Likelihood-free inference, effectively.

With its applications at the LHC.

Prof. Gilles Louppe
g.louppe@uliege.be

Refs: GaltonBoard.com

Sir Francis Galton saw the quincunx as an analogy for the inheritance of genetic traits like stature. The pinballs accumulate in a bell-shaped curve that is similar to the distribution of human heights.

The puzzle of why human heights do not spread out from one generation to the next, as the balls would, led him to the discovery of "regression to the mean".

FIG. 7.

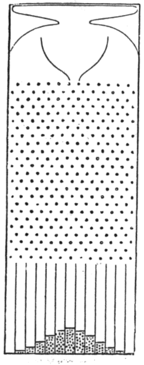


FIG. 8.

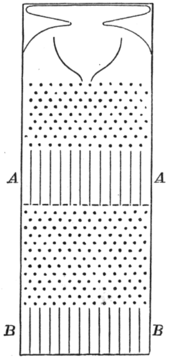
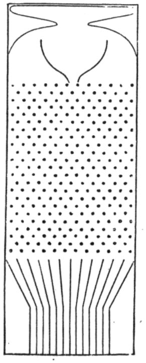
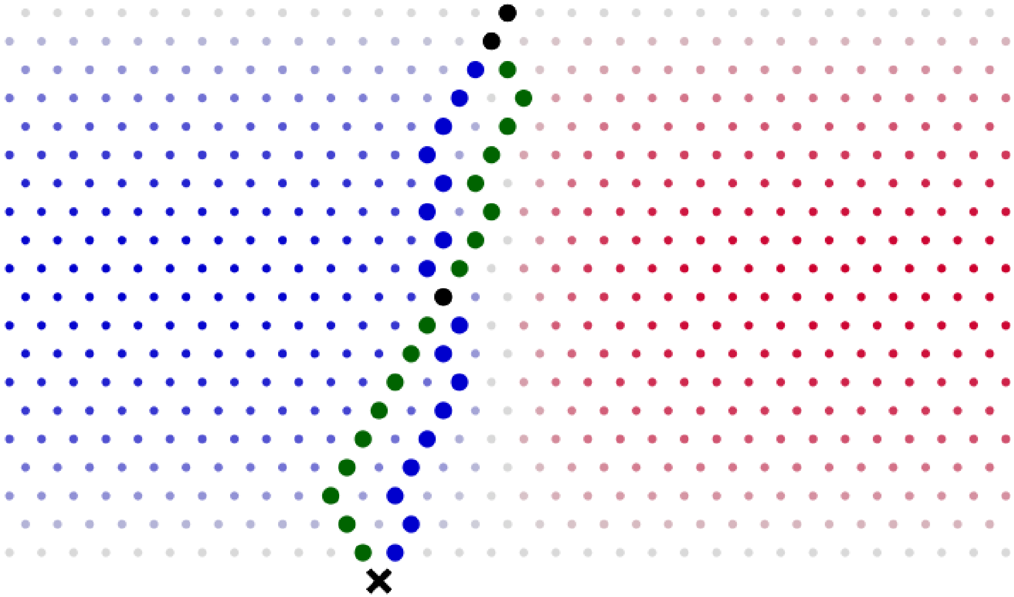


FIG. 9.





The probability of ending in bin x corresponds to the cumulative probability of all the paths z from start to x.

$$p(x) = \int_{\mathcal{Z}} p(x, z) dP_{\mathcal{Z}}$$

Assume pins all have the same effect on the balls.

Each time a ball hits a pin on its way down, it either bounces right with probability θ or left with probability $1-\theta$.

Therefore, at the last row n, each ball arrives in bin x (for $0 \leq k \leq n$) if and only if it has taken exactly x right turns (regardless of their position). This occurs with probability

$$\begin{aligned} p(x|\theta) &= \int_{\mathcal{Z}} p(x, z|\theta) dP_{\mathcal{Z}} \\ &= \binom{n}{x} \theta^x (1-\theta)^{n-x}. \end{aligned}$$

That is, the ball distribution over the bins corresponds to a binomial distribution.

Inference

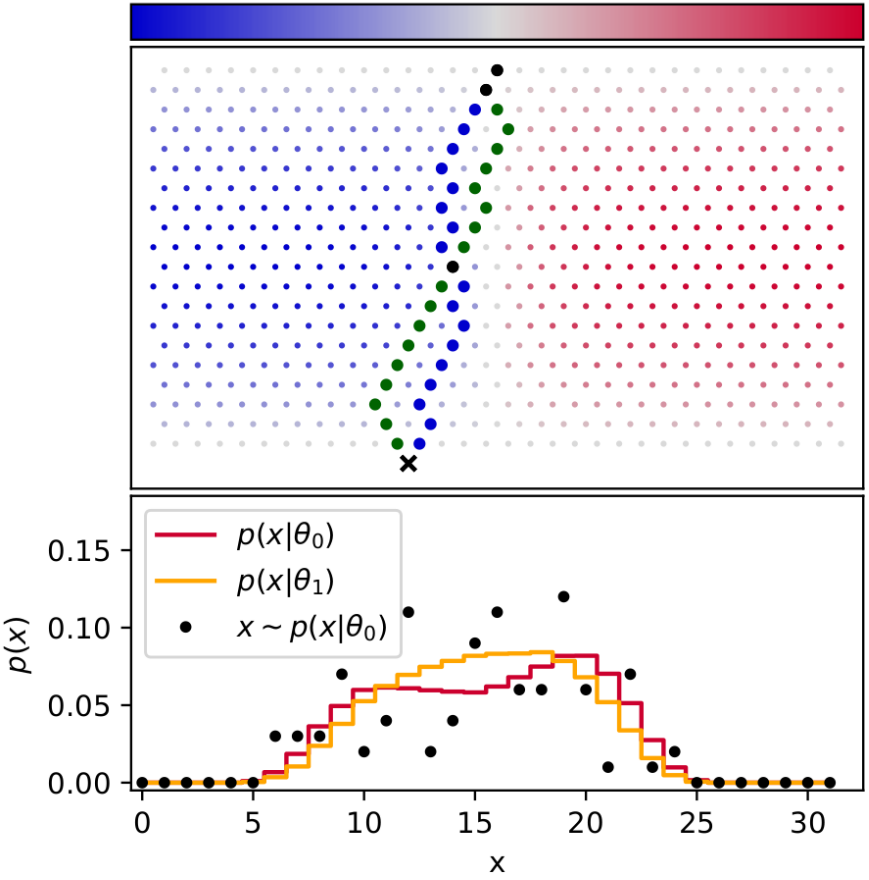
Given a set of realizations $\mathbf{d} = \{x_i\}$ at the bins, inference consists in determining the value of θ that best describes these observations.

For example, following the principle of maximum likelihood estimation, we have

$$\hat{\theta} = \arg\max_{\theta} \prod_{x_i \in \mathbf{d}} p(x_i|\theta).$$

In general, when $p(x_i|\theta)$ can be evaluated, this problem can be solved either analytically or using optimization algorithms.

What if pins are placed asymmetrically, such that the probability of bouncing right at (i, j) is different from the probability at (i', j'), but still indirectly depends on some parameters θ?



The probability of ending in bin x still corresponds to the cumulative probability of all the paths from start to x:

$$p(x|\theta) = \int_{\mathcal{Z}} p(x, z|\theta) dP_{\mathcal{Z}}$$

- But this integral can no longer be simplified analytically!
- As n grows larger, evaluating $p(x|\theta)$ becomes intractable since the number of paths grows combinatorially.
- Generating observations remains easy: drop the balls.

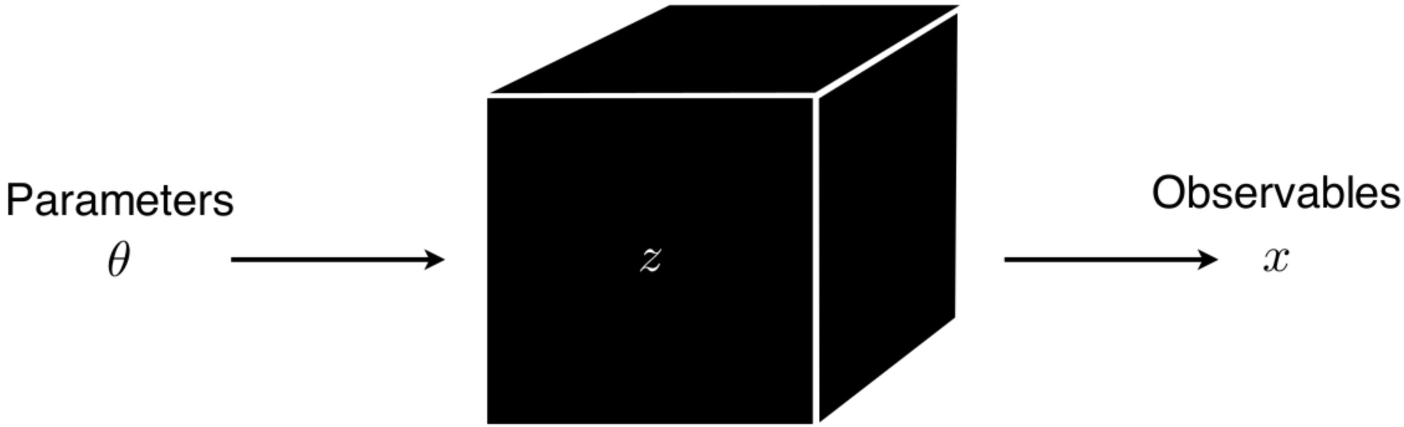
Since $p(x|\theta)$ cannot be evaluated, does this mean inference is no longer possible?

No! But we do need new tools.

The Galton board is a metaphore for the simulator-based scientific method:

- the Galton board device is the equivalent of the scientific simulator
- θ are parameters of interest
- z are stochastic execution traces through the simulator
- x are observables

For the same reasons, inference in this context requires likelihood-free algorithms.



Prediction (simulation):

- Well-understood mechanistic model
- Simulator can generate samples

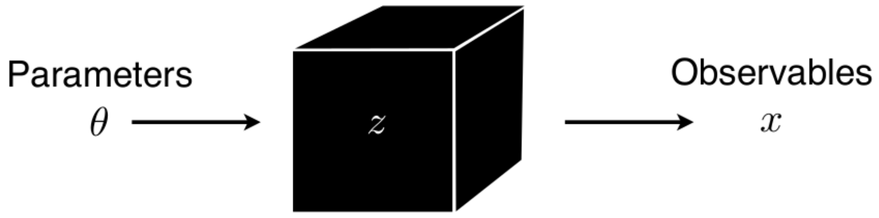
Inference:

- Likelihood function $p(x|\theta)$ is intractable
- Goal: estimator $\hat{p}(x|\theta)$

Credits: Johann Brehmer

Applications

Cosmological N-body simulations



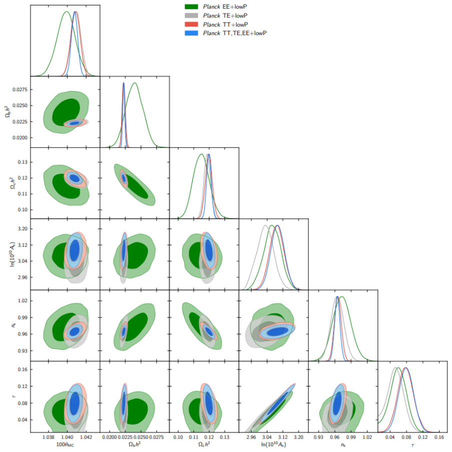
Parameters

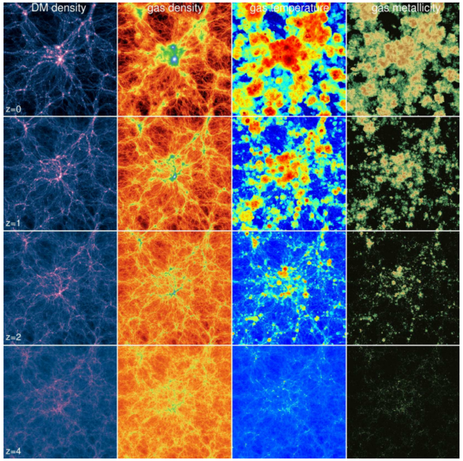
θ

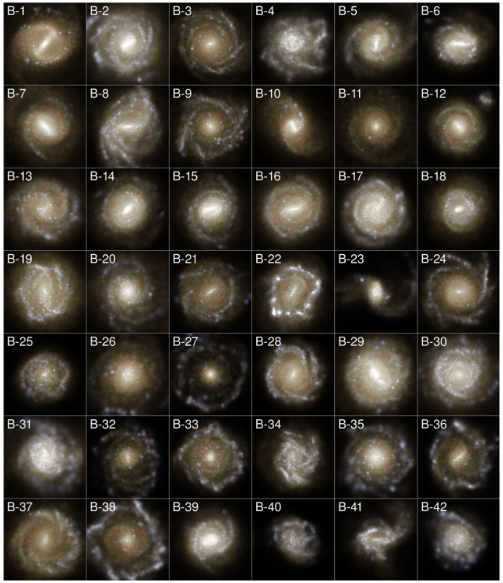
z

Observables

x

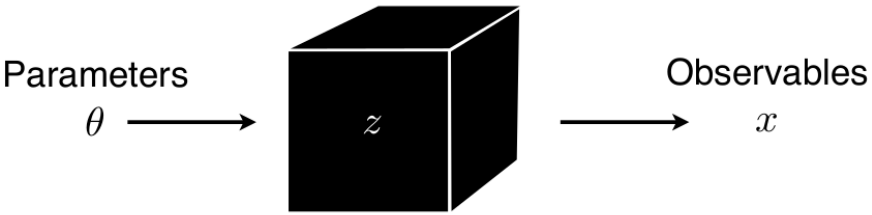


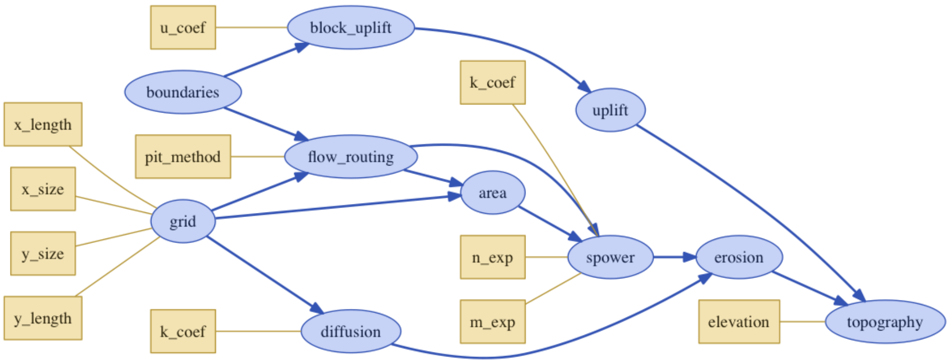


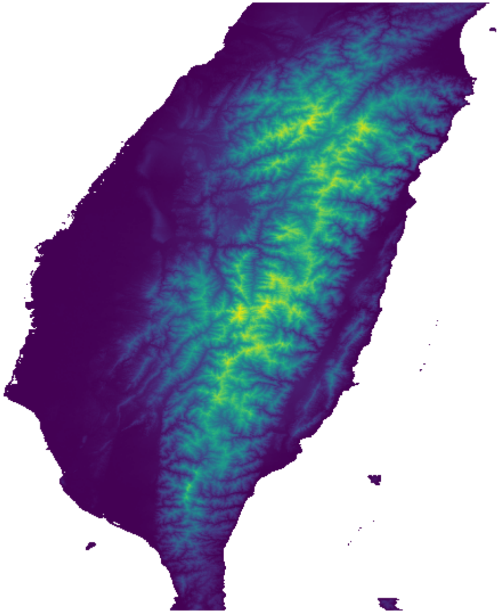


Refs: Planck Collaboration, 2015 (arXiv:1502.01589); Vogelsberger et al, 2014 (arXiv:1405.2921)

Computational topography

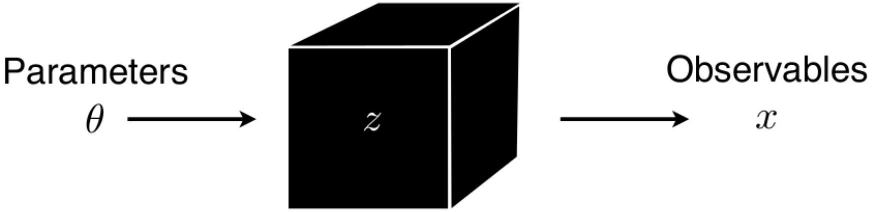






Refs: Benoit Bovy (xarray-simlab)

Climatology



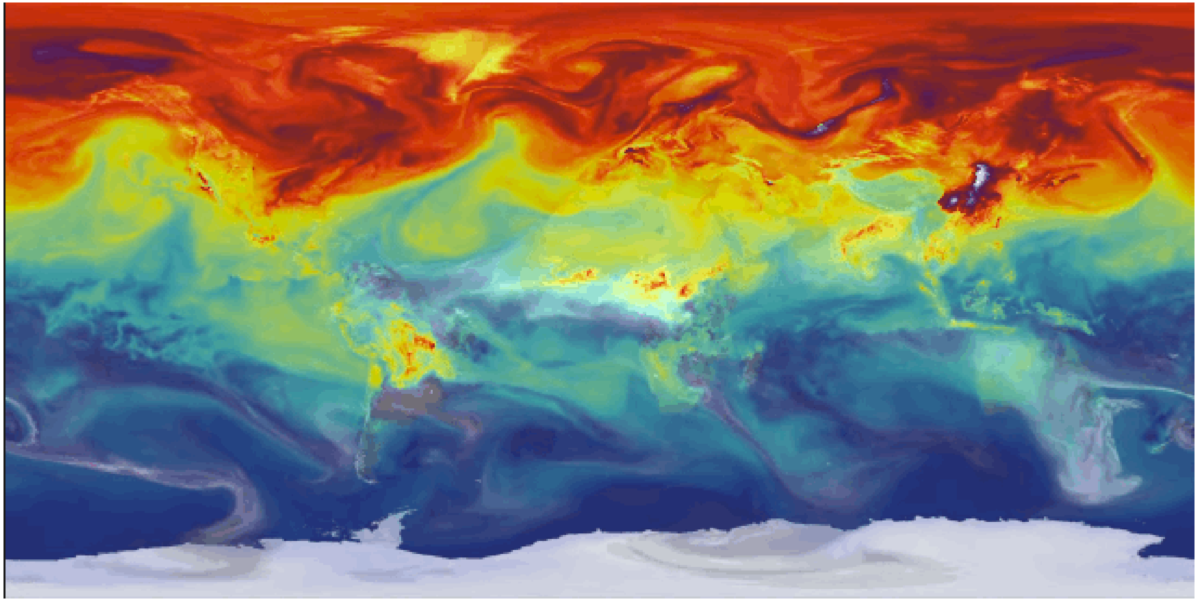
Parameters

θ

z

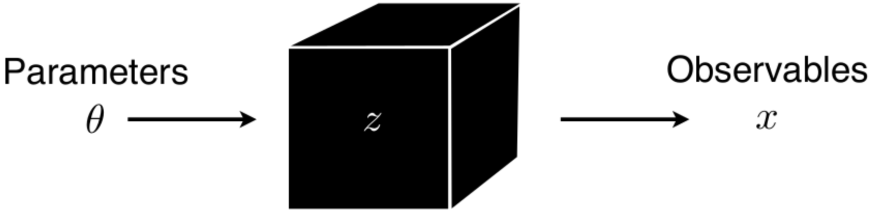
Observables

x



Refs: NASA's Goddard Space Flight Center / B. Putman, 2014 (press release)

Epidemiology



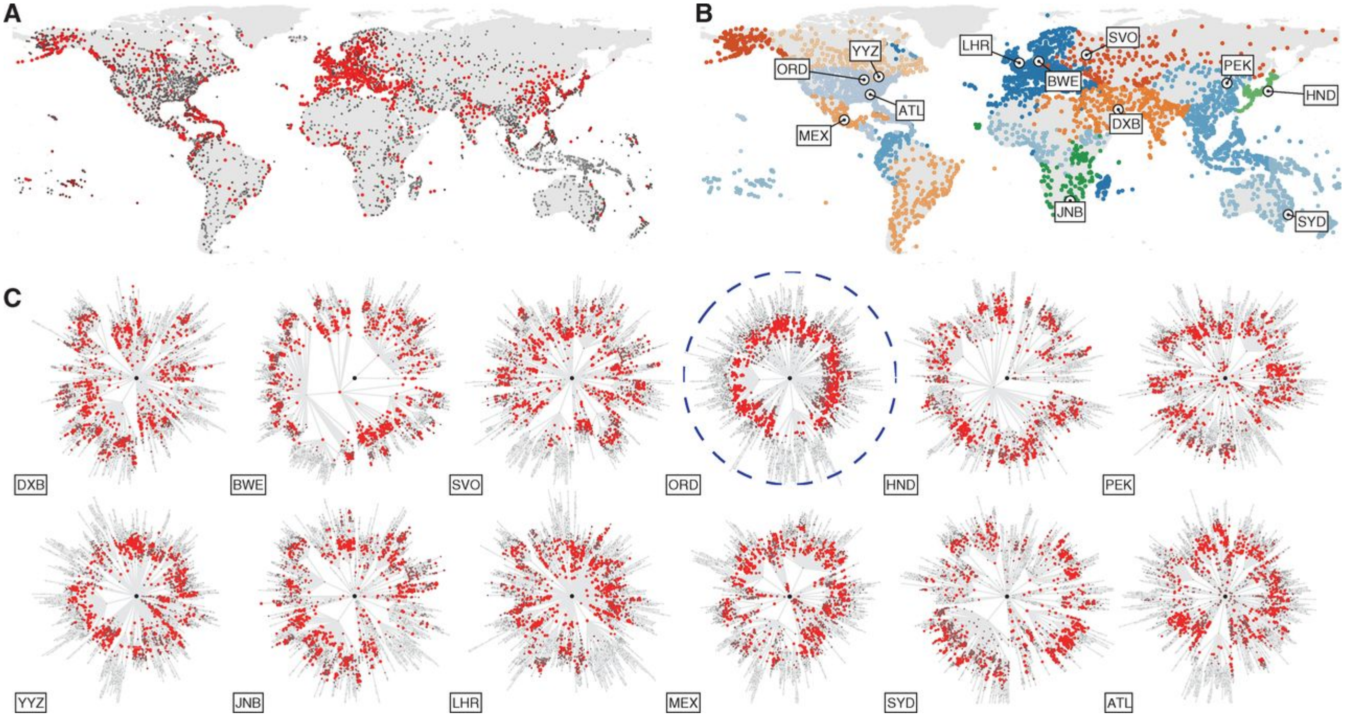
Parameters

θ

z

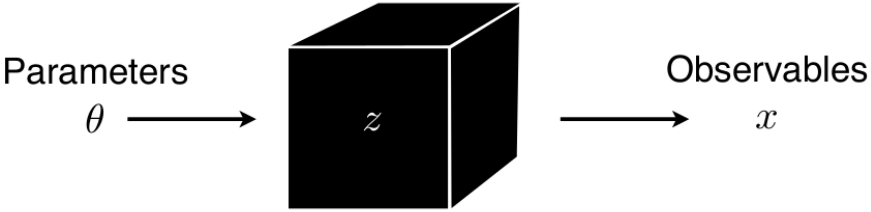
Observables

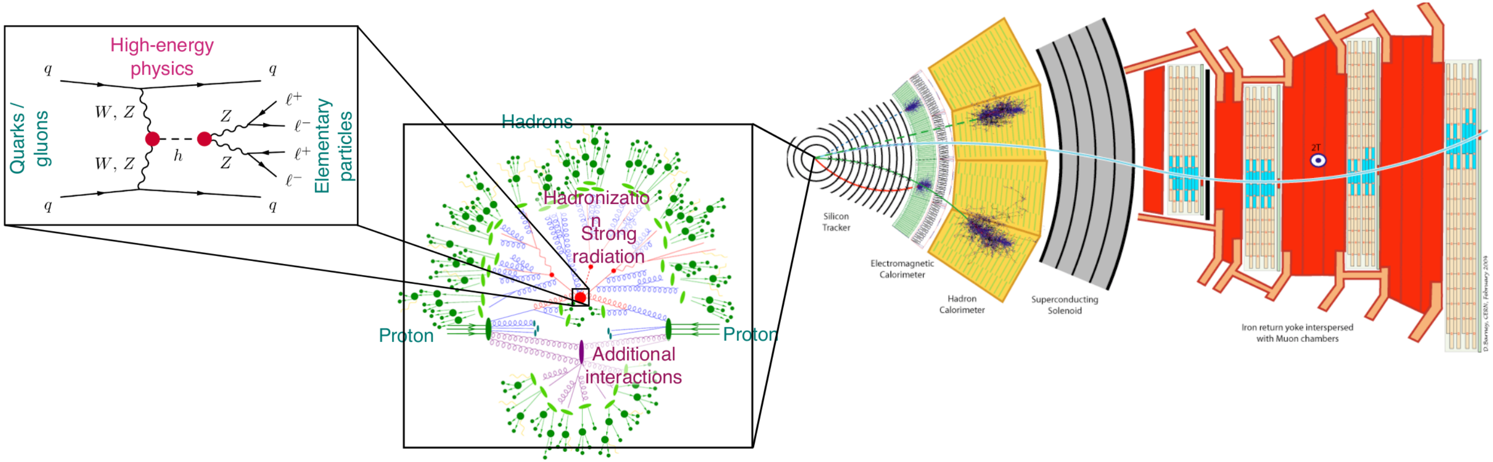
x



Refs: Brockmann and Helbing, 2013 (doi:10.1126/science.1245200)

Particle physics

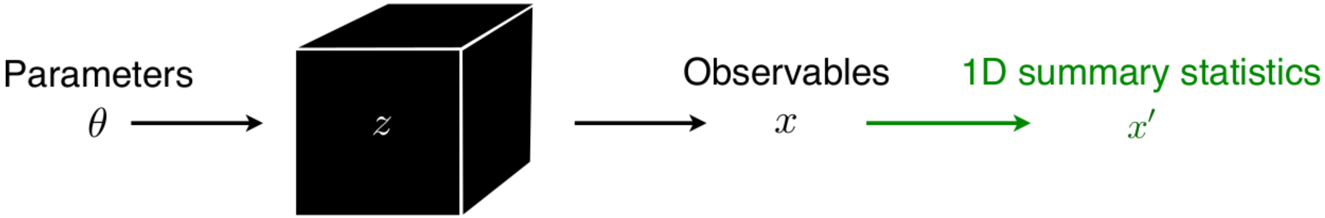




The Galton board of particle physics

Likelihood-free inference

The physicist's way



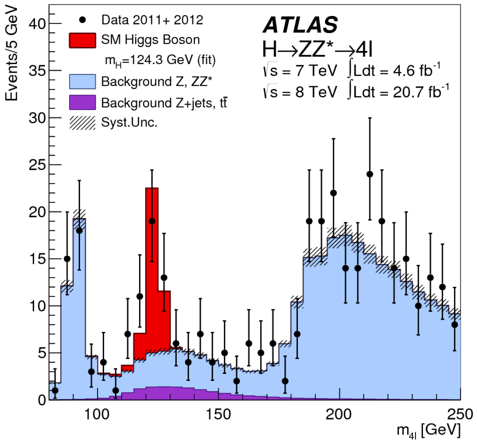
Parameters θ → z → Observables x → 1D summary statistics x'

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

Then, approximate the likelihood $p(x|\theta)$ as

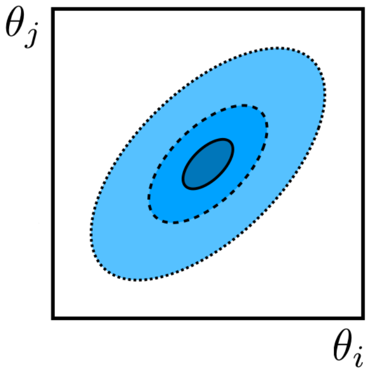
$$p(x|\theta) \approx \hat{p}(x|\theta) = p(x'|\theta),$$

where $p(x'|\theta)$ can be estimated by running the simulator for different parameter values θ and filling histograms.



Hypothesis testing

We are not only interested in $\hat{\theta}$, we also want to reject all those hypotheses that do not fit the observations with high probability.



According to the Neyman-Pearson lemma, the likelihood ratio

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

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

In the likelihood-free setup, the ratio is difficult to compute. However, using the approximate likelihood we can define

$$\frac{p(x|\theta_0)}{p(x|\theta_1)} \approx \frac{\hat{p}(x|\theta_0)}{\hat{p}(x|\theta_1)}$$

When testing a null θ_0 against a set of alternatives Θ (e.g., background only vs. background + signal), the generalized likelihood ratio is defined as

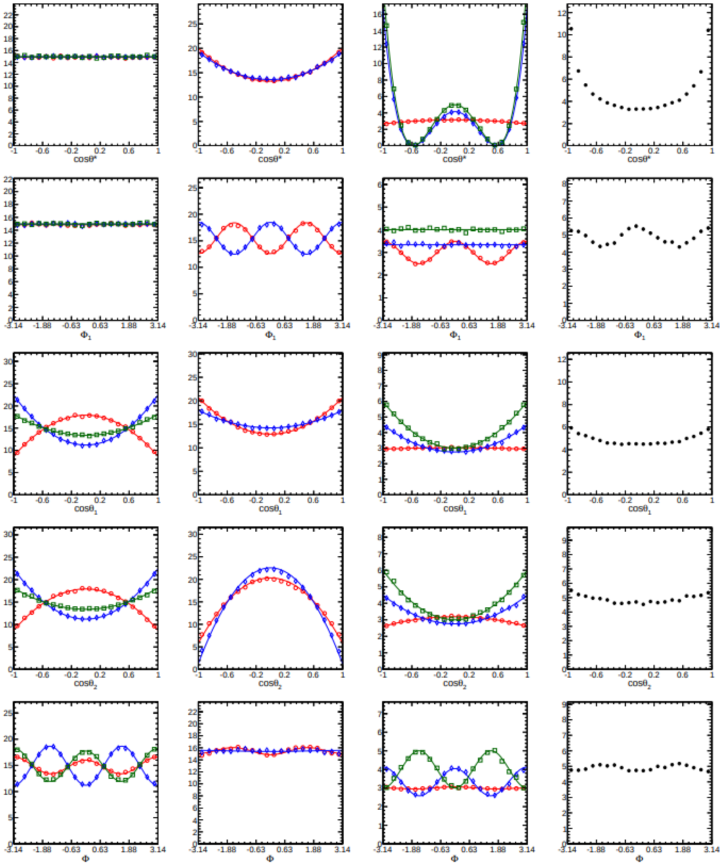
$$\begin{aligned} r(x|\theta_0, \Theta) &= \frac{p(x|\theta_0)}{\sup_{\theta \in \Theta} p(x|\theta)} \\ &= \frac{p(x|\theta_0)}{p(x|\hat{\theta})} \\ &\approx \frac{\hat{p}(x|\theta_0)}{\hat{p}(x|\hat{\theta})} \end{aligned}$$

where the MLE $\hat{\theta}$ can be approximated by scanning over $\hat{p}(x|\theta)$.

This methodology has worked great for physicists for the last 20-30 years, but ...

- Choosing the projection s is difficult and problem-dependent.
- Often there is no single good variable: compressing to any x' loses information.
- Ideally: analyse high-dimensional x', including all correlations.

Unfortunately, because of the curse of dimensionality, filling high-dimensional histograms is not tractable.



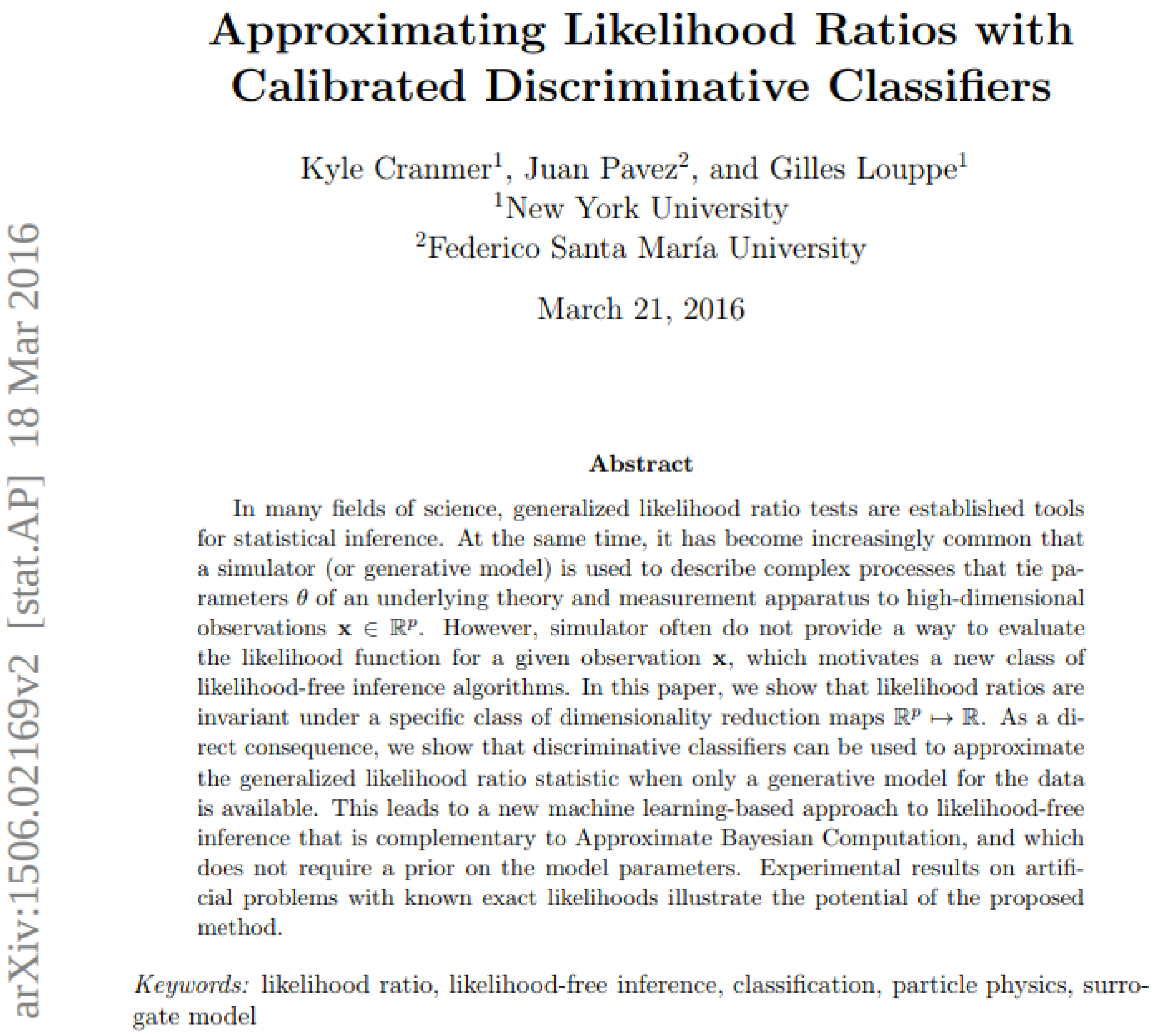
Who you gonna call? Machine learning!

Refs: Bolognesi et al, 2012 (arXiv:1208.4018)

(Some) established solutions

- Histograms of observables
    - Summary statistics
- Approximate Bayesian Computation
    - Summary statistics.
- Calibrated classifiers (CARL)
    - Optimal summary statistics.
- Neural density estimation
    - Density networks, autoregressive models, normalizing flows, etc.
- Matrix Element Method
    - Neglect or approximate shower+detector, explicitly calculate integral

$$\hat{p}(x|\theta) = \int p(z_p|\theta)\tilde{p}(x|z_p)dz_p$$



Approximating Likelihood Ratios with Calibrated Discriminative Classifiers

Kyle Cranmer[1], Juan Pavez[2], and Gilles Louppe[1]
[1]New York University
[2]Federico Santa María University

March 21, 2016

Abstract

In many fields of science, generalized likelihood ratio tests are established tools for statistical inference. At the same time, it has become increasingly common that a simulator (or generative model) is used to describe complex processes that tie parameters θ of an underlying theory and measurement apparatus to high-dimensional observations $\mathbf{x} \in \mathbb{R}^p$. However, simulator often do not provide a way to evaluate the likelihood function for a given observation $\mathbf{x}$, which motivates a new class of likelihood-free inference algorithms. In this paper, we show that likelihood ratios are invariant under a specific class of dimensionality reduction maps $\mathbb{R}^p \mapsto \mathbb{R}$. As a direct consequence, we show that discriminative classifiers can be used to approximate the generalized likelihood ratio statistic when only a generative model for the data is available. This leads to a new machine learning-based approach to likelihood-free inference that is complementary to Approximate Bayesian Computation, and which does not require a prior on the model parameters. Experimental results on artificial problems with known exact likelihoods illustrate the potential of the proposed method.

Keywords: likelihood ratio, likelihood-free inference, classification, particle physics, surrogate model

Refs: Cranmer et al, 2016 (arXiv:1506.02169)

Key insights:

- The likelihood ratio is sufficient for maximum likelihood estimation.
- Evaluating the likelihood ratio does not require evaluating the individual likelihoods.

The likelihood ratio is sufficient for maximum likelihood estimation:

$$\begin{aligned}
\hat{\theta} &= \arg\max_{\theta} p(\mathbf{d}|\theta) \\
&= \arg\max_{\theta} \frac{p(\mathbf{d}|\theta)}{\text{constant}} \\
&= \arg\max_{\theta} \frac{p(\mathbf{d}|\theta)}{p(\mathbf{d}|\theta_{\text{ref}})} \\
&= \arg\max_{\theta} \prod_{x_i \in \mathbf{d}} \frac{p(x_i|\theta)}{p(x_i|\theta_{\text{ref}})} \\
&= \arg\max_{\theta} \prod_{x_i \in \mathbf{d}} r(x_i|\theta, \theta_{\text{ref}})
\end{aligned}$$

$$\begin{tikzcd} \frac{p_{\mathbf{x}}(\mathbf{x}|\theta_0)}{p_{\mathbf{x}}(\mathbf{x}|\theta_1)} \arrow[rr, bend left] & = & r(\mathbf{x};\theta_0,\theta_1) \arrow[ll, bend left, "/" marking] \end{tikzcd}$$

Evaluating the likelihood ratio does not require evaluating the individual likelihoods:

- From $p(x|\theta_0)$ and $p(x|\theta_1)$ we can evaluate $r(x|\theta_0,\theta_1)$.
- However, from $r(x|\theta_0,\theta_1)$ the individual likelihoods $p(x|\theta_0)$ and $p(x|\theta_1)$ cannot be reconstructed.

Therefore, MLE inference and likelihood ratio estimation are strictly simpler problems than density estimation.

Carl

Theorem. The likelihood ratio is invariant under the change of variable $U = s(X)$, provided $s(x)$ is monotonic with $r(x)$.

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

- Note that the equality is strict.
- No information relevant for determining the ratio is lost.
- Although information about x may be lost through s.

Supervised learning provides a way to automatically construct s:

- A binary classifier $\hat{s}$ (e.g., a neural network) trained to distinguish $x \sim p(x|\theta_0)$ from $x \sim p(x|\theta_1)$ approximates the optimal classifier

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

  which is monotonic with r.

- Therefore, when $\hat{s} = s^*$,

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

That is, supervised classification is equivalent to likelihood ratio estimation and can therefore be used for MLE inference.

In practice, $\hat{s} \neq s^*$ because of approximation, estimation or optimization errors.

- Still, the result states that calibrating $\hat{s}(x)$ to build $p(\hat{s}(x)|\theta)$ is sufficient for recovering the true likelihood ratio $r(x|\theta_0, \theta_1)$, provided $\hat{s}(x)$ is monotonic with $r(x|\theta_0, \theta_1)$.
- This step can be carried with 1D density estimation or calibration algorithms (histograms, KDE, isotonic regression, etc).
- If not monotonic with r, then the resulting statistic is strictly less powerful than the true ratio.

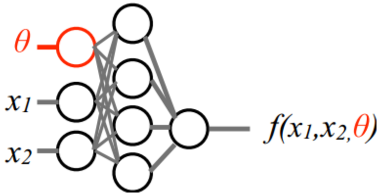
Procedure

For inference, we have

$$\begin{aligned}\hat{\theta} &= \arg\max_{\theta} \prod_{x_i \in \mathbf{d}} r(x_i|\theta, \theta_{\text{ref}}) \\ &= \arg\max_{\theta} \prod_{x_i \in \mathbf{d}} \frac{p(\hat{s}(x|\theta, \theta_{\text{ref}})|\theta)}{p(\hat{s}(x|\theta, \theta_{\text{ref}})|\theta_{\text{ref}})}\end{aligned}$$

where $\hat{s}(x|\theta, \theta_{\text{ref}})$ denotes a classifier trained to distinguish between θ and θ_{ref}.

- Point by point optimization: Keep θ_{ref} fixed, scan for θ, train a new classifier $\hat{s}$ for each θ and evaluate the ratio.
- Parameterized classifier: Train a single classifier $\hat{s}$ taking both x and θ as inputs, scan for θ and evaluate the ratio.



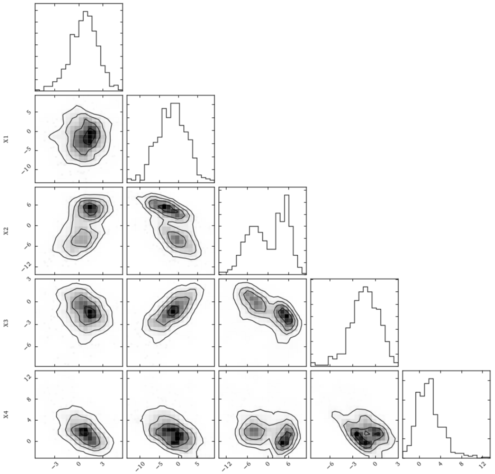
Refs: Baldi et al, 2016 (arXiv:1601.07913)

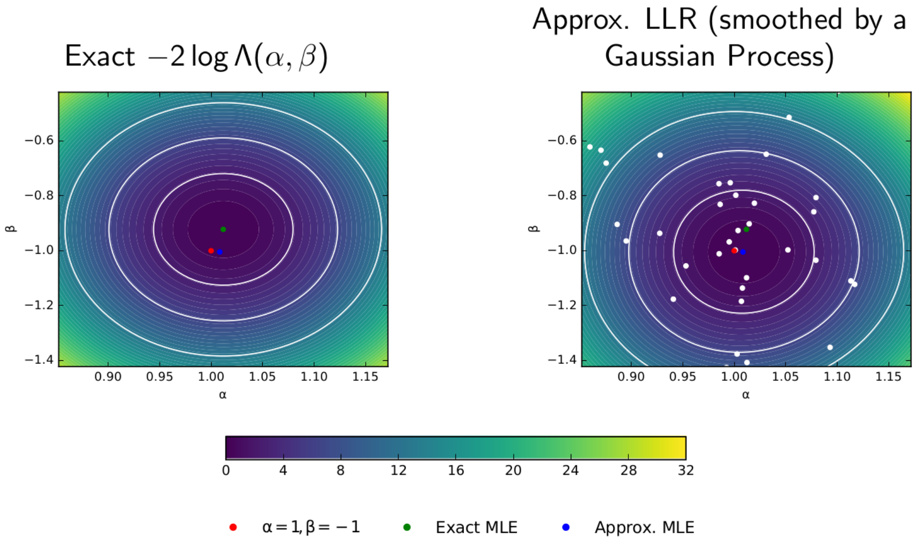
For composite hypothesis testing, the previous procedure can be used to find $\hat{\theta}$.

Then a classifier s between θ_0 and $\hat{\theta}$ is built, from which is derived the generalized likelihood ratio statistic.

Toy example

Simulator generating 5D observables x, with parameters of interest α and β. Given observed data $\mathbf{d}$, we want to find $\hat{\alpha}$ and $\hat{\beta}$ along with its σ-contours.

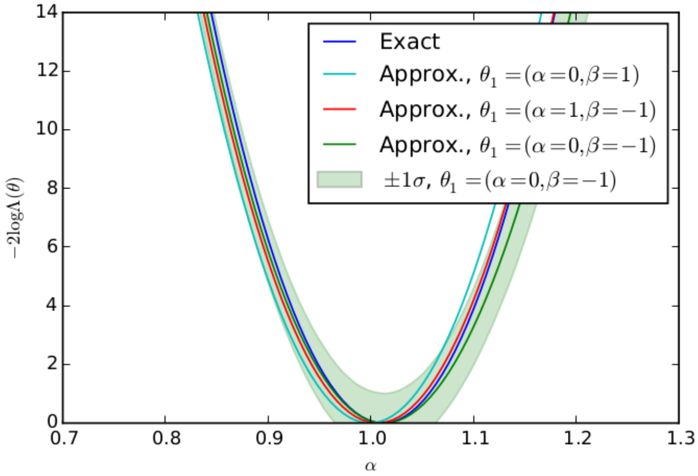


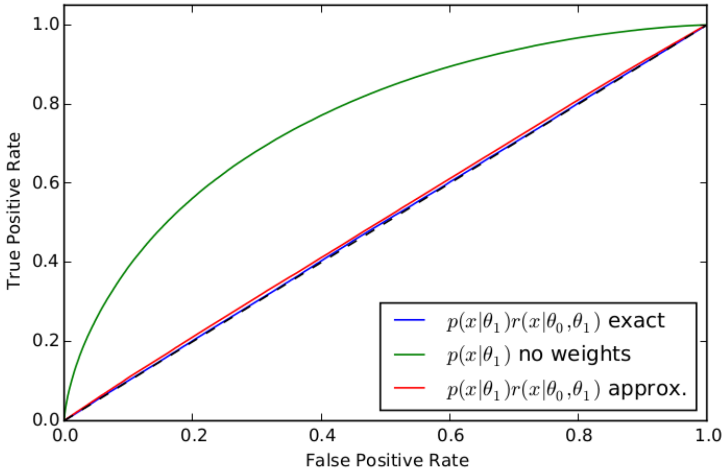

Exact $-2\log\Lambda(\alpha,\beta)$

Approx. LLR (smoothed by a Gaussian Process)

$$-2\log\Lambda(\alpha,\beta) = -2\log\frac{p(\mathbf{d}|\alpha,\beta)}{p(\mathbf{d}|\hat{\alpha},\hat{\beta})}$$

Diagnostics



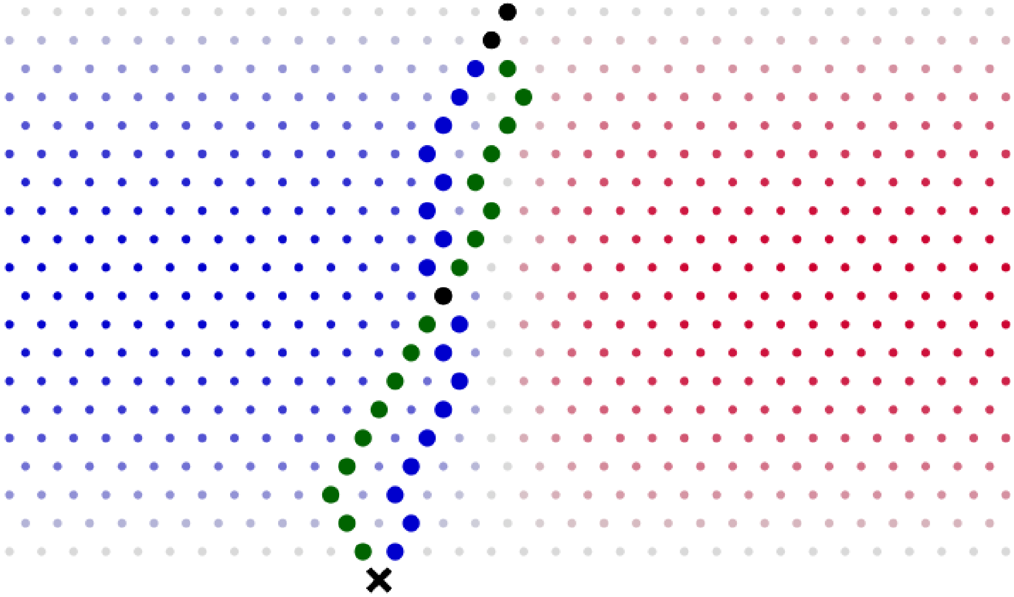


We need procedures to assess the quality of the approximated ratio $\hat{r}$:

- For inference, the value of the MLE $\hat{\theta}$ should be independent of the value of θ_{ref} used in the denominator of the ratio.
- Train a classifier to distinguish between unweighted samples from $p(x|\theta_0)$ and samples from $p(x|\theta_1)$ weighted by $\hat{r}(x|\theta_0, \theta_1)$.

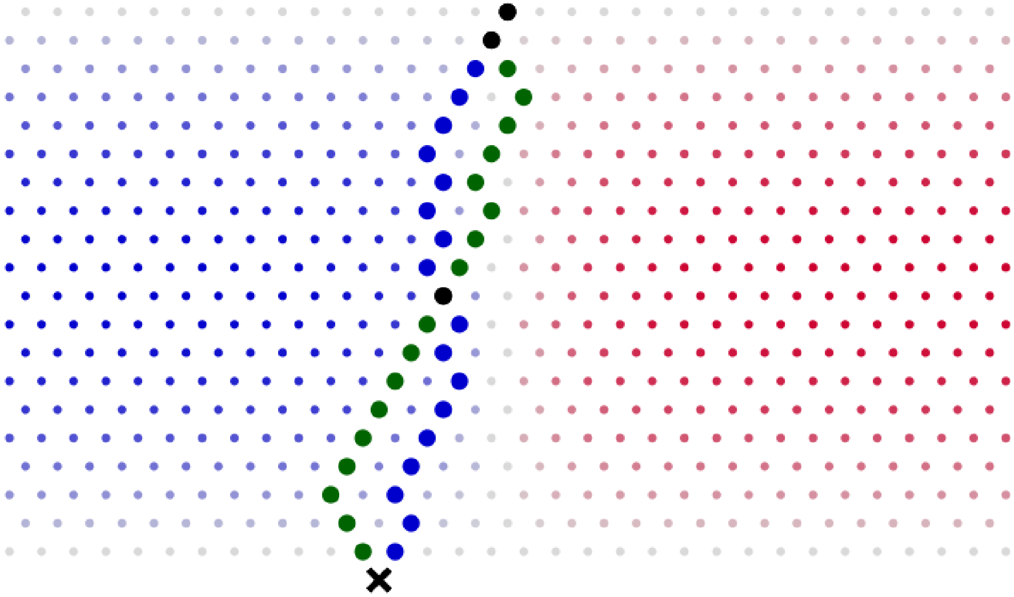
Mining gold from simulators

Refs: Brehmer et al, 2018 (arXiv:1805.12244)



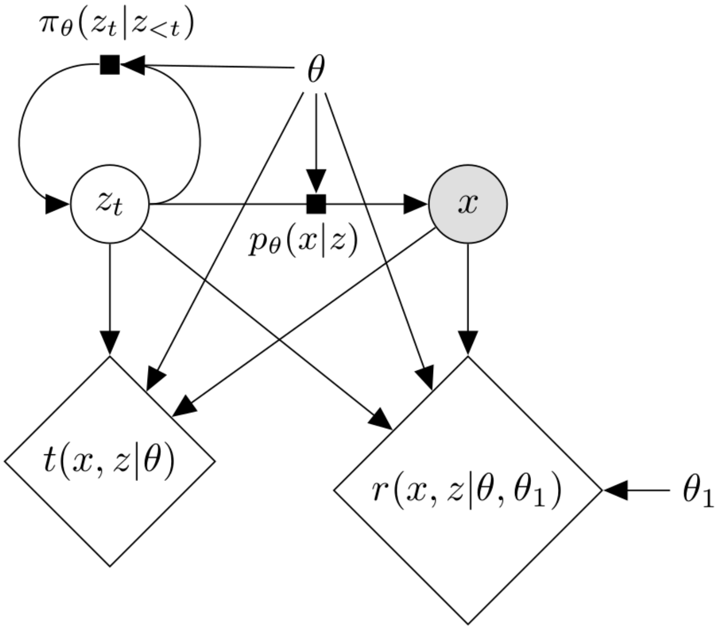
$p(x|\theta)$ is usually intractable.

What about $p(x, z|\theta)$?



$$\begin{aligned} p(x, z|\theta) &= p(z_1|\theta)p(z_2|z_1, \theta) \ldots p(z_T|z_{<T}, \theta)p(x|z_{\le T}, \theta) \\ &= p(z_1|\theta)p(z_2|\theta) \ldots p(z_T|\theta)p(x|z_T) \\ &= p(x|z_T) \prod_t \theta^{z_t}(1-\theta)^{1-z_t} \end{aligned}$$

This can be computed as the ball falls down the board!



The simulator can be viewed as a graphical model that abstracts the simulation as a probabilistic sequence of latent states z_t.

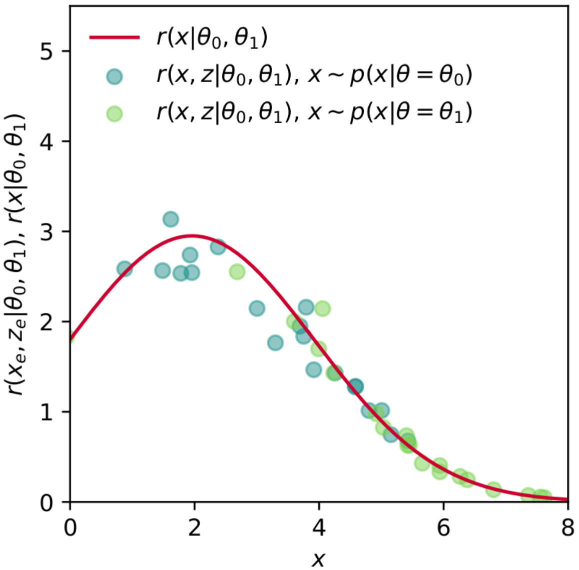
- The simulator implements a probabilistic transition $\pi_\theta(z_t|z_{<t})$.
- The simulator emits an observation x based on $p(x|z,\theta)$.

Mining gold

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

- to calculate the joint likelihood $p(x, z|\theta)$ as $\pi_\theta(z_1)\pi_\theta(z_2|z_1)\ldots\pi_\theta(z_T|z_{<T})p(x|z,\theta)$;
- 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!



Observe that the joint likelihood ratios

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

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:

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

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

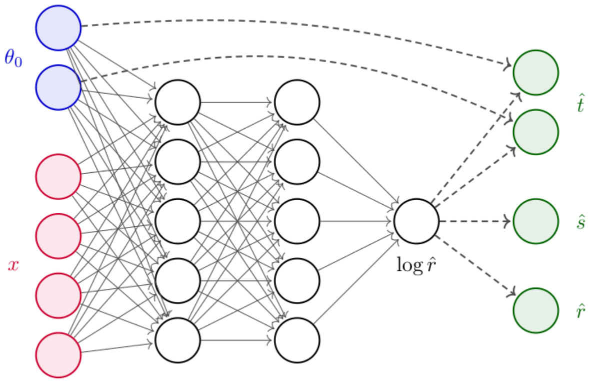
$$\begin{aligned} g^*(x) &= \frac{1}{p(x|\theta)} \int p(x, z|\theta) g(x, z) dz \\ &= \mathbb{E}_{p(z|x,\theta)} \left[g(x, z) \right] \end{aligned}$$

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

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

which is minimized by

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

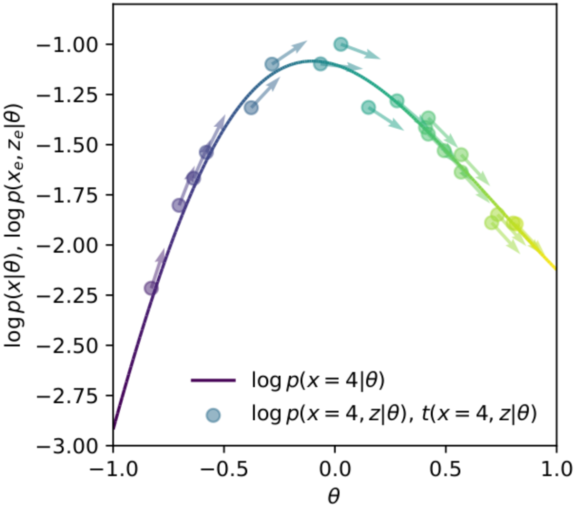


How does one find r^*?

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

Minimizing functionals is exactly what machine learning does. In our case,

- $\hat{r}$ are neural networks (or the parameters thereof);
- L_r is the loss function;
- minimization is carried out using stochastic gradient descent from the data extracted from the simulator.



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

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

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

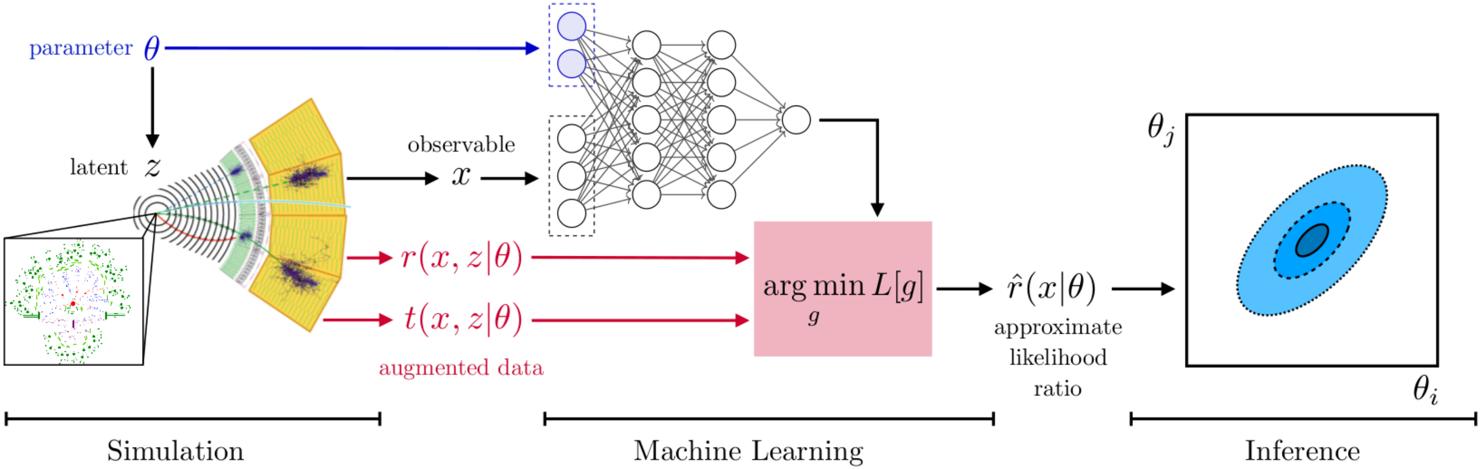
$$L_t = \mathbb{E}_{p(x,z|\theta_0)} \left[(t(x, z|\theta_0) - \hat{t}\,(x))^2\right],$$

which is minimized by

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

Rascal

$$L = L_r + L_t$$



Family of likelihood-free inference strategies

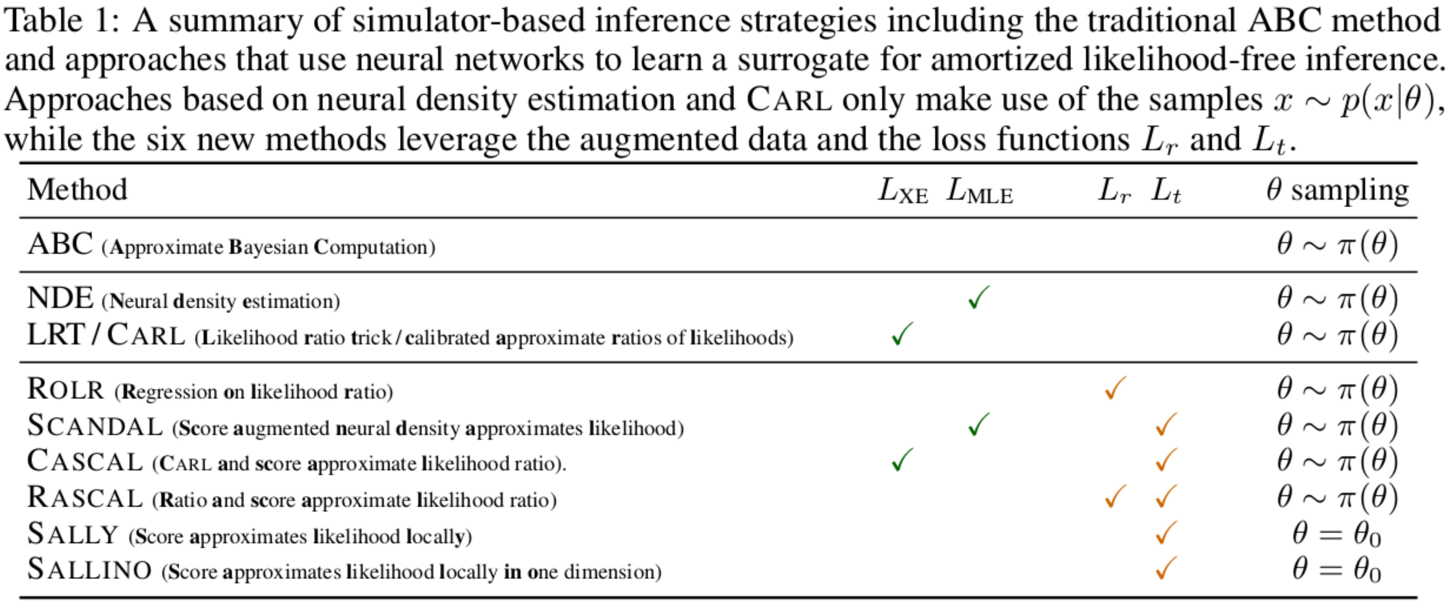
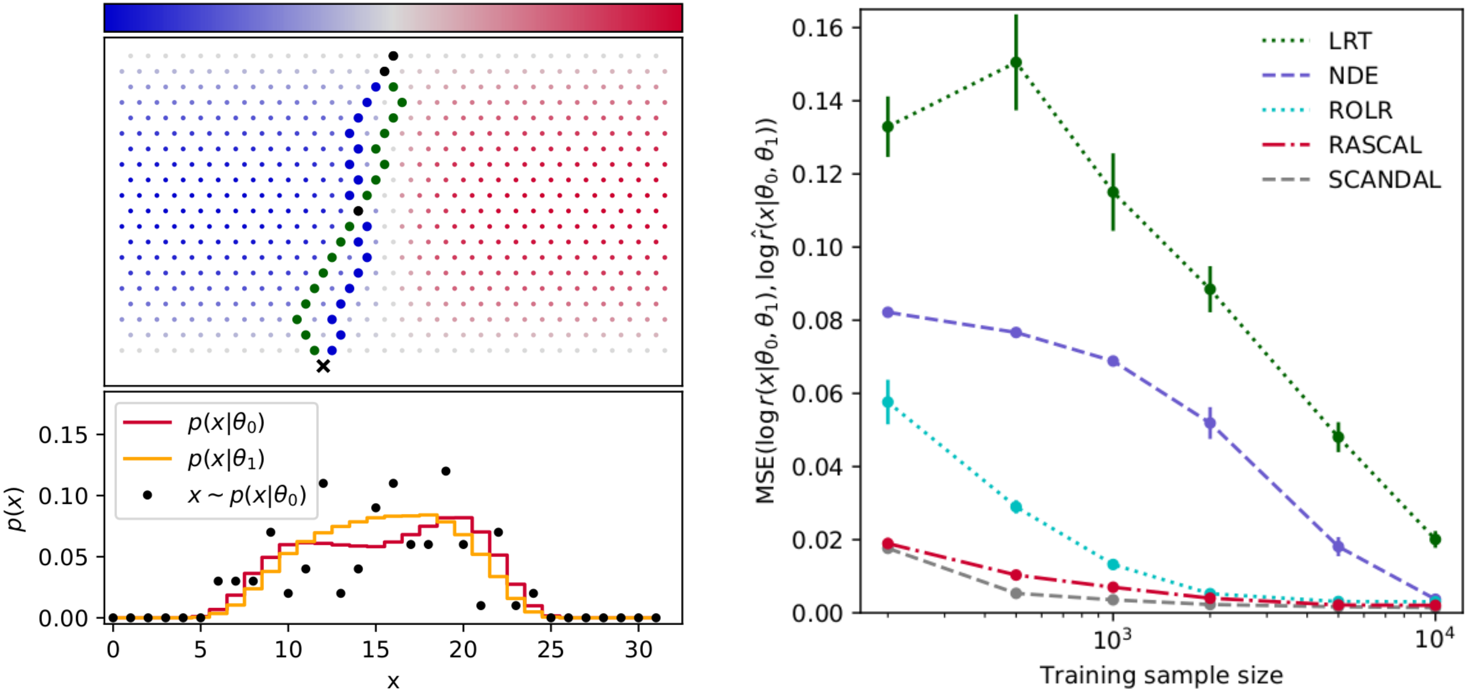


Table 1: A summary of simulator-based inference strategies including the traditional ABC method and approaches that use neural networks to learn a surrogate for amortized likelihood-free inference. Approaches based on neural density estimation and CARL only make use of the samples $x \sim p(x|\theta)$, while the six new methods leverage the augmented data and the loss functions L_r and L_t.

Method	L_{XE}	L_{MLE}	L_r	L_t	θ sampling
ABC (**A**pproximate **B**ayesian **C**omputation)					$\theta \sim \pi(\theta)$
NDE (**N**eural **d**ensity **e**stimation)		✓			$\theta \sim \pi(\theta)$
LRT / CARL (**L**ikelihood **r**atio **t**rick / **c**alibrated **a**pproximate **r**atios of **l**ikelihoods)	✓				$\theta \sim \pi(\theta)$
ROLR (**R**egression **o**n **l**ikelihood **r**atio)			✓		$\theta \sim \pi(\theta)$
SCANDAL (**Sc**ore **a**ugmented **n**eural **d**ensity **a**pproximates **l**ikelihood)		✓		✓	$\theta \sim \pi(\theta)$
CASCAL (**CA**RL and **sc**ore **a**pproximate **l**ikelihood ratio).	✓			✓	$\theta \sim \pi(\theta)$
RASCAL (**Ra**tio and **sc**ore **a**pproximate **l**ikelihood ratio)			✓	✓	$\theta \sim \pi(\theta)$
SALLY (**S**core **a**pproximates **l**ikelihood **l**ocall**y**)				✓	$\theta = \theta_0$
SALLINO (**S**core **a**pproximates **l**ikelihood **l**ocally **in** **o**ne dimension)				✓	$\theta = \theta_0$

Effective inference



Toy experiment on the Galton board.

Constraining Effective Field Theories, effectively

LHC processes

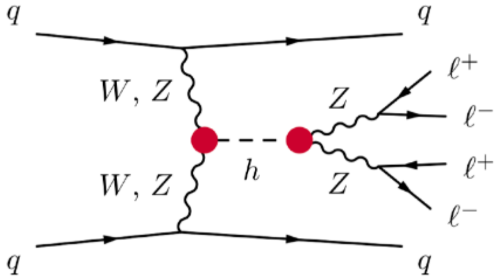
Latent variables

Parameters of interest

Parton-level momenta

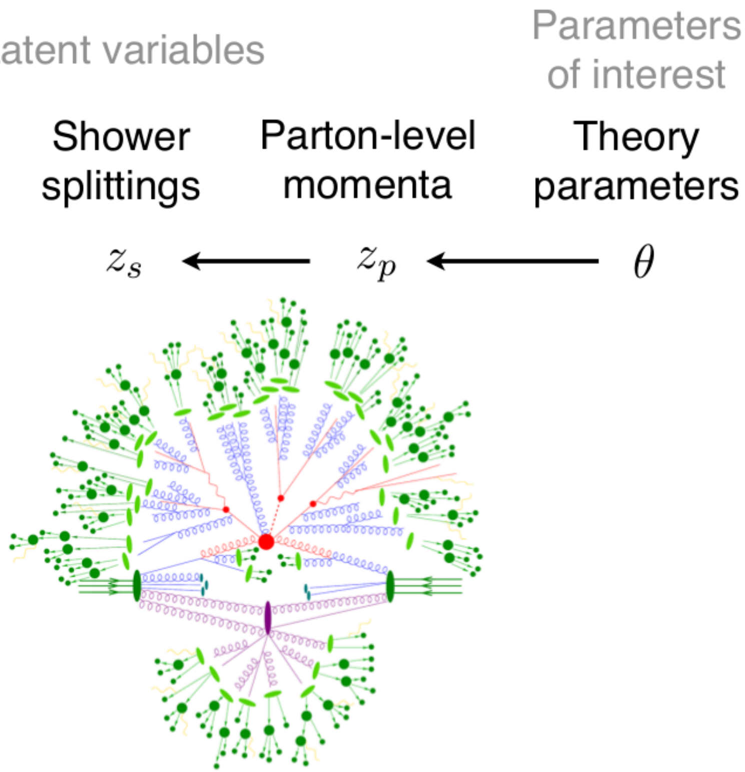
Theory parameters

$$z_p \longleftarrow \theta$$



Credits: Johann Brehmer

LHC processes



Latent variables

Parameters of interest

Shower splittings

Parton-level momenta

Theory parameters

$$z_s \longleftarrow z_p \longleftarrow \theta$$

Credits: Johann Brehmer

LHC processes

Latent variables

Parameters of interest

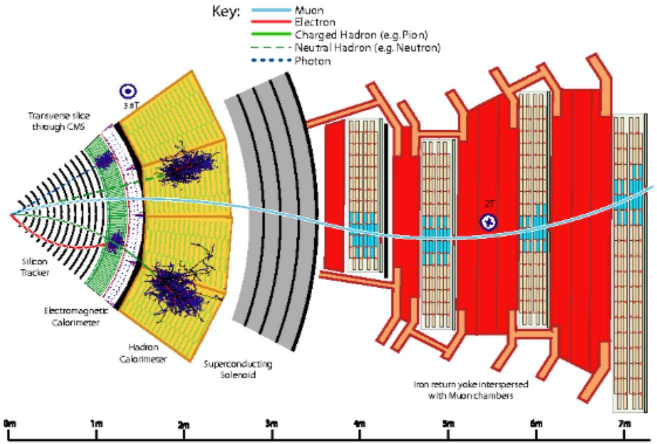
Detector interactions

Shower splittings

Parton-level momenta

Theory parameters

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


Key:
Muon
Electron
Charged Hadron (e.g. Pion)
Neutral Hadron (e.g. Neutron)
Photon
Transverse slice through CMS
3.8T
2T
Silicon Tracker
Electromagnetic Calorimeter
Hadron Calorimeter
Superconducting Solenoid
Iron return yoke interspersed with Muon chambers
0m
1m
2m
3m
4m
5m
6m
7m

Credits: Johann Brehmer

LHC processes

Features

Latent variables

Parameters of interest

Observables

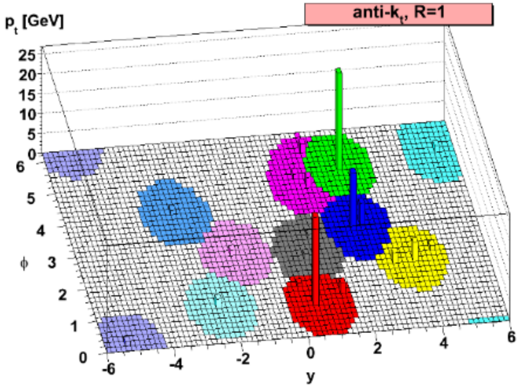
Detector interactions

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]

Credits: Johann Brehmer

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

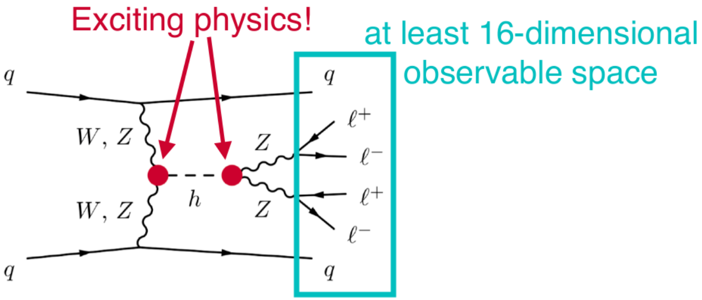
Key insights:

- The distribution of parton-level four-momenta
$$p(z_p|\theta) = \frac{1}{\sigma(\theta)} \frac{d\sigma(\theta)}{dz_p},$$
where $\sigma(\theta)$ and $\frac{d\sigma(\theta)}{dz_p}$ are the total and differential cross sections, is tractable.
- Downstream processes $p(z_s|z_p), p(z_d|z_s)$ and $p(x|z_d)$ do not depend on θ.

This implies that both $r(x, z|\theta_0, \theta_1)$ and $t(x, z|\theta_0)$ can be mined. E.g.,

$$\begin{aligned} r(x, z|\theta_0, \theta_1) &= \frac{p(z_p|\theta_0)}{p(z_p|\theta_1)} \frac{p(z_s|z_p)}{p(z_s|z_p)} \frac{p(z_d|z_s)}{p(z_d|z_s)} \frac{p(x|z_d)}{p(x|z_d)} \\ &= \frac{p(z_p|\theta_0)}{p(z_p|\theta_1)} \end{aligned}$$

Proof of concept


Exciting physics!

at least 16-dimensional observable space

- Context: Higgs production in weak boson fusion.
- Goal: constraints on two theory parameters.

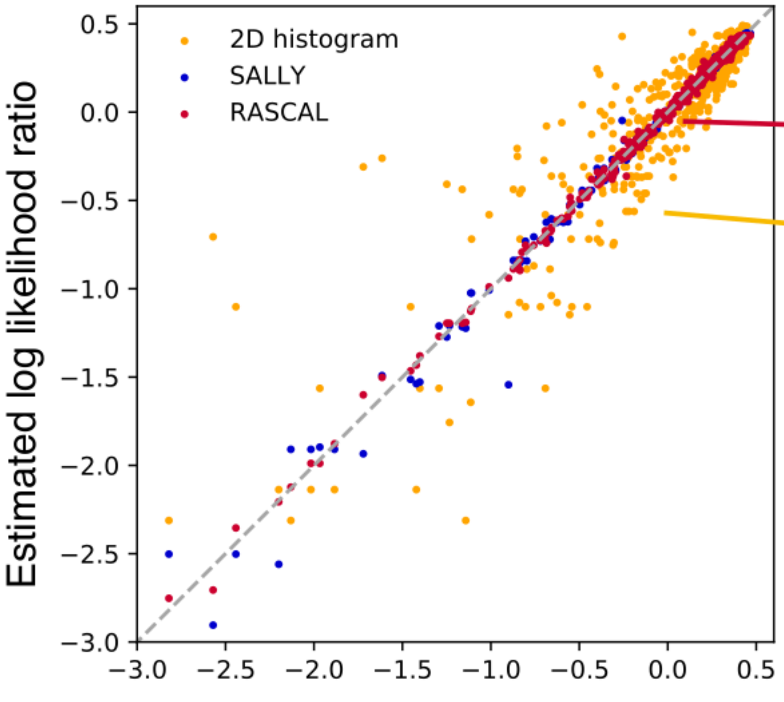
$$\mathcal{L} = \mathcal{L}_{SM} + \frac{f_W}{\underbrace{\Lambda^2}_{}} \frac{ig}{2} (D^\mu \phi)^\dagger \sigma^a D^\nu \phi \, W^a_{\mu\nu} - \frac{f_{WW}}{\underbrace{\Lambda^2}_{}} \frac{g^2}{4} (\phi^\dagger \phi) \, W^a_{\mu\nu} W^{\mu\nu\, a}$$

- Two setups:
  - Simplified setup in which we can compare to true likelihood.
  - Realistic simulation with approximate detector effects.

Credits: Johann Brehmer

Precise likelihood ratio estimates

Each point = 1 event

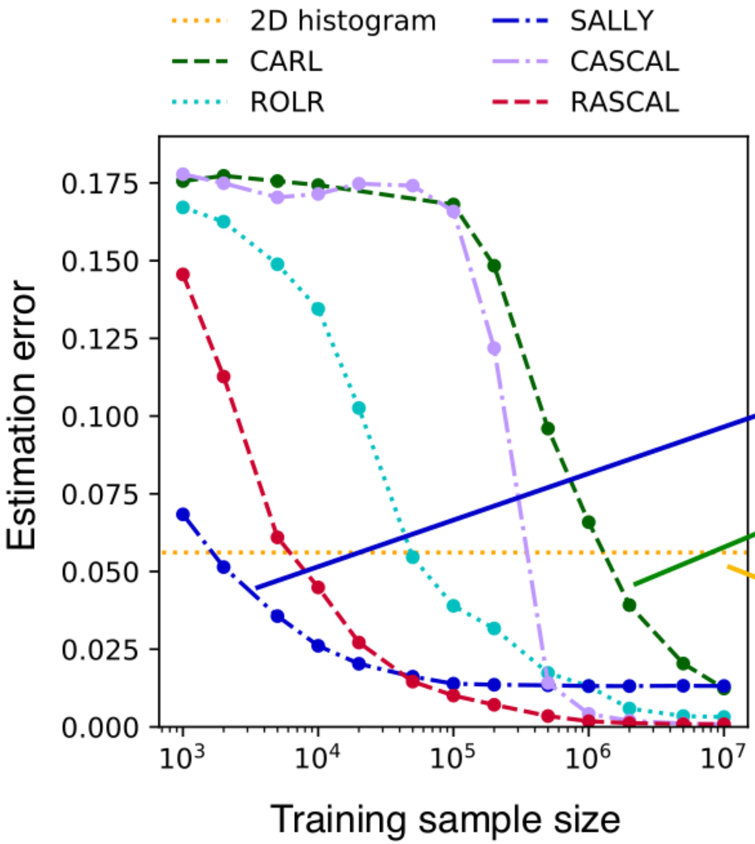


True log likelihood ratio
(usually don’t have this)

RASCAL
much more precise than
traditional histogram

Credits: Johann Brehmer

Increased data efficiency

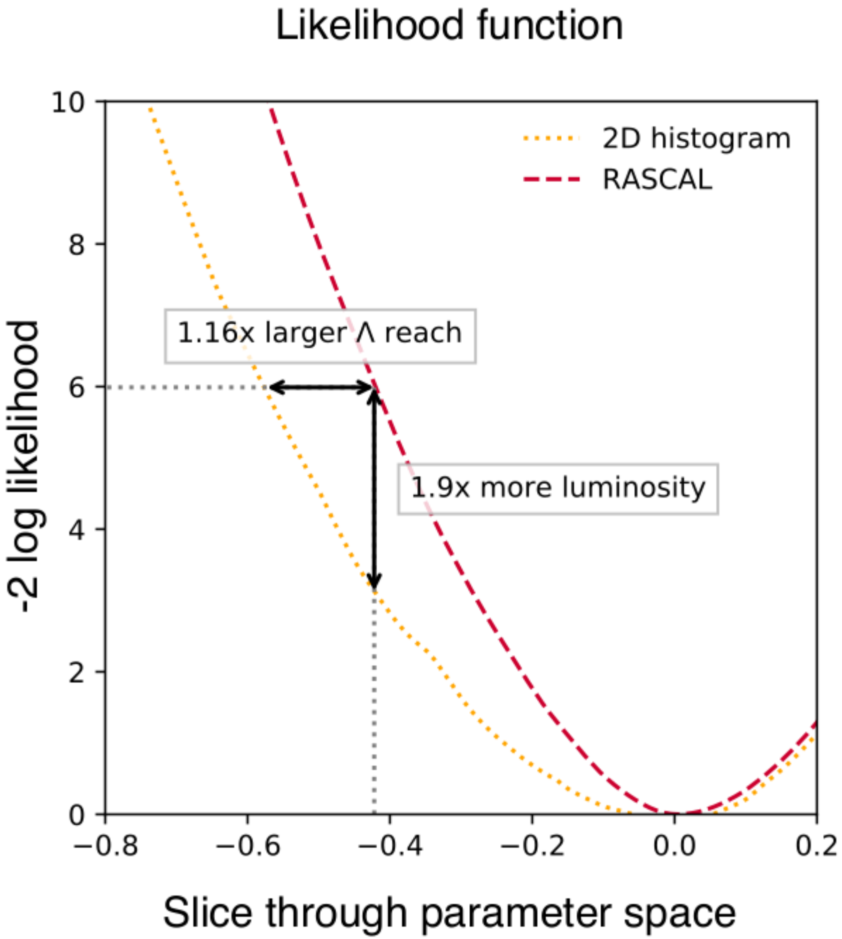


New techniques
require less data than
generic ML method
[CARL, see K. Cranmer, G. Louppe,
J. Pavez 1506.02169]

Traditional Histogram

Credits: Johann Brehmer

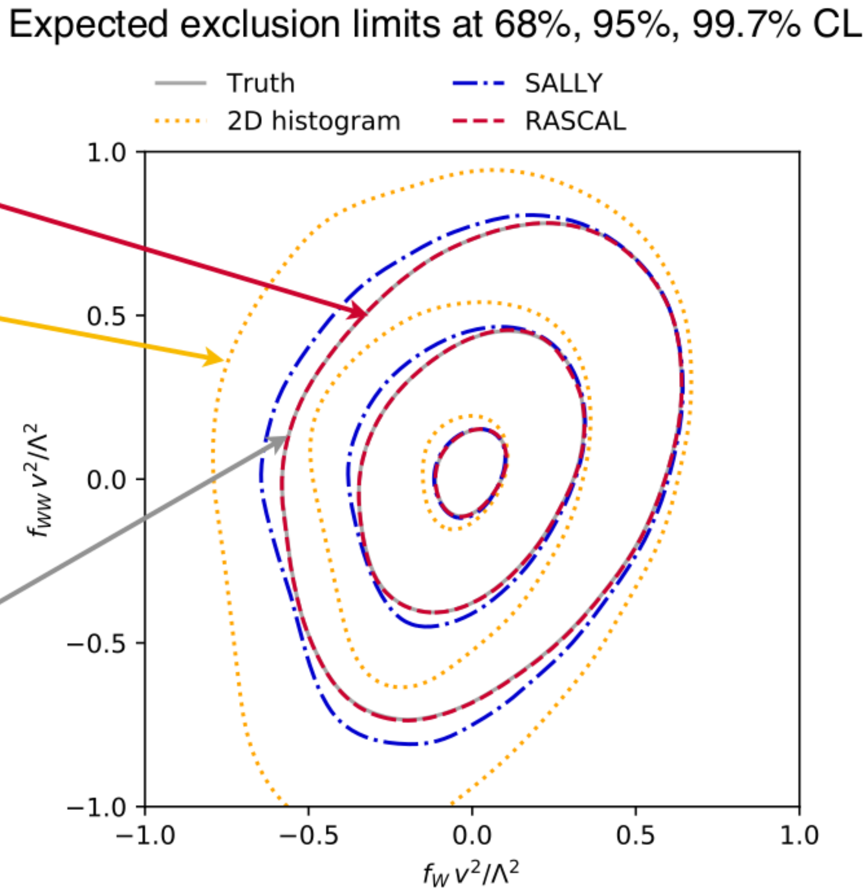
Better sensitivity



36 events, assuming SM

Credits: Johann Brehmer

Stronger bounds



RASCAL enables stronger limits than traditional histogram

Limits from RASCAL virtually indistinguishable from true likelihood (usually we don't have that)

36 events, assuming SM

Credits: Johann Brehmer

Summary

- Machine learning provides several solutions for simulation-based likelihood-free inference.
- CARL defines an optimal solution for likelihood ratio estimation, which is itself sufficient for inference.
- It is often possible to mine the joint likelihood, the joint likelihood ratio or the joint score, which enables effective likelihood-free inference.

Collaborators

References

- 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.
- Cranmer, K., Pavez, J., & Louppe, G. (2015). Approximating likelihood ratios with calibrated discriminative classifiers. arXiv preprint arXiv:1506.02169.

The end.