Implicit Regularization in Overparameterized Bilevel Optimization

Paul Vicol, Jonathan Lorraine, David Duvenaud, Roger Grosse

ICML 2021 Workshop
Beyond First-Order Methods in ML Systems

UNIVERSITY OF TORONTO

VECTOR INSTITUTE

Bilevel Optimization & Hypergradients

- A bilevel optimization problem consists of two *nested sub-problems*, where the outer problem must be solved subject to optimality of the inner problem:

$$\mathbf{x}^* \in \arg\min_{\mathbf{x}} F(\mathbf{x}, \mathbf{y}^*)$$

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

- Hypergradient for a given solution $\mathbf{y}^* \in \mathcal{S}(\mathbf{x})$ is:

$$\frac{dF(\mathbf{x}, \mathbf{y}^*)}{d\mathbf{x}} = \overbrace{\frac{\partial F}{\partial \mathbf{x}}}^{\textit{direct gradient}} + \underbrace{\frac{\partial F}{\partial \mathbf{y}^*} \overset{\textit{response Jacobian}}{\boxed{\frac{\partial \mathbf{y}^*}{\partial \mathbf{x}}}}}_{\textit{response gradient}}$$

The Many Uses of Response Jacobians

- The response Jacobian $\frac{\partial \mathbf{y}^*}{\partial \mathbf{x}}$ is a central quantity of interest for many applications

Response Jacobians in Bilevel Settings

- Hyperparameter optimization (including data augmentation and NAS)
- Dataset distillation
- GANs
- Actor-critic methods and multi-agent RL
- Adversarial training
- Meta-learning

Response Jacobians in Non-Bilevel Settings

- Influence functions to estimate the effect of changing the weighting of a training point
- Implicit layers in equilibrium models
- Optimizing convergent recurrent neural networks (via recurrent backpropagation)

Computing the Response and its Jacobian

- Exactly computing a *best-response or its Jacobian is expensive*
  - We typically approximate $\mathbf{y}^*$ or $\frac{d\mathbf{y}^*}{d\mathbf{x}}$ or both → leads to *two sources of approximation error for the hypergradient*.

- Common to approximate the best-response via truncated unrolls of the inner problem:

$$\mathbf{y}^*(\mathbf{x}) \approx \Phi_k(\mathbf{y}_0, \mathbf{x})$$

- The two main ways to compute the best-response Jacobian are:
  1. Differentiation through unrolling (a.k.a. *iterative differentiation*)

$$\frac{d\mathbf{y}^*}{d\mathbf{x}} \approx \frac{d\Phi_k(\mathbf{y}_0, \mathbf{x})}{d\mathbf{x}}$$

  2. *Implicit differentiation*, applicable when we are at the converged solution to the inner problem:

$$\frac{d\mathbf{y}^*}{d\mathbf{x}} = -\left(\frac{\partial^2 f}{\partial \mathbf{y} \partial \mathbf{y}^\top}\right)^{-1} \frac{\partial^2 f}{\partial \mathbf{y} \partial \mathbf{x}}$$

Implicit Differentiation

$$\frac{d\mathbf{y}^*}{d\mathbf{x}} = -\left(\frac{\partial^2 f}{\partial \mathbf{y} \partial \mathbf{y}^\top}\right)^{-1} \frac{\partial^2 f}{\partial \mathbf{y} \partial \mathbf{x}}$$

- The *inverse Hessian is intractable* to compute for large networks
- Two main approximations to the inverse Hessian have been proposed in the literature
  - Both can be *implemented efficiently through Hessian-vector products*

Conjugate Gradient (CG)

- Solve the linear system with CG

$$\left(\frac{\partial^2 f}{\partial \mathbf{y} \partial \mathbf{y}^\top}\right) \frac{d\mathbf{y}^*}{d\mathbf{x}} = -\frac{\partial^2 f}{\partial \mathbf{y} \partial \mathbf{x}}$$

- CG is only applicable when the matrix to be inverted is PSD
- Empirically, using truncated CG can have very different inductive bias than truncated Neumann

Neumann Series

$$T^{-1} = \sum_{k=0}^{\infty} (I - T)^k$$

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

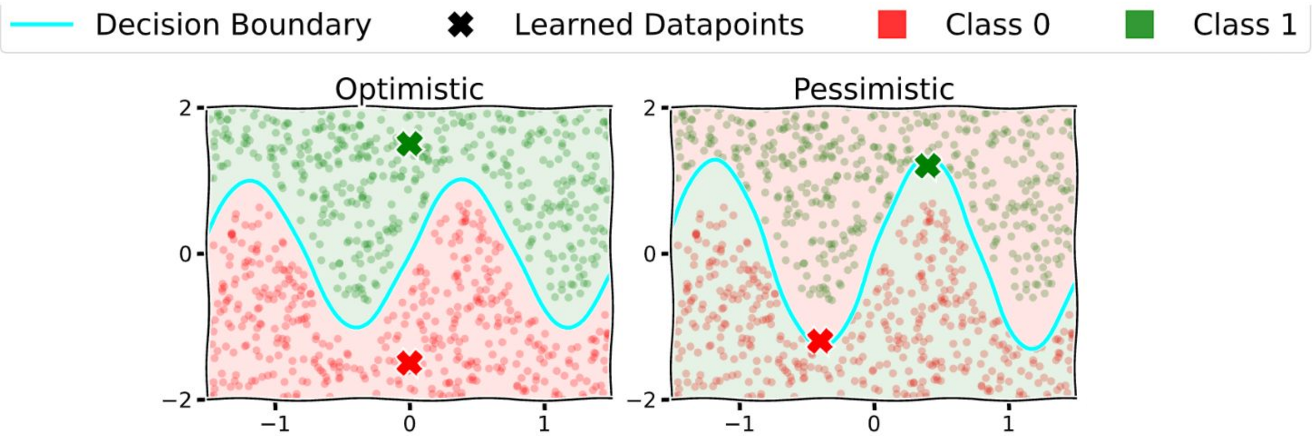
Connection between diff-through-opt and implicit differentiation: unrolling differentiation for i steps starting from optimal inner parameters is equivalent to approximating the inverse Hessian with the first i terms in the Neumann series.

Uniqueness of the Inner Solution

$$\mathbf{x}^* \in \arg\min_{\mathbf{x}} F(\mathbf{x}, \mathbf{y}^*)$$

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

- When the inner problem is *overparameterized*, there are many equally good solutions to the inner optimization, so the best response is a *set and not unique*
- Different choices of inner parameters yield different best-response Jacobians, which lead to different hypergradients



Optimistic

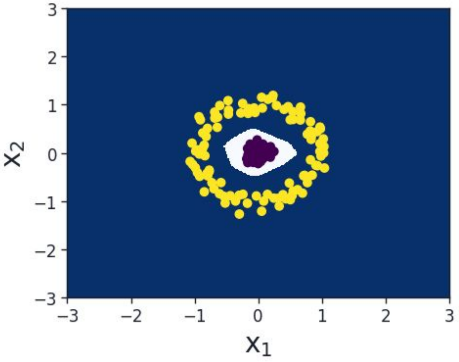
Pessimistic

$$\arg\min_{\mathbf{y} \in \mathcal{S}(\mathbf{x})} F(\mathbf{x}, \mathbf{y}) \qquad \arg\max_{\mathbf{y} \in \mathcal{S}(\mathbf{x})} F(\mathbf{x}, \mathbf{y})$$

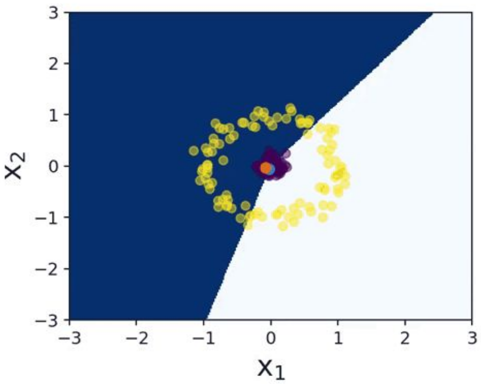
Cold-Start and Warm-Start Bilevel Optimization

- **Cold-start:** re-initialize the inner parameters and run the inner optimization to convergence each time we compute the gradient for the outer parameters
  - Impractical due to the computational expense of full inner optimization
- **Warm-start:** jointly optimize the inner and outer parameters in an *online fashion*, e.g., alternating gradient steps with their respective objectives
  - Here, the inner params are *warm-started from the approximate solution to the inner optimization given the outer params in the previous iteration*
  - The *optimization dynamics* can lead to an implicit regularization effect

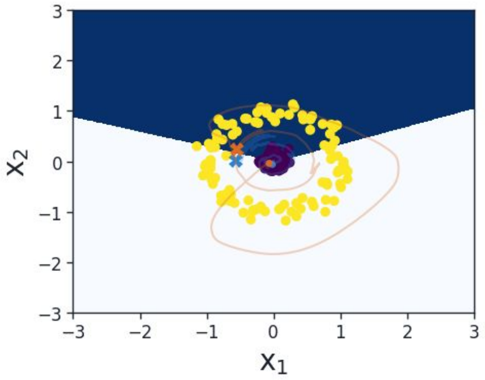
Training on the data directly



Warm-start joint optimization



Training from scratch on final distillation

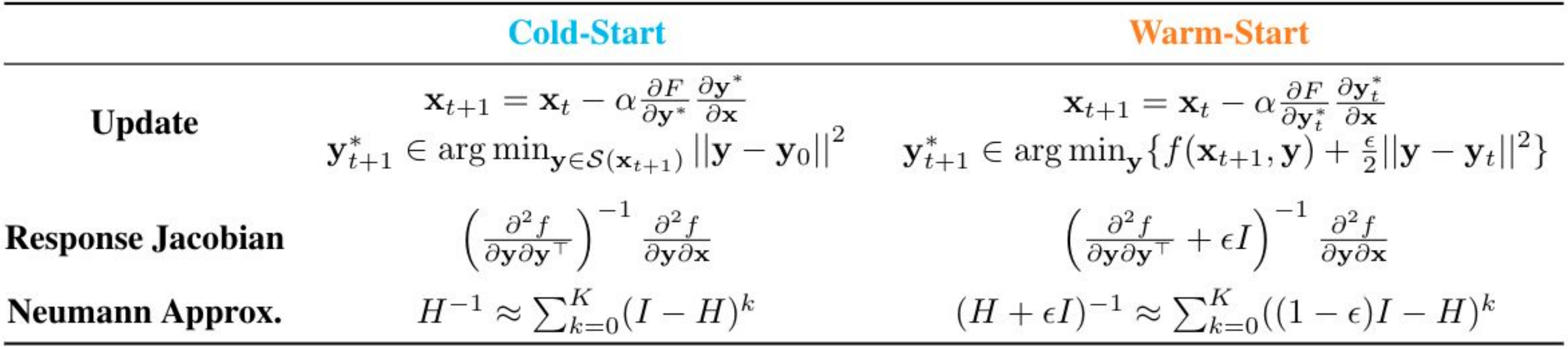


Proximal Inner Objective

- We can formalize warm-started joint optimization by considering a *proximally regularized inner objective*:

$$\mathbf{y}^* \in \arg\min_{\mathbf{y}} \{ f(\mathbf{x}, \mathbf{y}) + \frac{\epsilon}{2} ||\mathbf{y} - \mathbf{y}_k||^2 \}$$

- We define notions of cold-start and warm-start equilibria, which correspond to different solutions than optimistic and pessimistic bilevel opt



	Cold-Start	**Warm-Start**
Update	$\mathbf{x}_{t+1} = \mathbf{x}_t - \alpha \frac{\partial F}{\partial \mathbf{y}^*} \frac{\partial \mathbf{y}^*}{\partial \mathbf{x}}$ $\mathbf{y}^*_{t+1} \in \arg\min_{\mathbf{y} \in \mathcal{S}(\mathbf{x}_{t+1})} ||\mathbf{y} - \mathbf{y}_0||^2$	$\mathbf{x}_{t+1} = \mathbf{x}_t - \alpha \frac{\partial F}{\partial \mathbf{y}^*_t} \frac{\partial \mathbf{y}^*_t}{\partial \mathbf{x}}$ $\mathbf{y}^*_{t+1} \in \arg\min_{\mathbf{y}} \{ f(\mathbf{x}_{t+1}, \mathbf{y}) + \frac{\epsilon}{2} ||\mathbf{y} - \mathbf{y}_t||^2 \}$
Response Jacobian	$\left(\frac{\partial^2 f}{\partial \mathbf{y} \partial \mathbf{y}^\top} \right)^{-1} \frac{\partial^2 f}{\partial \mathbf{y} \partial \mathbf{x}}$	$\left(\frac{\partial^2 f}{\partial \mathbf{y} \partial \mathbf{y}^\top} + \epsilon I \right)^{-1} \frac{\partial^2 f}{\partial \mathbf{y} \partial \mathbf{x}}$
Neumann Approx.	$H^{-1} \approx \sum_{k=0}^{K} (I - H)^k$	$(H + \epsilon I)^{-1} \approx \sum_{k=0}^{K} ((1 - \epsilon) I - H)^k$

Intuition for Cold-Start and Warm-Start Behavior

- Toy linear regression w/ one learned datapoint constrained to move along a line in data-space

- *Cold-start always projects from the origin onto the solution set for the current datapoint*
- *Warm-start projects from the current weights onto the solution set*
  - By successive projection between solution sets, the inner parameters will *converge to the intersection of the solution sets, yielding inner params that perform well for multiple outer params simultaneously*
  - Note that we do not necessarily converge to the optimal validation loss

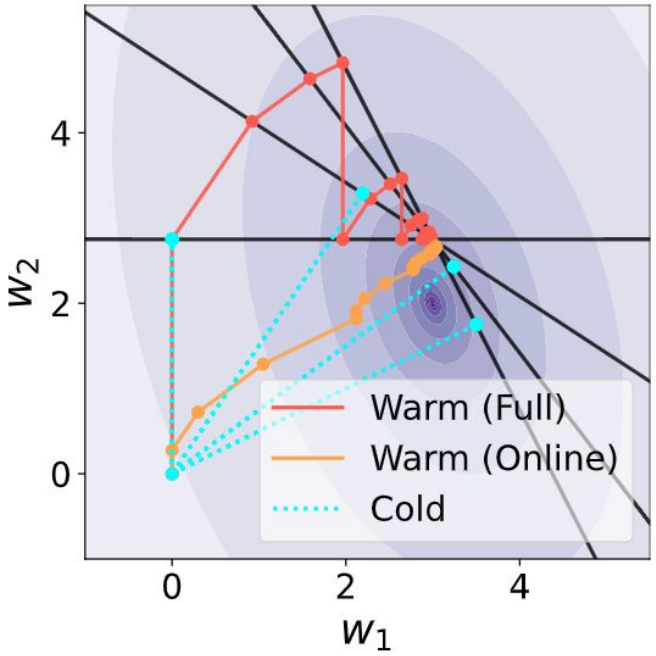
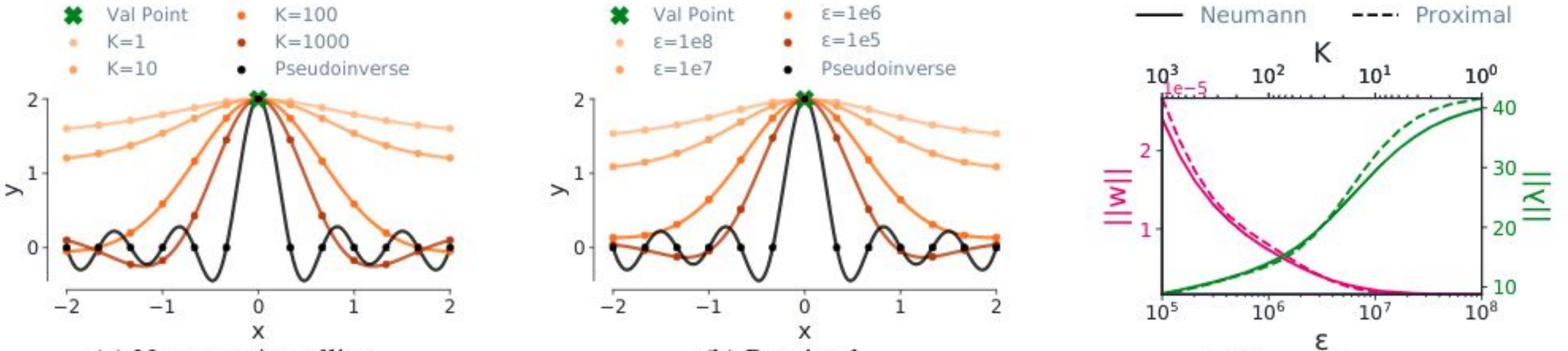


Figure 4: Parameter-space view of warm-start with full inner optimization, warm-start with partial inner optimization (denoted the "online" setting, which most closely resembles what is done in practice), and cold-start optimization.

Outer Overparameterization: Anti-Distillation

- **Anti-distillation:** *more learned datapoints than original dataset examples*
- One validation data point and 13 synthetic training points, so any solution that places a learned datapoint on top of the validation point perfectly fits the outer objective.
- We use Fourier-basis regression, where the low frequency terms have larger amplitude than high frequency terms
- The *quality of hypergradient approximations induces a trade-off between the inner and outer parameter norms*—e.g., we can achieve the good performance for the outer objective by either making larger updates to the inner or the outer parameters

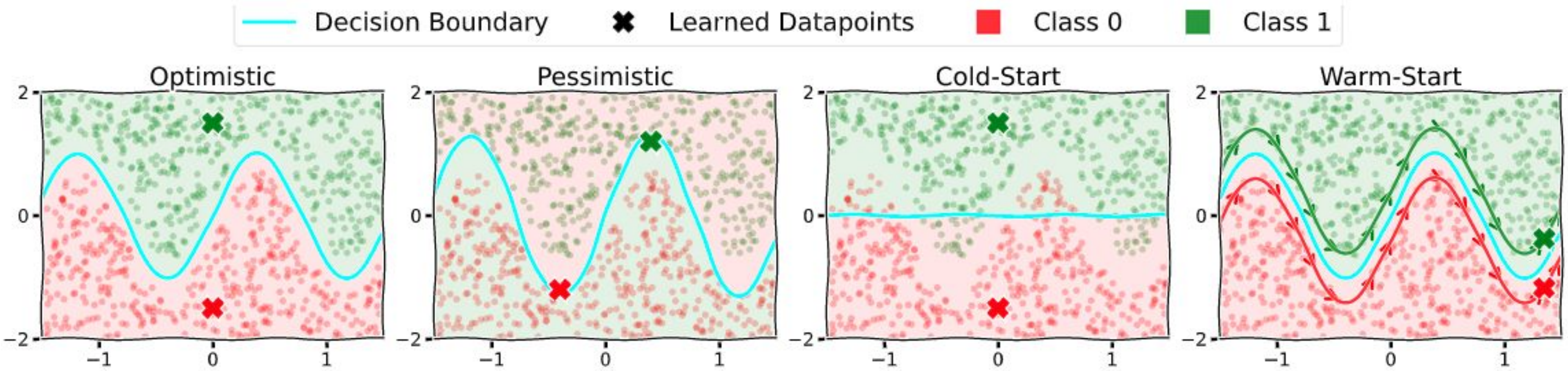


(a) Neumann/unrolling

(b) Proximal

(c) Parameter norms

Revisiting Overparam Bilevel Solutions



$$\underset{\mathbf{y}\in\mathcal{S}(\mathbf{x})}{\arg\min}\, F(\mathbf{x},\mathbf{y})$$

$$\underset{\mathbf{y}\in\mathcal{S}(\mathbf{x})}{\arg\max}\, F(\mathbf{x},\mathbf{y})$$

$$\underset{\mathbf{y}\in\mathcal{S}(\mathbf{x})}{\arg\min}\, ||\mathbf{y}-\mathbf{y}_0||_2^2$$

$$\underset{\mathbf{y}\in\mathcal{S}(\mathbf{x})}{\arg\min}\, ||\mathbf{y}-\mathbf{y}_t||_2^2$$

↑

(a bit more complicated)

Q/A