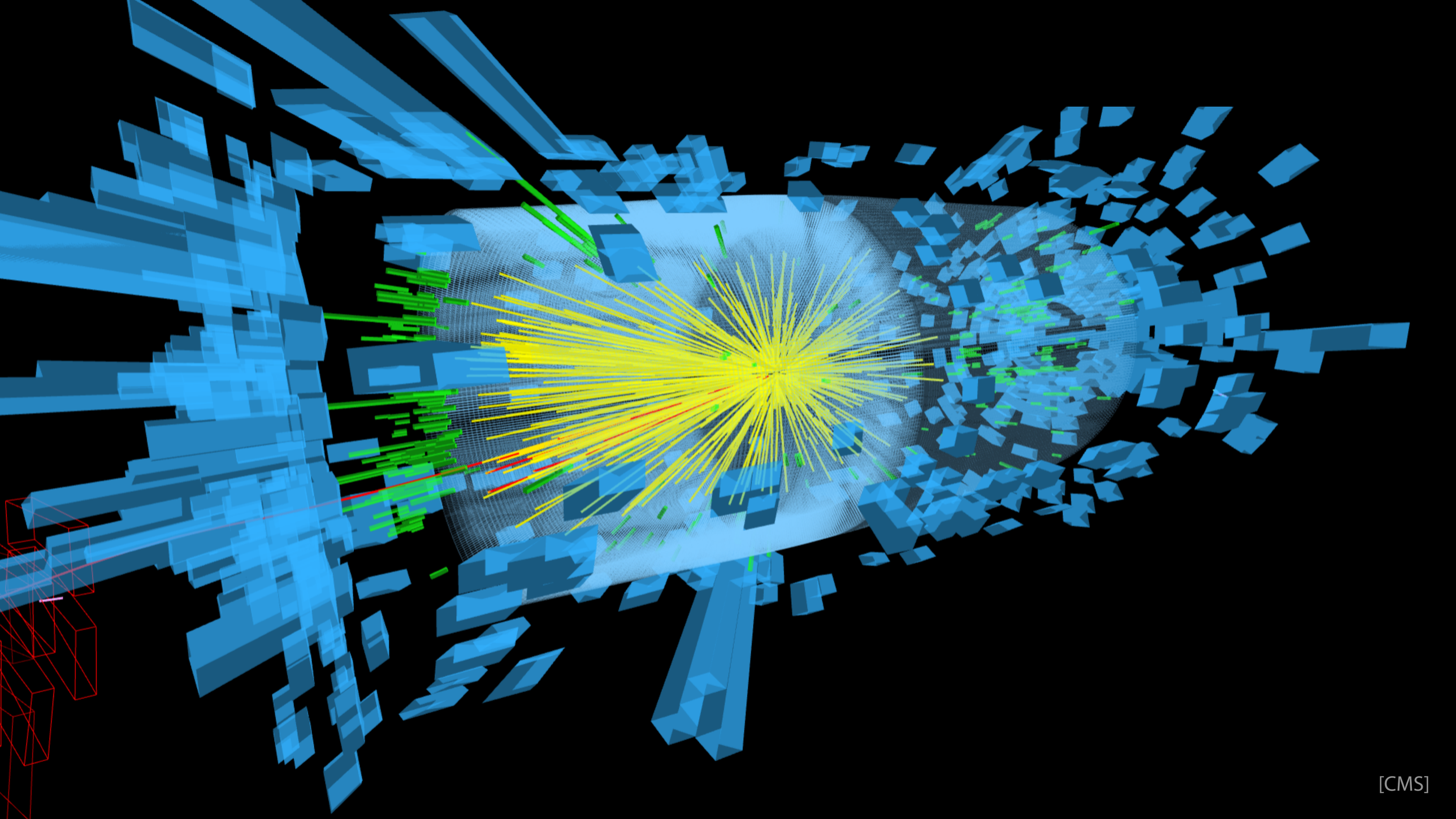
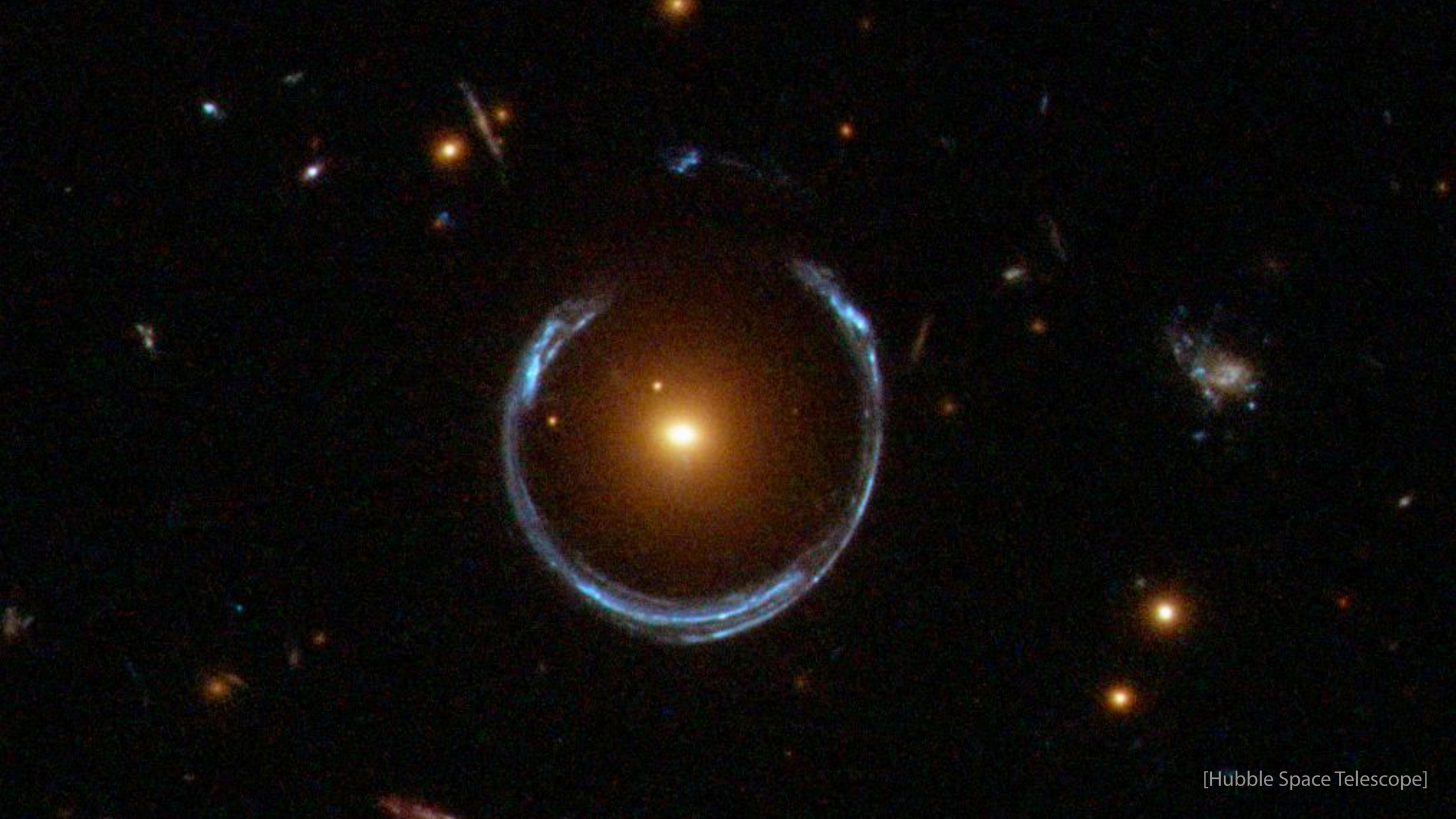
Simulation-based inference: The likelihood is dead, long live the likelihood

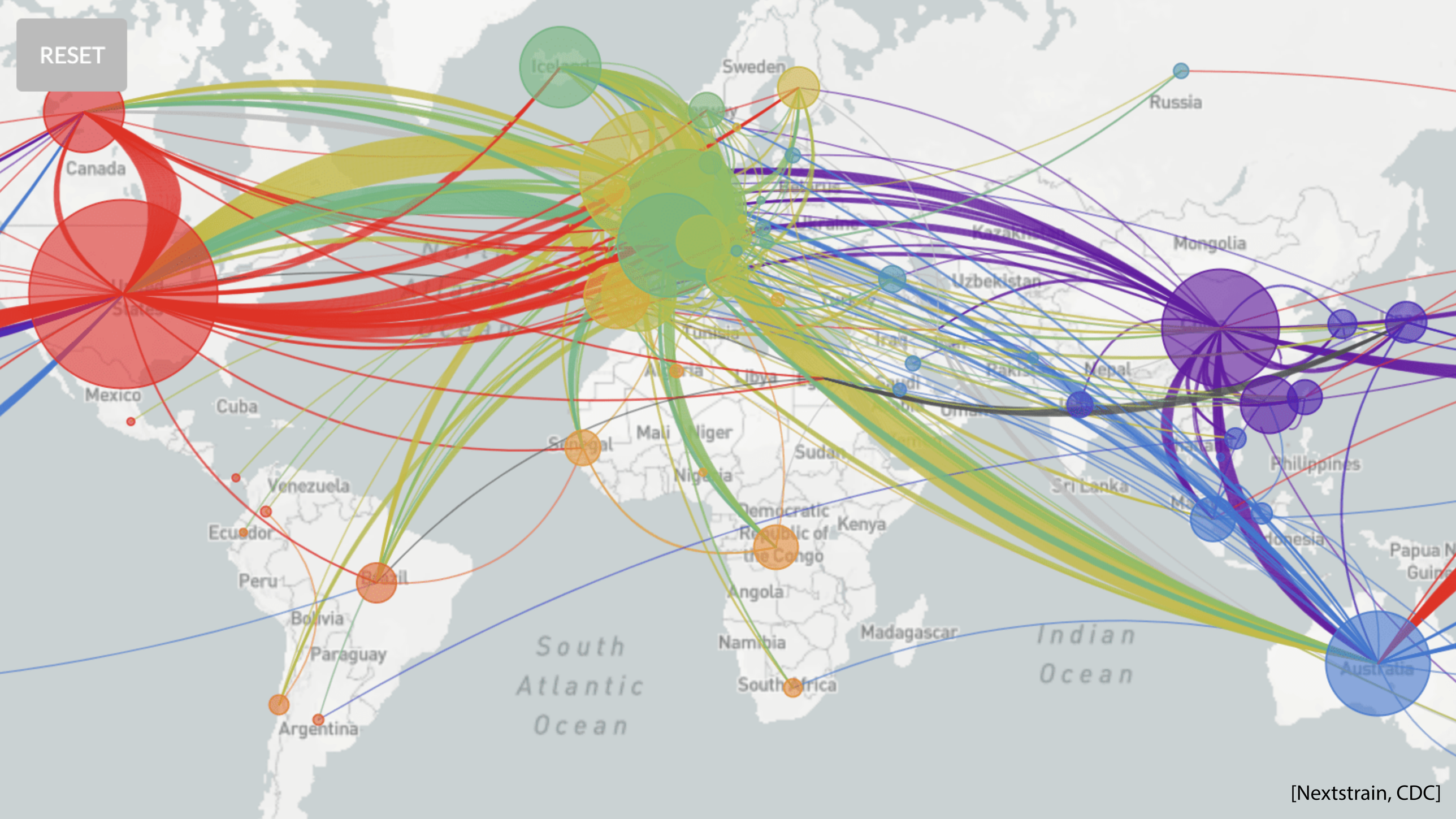
Johann Brehmer
johannbrehmer.de
@johannbrehmer

RODEM Sinergia seminar
July 6, 2022


[CMS]



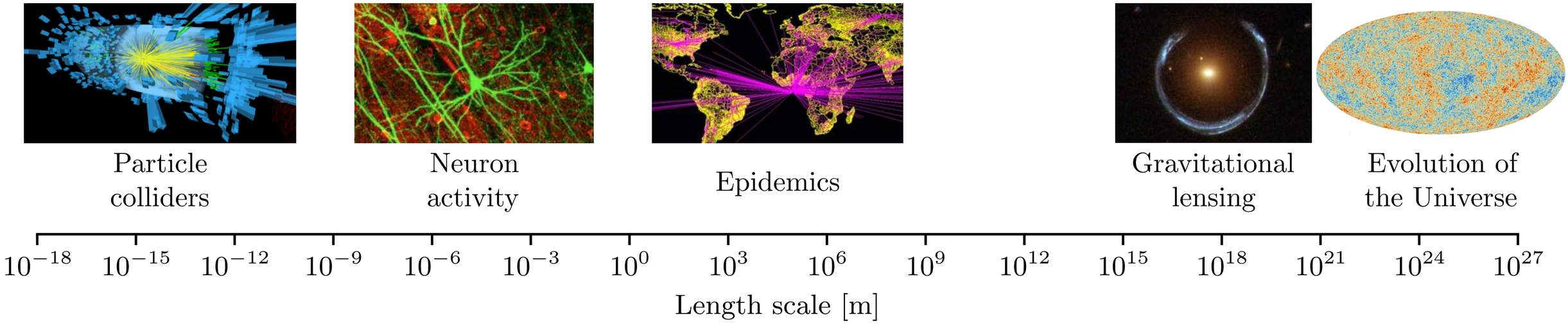
[Hubble Space Telescope]



[Nextstrain, CDC]

Big picture

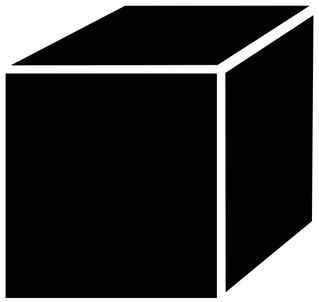
[K. Cranmer, JB, G. Louppe 1911.01429]



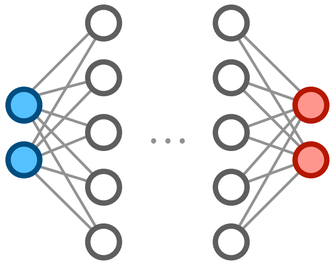
Simulators give high-precision predictions for many phenomena in science and engineering.

Unfortunately, they are poorly suited for inference.

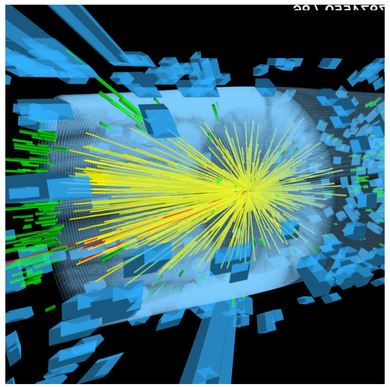
Machine learning can provide efficient models for powerful inference algorithms.



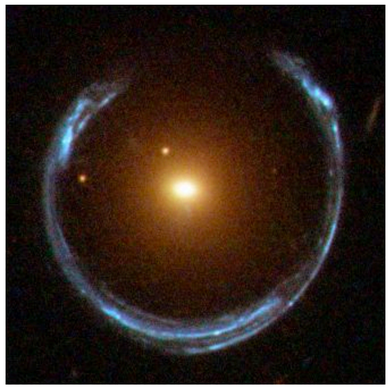
1. The simulation-based inference problem



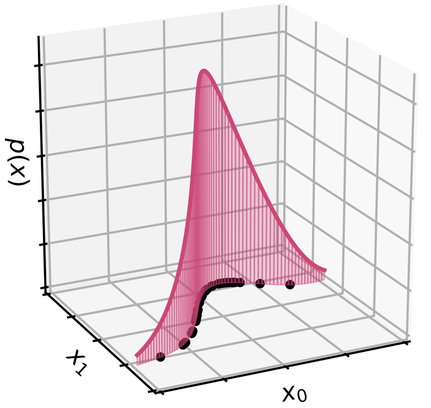
2. ML-based solutions



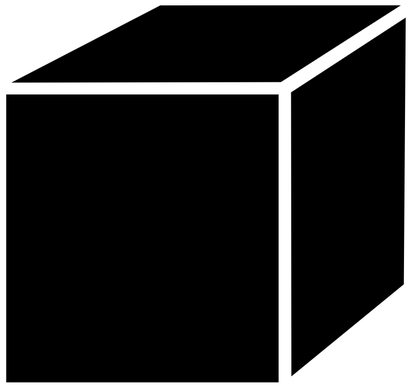
3. Particle physics



4. Astrophysics

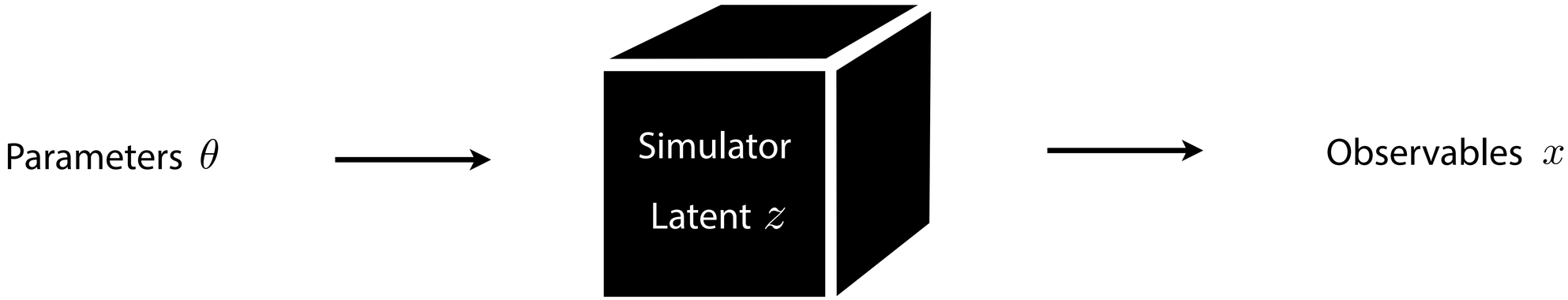


5. Tangents

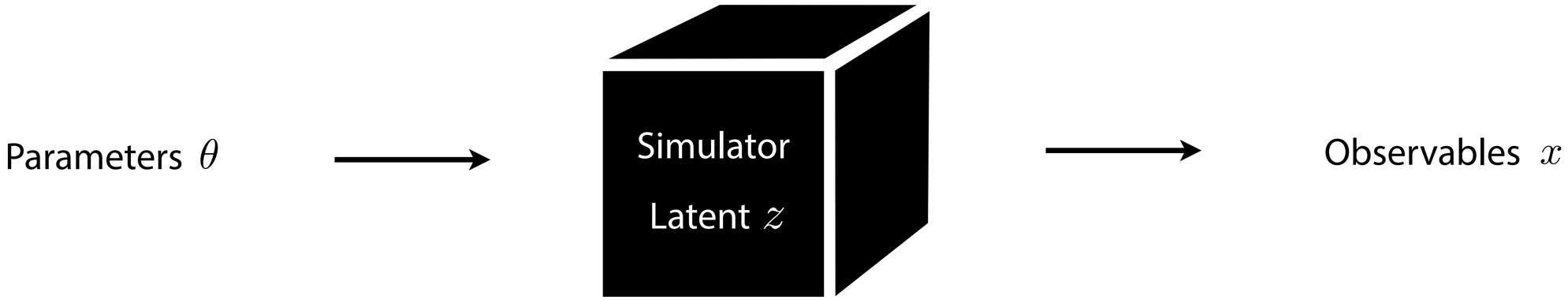


1. The simulation-based inference problem

The problem of simulation-based (“likelihood-free”) inference



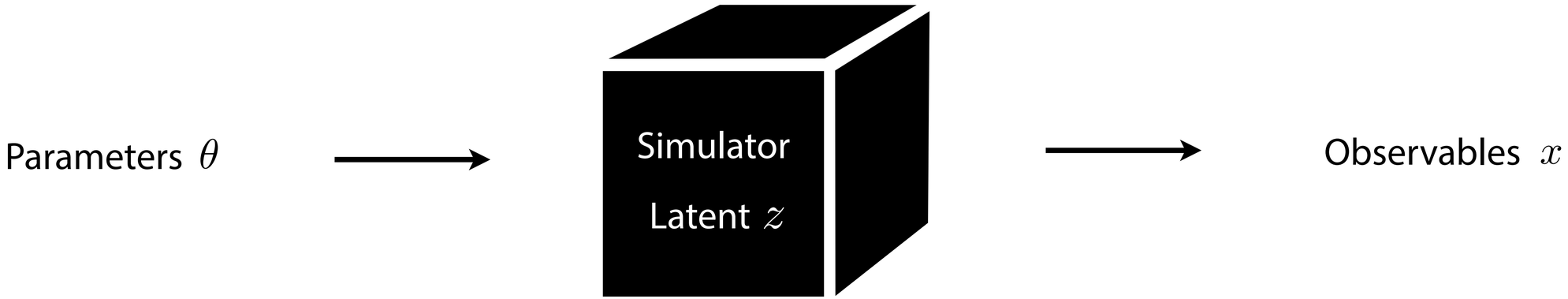
The problem of simulation-based (“likelihood-free”) inference



Prediction:

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

The problem of simulation-based (“likelihood-free”) inference



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

Three problem statements

Given

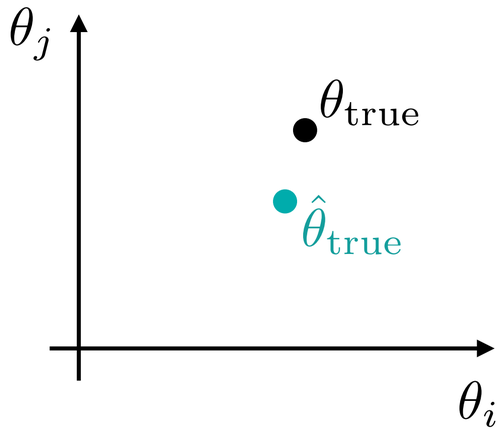
- a simulator that lets you generate N samples $x_i \sim p(x_i|\theta_i)$ (for parameters θ_i of our choice),
- observed data $x_{\mathrm{obs}} \sim p(x_{\mathrm{obs}}|\theta_{\mathrm{true}})$, and
- a prior $p(\theta)$,

Three problem statements

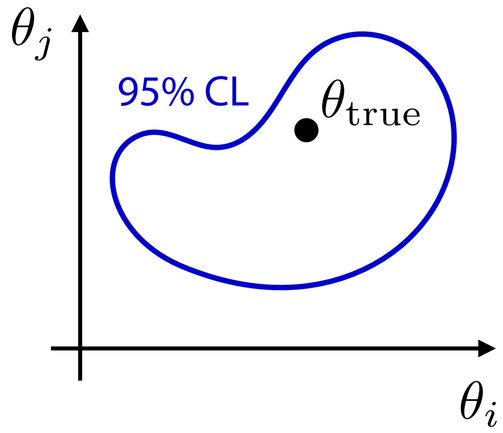
Given

- a simulator that lets you generate N samples $x_i \sim p(x_i|\theta_i)$ (for parameters θ_i of our choice),
- observed data $x_{\rm obs} \sim p(x_{\rm obs}|\theta_{\rm true})$, and
- a prior $p(\theta)$,

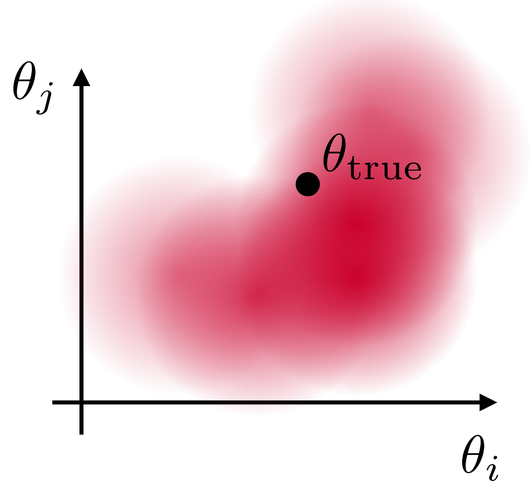
a) estimate $\hat{\theta}_{\rm true}$
(e.g. MLE)

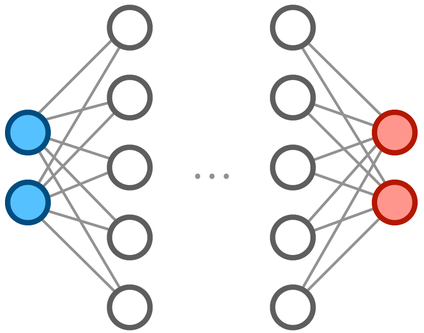


b) construct confidence sets
(e.g. likelihood ratio tests)



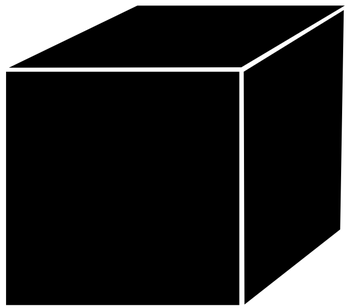
c) estimate the posterior
(or sample from posterior)





2. ML-based solutions

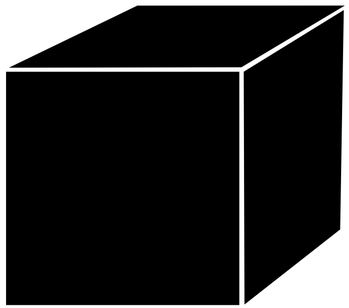
Get the best of two worlds



Simulators: focus on understanding

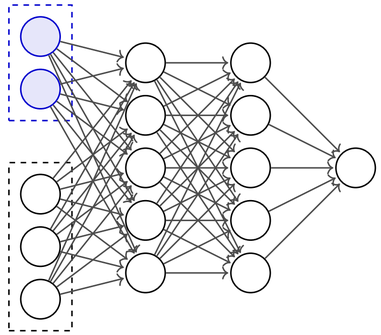
- based on mechanistic, causal model
- interpretable parameters

Get the best of two worlds



Simulators: focus on understanding

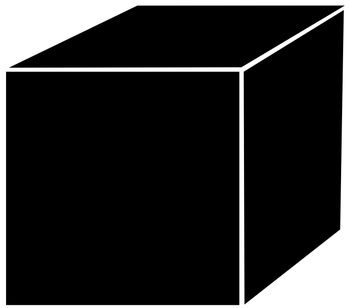
- based on mechanistic, causal model
- interpretable parameters



Machine learning models: focus on performance

- good at learning representations from data
- good inductive biases (images, sequences, graphs, symmetries, hierarchical structures…)
- differentiable, often invertible, probabilistic: well-suited for inference / fitting

Get the best of two worlds

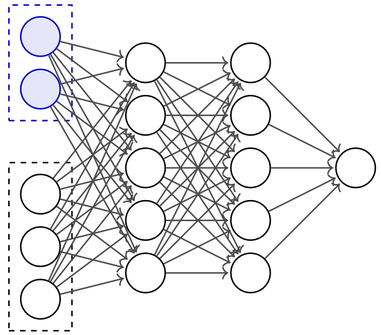


Simulators: focus on understanding

- based on mechanistic, causal model
- interpretable parameters

Can we use ML models to fit simulators to data?

Can we inject domain knowledge into ML models?

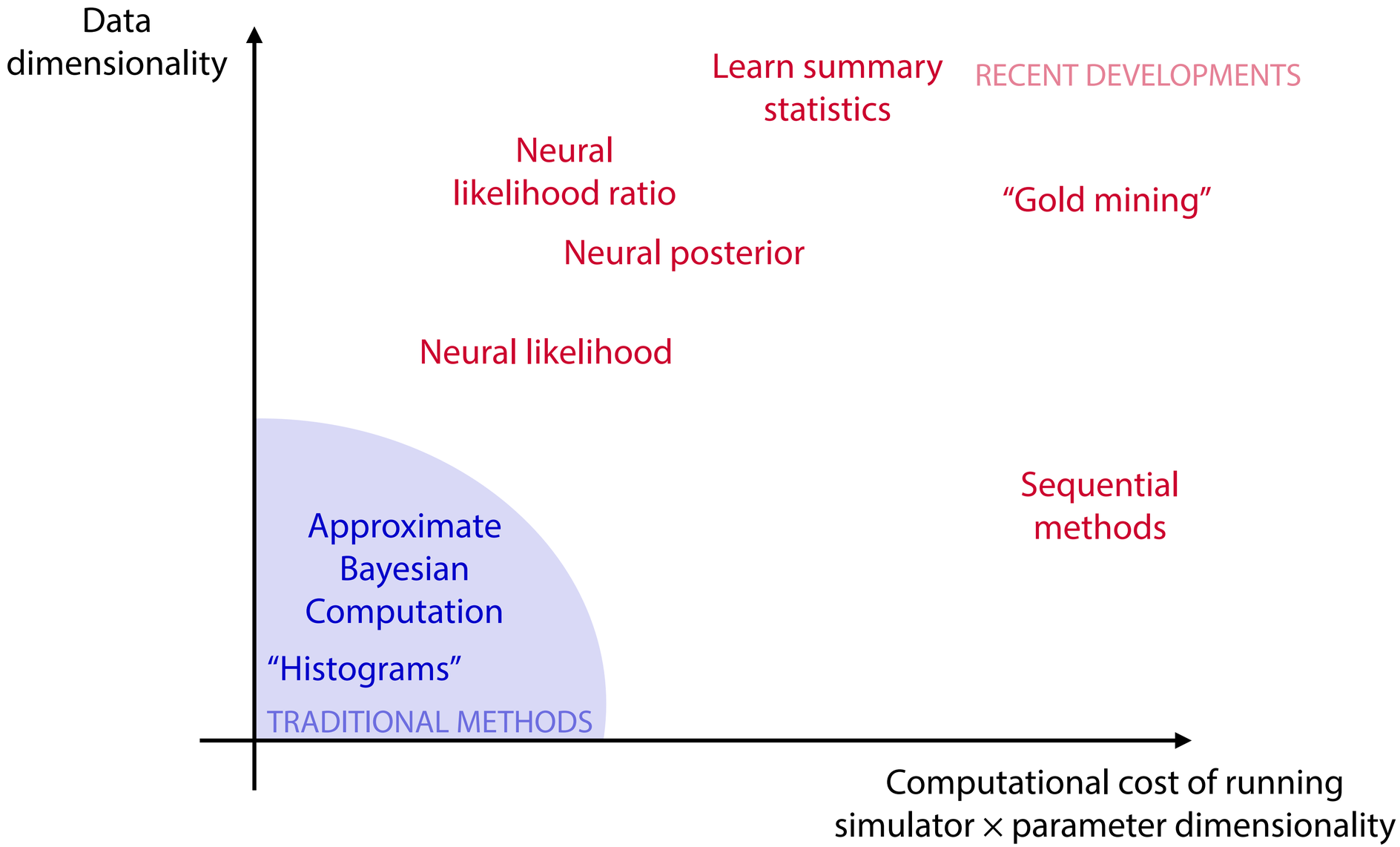


Machine learning models: focus on performance

- good at learning representations from data
- good inductive biases (images, sequences, graphs, symmetries, hierarchical structures…)
- differentiable, often invertible, probabilistic: well-suited for inference / fitting

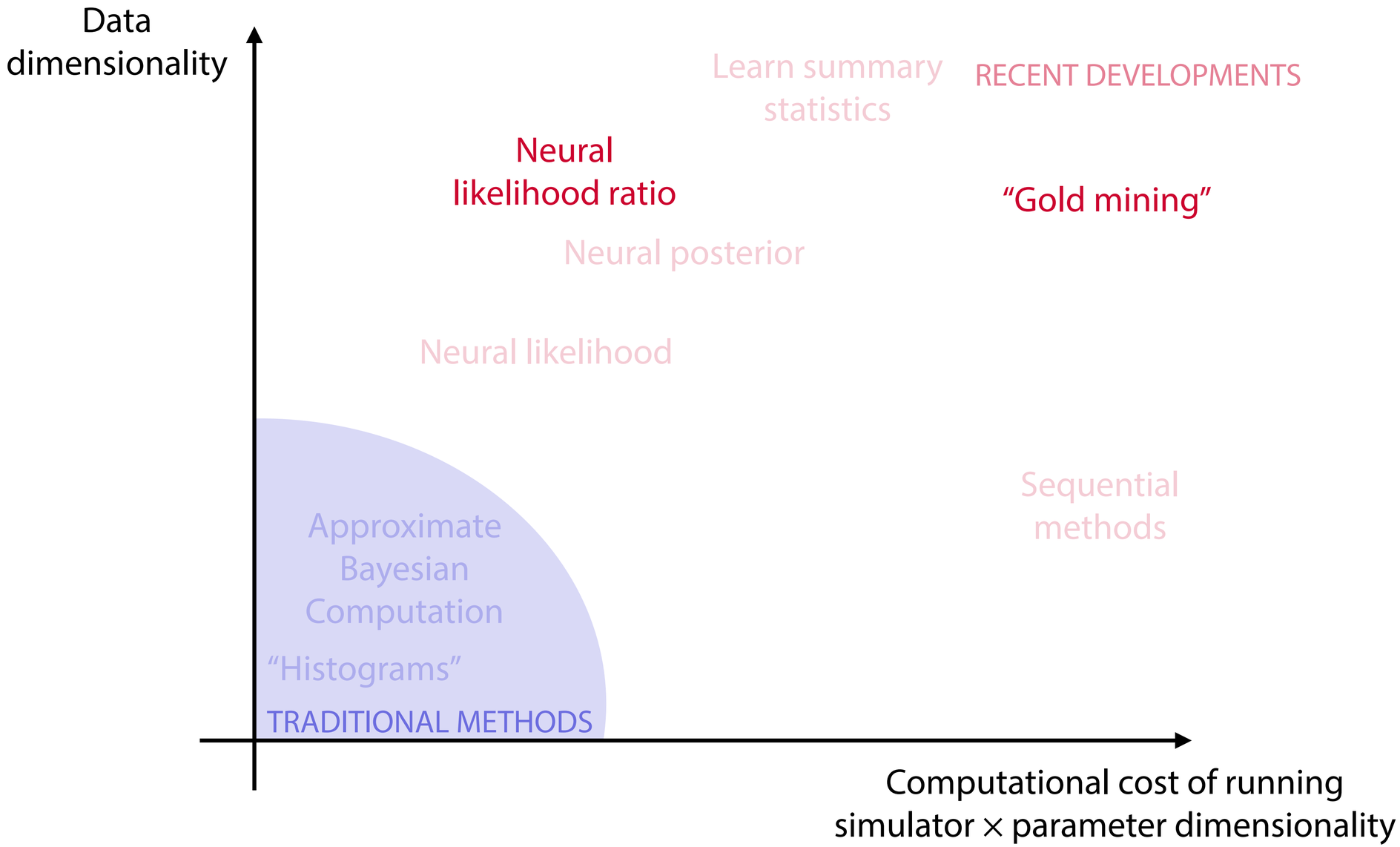
Simulation-based inference methods

[K. Cranmer, JB, G. Louppe 1911.01429]



Simulation-based inference methods

[K. Cranmer, JB, G. Louppe 1911.01429]

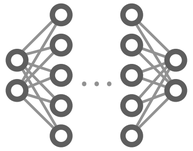


Idea 1: the likelihood ratio trick

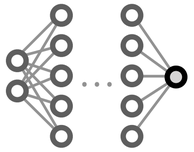
- Generative Adversarial Networks (GANs):

[I. Goodfellow et al. 1406.2661]

Generator
(intractable likelihood $p_{\text{gen}}(x)$)



Examples
(intractable likelihood $p_{\text{true}}(x)$)



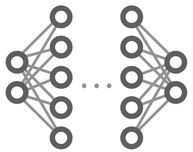
Discriminator

Idea 1: the likelihood ratio trick

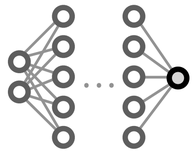
- Generative Adversarial Networks (GANs):

[I. Goodfellow et al. 1406.2661]

Generator
(intractable likelihood $p_{\text{gen}}(x)$)



Examples
(intractable likelihood $p_{\text{true}}(x)$)



Discriminator

Discriminator learns decision function

$$s(x) \rightarrow \frac{p_{\text{true}}(x)}{p_{\text{gen}}(x) + p_{\text{true}}(x)}$$

Idea 1: the likelihood ratio trick

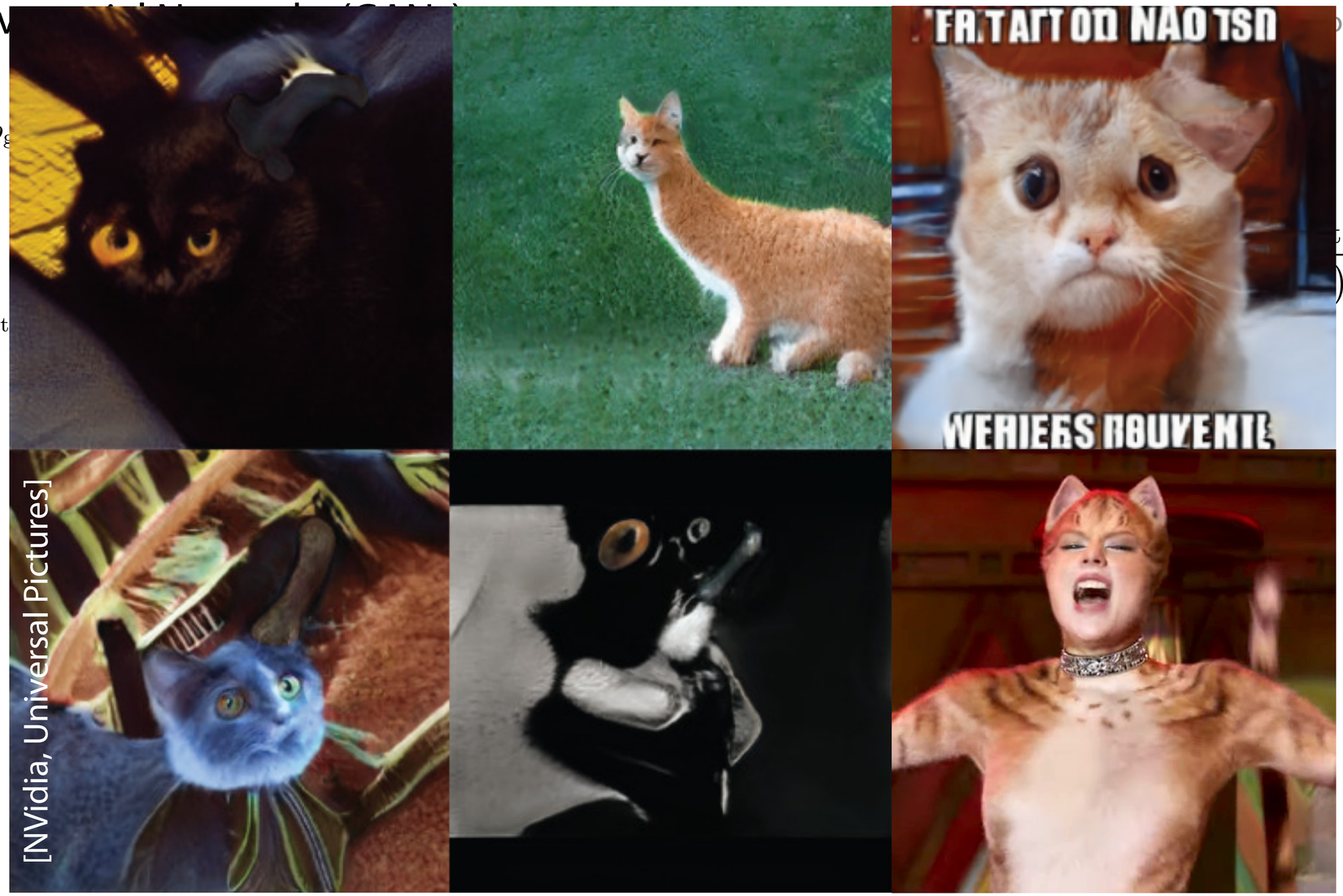
- Generative Adv [Goodfellow et al. 1406.2661]

Generator
(intractable likelihood p_g

Examples
(intractable likelihood p_t

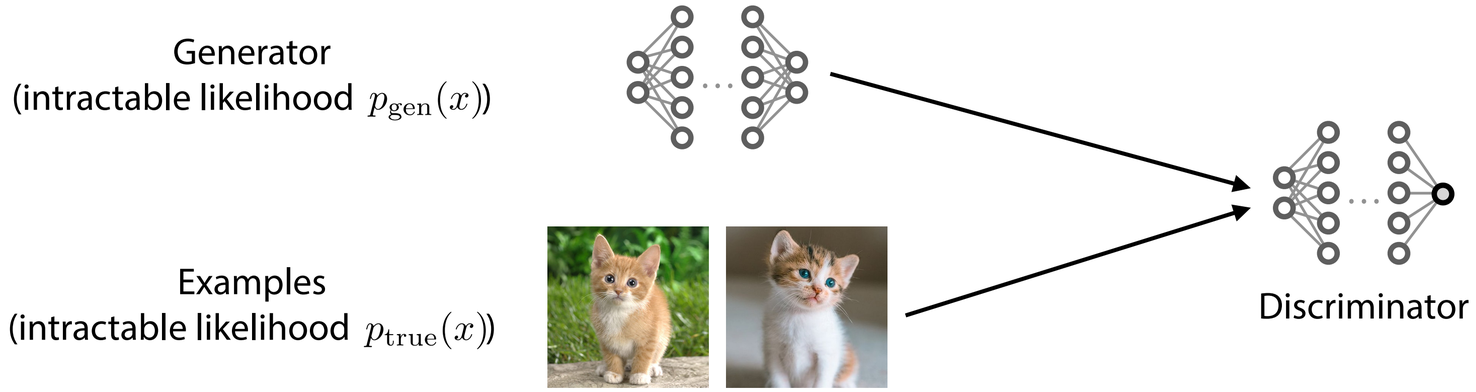
decision function

$$\frac{p_{\text{true}}(x)}{) + p_{\text{true}}(x)}$$



Idea 1: the likelihood ratio trick

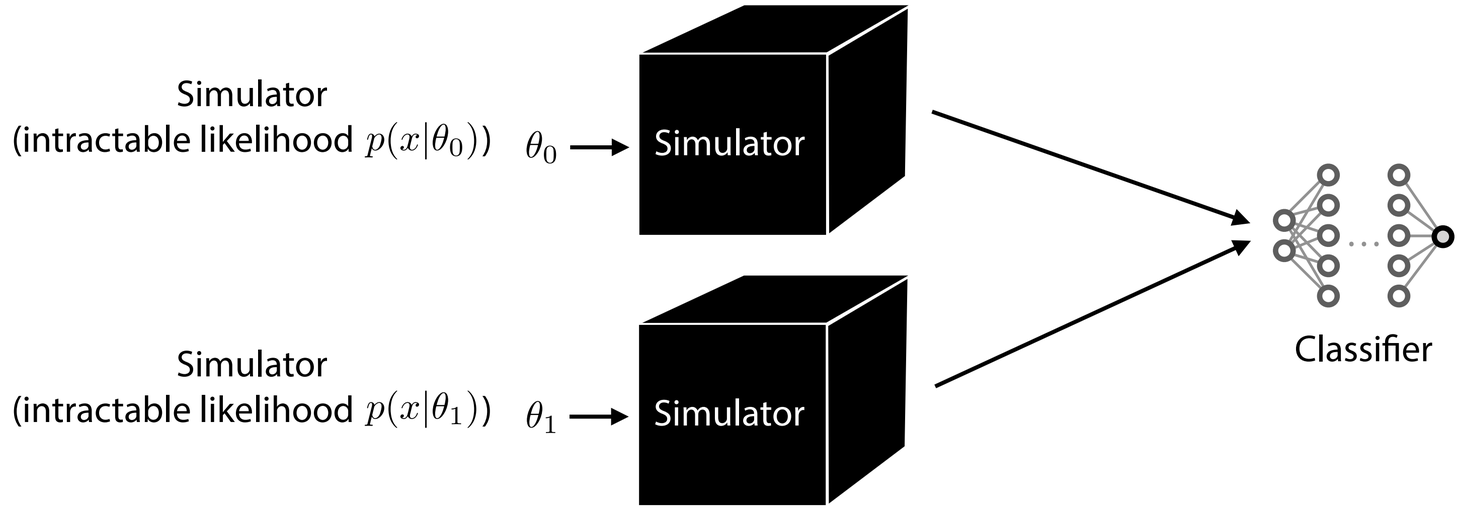
- Generative Adversarial Networks (GANs): [I. Goodfellow et al. 1406.2661]



Discriminator learns decision function

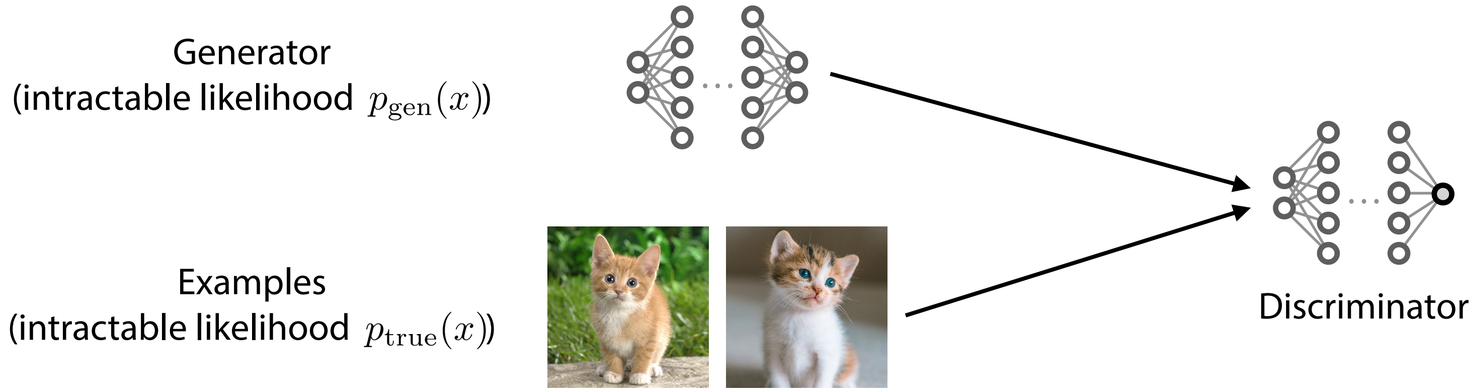
$$s(x) \to \frac{p_{\rm true}(x)}{p_{\rm gen}(x) + p_{\rm true}(x)}$$

- Similarly, we can train a classifier between two sets of simulated samples [K. Cranmer, J. Pavez, G. Louppe 1506.02169]



Idea 1: the likelihood ratio trick

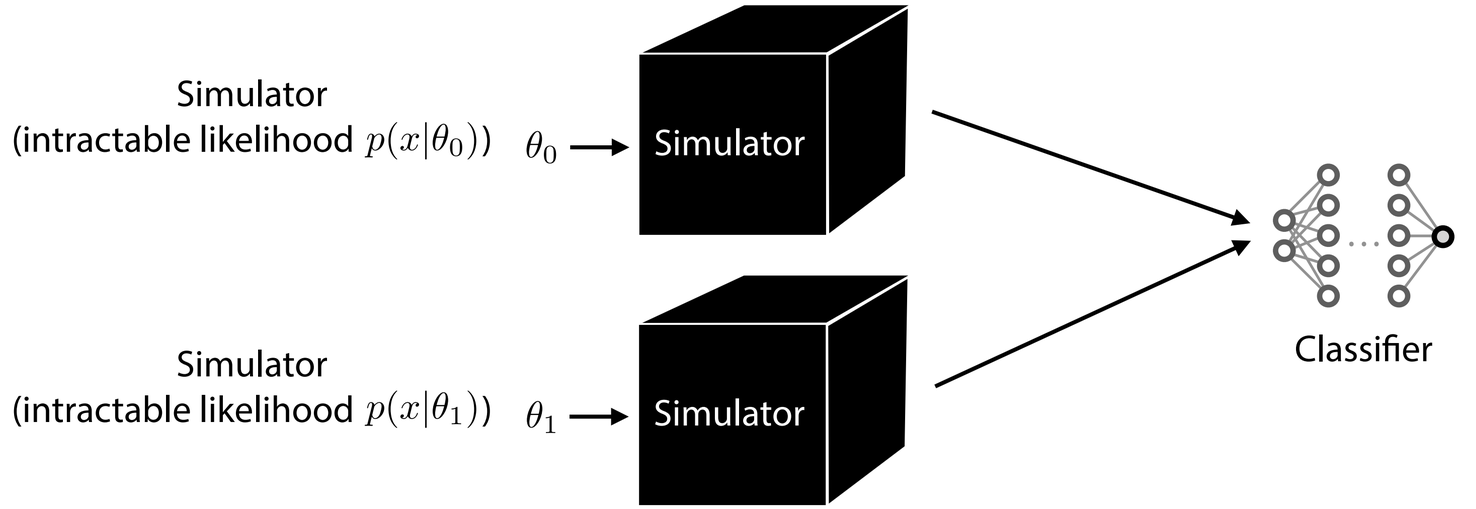
- Generative Adversarial Networks (GANs): [I. Goodfellow et al. 1406.2661]



Discriminator learns decision function

$$s(x) \to \frac{p_{\rm true}(x)}{p_{\rm gen}(x) + p_{\rm true}(x)}$$

- Similarly, we can train a classifier between two sets of simulated samples [K. Cranmer, J. Pavez, G. Louppe 1506.02169]



Classifier learns decision function

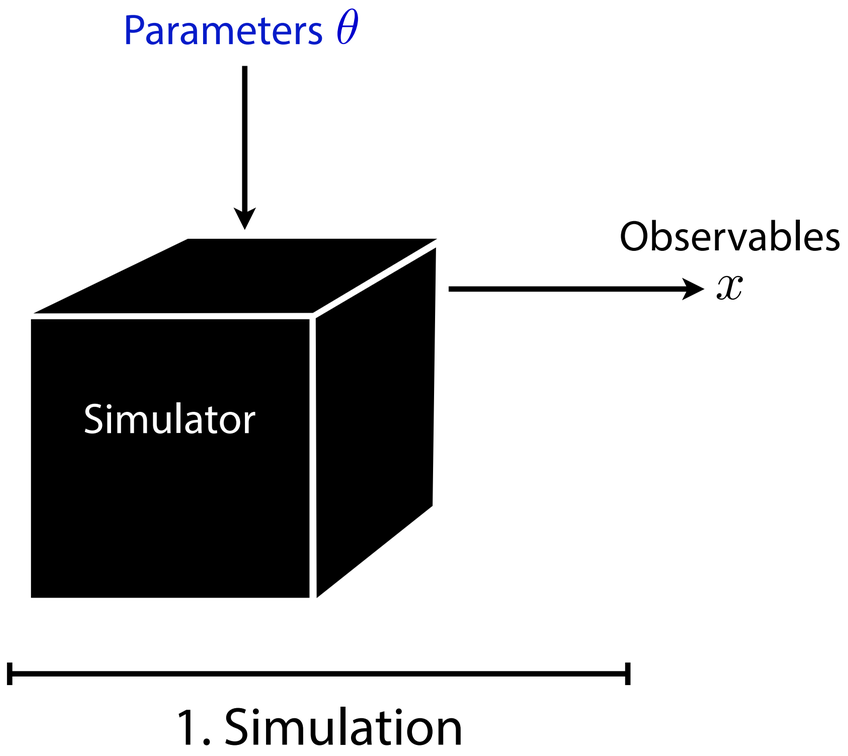
$$s(x) \to \frac{p(x|\theta_1)}{p(x|\theta_0) + p(x|\theta_1)}$$

⇒ Estimator for likelihood ratio

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

Inference by likelihood ratio trick

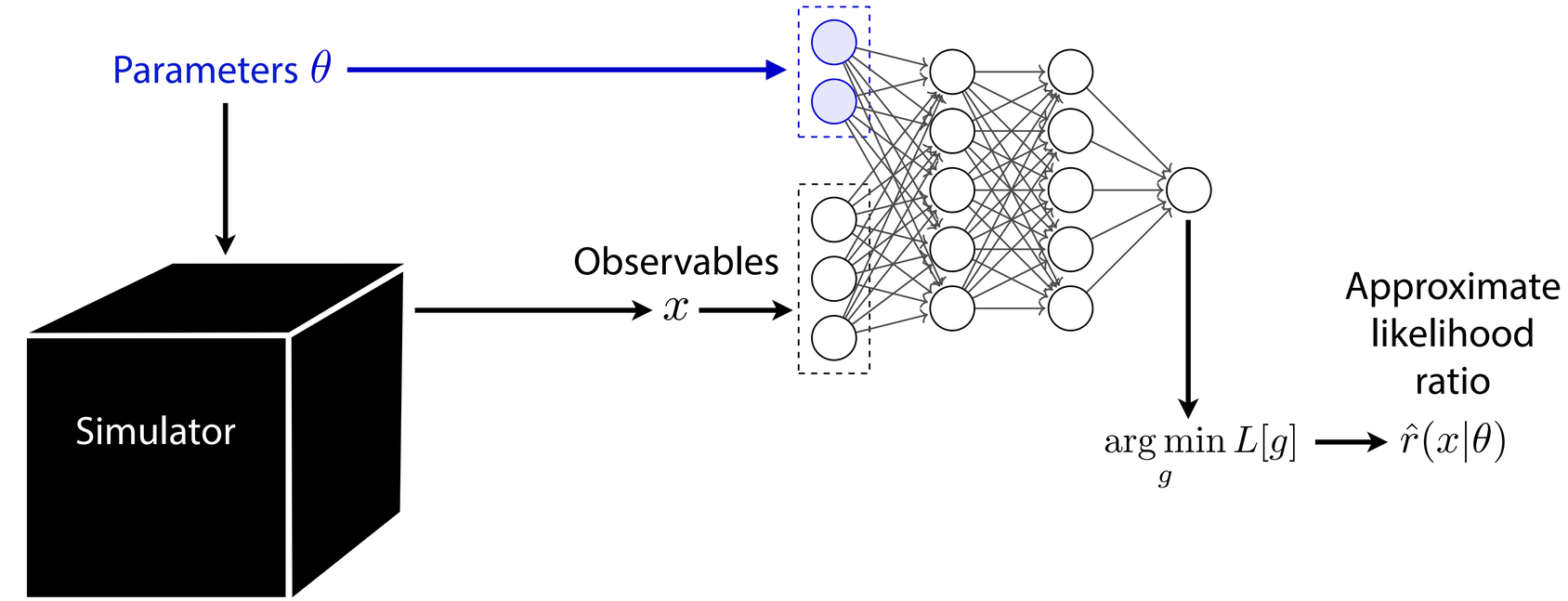
[K. Cranmer, J. Pavez, G. Louppe 1506.02169]



Run simulator and save data

Inference by likelihood ratio trick

[K. Cranmer, J. Pavez, G. Louppe 1506.02169]



1. Simulation

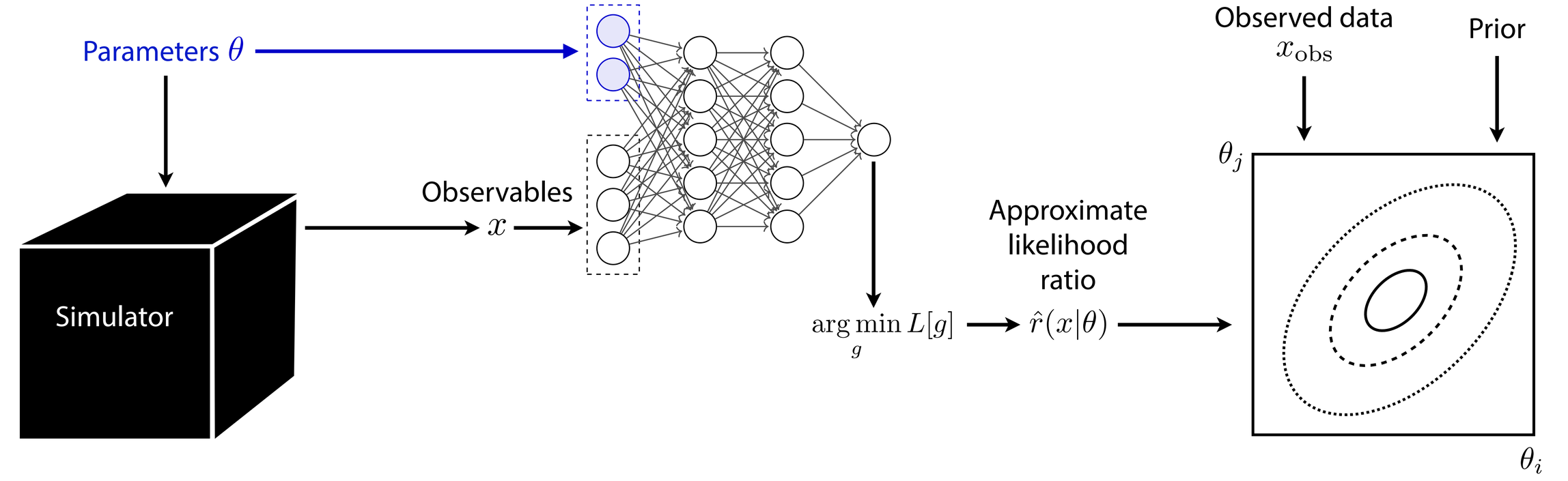
2. Machine Learning

Run simulator and save data

Train NN classifier, interpret as likelihood ratio estimator

Inference by likelihood ratio trick

[K. Cranmer, J. Pavez, G. Louppe 1506.02169]



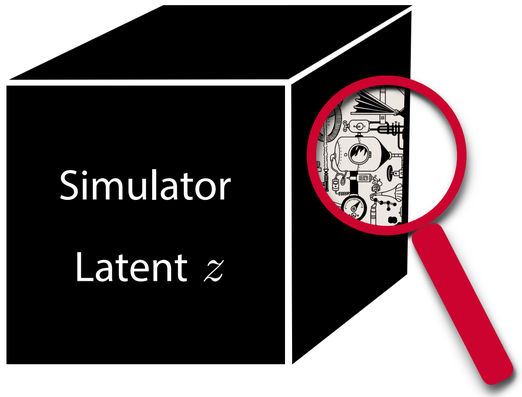
Run simulator and save data

Train NN classifier, interpret as likelihood ratio estimator

Amortized: cheap to repeat for new data

Idea 2: gold mining

[JB, G. Louppe, J. Pavez, K. Cranmer 1805.12244, 1805.00013, 1805.00020]



We cannot compute $p(x|\theta) = \int \mathrm{d}z\, p(x,z|\theta)$,

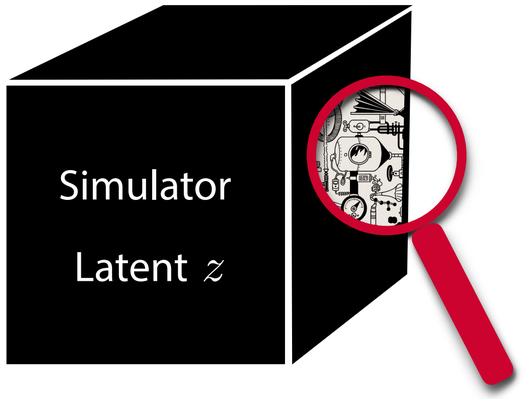
but often we can use domain knowledge (or probabilistic programming methods) to compute

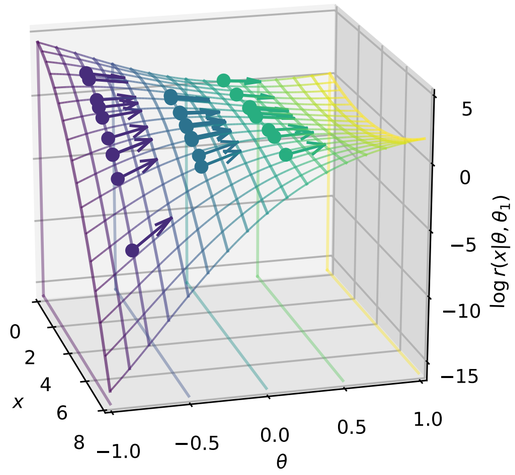
- the joint likelihood ratio $r(x,z|\theta) = \dfrac{p(x,z|\theta)}{p_{\mathrm{ref}}(x,z)}$
- the joint score $t(x,z|\theta) = \nabla_\theta \log p(x,z|\theta)$

(Both depend on the simulator latent variables z)

Idea 2: gold mining

[JB, G. Louppe, J. Pavez, K. Cranmer 1805.12244, 1805.00013, 1805.00020]





We cannot compute $p(x|\theta) = \int \mathrm{d}z\, p(x,z|\theta)$,

but often we can use domain knowledge (or probabilistic programming methods) to compute

- the joint likelihood ratio $r(x,z|\theta) = \dfrac{p(x,z|\theta)}{p_{\mathrm{ref}}(x,z)}$
- the joint score $t(x,z|\theta) = \nabla_\theta \log p(x,z|\theta)$

(Both depend on the simulator latent variables z)

Pleasant surprises: we have shown that

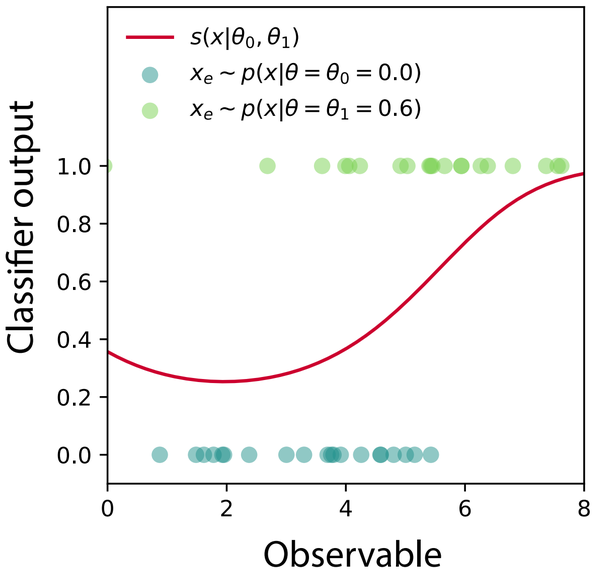
- the joint likelihood ratio is an unbiased estimator of the likelihood ratio
- the joint score provides unbiased gradient information

⇒ use them as labels in supervised NN training!

Mining gold adds information

[JB, G. Louppe, J. Pavez, K. Cranmer
1805.12244, 1805.00013, 1805.00020]

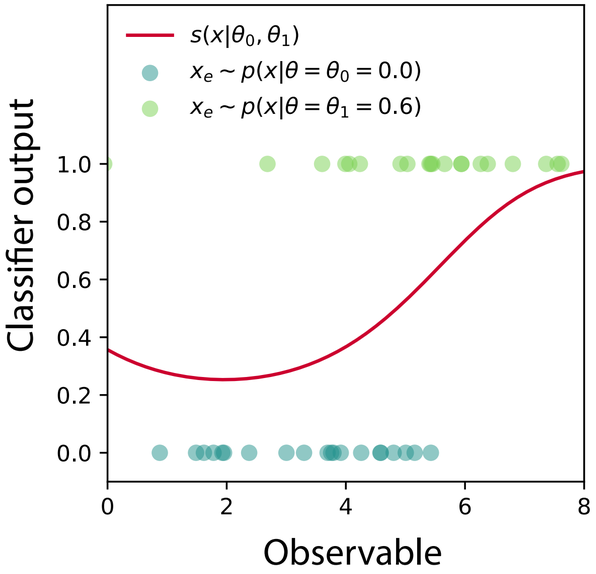
Likelihood ratio trick



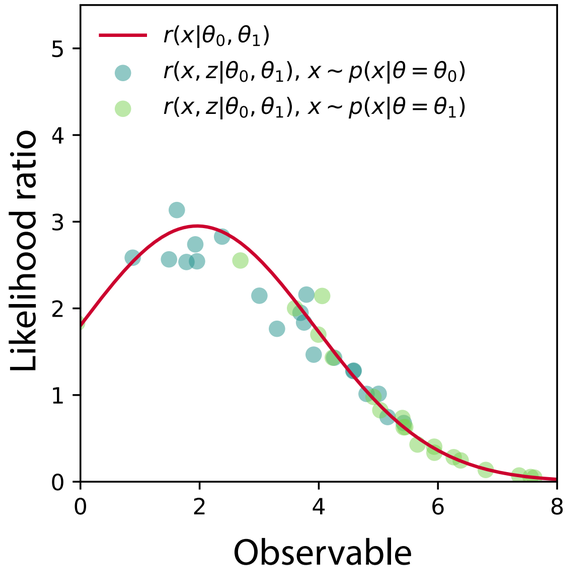
Mining gold adds information

[JB, G. Louppe, J. Pavez, K. Cranmer
1805.12244, 1805.00013, 1805.00020]

Likelihood ratio trick



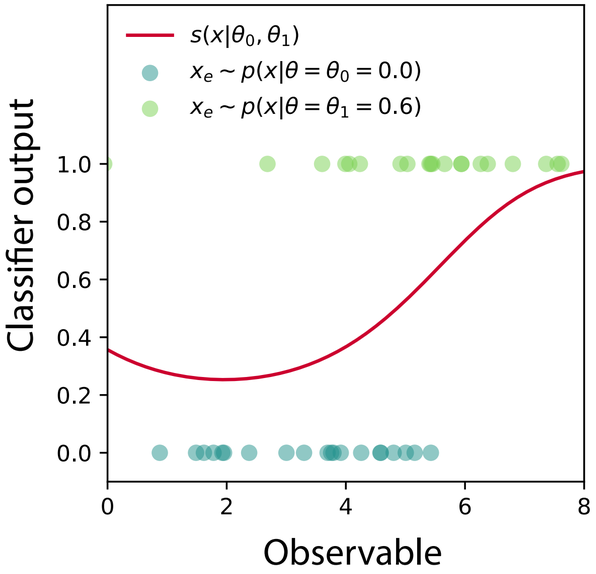
+ joint likelihood ratio



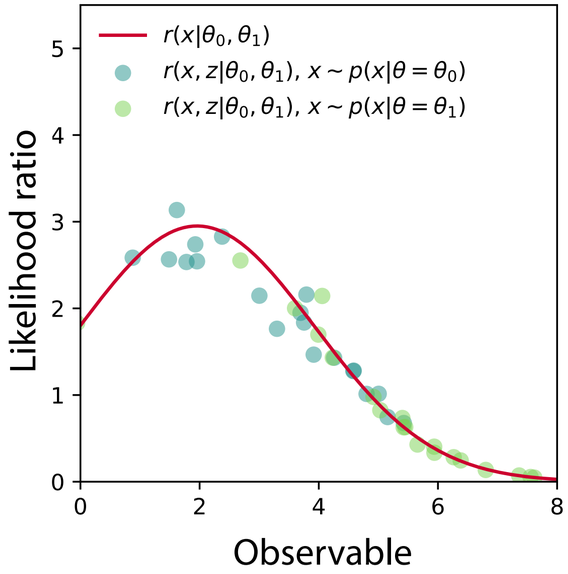
Mining gold adds information

[JB, G. Louppe, J. Pavez, K. Cranmer
1805.12244, 1805.00013, 1805.00020]

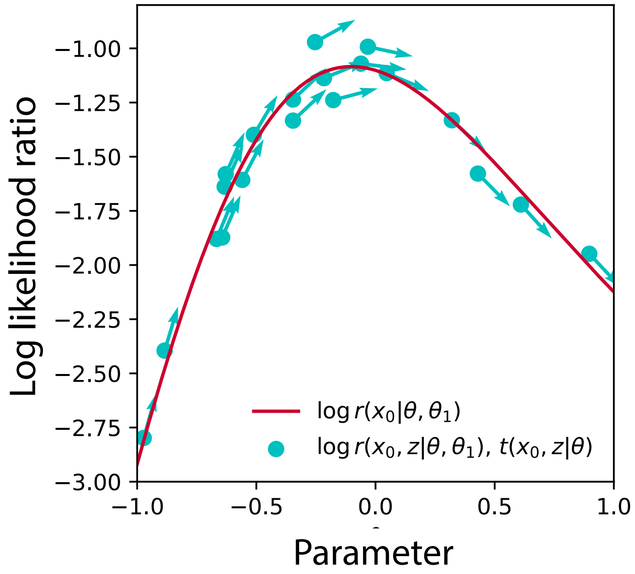
Likelihood ratio trick



+ joint likelihood ratio



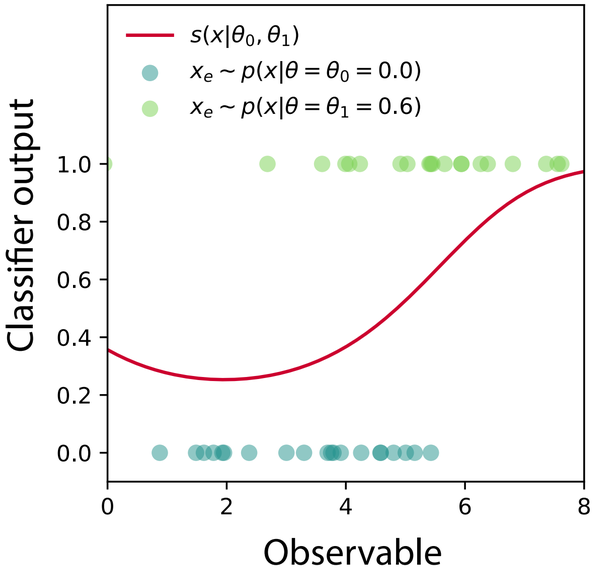
+ joint score



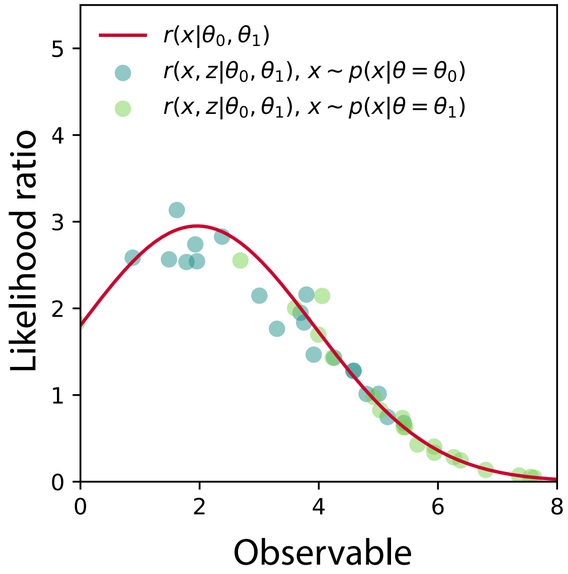
Mining gold adds information

[JB, G. Louppe, J. Pavez, K. Cranmer
1805.12244, 1805.00013, 1805.00020]

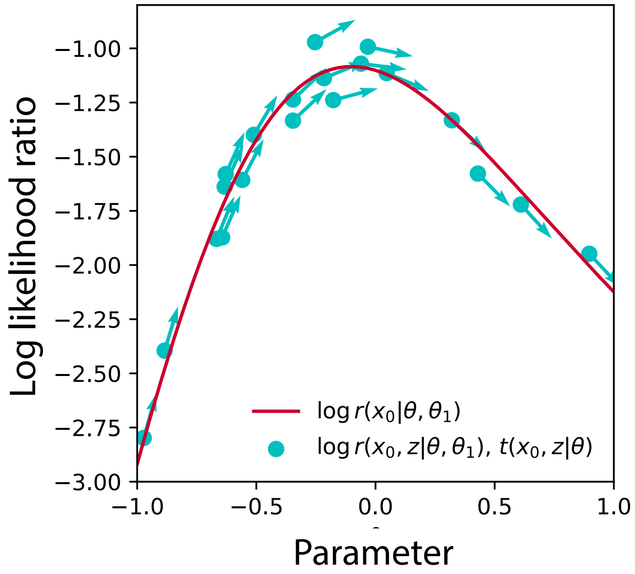
Likelihood ratio trick



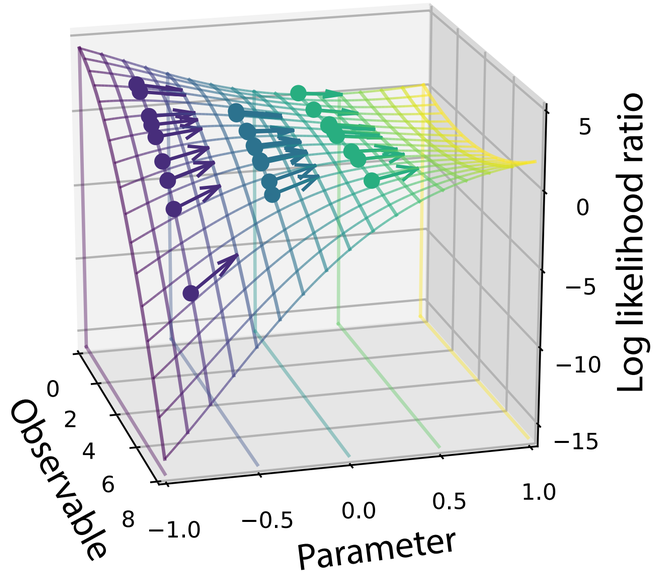
+ joint likelihood ratio



+ joint score



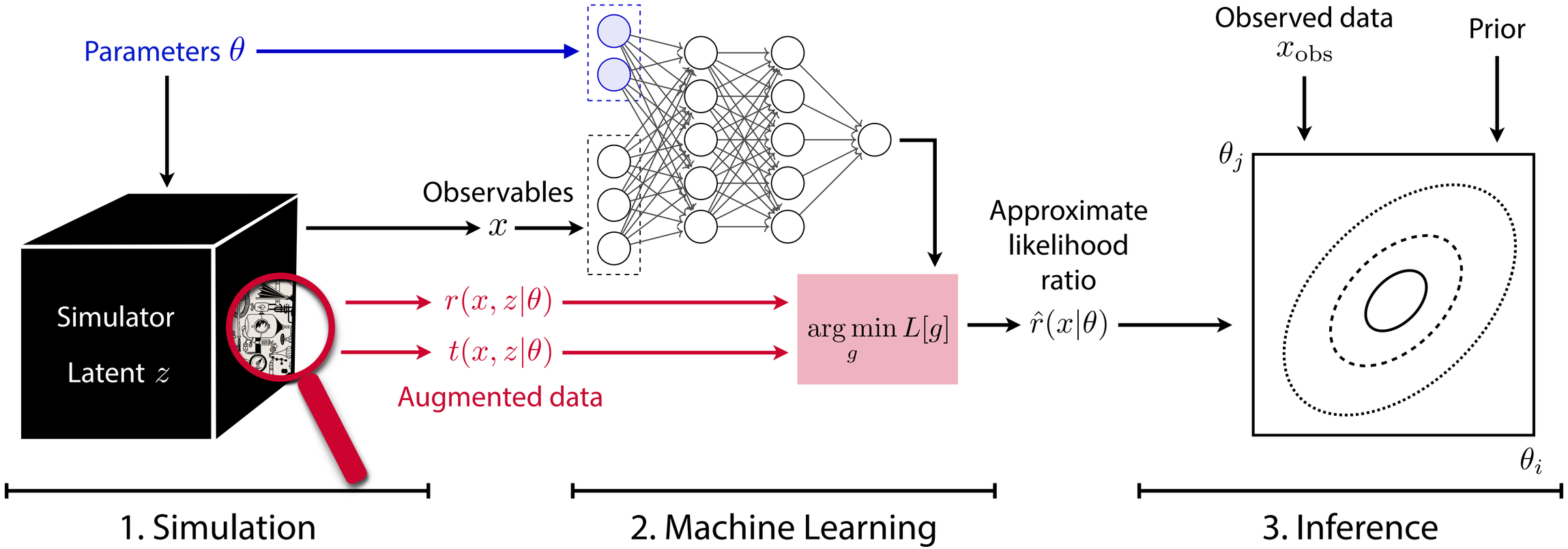
= RASCAL



Using more information = more sample-efficient inference

RASCAL: Likelihood ratio trick + gold mining

[JB, G. Louppe, J. Pavez, K. Cranmer
1805.12244, 1805.00013, 1805.00020]

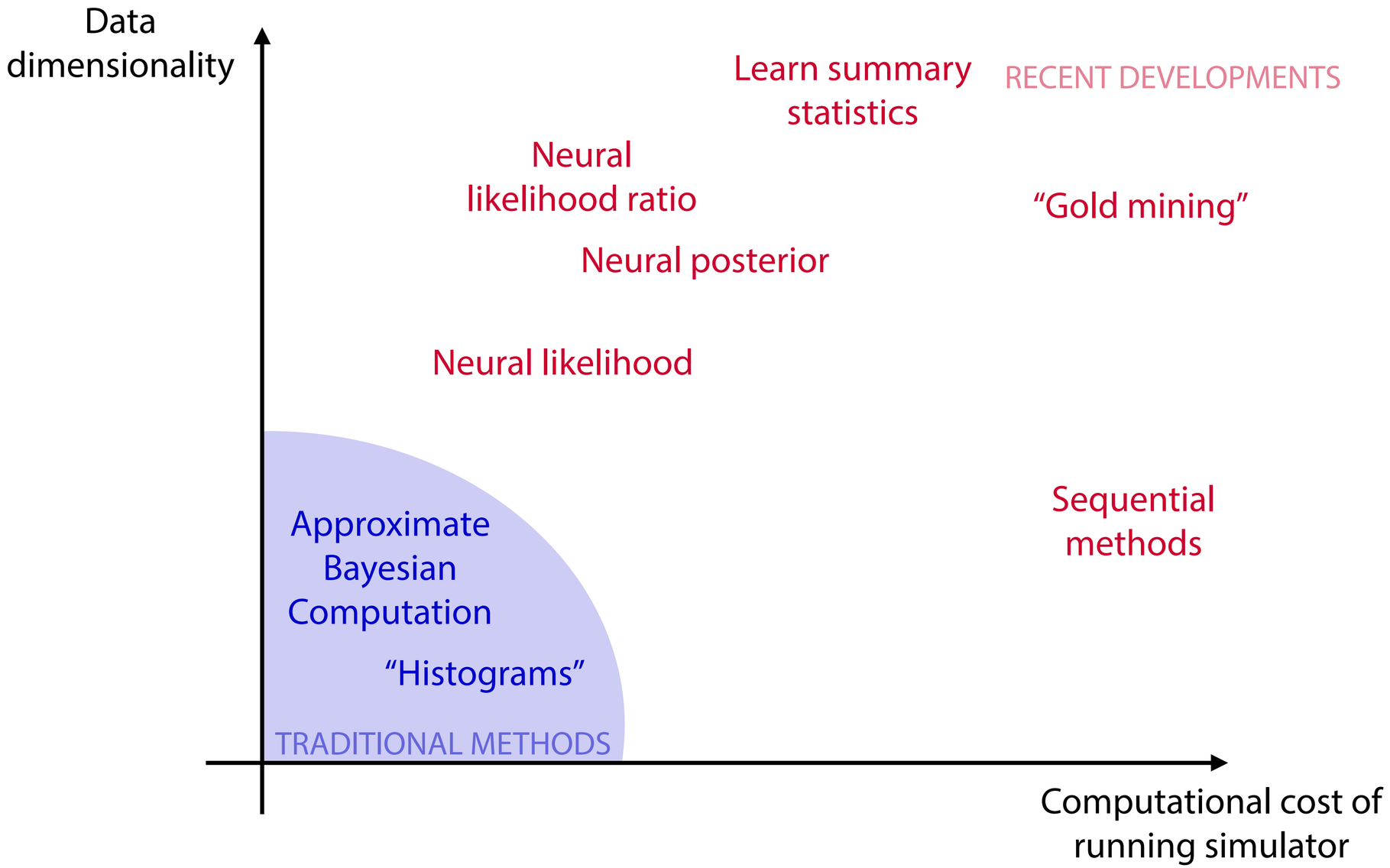


Extract joint likelihood ratio
and joint score from simulator

Augment training data &
use as labels in new loss functions
⇒ improve training efficiency

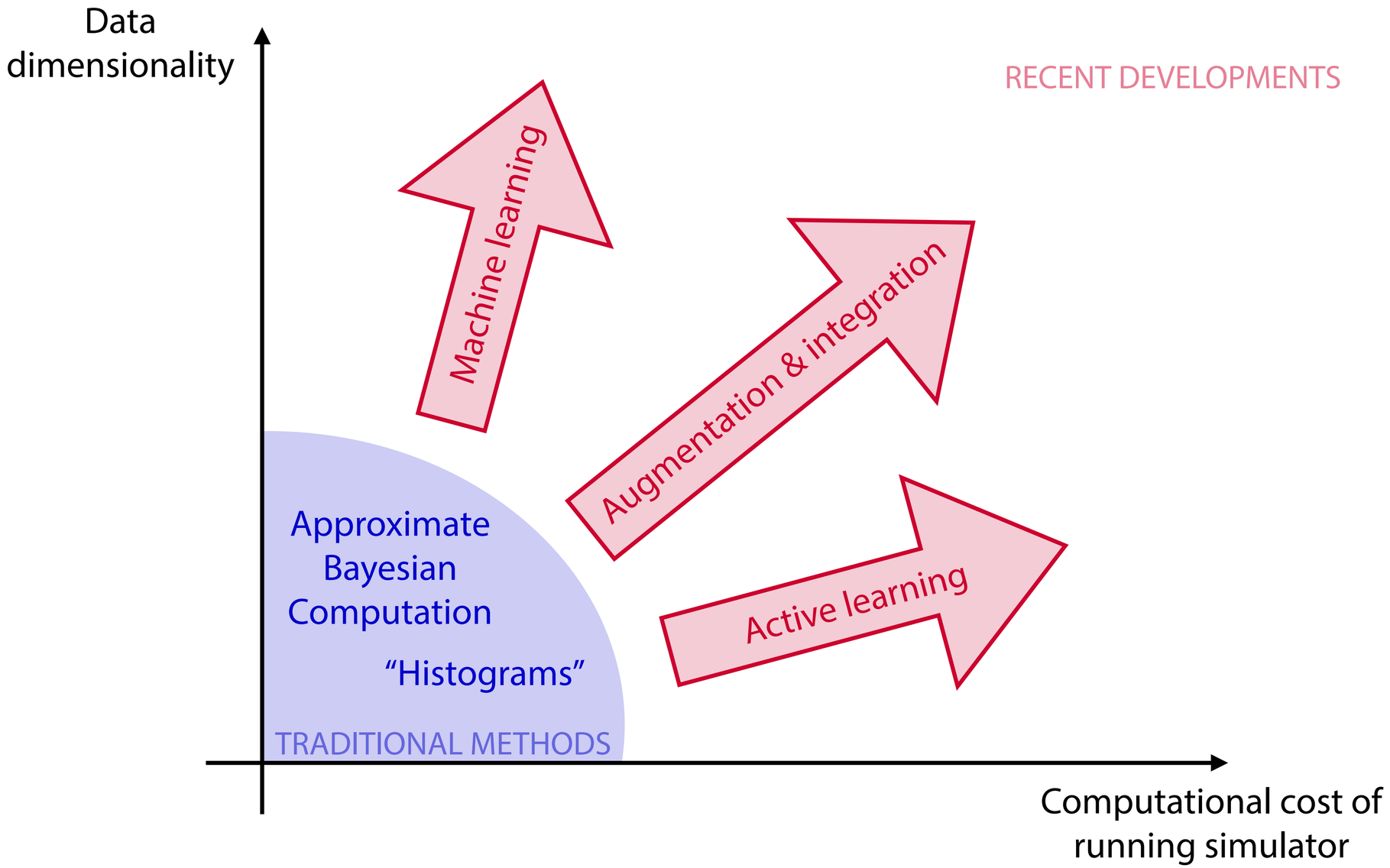
The frontier of simulation-based inference

[K. Cranmer, JB, G. Louppe 1911.01429]

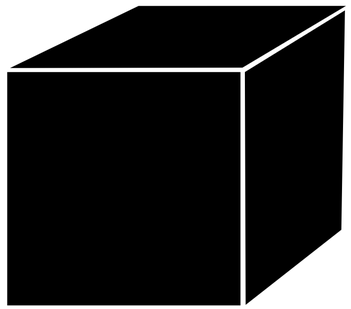


The frontier of simulation-based inference

[K. Cranmer, JB, G. Louppe 1911.01429]



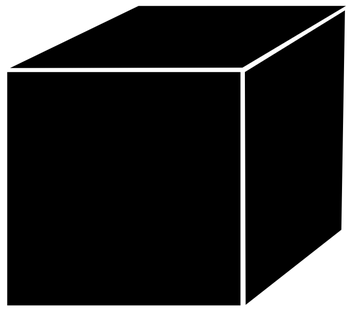
The Achilles heel: model misspecification

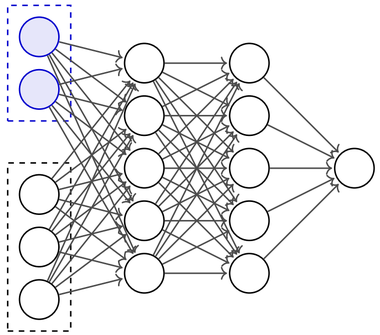


Can you trust the simulator?

- Model uncertainties explicitly:
  nuisance parameters + profiling / marginalization
- Make analysis robust:
  ideas from domain adaptation, algorithmic fairness
  [G. Louppe, M. Kagan, K. Cranmer 1611.01046; J. Alsing, B. Wandelt 1903.01473; P. de Castro, T. Dorigo 1806.04743]

The Achilles heel: model misspecification



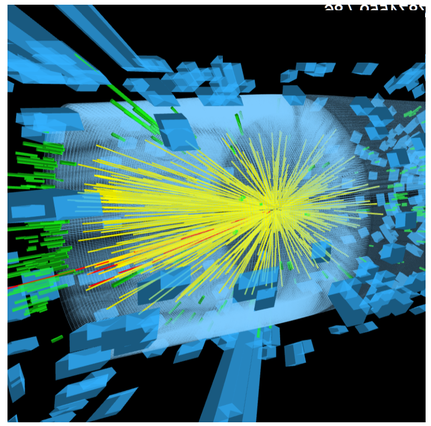


Can you trust the simulator?

- Model uncertainties explicitly:
  nuisance parameters + profiling / marginalization
- Make analysis robust:
  ideas from domain adaptation, algorithmic fairness
  [G. Louppe, M. Kagan, K. Cranmer 1611.01046; J. Alsing, B. Wandelt 1903.01473; P. de Castro, T. Dorigo 1806.04743]

Can you trust the neural network?

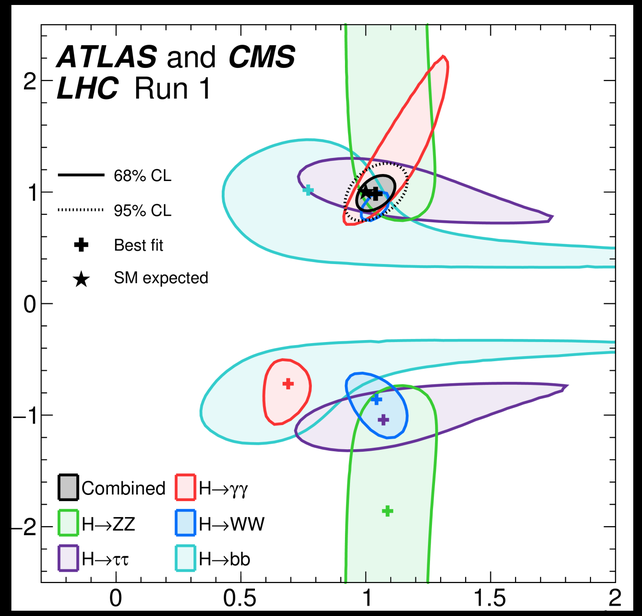
- Sanity checks: expectation values, "critic" tests
- Neyman construction with toys
  (badly trained network can lead to suboptimal limits, but not to wrong limits)
  [JB, G. Louppe, J. Pavez, K. Cranmer 1805.00020]
- Empirically, ensembling and calibration help
  [JB, G. Louppe, J. Pavez, K. Cranmer 1805.00020; J. Hermans et al 2110.06581]



3. Particle physics

High-dimensional event data

[CMS]

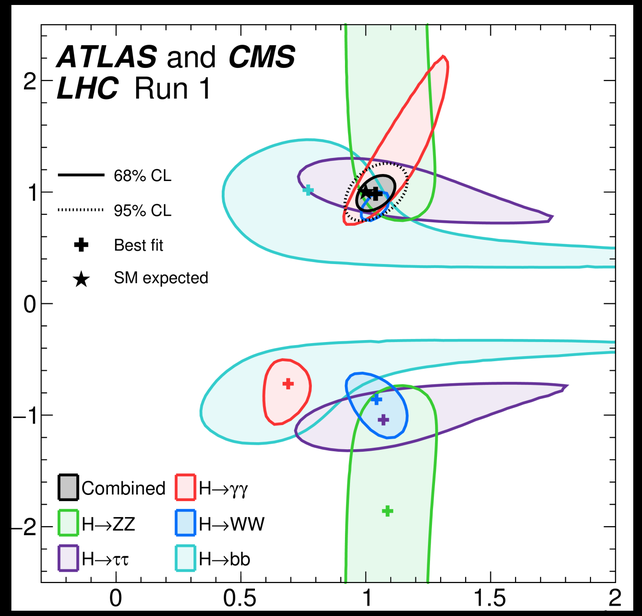


[ATLAS, CMS 1606.02266]

Precision constraints on new physics

High-dimensional event data

[CMS]



[ATLAS, CMS 1606.02266]

Precision constraints on new physics

Parameterization e.g. in Effective Field Theory:

$$\mathcal{L}_{\mathrm{EFT}} = \mathcal{L}_{\mathrm{SM}} + \sum_i \frac{f_i}{\Lambda^2} \mathcal{O}_i + \dots$$

systematic expansion of new physics around Standard Model

10s to 100s "universal" parameters to measure

Modelling LHC processes

Theory
parameters
θ

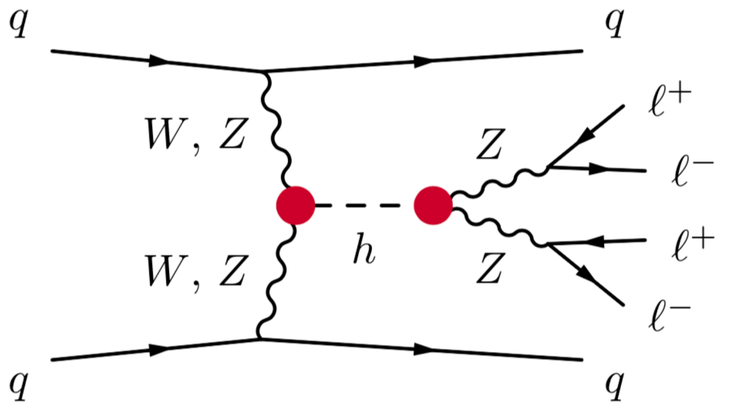
Evolution

Modelling LHC processes

Latent variables

Parton-level momenta z_p

Theory parameters θ

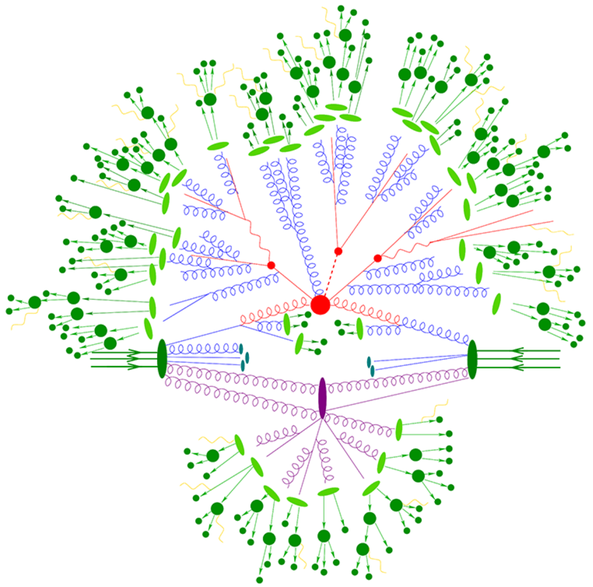


Evolution

Modelling LHC processes

Latent variables

Shower splittings z_s ← Parton-level momenta z_p ← Theory parameters θ



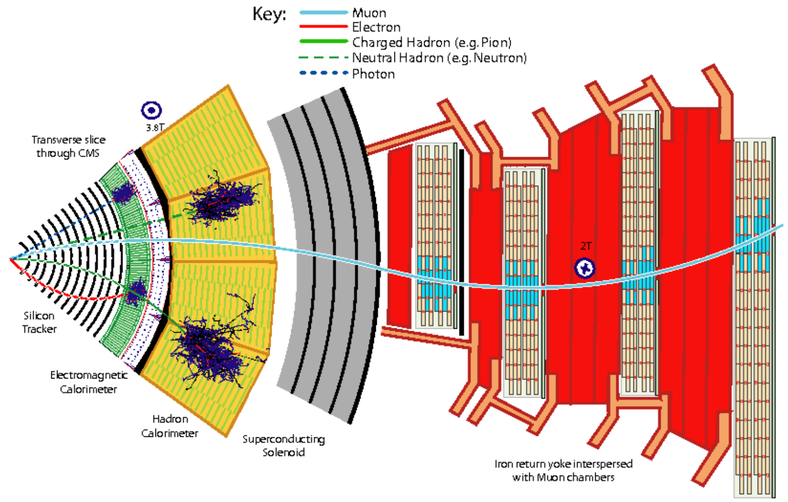
[F. Krauss]

Evolution

Modelling LHC processes

Latent variables

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



[CMS]

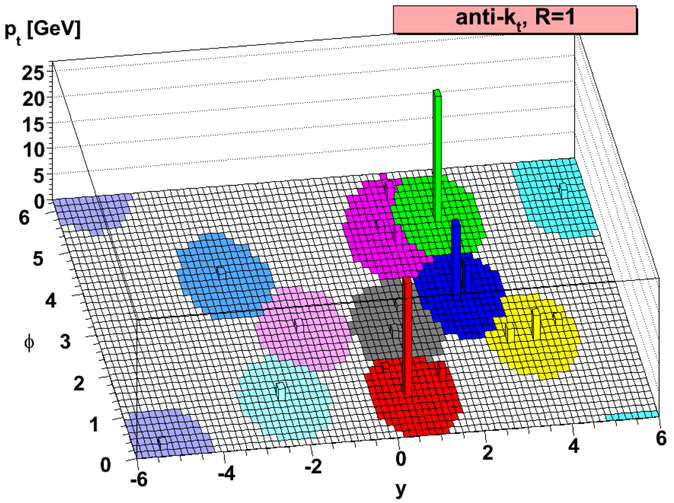
Evolution

Modelling LHC processes

Latent variables

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

$x \leftarrow z_d \leftarrow z_s \leftarrow z_p \leftarrow \theta$



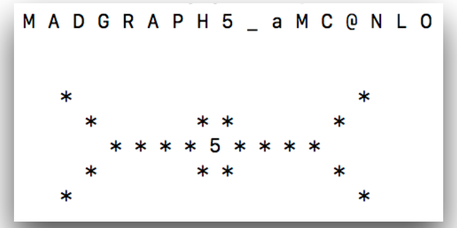
[M. Cacciari, G. Salam, G. Soyez 0802.1189]

Evolution

Modelling LHC processes

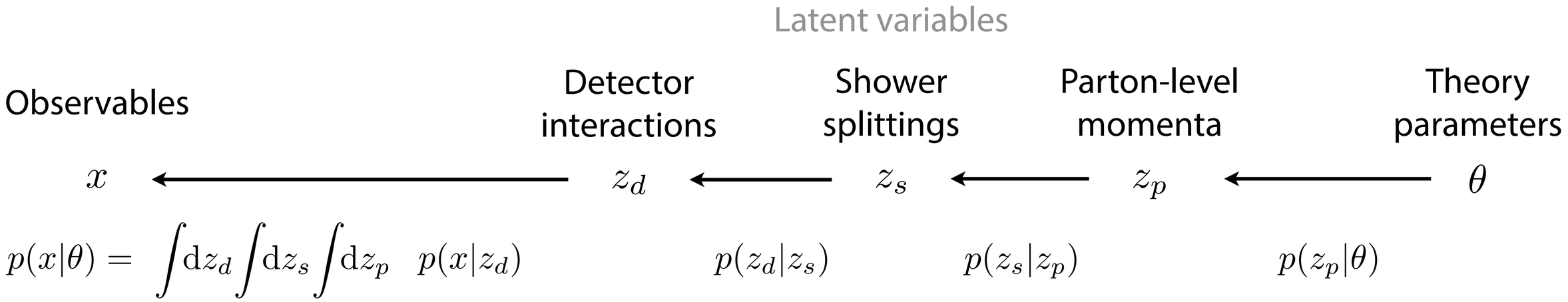
Latent variables

Observables	Detector interactions	Shower splittings	Parton-level momenta	Theory parameters
x	z_d	z_s	z_p	θ
Sample from	$p(x\|z_d)$	$p(z_d\|z_s)$	$p(z_s\|z_p)$	$p(z_p\|\theta)$



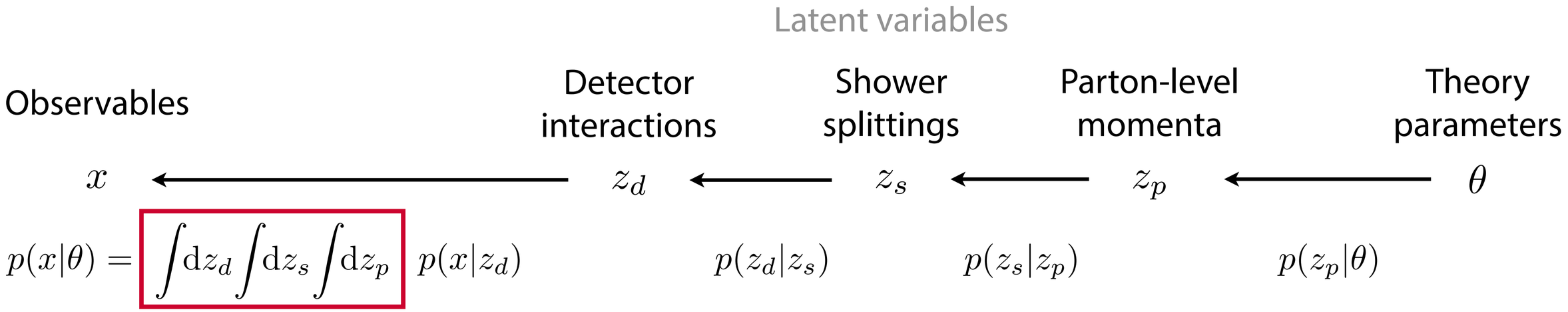
Prediction (simulation)

Modelling LHC processes



Inference

Modelling LHC processes

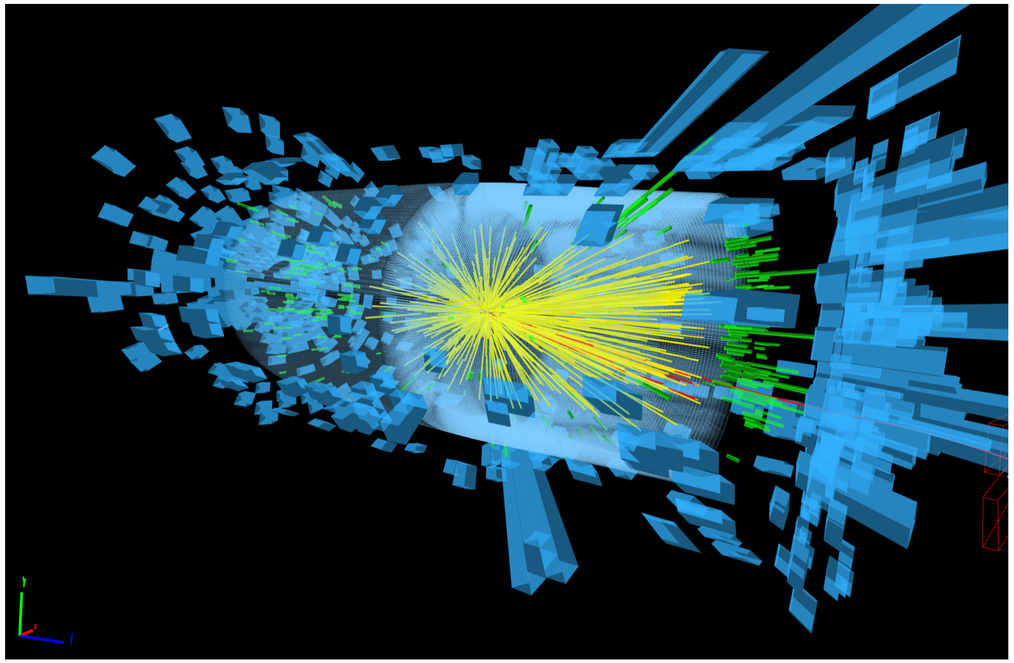


It's infeasible to calculate the integral over this enormous space!

(More subtle: We cannot sample from $p(z|x,\theta)$ efficiently.)

Inference

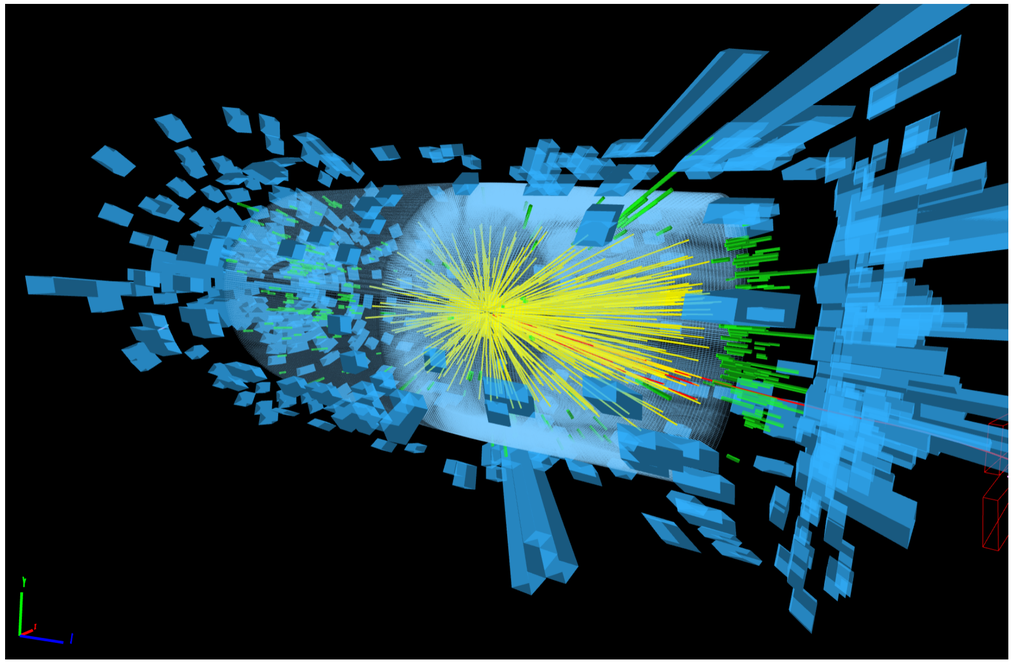
The traditional solution

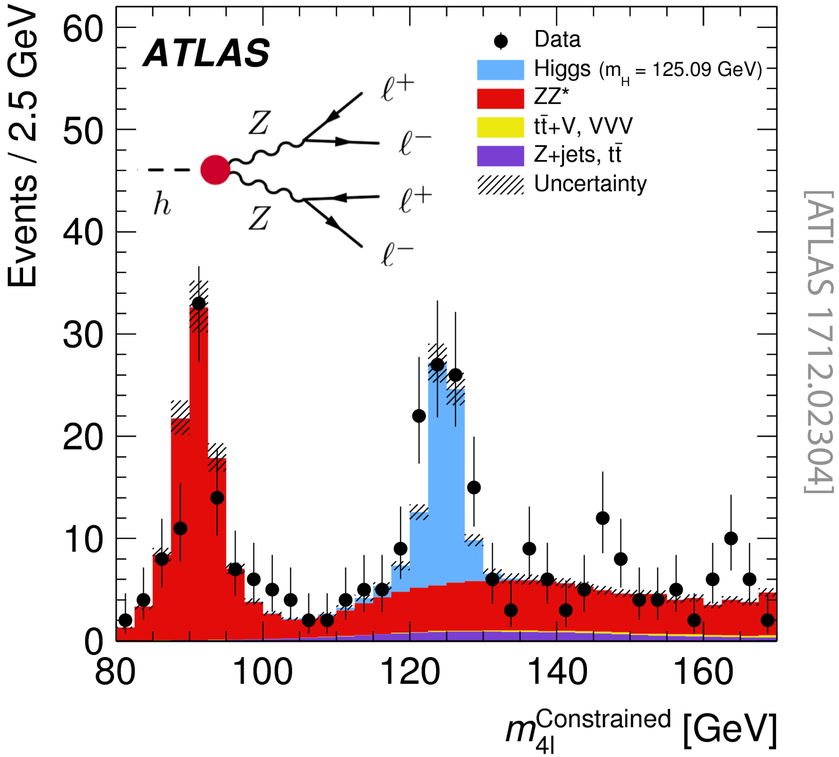


High-dimensional event data x

$p(x|\theta)$ cannot be calculated

The traditional solution





High-dimensional event data x

$p(x|\theta)$ cannot be calculated

One or two summary statistics x'

$p(x'|\theta)$ can be estimated with histograms, KDE, …

Summary statistics for LHC measurements?

- In many LHC problems there is no single good summary statistics: compressing to any x' loses information!

  [JB, K. Cranmer, F. Kling, T. Plehn 1612.05261; JB, F. Kling, T. Plehn, T. Tait 1712.02350]

- Ideally: analyze all trustworthy high-level features (reconstructed four-momenta…), or some form of low-level features, including correlations

  ("fully differential cross section")

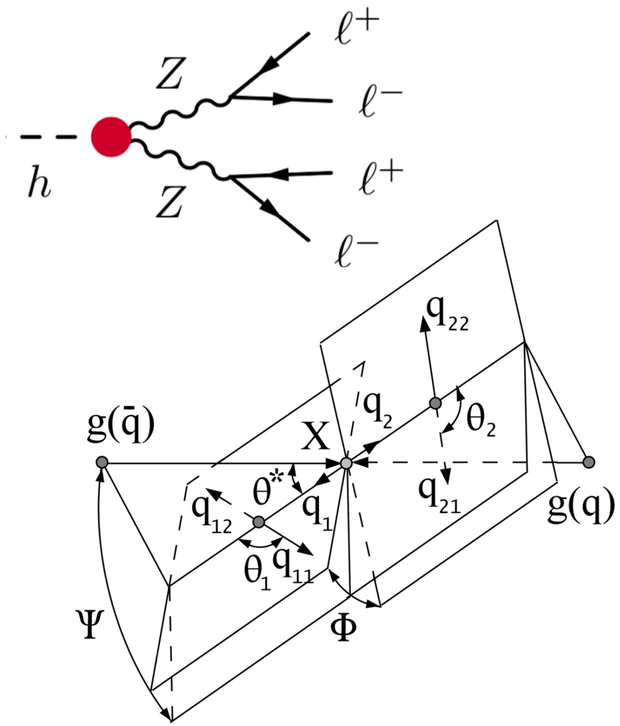
Summary statistics for LHC measurements?

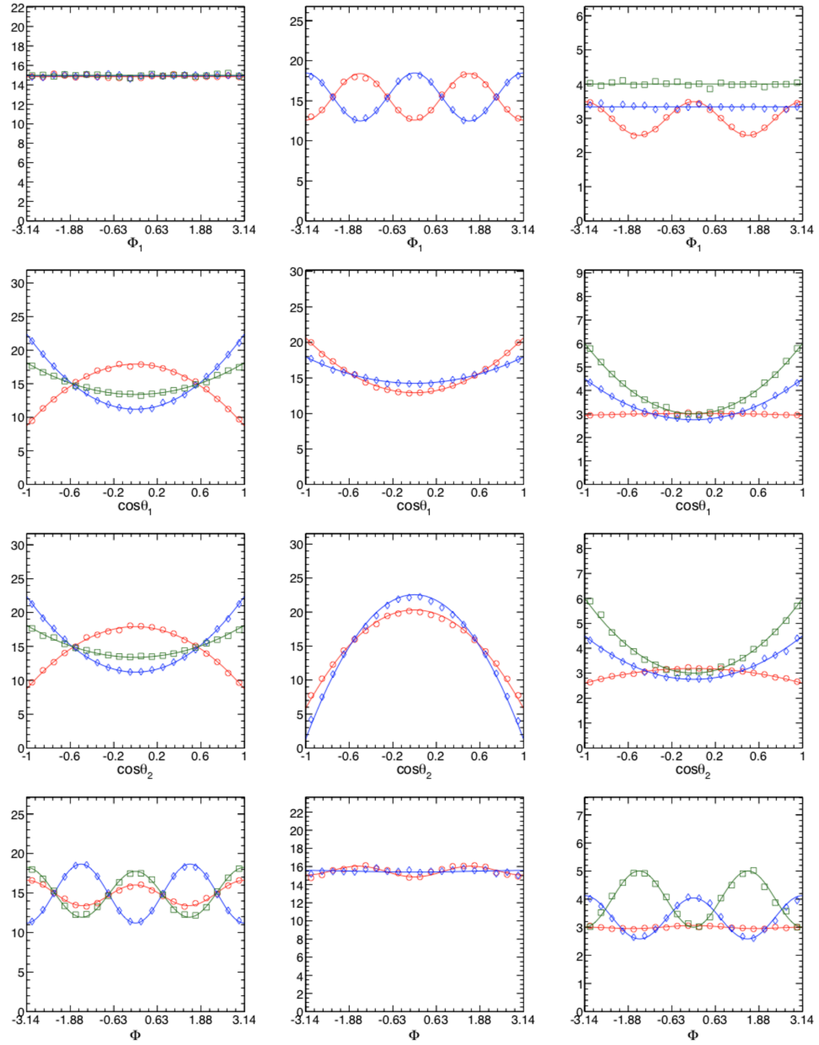
- In many LHC problems there is no single good summary statistics: compressing to any x' loses information!

  [JB, K. Cranmer, F. Kling, T. Plehn 1612.05261; JB, F. Kling, T. Plehn, T. Tait 1712.02350]

- Ideally: analyze all trustworthy high-level features (reconstructed four-momenta…), or some form of low-level features, including correlations

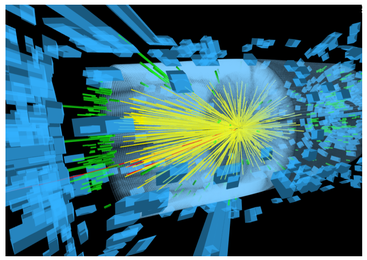
  ("fully differential cross section")



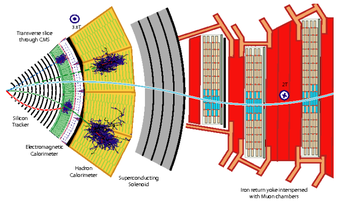


[Bolognesi et al. 1208.4018]

Sales pitch: Our SBI methods…



…leverage all the information in high-dimensional data (no need for summary statistics)

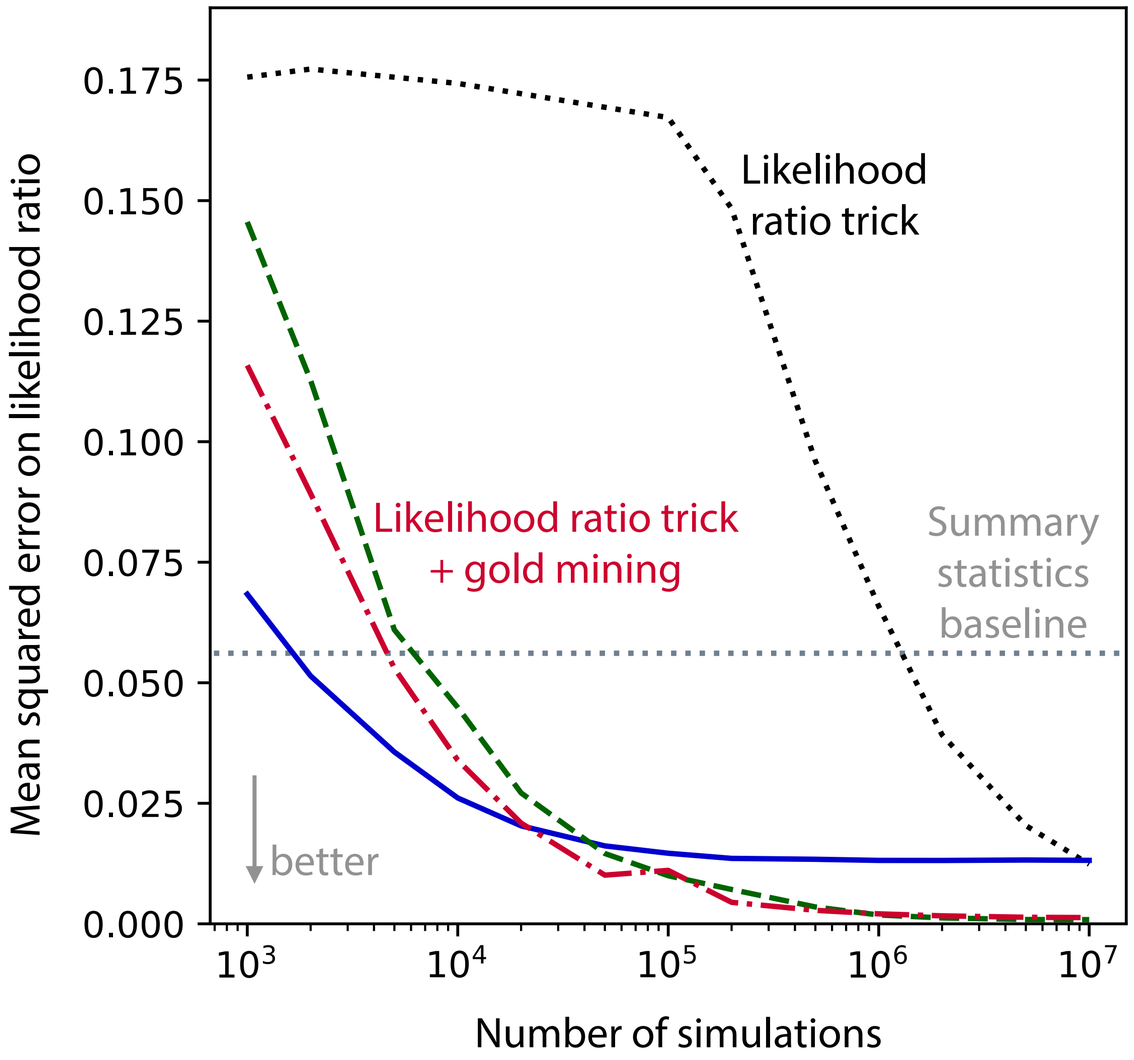


…let us use state-of-the-art shower and detector models (no transfer fns)

…can evaluate events in microseconds (amortized inference)

…need less training data than black-box ML methods (latent simulator information)

Improving sample efficiency

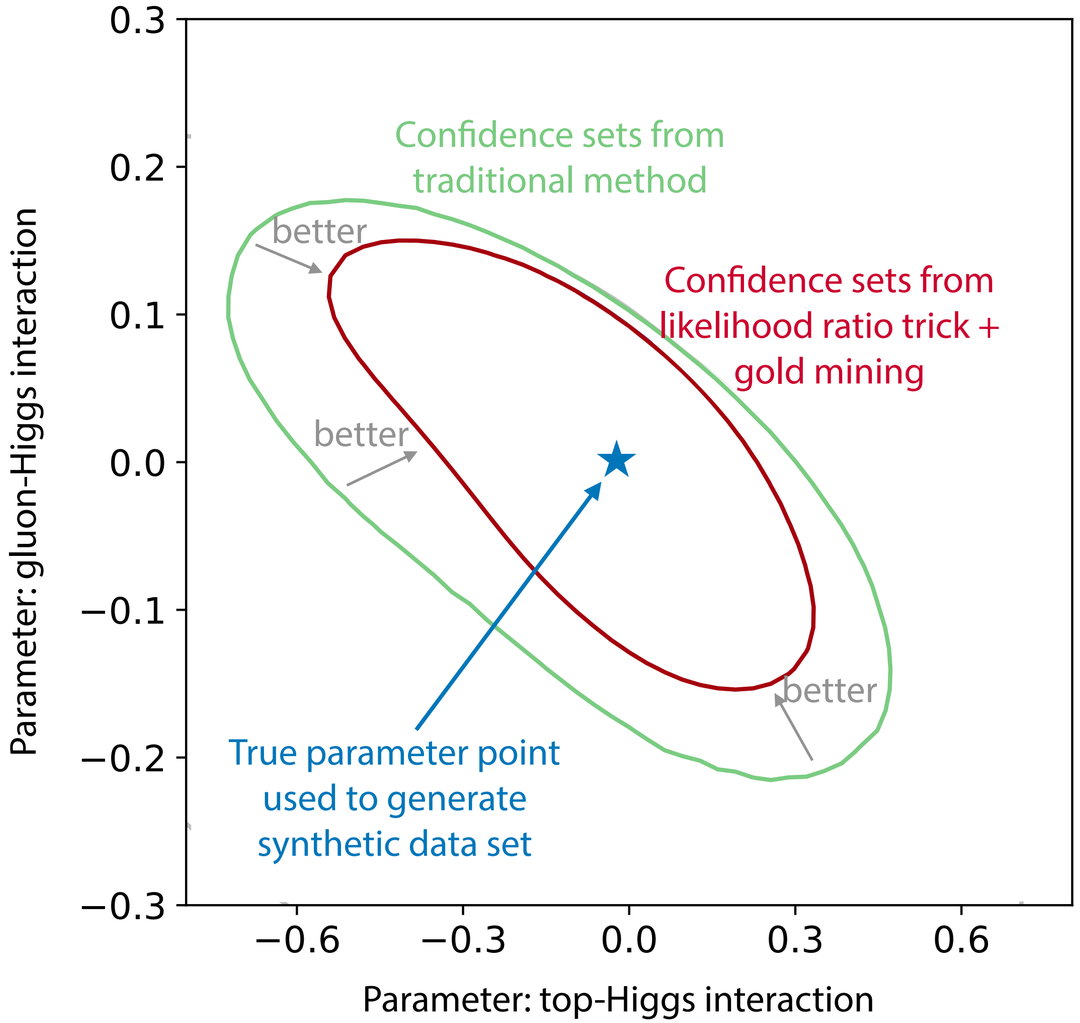


With enough training data, the ML algorithms get the likelihood function right.

Using more information from the simulator improves sample efficiency substantially.

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013; 1805.00020;
M. Stoye, JB, K. Cranmer, G. Louppe, J. Pavez 1808.00973]

Improving quality of inference



In some processes, the ML-based inference techniques improve the precision as much as taking 90% more data would!

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013; 1805.00020;
JB, F. Kling, I. Espejo, K. Cranmer 1907.10621]

Automation

[JB, F. Kling, I. Espejo, K. Cranmer 1907.10621]

Our open-source Python package MadMiner makes it straightforward to apply these ML-based inference techniques

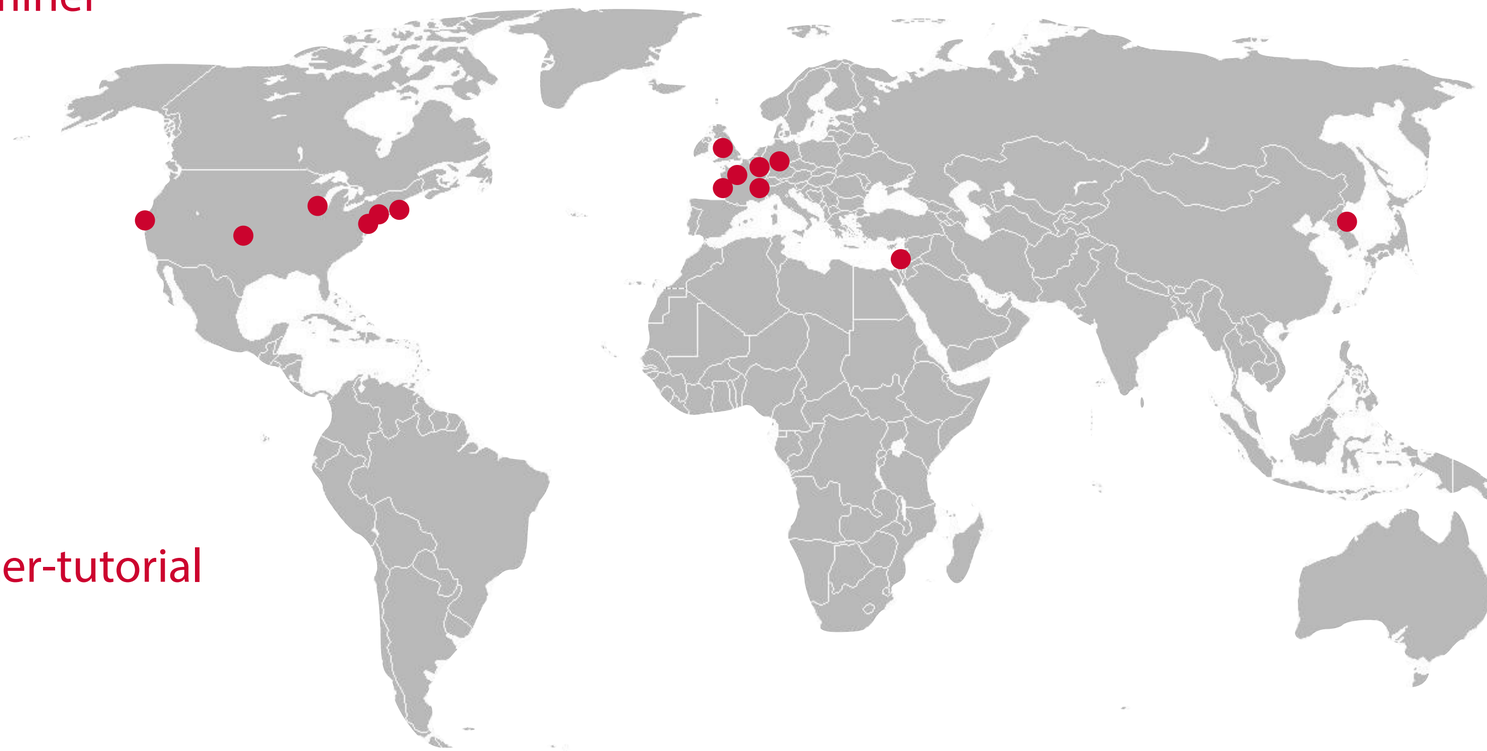
github.com/madminer-tool/madminer

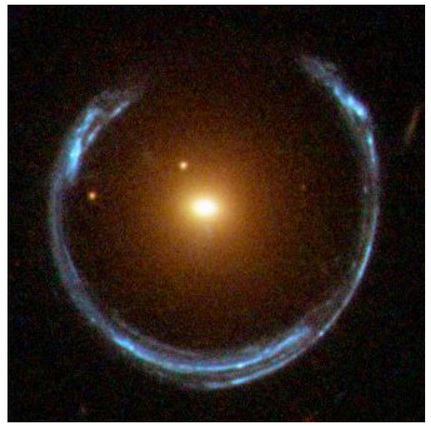
`pip install madminer`

hub.docker.com/u/madminertool

madminer-tool.github.io/madminer-tutorial

madminer.readthedocs.io

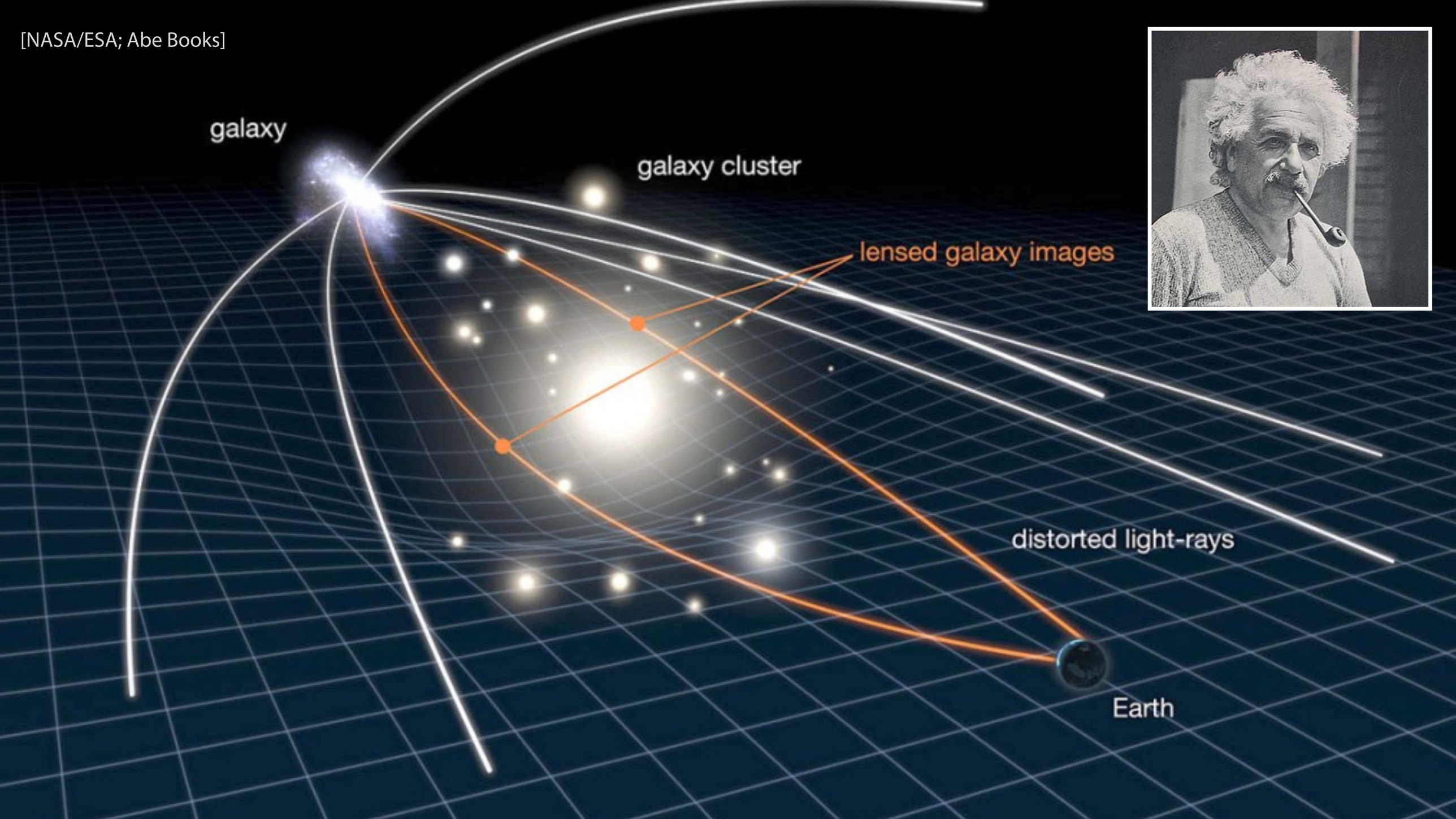


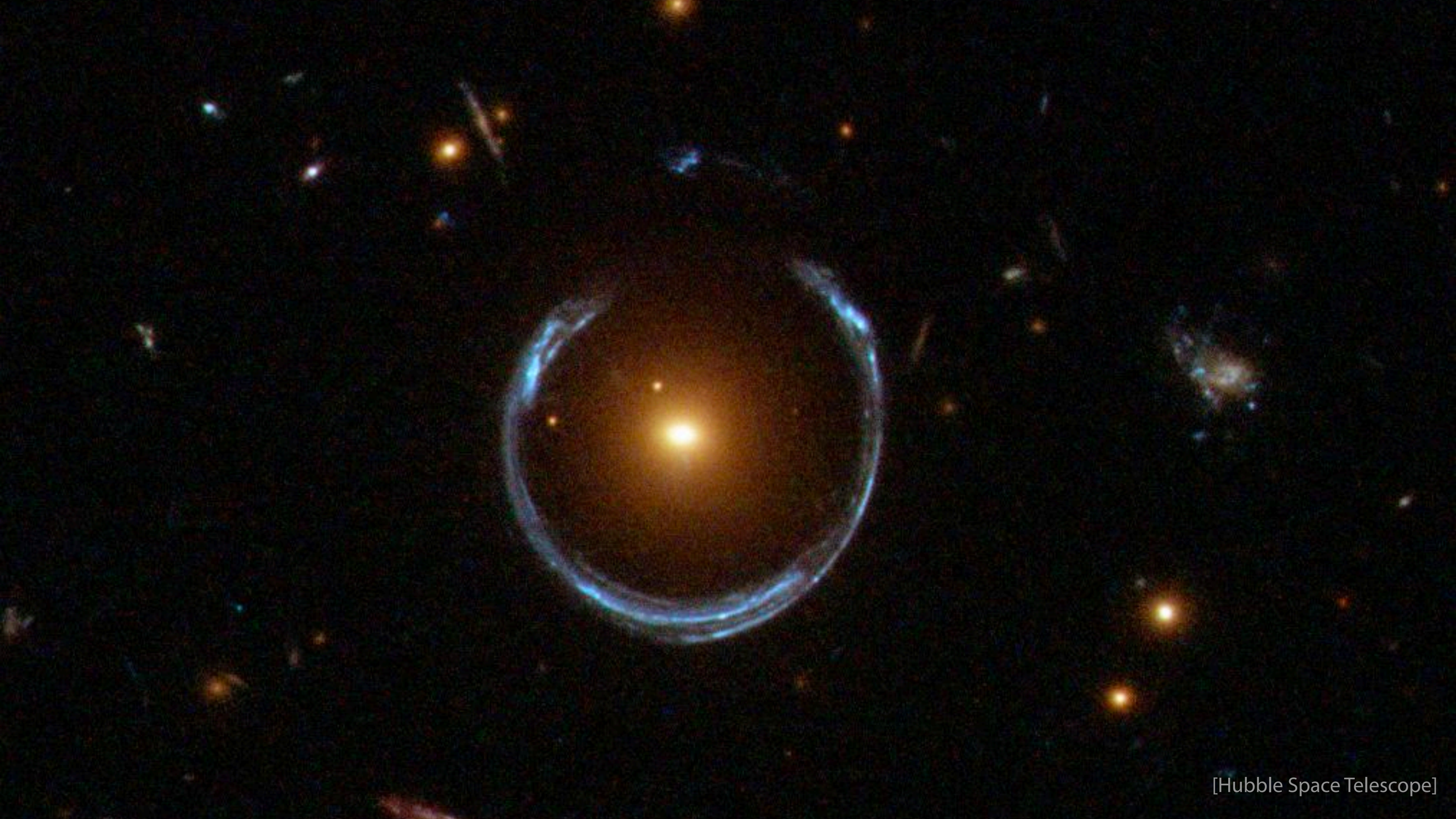


4. Astrophysics


[T. Brown, J.Tumlinson]

[NASA/ESA; Abe Books]

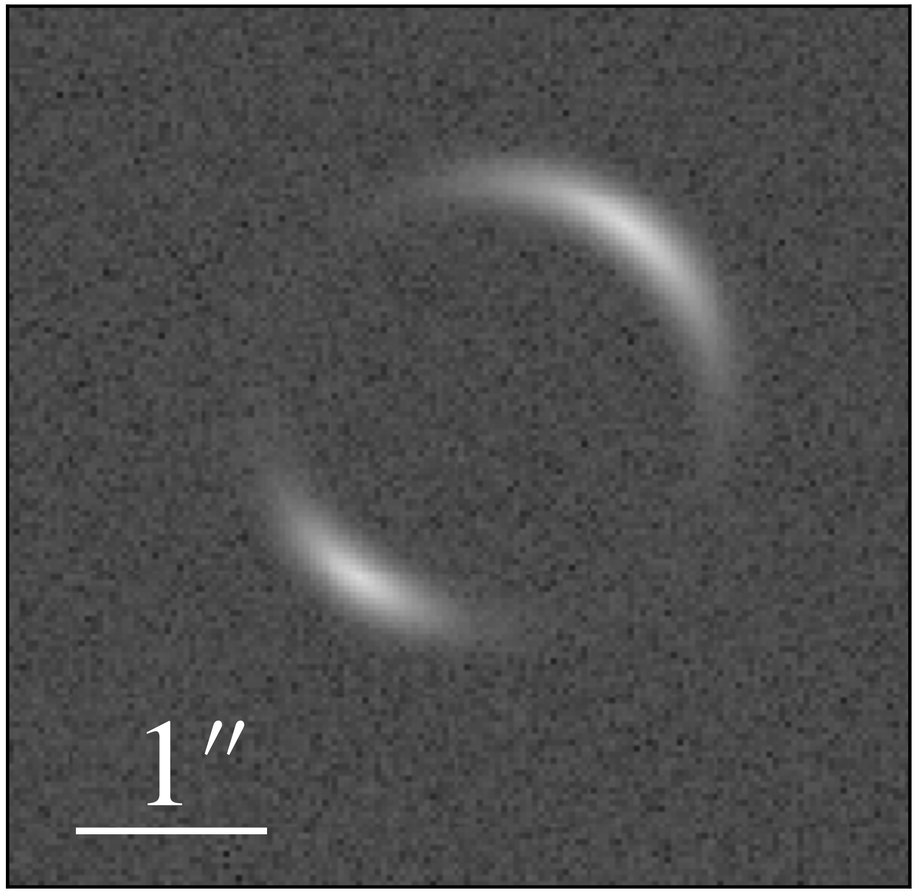




[Hubble Space Telescope]

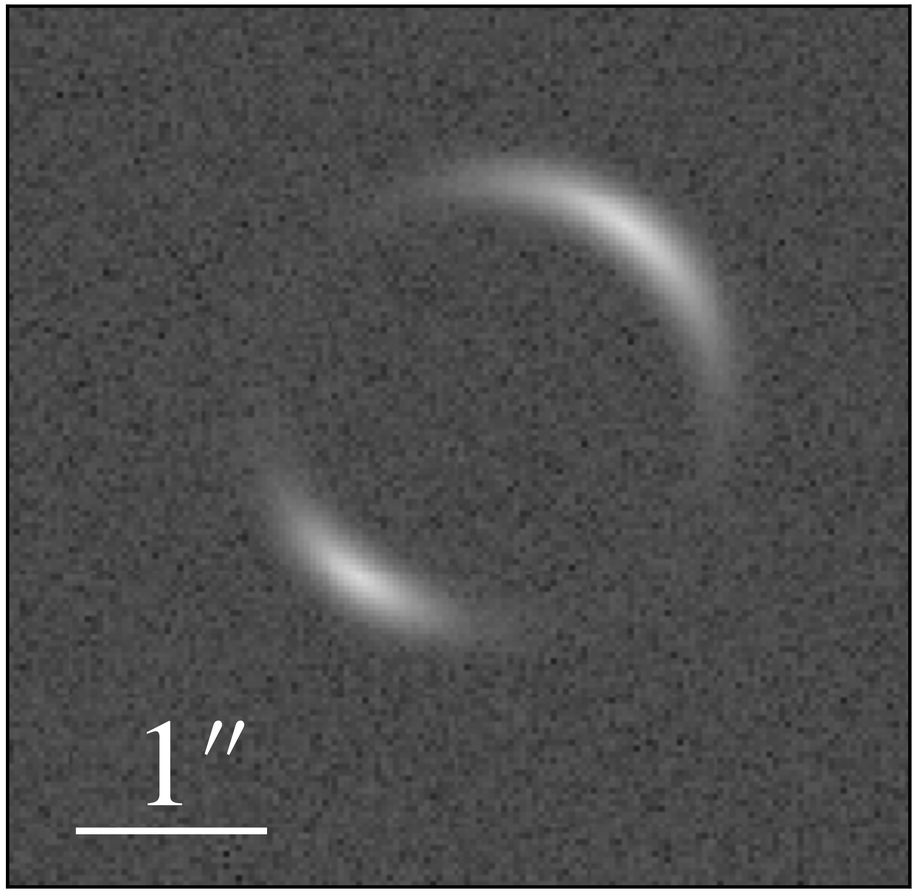
Subhalos affect strong lensing

Smooth halo only

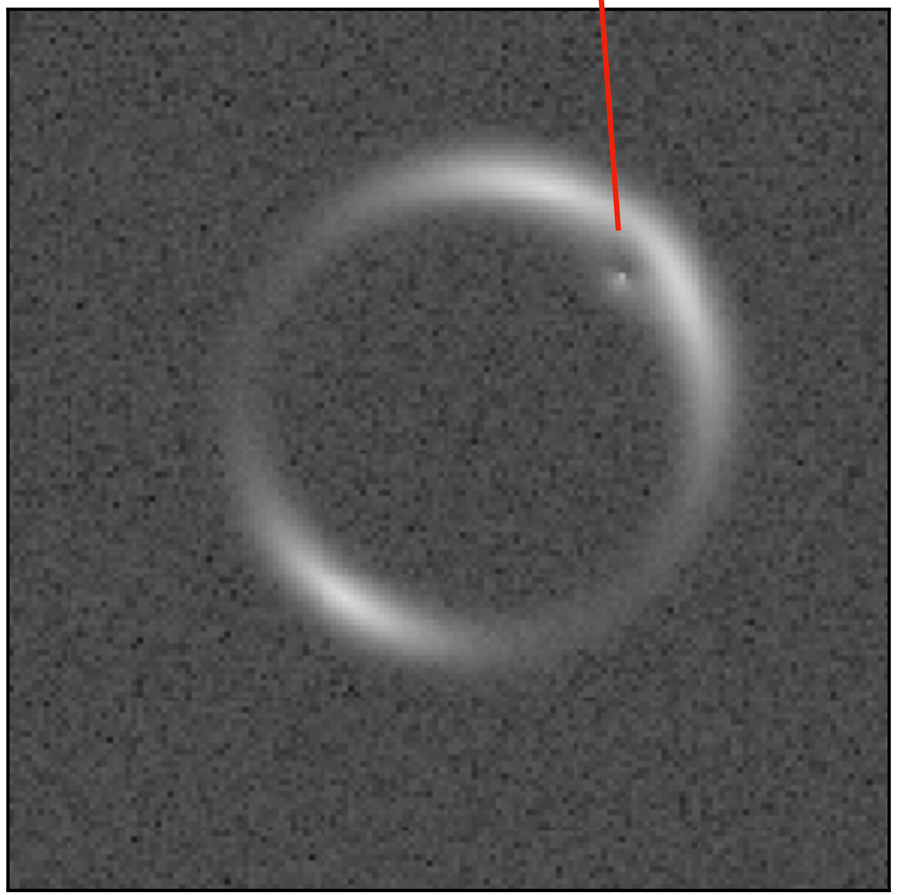


Subhalos affect strong lensing

Smooth halo only

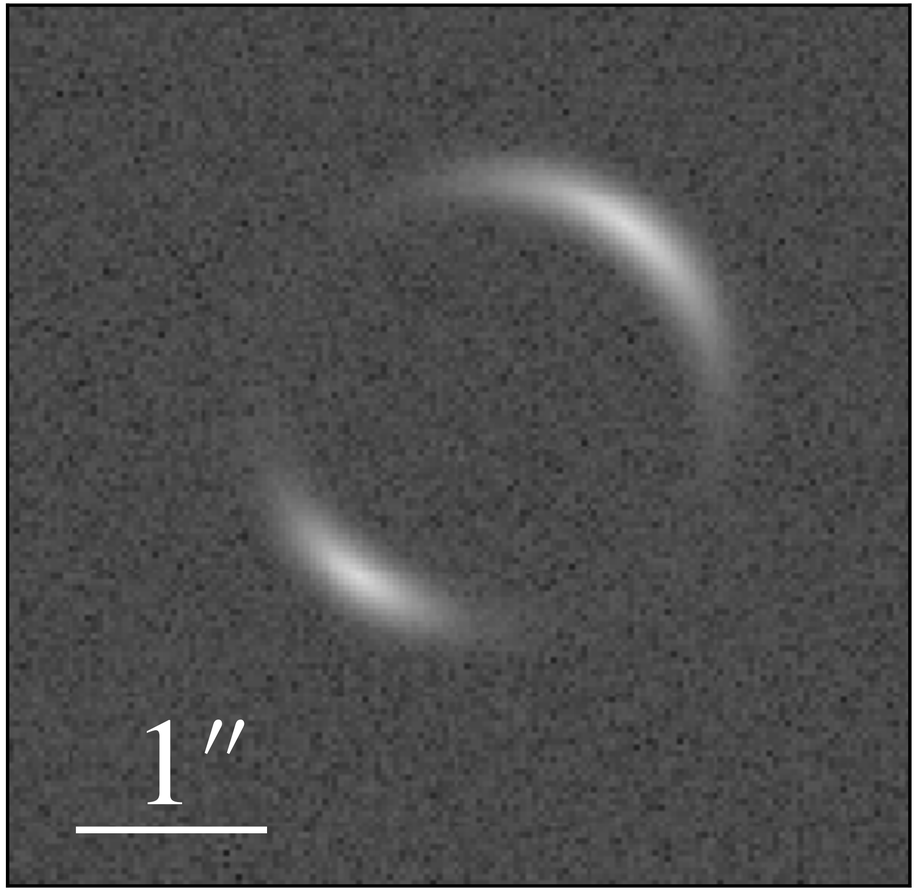


Smooth halo + subhalo

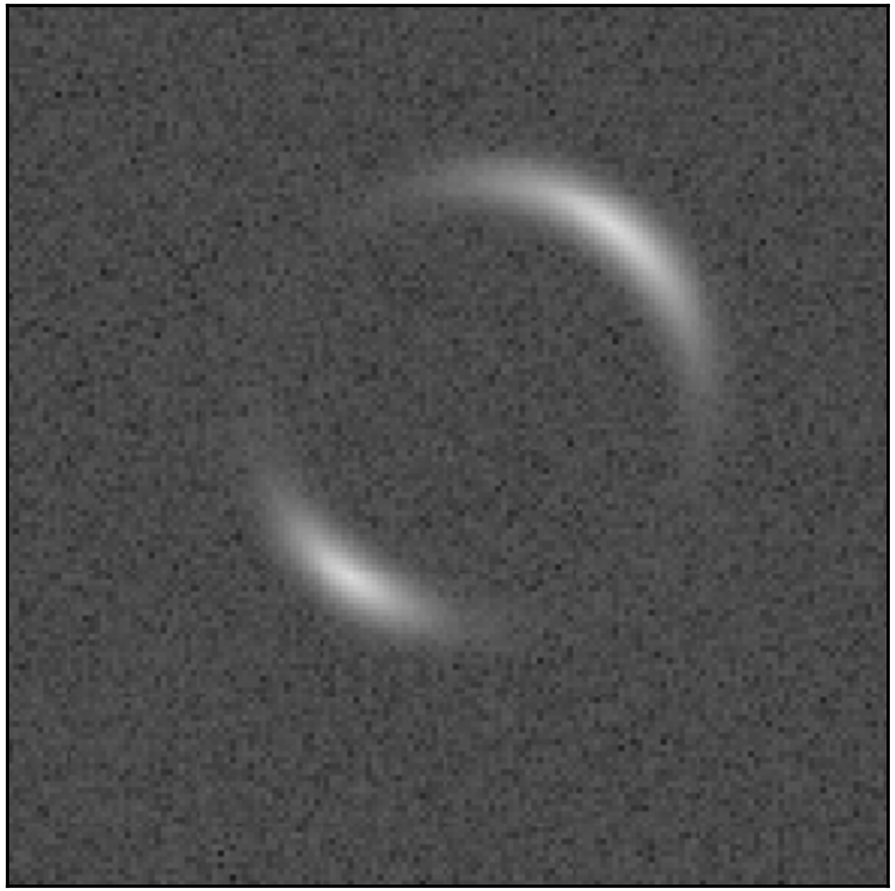


Subhalos affect strong lensing… realistically

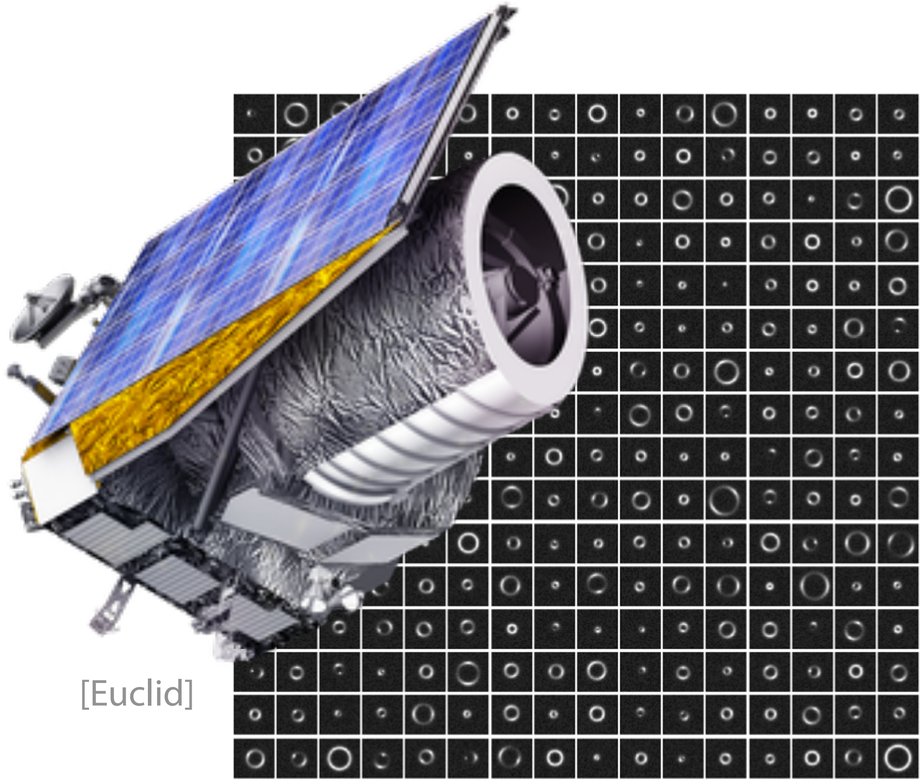
Smooth halo only



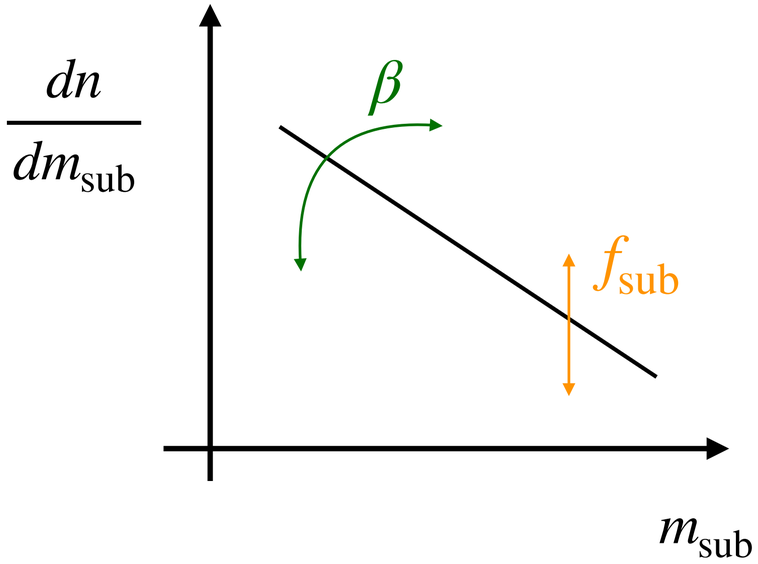
Smooth halo + subhalos



Scalable inference for small subhalos



[Euclid]

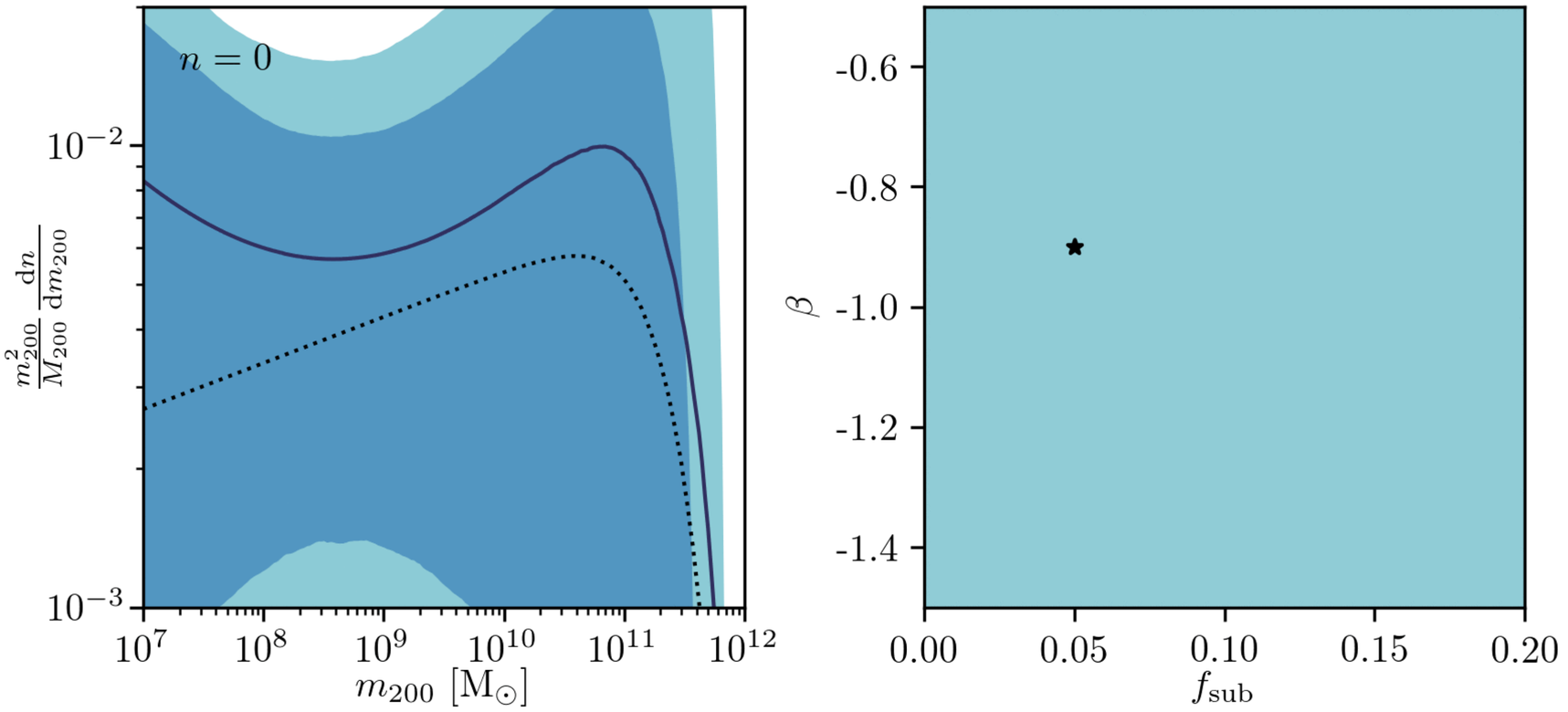


Near-future telescopes and satellites will collect hundreds of lensing images [Collett et al 1507.02657]

Goal: infer DM properties from all images and all clumps at once

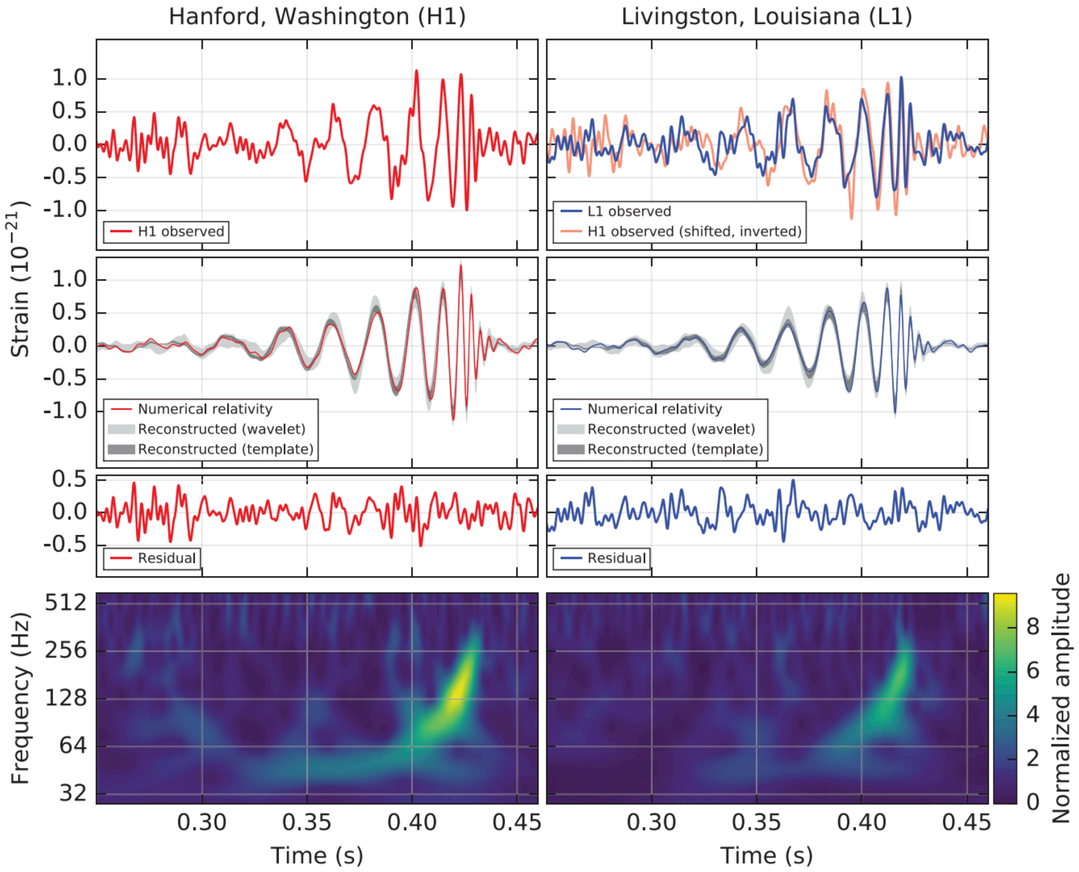
ML-based Bayesian inference

[JB, S. Mishra-Sharma, J. Hermans, G. Louppe, K. Cranmer 1909.02005]



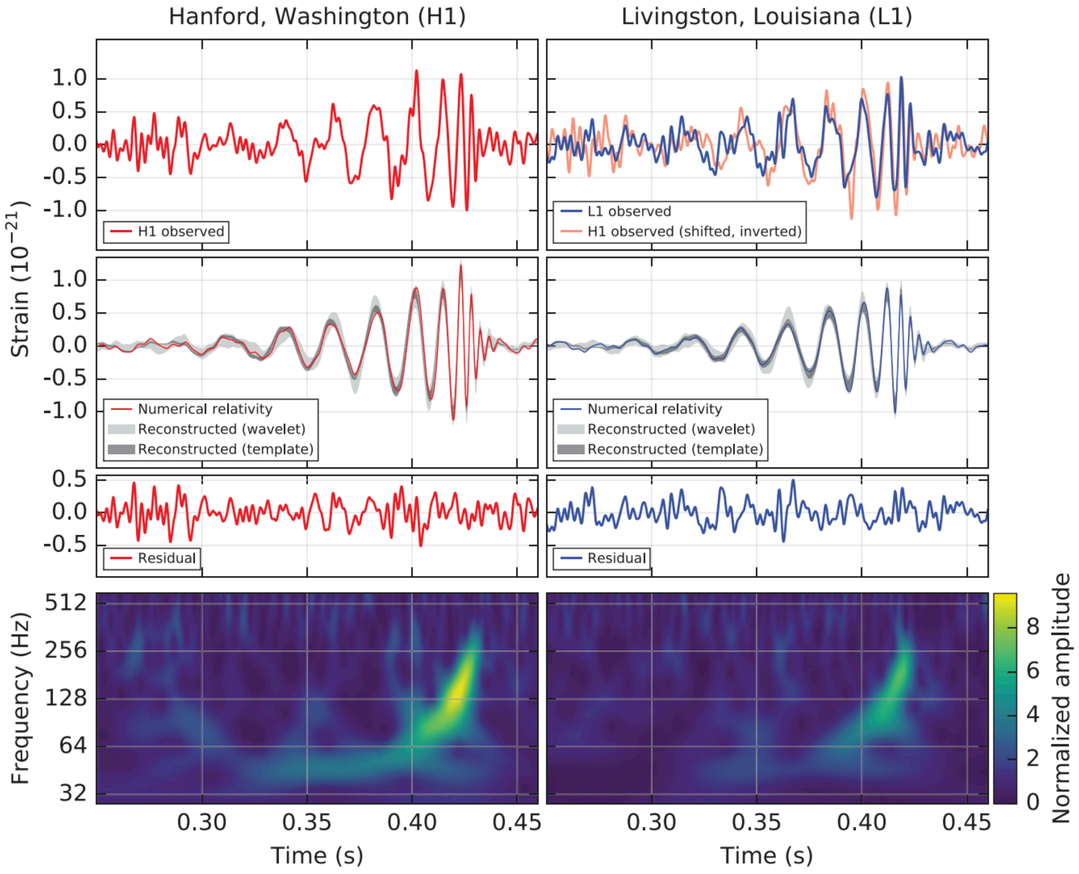
2 years later: BH merger inference

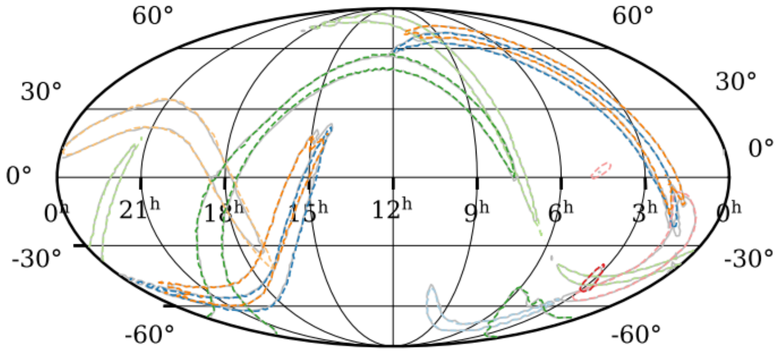
[M. Dax, S. Green, J. Gair, J. Macke, A. Buonanno, B. Schölkopf 2106.12594]

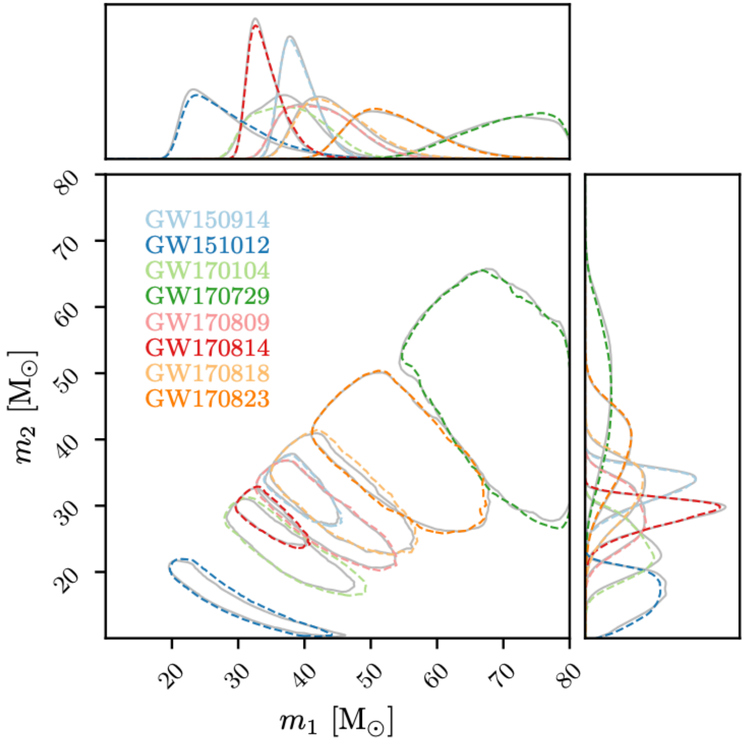


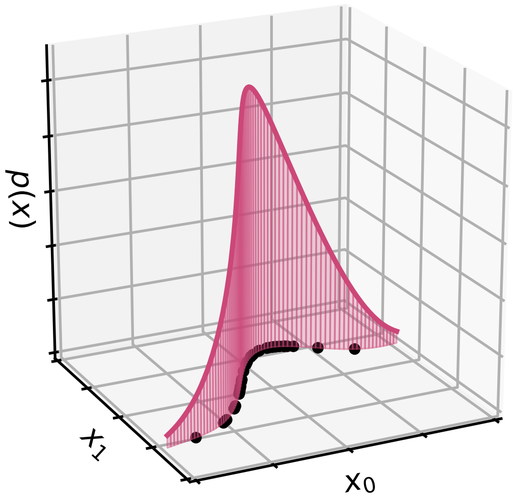
2 years later: BH merger inference

[M. Dax, S. Green, J. Gair, J. Macke, A. Buonanno, B. Schölkopf 2106.12594]









5. Tangents

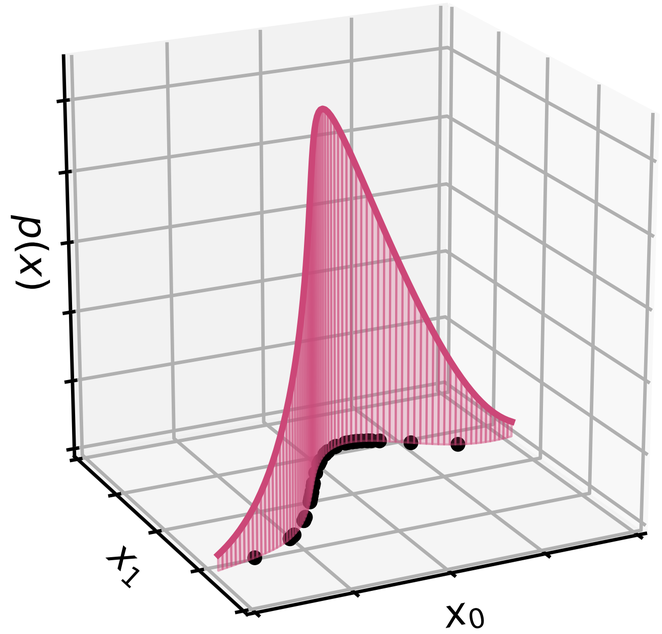
$\mathcal{M}$-flows

[JB, K. Cranmer 2003.13913; see also e.g. S. Klein et al, 2112.08069]

Often data is restricted to a lower-dimensional manifold embedded in the data space

$\mathcal{M}$-flows are a new probabilistic / generative model that

- describe data as a tractable probability density on a lower-dimensional manifold
- learn manifold and density from data



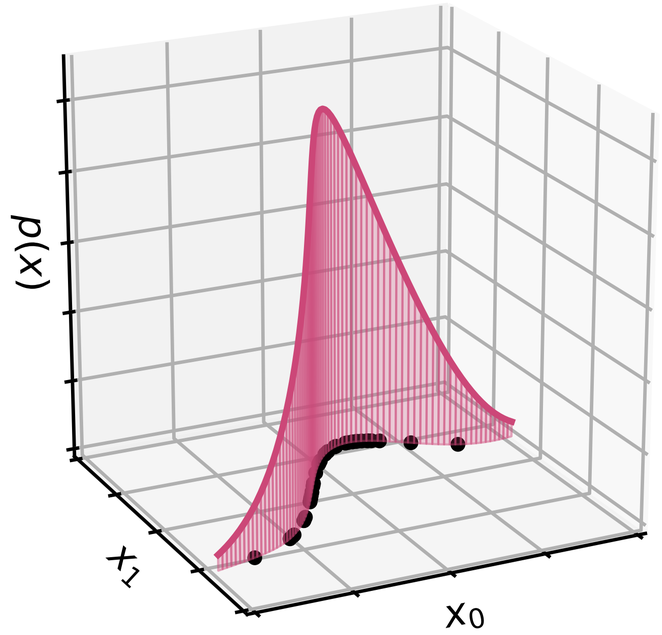
$\mathcal{M}$-flows

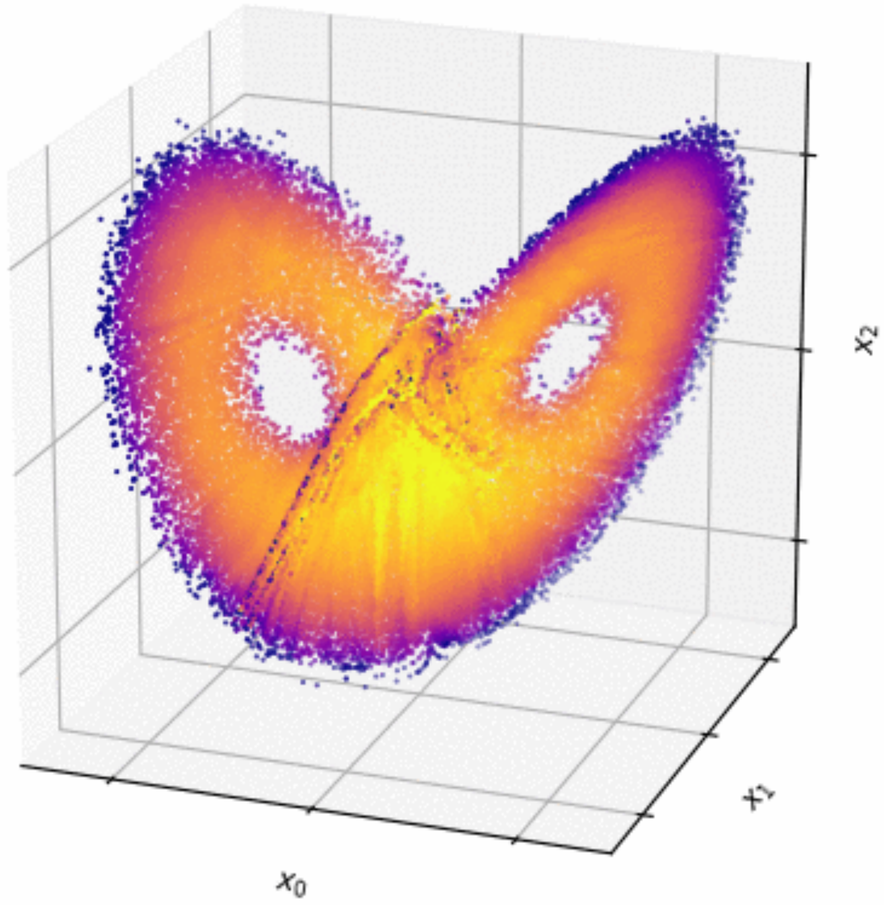
[JB, K. Cranmer 2003.13913; see also e.g. S. Klein et al, 2112.08069]

Often data is restricted to a lower-dimensional manifold embedded in the data space

$\mathcal{M}$-flows are a new probabilistic / generative model that

- describe data as a tractable probability density on a lower-dimensional manifold
- learn manifold and density from data





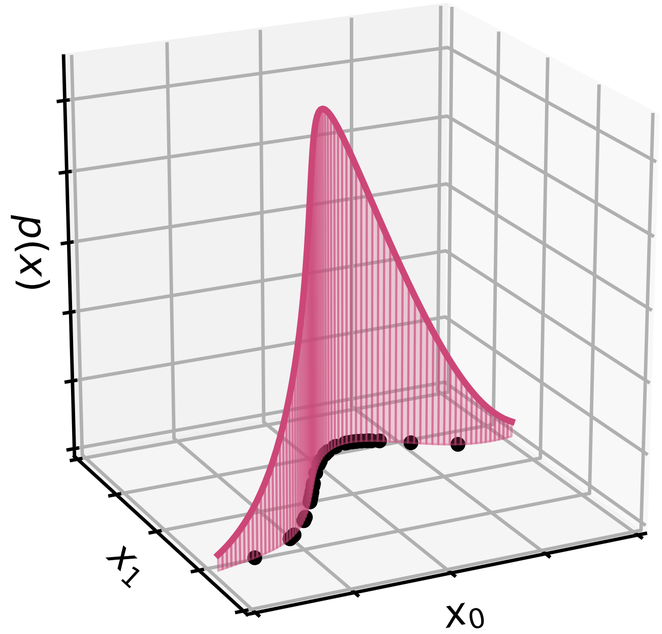
$\mathcal{M}$-flows

[JB, K. Cranmer 2003.13913; see also e.g. S. Klein et al, 2112.08069]

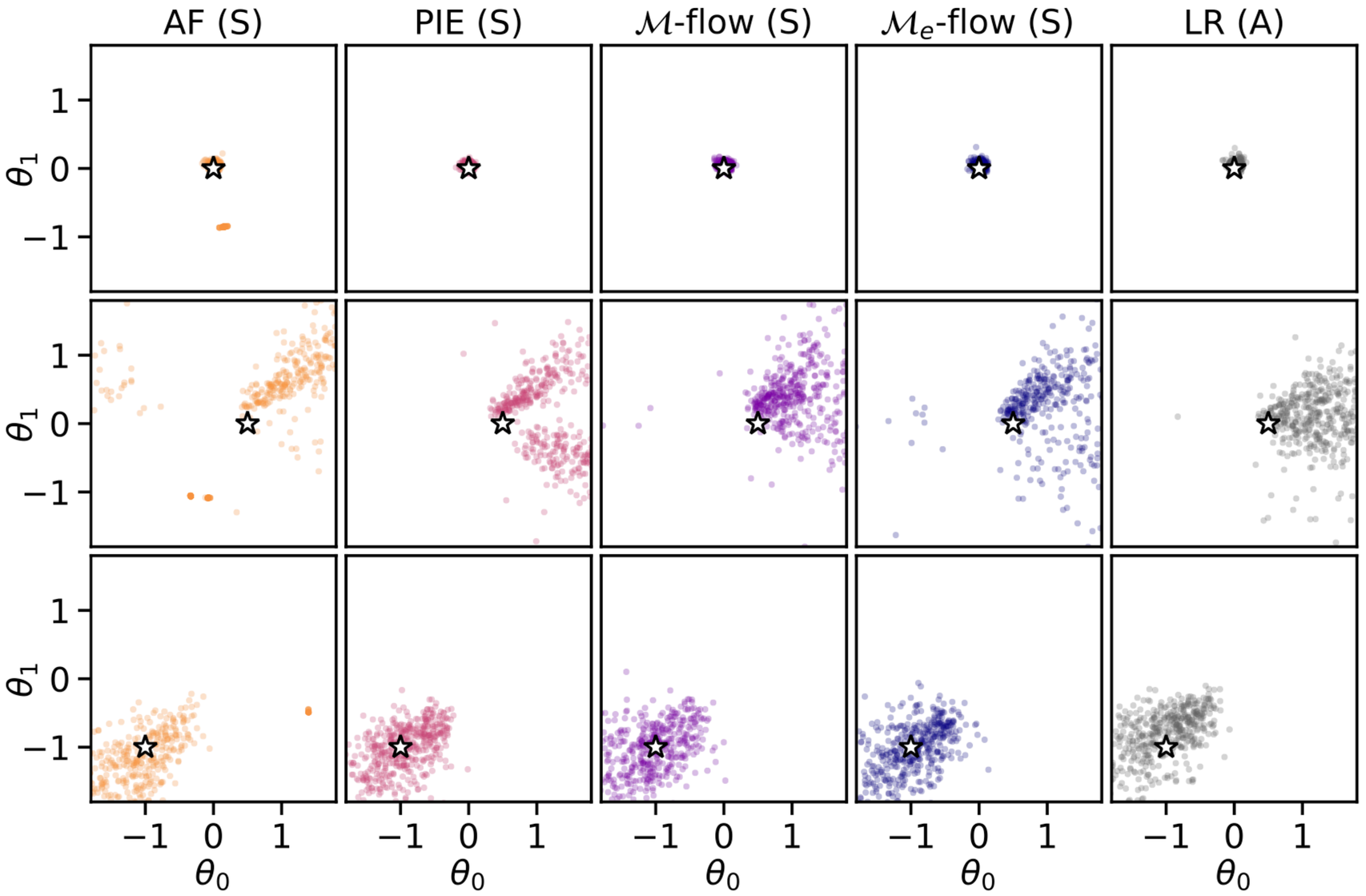
Often data is restricted to a lower-dimensional manifold embedded in the data space

$\mathcal{M}$-flows are a new probabilistic / generative model that

- describe data as a tractable probability density on a lower-dimensional manifold
- learn manifold and density from data

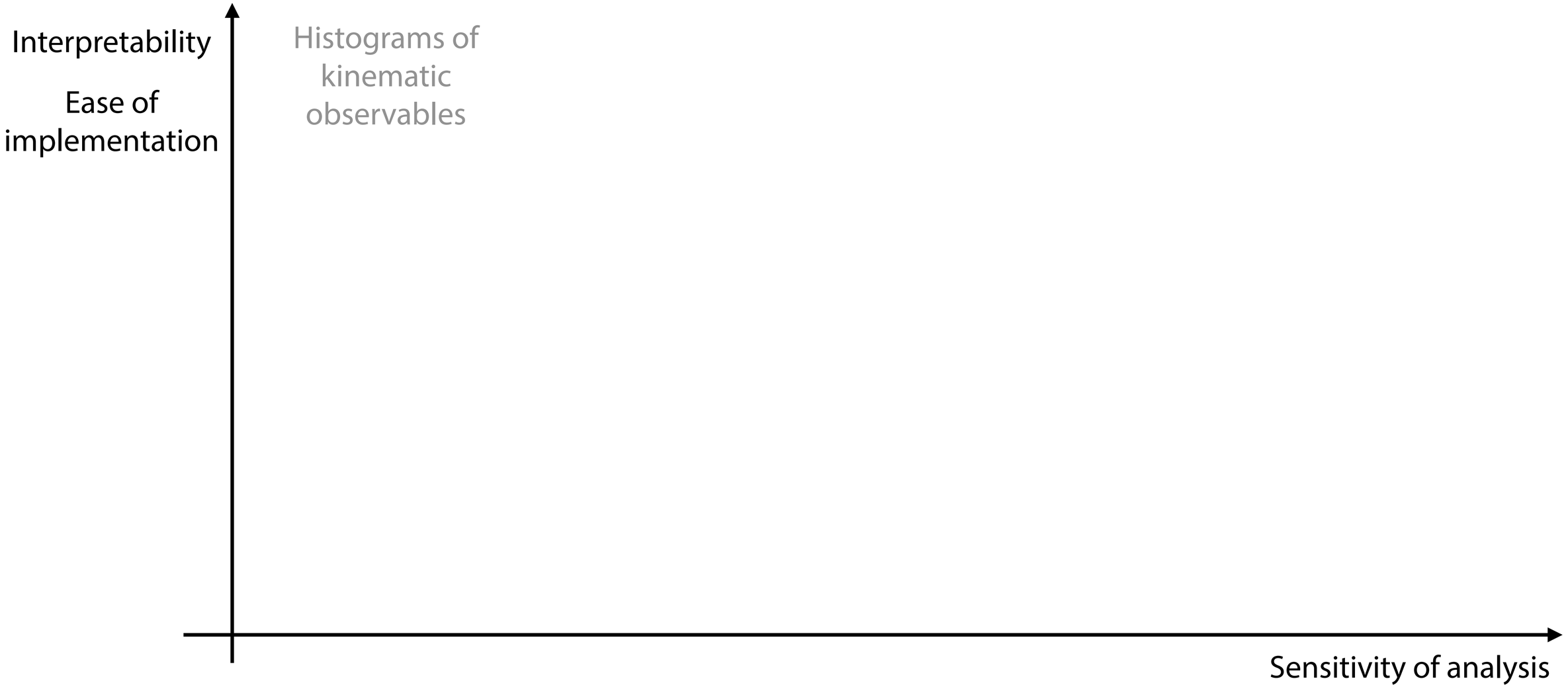


$\mathcal{M}$-flows in simulation-based inference



Learning optimal summary stats

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013, 1805.00020, 1805.12244; N. Soybelmann, A. Butter, T. Plehn, JB 2109.10414]

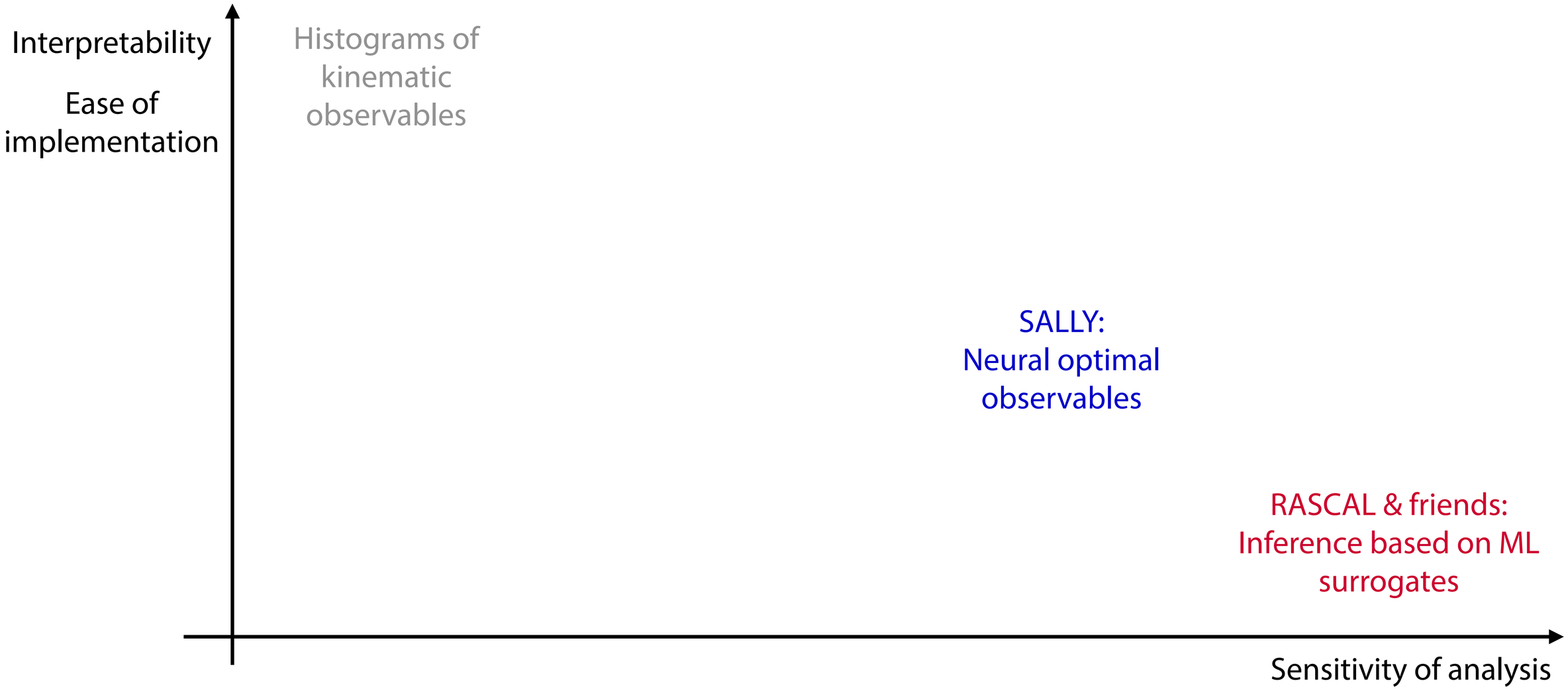


Learning optimal summary stats

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013, 1805.00020, 1805.12244; N. Soybelmann, A. Butter, T. Plehn, JB 2109.10414]

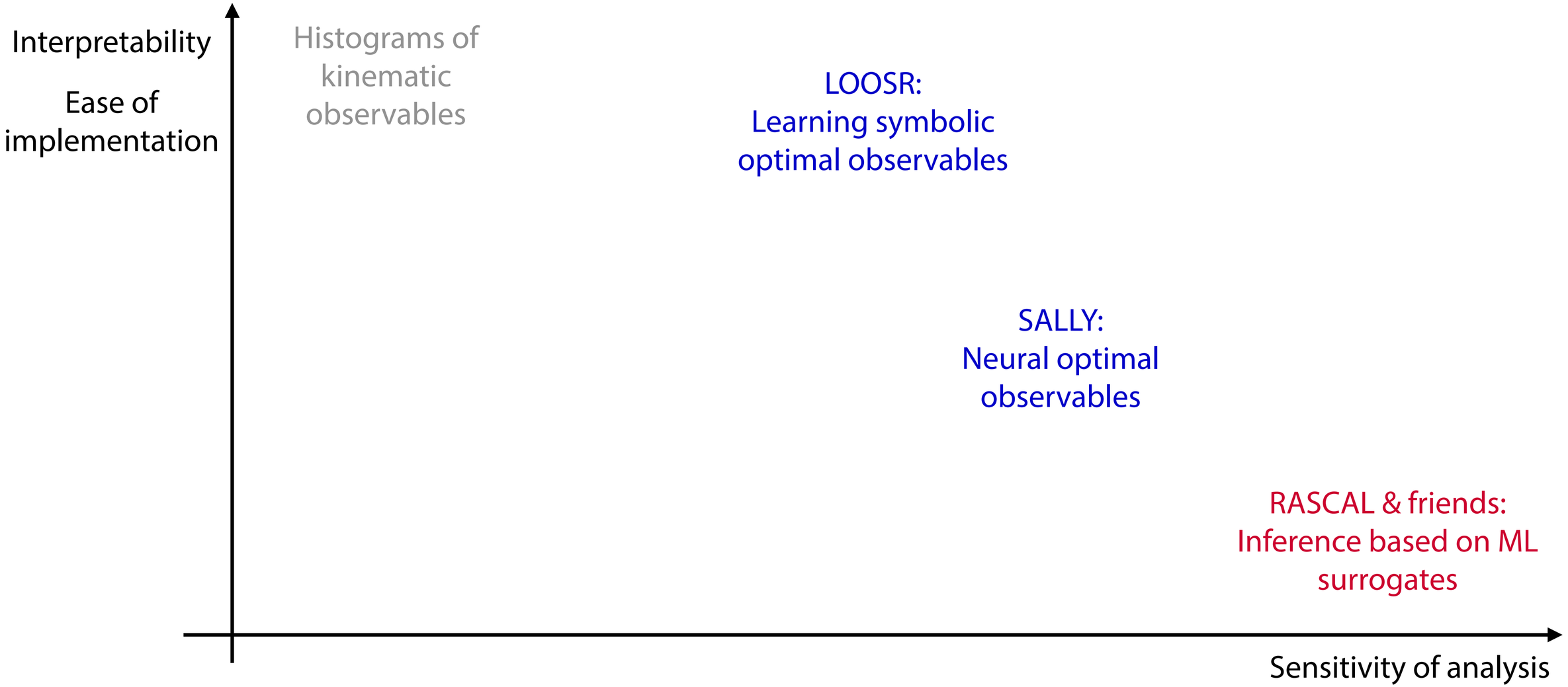
Learning optimal summary stats

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013, 1805.00020, 1805.12244; N. Soybelmann, A. Butter, T. Plehn, JB 2109.10414]



Learning optimal summary stats

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013, 1805.00020, 1805.12244; N. Soybelmann, A. Butter, T. Plehn, JB 2109.10414]



Learning optimal summary stats

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013, 1805.00020, 1805.12244; N. Soybelmann, A. Butter, T. Plehn, JB 2109.10414]

- Rather than learning the likelihood ratio, can we just use ML to find better observables?
  - Holy grail: sufficient statistics (contain all information in data on the theory parameters)
  - Histogram analysis of sufficient statistics will give best possible sensitivity

Learning optimal summary stats

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013, 1805.00020, 1805.12244; N. Soybelmann, A. Butter, T. Plehn, JB 2109.10414]

- Rather than learning the likelihood ratio, can we just use ML to find better observables?
  - Holy grail: sufficient statistics (contain all information in data on the theory parameters)
  - Histogram analysis of sufficient statistics will give best possible sensitivity

- In the neighborhood of a reference point $\theta_{\rm ref}$ (e.g. the SM), the sufficient statistics are just the score

$$t(x) = \nabla_\theta \log p(x|\theta)\Big|_{\theta_{\rm ref}}$$

Learning optimal summary stats

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013, 1805.00020, 1805.12244; N. Soybelmann, A. Butter, T. Plehn, JB 2109.10414]

- Rather than learning the likelihood ratio, can we just use ML to find better observables?
  - Holy grail: sufficient statistics (contain all information in data on the theory parameters)
  - Histogram analysis of sufficient statistics will give best possible sensitivity

- In the neighborhood of a reference point $\theta_{\rm ref}$ (e.g. the SM), the sufficient statistics are just the score

$$t(x) = \nabla_\theta \log p(x|\theta)\Big|_{\theta_{\rm ref}}$$

- We can machine-learn the score from simulated data!

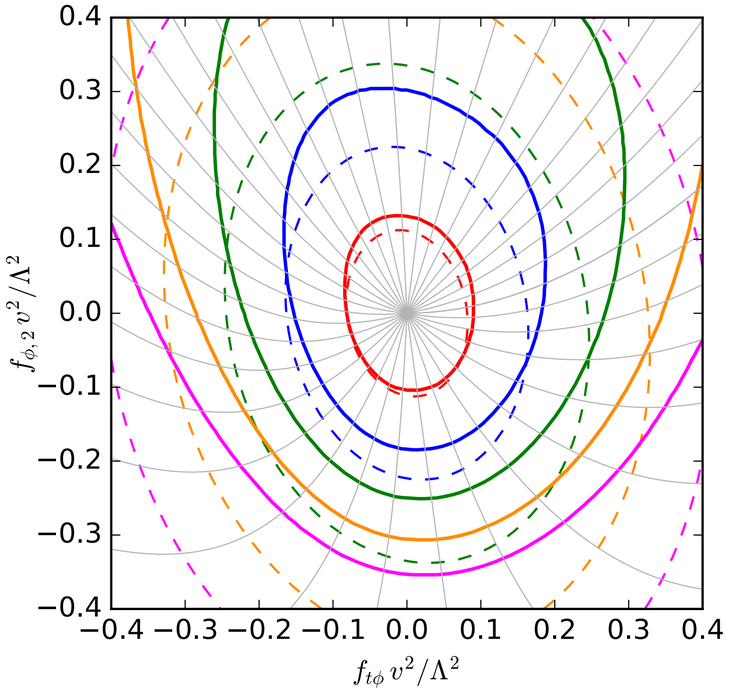
$$\hat{t}(x) = \arg\min_{g(x)} \mathbb{E}_{x,z} \Big| g(x) - \underbrace{t(x,z)}_{\sim \frac{\nabla_\theta \mathcal{M}^2(z|\theta_{\rm ref})}{\mathcal{M}^2(z|\theta_{\rm ref})}} \Big|^2$$

- SALLY: neural networks
  (straightforward to train, highly performant)
- LOOSR: symbolic expressions like "$p_{T1}\, p_{T2}\, \sin\Delta\phi_{jj}$"
  (physicist-interpretable, easy to implement)

Information geometry

[JB, K. Cranmer, F. Kling, T. Plehn 1612.05261;
JB, F. Kling, T. Plehn, T. Tait 1712.02350]

Study manifold of probability distributions geometrically

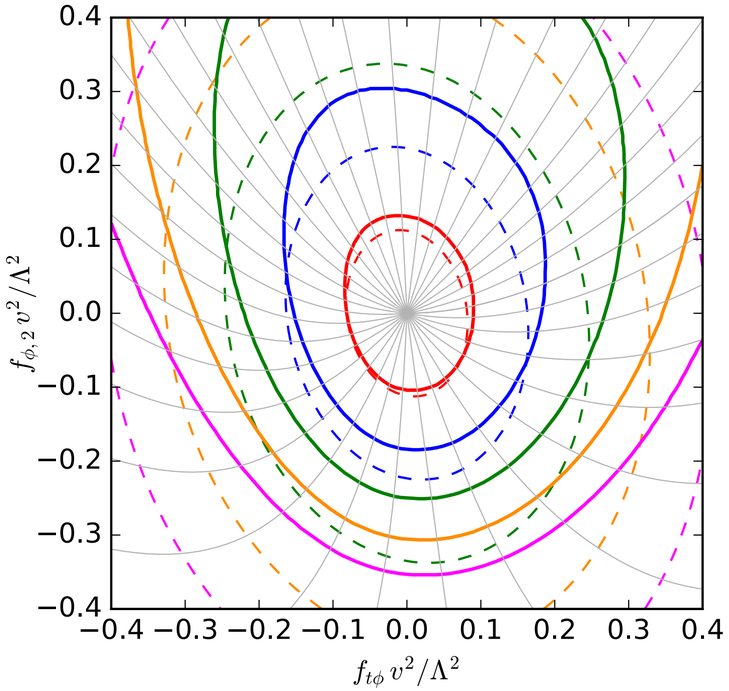


Distances ↔ sensitivity with which an experiment can distinguish parameter points

Information geometry

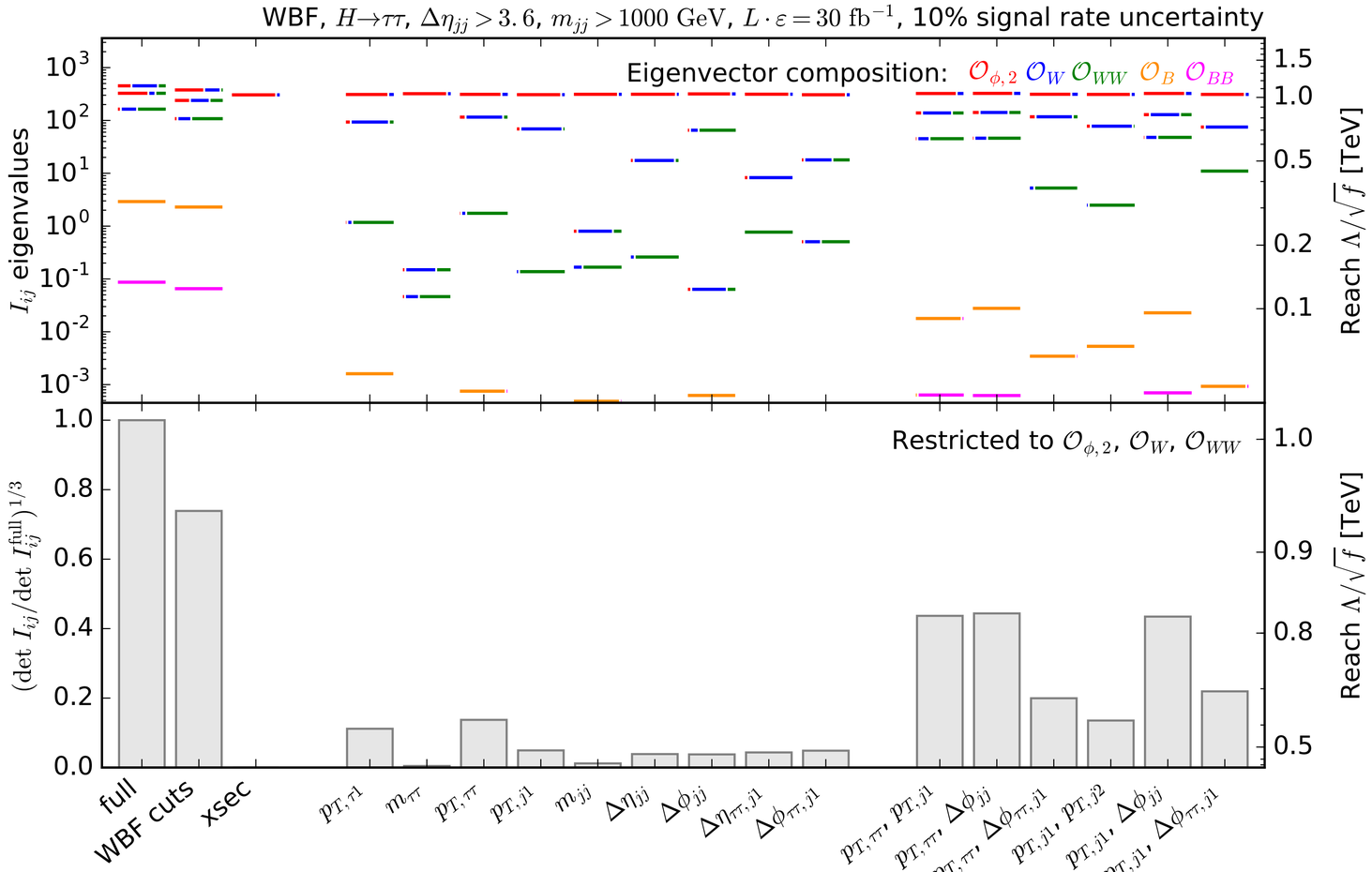
[JB, K. Cranmer, F. Kling, T. Plehn 1612.05261;
JB, F. Kling, T. Plehn, T. Tait 1712.02350]

Study manifold of probability distributions geometrically



Distances ↔ sensitivity with which an experiment can distinguish parameter points

In practice, we can use this for sensitivity forecasting, feature selection, experimental design



Kyle Cranmer

Gilles Louppe

Juan Pavez

Markus Stoye

Felix Kling

Irina Espejo

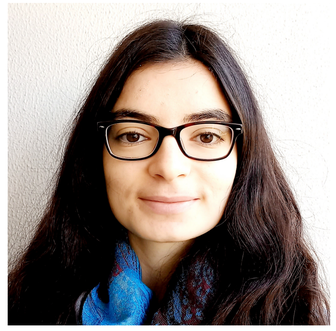
Sinclert Perez

Sid Mishra-Sharma

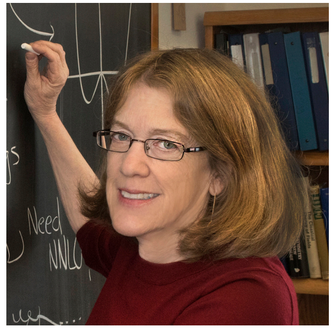
Zubair Bhatti

Tilman Plehn

Anja Butter



Nathalie Soybelman



Sally Dawson

Sam Homiller

Parts of this talk were inspired by great presentations by Kyle Cranmer, Gilles Louppe, Sid Mishra-Sharma, and Jakob Macke

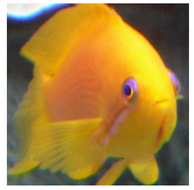
NYU

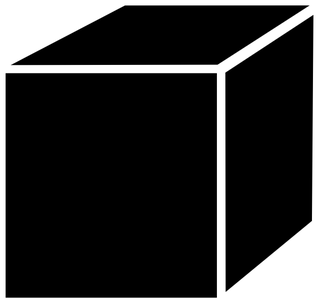
MSDSE
Moore - Sloan
Data Science Environments

diana hep

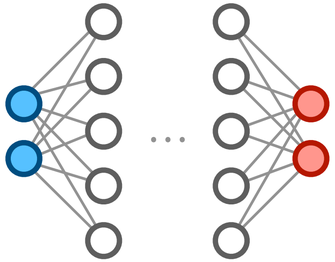
iris hep
Institute for Research and Innovation
in Software for High Energy Physics

The SCAILFIN Project
scailfin.github.io

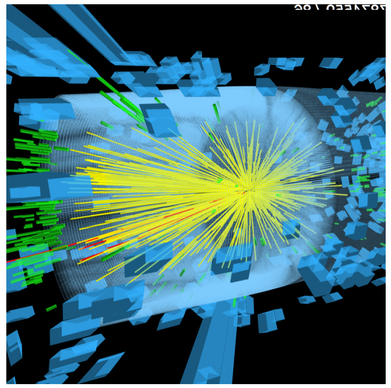




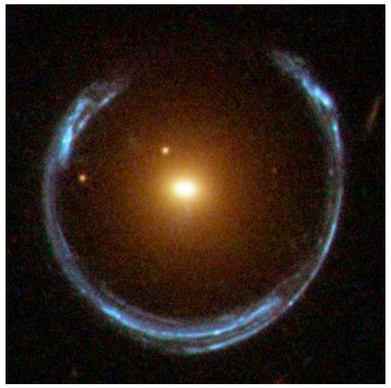
Simulators make precise predictions, but inference is challenging.



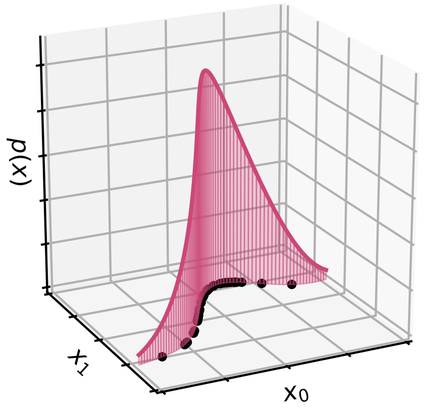
Machine learning enables powerful inference methods, especially when we inject more information.



They work in problems from the smallest…



… to the largest scales.



Further ML advances will translate into scientific progress.

Selected references

Review of simulation-based inference
K. Cranmer, JB, G. Louppe:
"The frontier of simulation-based inference"
PNAS, 1911.01429

Machine learning–based inference methods
JB, G. Louppe, J. Pavez, K. Cranmer:
"Mining gold from implicit models to improve likelihood-free inference"
PNAS, 1805.12244

M. Stoye, JB, K. Cranmer, G. Louppe, J. Pavez:
"Likelihood-free inference with an improved cross-entropy estimator"
NeurIPS workshop, 1808.00973

Generative models
JB and K. Cranmer:
"Flows for simultaneous manifold learning and density estimation"
NeurIPS, 2003.13913

SBI in Astrophysics
JB, S. Mishra-Sharma, J. Hermans, G. Louppe, K. Cranmer
"Mining for Dark Matter Substructure: Inferring subhalo population properties from strong lenses with machine learning"
ApJ, 1909.02005

SBI in particle physics
JB, K. Cranmer:
"Simulation-based inference methods for particle physics"
Book chapter in "Artificial Intelligence for Particle Physics", 2010.06439

JB, K. Cranmer, G. Louppe, J. Pavez:
"Constraining Effective Field Theories with machine learning"
PRL, 1805.00013

JB, K. Cranmer, G. Louppe, J. Pavez:
"A guide to constraining Effective Field Theories with machine learning"
PRD, 1805.00020

JB, F. Kling, I. Espejo, K. Cranmer:
"MadMiner: Machine learning—based inference for particle physics"
CSBS, 1907.10621, https://github.com/diana-hep/madminer

JB, K. Cranmer, F. Kling, and T. Plehn:
"Better Higgs Measurements Through Information Geometry"
PRD, 1612.05261

A. Butter, T. Plehn, N. Soybelman, and JB:
"Stronger symbolic summary statistics for the LHC"
NeurIPS workshop