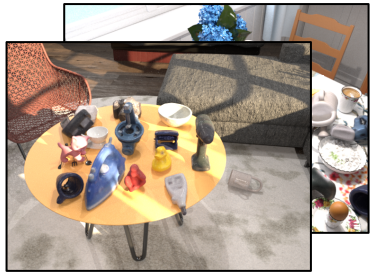




# AutoSimulate: (Quickly) Learning Synthetic Data Generation

Harkirat Singh Behl, AG Baydin, Ran Gal, Philip Torr, Vibhav Vineet

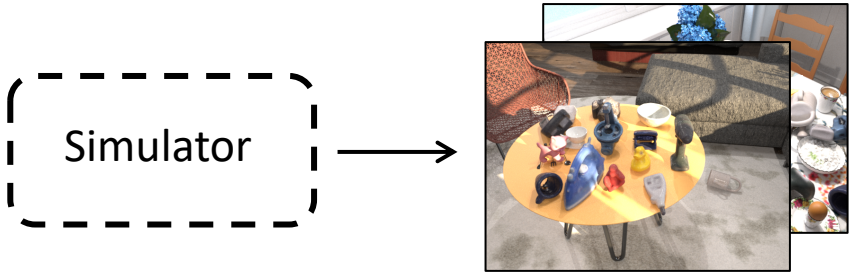
# Neural Networks are Data Hungry



Synthetic Dataset

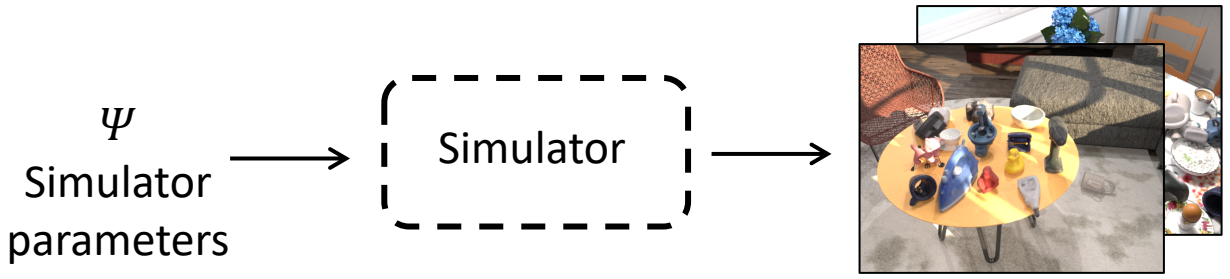

Training on
Synthetic Data

# Synthetic Data Generation




Synthetic Dataset

# Synthetic Data Generation




Object Shapes
Object Colors
Object Locations
Materials
Lighting
Etc.

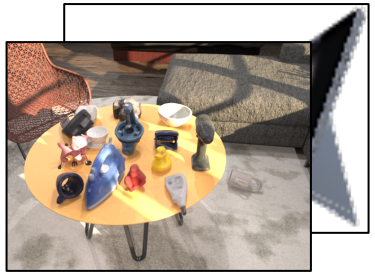
# Manual Synthetic Data Generation

Human Expert

$\Psi$
Simulator parameters

- Object Shapes
- Object Colors
- Object Locations
- Materials
- Lighting
- Etc.

Simulator



Synthetic Dataset

# What's Wrong with Manual Synthetic Data Generation?

NEEDS SIGNIFICANT HUMAN EFFORT

EXPENSIVE

CAN BE SUB-OPTIMAL

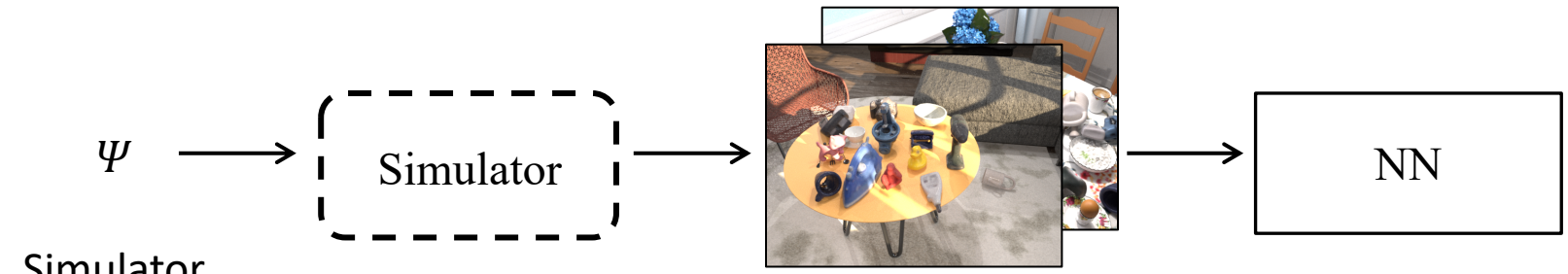
# What's Wrong with Manual Synthetic Data Generation?

NEEDS SIGNIFICANT HUMAN EFFORT

EXPENSIVE

CAN BE SUB-OPTIMAL

# AutoSimulate: Learning Synthetic Data Generation



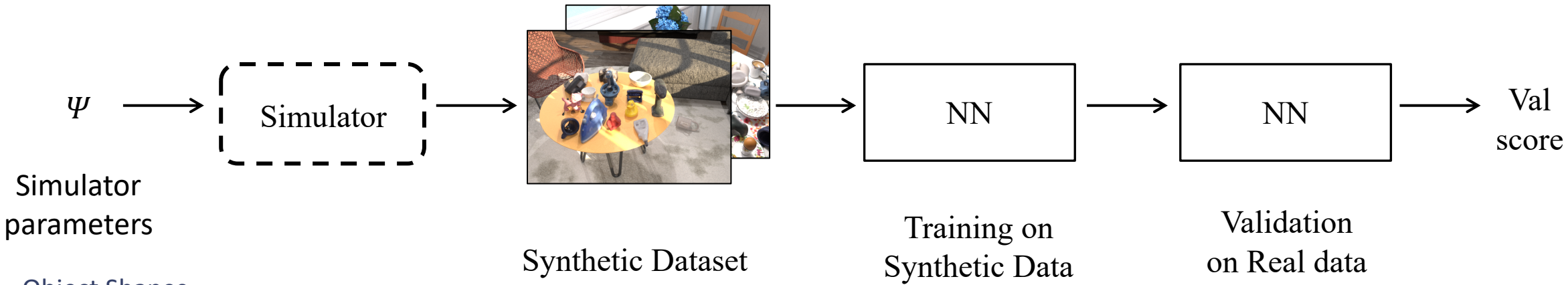

Simulator parameters

Object Shapes
Object Colors
Object Locations
Materials
Lighting
Etc.

Synthetic Dataset

Training on Synthetic Data

# AutoSimulate: Learning Synthetic Data Generation




Object Shapes
Object Colors
Object Locations
Materials
Lighting
Etc.

# AutoSimulate: Learning Synthetic Data Generation


Object Shapes
Object Colors
Object Locations
Materials
Lighting
Etc.

← Backpropagation: We propose Differentiable approximations for the non-differentiable bi-level objective.

# Problem Formulation

- Objective is to find optimal simulator parameters as

$$\min_{\psi} \quad \mathcal{L}_{\text{val}}\big(\hat{\theta}(\psi)\big) \tag{1a}$$

$$s.t. \quad \hat{\theta}(\psi) \in \arg\min_{\theta} \mathcal{L}_{\text{train}}(\theta, \psi). \tag{1b}$$

$\mathcal{L}_{\text{val}}\big(\hat{\theta}(\psi)\big)$: validation loss
$\mathcal{L}_{\text{train}}(\theta, \psi)$: training loss
$\hat{\theta}(\psi)$: neural network parameters after training on data from simulator $\psi$

- Eqns. (1a) and (1b) represent a bi-level optimization problem.

# Problem Formulation

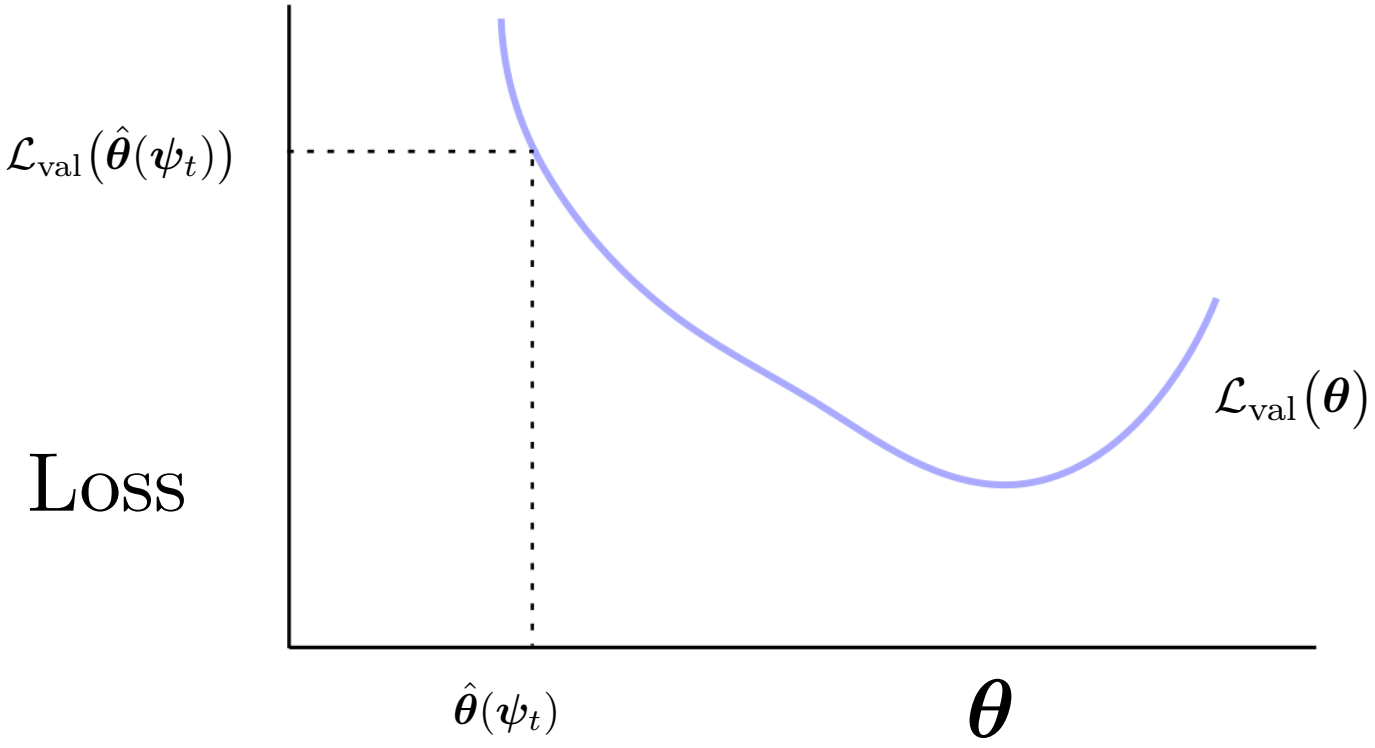
Computing gradient of the bi-level optimization is difficult as:

- Simulator can be non-differentiable
- Backpropagating through NN training is impracticable

We propose differentiable approximations for the non-differentiable bi-level objective

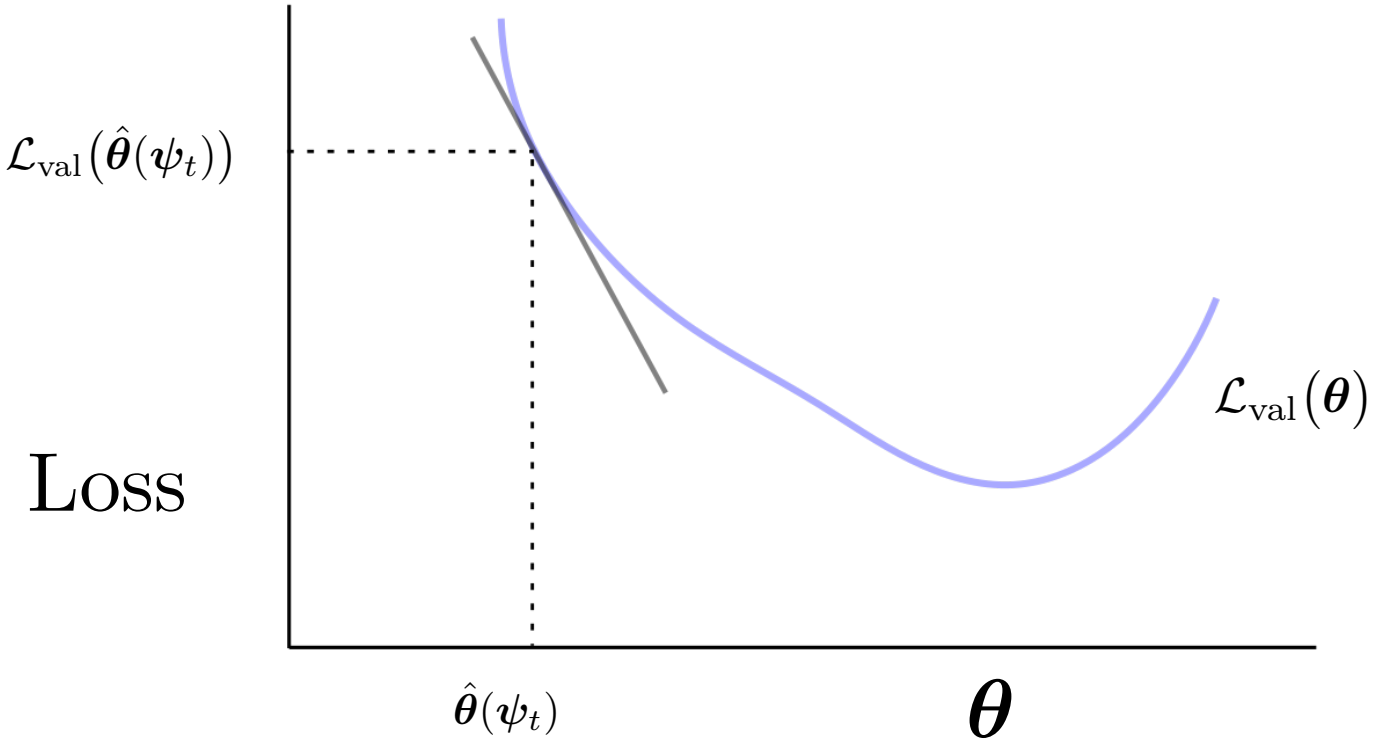
# AutoSimulate

Objective: Minimize $\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)$ with respect to $\boldsymbol{\psi}$



# AutoSimulate

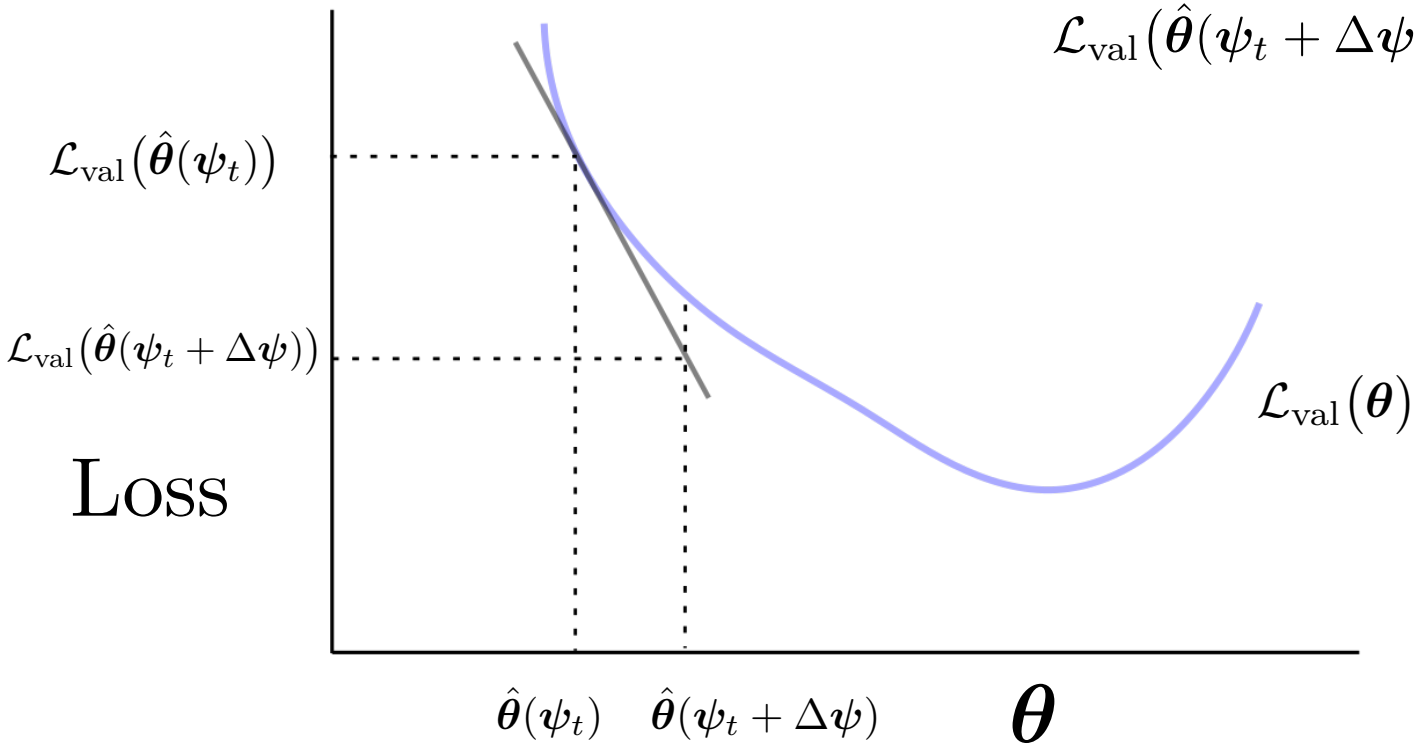
Objective: Minimize $\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)$ with respect to $\boldsymbol{\psi}$



# AutoSimulate

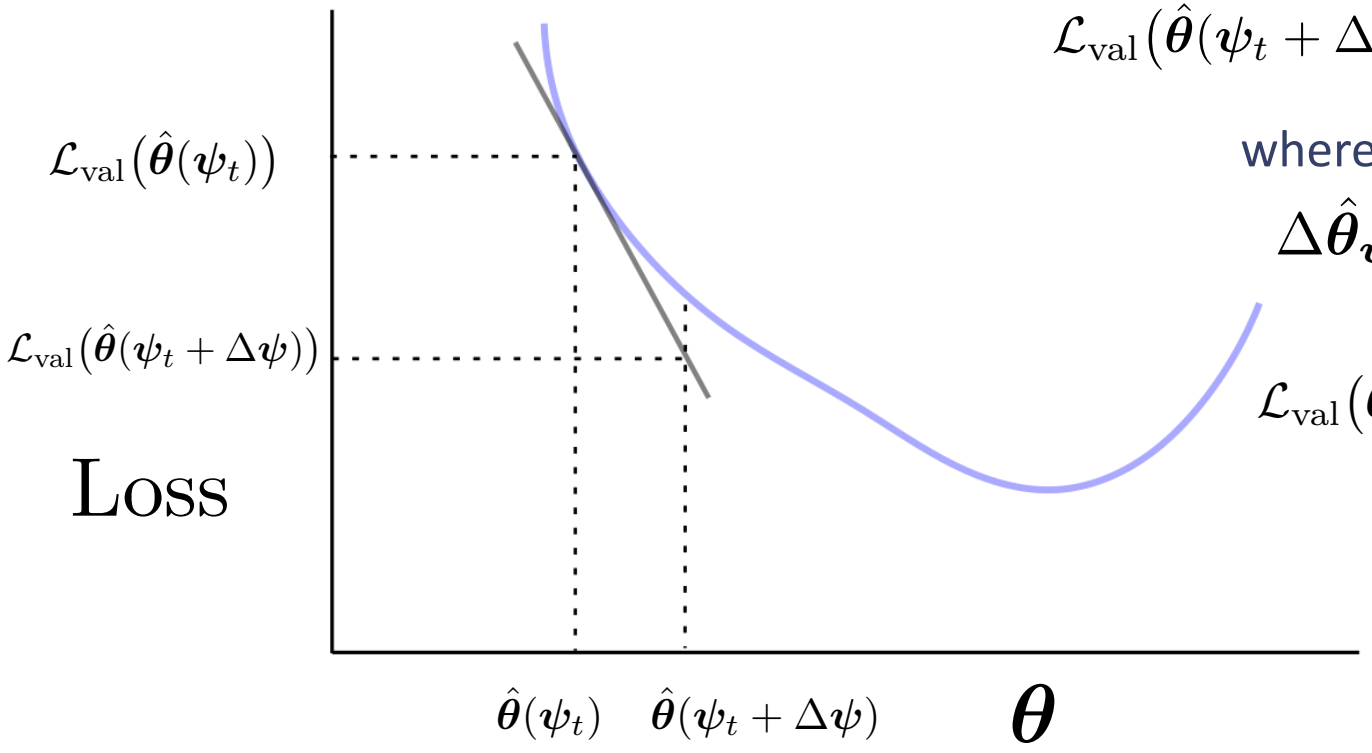
Objective: Minimize $\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)$ with respect to $\boldsymbol{\psi}$

$$\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + \Delta\boldsymbol{\psi})\right) = \mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right) + \Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \frac{d\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)}{d\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)} \quad (2)$$



# AutoSimulate

Objective: Minimize $\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)$ with respect to $\boldsymbol{\psi}$



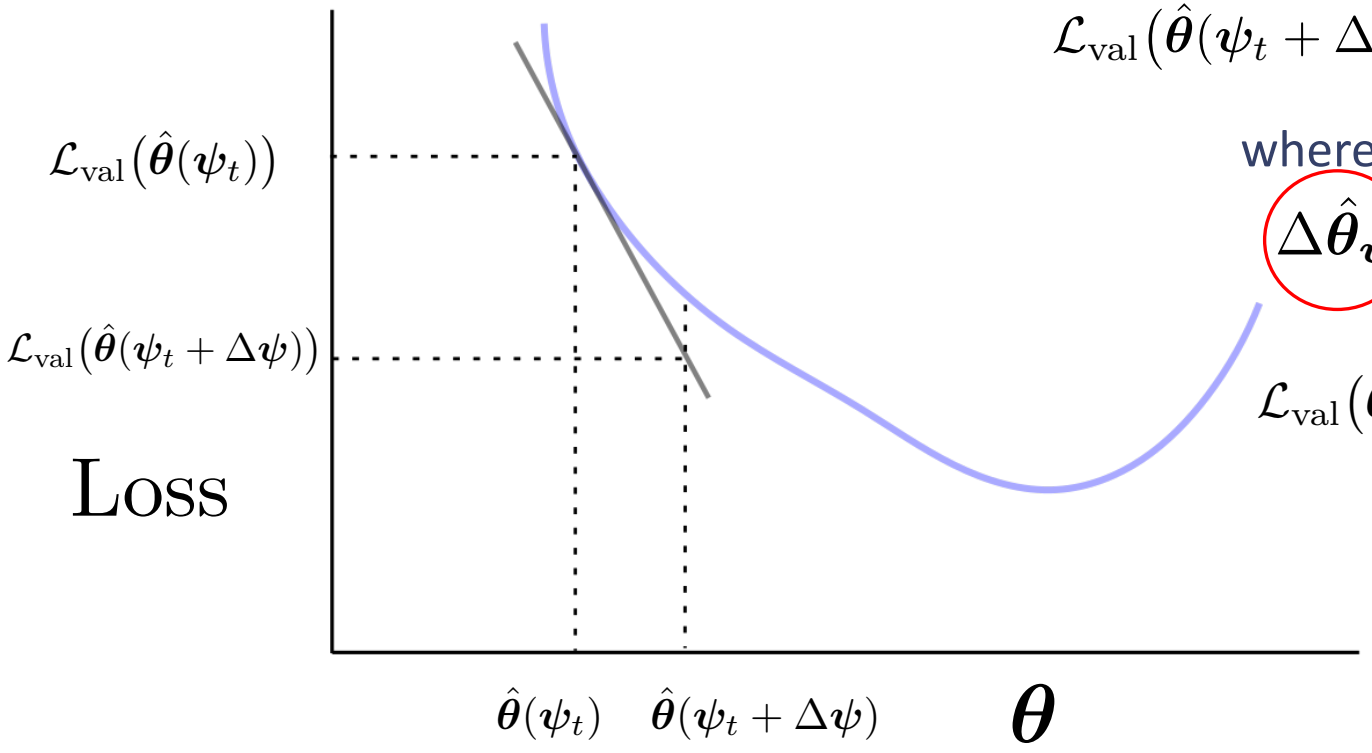

$$\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + \Delta\boldsymbol{\psi})\right) = \mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right) + \Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \frac{d\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)}{d\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)} \qquad (2)$$

where

$$\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} = \Delta\boldsymbol{\psi} \frac{d\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)}{d\boldsymbol{\psi}} \approx \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + d\boldsymbol{\psi}) - \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)$$

# AutoSimulate

Objective: Minimize $\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)$ with respect to $\boldsymbol{\psi}$




$$\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + \Delta\boldsymbol{\psi})\right) = \mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right) + \Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \frac{d\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)}{d\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)} \quad (2)$$

where

$$\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} = \Delta\boldsymbol{\psi} \frac{d\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)}{d\boldsymbol{\psi}} \approx \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + d\boldsymbol{\psi}) - \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)$$

How to find?

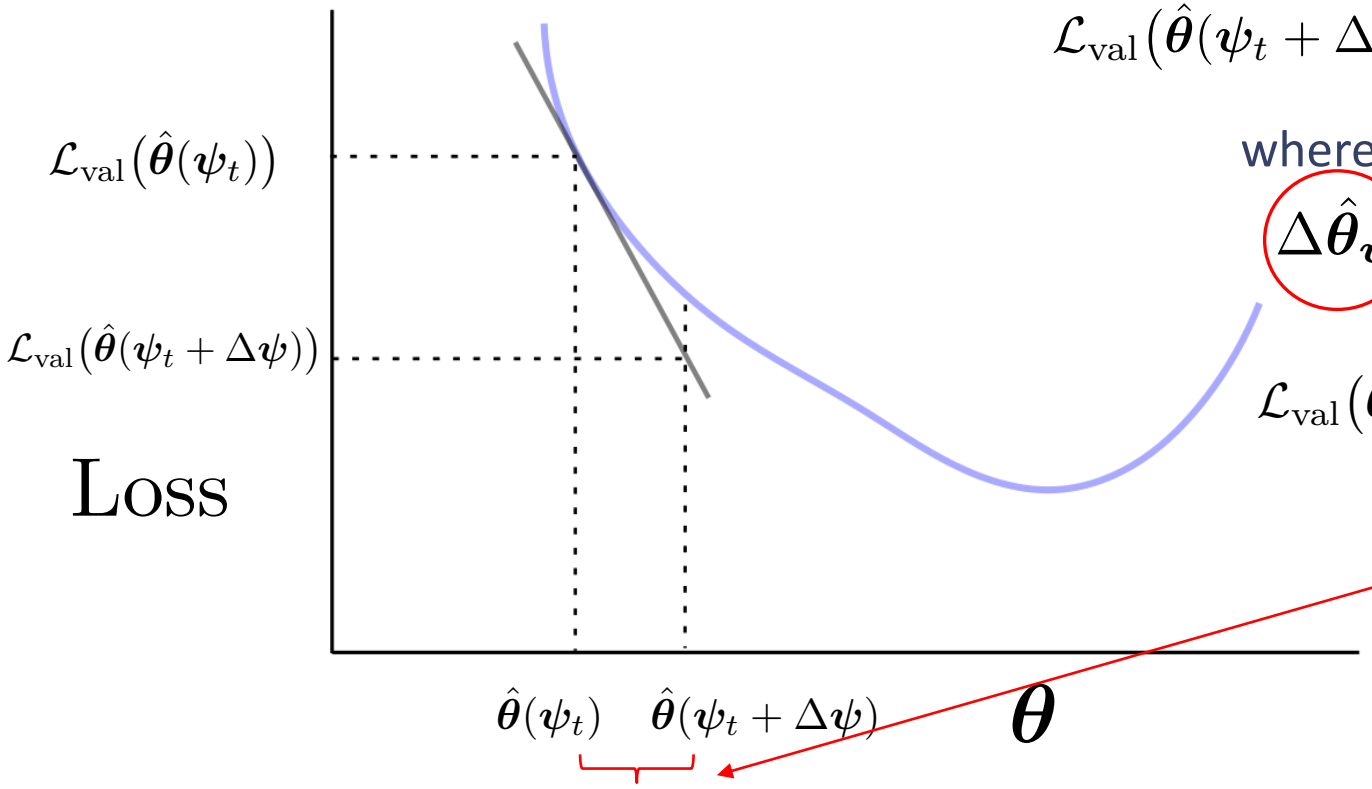
# AutoSimulate

Objective: Minimize $\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)$ with respect to $\boldsymbol{\psi}$

$$\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + \Delta\boldsymbol{\psi})\right) = \mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right) + \Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \frac{d\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)}{d\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)} \quad (2)$$

where

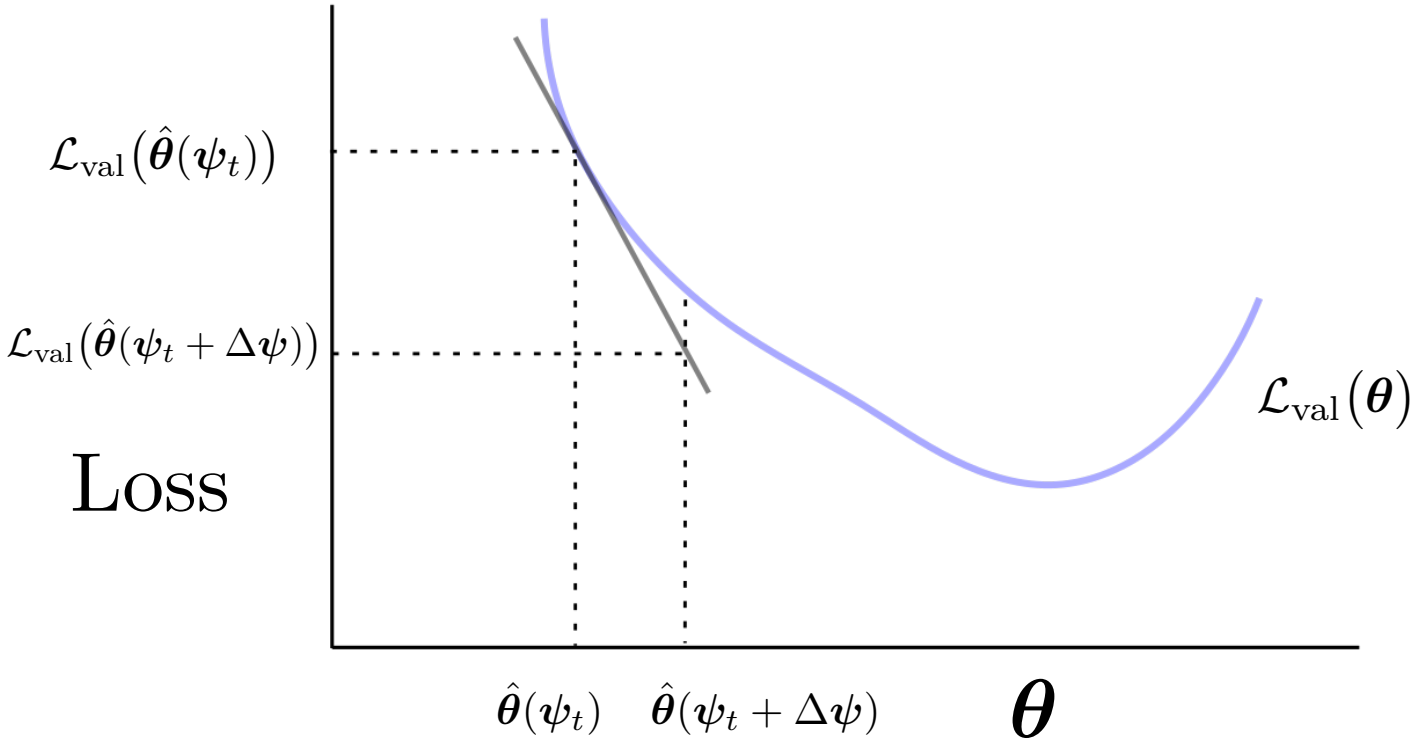
$$\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} = \Delta\boldsymbol{\psi} \frac{d\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)}{d\boldsymbol{\psi}} \approx \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + d\boldsymbol{\psi}) - \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)$$




How to find?

# AutoSimulate

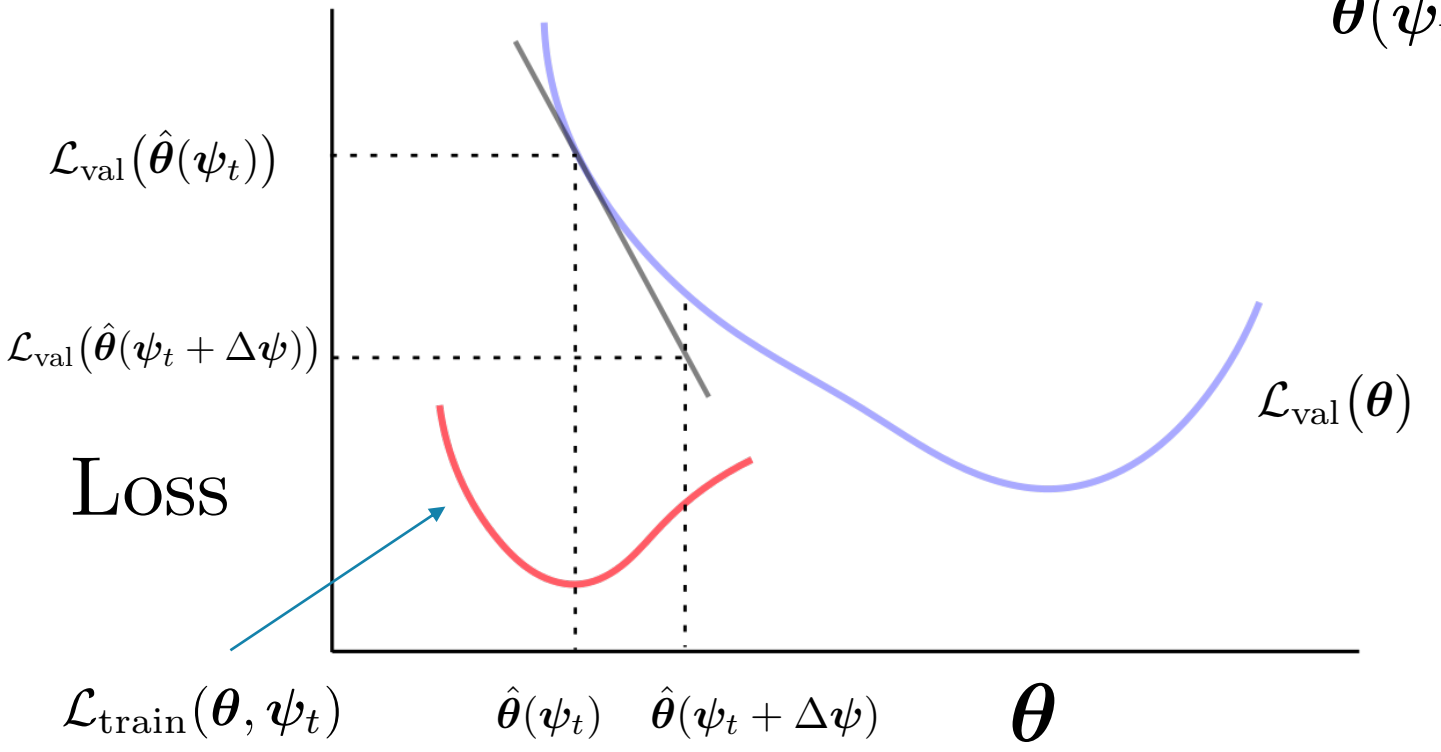
Find: $\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \approx \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + d\boldsymbol{\psi}) - \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)$



# AutoSimulate

Find: $\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \approx \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + d\boldsymbol{\psi}) - \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)$

$$\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t) = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\text{train}}(\boldsymbol{\theta}, \boldsymbol{\psi}_t)$$

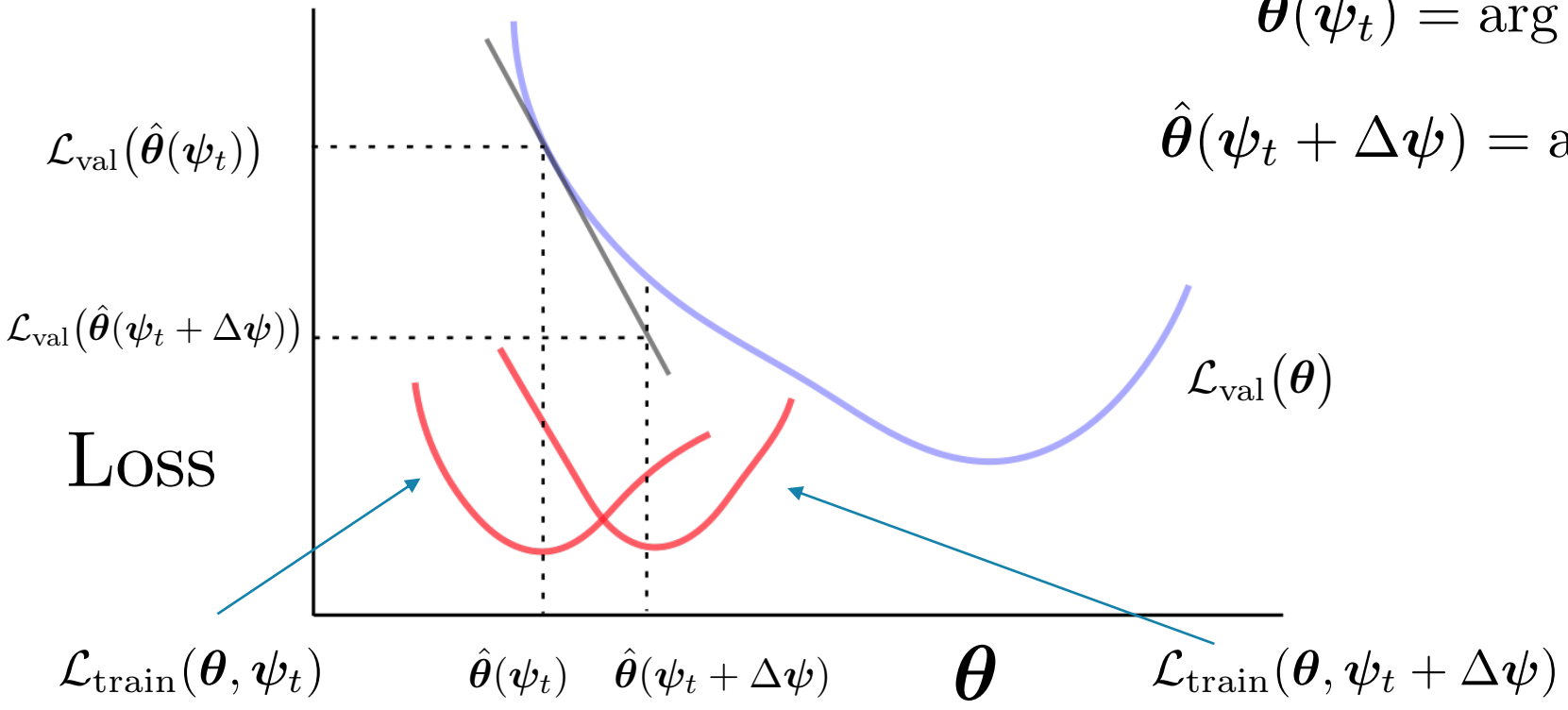


# AutoSimulate

Find: $\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \approx \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + d\boldsymbol{\psi}) - \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)$

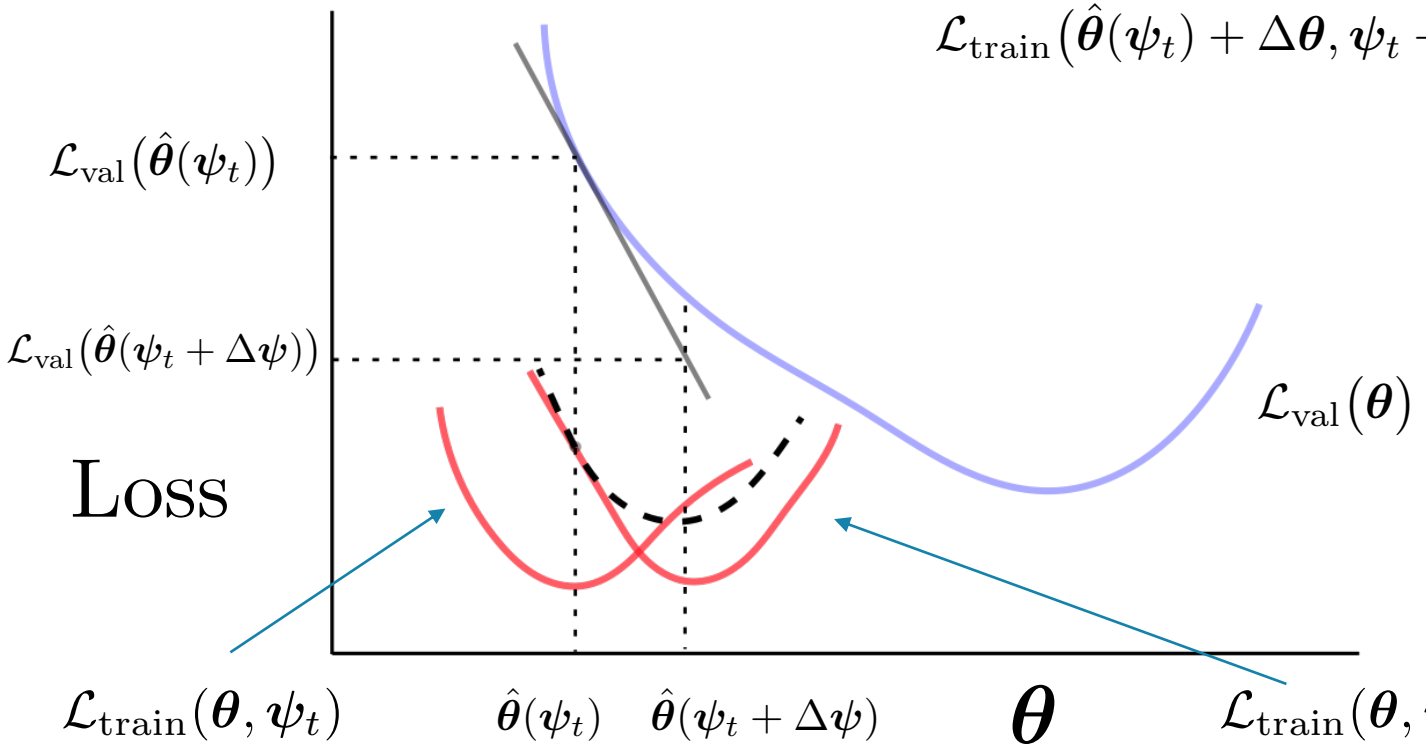
$$\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t) = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\text{train}}(\boldsymbol{\theta}, \boldsymbol{\psi}_t)$$

$$\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}) = \arg\min_{\boldsymbol{\theta}} \mathcal{L}_{\text{train}}(\boldsymbol{\theta}, \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi})$$



# AutoSimulate

Find: $\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \approx \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + d\boldsymbol{\psi}) - \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)$



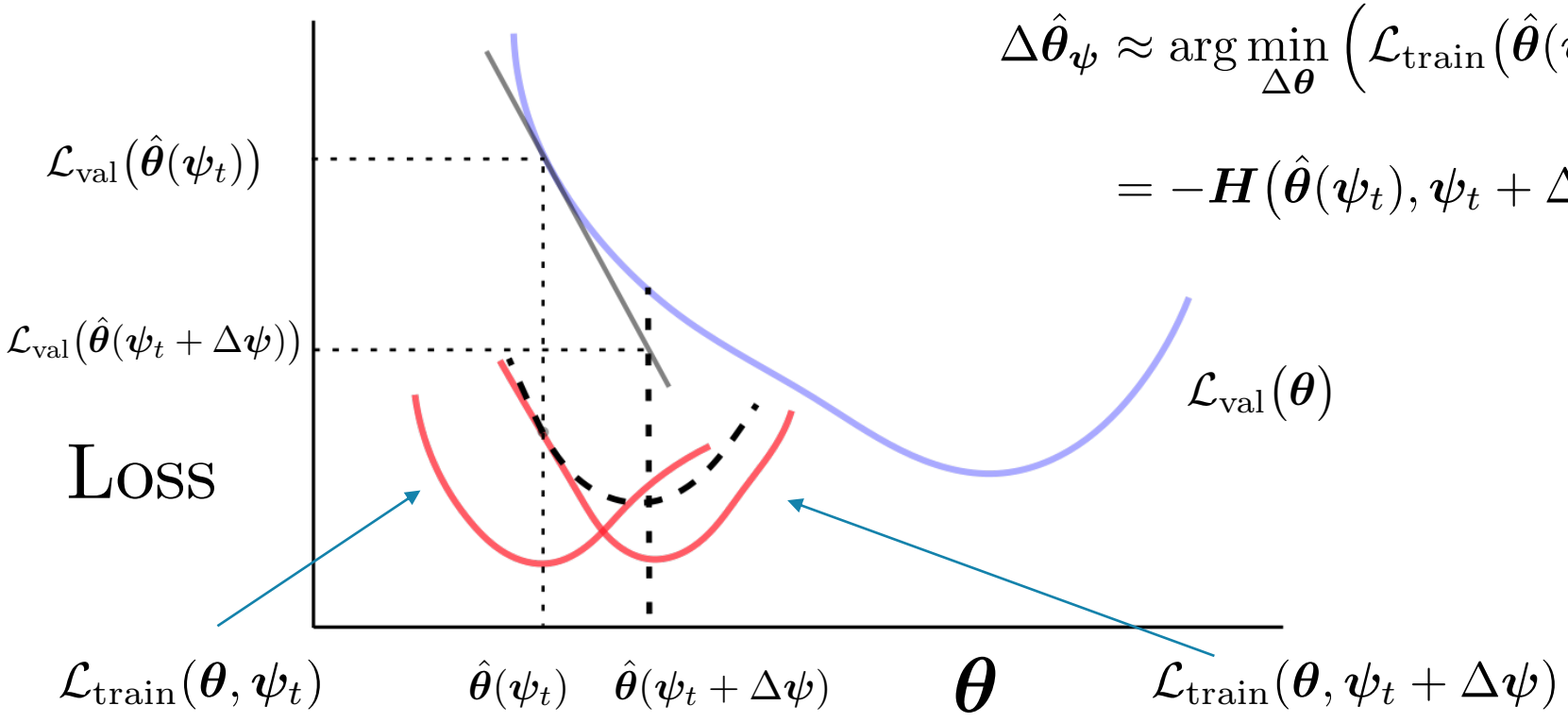

$$\mathcal{L}_{\text{train}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t) + \Delta\boldsymbol{\theta}, \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\big) = \mathcal{L}_{\text{train}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\big) + \Delta\boldsymbol{\theta}^\top \frac{\partial}{\partial\boldsymbol{\theta}} \mathcal{L}_{\text{train}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\big) + \frac{1}{2}\Delta\boldsymbol{\theta}^\top \boldsymbol{H}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\big)\Delta\boldsymbol{\theta} + \ldots$$

# AutoSimulate

Find: $\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \approx \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + d\boldsymbol{\psi}) - \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)$

$$\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \approx \arg\min_{\Delta\boldsymbol{\theta}} \left( \mathcal{L}_{\text{train}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t) + \Delta\boldsymbol{\theta}, \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\right) \right)$$

$$= -\boldsymbol{H}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\right)^{-1} \frac{\partial \mathcal{L}_{\text{train}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\right)}{\partial \boldsymbol{\theta}}$$

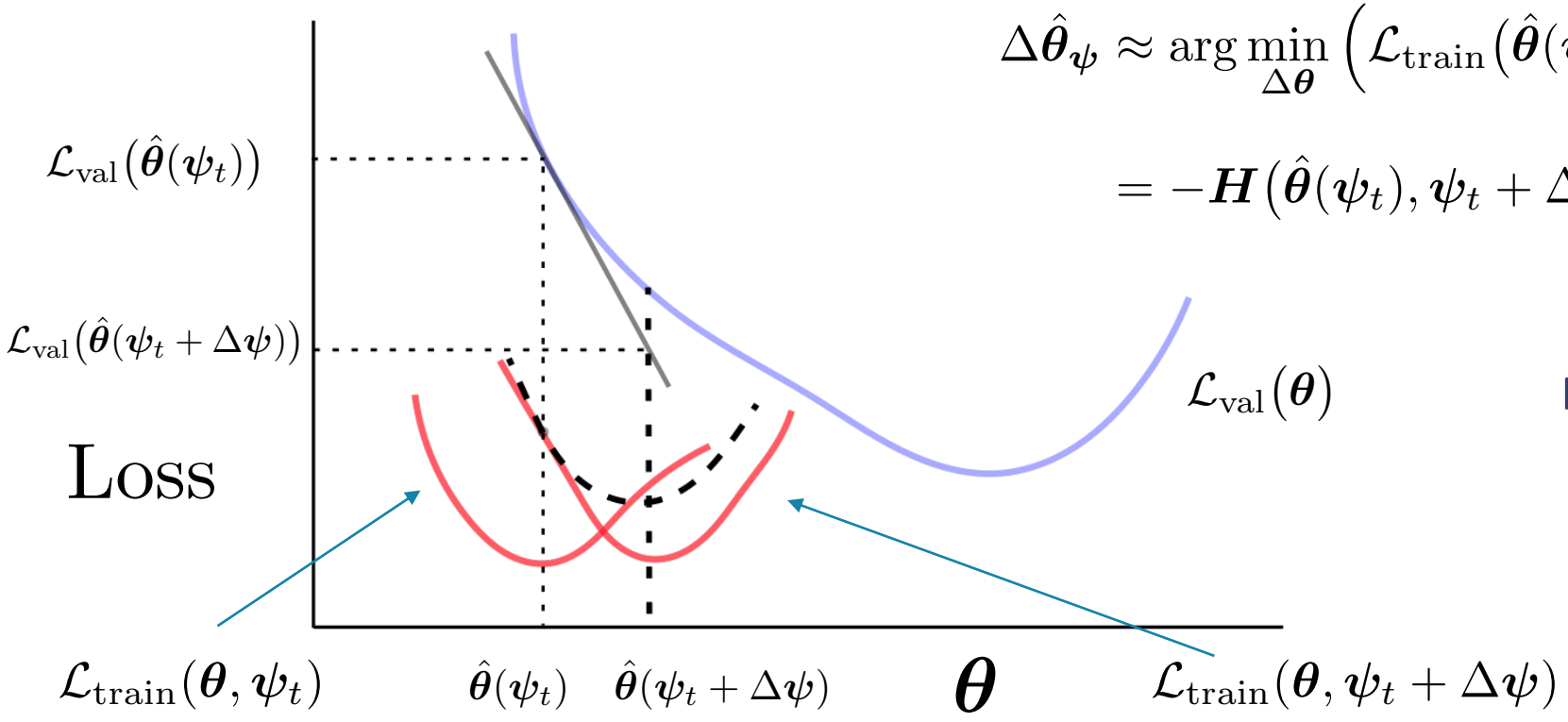


# AutoSimulate

Find: $\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \approx \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + d\boldsymbol{\psi}) - \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)$

$$\Delta\hat{\boldsymbol{\theta}}_{\boldsymbol{\psi}} \approx \arg\min_{\Delta\boldsymbol{\theta}} \left( \mathcal{L}_{\text{train}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t) + \Delta\boldsymbol{\theta}, \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\right) \right)$$

$$= -\boldsymbol{H}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\right)^{-1} \frac{\partial \mathcal{L}_{\text{train}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\right)}{\partial \boldsymbol{\theta}}$$




Putting back into Eq.2

# AutoSimulate

Differentiable Approximation:

$$\widetilde{\mathcal{L}}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + \Delta\boldsymbol{\psi})\big) = \mathcal{L}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\big) - \frac{\partial \mathcal{L}_{\text{train}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\big)}{\partial \boldsymbol{\theta}}^{\top} \boldsymbol{H}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\big)^{-1} \frac{d\mathcal{L}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\big)}{d\boldsymbol{\theta}}$$

# AutoSimulate

Differentiable Approximation:

$$\widetilde{\mathcal{L}}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t + \Delta\boldsymbol{\psi})\big) = \mathcal{L}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\big) - \frac{\partial \mathcal{L}_{\text{train}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\big)}{\partial \boldsymbol{\theta}}^{\top} \boldsymbol{H}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t + \Delta\boldsymbol{\psi}\big)^{-1} \frac{d\mathcal{L}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\big)}{d\boldsymbol{\theta}}$$

$$\Rightarrow \widetilde{\mathcal{L}}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\big) = \mathcal{L}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\big) - \frac{\partial \mathcal{L}_{\text{train}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\big)}{\partial \boldsymbol{\theta}}^{\top} \boldsymbol{H}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\big)^{-1} \frac{d\mathcal{L}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\big)}{d\boldsymbol{\theta}}$$

# AutoSimulate

Differentiable Approximation:

$$\widetilde{\mathcal{L}}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\big) = \mathcal{L}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\big) - \frac{\partial \mathcal{L}_{\text{train}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\big)}{\partial \boldsymbol{\theta}}^{\top} \boldsymbol{H}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\big)^{-1} \frac{d\mathcal{L}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\big)}{d\boldsymbol{\theta}}$$

Derivative:

$$\frac{\partial \widetilde{\mathcal{L}}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\big)}{\partial \boldsymbol{\psi}}\bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} = -\frac{\partial}{\partial \boldsymbol{\psi}}\left[\frac{\partial \mathcal{L}_{\text{train}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\big)}{\partial \boldsymbol{\theta}}^{\top}\right]\bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} \boldsymbol{H}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\big)^{-1} \frac{d\mathcal{L}_{\text{val}}\big(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\big)}{d\boldsymbol{\theta}}$$

# AutoSimulate

Derivative:

$$\frac{\partial \widetilde{\mathcal{L}}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)}{\partial \boldsymbol{\psi}}\bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} = -\frac{\partial}{\partial \boldsymbol{\psi}}\left[\frac{\partial \mathcal{L}_{\text{train}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\right)}{\partial \boldsymbol{\theta}}^{\top}\right]\bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} \boldsymbol{H}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\right)^{-1} \frac{d\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)}{d\boldsymbol{\theta}}$$

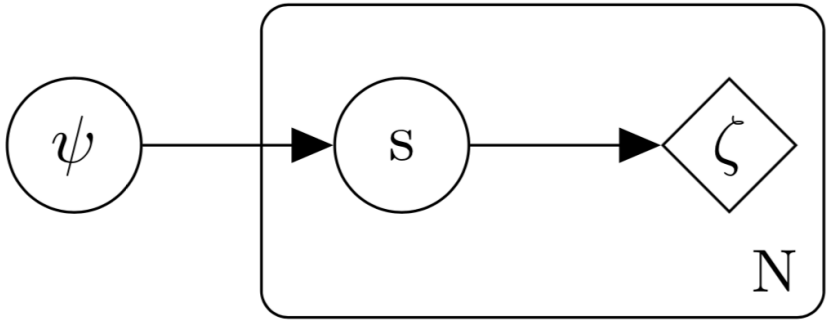
How to find?

# Stochastic Simulator

- The term requires backpropagation through the dataset generation.

$$\frac{\partial}{\partial \boldsymbol{\psi}}\left[\frac{\partial \mathcal{L}_{\text{train}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\right)}{\partial \boldsymbol{\theta}}^{\top}\right] = \frac{\partial}{\partial \boldsymbol{\psi}} \mathbb{E}_{\zeta \sim p_{\psi}}\left[\frac{\partial}{\partial \boldsymbol{\theta}} l\left(\zeta, \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)\right]$$

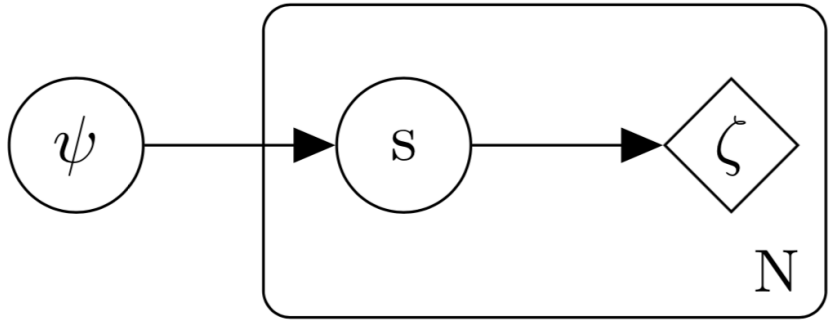
- We assume a stochastic simulator that involves a deterministic renderer.
- We make the stochasticity in the process explicit by separating the stochastic part of the simulator from the deterministic rendering.



# Stochastic Simulator

Given the deterministic renderer component $\zeta = \mathrm{r}(s)$, we would like to find the optimal values of simulator parameters $\boldsymbol{\psi}$ that parameterize $s \sim q_{\boldsymbol{\psi}}(s)$ representing the stochastic component, expressing the overall simulator as $\zeta \sim p_{\boldsymbol{\psi}}(\zeta)$.

$$p_{\boldsymbol{\psi}}(\zeta) = \int_{s \in \{s | r(s) = \zeta\}} q_{\boldsymbol{\psi}}(s) ds$$



# AutoSimulate

Derivative:

$$\left.\frac{\partial \widetilde{\mathcal{L}}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)}{\partial \boldsymbol{\psi}}\right|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} = -\frac{\partial}{\partial \boldsymbol{\psi}}\left[\frac{\partial \mathcal{L}_{\text{train}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\right)}{\partial \boldsymbol{\theta}}^{\top}\right]\Bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} \boldsymbol{H}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\right)^{-1} \frac{d\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)}{d\boldsymbol{\theta}}$$

How to find?

How to find?

# AutoSimulate

Derivative:

$$\frac{\partial \widetilde{\mathcal{L}}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)}{\partial \boldsymbol{\psi}}\bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} = -\frac{\partial}{\partial \boldsymbol{\psi}}\left[\frac{\partial \mathcal{L}_{\text{train}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\right)}{\partial \boldsymbol{\theta}}^{\top}\right]\bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} \boldsymbol{H}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\right)^{-1} \frac{d\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)}{d\boldsymbol{\theta}}$$

How to find?

See paper for more details

How to find?

# Approximations for $\Delta\hat{\theta}_{\psi}$

- Earlier we proposed a quadratic approximation.
- To further reduce the compute overhead, we propose more approximations.

| | Approximation ($\Delta\hat{\theta}_{\psi}$) | Derivative Term ($\frac{\partial}{\partial \boldsymbol{\psi}} \widetilde{\mathcal{L}}_{\text{val}}(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}))$) |
|---|---|---|
| Quadratic | $-H(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi})^{-1} \frac{\partial}{\partial \boldsymbol{\theta}} \mathcal{L}_{\text{tr.}}(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi})$ | $-\frac{\partial}{\partial \boldsymbol{\psi}} \mathbb{E}_{\zeta \sim p_{\psi}}[\frac{\partial}{\partial \boldsymbol{\theta}} l(\zeta, \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t))]^{\top} \Big\vert_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} \boldsymbol{H}(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t)^{-1} \frac{d}{d\boldsymbol{\theta}} \mathcal{L}_{\text{val}}(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t))$ |
| Approx. Quadratic | $H(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}) \frac{\partial}{\partial \boldsymbol{\theta}} \mathcal{L}_{\text{tr.}}(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi})$ | $\frac{\partial}{\partial \boldsymbol{\psi}} \mathbb{E}_{\zeta \sim p_{\psi}}[\frac{\partial}{\partial \boldsymbol{\theta}} l(\zeta, \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t))]^{\top} \Big\vert_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} \boldsymbol{H}(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t) \frac{d}{d\boldsymbol{\theta}} \mathcal{L}_{\text{val}}(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t))$ |
| Linear | $-\frac{\partial}{\partial \boldsymbol{\theta}} \mathcal{L}_{\text{tr.}}(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi})$ | $-\frac{\partial}{\partial \boldsymbol{\psi}} \mathbb{E}_{\zeta \sim p_{\psi}}[\frac{\partial}{\partial \boldsymbol{\theta}} l(\zeta, \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t))]^{\top} \Big\vert_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} \frac{d}{d\boldsymbol{\theta}} \mathcal{L}_{\text{val}}(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t))$ |
| No Val | $\mathbf{1}$ | $-\frac{\partial}{\partial \boldsymbol{\psi}} \mathbb{E}_{\zeta \sim p_{\psi}}[\frac{\partial}{\partial \boldsymbol{\theta}} l(\zeta, \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t))]^{\top} \Big\vert_{\boldsymbol{\psi}=\boldsymbol{\psi}_t}$ |

# AutoSimulate

Derivative:

$$\frac{\partial \widetilde{\mathcal{L}}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)}{\partial \boldsymbol{\psi}}\bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} = -\frac{\partial}{\partial \boldsymbol{\psi}}\left[\frac{\partial \mathcal{L}_{\text{train}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\right)}{\partial \boldsymbol{\theta}}^{\top}\right]\bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} \boldsymbol{H}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}\right)^{-1} \frac{d\mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)}{d\boldsymbol{\theta}}$$

Optimizing Simulator:

$$\boldsymbol{\psi}_{t+1} \leftarrow \boldsymbol{\psi}_t - \alpha \frac{\partial \widetilde{\mathcal{L}}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi})\right)}{\partial \boldsymbol{\psi}}\bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t}$$

# Algorithm: AutoSimulate

**Algorithm 1:** AutoSimulate

**for** number of iterations **do**

Sample dataset of size K: $D_{\text{train}} \sim p_{\boldsymbol{\psi_t}}(\zeta)$

Fine-tune model for $\epsilon$ epochs on $D_{\text{train}}$

Compute $\boldsymbol{H}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t\right)^{-1} \frac{d}{d\boldsymbol{\theta}} \mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)$ using CG

Compute gradient of expectation as $\sum_{k=1}^{K} \frac{d}{d\boldsymbol{\psi}} \log q_{\boldsymbol{\psi}}(s_k) . \left[\frac{\partial}{\partial \boldsymbol{\theta}} l\left(r(s_k), \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)\right]^{\top}$

Update simulator by descending the gradient

$$-\frac{\partial}{\partial \boldsymbol{\psi}} \mathbb{E}_{\zeta \sim p_{\psi}} \left[\frac{\partial}{\partial \boldsymbol{\theta}} l\left(\zeta, \hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)\right]^{\top} \Bigg|_{\boldsymbol{\psi}=\boldsymbol{\psi}_t} \boldsymbol{H}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t), \boldsymbol{\psi}_t\right)^{-1} \frac{d}{d\boldsymbol{\theta}} \mathcal{L}_{\text{val}}\left(\hat{\boldsymbol{\theta}}(\boldsymbol{\psi}_t)\right)$$

**end for**

# Experiments

- Optimizing photorealistic Arnold renderer (non-differentiable)
- For object detection on real-world dataset LM-O (with 12 classes)

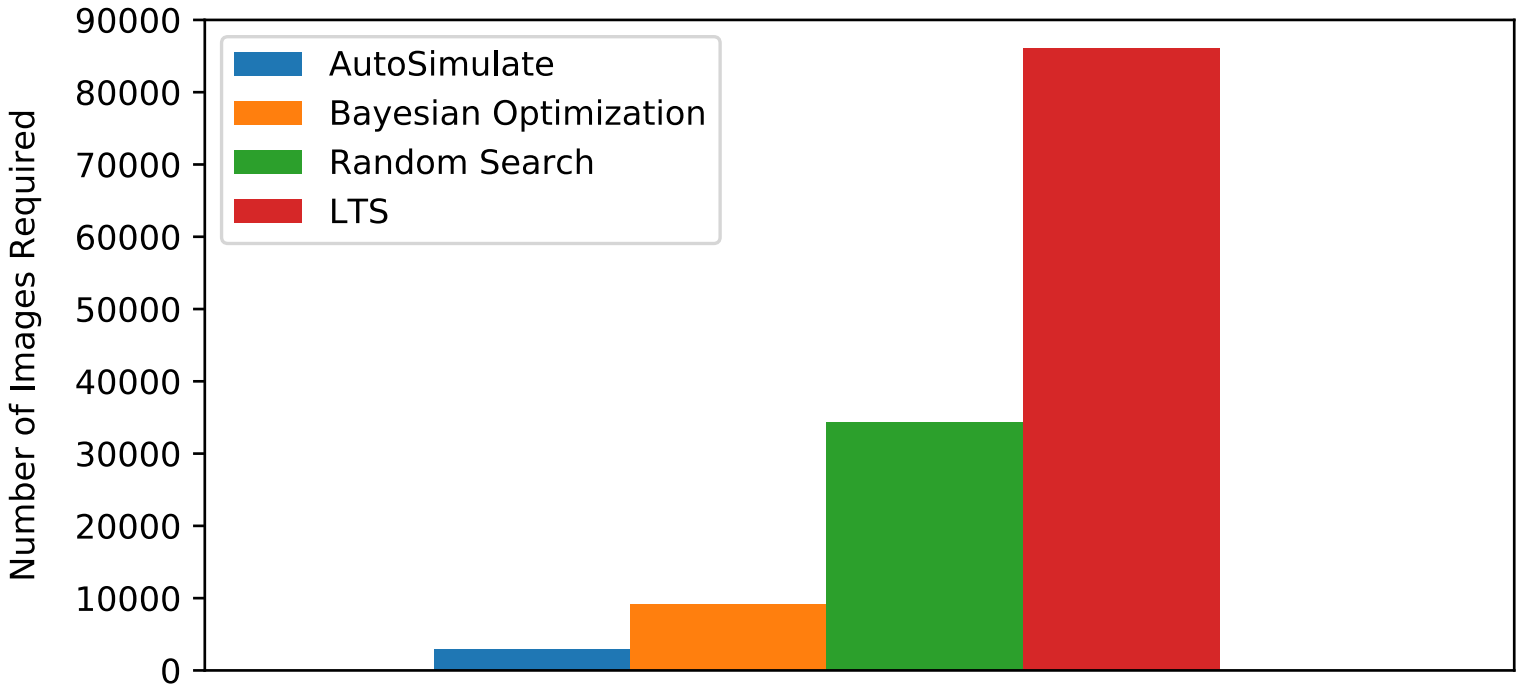
# Results: Faster

50x

# Results: More Accurate (mAP)

+8.5%

# Results: Requires Lesser Data

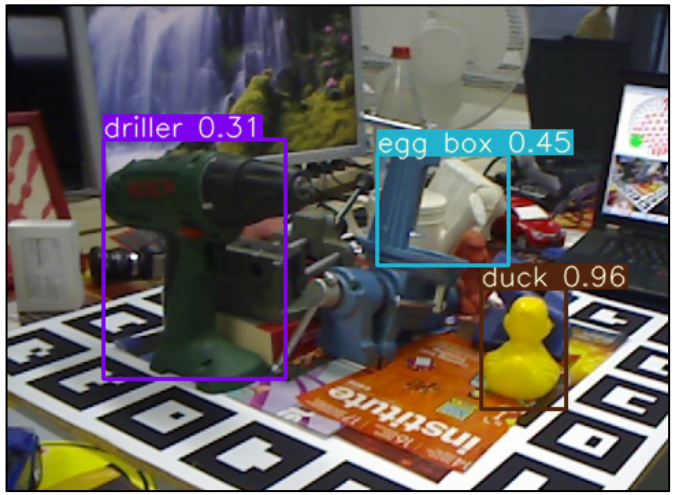


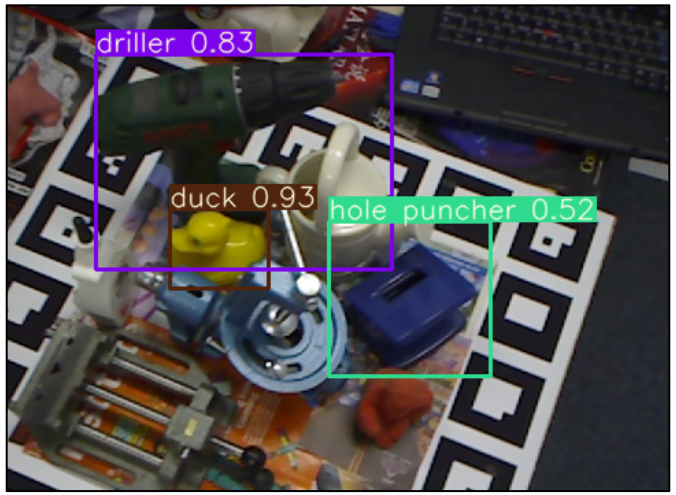
# Real World Object Detection

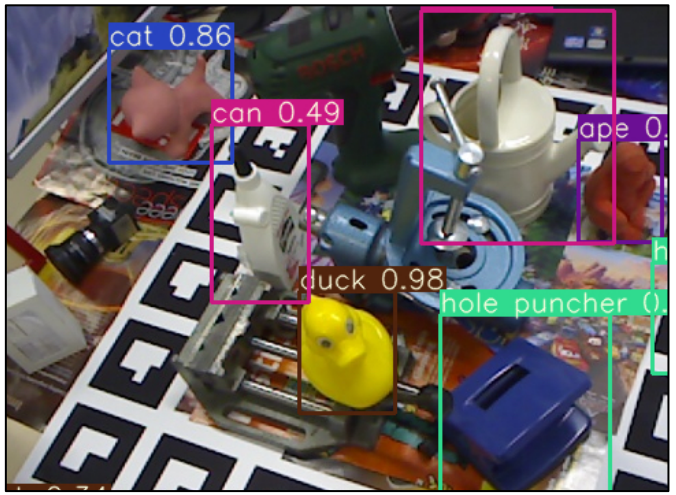
- We run each method for a 1,000 epochs
- Val. mAP: maximum validation mAP, Test mAP: test mAP.
- Images and Time: number of images generated and time spent to reach maximum validation mAP.

| Method | Val. mAP | Images | Time(s) | Test mAP |
|---|---|---|---|---|
| REINFORCE (LTS) | 40.2 | 86,150 | 114,360 | 37.2 |
| Bayesian Optimization | 39.3 | 9,200 | 83,225 | 37.5 |
| Random Search | 40.3 | 34,300 | 134,318 | 37.0 |
| AutoSimulate | 37.1 | 8,950 | 23,193 | 36.1 |
| AutoSimulate(Approx Quad) | 40.1 | **2,950** | **2,321** | 37.4 |
| AutoSimulate (Linear) | **41.4** | 17,850 | 30,477 | **45.9** |

# Results: Object Detections



# Ablation: Generalization

- In this ablation, we study whether a simulator trained on a shallow network generalizes to a deeper network.
- Simulator was trained on Yolo and tested on Faster-rcnn.

| Method | mAP (Faster-rcnn) | mAP (Yolo) |
|---|---|---|
| REINFORCE (LTS) | 51.8 | 37.2 |
| Bayesian Optimization | 46.0 | 37.5 |
| Random Search | 50.3 | 36.8 |
| Ours | **55.1** | **45.9** |

# Ablation: Effect of Freezing Layers

- It is a common practise to train on synthetic data with the initial layers of the network frozen and trained on real data.
- We see the relative performance of different simulator optimization methods under this scenario

| | 0 frozen layers | | | 98 frozen layers | | | 104 frozen layers | | |
|---|---|---|---|---|---|---|---|---|---|
| Method | mAP | Time(s) | Images | mAP | Time(s) | Images | mAP | Time(s) | Images |
| REINFORCE (LTS) | 37.2 | 114,360 | 86,150 | 33.0 | 114,360 | 86,150 | 31.9 | 145,193 | 104,600 |
| Bayesian Optimization | 37.5 | 83,225 | **9,225** | 31.7 | 13,940 | 3,550 | 31.7 | 30,538 | 6,050 |
| Random Search | 36.8 | 134,137 | 34,300 | 30.2 | 8,913 | 3,500 | 28.9 | 73,411 | 21,650 |
| Ours | **45.9** | **30,477** | 17,850 | **37.1** | **2,321** | **2,950** | **35.8** | **958** | **1,000** |

# Ablation: Effect of Network Size

- We first examine the effect of network depth on simulator training
- Our method is even faster in optimizing smaller networks.

| | Yolo-spp | | | Yolo-Tiny | | |
|---|---|---|---|---|---|---|
| Method | mAP | Time(s) | Images | mAP | Time(s) | Images |
| REINFORCE (LTS) | 37.2 | 114,360 | 86,150 | **24.7** | 3,475 | 11,550 |
| Bayesian Optimization | 37.5 | 83,225 | **9,225** | 19.5 | 65,760 | 35,700 |
| Random Search | 36.8 | 134,137 | 34,300 | 20.6 | 7,319 | 11,620 |
| Ours | **45.9** | **30,477** | 17,850 | 21.2 | **484** | **280** |

# Acknowledgements


- Vibhav Vineet (MSR Redmond)
- Tencent
- Ondrej Miksik (Microsoft AI, Zurich), Tomas Hodan (Czech Technical University Prague)
- Emanuel Shalev, Pedro Urbina (Microsoft Hololens Synthetics Team)
- Prof. M. Pawan Kumar (Oxford)

# Thank You