# Deep Learning for Simulation-based Inference

EDT STAT-ACTU PhD day, UNamur

May 30, 2023

Gilles Louppe
g.louppe@uliege.be

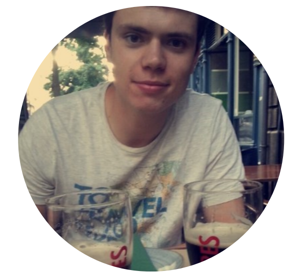
LIÈGE université

Kyle Cranmer

Johann Brehmer

Joeri Hermans

Antoine Wehenkel



Norman Marlier

Siddharth Mishra-Sharma

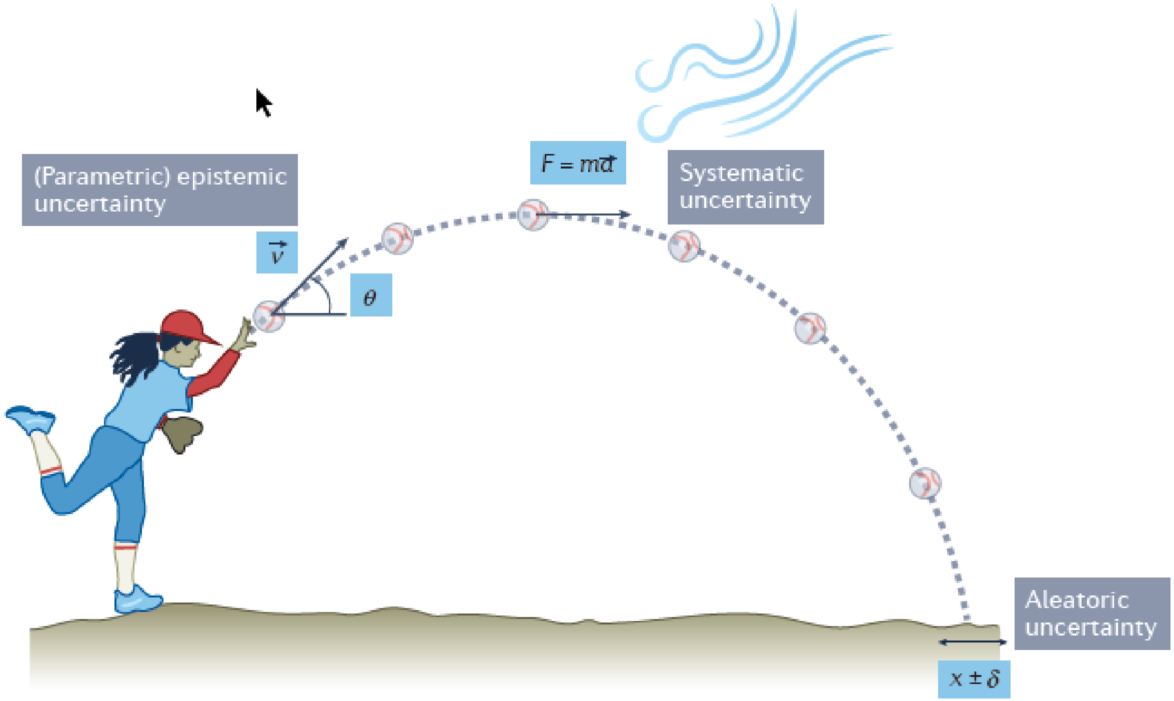
Christoph Weniger

Arnaud Delaunoy

Malavika Vasist

Francois Rozet


CSN
LAD 0
SF 2
3RD
PITCHES 40
0-0
2 OUT
GIANTS

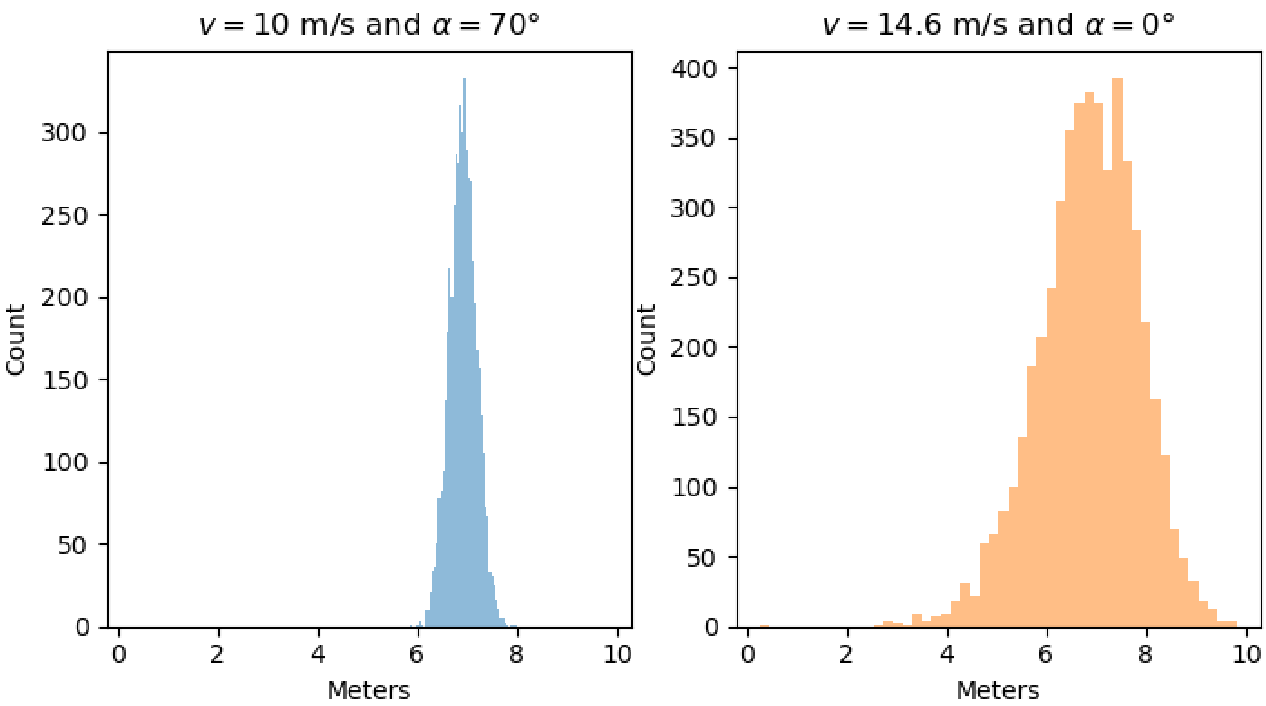

(Parametric) epistemic uncertainty
$\vec{v}$
$\theta$
$F = m\vec{a}$
Systematic uncertainty
Aleatoric uncertainty
$x \pm \delta$

$$v_x = v\cos(\alpha), \quad v_y = v\sin(\alpha),$$

$$\frac{dx}{dt} = v_x, \quad \frac{dy}{dt} = v_y, \frac{dv_y}{dt} = -G.$$

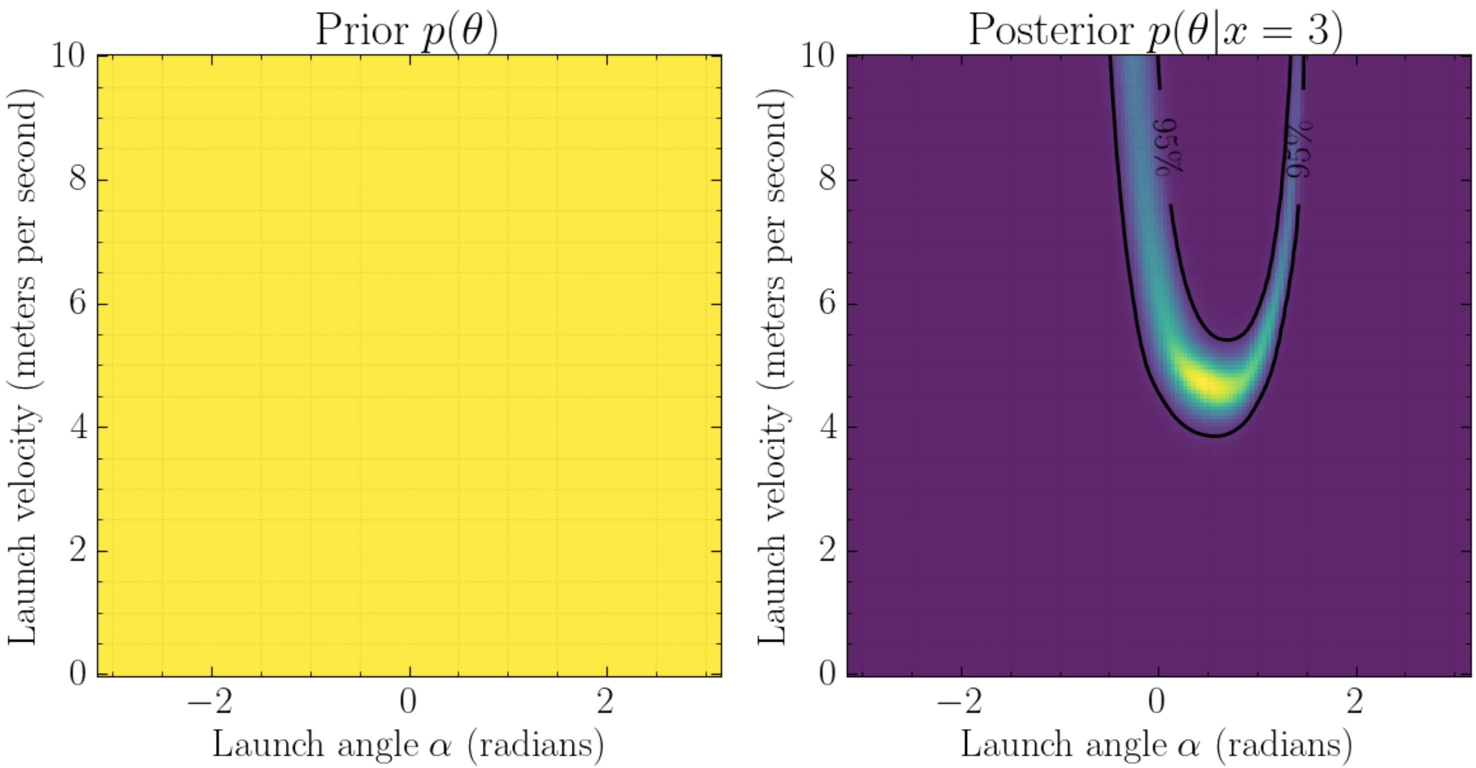
```python
def simulate(v, alpha, dt=0.001):
    v_x = v * np.cos(alpha)  # x velocity m/s
    v_y = v * np.sin(alpha)  # y velocity m/s
    y = 1.1 + 0.3 * random.normal()
    x = 0.0

    while y > 0: # simulate until ball hits floor
        v_y += dt * -G  # acceleration due to gravity
        x += dt * v_x
        y += dt * v_y

    return x + 0.25 * random.normal()
```

The computer simulator defines the likelihood function $p(x|\theta)$ implicitly.



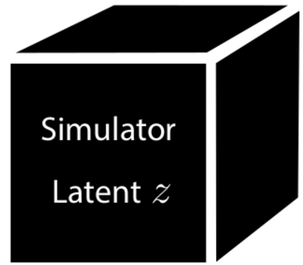
What parameter values $\theta$ are the most plausible?

$$p(\theta|x_{\mathrm{obs}}) = \frac{p(x_{\mathrm{obs}}|\theta)p(\theta)}{p(x_{\mathrm{obs}})}$$



Prior $p(\theta)$
Posterior $p(\theta|x=3)$
Launch velocity (meters per second)
Launch angle $\alpha$ (radians)
95%
95%
0
2
4
6
8
10
−2
0
2

Parameters $\theta$ $\longrightarrow$



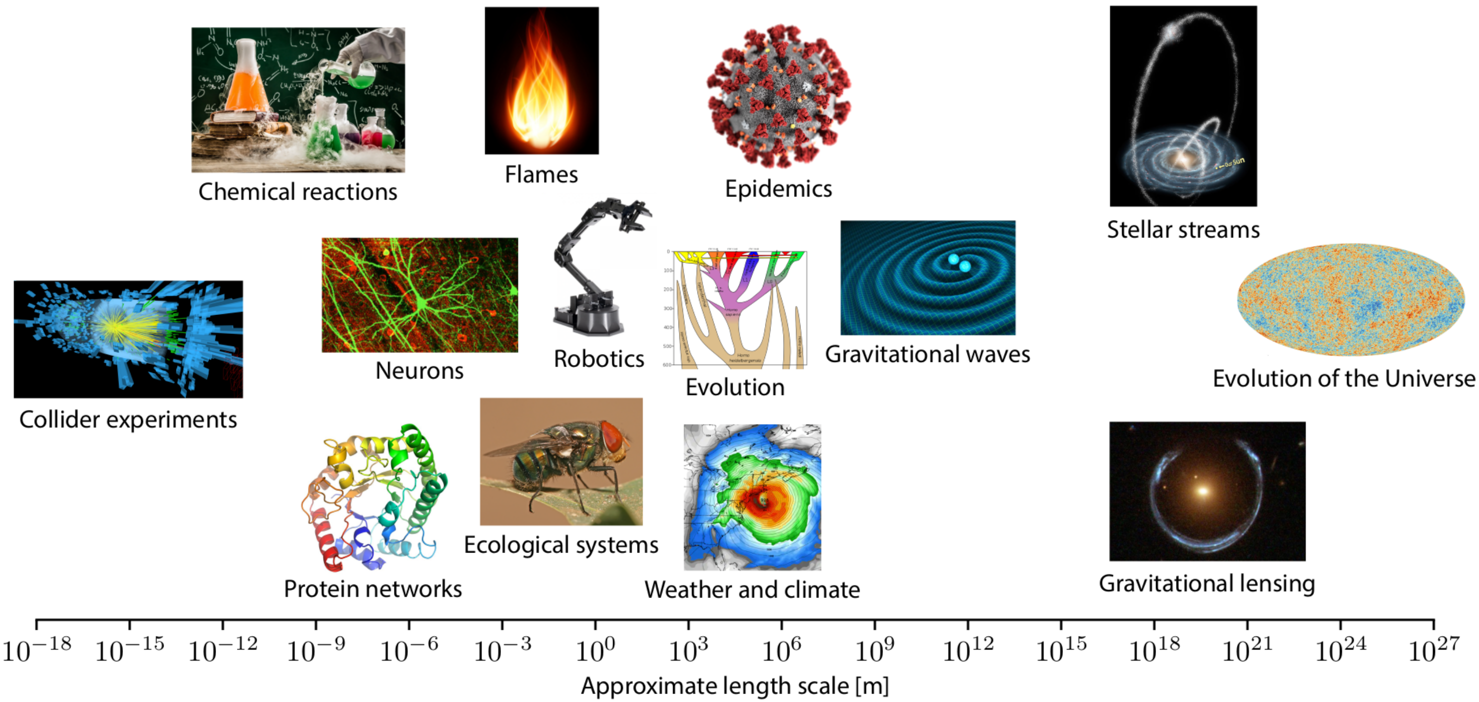

$\longrightarrow$ Observables $x$

Prediction:

- Well-motivated mechanistic, causal model
- Simulator can generate samples $x \sim p(x|\theta)$

Inference:

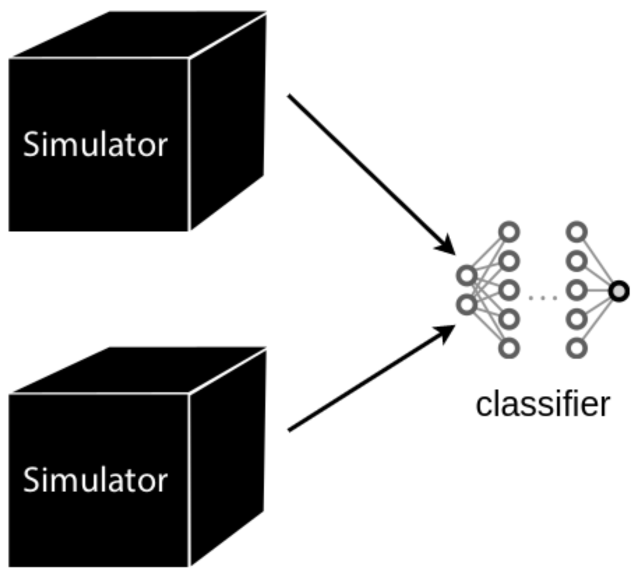
- Interactions between low-level components lead to challenging inverse problems
- Likelihood $p(x|\theta) = \int \mathrm{d}z \; p(x, z|\theta)$ is intractable


Chemical reactions
Flames
Epidemics
Stellar streams
Collider experiments
Neurons
Robotics
Evolution
Gravitational waves
Evolution of the Universe
Protein networks
Ecological systems
Weather and climate
Gravitational lensing
$10^{-18}$ $10^{-15}$ $10^{-12}$ $10^{-9}$ $10^{-6}$ $10^{-3}$ $10^{0}$ $10^{3}$ $10^{6}$ $10^{9}$ $10^{12}$ $10^{15}$ $10^{18}$ $10^{21}$ $10^{24}$ $10^{27}$
Approximate length scale [m]

# Neural ratio estimation

The likelihood-to-evidence $r(x|\theta) = \frac{p(x|\theta)}{p(x)} = \frac{p(x,\theta)}{p(x)p(\theta)}$ ratio can be learned, even if neither the likelihood nor the evidence can be evaluated:

$x, \theta \sim p(x, \theta)$




$x, \theta \sim p(x)p(\theta)$

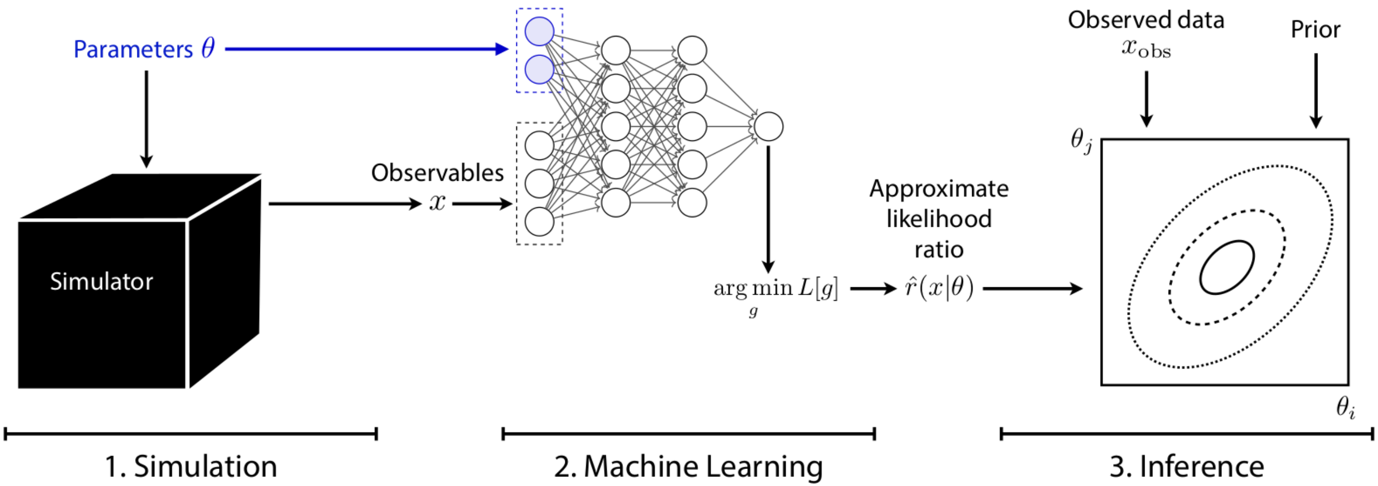
$\hat{r}(x|\theta)$

The solution $d$ found after training approximates the optimal classifier

$$d(x,\theta) \approx d^*(x,\theta) = \frac{p(x,\theta)}{p(x,\theta) + p(x)p(\theta)}.$$

Therefore,

$$r(x|\theta) = \frac{p(x|\theta)}{p(x)} = \frac{p(x,\theta)}{p(x)p(\theta)} \approx \frac{d(x,\theta)}{1 - d(x,\theta)} = \hat{r}(x|\theta).$$




Run simulator and save data

Train NN classifier, interpret as likelihood ratio estimator

Amortized: cheap to repeat for new data

$$p(\theta|x) \approx r(x|\theta)p(\theta)$$

# Constraining dark matter with stellar streams

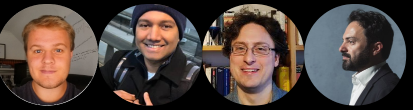
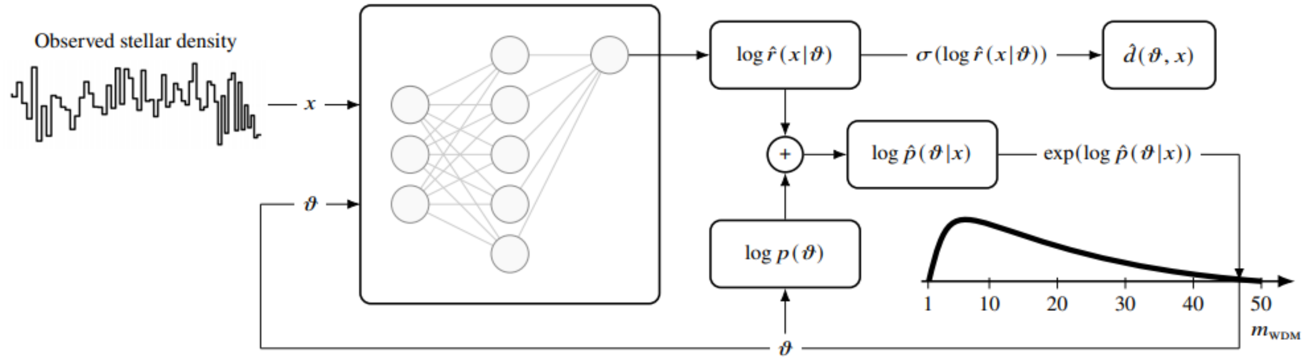
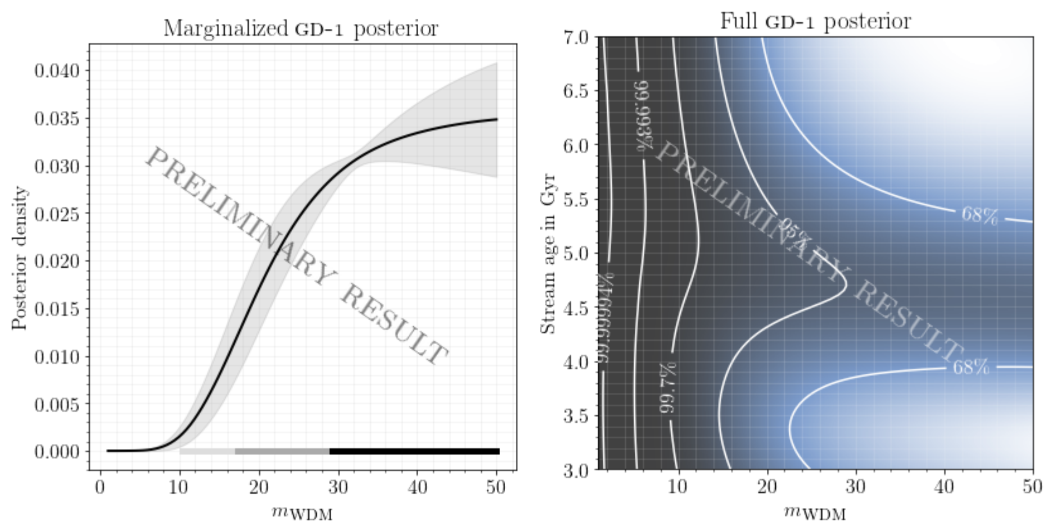



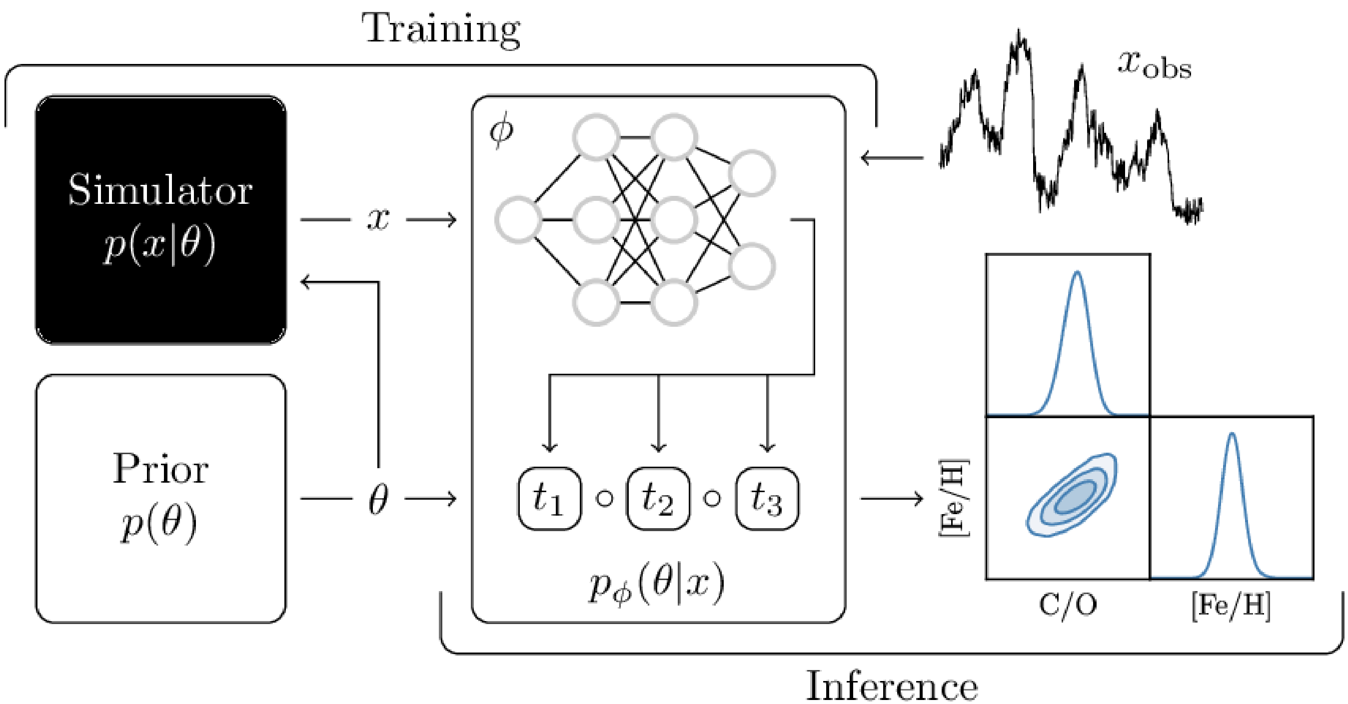
Image credits: C. Bickel/Science, D. Erkal.


Observed stellar density
$x$
$\vartheta$
$\log \hat{r}(x|\vartheta)$
$\sigma(\log \hat{r}(x|\vartheta))$
$\hat{d}(\vartheta, x)$
$\log \hat{p}(\vartheta|x)$
$\exp(\log \hat{p}(\vartheta|x))$
$\log p(\vartheta)$
1 10 20 30 40 50
$m_{\text{WDM}}$


Marginalized GD-1 posterior
Posterior density
$m_{\text{WDM}}$
PRELIMINARY RESULT
Full GD-1 posterior
Stream age in Gyr
68%
95%
99.7%
99.993%
99.9994%


Preliminary results for GD-1 suggest a **preference for CDM over WDM**.

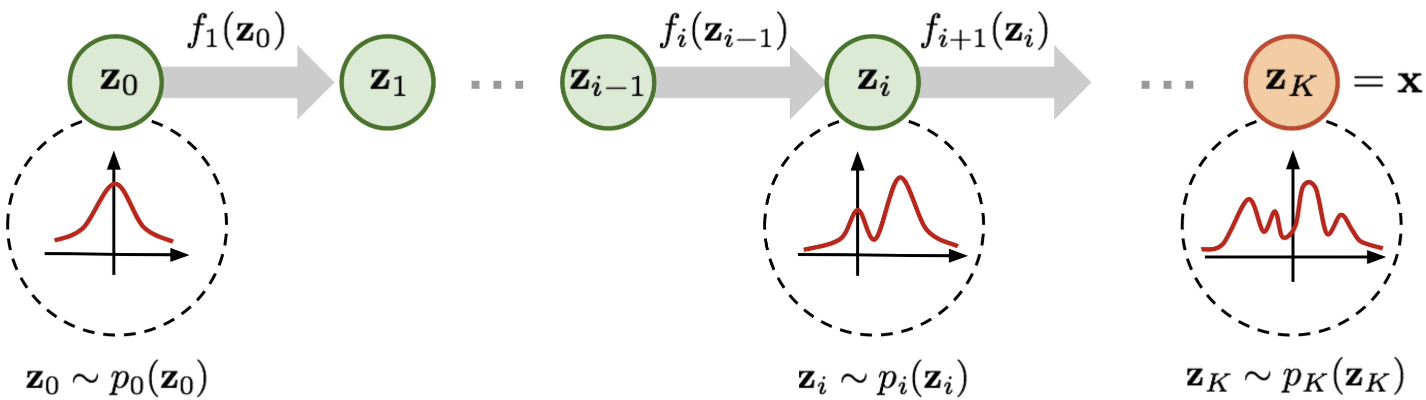
# Neural Posterior Estimation




$$\min_{q_\phi} \mathbb{E}_{p(x)} \left[ \mathrm{KL}(p(\theta|x) || q_\phi(\theta|x)) \right]$$

## Normalizing flows

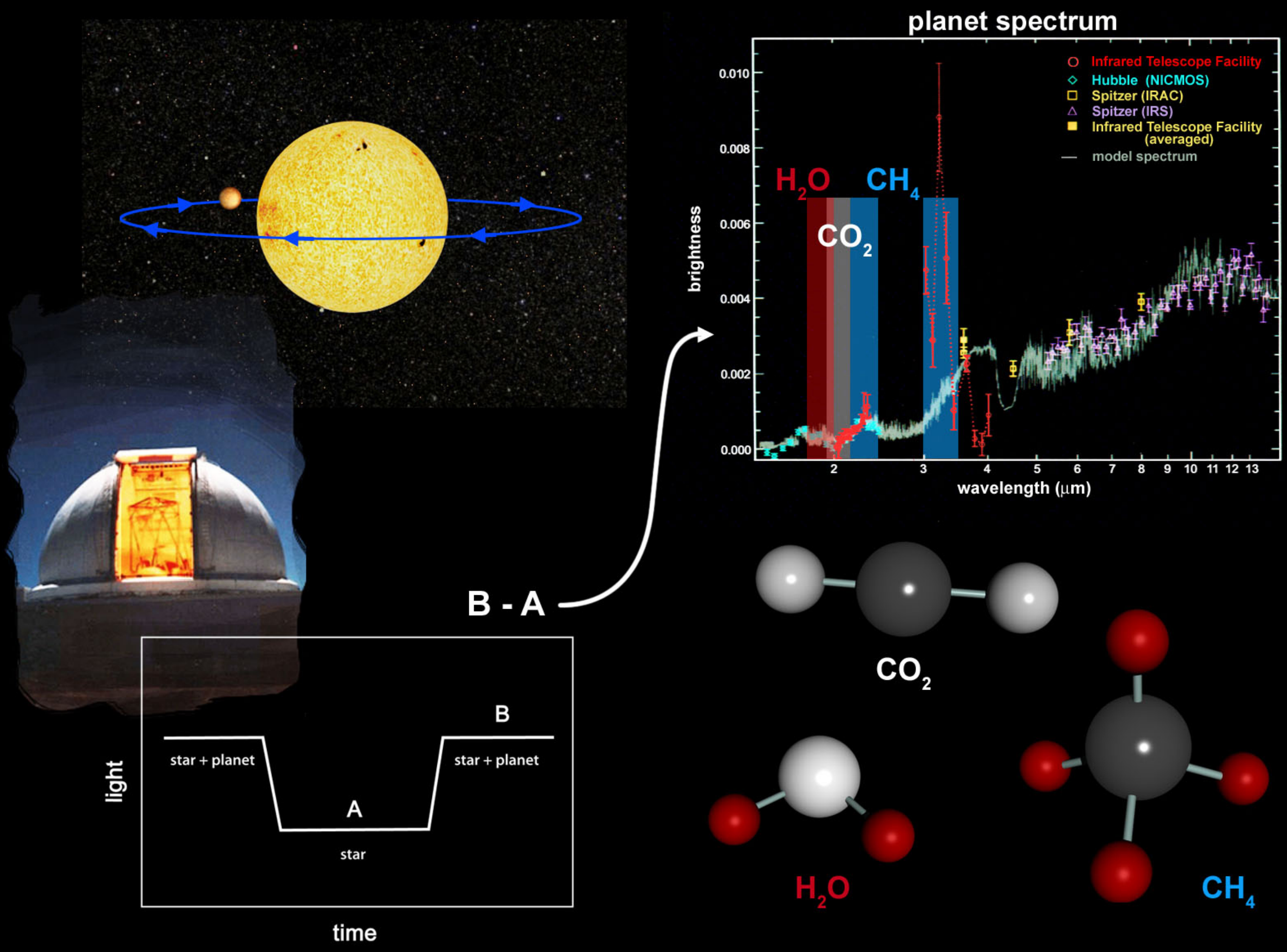
A normalizing flow is a sequence of invertible transformations $f_k$ that map a simple distribution $p_0$ to a more complex distribution $p_K$:



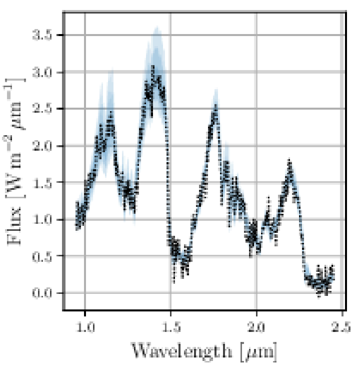

By the change of variables formula, the log-likelihood of a sample $x$ is given by

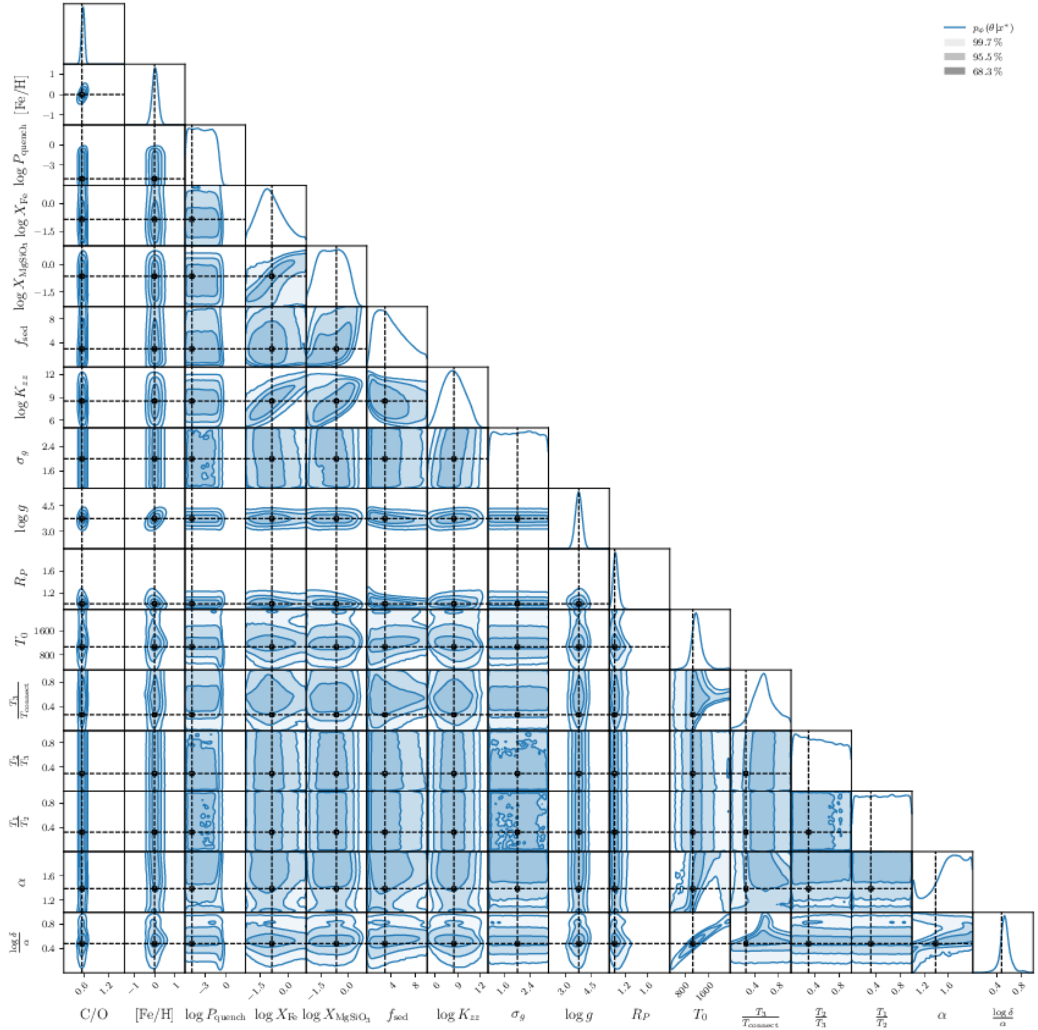
$$\log p_K(x) = \log p_0(z) - \sum_{k=1}^{K} \log \left| \det \frac{\partial f_k}{\partial z_{k-1}} \right| .$$

# Exoplanet atmosphere characterization




—
Credits: NSA/JPL-Caltech, 2010.



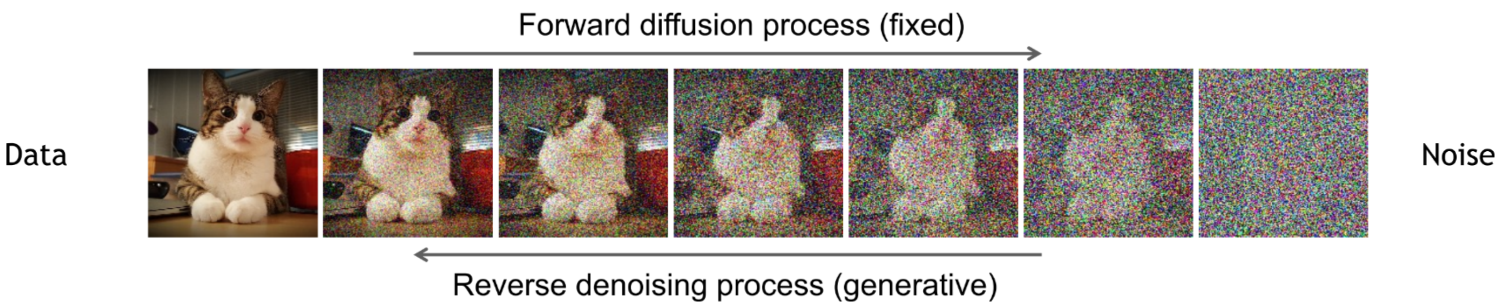



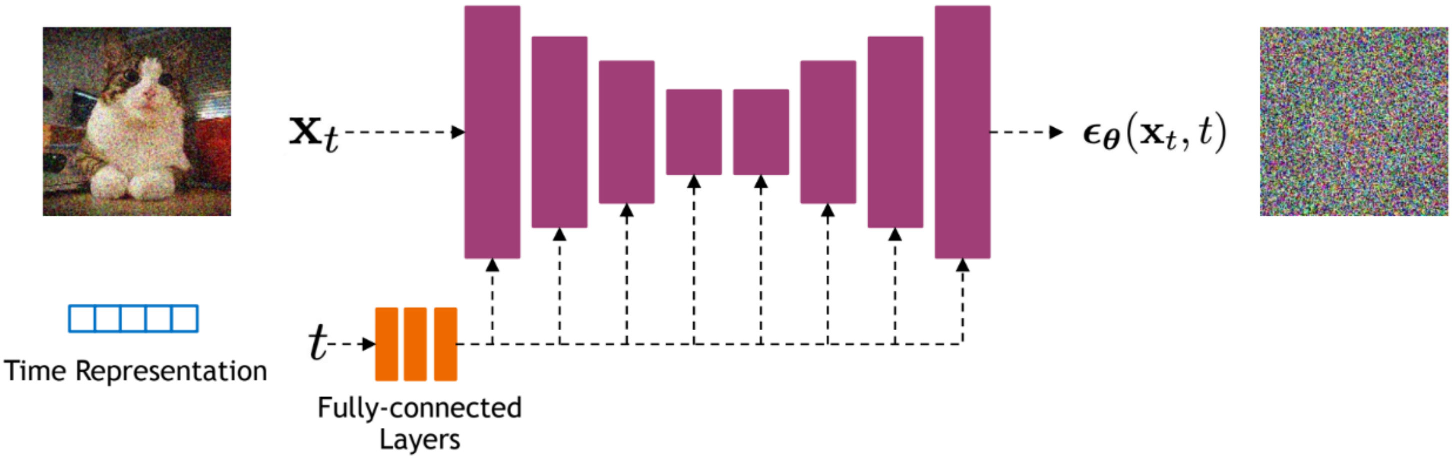

—

Credits: Vasist et al, 2023.

# Score-based data assimilation

# Diffusion models



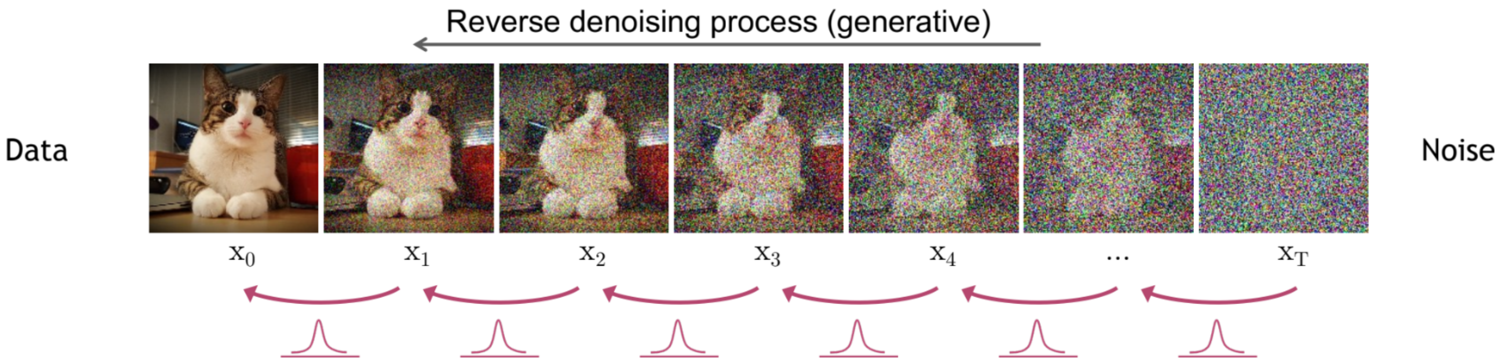



This neural network can be trained by denoising score matching,

$$\arg\min_{\theta} \mathbb{E}_{p(x)p(t)p(\epsilon)} \left[ \left|\epsilon_\theta(\mu(t)x + \sigma(t)\epsilon, t) - \epsilon\right|_2^2 \right],$$

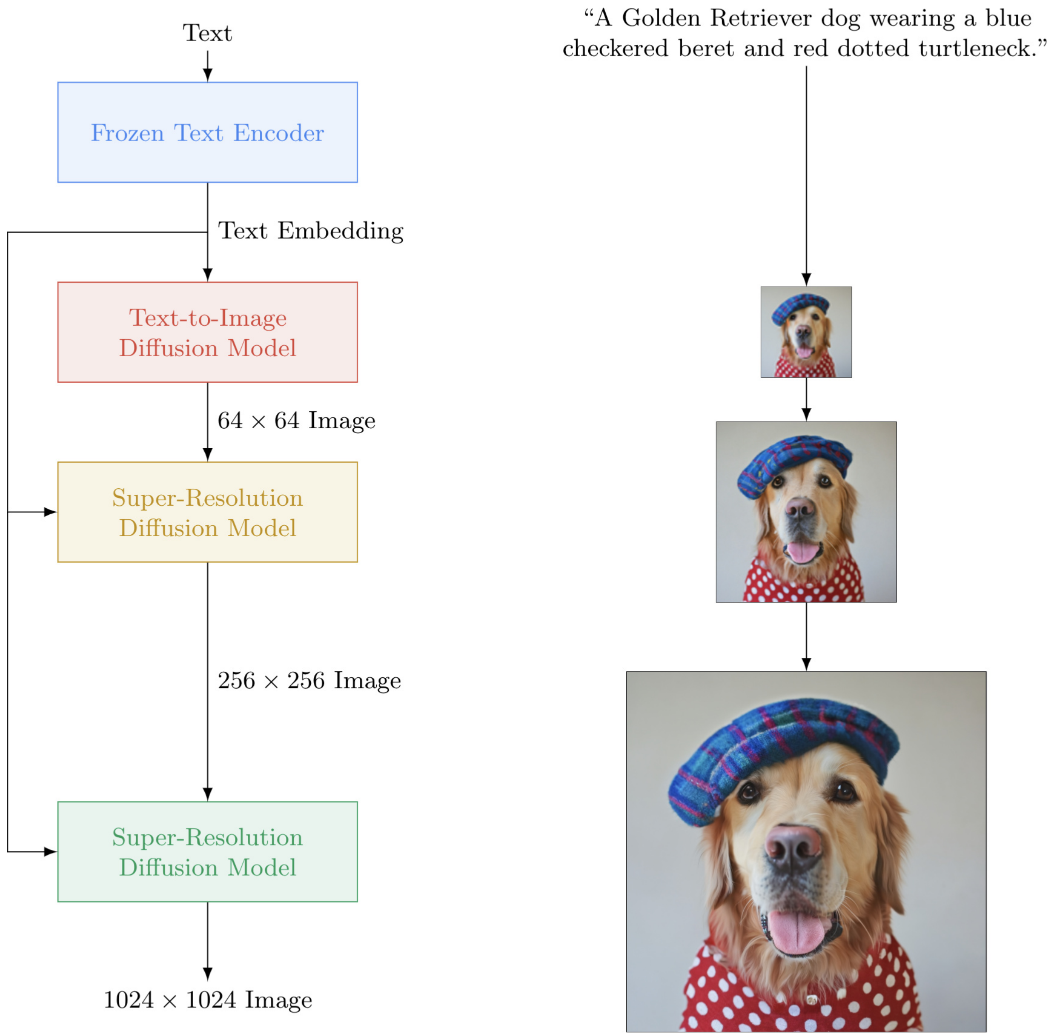
where $\epsilon_\theta(\mathbf{x}_t, t) = -\sigma(t)s_\theta(\mathbf{x}_t, t)$ and $s_\theta(\mathbf{x}_t, t)$ eventually converges to the score $\nabla_{\mathbf{x}_t} \log q(\mathbf{x}_t)$.

New samples can be generated by following the reverse denoising process.




**Algorithm 2** Sampling

$\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
**for** $t = T, \dots, 1$ **do**
  $\mathbf{z} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ if $t > 1$, else $\mathbf{z} = \mathbf{0}$
  $\mathbf{x}_{t-1} = \frac{1}{\sqrt{\alpha_t}}\left(\mathbf{x}_t - \frac{1-\alpha_t}{\sqrt{1-\bar{\alpha}_t}}\boldsymbol{\epsilon}_\theta(\mathbf{x}_t, t)\right) + \sigma_t \mathbf{z}$
**end for**
**return** $\mathbf{x}_0$


Text
Frozen Text Encoder
Text Embedding
Text-to-Image
Diffusion Model
64 × 64 Image
Super-Resolution
Diffusion Model
256 × 256 Image
Super-Resolution
Diffusion Model
1024 × 1024 Image
"A Golden Retriever dog wearing a blue checkered beret and red dotted turtleneck."

For conditional sampling, we can also use the Bayes rule and notice that

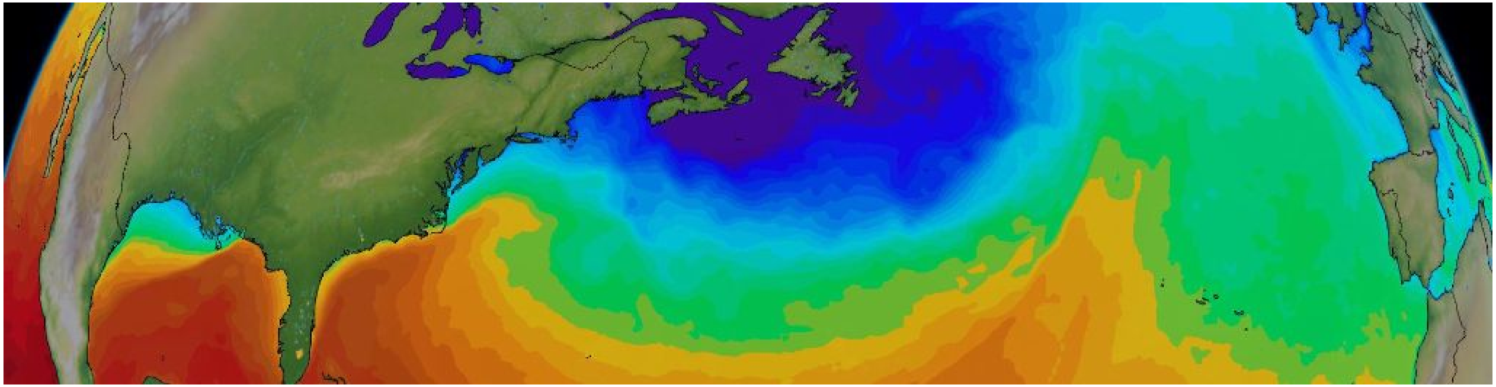
$$\nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t|y) = \nabla_{\mathbf{x}_t} \log p(\mathbf{x}_t) + \nabla_{\mathbf{x}_t} \log p(y|\mathbf{x}_t),$$

where we leverage the fact that the gradient of $\log p(y)$ with respect to $\mathbf{x}_t$ is zero.

## Score-based data assimilation

Diffusion models can be turned into data assimilation models over large-scale dynamical systems:

- Train a diffusion model on a large set of state trajectories $p(x_{1:L})$.
- Assimilate observations $y$ by conditional sampling, resulting in the posterior $p(x_{1:L}|y)$.



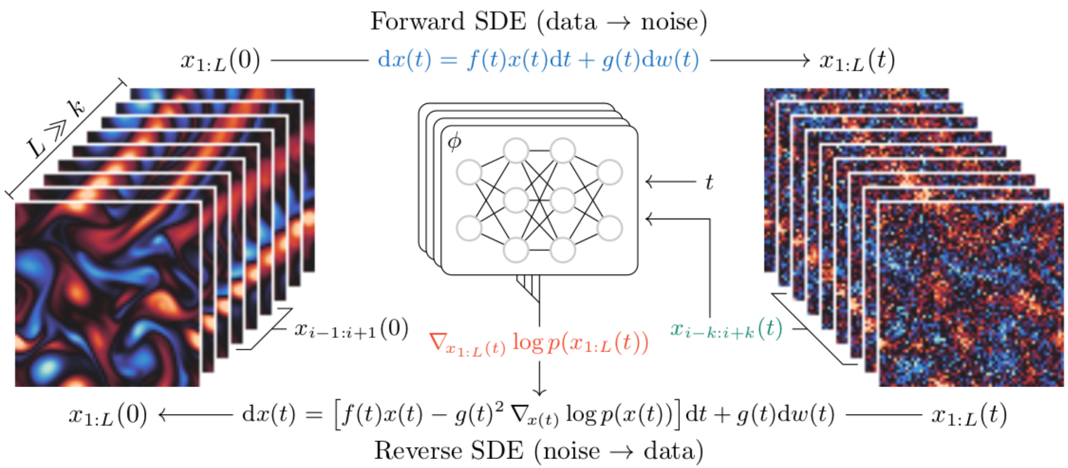



Figure 1. Trajectories $x_{1:L}$ of a dynamical system are transformed to noise via a diffusion process. Reversing this process generates new trajectories, but requires the score of $p(x_{1:L}(t))$. We approximate it by combining the outputs of a score network over subsequences of $x_{1:L}(t)$.

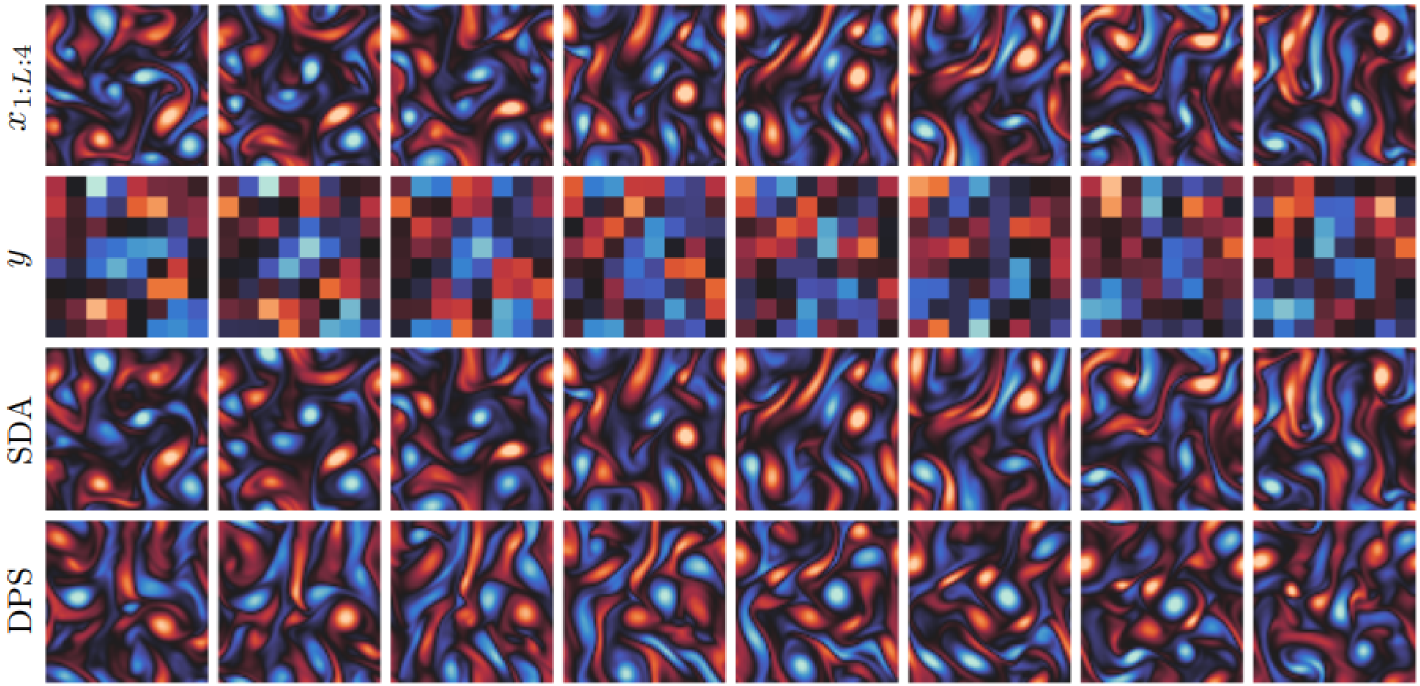



Figure 4: Example of sampled trajectory from coarse, intermittent and noisy observations. States are visualized by their vorticity field $\omega = \nabla \times \boldsymbol{u}$, that is the curl of the velocity field. Positive values (red) indicate clockwise rotation and negative values (blue) indicate counter-clockwise rotation. SDA closely recovers the original trajectory, despite the limited amount of available data. Replacing SDA's likelihood score approximation with the one of DPS [37] yields trajectories inconsistent with the observation.

# Summary

Advances in deep learning have enabled new approaches to statistical inference.

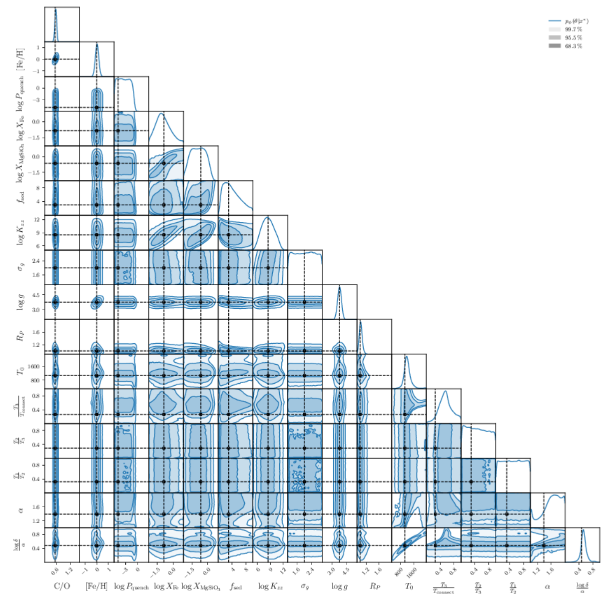
This is major evolution in the statistical capabilities for science, as it enables the analysis of complex models and data without simplifying assumptions.

The end.

# Computational faithfulness

$$\hat{p}(\theta|x) = \mathrm{sbi}(p(x|\theta), p(\theta), x)$$

We must make sure our approximate simulation-based inference algorithms can (at least) actually realize faithful inferences on the (expected) observations.

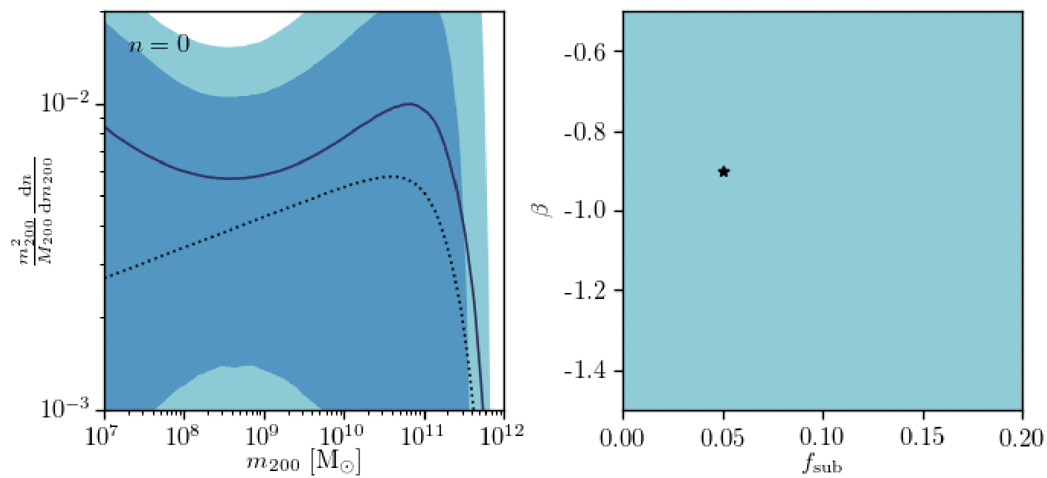


*How do we know this is good enough?*

*Mode convergence:*

The maximum a posteriori estimate converges towards the nominal value $\theta^*$ for an increasing number of independent and identically distributed observables $x_i \sim p(x|\theta^*)$:

$$\lim_{N\to\infty} \arg\max_\theta p(\theta|\{x_i\}_{i=1}^N)$$
$$= \lim_{N\to\infty} \arg\max_\theta p(\theta) \prod_{x_i} r(x_i|\theta) = \theta^*$$




—

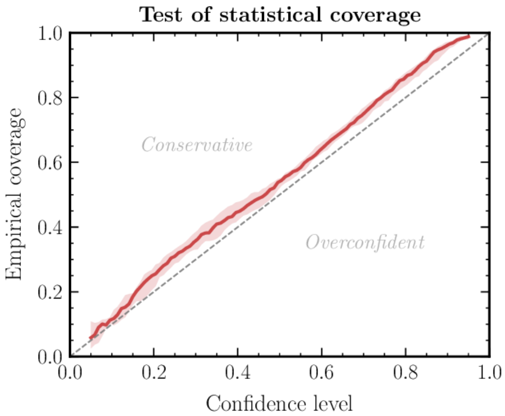
Credits: Brehmer et al, 2019.

A common observation at the root of several other diagnostics is to check for the **self-consistency** of the Bayesian joint distribution,

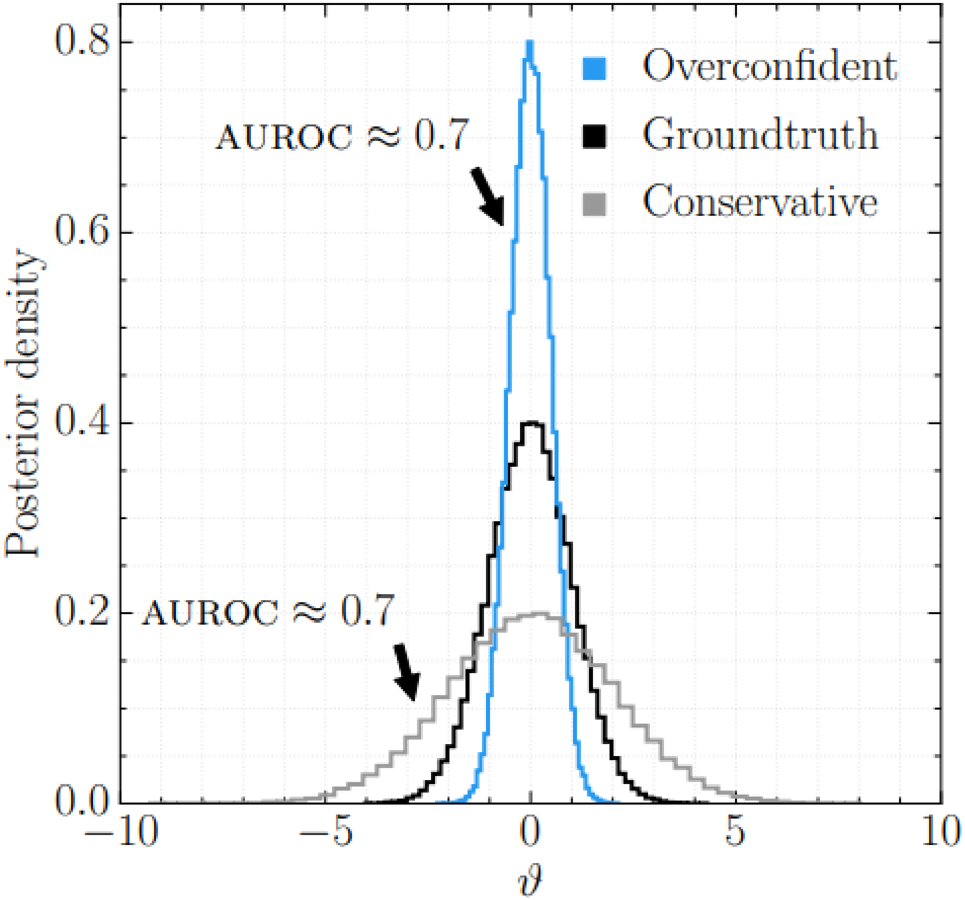
$$p(\theta) = \int p(\theta')p(x|\theta')p(\theta|x)d\theta'\,dx.$$

*Coverage diagnostic:*

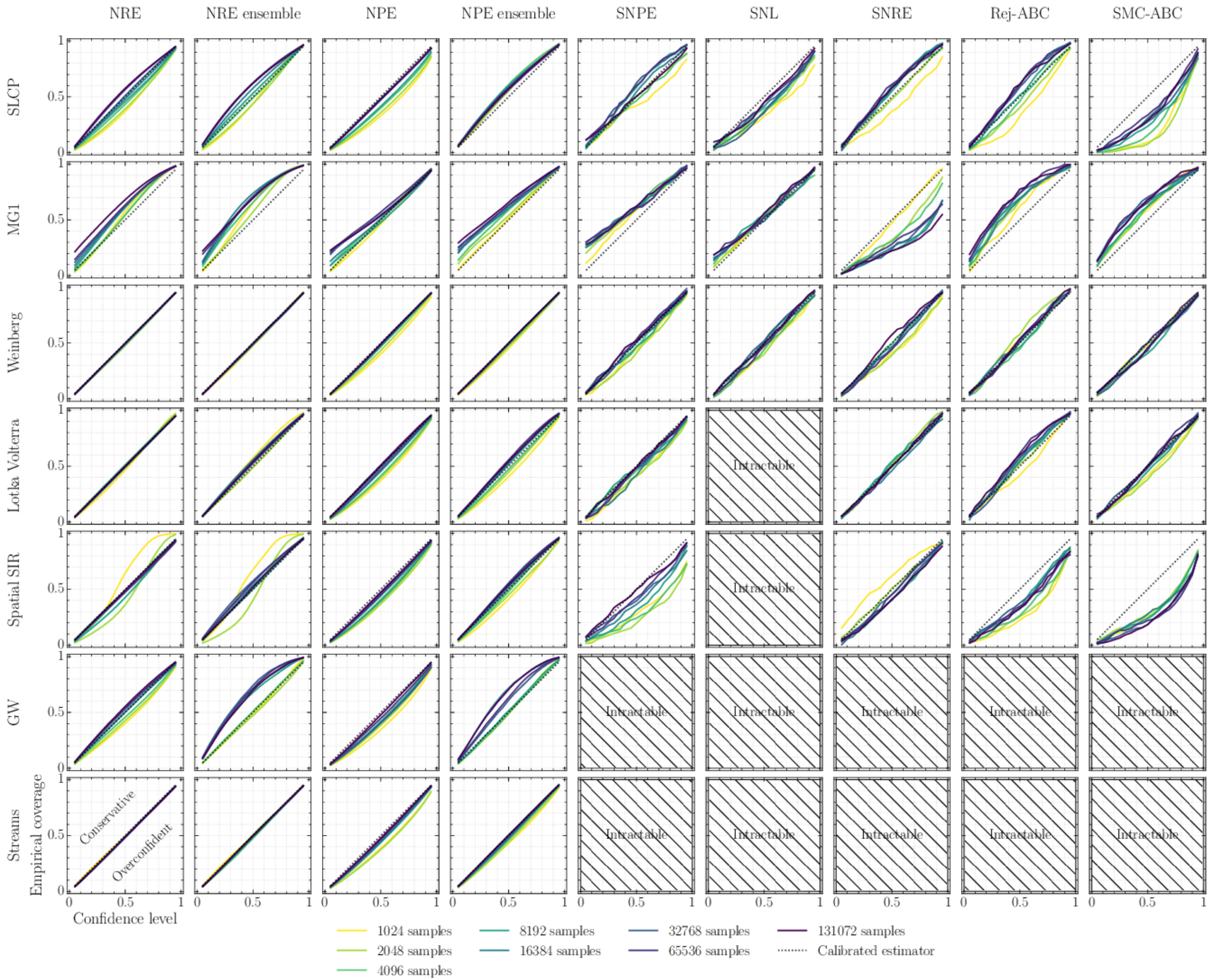
- For $x, \theta \sim p(x, \theta)$, compute the $1-\alpha$ credible interval based on $\hat{p}\left(\theta|x\right)$.
- If the fraction of samples for which $\theta$ is contained within the interval is larger than the nominal coverage probability $1-\alpha$, then the approximate posterior $\hat{p}\left(\theta|x\right)$ has coverage.




—
Credits: Hermans et al, 2021; Siddharth Mishra-Sharma, 2021.


Overconfident
Groundtruth
Conservative
AUROC ≈ 0.7
AUROC ≈ 0.7
Posterior density
0.0
0.2
0.4
0.6
0.8
−10
−5
0
5
10
ϑ



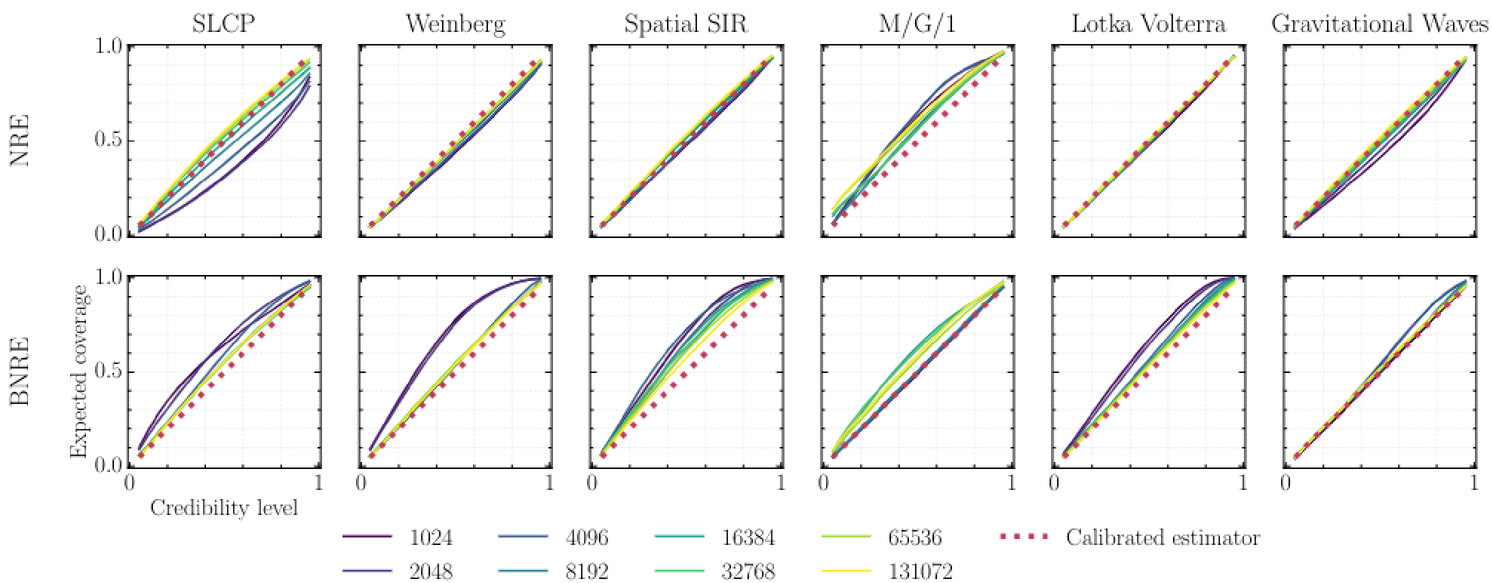

—
Credits: Hermans et al, 2021.

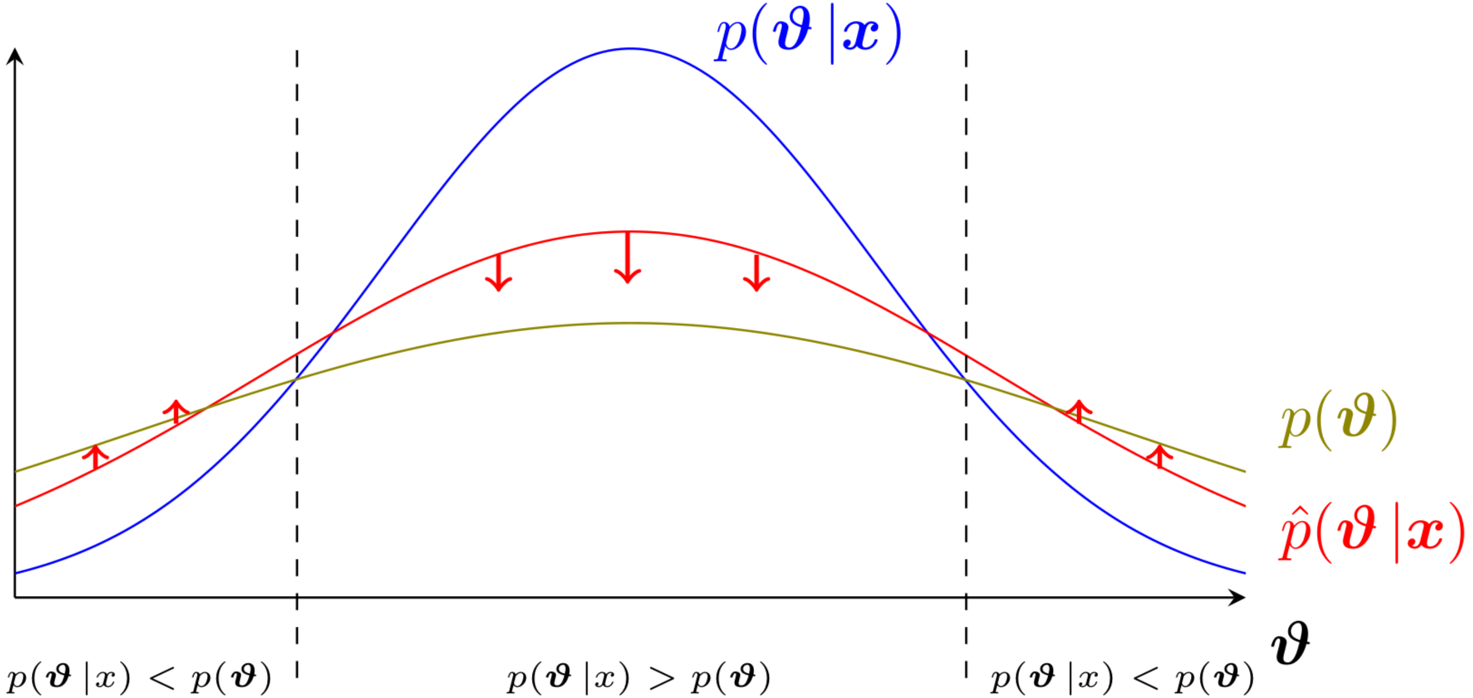
What if diagnostics fail?

# Balanced NRE

Enforce neural ratio estimation to be **conservative** by using binary classifiers $\hat{d}$ that are balanced, i.e. such that

$$\mathbb{E}_{p(\theta,x)}\left[\hat{d}\,(\theta,x)\right] = \mathbb{E}_{p(\theta)p(x)}\left[1 - \hat{d}\,(\theta,x)\right].$$






—
Credits: Delaunoy et al, 2022.