Implicit Regularization in Overparameterized Bilevel Optimization

Paul Vicol[1,2], Jonathan Lorraine[1,2], David Duvenaud[1,2], Roger Grosse[1,2]
[1]University of Toronto, [2]Vector Institute

Motivation & Summary

- Bilevel problems involve inner and outer parameters, each optimized for its own objective.

$$\mathrm{x}^* \in \arg\min_{\mathrm{x}} F(\mathrm{x}, \mathrm{y}^*)$$
$$\mathrm{y}^* \in \mathcal{S}(\mathrm{x}) = \arg\min_{\mathrm{y}} f(\mathrm{x}, \mathrm{y})$$

- **Examples**: hyperparameter optimization, dataset distillation, meta-learning, NAS, and GANs.
- Most prior work assumes that the inner & outer objectives have unique solutions, but often in practice, at least one of them is overparameterized → non-unique.
- We investigate the inductive biases of different gradient-based algorithms for jointly optimizing the inner and outer parameters.
- We distinguish between two different solution concepts—cold-start and warm-start equilibria
- The behavior depends on algorithmic choices such as the hypergradient approximation.

Gradient-Based Bilevel Optimization

- Gradient-based bilevel opt requires the gradient of the outer objective with respect to the outer parameters, called the *hypergradient*. For a given solution $\mathrm{y}^* \in \mathcal{S}(\mathrm{x})$, which is called a *best-response* to x:

$$\frac{dF(\mathrm{x}, \mathrm{y}^*)}{d\mathrm{x}} = \frac{\partial F}{\partial \mathrm{x}} + \frac{\partial F}{\partial \mathrm{y}^*} \frac{\partial \mathrm{y}^*}{\partial \mathrm{x}}$$

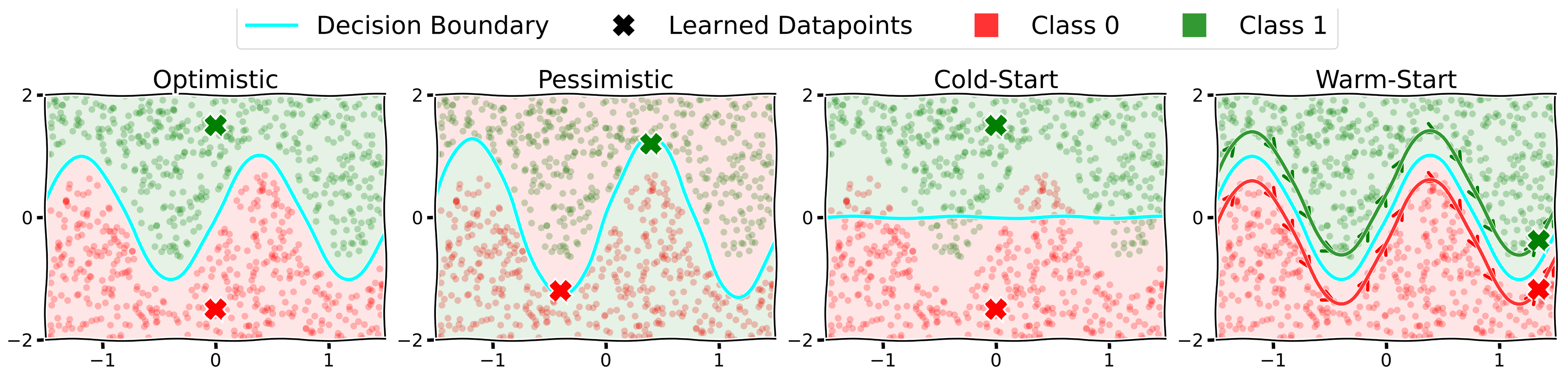
- Common ways to compute the response Jacobian are:
- Differentiation through unrolling: $\frac{d\mathrm{y}^*}{d\mathrm{x}} \approx \frac{d\Phi_k(\mathrm{y}_0, \mathrm{x})}{d\mathrm{x}}$
- Implicit differentiation: $\frac{d\mathrm{y}^*}{d\mathrm{x}} = -\left(\frac{\partial^2 f}{\partial \mathrm{y} \partial \mathrm{y}^\top}\right)^{-1} \frac{\partial^2 f}{\partial \mathrm{y} \partial \mathrm{x}}$
- Common approximations to the inverse Hessian include: 1) truncated CG, and 2) the truncated Neumann series:

$$\left(\frac{\partial^2 f}{\partial \mathrm{y} \partial \mathrm{y}^\top}\right)^{-1} \approx \sum_{j=0}^{K} \left(I - \frac{\partial^2 f}{\partial \mathrm{y} \partial \mathrm{y}^\top}\right)^j$$

Warm-Start vs Cold-Start

- **Cold-start:** re-initialize the inner parameters and run the inner optimization to convergence each time we compute the gradient for the outer parameters
- **Warm-start:** jointly optimize the inner and outer parameters in an online fashion, e.g., alternating gradient steps with their respective objectives

Warm-Start vs Cold-Start (Contd.)



Solutions for Overparameterized Inner Problems

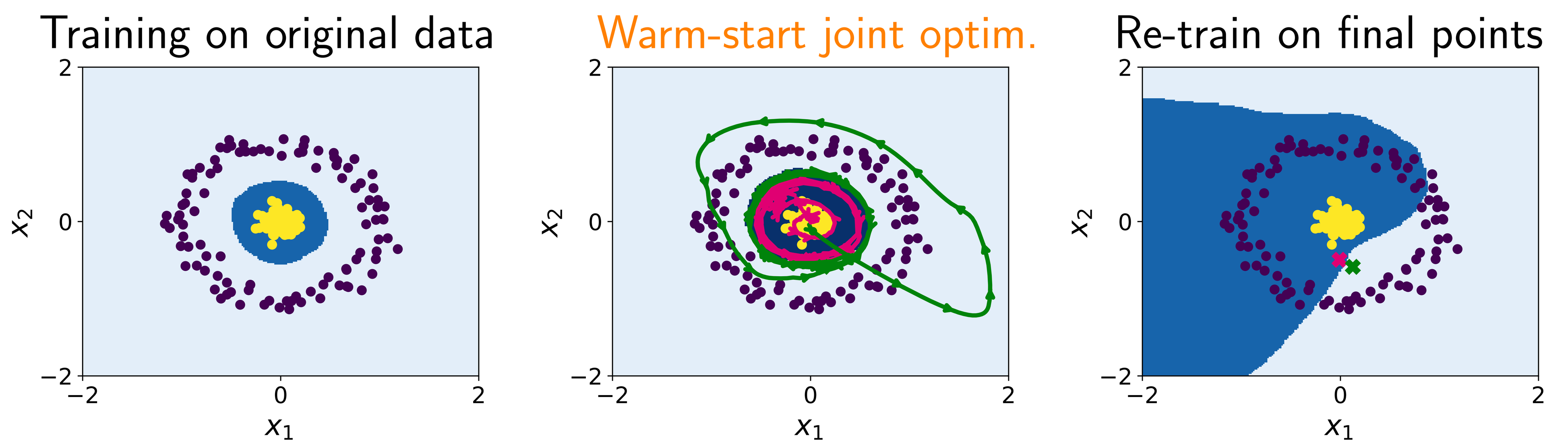
- The *optimistic* solution chooses the inner parameters that achieve the *best outer-objective value*, $\arg\min_{\mathrm{y} \in \mathcal{S}(\mathrm{x})} F(\mathrm{x}, \mathrm{y})$.
- The *pessimistic* solution chooses $\mathrm{y} \in \mathcal{S}(\mathrm{x})$ that achieves the *worst outer-objective value*, $\arg\max_{\mathrm{y} \in \mathcal{S}(\mathrm{x})} F(\mathrm{x}, \mathrm{y})$.
- In practice, due to the implicit bias of gradient descent, the $\mathrm{y} \in \mathcal{S}(\mathrm{x})$ we end up at depends on the inner initialization y_0: with cold-start, we obtain y that minimize the distance from y_0: $\arg\min_{\mathrm{y} \in \mathcal{S}(\mathrm{x})} ||\mathrm{y} - \mathrm{y}_0||_2^2$.
- With warm-start, the trajectory of outer parameters x during joint optimization (shown by the arrows) influences the inner parameters y.

Proximal Inner Optimization

- We can formalize warm-started joint optimization by considering a proximally regularized inner objective: $\mathrm{y}^* \in \arg\min_{\mathrm{y}} \{ f(\mathrm{x}, \mathrm{y}) + \frac{\epsilon}{2} ||\mathrm{y} - \mathrm{y}_k||^2 \}$

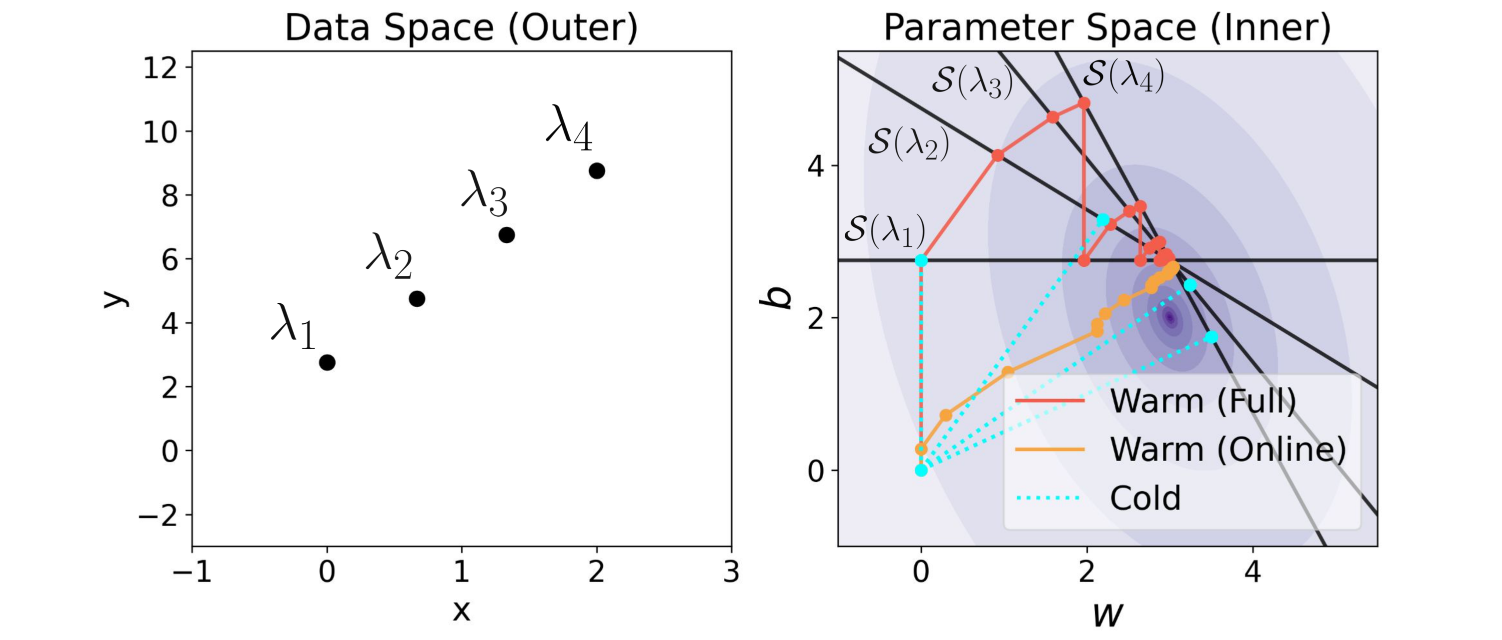
Cold-Start	Warm-Start
$\mathrm{x}_{t+1} = \mathrm{x}_t - \alpha \frac{\partial F}{\partial \mathrm{y}^*} \frac{\partial \mathrm{y}^*}{\partial \mathrm{x}}$	$\mathrm{x}_{t+1} = \mathrm{x}_t - \alpha \frac{\partial F}{\partial \mathrm{y}_t^*} \frac{\partial \mathrm{y}_t^*}{\partial \mathrm{x}}$
$\mathrm{y}_{t+1}^* \in \arg\min_{\mathrm{y} \in \mathcal{S}(\mathrm{x}_{t+1})} ||\mathrm{y} - \mathrm{y}_0||^2$	$\mathrm{y}_{t+1}^* \in \arg\min_{\mathrm{y}} \{ f(\mathrm{x}_{t+1}, \mathrm{y}) + \frac{\epsilon}{2} ||\mathrm{y} - \mathrm{y}_t||^2 \}$

Inner Overparameterization: Dataset Distillation



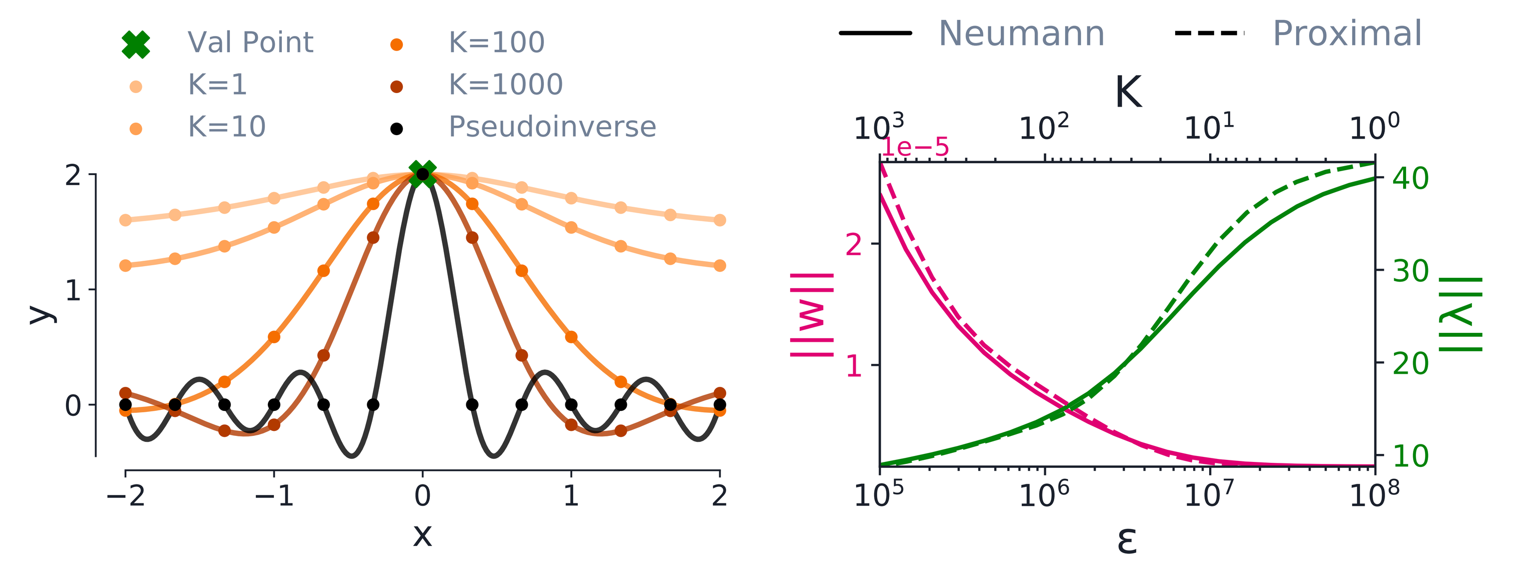
- Dataset distillation for binary classification, with two learned datapoints (outer parameters) adapted jointly with the model weights (inner parameters).
- Because the outer obj is only used to update the outer params, one would think that all of the info about the outer obj is compressed into the outer params.
- Warm-starting yields a *trajectory* that traces out the boundary between classes.
- **Takeaway:** inner params can encode a surprising amount of information about the outer objective, even when the outer params are low-dimensional.

Inner Overparameterization (Contd.)



- Parameter-space view of warm-start with full inner optimization, warm-start with partial inner optimization (denoted "online"), and cold-start optimization.
- Cold-start projects from the origin onto the solution set for the current datapoint
- Warm-start projects from the current weights onto the solution set for the current datapoint
- By successive projection between solution sets, the inner parameters converge to the intersection of the solution sets, yielding inner params that perform well for multiple outer params simultaneously.

Outer Overparameterization: Anti-Distillation



- Fourier-basis 1D linear regression: we learn the y-coord of 13 synthetic datapoints such that a regressor trained on them will fit a single "val" datapoint, at the green X.
- **Left:** learned datapoints (outer params) from different hypergrad approximations: truncated Neumann/diff-through-unrolling with different # steps K
- **Right:** The norms of the inner and outer parameters, $||\mathrm{w} - \mathrm{w}_0||^2$ and $||\boldsymbol{\lambda} - \boldsymbol{\lambda}_0||^2$ as a function of K (for Neumann/unrolling) or ϵ (for proximal).
- **Takeaway:** Empirically, the amount of inner optimization we perform affects the trade-off between the norms of the inner and outer params