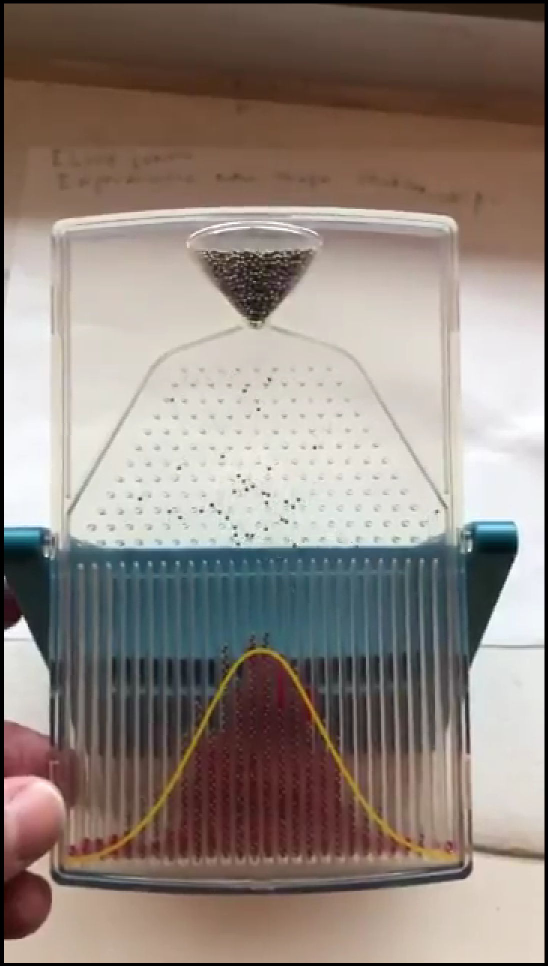
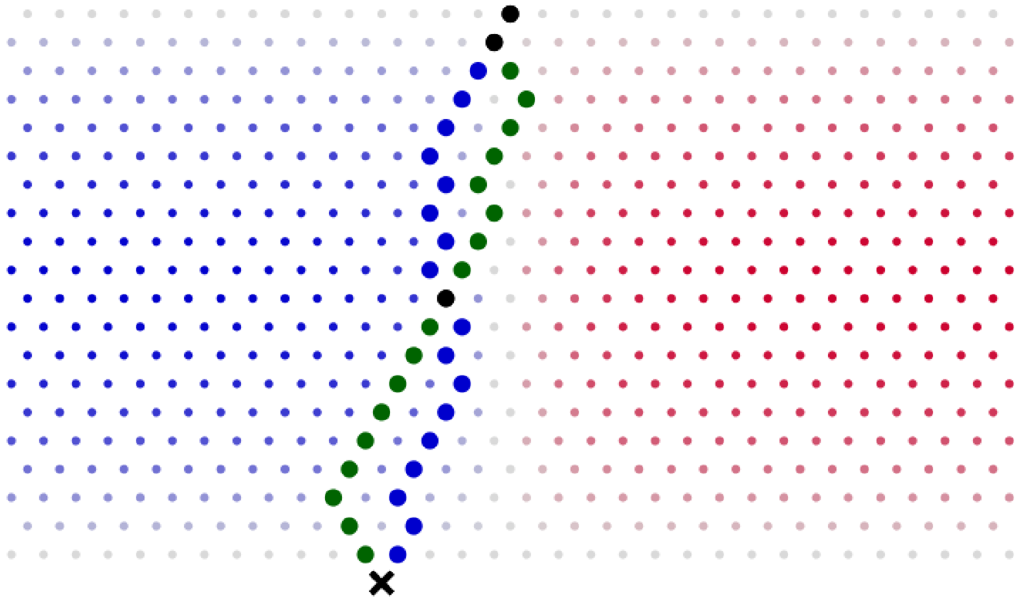
Neural Likelihood-free Inference

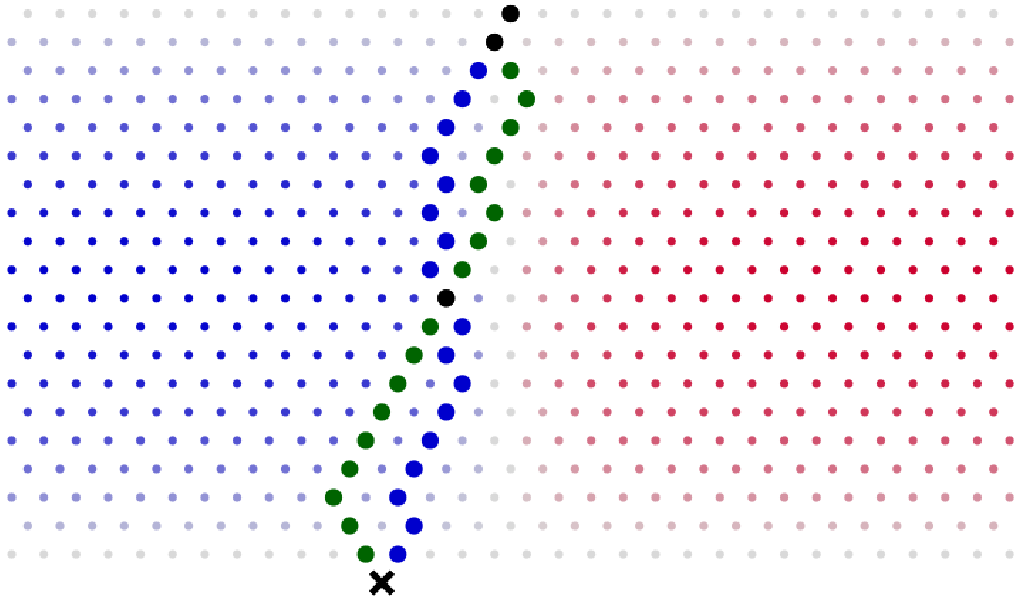
1st Pan-European Advanced School of Statistics in HEP
October 29, DESY, Hamburg

Gilles Louppe
g.louppe@uliege.be





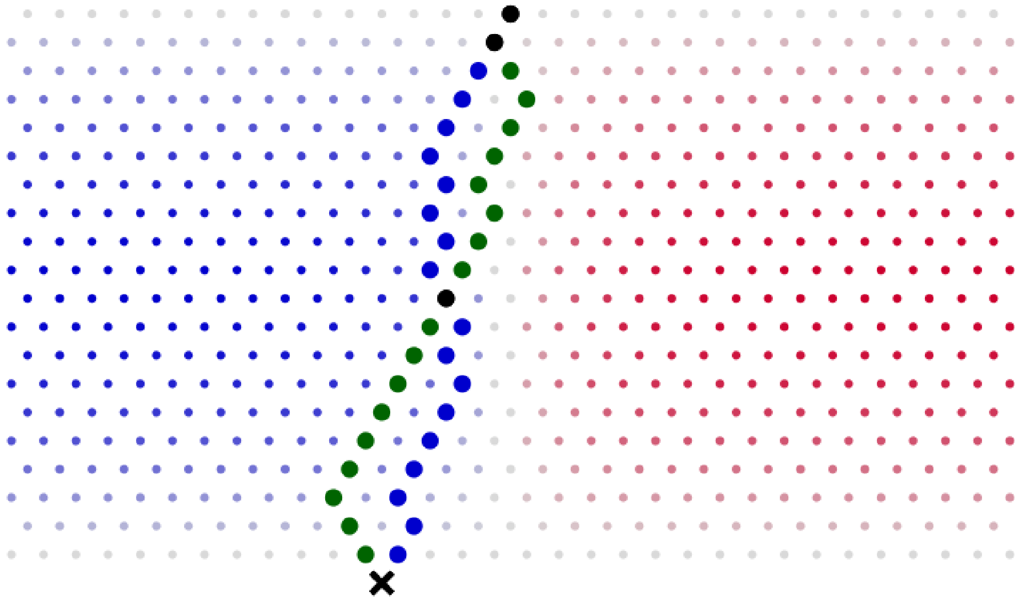
How can we estimate the probability θ of going left when hitting a pin?



The probability of ending in bin x corresponds to the total probability of all the paths z from start to x,

$$p(x|\theta) = \int p(x, z|\theta) dz = \binom{n}{x} \theta^x (1-\theta)^{n-x}.$$

Therefore $\hat{\theta} = \arg\max \prod_{x_i} p(x_i|\theta)\pi(\theta)$.

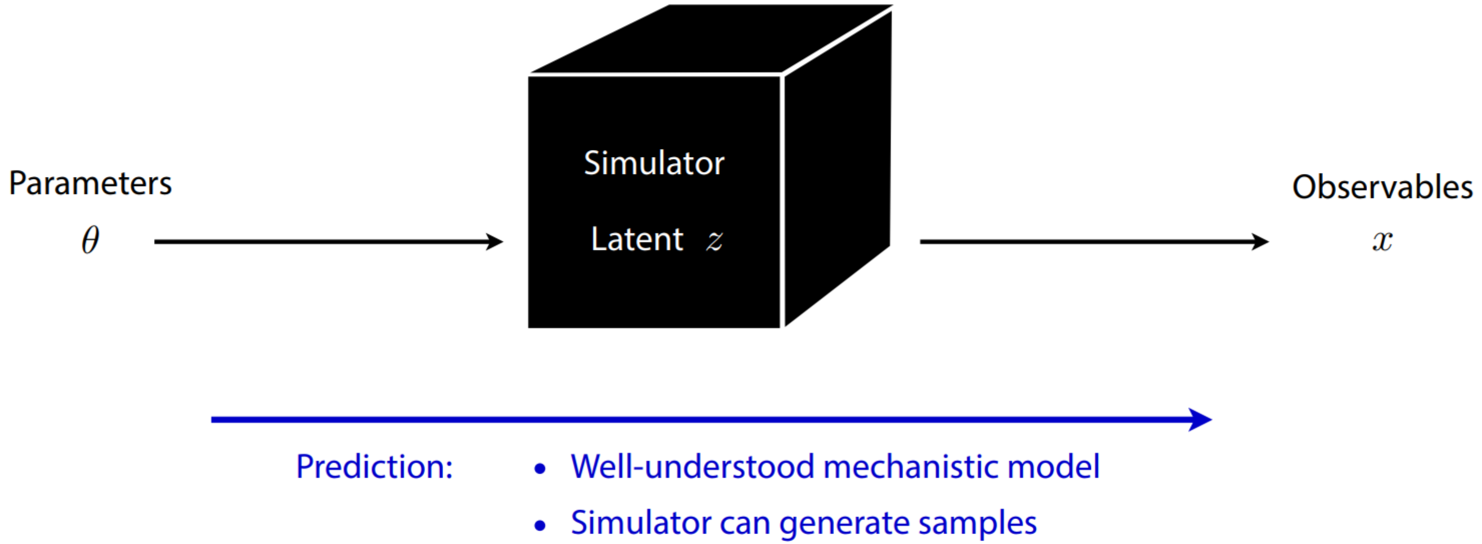


But what if we shift or remove some of the pins?

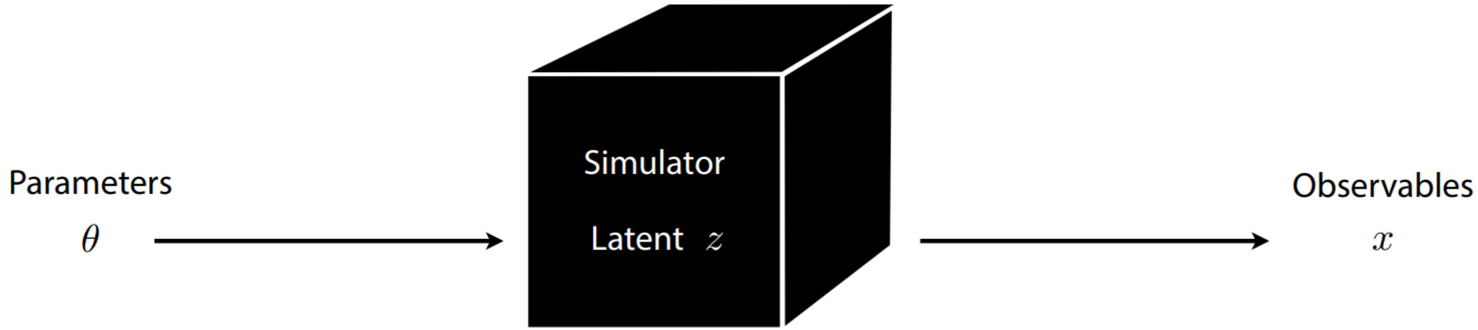
The Galton board is a metaphore of simulation-based science:

Galton board device	$\rightarrow$	Computer simulation
Parameters θ	$\rightarrow$	Model parameters θ
Buckets x	$\rightarrow$	Observables x
Random paths z	$\rightarrow$	Latent variables z (stochastic execution traces through simulator)

Inference in this context requires likelihood-free algorithms.



Credits: Johann Brehmer.



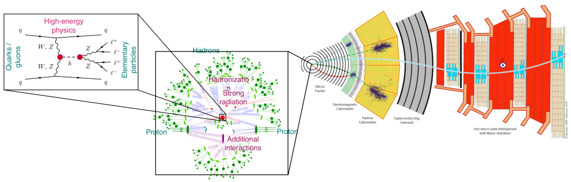
Prediction:

- Well-understood mechanistic model
- Simulator can generate samples

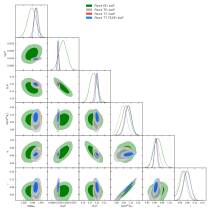
Inference:

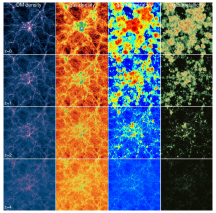
- Likelihood function $p(x|\theta)$ is intractable
- Inference based on estimator $\hat{p}(x|\theta)$

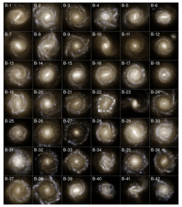
A thriving field of research



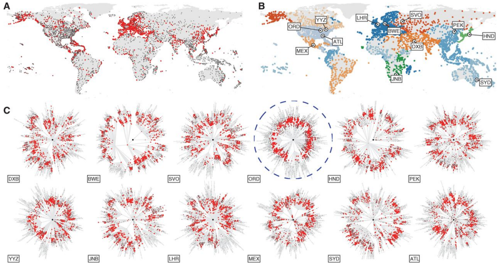
Particle physics



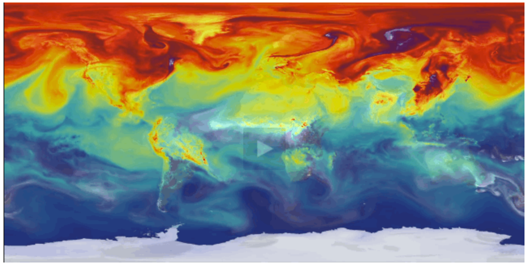




Astrophysics



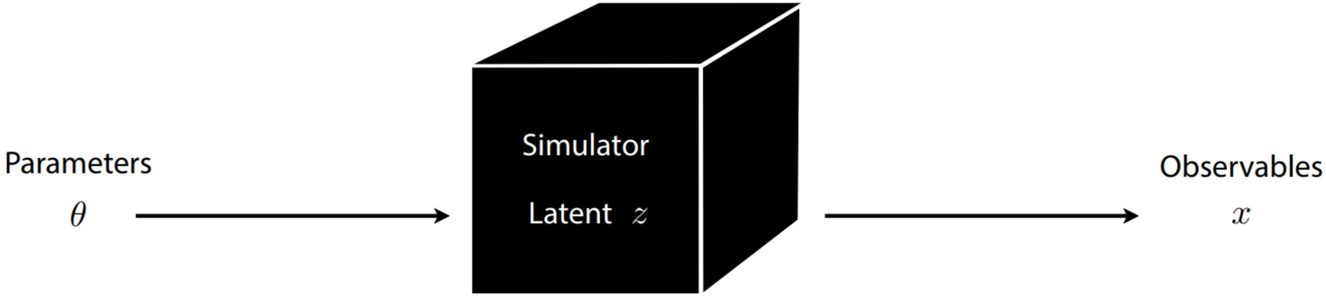
Epidemiology



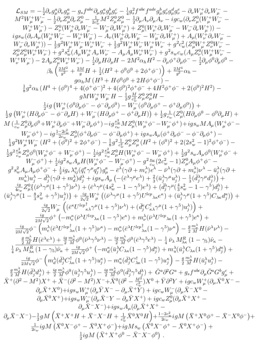
Climatology

(... and many others!)

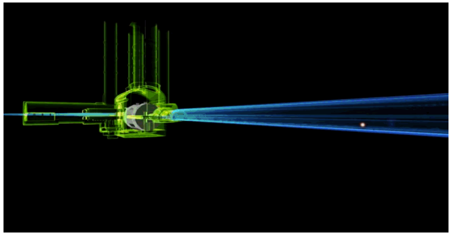
Particle physics



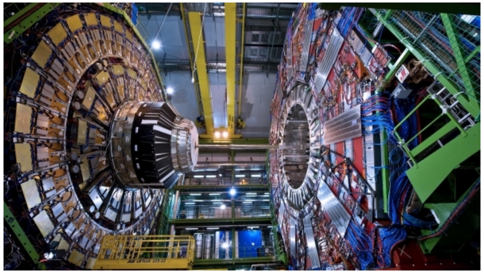
SM with parameters θ



Simulated observables x



Real observations x_{obs}



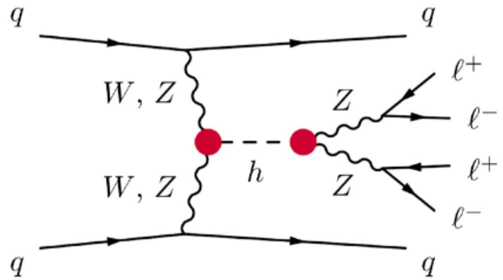
Latent variables

Parameters of interest

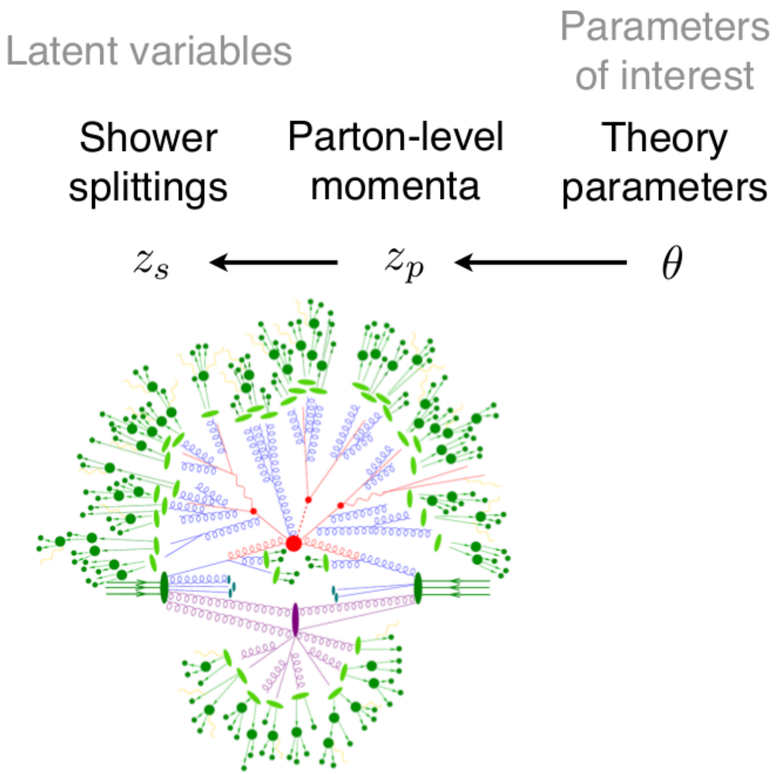
Parton-level momenta

Theory parameters

$z_p \longleftarrow \theta$



Credits: Johann Brehmer.

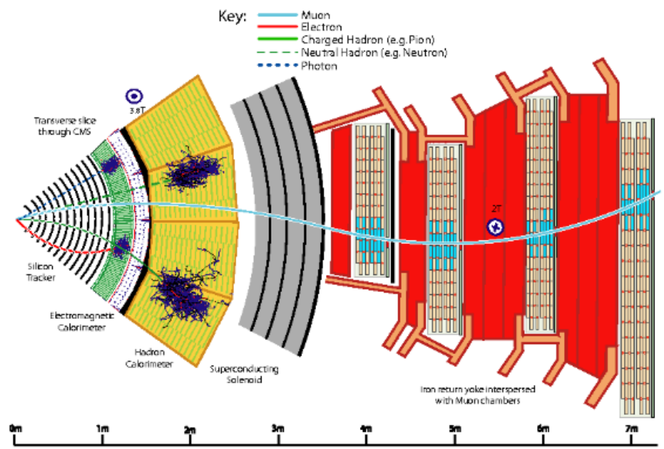


Credits: Johann Brehmer.

Latent variables

Parameters of interest

Detector interactions		Shower splittings		Parton-level momenta		Theory parameters
z_d	$\longleftarrow$	z_s	$\longleftarrow$	z_p	$\longleftarrow$	θ

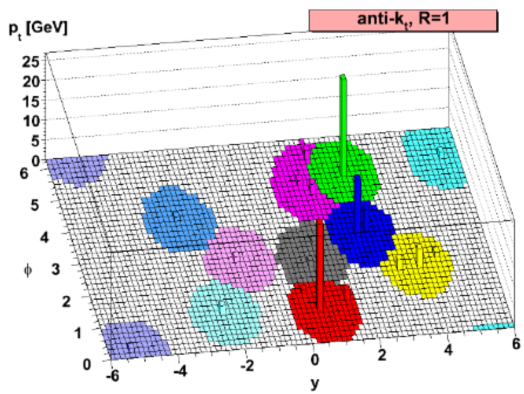


Credits: Johann Brehmer.

Features | Latent variables | Parameters of interest

Observables | Detector interactions | Shower splittings | Parton-level momenta | Theory parameters

$$x \longleftarrow z_d \longleftarrow z_s \longleftarrow z_p \longleftarrow \theta$$



[Image source: M. Cacciari, G. Salam, G. Soyez 0802.1189]

$$p(x|\theta) = \underbrace{\iiint}_{\text{intractable}} p(z_p|\theta)p(z_s|z_p)p(z_d|z_s)p(x|z_d)dz_p dz_s dz_d$$

Likelihood ratio

The likelihood ratio

$$r(x|\theta_0, \theta_1) = \frac{p(x|\theta_0)}{p(x|\theta_1)}$$

is the quantity that is central to many statistical inference procedures.

Examples

- Frequentist hypothesis testing
- Bayesian model comparison
- Bayesian posterior sampling with MCMC
- Bayesian posterior inference through Variational Inference
- Supervised learning
- Generative adversarial networks
- Empirical Bayes with Adversarial Variational Optimization
- Optimal compression

When solving a problem of interest, do not solve a more general problem as an intermediate step. – Vladimir Vapnik

$$\frac{p_{\mathbf{X}}(\mathbf{x}|\theta_0)}{p_{\mathbf{X}}(\mathbf{x}|\theta_1)} = r(\mathbf{x}; \theta_0, \theta_1)$$

Direct likelihood ratio estimation is simpler than density estimation.

(This is fortunate, we are in the likelihood-free scenario!)

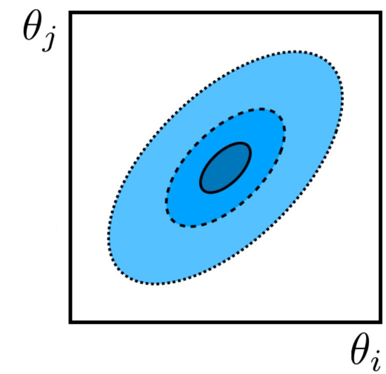
Frequentist inference

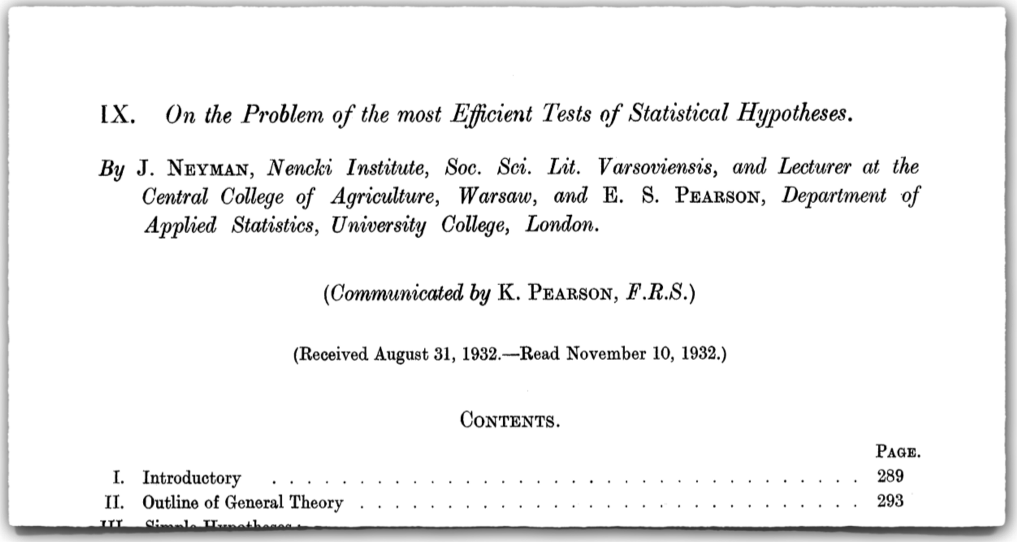
The frequentist (physicist's) way

The Neyman-Pearson lemma states that the likelihood ratio

$$r(x|\theta_0, \theta_1) = \frac{p(x|\theta_0)}{p(x|\theta_1)}$$

is the most powerful test statistic to discriminate between a null hypothesis θ_0 and an alternative θ_1.





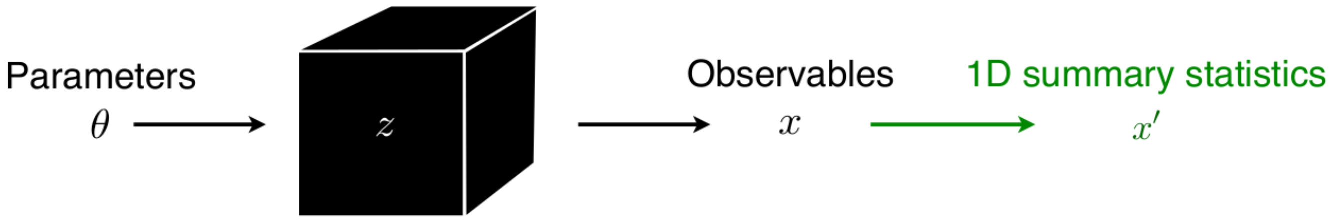
IX. *On the Problem of the most Efficient Tests of Statistical Hypotheses.*

By J. Neyman, *Nencki Institute, Soc. Sci. Lit. Varsoviensis, and Lecturer at the Central College of Agriculture, Warsaw, and* E. S. Pearson, *Department of Applied Statistics, University College, London.*

(*Communicated by* K. Pearson, *F.R.S.*)

(Received August 31, 1932.—Read November 10, 1932.)

Contents.



Define a projection function $s : \mathcal{X} \to \mathbb{R}$ mapping observables x to a summary statistic $x' = s(x)$.

Then, approximate the likelihood $p(x|\theta)$ with the surrogate $\hat{p}(x|\theta) = p(x'|\theta)$.

From this it comes

$$\frac{p(x|\theta_0)}{p(x|\theta_1)} \approx \frac{\hat{p}(x|\theta_0)}{\hat{p}(x|\theta_1)} = \hat{r}(x|\theta_0, \theta_1).$$

Wilks theorem

Consider the test statistic

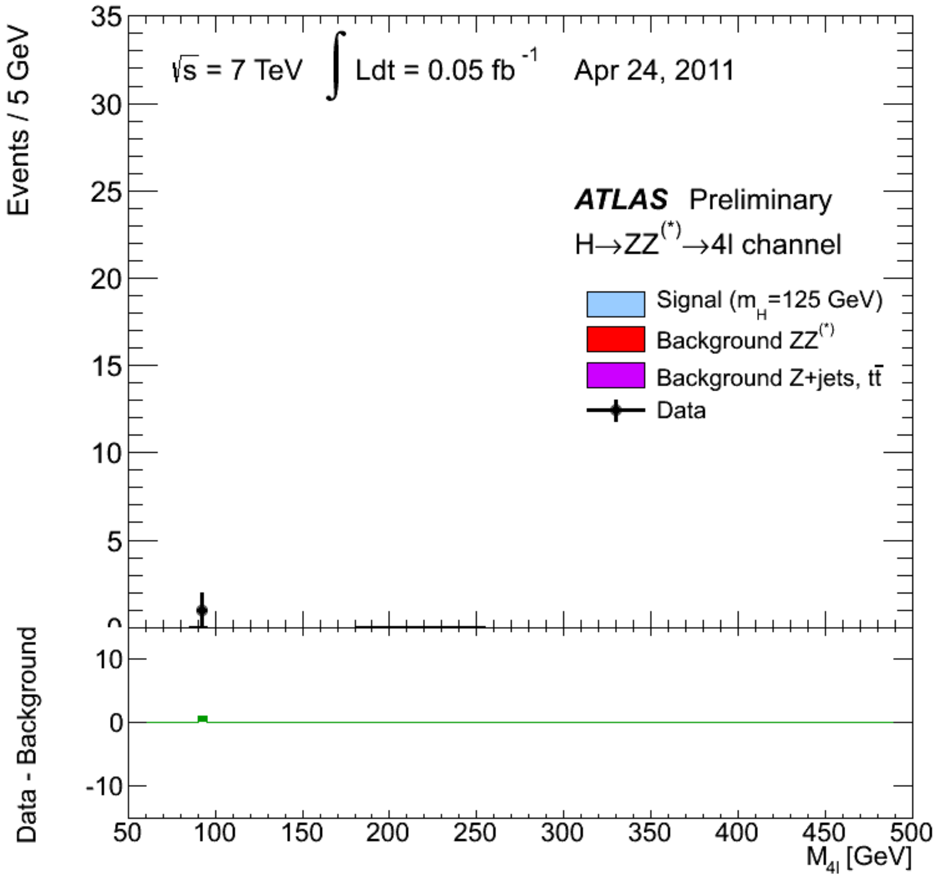
$$q(\theta) = -2 \sum_x \log \frac{p(x|\theta)}{p(x|\hat{\theta})} = -2 \sum_x \log r(x|\theta, \hat{\theta})$$

for a fixed number N of observations $\{x\}$ and where $\hat{\theta}$ is the maximum likelihood estimator.

When $N \to \infty, q(\theta) \sim \chi_2$.

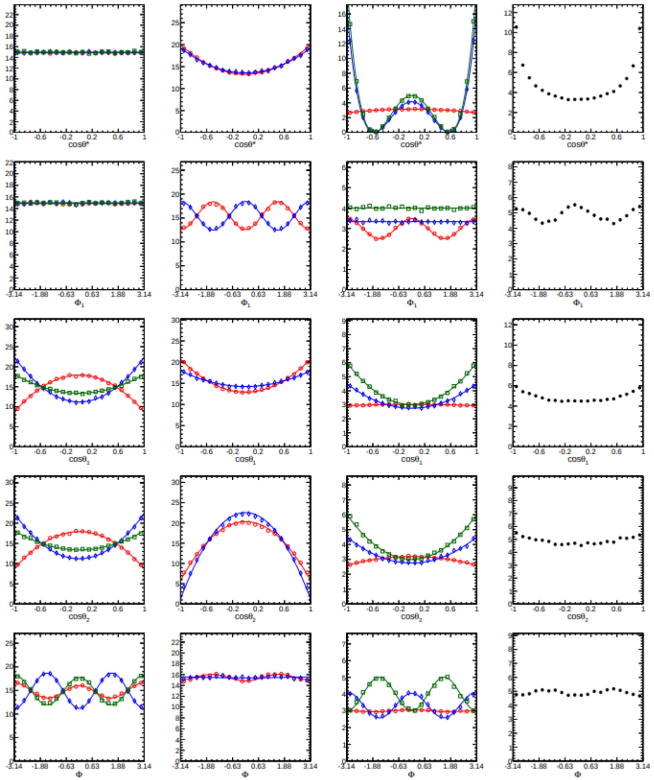
Therefore (and provided the assumptions apply!), an observed value $q_{\text{obs}}(\theta)$ translates directly to a p-value that measures the confidence with which θ can be excluded:

$$p_\theta \equiv \int_{q_{\text{obs}}(\theta)}^{\infty} \mathrm{d}q\, p(q|\theta) = 1 - F_{\chi_2}(q_{\text{obs}}(\theta)).$$



Discovery of the Higgs boson at 5-σ

- Choosing the projection s is difficult and problem-dependent.
- Often there is no single good variable: compressing to any x' loses information.
- Ideally, analyze high-dimensional x', including all correlations.
- Unfortunately, filling high-dimensional histograms is not tractable.



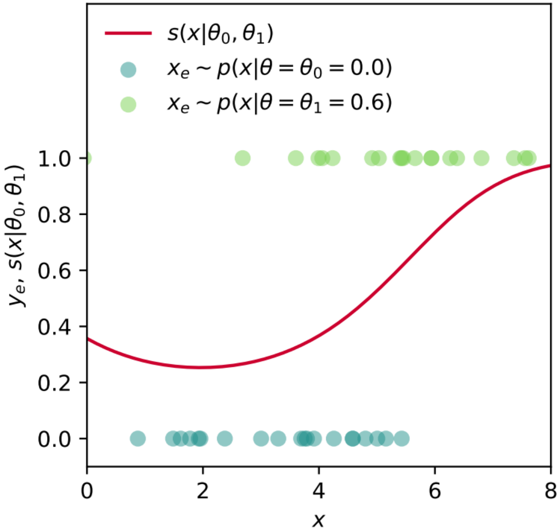
Bolognesi et al, 2012 [arXiv:1208.4018].

CARL

Supervised learning provides a way to automatically construct s:

- Let us consider a neural network classifier $\hat{s}$ tasked to distinguish $x \sim p(x|\theta_0)$ from $x \sim p(x|\theta_1)$.
- Train $\hat{s}$ by minimizing the cross-entropy loss

$$L_{XE}[\hat{s}] = -\mathbb{E}_{p(x|\theta)\pi(\theta)}[1(\theta = \theta_0) \log \hat{s}(x) + 1(\theta = \theta_1) \log(1 - \hat{s}(x))].$$



Cranmer, Pavez and Louppe, 2015 [arXiv:1506.02169].

The solution $\hat{s}$ found after training approximates the optimal classifier

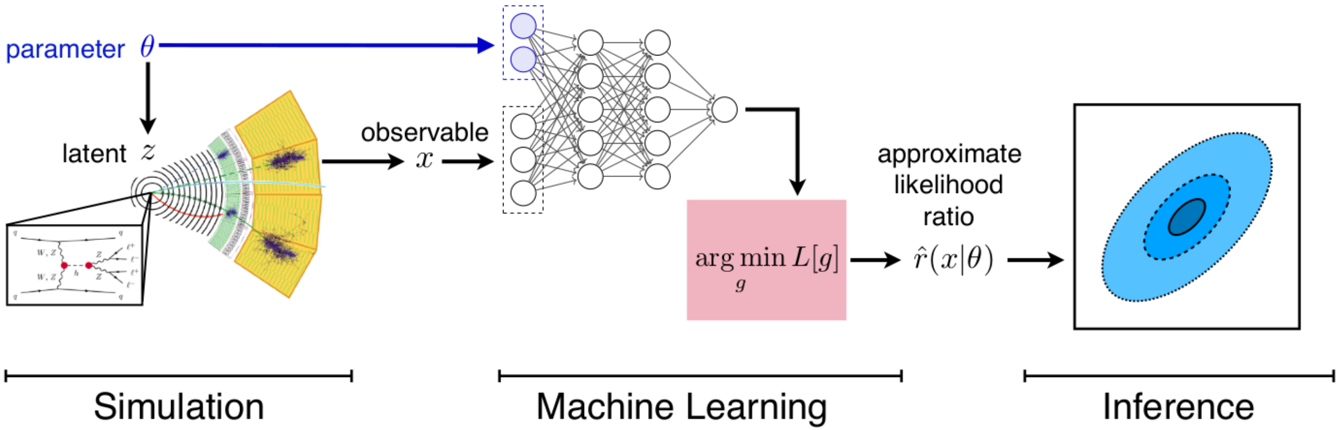
$$\hat{s}(x) \approx s^*(x) = \frac{p(x|\theta_1)}{p(x|\theta_0) + p(x|\theta_1)}.$$

Therefore,

$$r(x|\theta_0, \theta_1) \approx \hat{r}(x|\theta_0, \theta_1) = \frac{1 - \hat{s}(x)}{\hat{s}(x)}$$

That is, supervised classification is equivalent to likelihood ratio estimation.

Cranmer, Pavez and Louppe, 2015 [arXiv:1506.02169].



To avoid retraining a classifier $\hat{s}$ for every (θ_0, θ_1) pair, fix θ_1 to θ_{ref} and train a single parameterized classifier $\hat{s}(x|\theta_0, \theta_{\text{ref}})$ where θ_0 is also given as input.

Therefore, we have

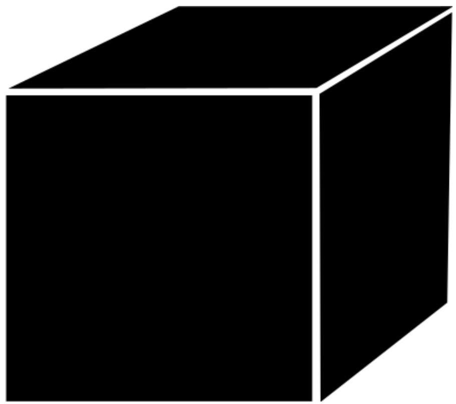
$$\hat{r}(x|\theta_0, \theta_{\text{ref}}) = \frac{1 - \hat{s}(x|\theta_0, \theta_{\text{ref}})}{\hat{s}(x|\theta_0, \theta_{\text{ref}})}$$

such that for any (θ_0, θ_1),

$$r(x|\theta_0, \theta_1) \approx \frac{\hat{r}(x|\theta_0, \theta_{\text{ref}})}{\hat{r}(x|\theta_1, \theta_{\text{ref}})}.$$

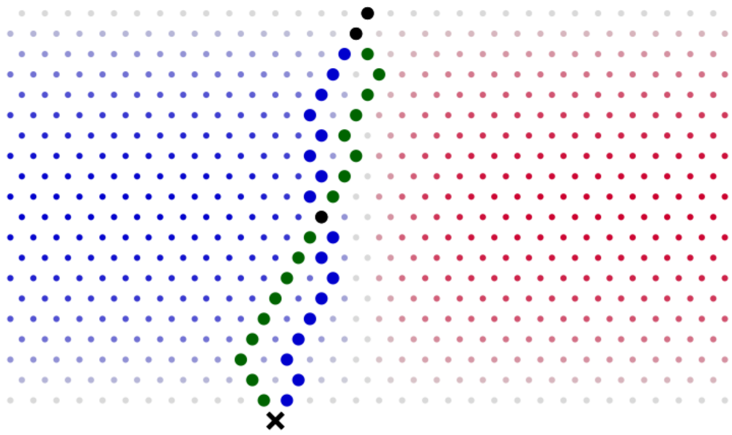
Cranmer, Pavez and Louppe, 2015 [arXiv:1506.02169].

Opening the black box

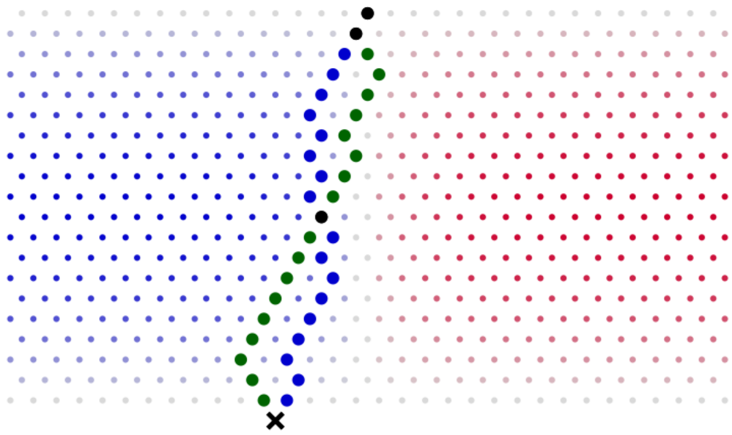


Traditional likelihood-free inference treats the simulator as a generative black box: parameters in, samples out.

But in most real-life problems, we have access to the simulator code and some understanding of the microscopic processes.



$p(x|\theta)$ is usually intractable. What about $p(x, z|\theta)$?



$p(x|\theta)$ is usually intractable. What about $p(x, z|\theta)$?

As the trajectory $z_1, ..., z_T$ and the observable x are emitted, it is often possible:

- to calculate the joint likelihood $p(x, z|\theta)$;
- to calculate the joint likelihood ratio $r(x, z|\theta_0, \theta_1)$;
- to calculate the joint score $t(x, z|\theta_0) = \nabla_\theta \log p(x, z|\theta)\big|_{\theta_0}$.

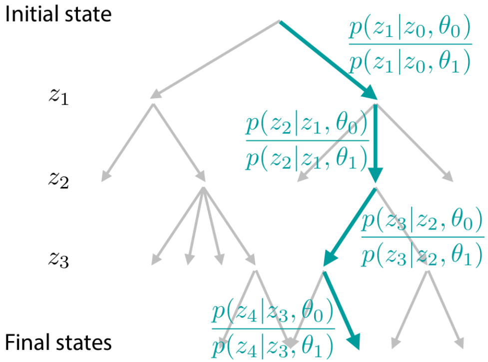
We call this process mining gold from your simulator!

Extracting the joint likelihood ratio

- Computer simulations typically evolve along a tree-like structure of successive random branchings.
- The probabilities of each branching $p(z_i|z_{i-1}, \theta)$ are often clearly defined in the code:

```
if random() > 0.1+2.5+model_parameter:
    do_one_thing()
else:
    do_another_thing()
```

- For each run, we can calculate the probability of the chosen path for different values of the parameters and the joint likelihood-ratio:



$$r(x, z|\theta_0, \theta_1) = \frac{p(x, z|\theta_0)}{p(x, z|\theta_1)} = \prod_i \frac{p(z_i|z_{i-1}, \theta_0)}{p(z_i|z_{i-1}, \theta_1)}$$

Credits: Johann Brehmer.

ALICE

When the joint likelihood ratio $r(x, z|\theta_0, \theta_1)$ is available from the simulator, the corresponding $s(x, z|\theta_0, \theta_1)$ are also tractable.

Therefore, the original CARL cross-entropy can be adapted to make use of the exact $s(x, z|\theta_0, \theta_1)$ instead of using labels $y \in \{0, 1\}$:

$$L_{ALICE}[\hat{s}] = -\mathbb{E}_{p(x,z)}[s(x, z|\theta_0, \theta_1) \log(\hat{s}(x)) + (1 - s(x, z|\theta_0, \theta_1)) \log(1 - \hat{s}(x))],$$

where $p(x, z) = (p(x, z|\theta_0) + p(x, z|\theta_1))/2$.

RASCAL

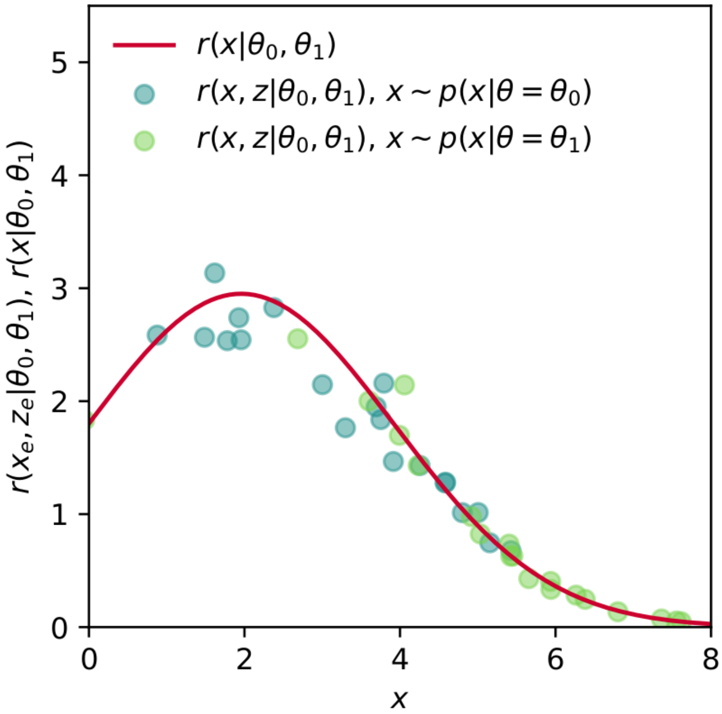
Regressing the likelihood ratio

Observe that the joint likelihood ratios

$$r(x, z|\theta_0, \theta_1) = \frac{p(x, z|\theta_0)}{p(x, z|\theta_1)}$$

are scattered around $r(x|\theta_0, \theta_1)$.

Can we use them to approximate $r(x|\theta_0, \theta_1)$?



Consider the squared error of a function $\hat{g}(x)$ that only depends on x, but is trying to approximate a function $g(x, z)$ that also depends on the latent z:

$$L_{\text{MSE}} = \mathbb{E}_{p(x,z|\theta)}\left[\left(g(x,z) - \hat{g}(x)\right)^2\right].$$

Via calculus of variations, we find that the function $g^*(x)$ that extremizes $L_{\text{MSE}}[g]$ is given by

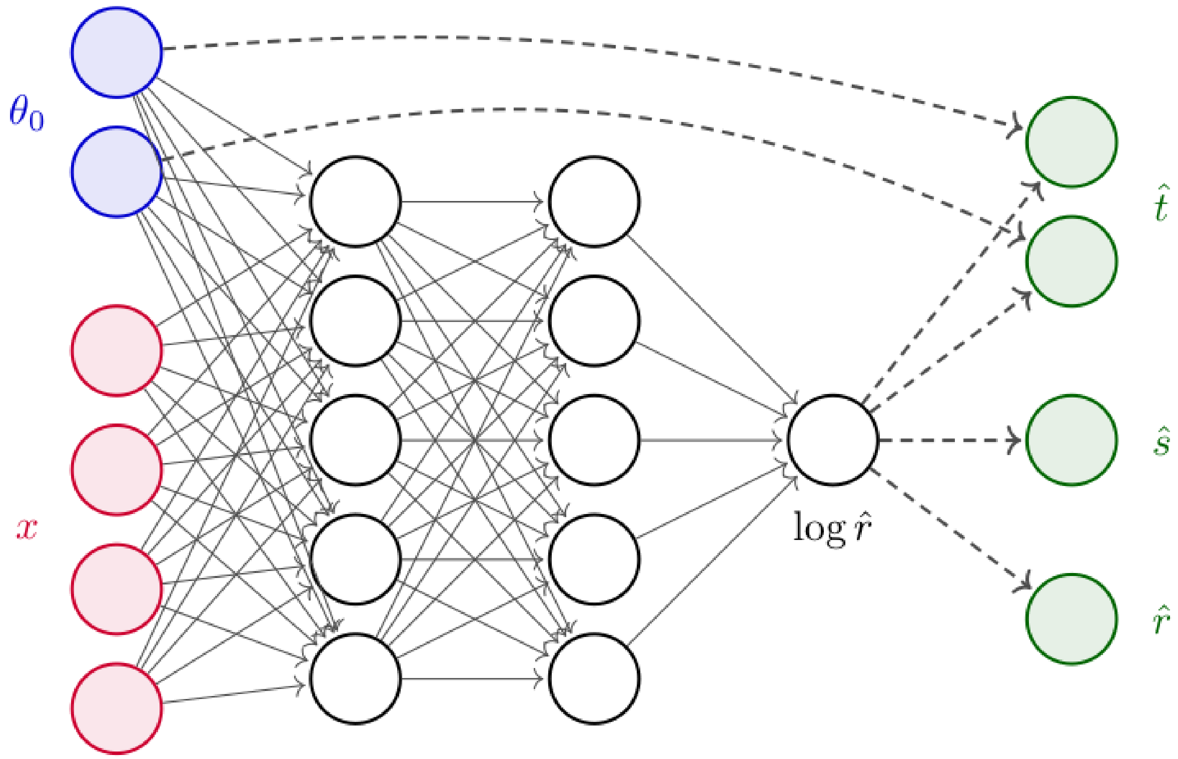
$$\begin{aligned} g^*(x) &= \frac{1}{p(x|\theta)} \int p(x,z|\theta) g(x,z) dz \\ &= \mathbb{E}_{p(z|x,\theta)}\left[g(x,z)\right] \end{aligned}$$

Therefore, by identifying the $g(x, z)$ with the joint likelihood ratio $r(x, z|\theta_0, \theta_1)$ and θ with θ_1, we define

$$L_r = \mathbb{E}_{p(x,z|\theta_1)} \left[\left(r(x, z|\theta_0, \theta_1) - \hat{r}(x) \right)^2 \right],$$

which is minimized by

$$\begin{aligned} r^*(x) &= \frac{1}{p(x|\theta_1)} \int p(x, z|\theta_1) \frac{p(x, z|\theta_0)}{p(x, z|\theta_1)} dz \\ &= \frac{p(x|\theta_0)}{p(x|\theta_1)} \\ &= r(x|\theta_0, \theta_1). \end{aligned}$$



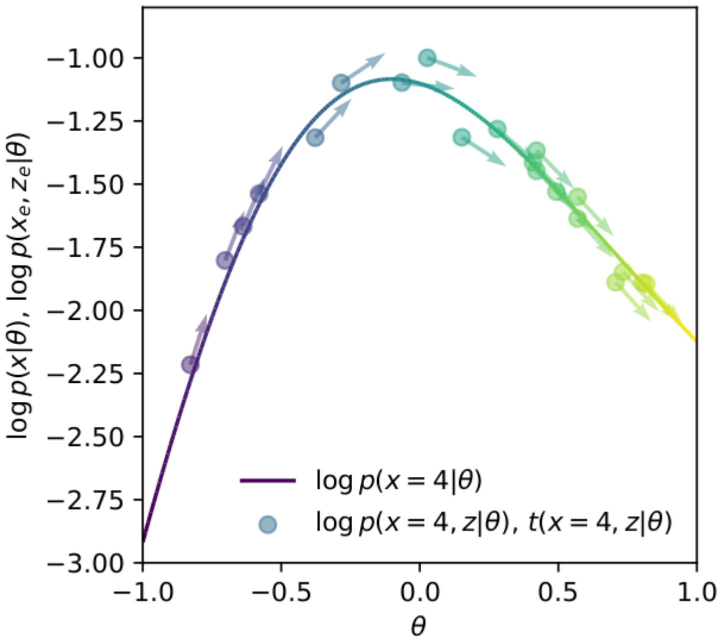
$$r^*(x|\theta_0, \theta_1) = \arg\min_{\hat{r}} L_r[\hat{r}]$$

Regressing the score

Similarly, we can mine the simulator to extract the joint score

$$t(x, z|\theta_0) = \nabla_\theta \log p(x, z|\theta)\Big|_{\theta_0},$$

which indicates how much more or less likely x, z would be if one changed θ_0.



Using the same trick, by identifying $g(x, z)$ with the joint score $t(x, z|\theta_0)$ and θ with θ_0, we define

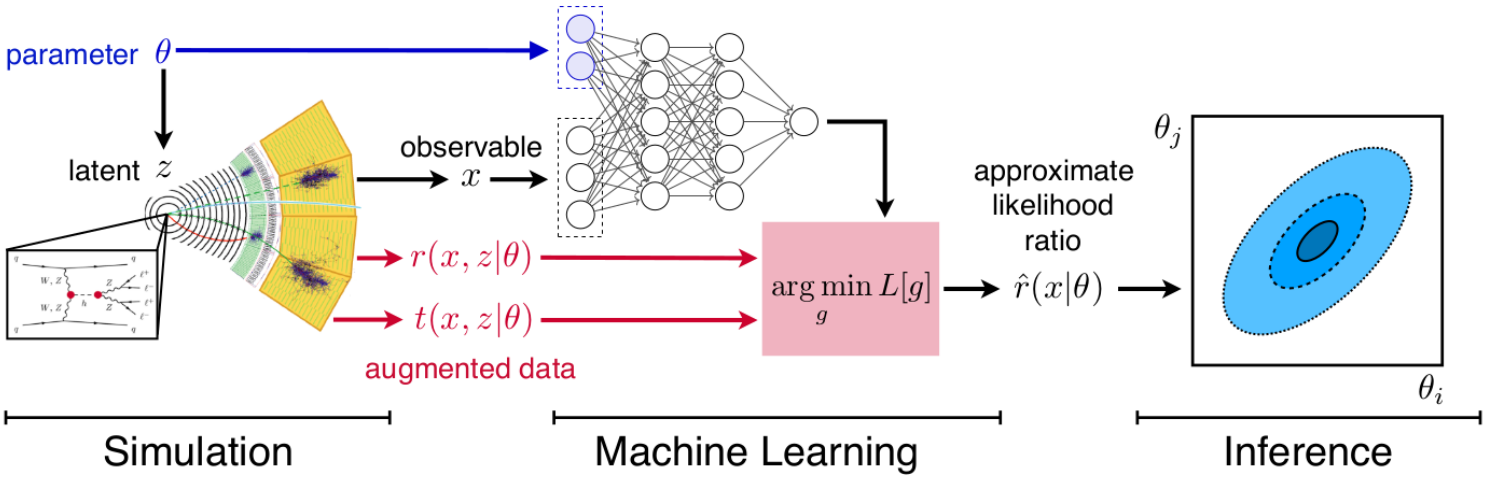
$$L_t = \mathbb{E}_{p(x,z|\theta_0)}\left[(t(x, z|\theta_0) - \hat{t}\,(x))^2\right],$$

which is minimized by

$$\begin{aligned}
t^*(x) &= \frac{1}{p(x|\theta_0)} \int p(x, z|\theta_0)(\nabla_\theta \log p(x, z|\theta)\big|_{\theta_0})dz \\
&= \frac{1}{p(x|\theta_0)} \int p(x, z|\theta_0) \frac{\nabla_\theta p(x, z|\theta)\big|_{\theta_0}}{p(x, z|\theta_0)} dz \\
&= \frac{\nabla_\theta p(x|\theta)\big|_{\theta_0}}{p(x|\theta_0)} \\
&= \nabla_\theta \log p(x|\theta)\big|_{\theta_0} \\
&= t(x|\theta_0).
\end{aligned}$$

Rascal

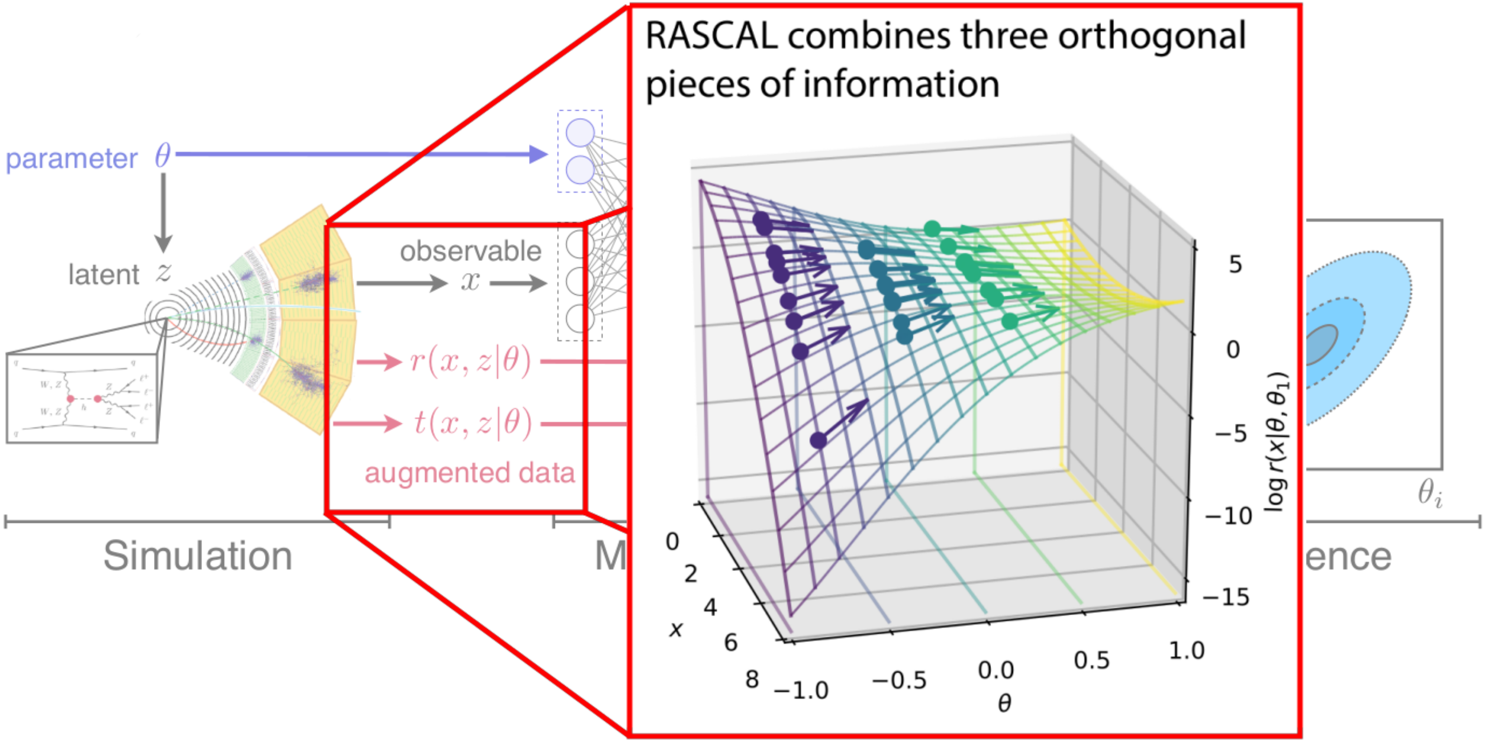
$$L_{\text{RASCAL}} = L_r + L_t$$



Brehmer, Louppe, Pavez and Cranmer, 2018 [arXiv:1805.12244]

RASCAL

$$L_{\text{RASCAL}} = L_r + L_t$$

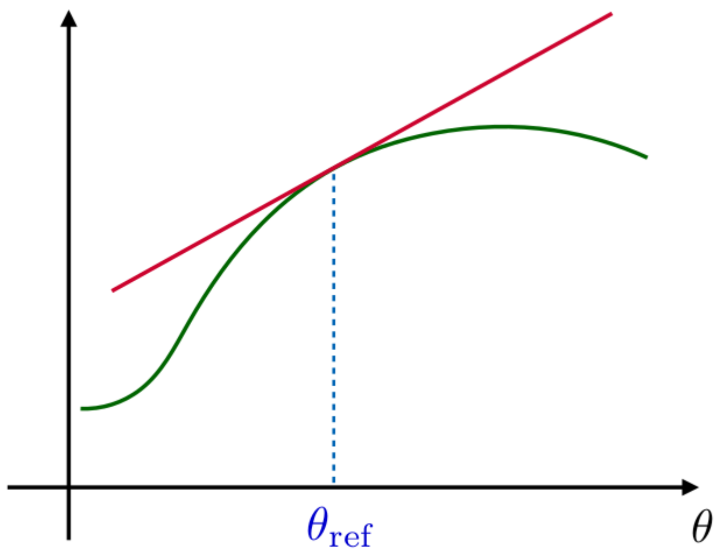


SALLY (= optimal compression)

The local model

In the neighborhood of θ_{ref}, the Taylor expansion of $\log p(x|\theta)$ is

$$\log p(x|\theta) = \log p(x|\theta_{\text{ref}}) + \underbrace{\nabla_\theta \log p(x|\theta)\Big|_{\theta_{\text{ref}}}}_{t(x|\theta_{\text{ref}})} \cdot (\theta - \theta_{\text{ref}}) + O((\theta - \theta_{\text{ref}})^2)$$



This results in the exponential model

$$p_{\text{local}}(x|\theta) = \frac{1}{Z(\theta)} p(t(x|\theta_{\text{ref}})|\theta_{\text{ref}}) \exp(t(x|\theta_{\text{ref}}) \cdot (\theta - \theta_{\text{ref}}))$$

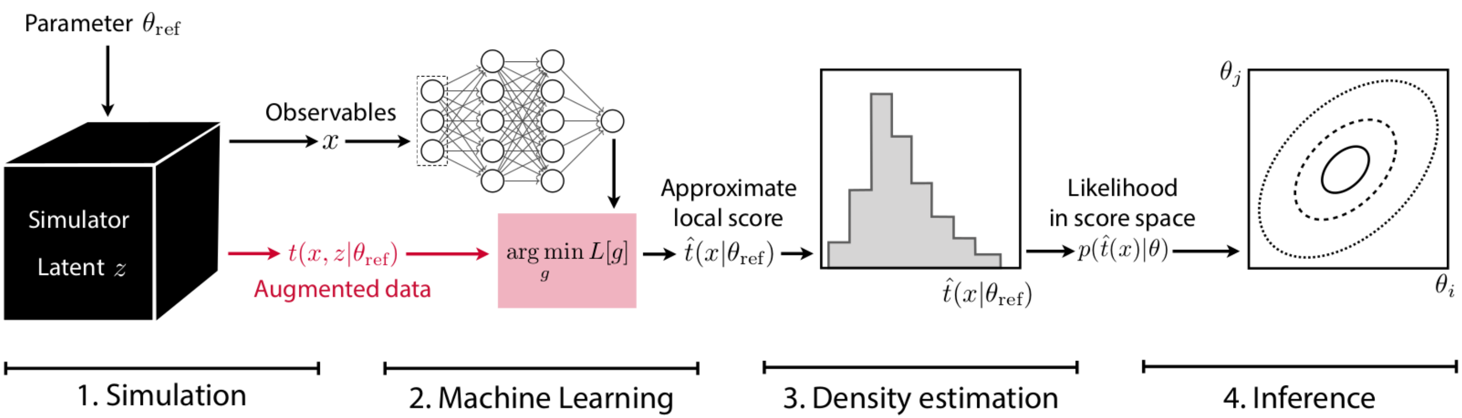
where the score $t(x|\theta_{\text{ref}})$ are its sufficient statistics.

That is,

- knowing $t(x|\theta_{\text{ref}})$ is just as powerful as knowing the full function $\log p(x|\theta)$.
- x can be compressed into a single scalar $t(x|\theta_{\text{ref}})$ without loss of power.

Brehmer, Louppe, Pavez and Cranmer, 2018 [arXiv:1805.12244].

SALLY



Brehmer, Louppe, Pavez and Cranmer, 2018 [arXiv:1805.12244].

There is more...

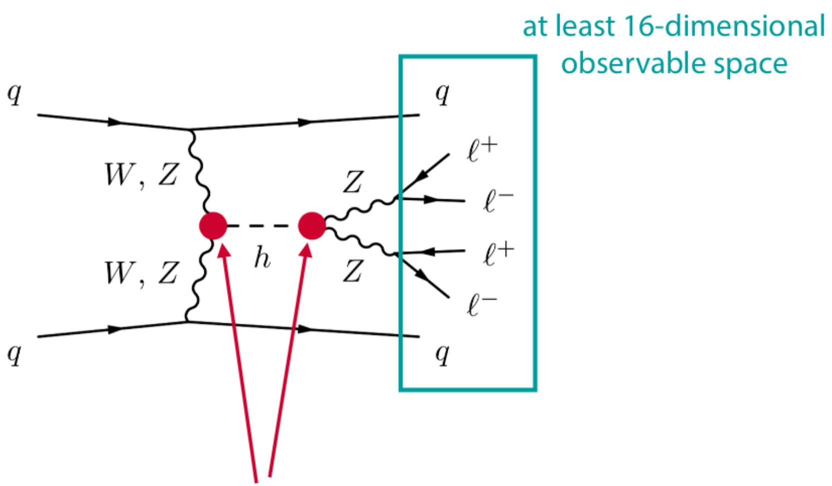
Method	Simulate	Extract $r(x,z)$	Extract $t(x,z)$	NN estimates	Asympt. exact	Generative
ROLR	$\theta_0 \sim \pi(\theta)$, θ_1	✓		$\hat{r}(x\|\theta_0,\theta_1)$	✓	
CASCAL	$\theta_0 \sim \pi(\theta)$, θ_1		✓	$\hat{r}(x\|\theta_0,\theta_1)$	✓	
ALICE	$\theta_0 \sim \pi(\theta)$, θ_1		✓	$\hat{r}(x\|\theta_0,\theta_1)$	✓	
RASCAL	$\theta_0 \sim \pi(\theta)$, θ_1	✓	✓	$\hat{r}(x\|\theta_0,\theta_1)$	✓	
ALICES	$\theta_0 \sim \pi(\theta)$, θ_1	✓	✓	$\hat{r}(x\|\theta_0,\theta_1)$	✓	
SCANDAL	$\theta \sim \pi(\theta)$		✓	$\hat{p}(x\|\theta)$	✓	✓
SALLY	θ_{ref}		✓	$\hat{t}(x\|\theta_{\text{ref}})$	in local approx.	
SALLINO	θ_{ref}		✓	$\hat{t}(x\|\theta_{\text{ref}})$	in local approx.	

Brehmer, Louppe, Pavez and Cranmer, 2018 [arXiv:1805.12244].

Examples

① Hunting new physics at particle colliders

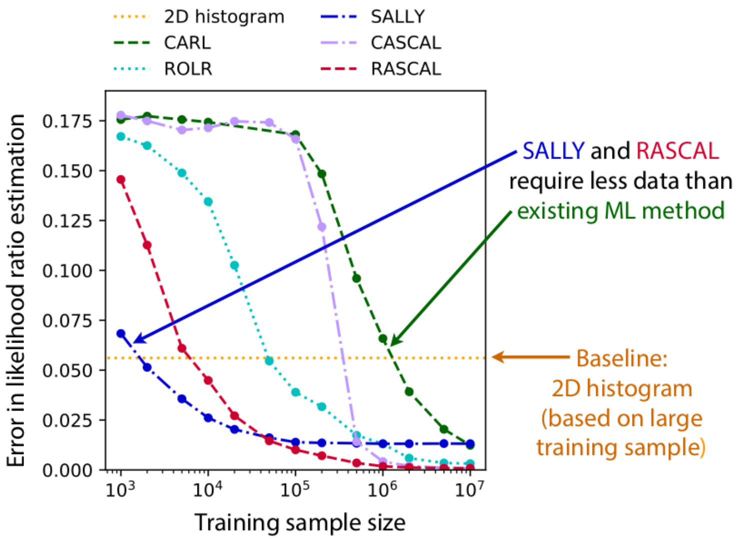
The goal is to constrain two EFT parameters and compare against traditional histogram analysis.

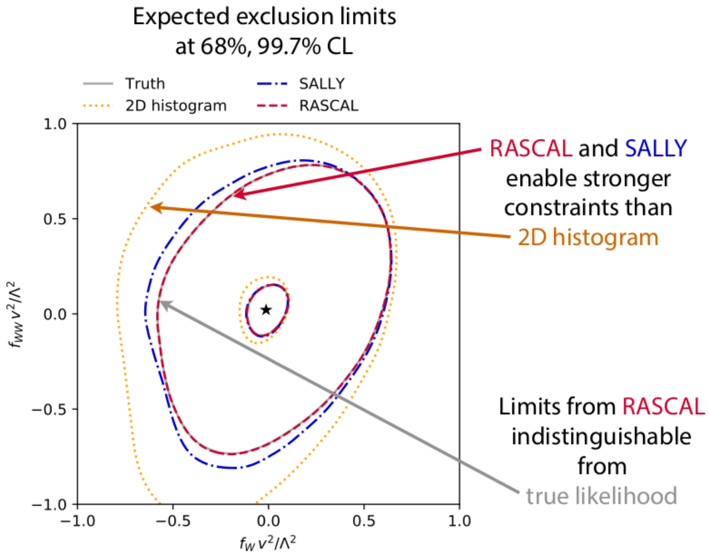


Exciting new physics might hide here!
We parameterize it with two EFT coefficients:

$$\mathcal{L} = \mathcal{L}_{\mathrm{SM}} + \boxed{\frac{f_W}{\Lambda^2}} \underbrace{\frac{ig}{2} \, (D^\mu \phi)^\dagger \, \sigma^a \, D^\nu \phi \, W^a_{\mu\nu}}_{\mathcal{O}_W} - \boxed{\frac{f_{WW}}{\Lambda^2}} \underbrace{\frac{g^2}{4} \, (\phi^\dagger \phi) \, W^a_{\mu\nu} \, W^{\mu\nu\, a}}_{\mathcal{O}_{WW}}$$

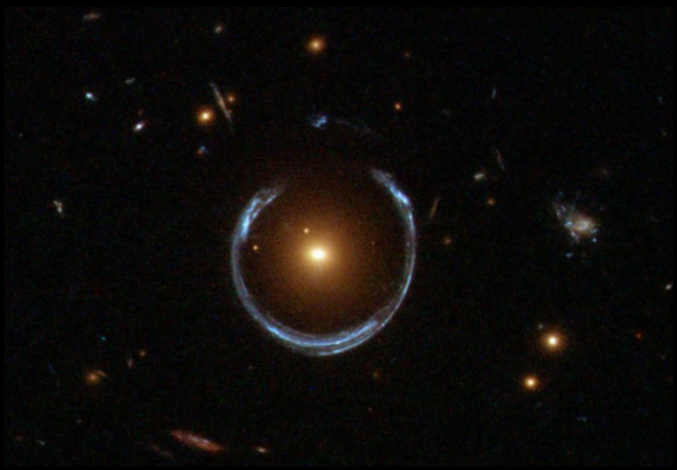
Brehmer, Cranmer, Louppe, and Pavez, 2018a [arXiv:1805.00020], 2018b [arXiv:1805.00013]; Brehmer, Louppe, Pavez and Cranmer, 2018 [arXiv:1805.12244].

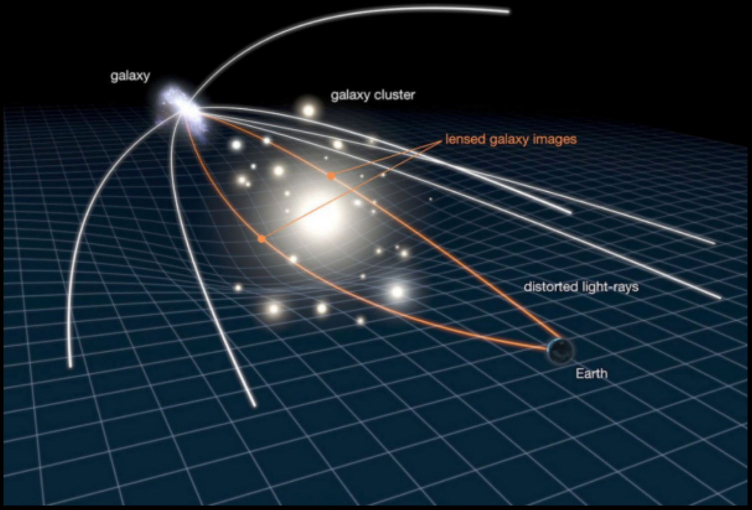




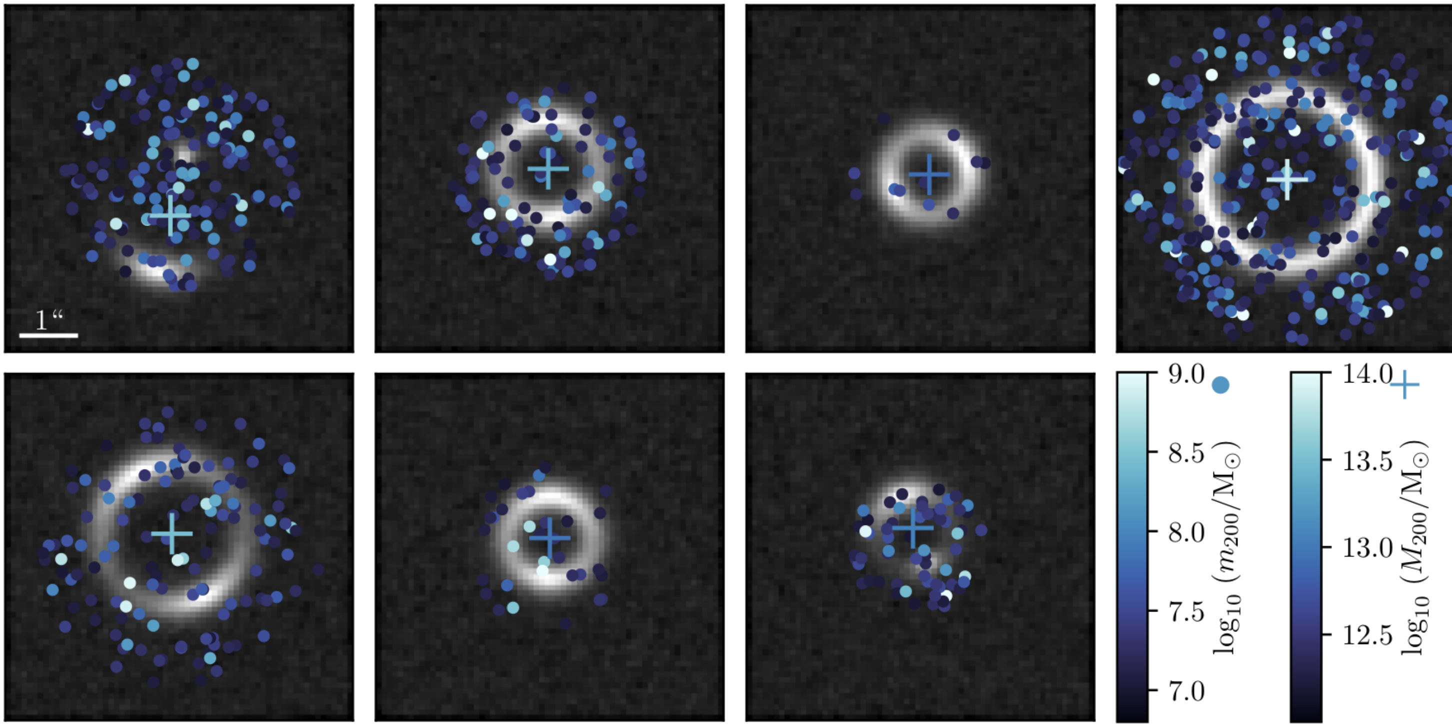
Brehmer, Cranmer, Louppe, and Pavez, 2018a [arXiv:1805.00020], 2018b [arXiv:1805.00013]; Brehmer, Louppe, Pavez and Cranmer, 2018 [arXiv:1805.12244].

② Dark matter substructure from gravitational lensing



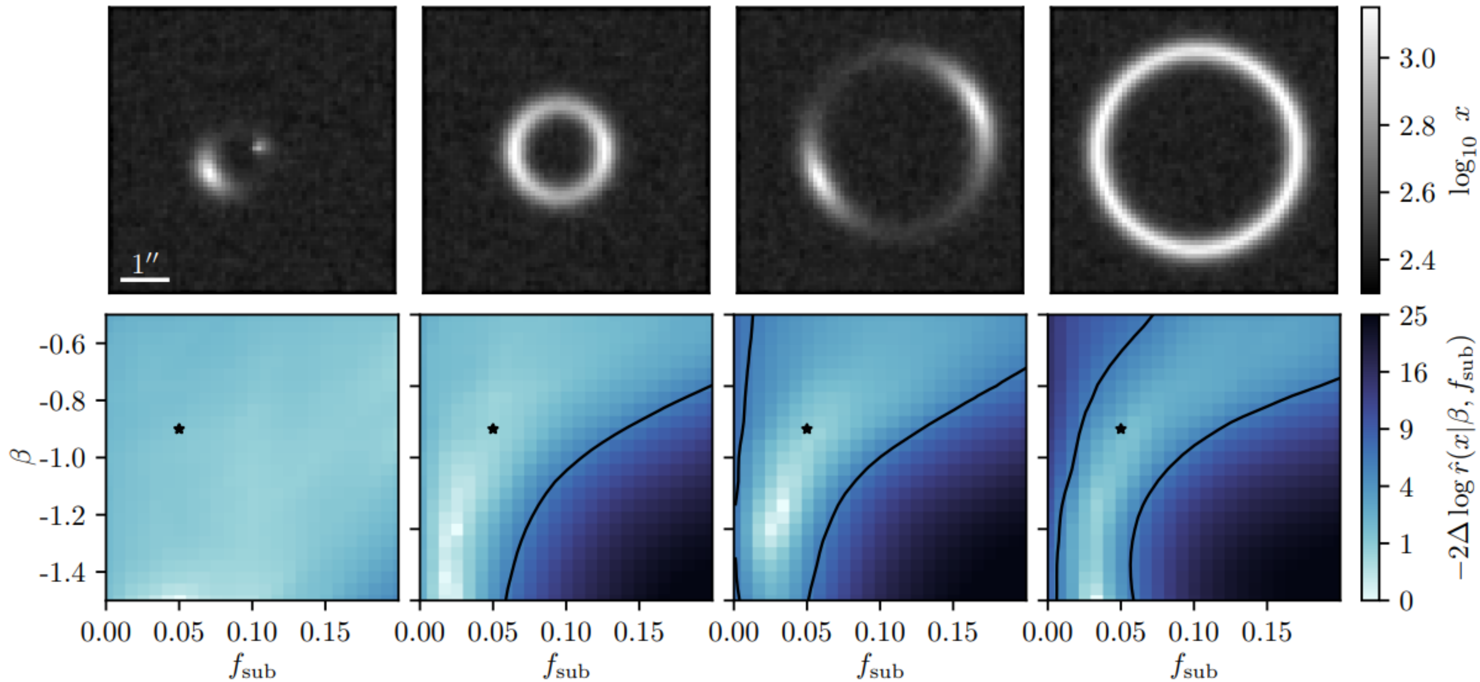


Brehmer, Mishra-Sharma, Hermans, Louppe, and Cranmer, 2019 [arXiv:1909.02005].



Number of dark matter subhalos and their mass and location lead to complex latent space of each image. The goal is the **inference of population parameters.**

Brehmer, Mishra-Sharma, Hermans, Louppe, and Cranmer, 2019 [arXiv:1909.02005].

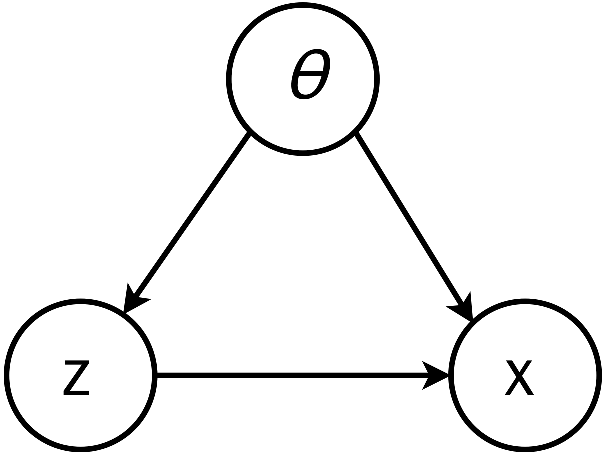


1″
3.0
2.8
2.6
2.4
$\log_{10} x$
-0.6
-0.8
-1.0
-1.2
-1.4
β
0.00 0.05 0.10 0.15
$f_{\rm sub}$
25
16
9
4
1
0
$-2\Delta \log \hat{r}(x|\beta, f_{\rm sub})$

Bayesian inference

Bayesian inference = computing the posterior

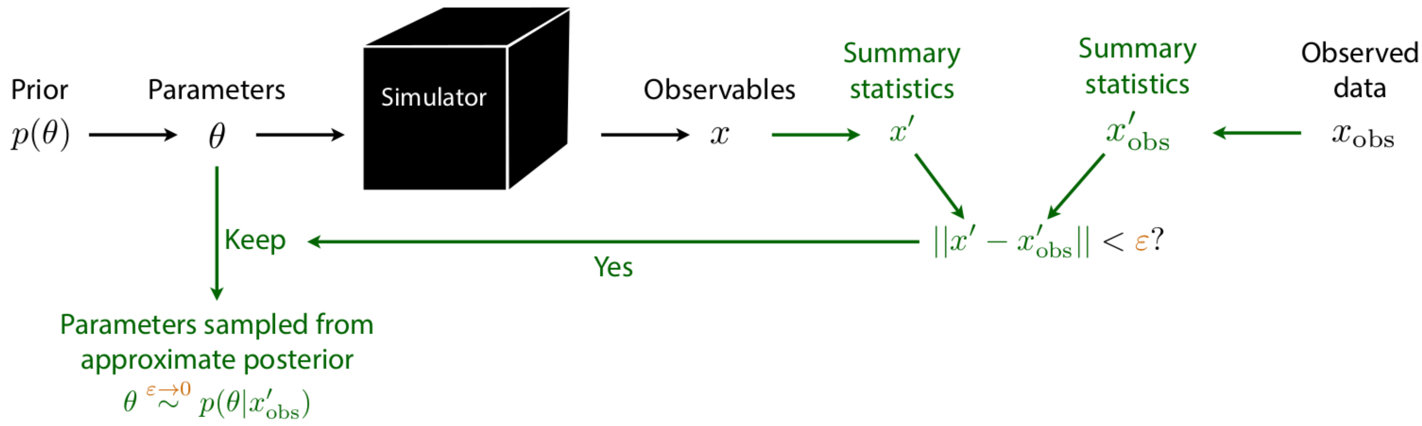
$$p(\theta|x) = \frac{p(x|\theta)p(\theta)}{p(x)}.$$



Doubly intractable in the likelihood-free scenario:

- Cannot evaluate the likelihood $p(x|\theta) = \int p(x, z|\theta)dz$.
- Cannot evaluate the evidence $p(x) = \int p(x|\theta)p(\theta)d\theta$.

Approximate Bayesian Computation (ABC)



Issues

- How to choose x'? ϵ? $|| \cdot ||$?
- No tractable posterior.
- Need to run new simulations for new data or new prior.

Credits: Johann Brehmer.

Amortizing Bayes

The Bayes rule can be rewritten as

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

where $r(x|\theta) = \frac{p(x|\theta)}{p(x)}$ is the likelihood-to-evidence ratio.

Amortizing Bayes

The Bayes rule can be rewritten as

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

where $r(x|\theta) = \frac{p(x|\theta)}{p(x)}$ is the likelihood-to-evidence ratio.

As before, the likelihood-to-evidence ratio can be approximated e.g. from a neural network classifier trained to distinguish $x \sim p(x|\theta)$ from $x \sim p(x)$, hence enabling direct and amortized posterior evaluation.

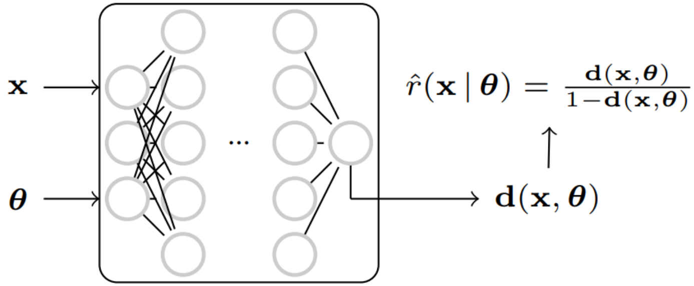
Algorithm 1 Optimization of $\mathbf{d}(\mathbf{x}, \boldsymbol{\theta})$.

Inputs: Criterion ℓ (e.g., BCE)
Implicit generative model $p(\mathbf{x} \mid \boldsymbol{\theta})$
Prior $p(\boldsymbol{\theta})$

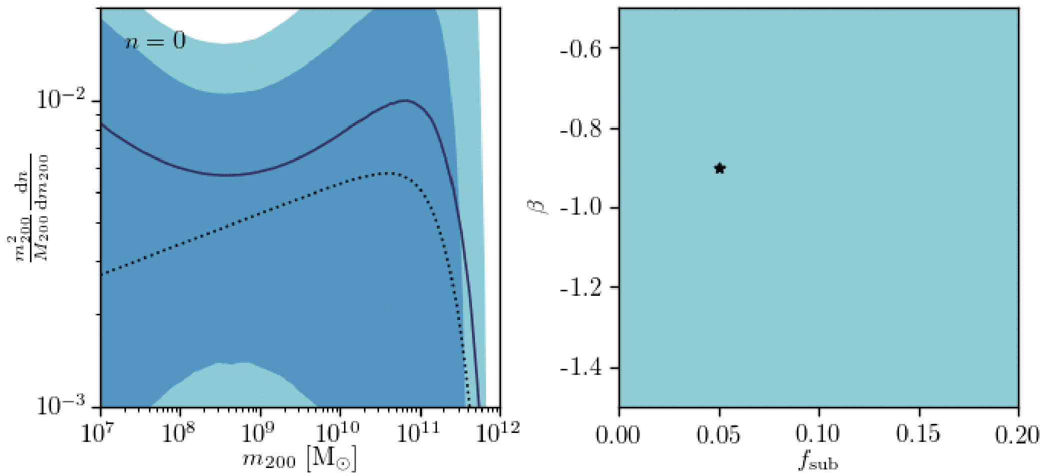
Outputs: Parameterized classifier $\mathbf{d}_\phi(\mathbf{x}, \boldsymbol{\theta})$

Hyperparameters: Batch-size M

while not converged **do**
    Sample $\boldsymbol{\theta} \leftarrow \{\boldsymbol{\theta}_m \sim p(\boldsymbol{\theta})\}_{m=1}^M$
    Sample $\boldsymbol{\theta}' \leftarrow \{\boldsymbol{\theta}'_m \sim p(\boldsymbol{\theta})\}_{m=1}^M$
    Simulate $\mathbf{x} \leftarrow \{\mathbf{x}_m \sim p(\mathbf{x} \mid \boldsymbol{\theta}_m)\}_{m=1}^M$
    $\mathcal{L} \leftarrow \ell(\mathbf{d}_\phi(\mathbf{x}, \boldsymbol{\theta}),\ 1) + \ell(\mathbf{d}_\phi(\mathbf{x}, \boldsymbol{\theta}'),\ 0)$
    $\phi \leftarrow \text{OPTIMIZER}(\phi,\ \nabla_\phi \mathcal{L})$
end while
return $\mathbf{d}_\phi$

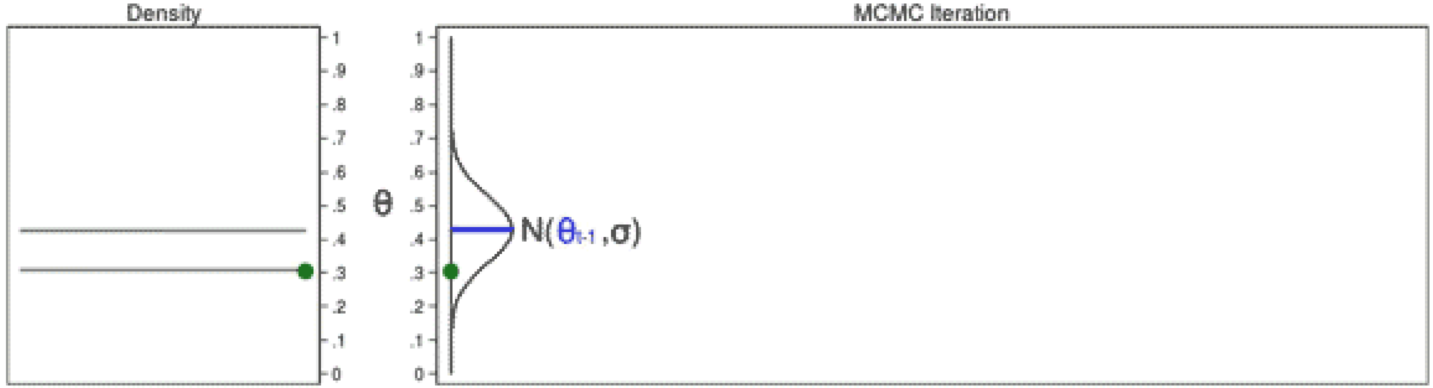


Bayesian inference of dark matter subhalo population parameters



Brehmer, Mishra-Sharma, Hermans, Louppe, and Cranmer, 2019 [arXiv:1909.02005].

MCMC posterior sampling



Step 1: $r(\theta_{new}, \theta_{t-1}) = \dfrac{\text{Posterior}(\theta_{new})}{\text{Posterior}(\theta_{t-1})} = \dfrac{\text{Beta}(1,1,0.306) \times \text{Binomial}(10,4,0.306)}{\text{Beta}(1,1,0.429) \times \text{Binomial}(10,4,0.429)} = 0.834$

Step 2: Acceptance probability $\alpha(\theta_{new}, \theta_{t-1}) = \min\{r(\theta_{new}, \theta_{t-1}), 1\} = \min\{0.834, 1\} = 0.834$

Step 3: Draw $u \sim \text{Uniform}(0,1) = 0.617$

Step 4: If $u < \alpha(\theta_{new}, \theta_{t-1})$ → If $0.617 < 0.834$ Then $\theta_t = \theta_{new} = 0.306$

Otherwise $\theta_t = \theta_{t-1} = 0.429$

Credits: Chuck Huber, 2016.

Likelihood-free MCMC

MCMC samplers require the evaluation of the posterior ratios:

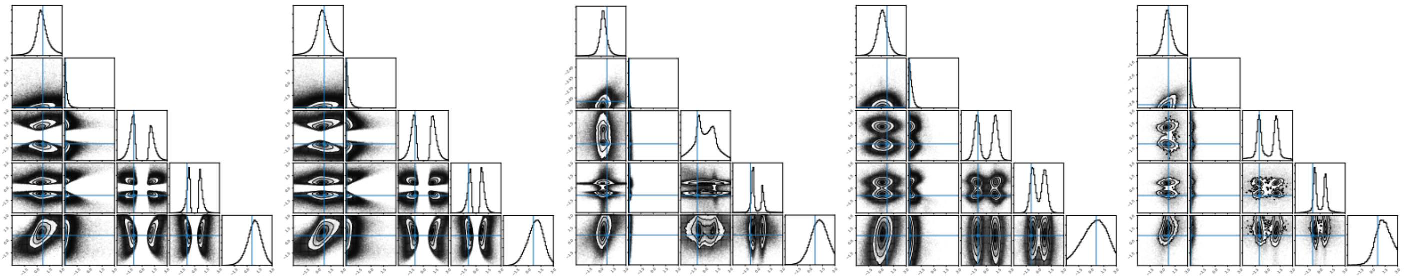
$$\begin{aligned}\frac{p(\theta_{\text{new}}|x)}{p(\theta_{t-1}|x)} &= \frac{p(x|\theta_{\text{new}})p(\theta_{\text{new}})/p(x)}{p(x|\theta_{t-1})p(\theta_{t-1})/p(x)} \\ &= \frac{p(x|\theta_{\text{new}})p(\theta_{\text{new}})}{p(x|\theta_{t-1})p(\theta_{t-1})} \\ &= r(x|\theta_{\text{new}}, \theta_{t-1})\frac{p(\theta_{\text{new}})}{p(\theta_{t-1})}\end{aligned}$$

Again, MCMC samplers can be made likelihood-free by plugging a learned approximation $\hat{r}(x|\theta_{\text{new}}, \theta_{t-1})$ of the likelihood ratio.

For MCMC, best results are obtained when using ratios of likelihood-to-evidence ratios:

$$\hat{r}(x|\theta_{\text{new}}, \theta_{t-1}) = \frac{\hat{r}(x|\theta_{\text{new}})}{\hat{r}(x|\theta_{t-1})}$$

Hermans, Begy and Louppe, 2019 [arXiv:1903.04057].



(a) MCMC groundtruth (b) AALR-MCMC (c) SNPE-A (d) SNPE-B (e) APT

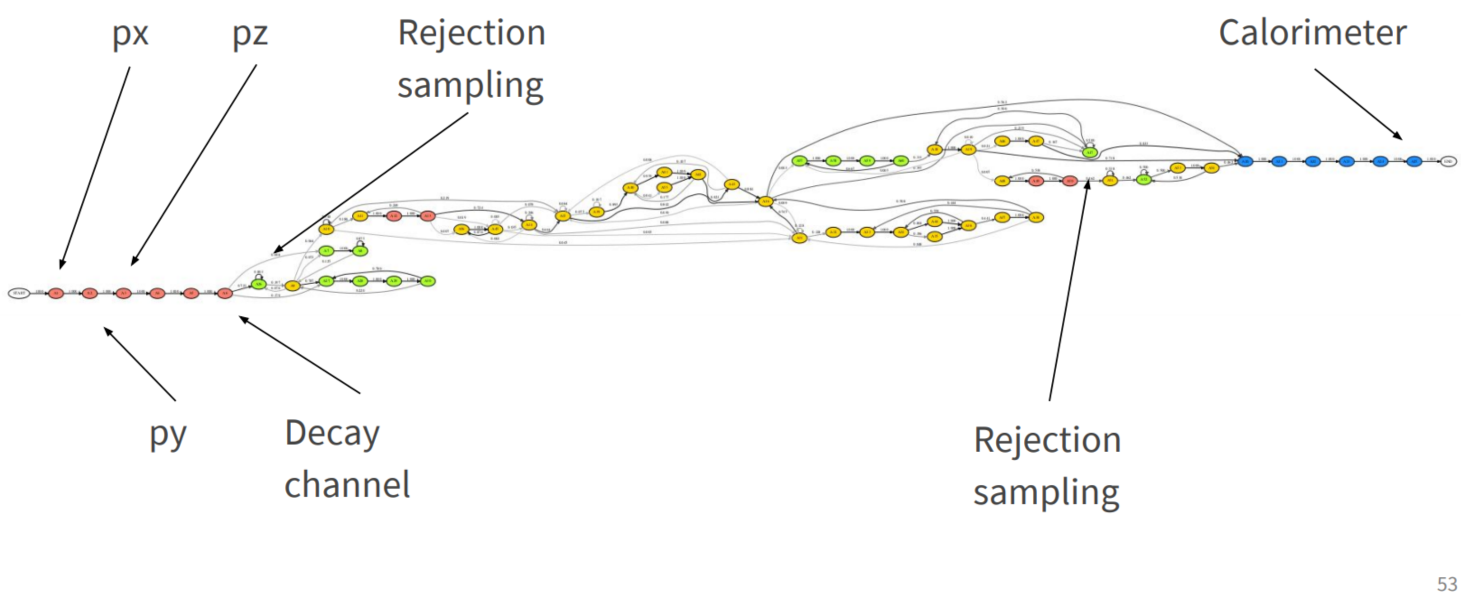
Figure 3: Posteriors from the tractable benchmark. The experiments are repeated 25 times and the approximate posteriors are subsampled from those runs. AALR-MCMC shares the same structure with the MCMC truth, demonstrating its accuracy. Some runs of the other methods were not consistent, contributing to the variance observed in Table 2.

Algorithm	MMD	ROC AUC
AALR-MCMC (ours)	$\mathbf{0.05} \pm 0.005$	$\mathbf{0.59} \pm 0.0010$
ABC ($\epsilon = 32$)	0.51 ± 0.001	0.99 ± 0.0001
ABC ($\epsilon = 16$)	0.50 ± 0.003	0.99 ± 0.0002
ABC ($\epsilon = 8$)	0.39 ± 0.001	0.99 ± 0.0003
ABC ($\epsilon = 4$)	0.29 ± 0.004	0.98 ± 0.0007
APT	0.17 ± 0.036	0.86 ± 0.0008
AALR-MCMC (LRT)	0.53 ± 0.004	0.99 ± 0.0001
SNPE-A	0.21 ± 0.070	0.97 ± 0.0098
SNPE-B	0.20 ± 0.061	0.92 ± 0.0181

Table 2: AALR-MCMC outperforms all other methods. Numerical errors introduced by MCMC might have contributed to these results. A comparison of the PDFs between the true posterior and our ratio estimator are shown in Figure 11 (Appendix D.2). The MMD scores are in agreement with [41].

Hermans, Begy and Louppe, 2019 [arXiv:1903.04057].

Probabilistic programming

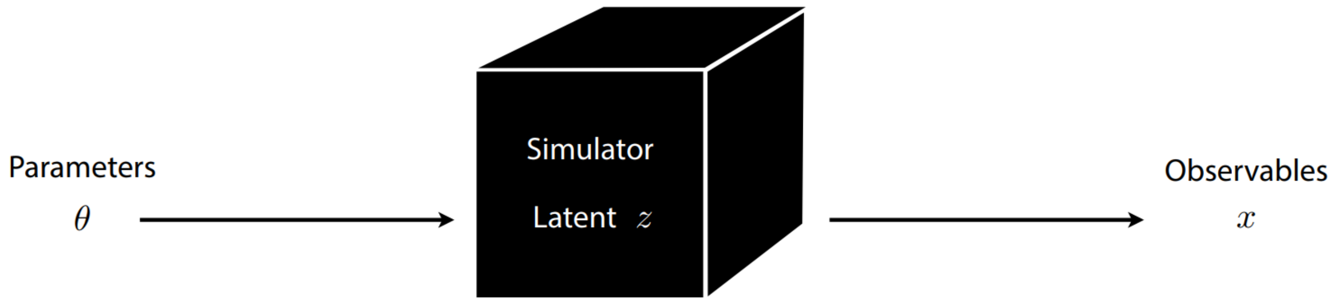


See Lukas' talk after the coffee break!

Summary

Summary

- Much of modern science is based on "likelihood-free" simulations.
- The likelihood-ratio is central to many statistical inference procedures, regardless of your religion.
- Supervised learning enables likelihood-ratio estimation.
- Better likelihood-ratio estimates can be achieved by mining simulators.
- (Probabilistic programming enables posterior inference in scientific simulators.)



Collaborators

References

- Brehmer, J., Mishra-Sharma, S., Hermans, J., Louppe, G., Cranmer, K. (2019). Mining for Dark Matter Substructure: Inferring subhalo population properties from strong lenses with machine learning. arXiv preprint arXiv 1909.02005.
- Hermans, J., Begy, V., & Louppe, G. (2019). Likelihood-free MCMC with Approximate Likelihood Ratios. arXiv preprint arXiv:1903.04057.
- Baydin, A. G., Shao, L., Bhimji, W., Heinrich, L., Meadows, L., Liu, J., ... & Ma, M. (2019). Etalumis: Bringing Probabilistic Programming to Scientific Simulators at Scale. arXiv preprint arXiv:1907.03382.
- Stoye, M., Brehmer, J., Louppe, G., Pavez, J., & Cranmer, K. (2018). Likelihood-free inference with an improved cross-entropy estimator. arXiv preprint arXiv:1808.00973.
- Baydin, A. G., Heinrich, L., Bhimji, W., Gram-Hansen, B., Louppe, G., Shao, L., ... & Wood, F. (2018). Efficient Probabilistic Inference in the Quest for Physics Beyond the Standard Model. arXiv preprint arXiv:1807.07706.
- Brehmer, J., Louppe, G., Pavez, J., & Cranmer, K. (2018). Mining gold from implicit models to improve likelihood-free inference. arXiv preprint arXiv:1805.12244.
- Brehmer, J., Cranmer, K., Louppe, G., & Pavez, J. (2018). Constraining Effective Field Theories with Machine Learning. arXiv preprint arXiv:1805.00013.
- Brehmer, J., Cranmer, K., Louppe, G., & Pavez, J. (2018). A Guide to Constraining Effective Field Theories with Machine Learning. arXiv preprint arXiv:1805.00020.
- Casado, M. L., Baydin, A. G., Rubio, D. M., Le, T. A., Wood, F., Heinrich, L., ... & Bhimji, W. (2017). Improvements to Inference Compilation for Probabilistic Programming in Large-Scale Scientific Simulators. arXiv preprint arXiv:1712.07901.
- Louppe, G., Hermans, J., & Cranmer, K. (2017). Adversarial Variational Optimization of Non-Differentiable Simulators. arXiv preprint arXiv:1707.07113.
- Cranmer, K., Pavez, J., & Louppe, G. (2015). Approximating likelihood ratios with calibrated discriminative classifiers. arXiv preprint arXiv:1506.02169.

The end.