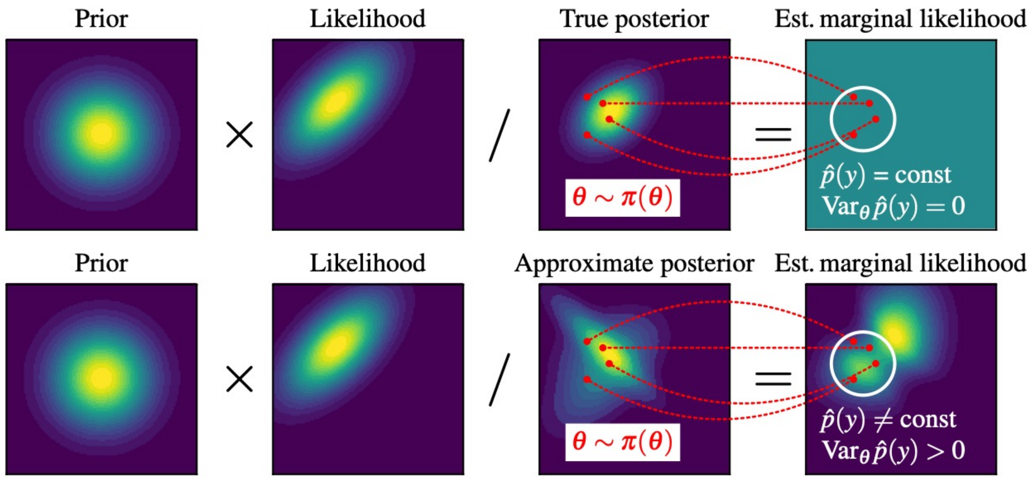



# Towards Reliable
# Amortized Bayesian Inference

*Marvin Schmitt*
*University of Stuttgart, Germany*

@MarvinSchmittML
www.marvinschmitt.com

# Focus of this talk:

Data-Efficient Learning via Self-Consistency Losses

*Extended abstract, NeurIPS UniReps workshop:* *arxiv.org/abs/2310.04395*

Joint work with

Desi Ivanova
Oxford, UK

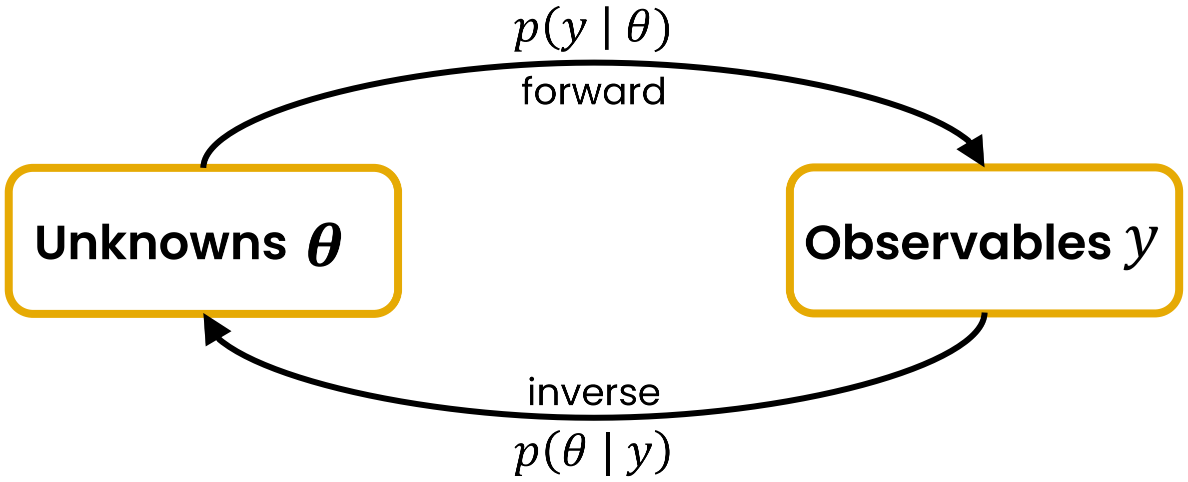
Daniel Habermann
Dortmund, GER

Ullrich Köthe
Heidelberg, GER

Paul Bürkner
Dortmund, GER

Stefan Radev
RPI, US

# Inverse Problems

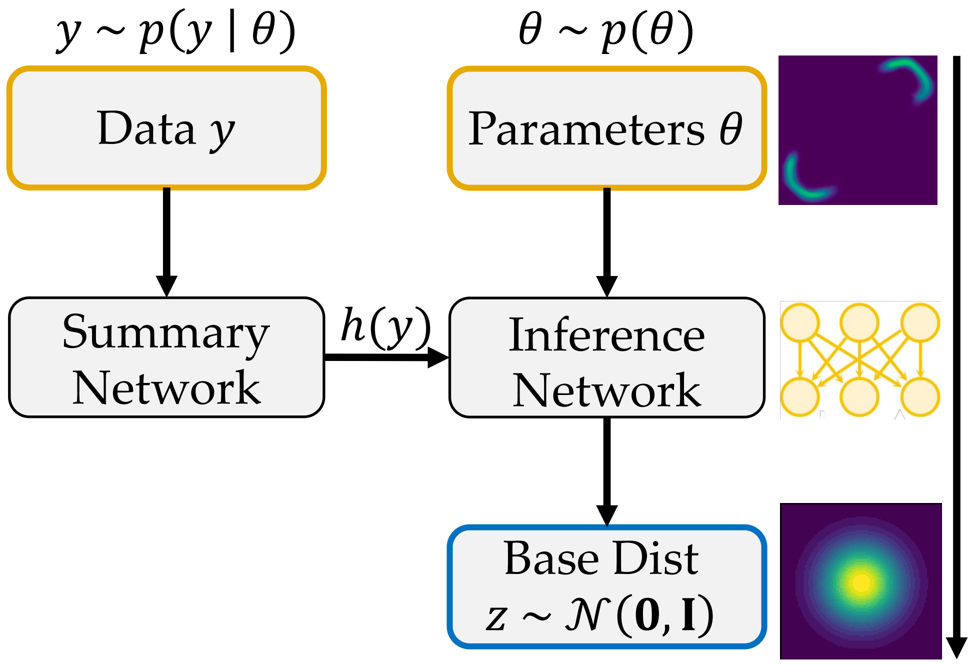



| | | |
|---|---|---|
| **Statistical modeling:** | **Parameters $\boldsymbol{\theta}$** | **Data $\boldsymbol{y}$** |
| **Epidemiology:** | Virus attributes | Infection curve (time series) |
| **Image processing:** | Crisp image | Blurry image |
| **Psychology:** | Cognitive parameters | Reaction times |

# Amortized Bayesian inference

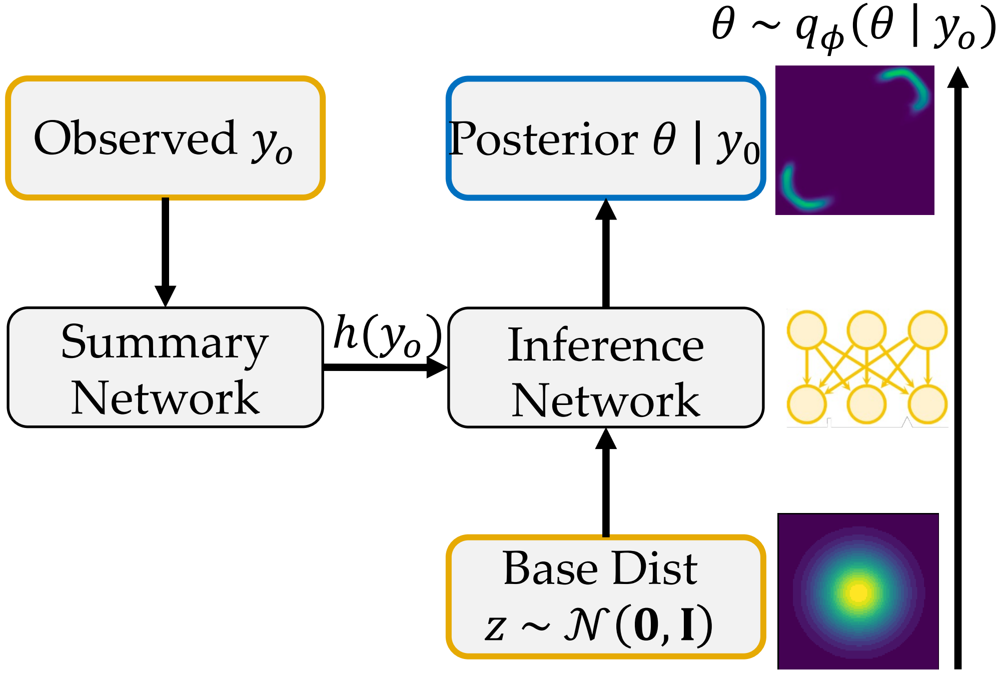
## Stage 1: Training (Approximation)

*potentially expensive*



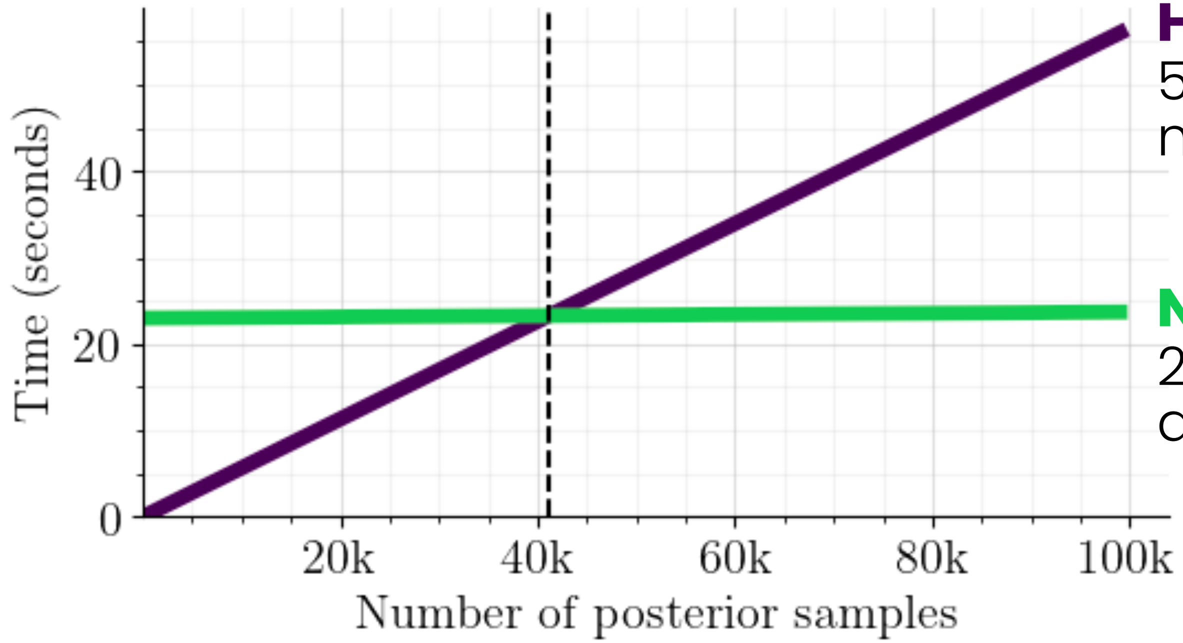

## Stage 2: Inference

*amortized over many data sets* $y_o$




Approximation and inference are **decoupled**. Pooling of resources.

# Potential of Amortized Bayesian Inference




**Hamiltonian Monte Carlo (HMC)**
5.66s for 10k posterior samples
non-amortized

**Neural Posterior Estimation (NPE)**
23s training, 0.07s for 10k posterior samples
amortized

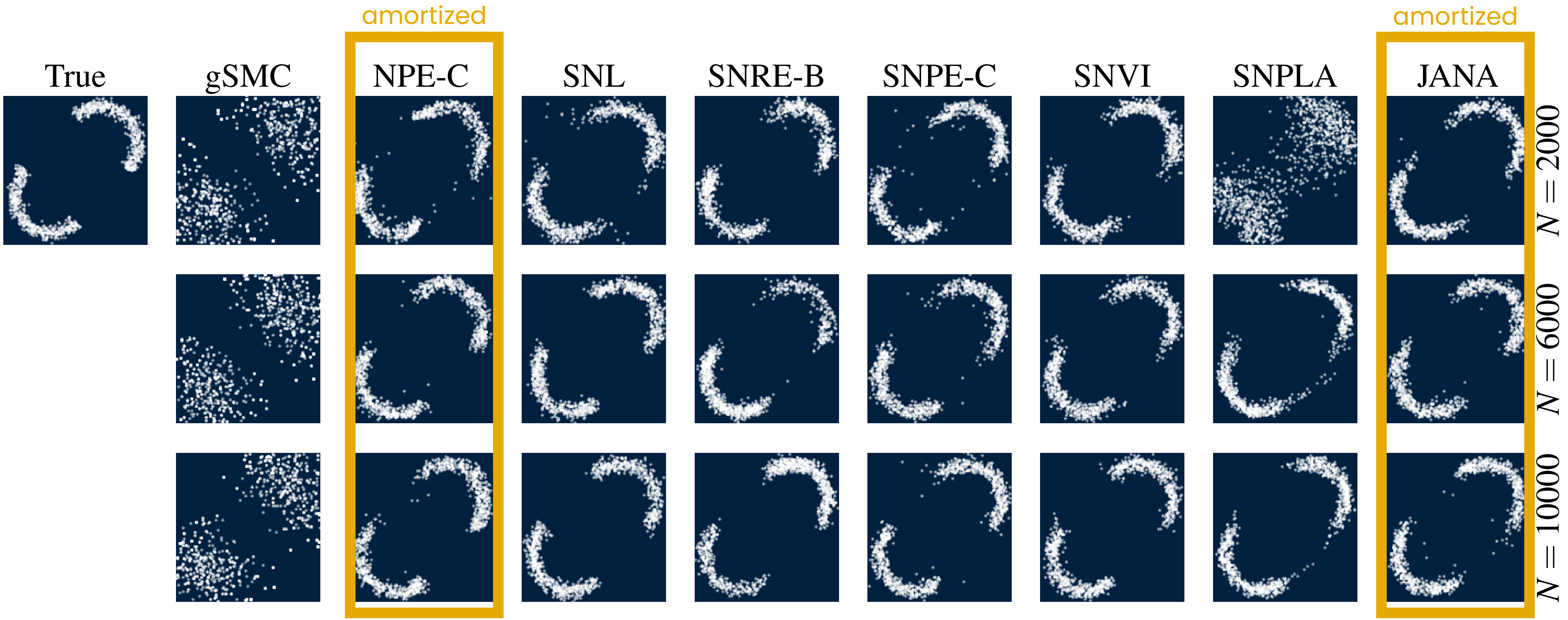
## (1) Many model re-fits

- Cross-validation
- Many data sets
- Sensitivity analyses

## (2) Real-time inference

- Neurological monitoring
- Adaptive experimental design
- Disease surveillance

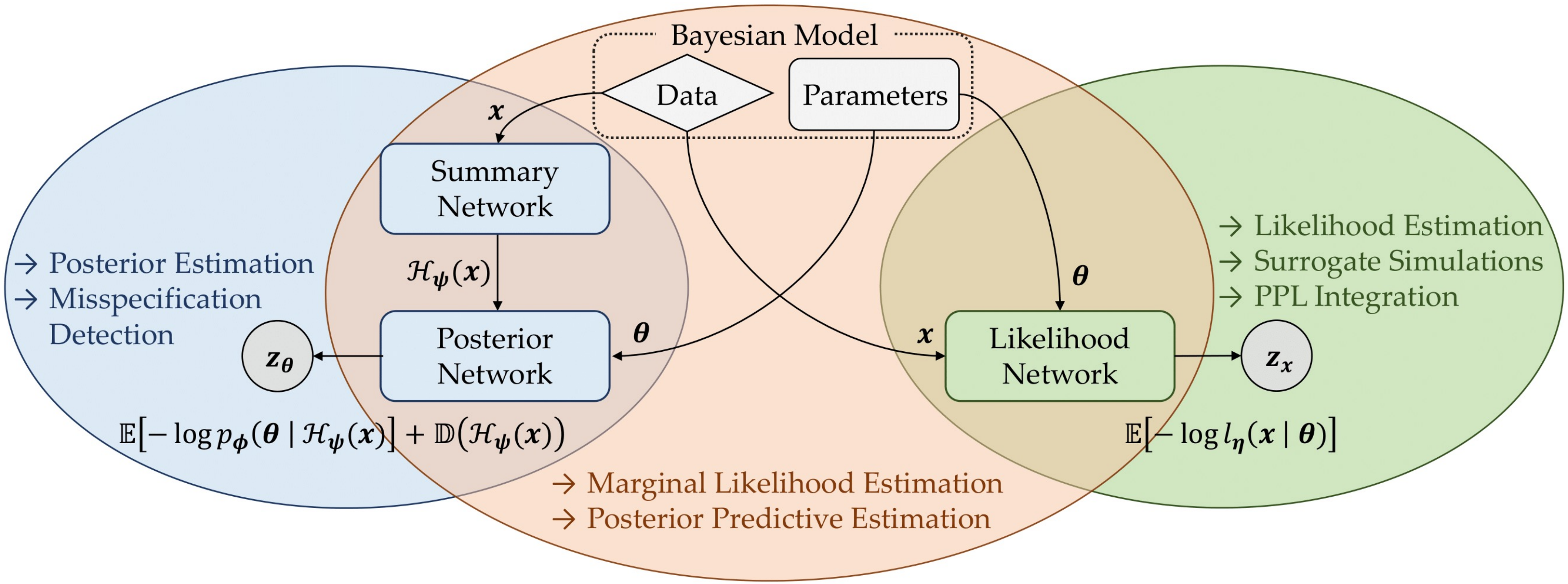
# Isn't amortized inference wasteful? No!




**Amortized** methods perform on-par with non-amortized counterparts!

# Jointly amortized learning: Posterior + Likelihood

- Jointly amortized neural approximation (JANA; Radev et al., 2023)



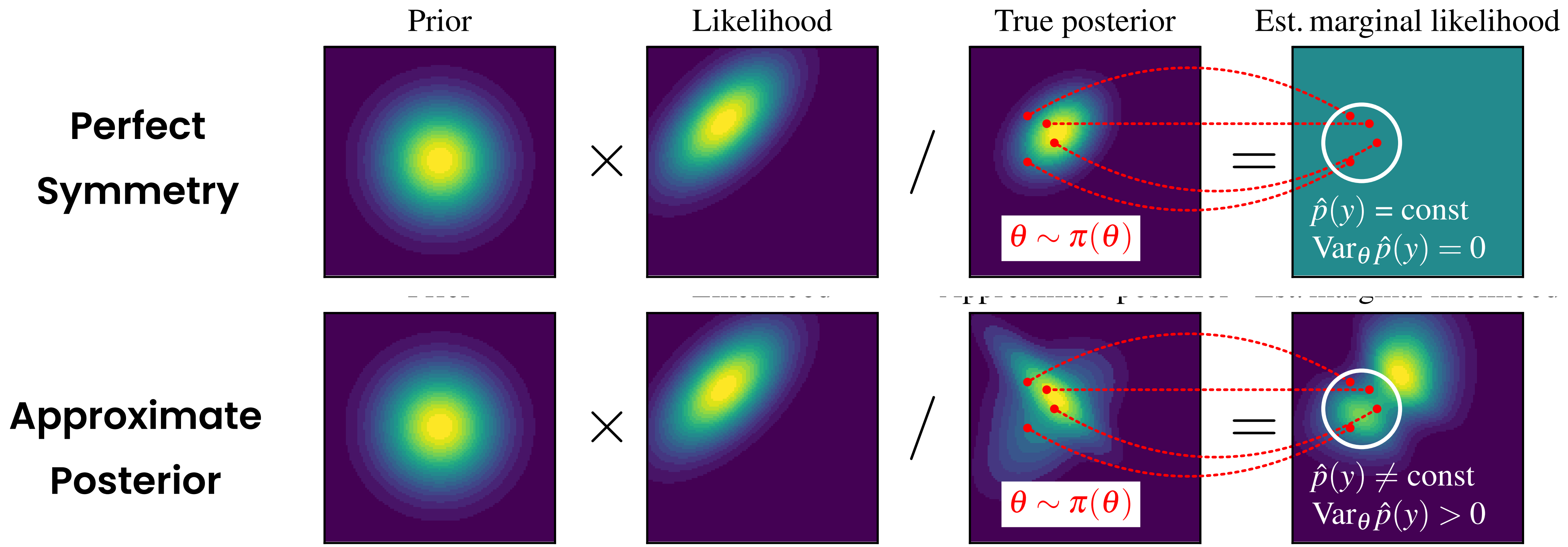
# Problems of vanilla Amortized Bayesian Inference

- Neural networks have a **bad user experience**
- **Model misspecification** invalidates training
- Normalizing flows restrict network **architecture**
- Simulation-based training requires **lots of training data**

# Self-consistency criterion

$$p(\boldsymbol{\theta} \mid \mathbf{Y}) = \frac{p(\boldsymbol{\theta})\, p(\mathbf{Y} \mid \boldsymbol{\theta})}{p(\mathbf{Y})} \iff p(\mathbf{Y}) = \frac{p(\boldsymbol{\theta})\, p(\mathbf{Y} \mid \boldsymbol{\theta})}{p(\boldsymbol{\theta} \mid \mathbf{Y})} \implies \frac{p(\boldsymbol{\theta}_1)\, p(\mathbf{Y} \mid \boldsymbol{\theta}_1)}{p(\boldsymbol{\theta}_1 \mid \mathbf{Y})} = \dots = \frac{p(\boldsymbol{\theta}_K)\, p(\mathbf{Y} \mid \boldsymbol{\theta}_K)}{p(\boldsymbol{\theta}_K \mid \mathbf{Y})}$$

$$\boldsymbol{\theta}_1, \dots, \boldsymbol{\theta}_K \in \Theta$$



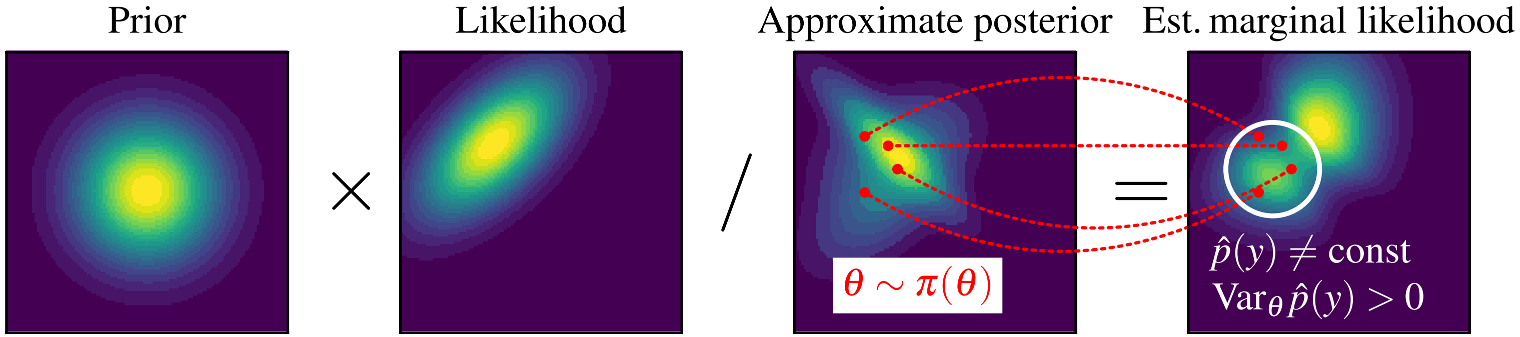
# Self-consistency loss

- **Idea:** Violations of self-consistency property as loss function

$$\mathcal{L}_{\text{SC}}(\mathbf{Y}, \boldsymbol{\phi}) = \text{Var}_{\pi(\boldsymbol{\theta})}\left(\log \frac{p(\boldsymbol{\theta})\, p(\mathbf{Y} \mid \boldsymbol{\theta})}{q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid \mathbf{Y})}\right)$$

- **Integration** into standard neural posterior estimation loss

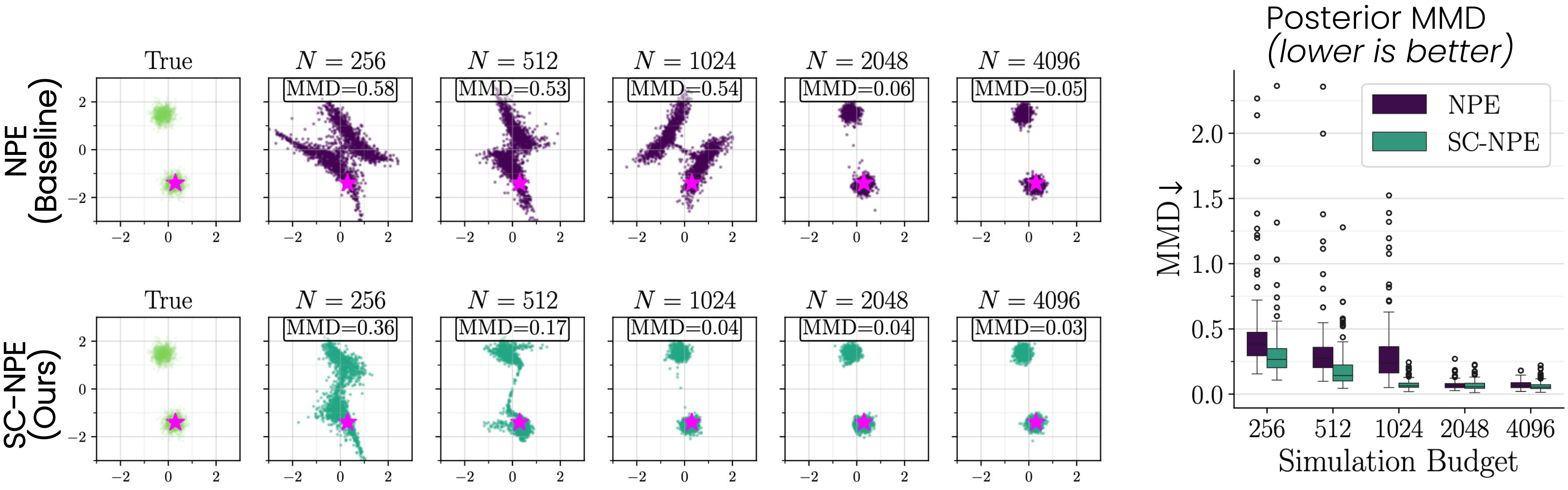
$$\mathcal{L}_{\text{SC-NPE}}(\boldsymbol{\phi}) = \mathbb{E}_{p(\mathbf{Y})}\Bigg[\underbrace{\mathbb{E}_{p(\boldsymbol{\theta} \mid \mathbf{Y})}\big[-\log q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid \mathbf{Y})\big]}_{\text{NPE loss (on fixed } \mathbf{Y})} + \underbrace{\lambda\, \text{Var}_{\pi(\boldsymbol{\theta})}\left(\log \frac{p(\boldsymbol{\theta})p(\mathbf{Y} \mid \boldsymbol{\theta})}{q_{\boldsymbol{\phi}}(\boldsymbol{\theta} \mid \mathbf{Y})}\right)}_{\text{self-consistency loss } \mathcal{L}_{\text{SC}} \text{ with weight } \lambda \geq 0}\Bigg]$$



# Experiment 1: Gaussian Mixture

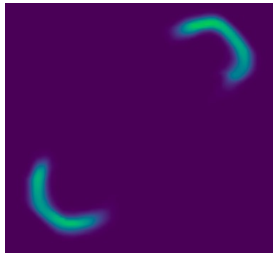
*Posterior estimation, varying training budget N*

- **Model:** $\theta \sim \mathcal{N}(\theta \,|\, \mathbf{0}, \mathbf{I}), \qquad y \sim 0.5\,\mathcal{N}(y \,|\, \theta, \mathbf{I}/2) + 0.5\,\mathcal{N}(y \,|\, -\theta, \mathbf{I}/2)$
- **Results:** Better posterior samples compared to vanilla NPE



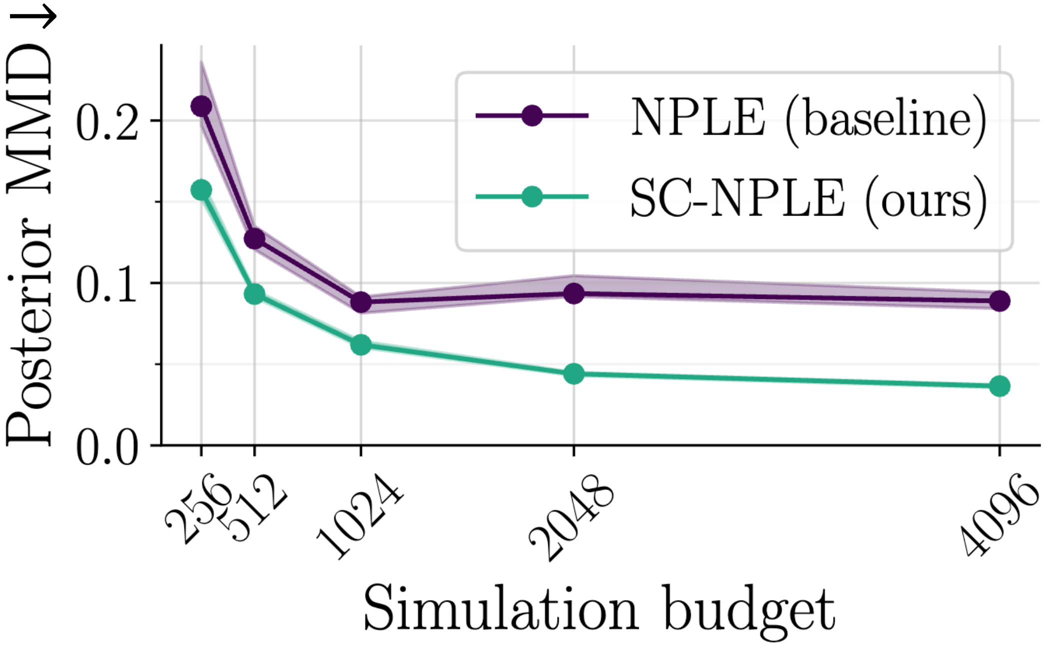
# Experiment 2: Two Moons

*Posterior and likelihood estimation, varying training budget N*



## (1) Better posterior samples

(MMD, lower is better)




## (2) Sharper log marginal likelihood

| **Method** | $N$=512 | $N$=1024 | $N$=2048 | $N$=4096 |
|---|---|---|---|---|
| NPLE | 6.51±0.11 | 7.28±0.10 | 9.07±0.06 | 10.21±0.08 |
| SC-NPLE | **1.70**±0.02 | **1.37**±0.02 | **1.21**±0.01 | **1.14**±0.01 |

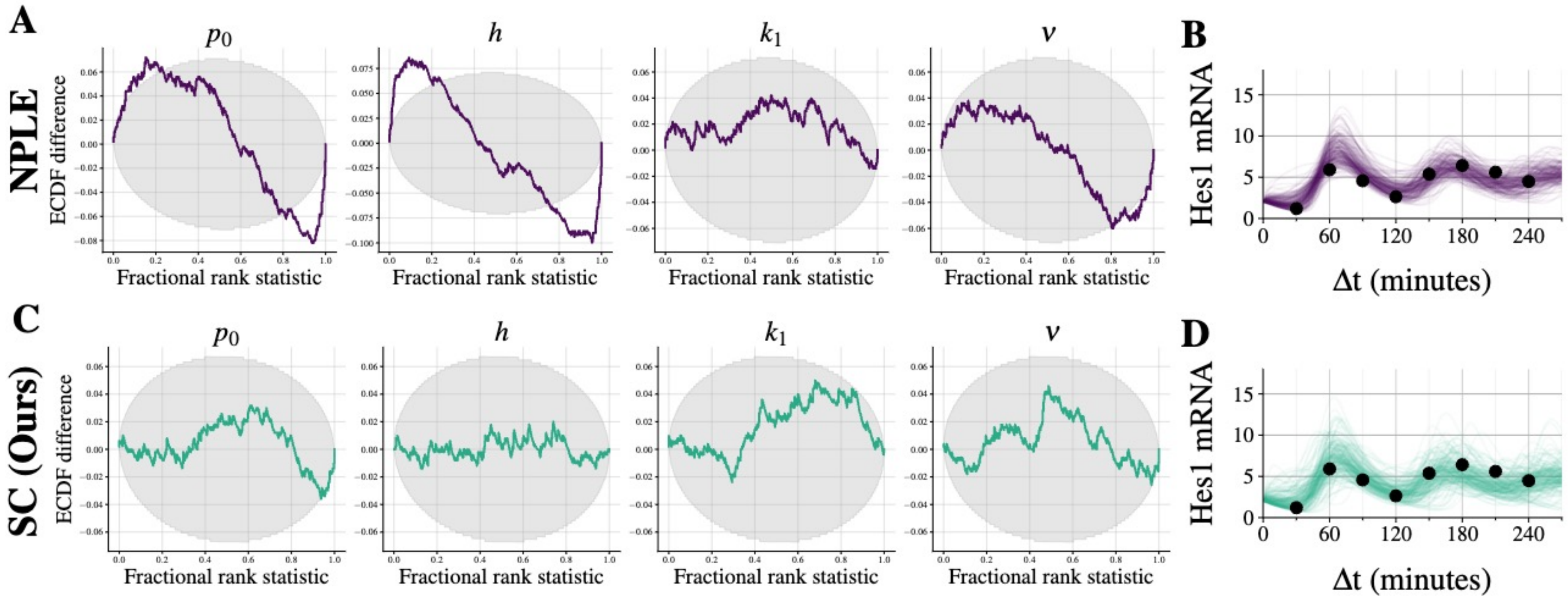
Width of 95% CI of the LML for a data set, mean ± SE

# Experiment 3: Hes1 Expression Model

*Posterior and likelihood estimation, N = 512 training budget*

## Results compared to NPLE baseline:

- Better simulation-based calibration (SBC; Talts et al., 2018)
- Similar posterior predictive results



# Summary and Outlook

*Self-consistency losses reward consistent marginal likelihood estimation*

Gains:

- Improved neural **posterior** estimation (SC-NPE)
- Improved neural **likelihood** estimation (SC-NPLE)
- Improved neural **marginal likelihood** estimation (SC-NPLE)
- Direct extension to popular loss functions in amortized inference

Limitations:

- More expensive upfront **training** → later break-even with non-amortized
- More **hyperparameters** → develop automated choices

# Acknowledgments and Contact


Thanks to my supportive colleagues and advisors:

- **Paul Bürkner**, TU Dortmund University, Germany, [paul-buerkner.github.io](paul-buerkner.github.io)
- **Stefan Radev**, Rensellaer Polytechnic Institute, US, [faculty.rpi.edu/stefan-radev](faculty.rpi.edu/stefan-radev)
- all BayesFlow contributors: www.**BayesFlow**.org

Partially funded by:

ellis

Cyber Valley

SimTech Cluster of Excellence

DFG Deutsche Forschungsgemeinschaft German Research Foundation

Informatics for Life

STRUCTURES CLUSTER OF EXCELLENCE


## CONTACT


@MarvinSchmittML

www.marvinschmitt.com


**I am on the job market for winter 2024. Let's chat!**