell(z_t;\cdot)$. For the first, let $\mathcal{F}_t = (z_1, \dots, z_t)$. Since $\mathcal{A}$ is deterministic, θ_t is measurable with respect to $\mathcal{F}_{t-1}$, and conditional on $\mathcal{F}_{t-1}$, z_t is a fresh draw from $\mathcal{D}(\theta_t)$. Hence

$$\mathbf{E}\big[\ell(z_t;\theta_t) - \ell(z_t;\theta_\star)\,\big|\,\mathcal{F}_{t-1}\big] = \mathop{\mathbf{E}}_{z\sim\mathcal{D}(\theta_t)}\big[\ell(z;\theta_t) - \ell(z;\theta_\star)\big],$$

so $\mathbf{E}[V_t \mid \mathcal{F}_{t-1}] = 0$. The V_t are bounded, so $V_1, \dots, V_T$ is a martingale difference sequence and $\mathbf{E}_S[\sum_t V_t] = 0$ by (3). Taking expectations over S and noting $\theta_\star$ was arbitrary proves the claim. □

This proof technique is known as an online-to-batch conversion. Normally, in an online-to-batch conversion, one assumes that the samples are drawn i.i.d. from a fixed distribution. Here, we made use of the fact that Nature could be adaptive to sample z_t from $\mathcal{D}(\theta_t)$, which changes every round. Crucially, we assumed nothing about $\mathcal{D}(\cdot)$: no Lipschitzness, no continuity. The distribution map could be anything. See the following animation for an empirical example.

4 Implications

Theorem 3.1 turns every regret bound into a stability guarantee. Throughout, assume gradients are bounded by G and Θ is compact and convex with diameter D:

$$\sup_{z,\theta} \|\nabla_\theta \ell(z;\theta)\|_2 \le G, \qquad \sup_{\theta,\theta' \in \Theta} \left\|\theta - \theta'\right\|_2 \le D.$$

In each case below, μ is the uniform mixture over the iterates $\theta_1, \dots, \theta_T$, and stability is in expectation over the samples.

- **Follow the leader [HAK07, SSK08].** If each ℓ_t is γ-strongly convex and

$$\theta_t \in \operatorname*{argmin}_{\theta \in \Theta} \sum_{j=1}^{t-1} \ell_j(\theta),$$

  then $\mathrm{Regret}_T \le \frac{G^2}{2\gamma}(\log T + 1)$. Consequently, if ℓ is γ-strongly convex and we run follow the leader for T rounds on the losses $\ell_t = \ell(z_t;\cdot)$ with $z_t \sim \mathcal{D}(\theta_t)$, that is, if we retrain on the entire history,

$$\theta_t \in \operatorname*{argmin}_{\theta \in \Theta} \sum_{j=1}^{t-1} \ell(z_j;\theta), \qquad z_j \sim \mathcal{D}(\theta_j), \tag{4}$$

  then μ is $\frac{G^2}{2\gamma} \cdot \frac{\log T + 1}{T}$-performatively stable.

- **Online gradient descent, strongly convex losses [HAK07].** If each ℓ_t is γ-strongly convex and

$$\theta_{t+1} = \mathrm{Proj}_\Theta\big[\theta_t - \eta_t \nabla \ell_t(\theta_t)\big], \qquad \eta_t = \frac{1}{\gamma t},$$

  then $\mathrm{Regret}_T \le \frac{G^2}{2\gamma}(1 + \log T)$. Consequently, if ℓ is γ-strongly convex and we run stochastic gradient descent for T rounds on the data induced by the current model,

$$\theta_{t+1} = \mathrm{Proj}_\Theta\big[\theta_t - \eta_t \nabla_\theta \ell(z_t;\theta_t)\big], \qquad z_t \sim \mathcal{D}(\theta_t), \tag{5}$$

  with $\eta_t = 1/(\gamma t)$, then μ is $\frac{G^2}{2\gamma} \cdot \frac{1 + \log T}{T}$-performatively stable.

- **Online gradient descent, convex losses [Zin03].** If each ℓ_t is convex and we run the same update with $\eta_t = D/(G\sqrt{t})$, then $\mathrm{Regret}_T \le \frac{3}{2} GD\sqrt{T}$. Consequently, if ℓ is convex and we run (5) for T rounds with $\eta_t = D/(G\sqrt{t})$, then μ is $\frac{3}{2} \cdot \frac{GD}{\sqrt{T}}$-performatively stable.

Retraining, revisited. The online learning perspective yields much improved guarantees for retraining. First, Theorem 1.1 only applied to exact minimization at the population level where we had exact access to $\mathcal{D}(\cdot)$. The follow the leader algorithm works while only seeing samples z drawn $\mathcal{D}(\theta_t)$. Second, Theorem 1.1 required $\mathcal{D}(\cdot)$ to be ε-sensitive, and hence continuous, with $\rho < 1$. The first bullet makes no assumption whatsoever on $\mathcal{D}(\cdot)$.

Gradient descent, revisited. Prior work [MDPZH20, DX23] showed that (5) converges to a stable model when ℓ is smooth and strongly convex and $\rho < 1$. The second bullet matches their $O(1/T)$ rate up to the $\log T$ factor, with no sensitivity assumption and no smoothness. The third bullet shows how to achieve stability even for losses that are convex but not strongly convex. For instance, the squared loss or log loss.

5 The Limits of Stability

On a more sobering note, the results we've presented are relevant for finding stable points, not optimal points. And stable points can be arbitrarily bad from the perspective of the performative risk.

The intuition is that stability is an echo chamber. We make predictions in such a way that the data looks optimal for those predictions. However, we could do better by actively steering the data. The following example tries to make this point concrete:

Example 5.1 (The stable point maximizes the performative risk)**.** *Let* $\Theta = [0, 1]$, $\ell(y; \theta) = (y-\theta)^2$, *and* $\mathcal{D}(\theta) = \mathrm{Ber}\big(\frac{1}{2} + \varepsilon(\theta - \frac{1}{2})\big)$ *for some* $\varepsilon \in (1/2, 1)$. $\mathcal{D}(\cdot)$ *is* ε*-sensitive and* $\rho = \varepsilon < 1$. *By Theorem 1.1 retraining converges to the unique stable point,* $\theta_{\mathrm{PS}} = 1/2$. *A direct computation gives*

$$\mathrm{PR}(\theta) = \mathop{\mathbf{E}}_{y \sim \mathcal{D}(\theta)} (y - \theta)^2 = \tfrac{1}{4} + (1 - 2\varepsilon)\big(\theta - \tfrac{1}{2}\big)^2.$$

Since $\varepsilon > 1/2$, PR *is maximized at* $\theta = 1/2 = \theta_{\mathrm{PS}}$, *and minimized at the endpoints* $\theta_{\mathrm{PO}} \in \{0, 1\}$, *where* $\mathrm{PR} = (1 - \varepsilon)/2 < 1/4$. *Retraining converges, quickly, to the worst model.*

One way to understand what is happening is to compute the gradient of the performative risk. Because θ appears both in the loss and in the distribution the expectation is taken over, the gradient has two terms. Suppose $\mathcal{D}(\theta)$ has density p_θ. Then

$$\nabla \mathrm{PR}(\theta) = \underbrace{\mathop{\mathbf{E}}_{z \sim \mathcal{D}(\theta)} \nabla_\theta \ell(z; \theta)}_{\text{(A)}} + \underbrace{\mathop{\mathbf{E}}_{z \sim \mathcal{D}(\theta)} \left[\ell(z; \theta)\, \nabla_\theta \log p_\theta(z)\right]}_{\text{(B)}}. \tag{6}$$

Term (A) is the gradient of the loss on the current data, holding the distribution fixed. It is the only term that retraining and gradient descent ever see, and at a stable point it is zero. Term (B) accounts for how the distribution moves when the model moves. It need not vanish at a stable point, and it can be as large as we like. A nonzero gradient means that we could lower the performative risk by moving the model to actively change the data to a "better" place.

In Example 5.1, the outcome is $\mathrm{Ber}(q(\theta))$ with $q(\theta) = \frac{1}{2} + \varepsilon(\theta - \frac{1}{2})$, and the two terms work out to

$$\text{(A)} = -2 \mathop{\mathbf{E}}_{y \sim \mathcal{D}(\theta)} [y - \theta] = 2(1 - \varepsilon)\big(\theta - \tfrac{1}{2}\big), \qquad \text{(B)} = q'(\theta)\big[(1 - \theta)^2 - \theta^2\big] = -2\varepsilon\big(\theta - \tfrac{1}{2}\big).$$

For $\varepsilon > 1/2$, term (B) is larger in magnitude than term (A) and points the opposite way, so the gradient $\nabla \mathrm{PR}(\theta) = 2(1 - 2\varepsilon)(\theta - \frac{1}{2})$ has the opposite sign of (A). The direction that increases the loss on the current data decreases the performative risk. At $\theta_{\mathrm{PS}} = 1/2$ both terms vanish, since the stable point is the maximizer of PR, and any move away from it lowers the risk.

When performatively optimal points can be computed efficiently is a largely open question. Even when ℓ is convex, PR is in general not convex in θ, since θ enters both the loss and the distribution. We mention two settings where progress has been made.

Linear label shift. Suppose again that performativity is only in the labels, and that θ enters linearly: $x \sim \mathcal{D}_x$, and

$$y = y_0 + \Phi(x)^\top \theta, \qquad (x, y_0) \sim \mathcal{D}_0,$$

for a base distribution $\mathcal{D}_0$ and a fixed map $\Phi : \mathcal{X} \to \mathbb{R}^d$. For the squared loss $\ell(z;\theta) = (y - \theta^\top x)^2$,

$$\mathrm{PR}(\theta) = \mathop{\mathbf{E}}_{(x,y_0)\sim\mathcal{D}_0} \left(y_0 + (\Phi(x) - x)^\top \theta\right)^2$$

is a convex function of an affine function of θ, and hence convex. Note that this holds for every Φ, so ρ can be arbitrarily large; see [MPZ21] for the general treatment. Convexity says nothing about optimization, though: term (B) in (6) depends on Φ, which is unknown and only observable through the models we deploy.

Outcome performativity. Kim and Perdomo [KP23] study the related outcome performative setting, where $x \sim \mathcal{D}_x$ is sampled from a fixed distribution and the binary outcome $y \sim \mathcal{D}_y(x, \widehat{y})$ depends on one of finitely many actions $\widehat{y}$. They show how, by applying some of the outcome indistinguishability machinery we learned earlier in the class, one can learn a performative omnipredictor for this setting: a single predictor from which performatively optimal decisions can be derived simultaneously for every loss in a class.

Both of these appraoches only apply to performativity in the labels and not features. They also make further restrictions like linearity.

Performative feedback. Jagadeesan, Zrnic, and Mendler-Dünner [JZMD22] use tools from bandit optimization to optimize the performative risk directly. They make use of the fact that the Learner gets richer feedback than in a standard bandit problem: after deploying θ_t, it sees samples drawn from $\mathcal{D}(\theta_t)$, rather than a single scalar loss. However, these approaches are computationally inefficient.

References

[ACP+26] Ioannis Anagnostides, Rohan Chauhan, Ioannis Panageas, Tuomas Sandholm, and Jingming Yan. On the computational complexity of performative prediction. *arXiv preprint arXiv:2601.20180*, 2026.

[DX23] Dmitriy Drusvyatskiy and Lin Xiao. Stochastic optimization with decision-dependent distributions. *Mathematics of Operations Research*, 2023.

[FP26] Gabriele Farina and Juan Carlos Perdomo. The stability of online algorithms in performative prediction. *Working paper*, 2026.

[HAK07] Elad Hazan, Amit Agarwal, and Satyen Kale. Logarithmic regret algorithms for online convex optimization. *Machine Learning*, 69, 2007.

[JZMD22] Meena Jagadeesan, Tijana Zrnic, and Celestine Mendler-Dünner. Regret minimization with performative feedback. In *International Conference on Machine Learning (ICML)*, 2022.

[KP23] Michael P. Kim and Juan C. Perdomo. Making decisions under outcome performativity. In *14th Innovations in Theoretical Computer Science Conference (ITCS)*, 2023.

[MDPZH20] Celestine Mendler-Dünner, Juan C. Perdomo, Tijana Zrnic, and Moritz Hardt. Stochastic optimization for performative prediction. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[MPZ21] John P. Miller, Juan C. Perdomo, and Tijana Zrnic. Outside the echo chamber: Optimizing the performative risk. In *International Conference on Machine Learning (ICML)*, 2021.

[PZMDH20] Juan C. Perdomo, Tijana Zrnic, Celestine Mendler-Dünner, and Moritz Hardt. Performative prediction. In *International Conference on Machine Learning (ICML)*, 2020.

[SSK08] Shai Shalev-Shwartz and Sham M. Kakade. Mind the duality gap: Logarithmic regret algorithms for online optimization. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2008.

[Zin03] Martin Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. In *International Conference on Machine Learning (ICML)*, 2003.