Learning to constrain new physics

Johann Brehmer
New York University

Pheno & Vino Seminar, Princeton University
August 27, 2018

LHC legacy measurements

- What's hiding in the electroweak sector?

  ⇒ precision constraints on dimension-6 EFT operators
  (or Pseudo-Observables, non-linear EFT, …)

LHC legacy measurements

- What's hiding in the electroweak sector?

  ⇒ precision constraints on dimension-6 EFT operators
  (or Pseudo-Observables, non-linear EFT, …)

- These measurements are difficult!

  1. Many parameters

$$
\begin{aligned}
S = \int \mathrm{d}^4 x \Bigg[& \mathcal{L}_{\mathrm{SM}} + \frac{f_{\phi,2}}{\Lambda^2} \frac{1}{2} \partial_\mu (\phi^\dagger \phi) \, \partial^\mu (\phi^\dagger \phi) + \frac{f_{\phi,3}}{\Lambda^2} \frac{1}{3} (\phi^\dagger \phi)^3 \\
& + \frac{f_{GG}}{\Lambda^2} (\phi^\dagger \phi) \, G^a_{\mu\nu} \, G^{\mu\nu\, a} - \frac{f_{BB}}{\Lambda^2} \frac{g'^2}{4} (\phi^\dagger \phi) \, B_{\mu\nu} \, B^{\mu\nu} - \frac{f_{WW}}{\Lambda^2} \frac{g^2}{4} (\phi^\dagger \phi) \, W^a_{\mu\nu} \, W^{\mu\nu\, a} \\
& + \frac{f_B}{\Lambda^2} \frac{\mathrm{i} g'}{2} (D^\mu \phi)^\dagger \, D^\nu \phi \, B_{\mu\nu} + \frac{f_W}{\Lambda^2} \frac{\mathrm{i} g}{2} (D^\mu \phi)^\dagger \, \sigma^a \, D^\nu \phi \, W^a_{\mu\nu} \\
& + \frac{f_\ell}{\Lambda^2} (\phi^\dagger \phi) \, \overline{L}_L \phi \, \ell_R + \frac{f_u}{\Lambda^2} (\phi^\dagger \phi) \, \overline{Q}_L \tilde{\phi} \, u_R + \frac{f_d}{\Lambda^2} (\phi^\dagger \phi) \, \overline{Q}_L \phi \, d_R \\
& + \frac{f_{G\widetilde{G}}}{\Lambda^2} (\phi^\dagger \phi) \, G^a_{\mu\nu} \, \widetilde{G}^{\mu\nu\, a} - \frac{f_{B\widetilde{B}}}{\Lambda^2} \frac{g'^2}{4} (\phi^\dagger \phi) \, B_{\mu\nu} \, \widetilde{B}^{\mu\nu} - \frac{f_{W\widetilde{W}}}{\Lambda^2} \frac{g^2}{4} (\phi^\dagger \phi) \, W^a_{\mu\nu} \, \widetilde{W}^{\mu\nu\, a} \Bigg]
\end{aligned}
$$

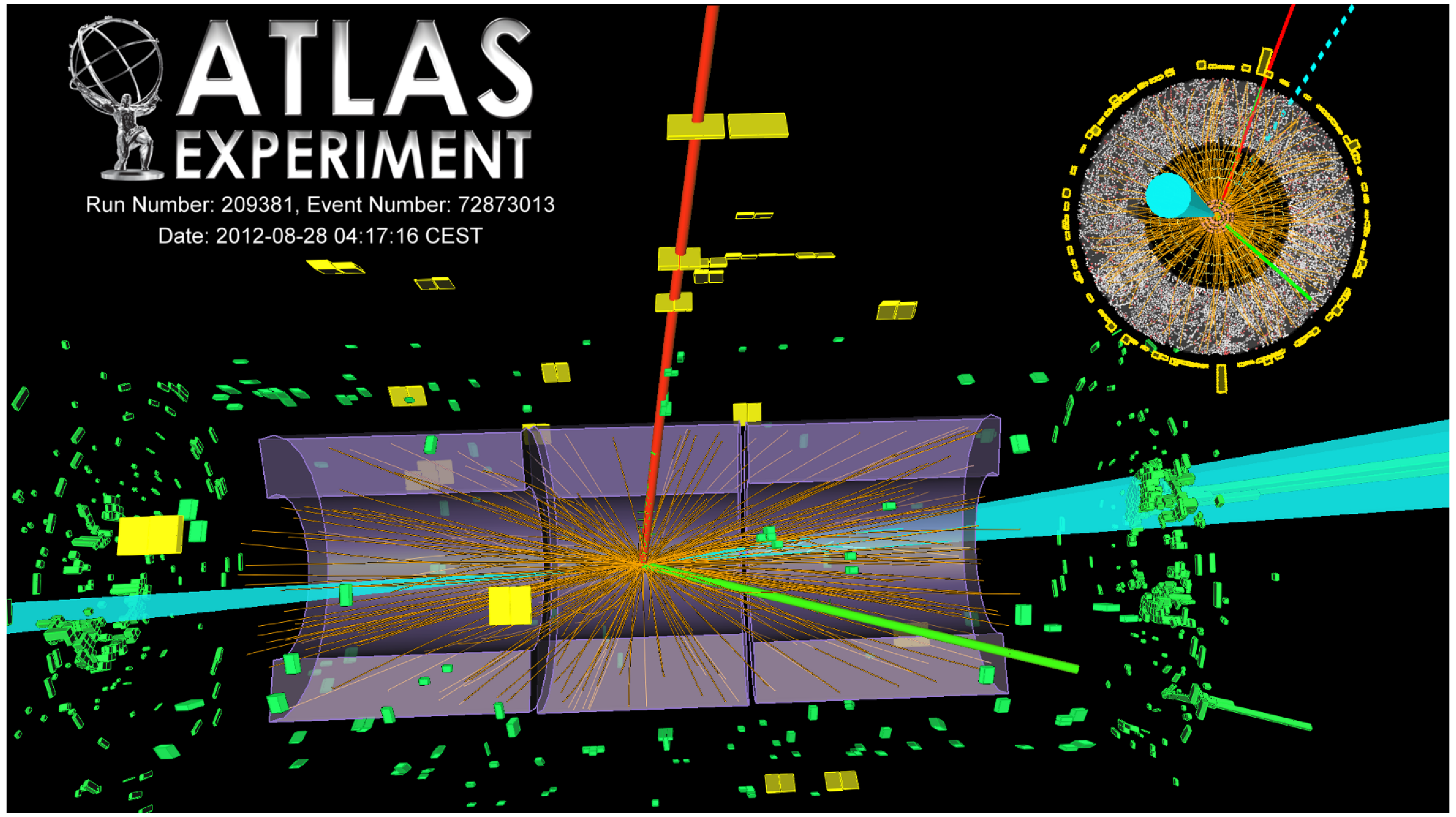
LHC legacy measurements

- What's hiding in the electroweak sector?

  ⇒ precision constraints on dimension-6 EFT operators
  (or Pseudo-Observables, non-linear EFT, …)

- These measurements are difficult!

  1. Many parameters
  2. Many observables



[ATLAS 1501.04943]

LHC legacy measurements

- What's hiding in the electroweak sector?

  ⇒ precision constraints on dimension-6 EFT operators
  (or Pseudo-Observables, non-linear EFT, …)

- These measurements are difficult!

  1. Many parameters
  2. Many observables
  3. Subtle kinematic effects

LHC legacy measurements

- What's hiding in the electroweak sector?

  ⇒ precision constraints on dimension-6 EFT operators
  (or Pseudo-Observables, non-linear EFT, …)

- These measurements are difficult!

  1. Many parameters
  2. Many observables
  3. Subtle kinematic effects

[M. Yao, idea for analogy: K. Cranmer]

LHC legacy measurements

- What's hiding in the electroweak sector?

  ⇒ precision constraints on dimension-6 EFT operators
  (or Pseudo-Observables, non-linear EFT, …)

- These measurements are difficult!

  1. Many parameters
  2. Many observables
  3. Subtle kinematic effects
  4. The likelihood function of high-dimensional observables cannot be calculated

LHC legacy measurements

- What's hiding in the electroweak sector?

  ⇒ precision constraints on dimension-6 EFT operators
  (or Pseudo-Observables, non-linear EFT, …)

- These measurements are difficult!

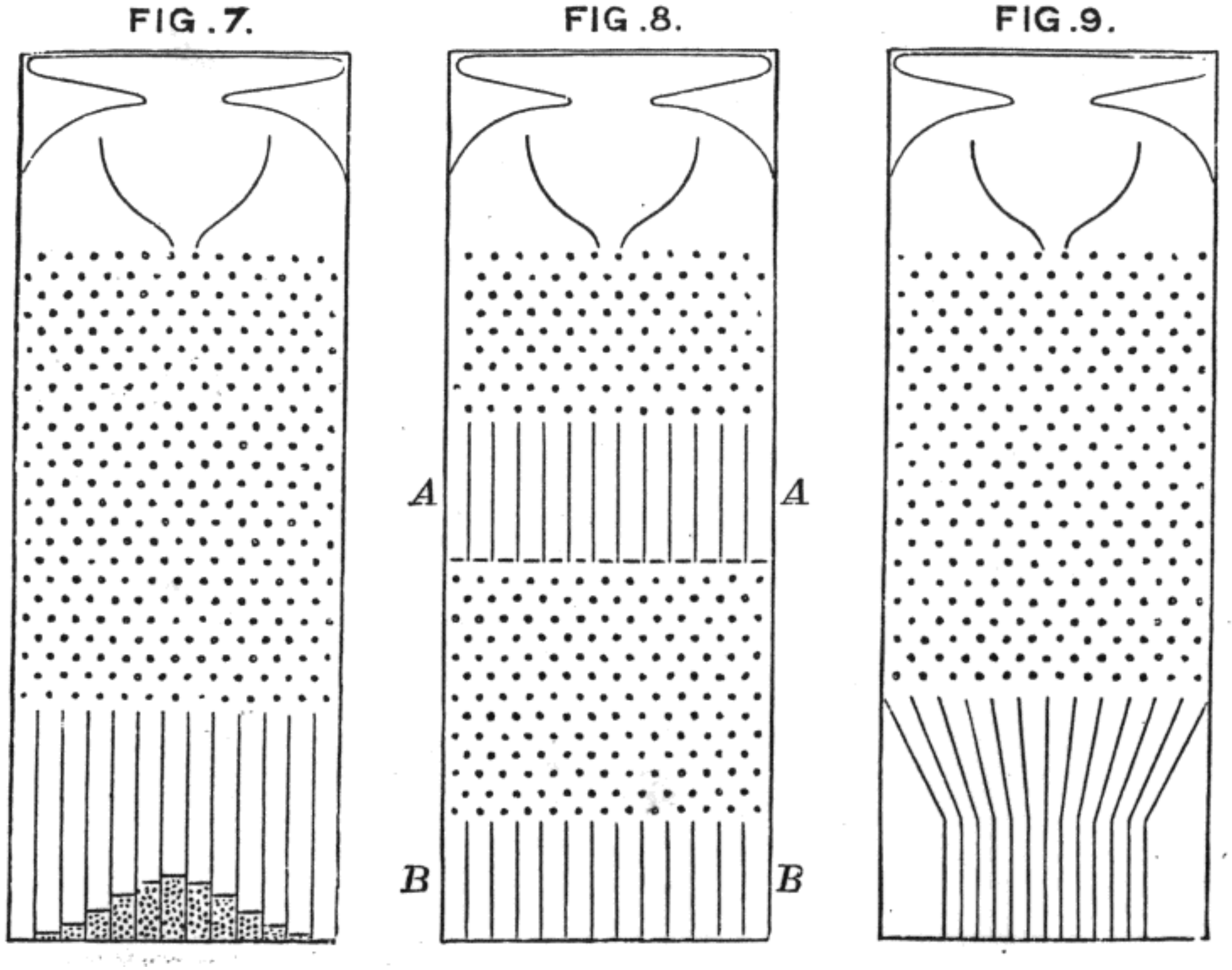
  1. Many parameters
  2. Many observables
  3. Subtle kinematic effects
  4. The likelihood function of high-dimensional observables cannot be calculated

- Established data analysis methods struggle…
  New ideas can improve the sensitivity to new physics!


PHENO
How you can use this
EFT example
Mining gold:
New inference techniques
Established methods
The problem of
"likelihood-free inference"
VINO
Großes Fass,
Heidelberg Castle,
1751

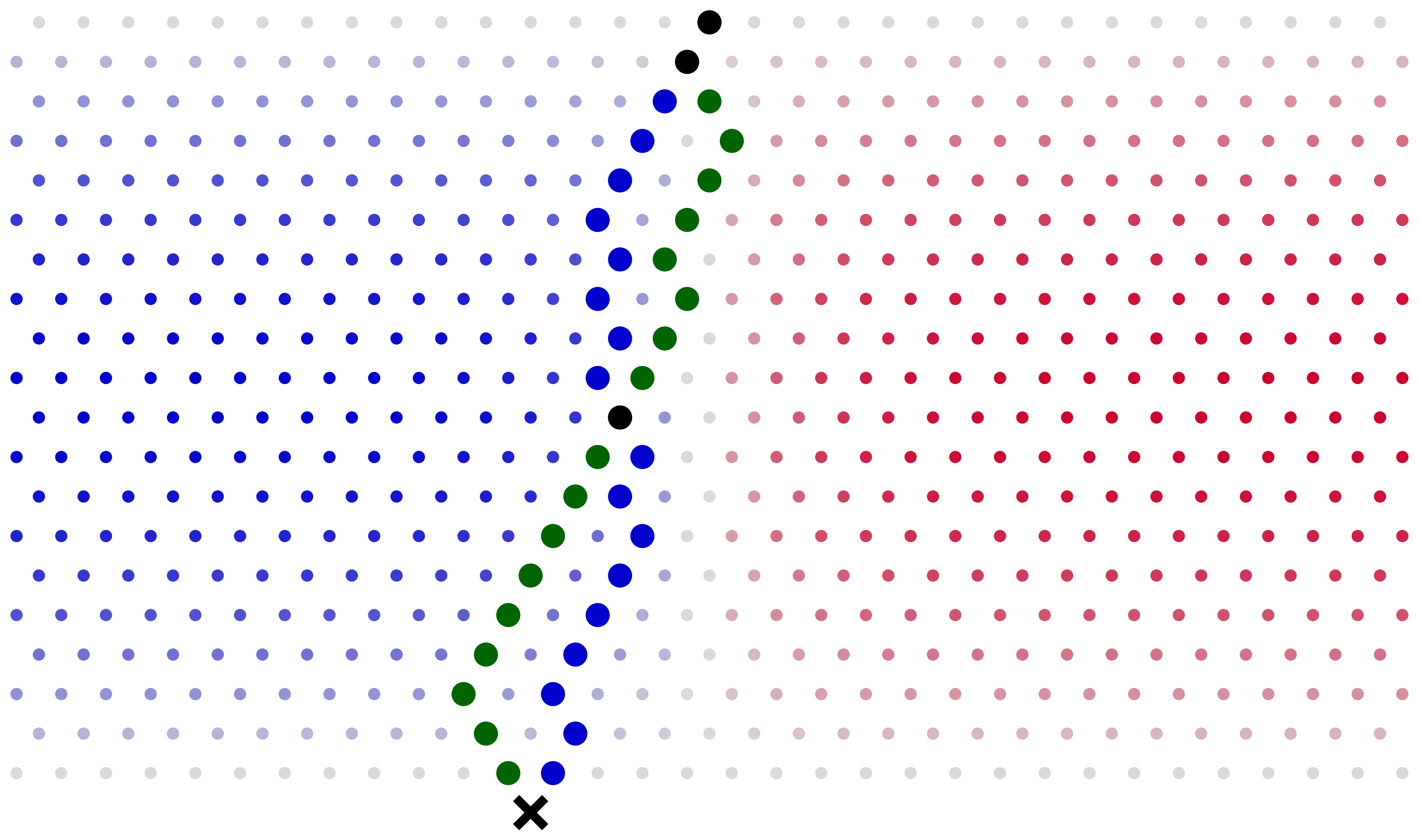
Likelihood-free inference

The Galton board



[F. Galton 1889]

Probabilities from integrating trajectories



Probability of ending in bin x : $p(x) = \int \mathrm{d}z \; p(x, z)$

Sum over
all trajectories
("latent variables")

Probability of
each path z
from start to x

The generalized Galton board

What if probability to go left at a nail is not always 0.5, but some (known) function of some parameters θ ?

- Prediction: given θ, generate samples of observations $\{x_i\}$.

  Simple: just drop balls!

The generalized Galton board

What if probability to go left at a nail is not always 0.5, but some (known) function of some parameters θ ?

- Prediction: given θ, generate samples of observations $\{x_i\}$.

  Simple: just drop balls!

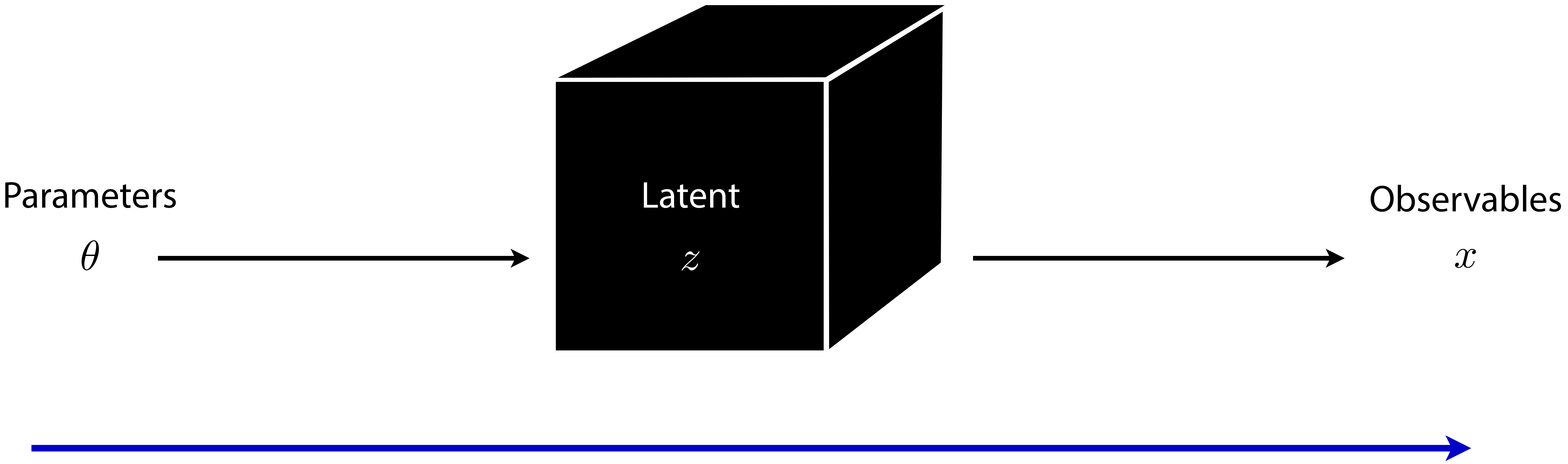
- Inference: given observations $\{x_i\}$, what are the most likely values for θ ?

  "Easy" problem if we can evaluate likelihood

$$p(x|\theta) = \int \mathrm{d}z \, p(x, z|\theta) \, .$$

  But the number of possible paths z can be huge, and it becomes impossible to calculate the integral!

“Likelihood-free inference”



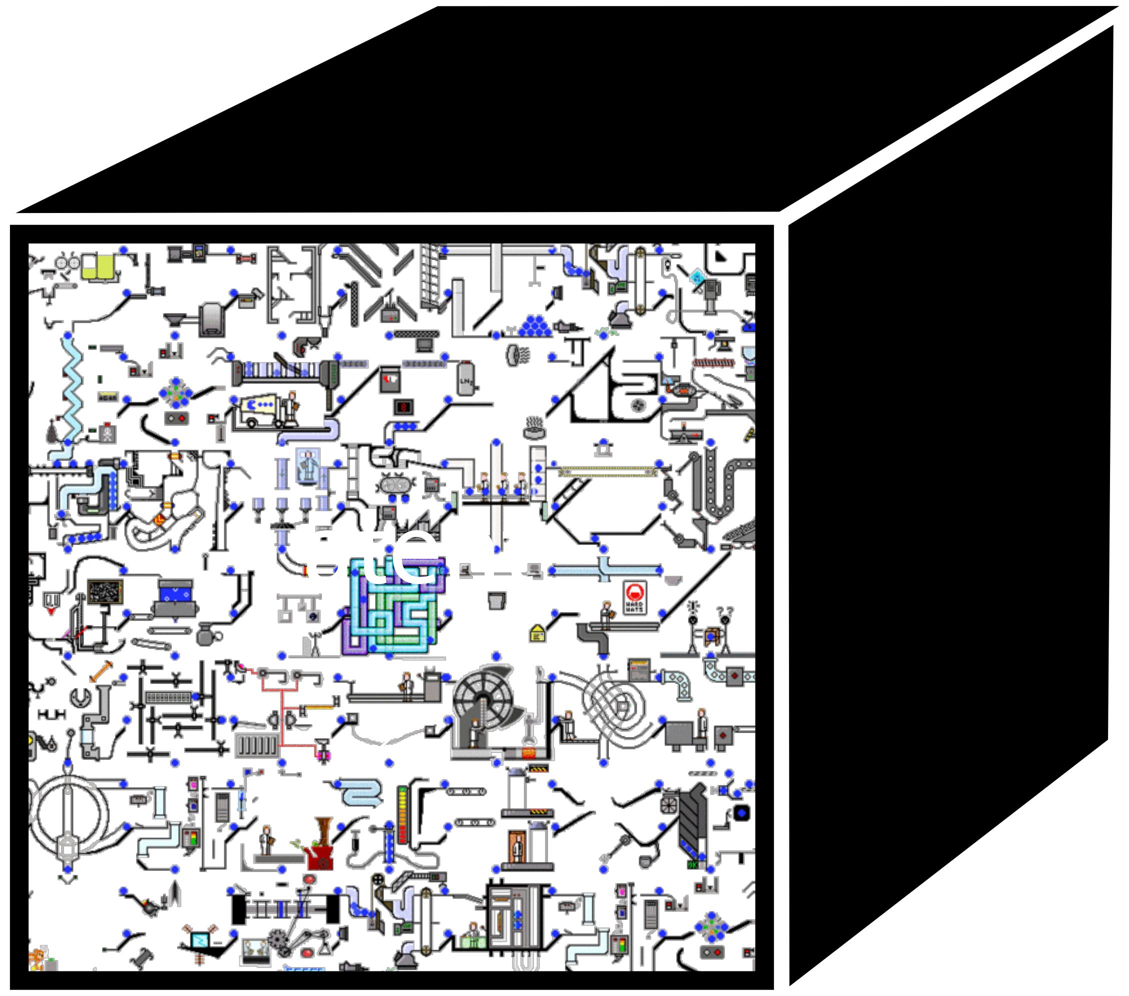
Prediction:

- Well-understood mechanistic model
- Simulator can generate samples

“Likelihood-free inference”

Parameters

θ



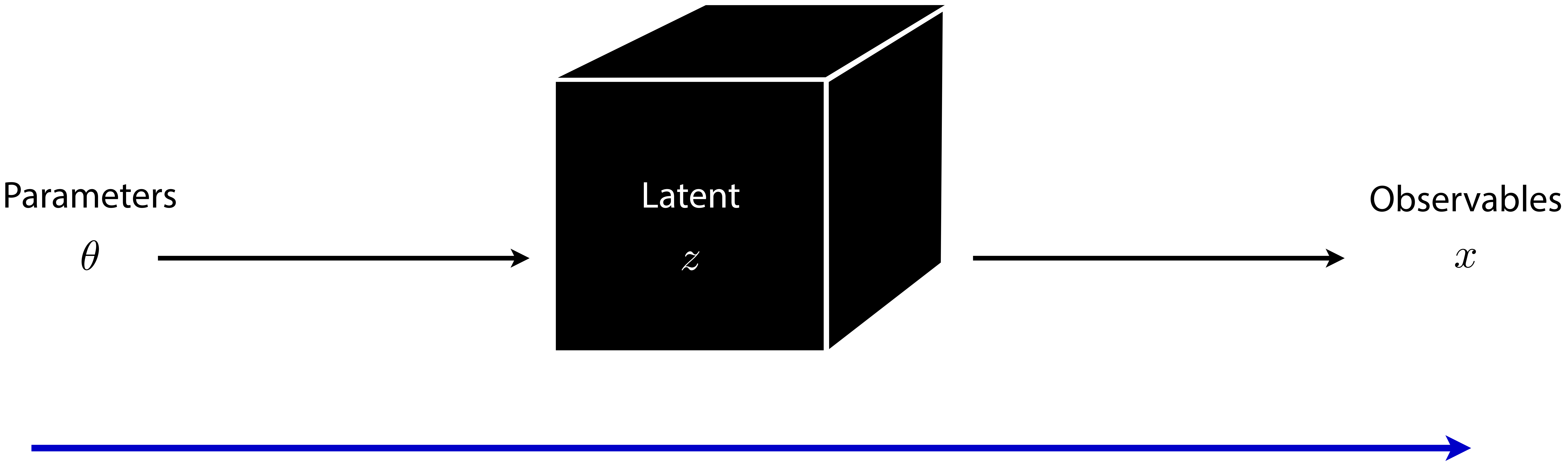
Observables

x

Prediction:

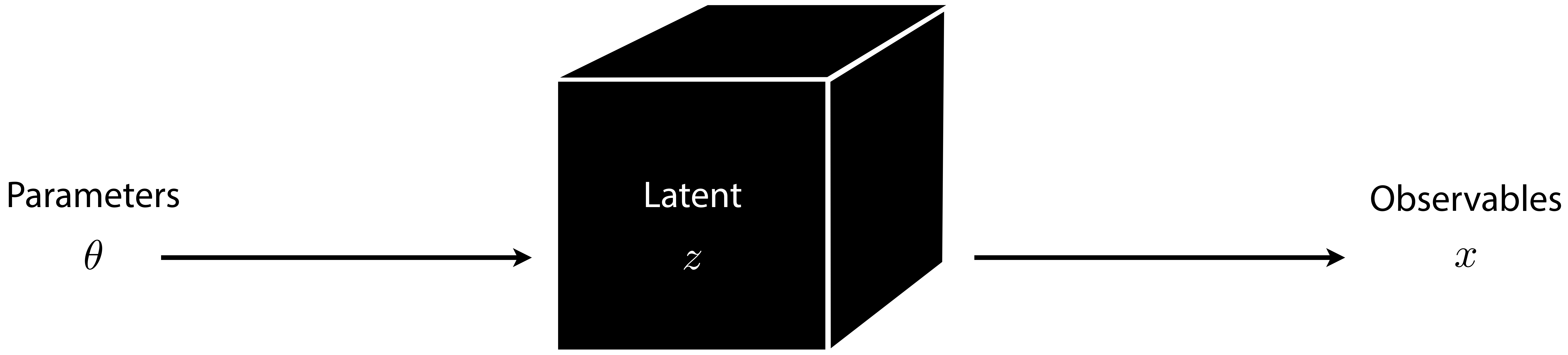
- Well-understood mechanistic model
- Simulator can generate samples

"Likelihood-free inference"



Prediction:
- Well-understood mechanistic model
- Simulator can generate samples

“Likelihood-free inference”



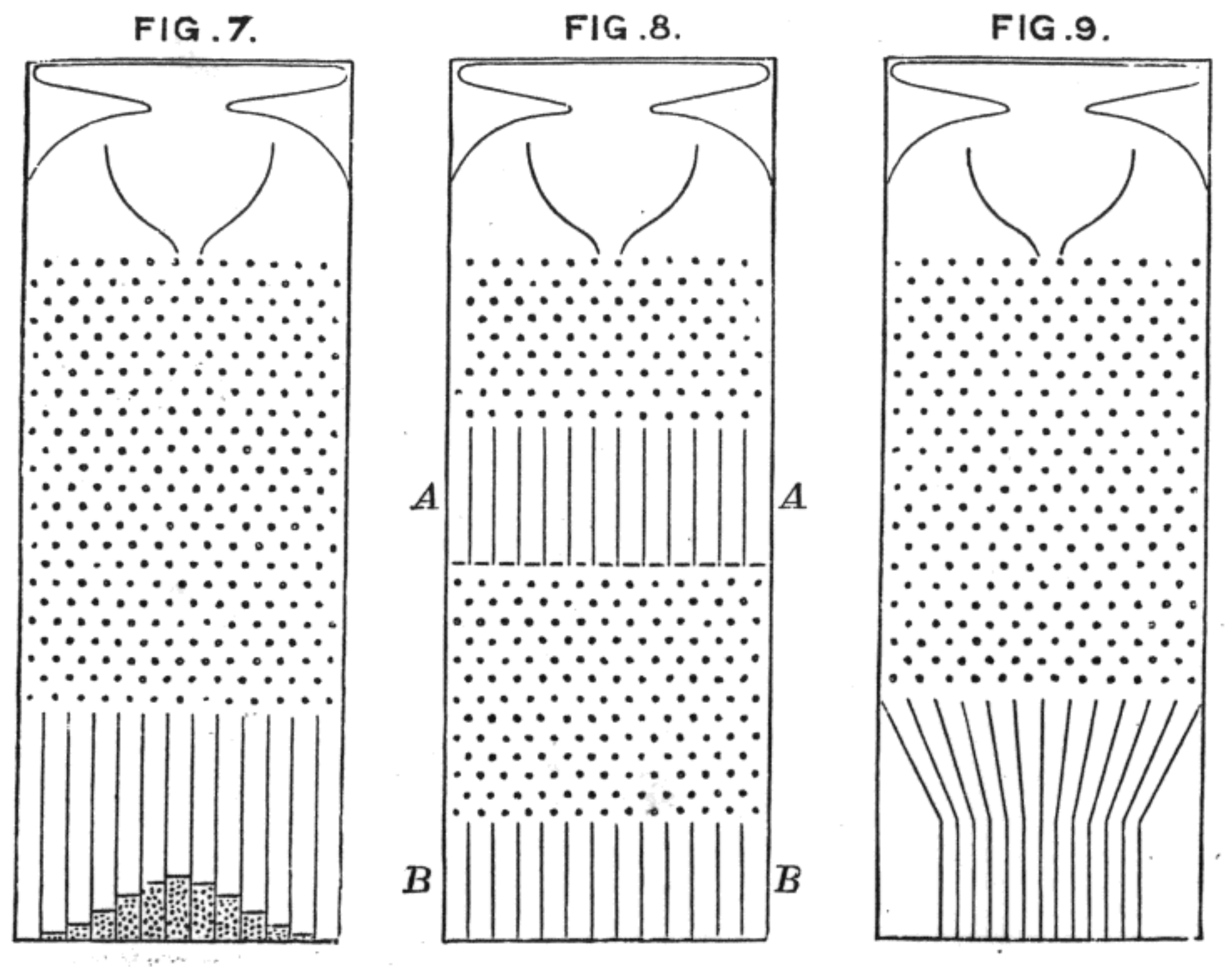
Prediction:

- Well-understood mechanistic model
- Simulator can generate samples

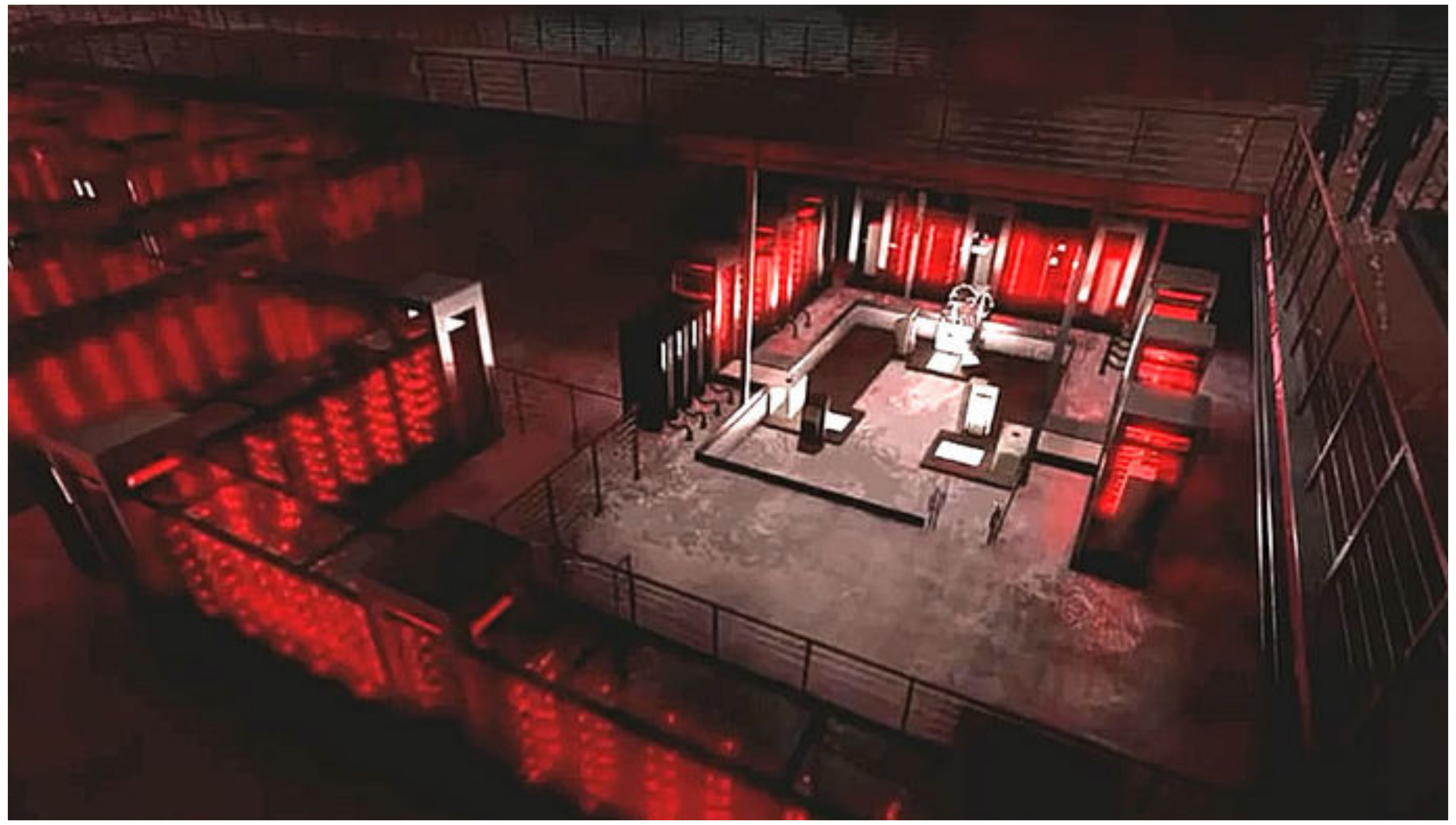
Inference:

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

Galton board: metaphor for simulator-based science



[F. Galton 1889]



[HBO 2018]

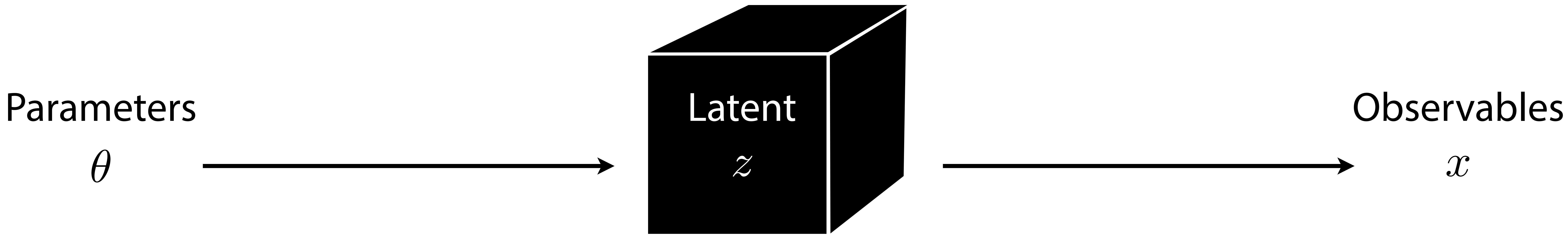
Galton board device $\longrightarrow$ Computer simulation

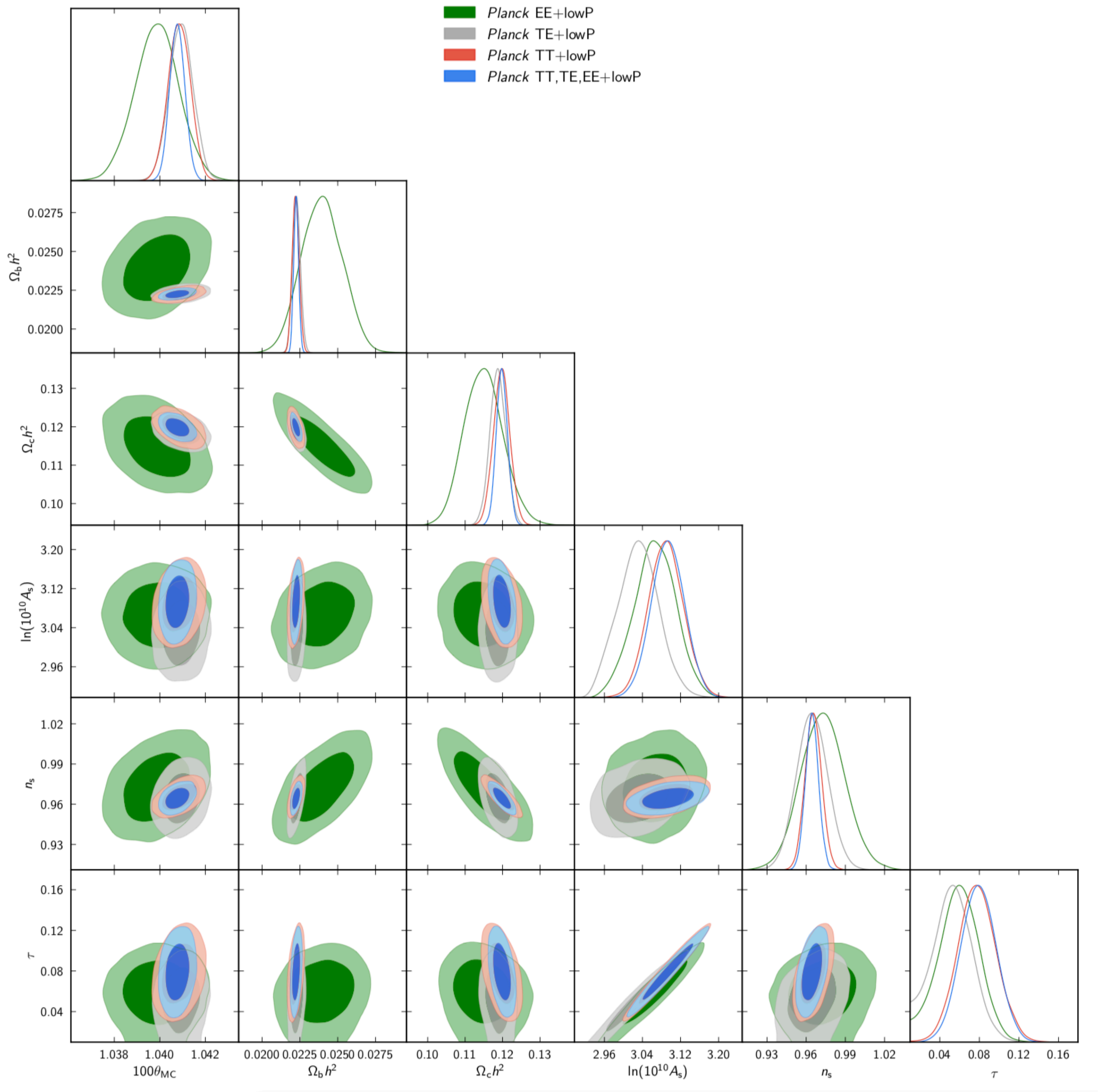
Parameters θ $\longrightarrow$ Model parameters θ

Bins x $\longrightarrow$ Observables x

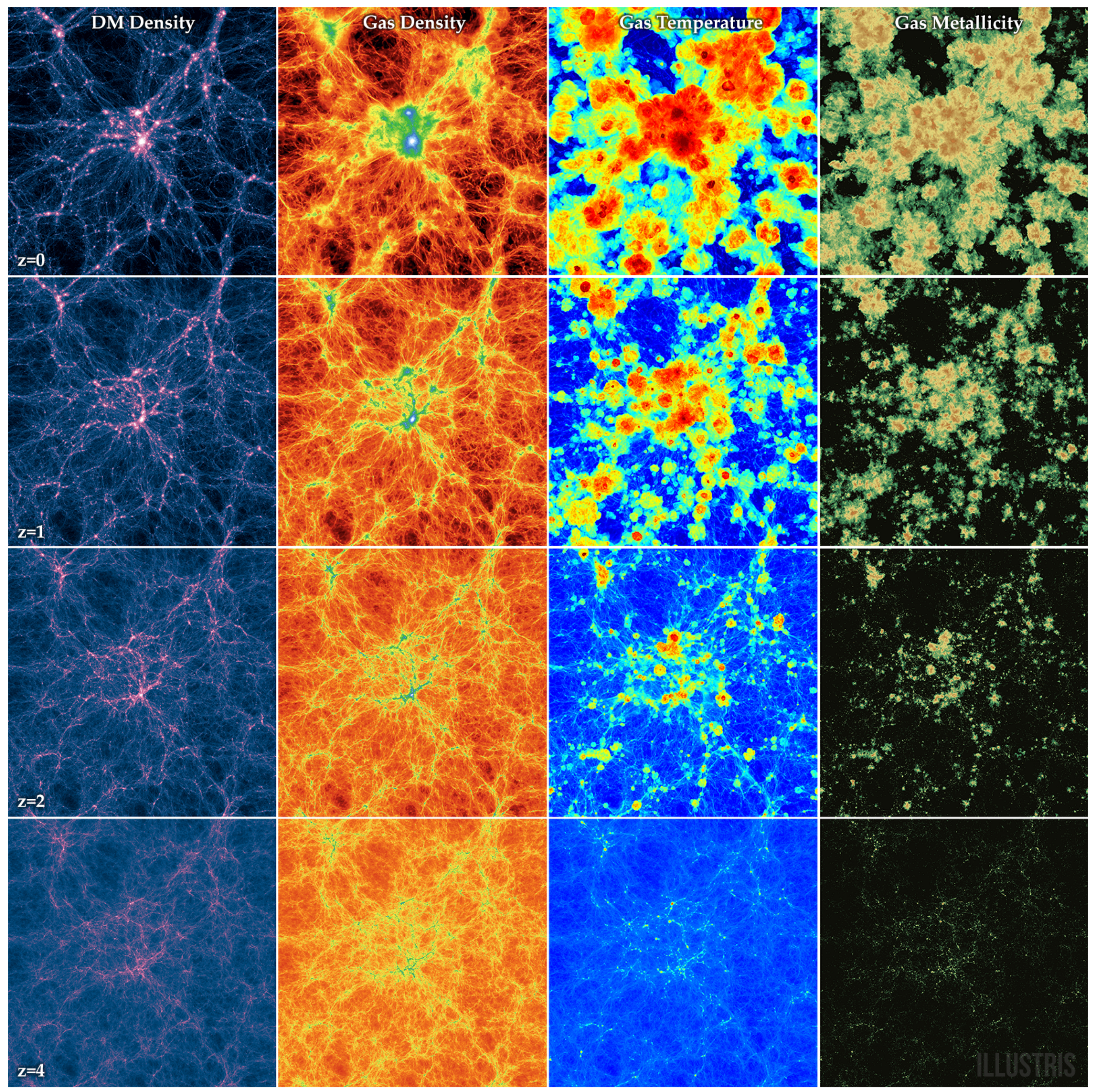
Path z $\longrightarrow$ Latent variables z
(stochastic execution trace through simulator)

Cosmological N-body simulations

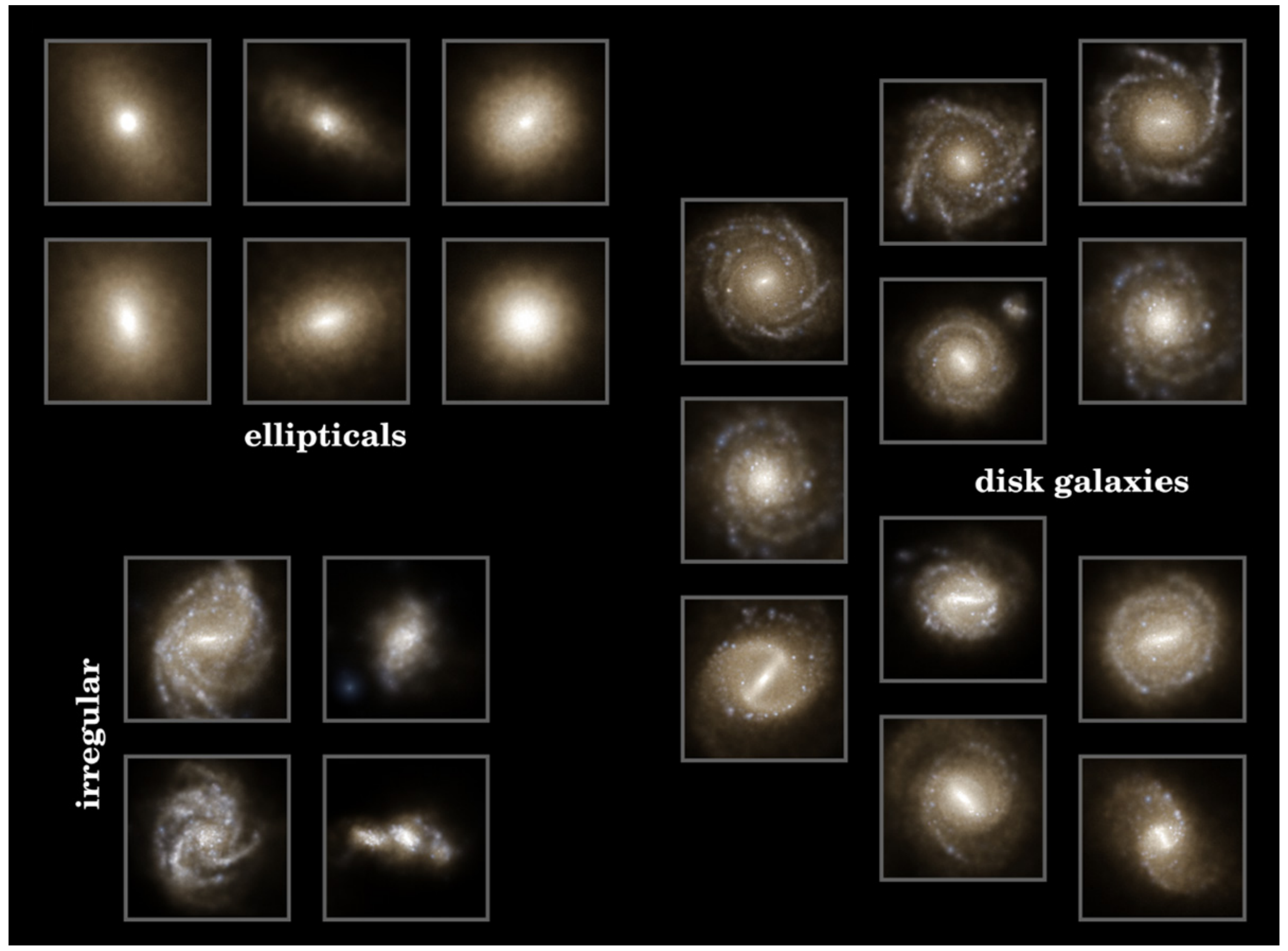




[Planck 1502.01589]

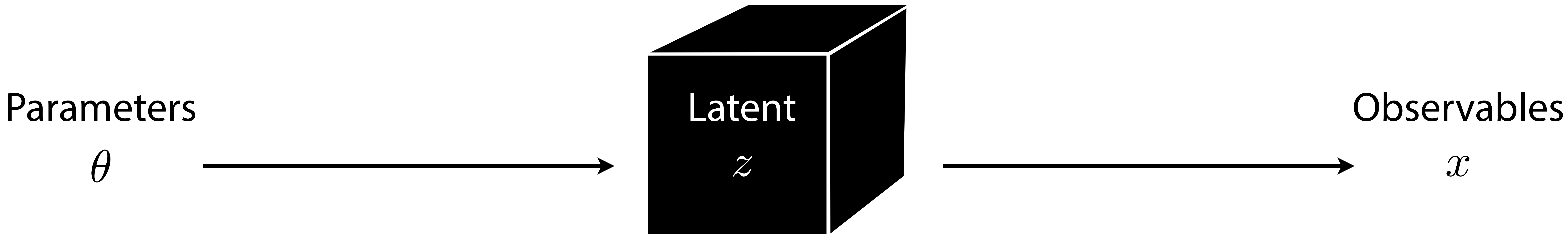


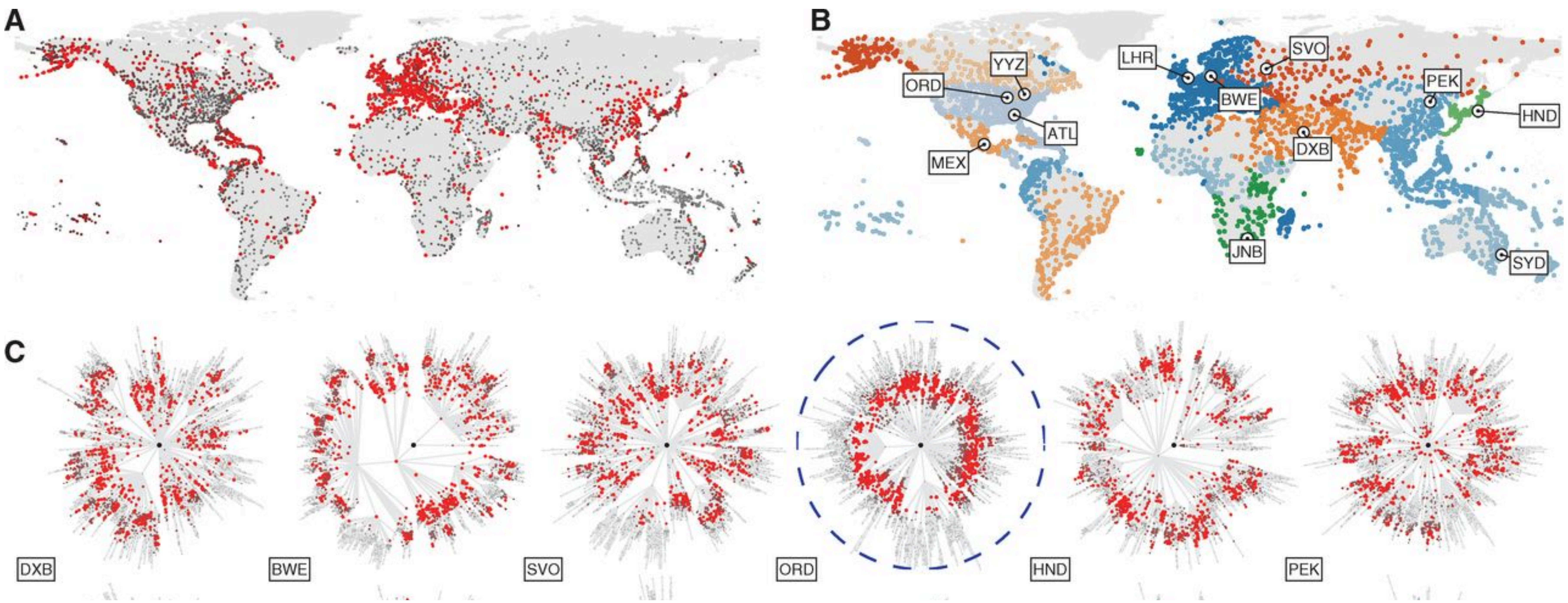
[Illustris 1405.2921]



[Illustris 1405.2921]

Epidemiology





[D. Brockmann, D. Helbing 2013]

Particle physics

Theory
parameters
θ

Evolution

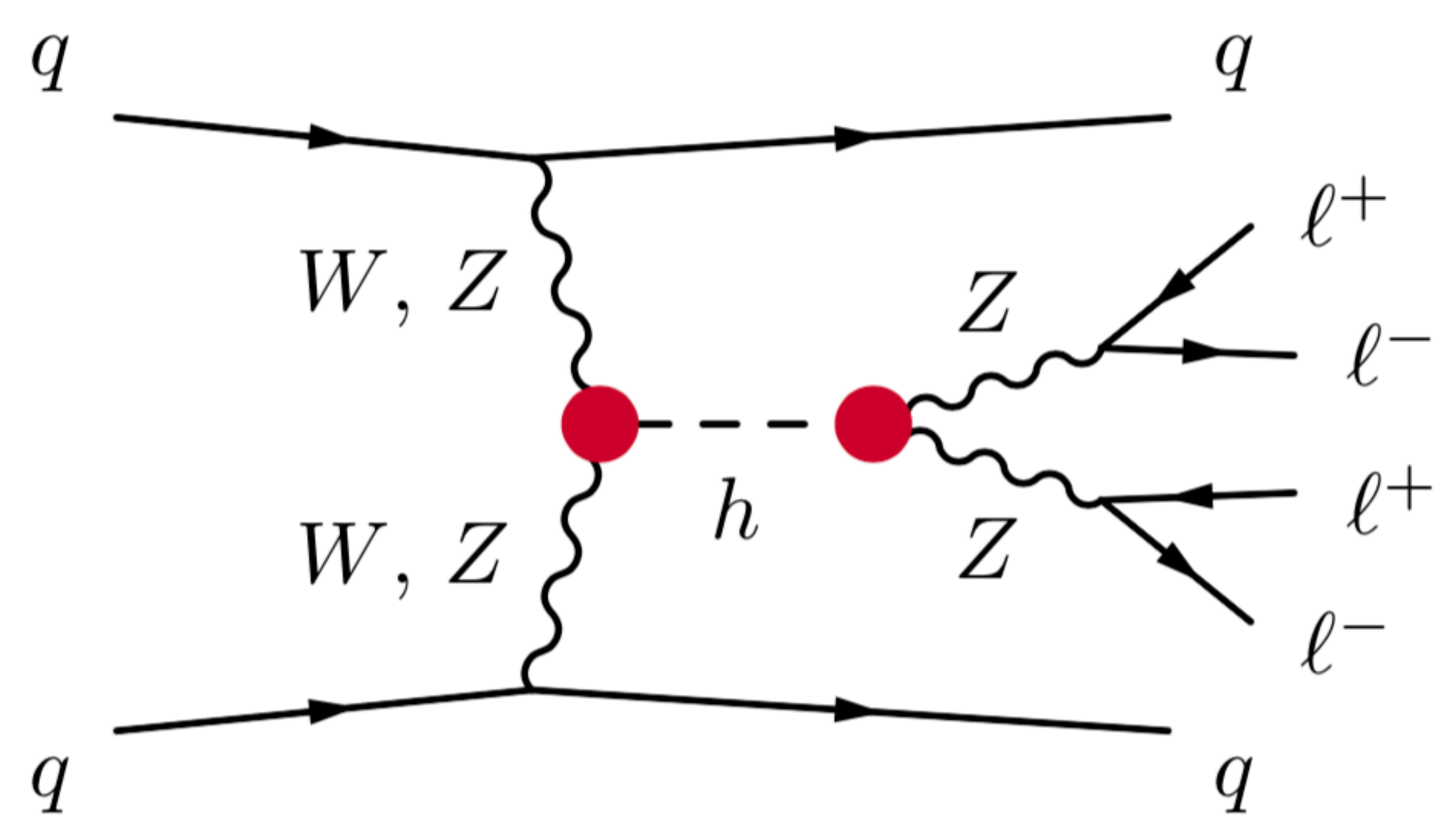
Particle physics

Latent variables

Parton-level momenta

Theory parameters

$z_p \longleftarrow \theta$

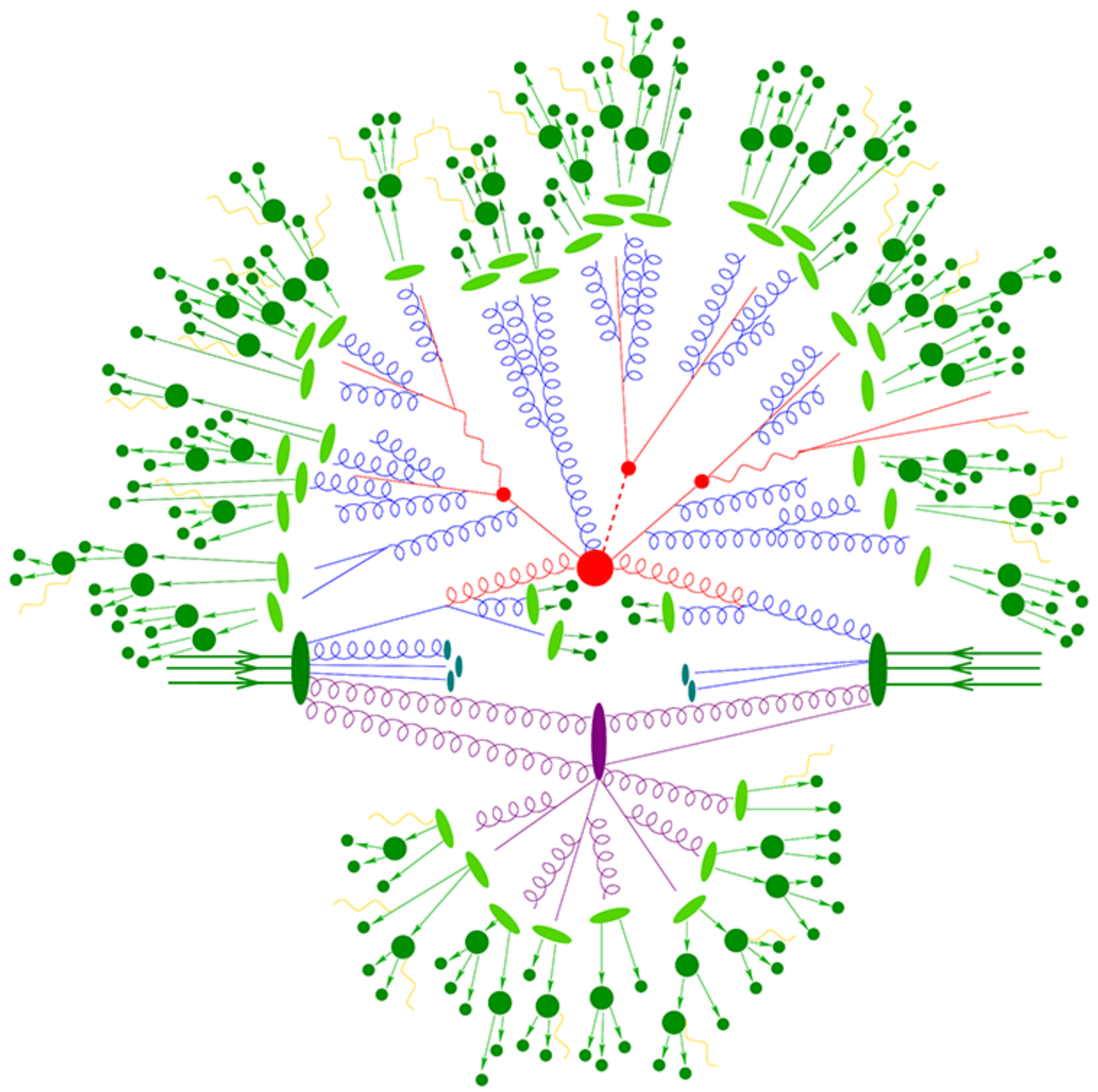


Evolution

Particle physics

Latent variables

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



[F. Krauss]

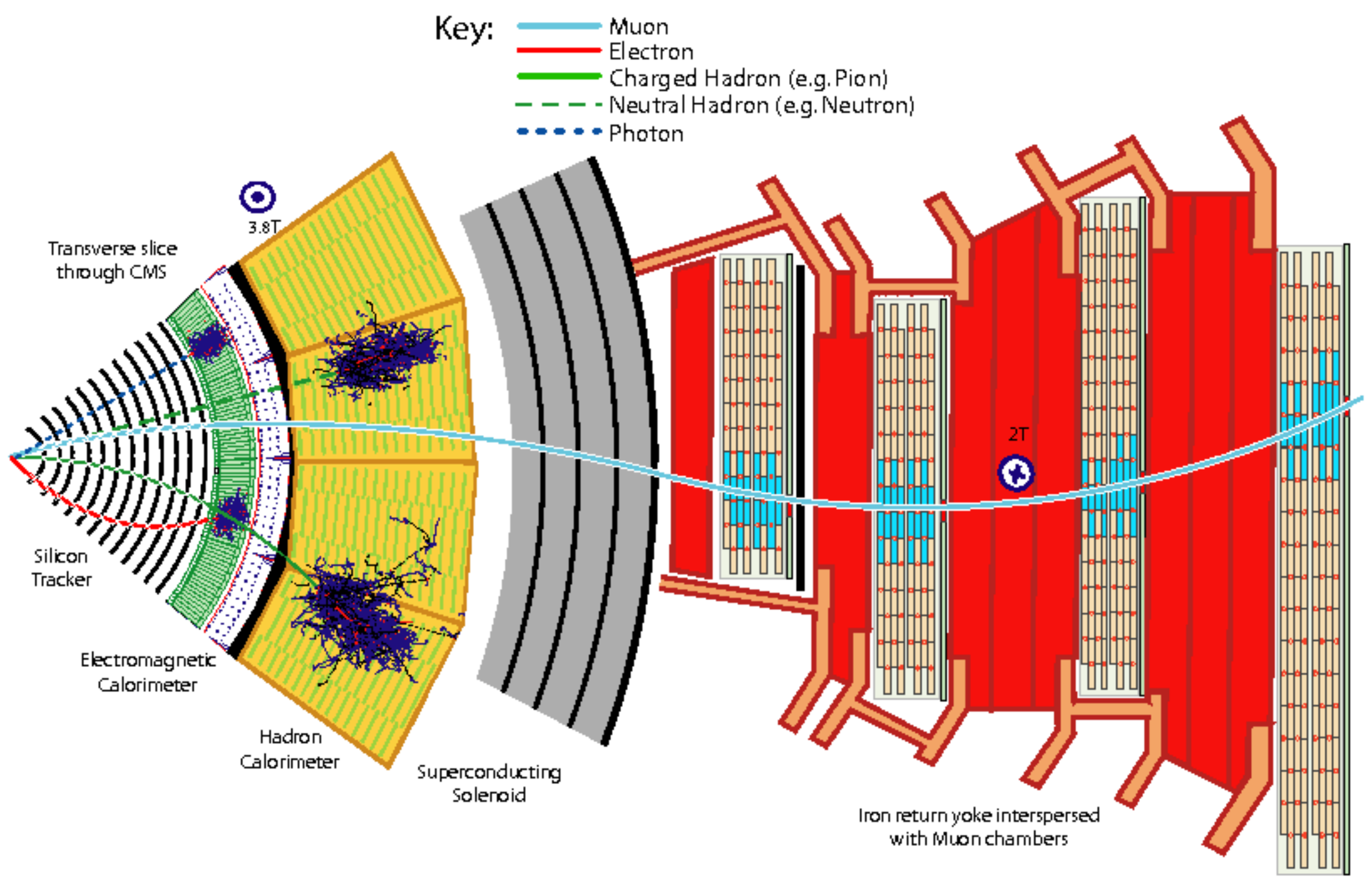
$\longleftarrow$

Evolution

Particle physics

Latent variables

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



[CMS]

$\longleftarrow$

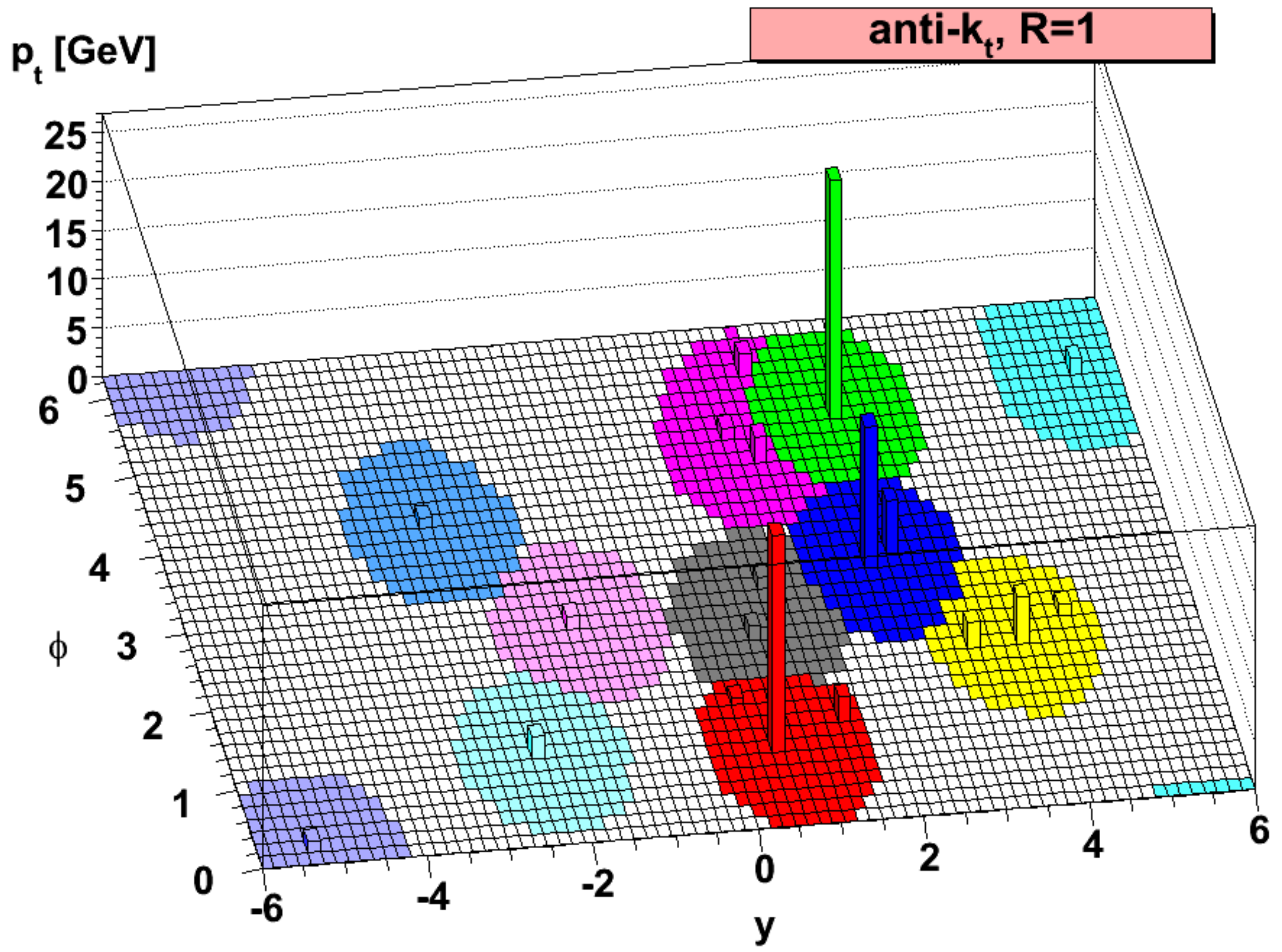
Evolution

Particle physics

Latent variables

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

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



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

Evolution

Particle physics

Latent variables

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

GEANT4
A SIMULATION TOOLKIT

MADGRAPH5_aMC@NLO

Prediction (simulation)

Particle physics

Latent variables

Observables	Detector interactions	Shower splittings	Parton-level momenta	Theory parameters

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

$$p(x|\theta) = \int \mathrm{d}z_d \int \mathrm{d}z_s \int \mathrm{d}z_p \quad p(x|z_d) \quad p(z_d|z_s) \quad p(z_s|z_p) \quad p(z_p|\theta)$$

Inference

Particle physics

Latent variables

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

$$p(x|\theta) = \boxed{\int \mathrm{d}z_d \int \mathrm{d}z_s \int \mathrm{d}z_p} \; p(x|z_d) \quad p(z_d|z_s) \quad p(z_s|z_p) \quad p(z_p|\theta)$$

Infeasible to calculate the integral over this enormous space!

$\longrightarrow$

Inference

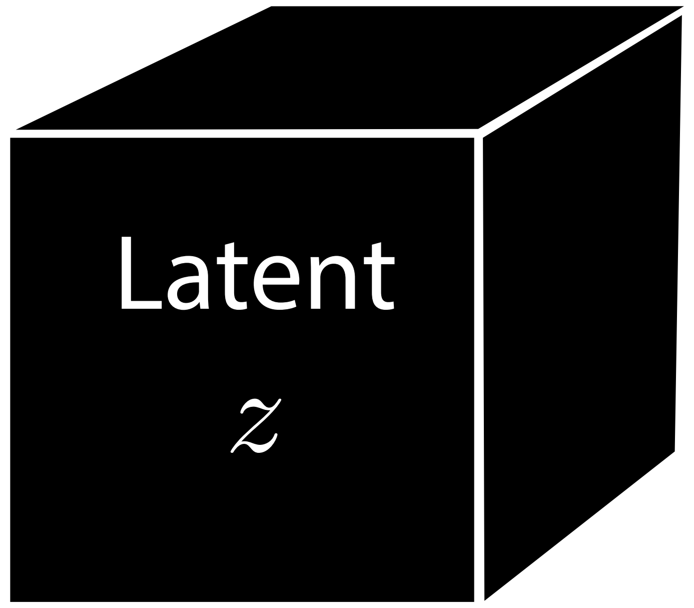
Why has that not stopped us so far?

Solve it by histogramming summary statistics

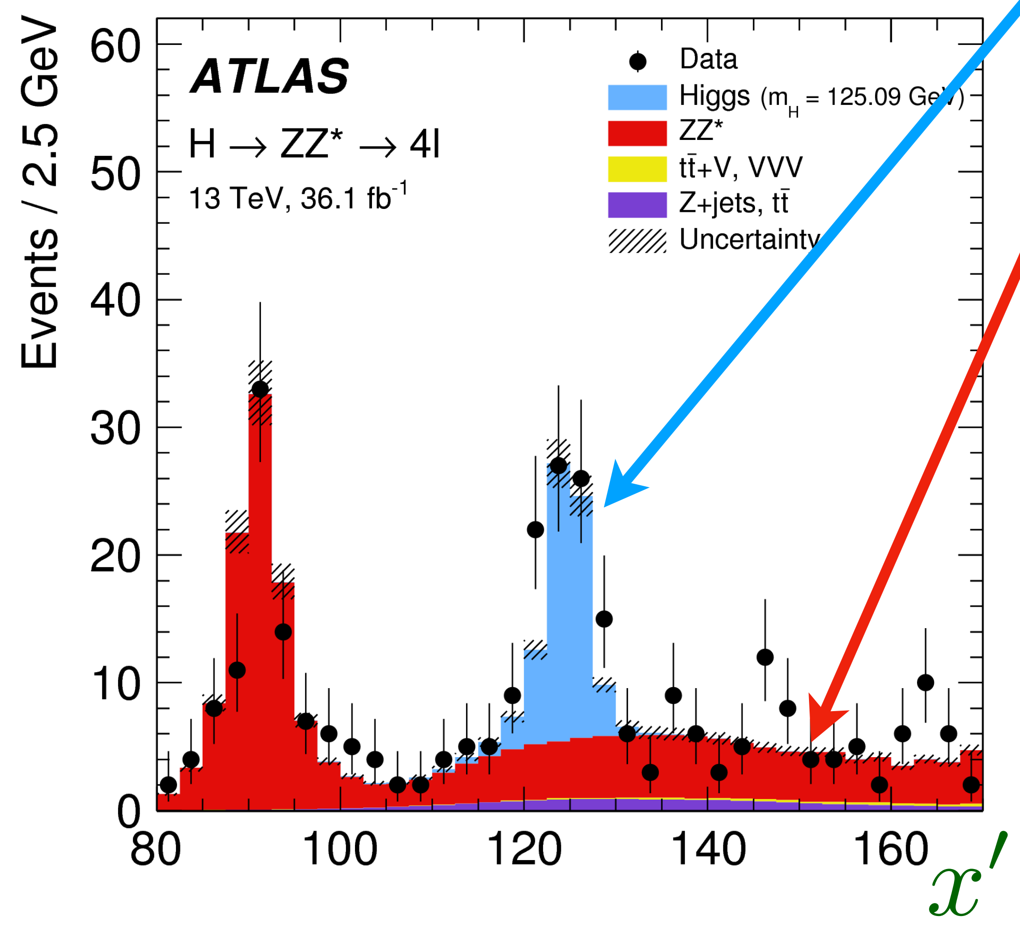
- Most physicists' solution:

Solve it by histogramming summary statistics

- Most physicists' solution:

Parameters θ →  → Observables x → Summary statistics x'

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



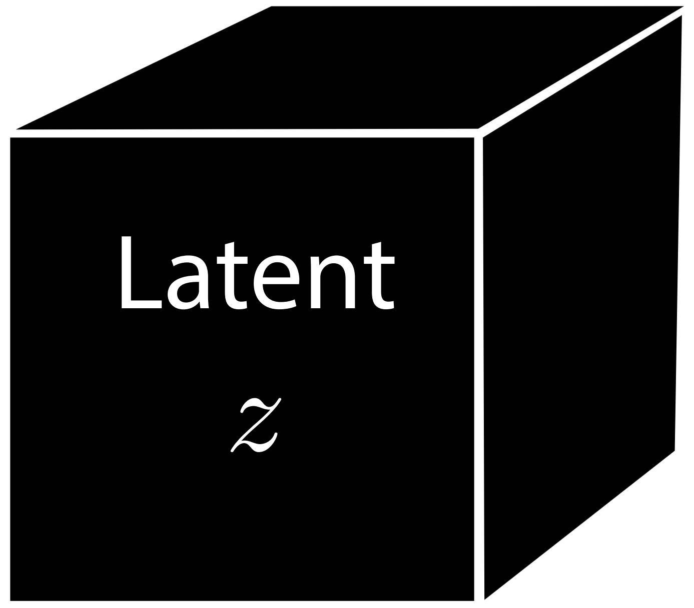
[ATLAS 1712.02304]

Run simulator for different θ, fill histograms

Solve it by histogramming summary statistics

- Most physicists' solution:

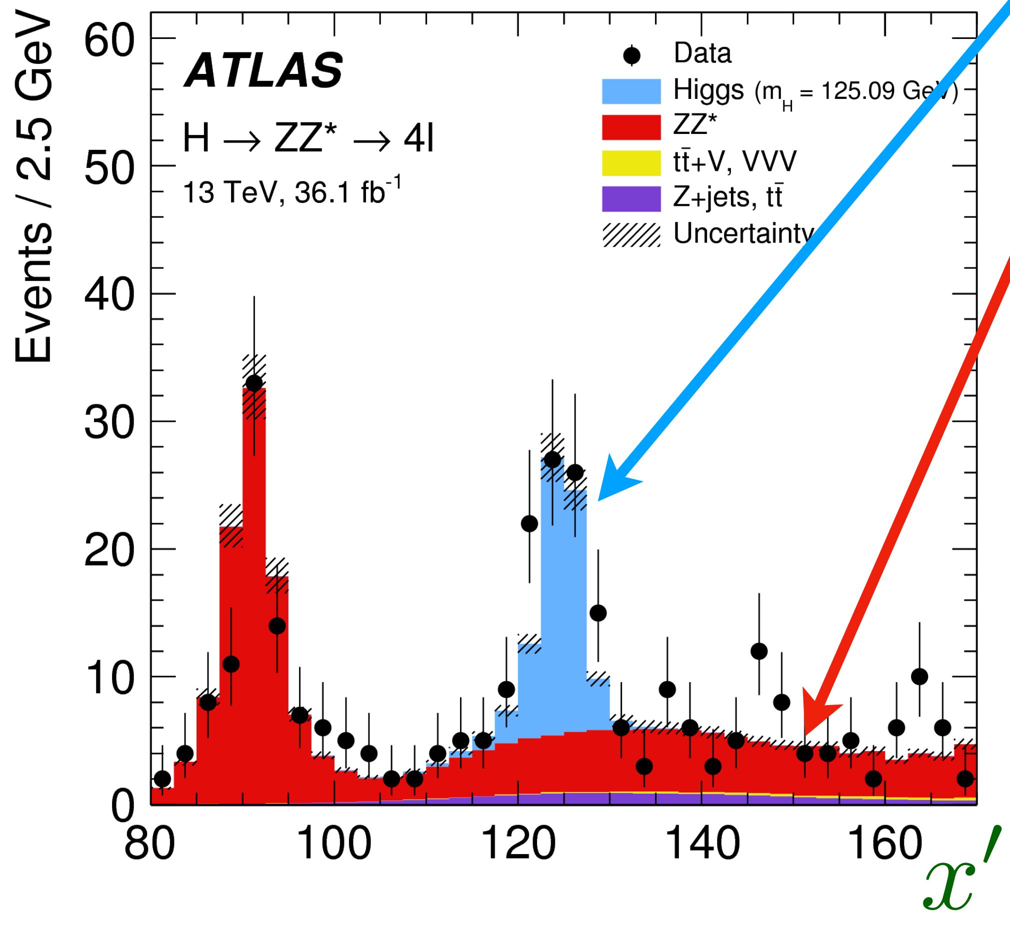
Parameters θ $\longrightarrow$



$\longrightarrow$ Observables x $\longrightarrow$ Summary statistics x'

Run simulator for different θ, fill histograms

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



[ATLAS 1712.02304]

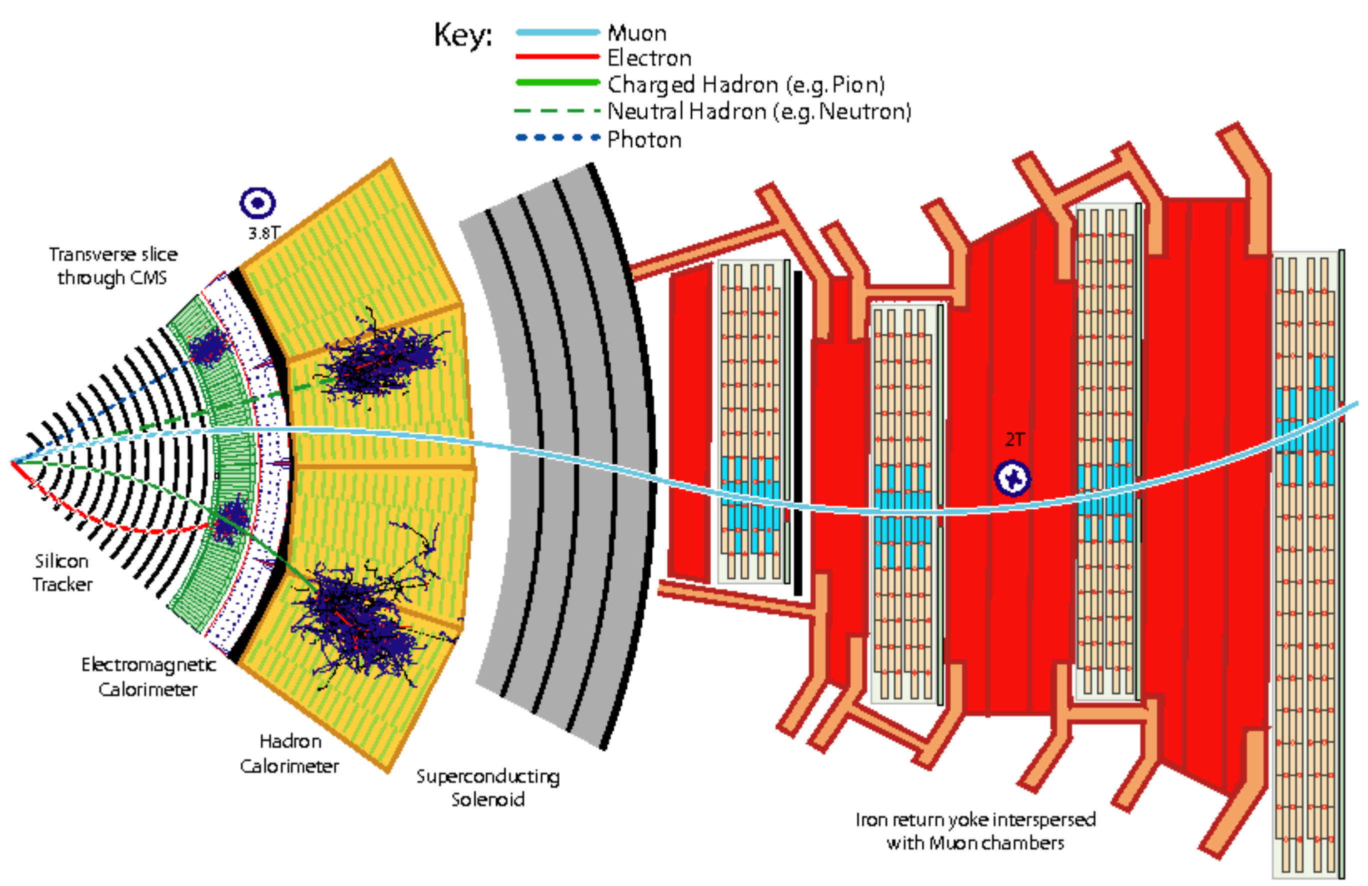
- How to choose x' ? Standard variables often lose information
  [JB, K. Cranmer, F. Kling, T. Plehn 1612.05261; JB, F. Kling, T. Plehn, T. Tait 1712.02350]

- "Curse of dimensionality": Histograms don't scale to high-dimensional x

Solve it by approximating the integral

- Problem: high-dim. integral over shower / detector trajectories

$$p(x|\theta) = \int \mathrm{d}z_d \int \mathrm{d}z_s \int \mathrm{d}z_p \; p(x|z_d)\, p(z_d|z_s)\, p(z_s|z_p)\, p(z_p|\theta)$$



[CMS]

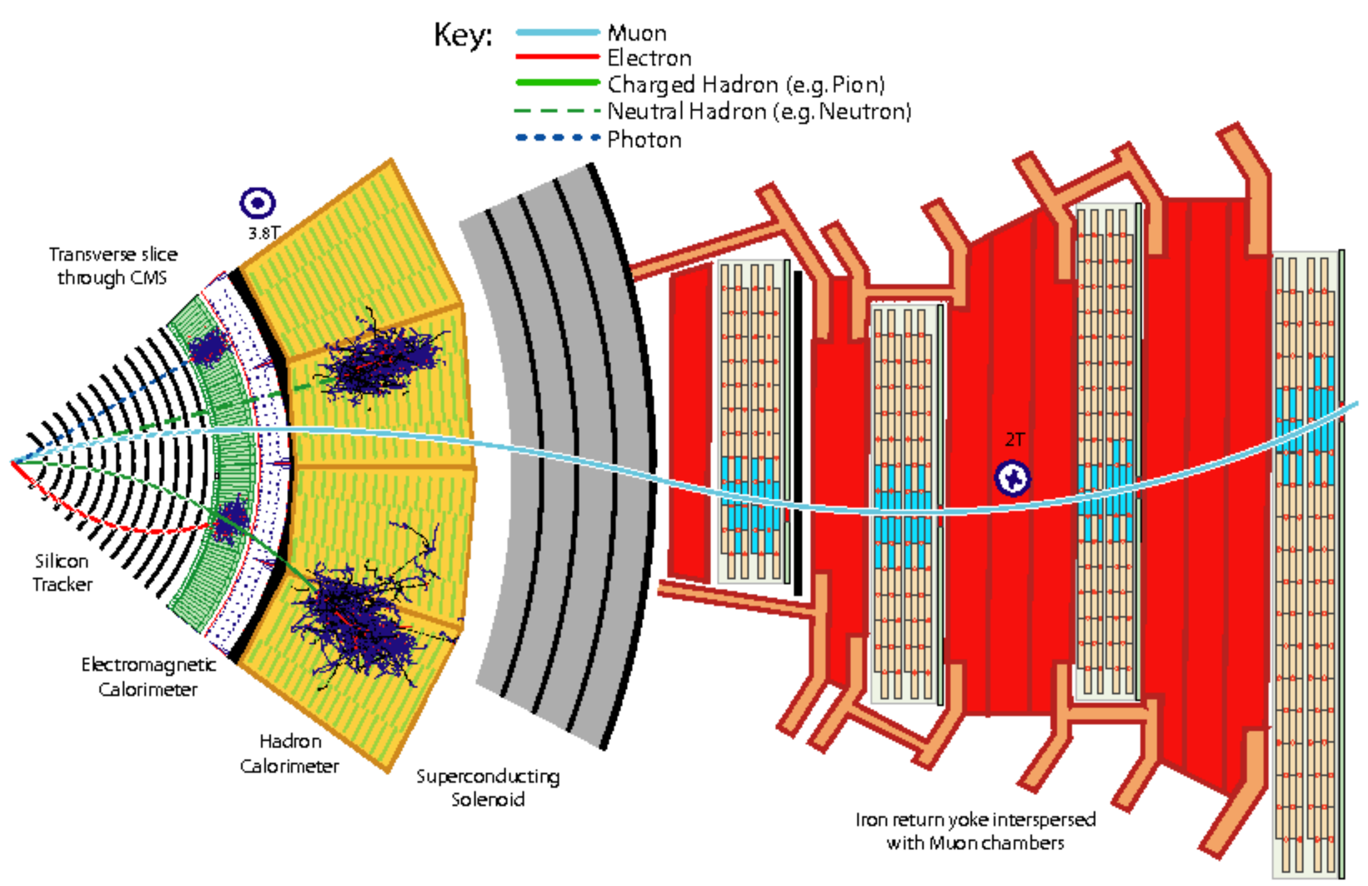
Solve it by approximating the integral

- Problem: high-dim. integral over shower / detector trajectories

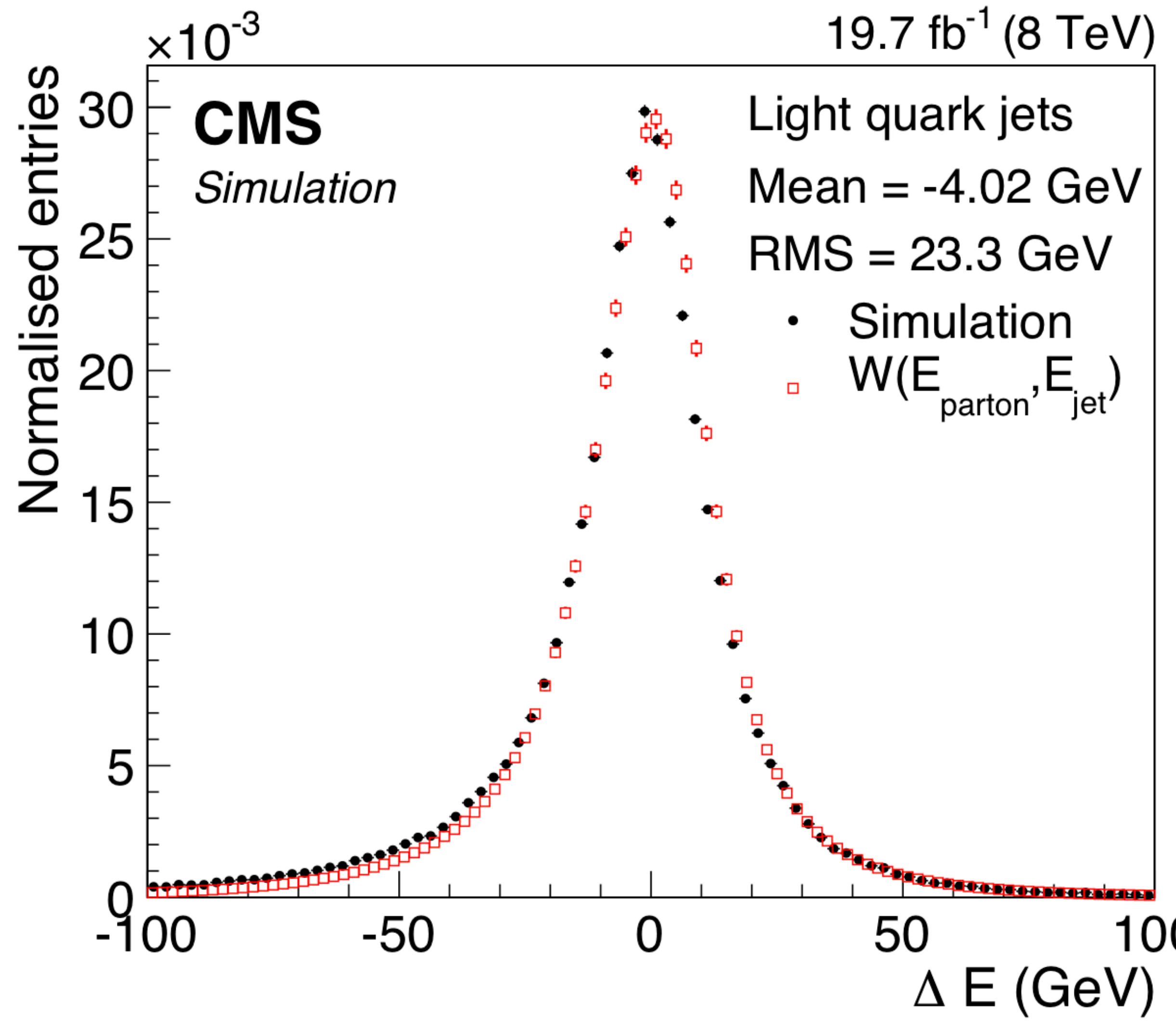
$$p(x|\theta) = \int \mathrm{d}z_d \int \mathrm{d}z_s \int \mathrm{d}z_p \; p(x|z_d)\, p(z_d|z_s)\, p(z_s|z_p)\, p(z_p|\theta)$$

- Matrix Element Method: [K. Kondo 1988]
  - approximate shower + detector effects into transfer function $\hat{p}(x|z_p)$
  - explicitly calculate remaining integral

$$\hat{p}(x|\theta) = \int \mathrm{d}z_p \; \hat{p}(x|z_p)\, p(z_p|\theta)$$



[CMS]



[CMS 1511.06170]

Solve it by approximating the integral

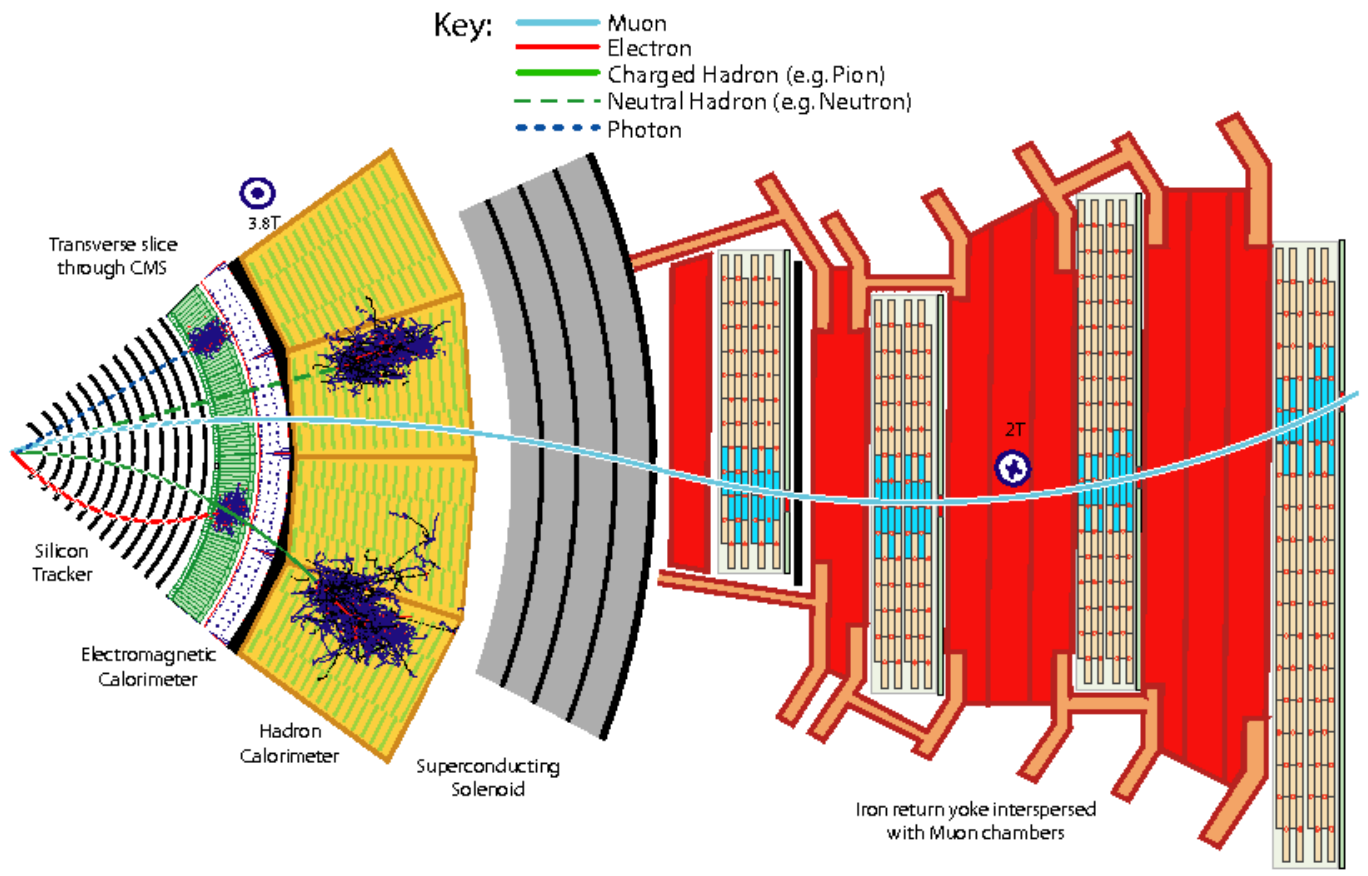
- Problem: high-dim. integral over shower / detector trajectories

$$p(x|\theta) = \int \mathrm{d}z_d \int \mathrm{d}z_s \int \mathrm{d}z_p \; p(x|z_d)\, p(z_d|z_s)\, p(z_s|z_p)\, p(z_p|\theta)$$

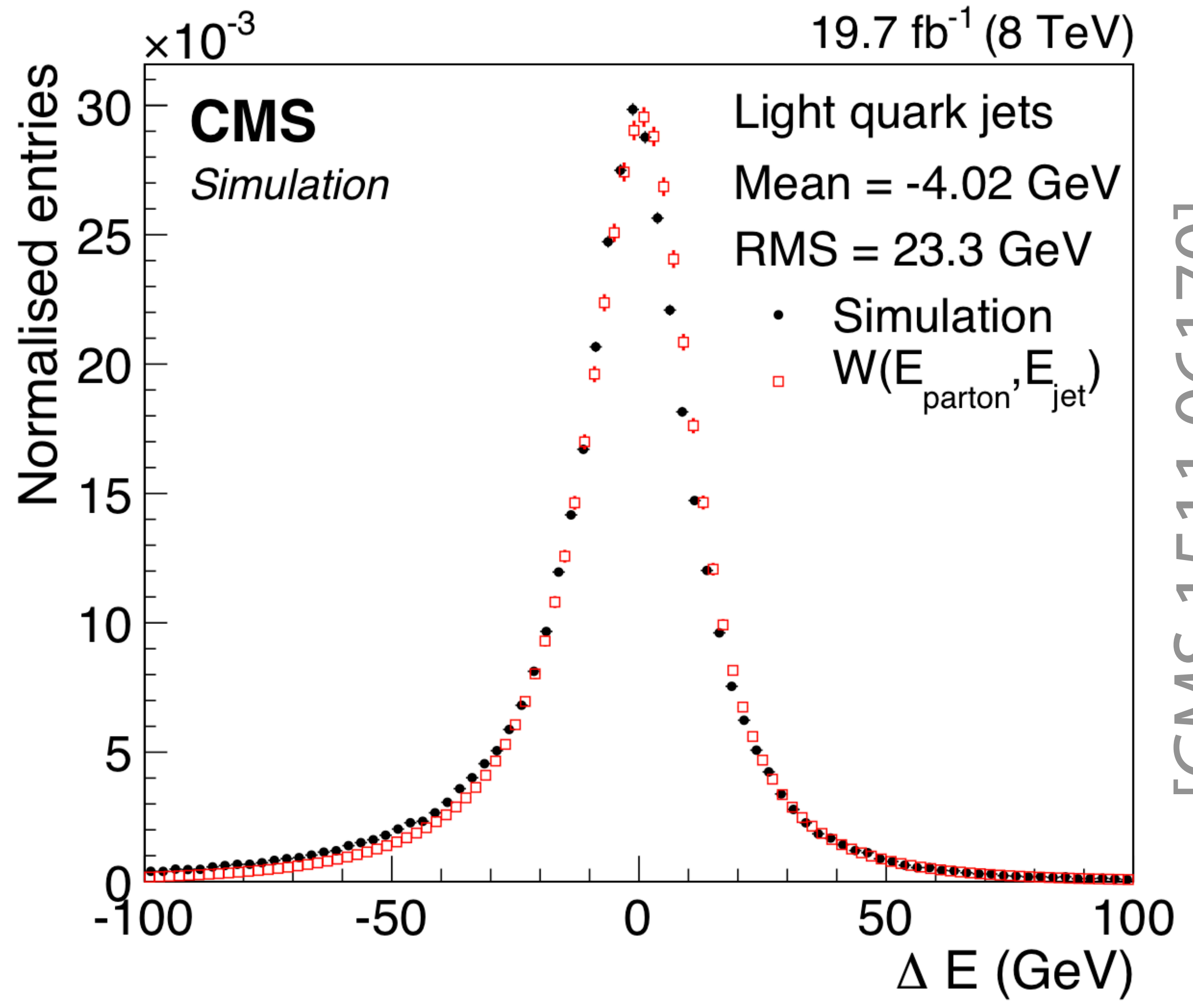
- Matrix Element Method: [K. Kondo 1988]
  - approximate shower + detector effects into transfer function $\hat{p}(x|z_p)$
  - explicitly calculate remaining integral

$$\hat{p}(x|\theta) = \int \mathrm{d}z_p \; \hat{p}(x|z_p)\, p(z_p|\theta)$$

- Uses matrix-element information, no summary statistics necessary, but:
  - ad-hoc transfer functions
  - evaluation still requires calculating an expensive integral



[CMS]



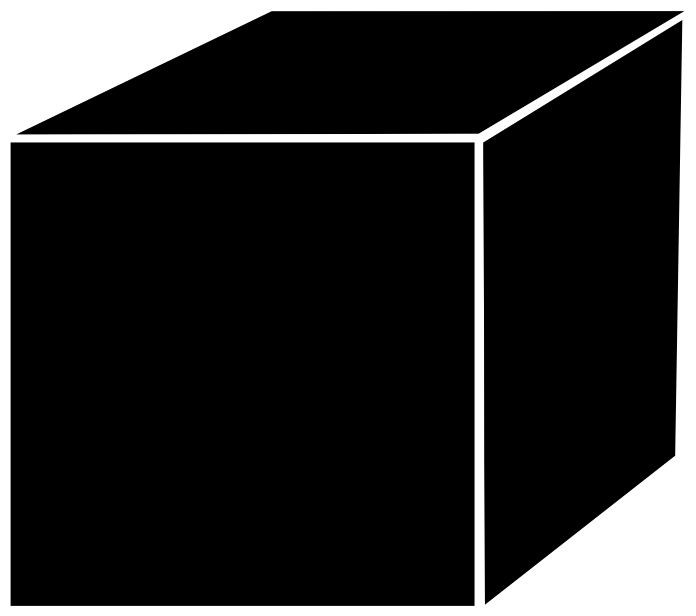
[CMS 1511.06170]

Likelihood-free inference methods

Treat simulator as black box:

- Histograms of observables,
  Approximate Bayesian Computation
  Rely on summary statistics

- Machine learning techniques
  Density networks, CARL, autoregressive models, normalizing flows, …

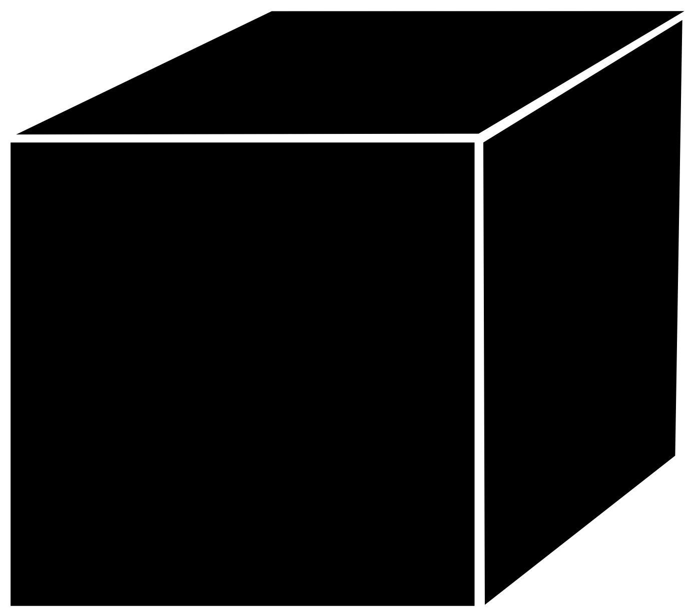


Likelihood-free inference methods

Treat simulator as black box:

- Histograms of observables, Approximate Bayesian Computation
  Rely on summary statistics

- Machine learning techniques
  Density networks, CARL, autoregressive models, normalizing flows, …



Use latent structure:

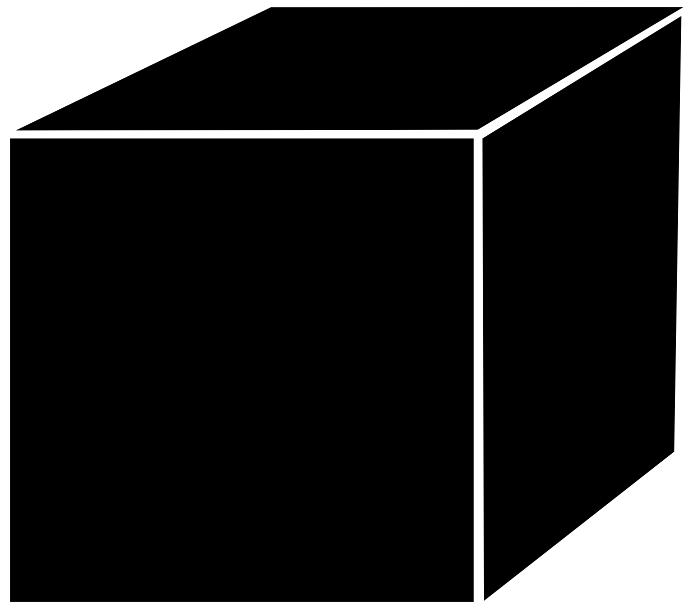
- Matrix Element Method, Optimal Observables, Shower Deconstruction
  Neglect or approximate shower + detector, explicitly calculate z integral

Likelihood-free inference methods

Treat simulator as black box:

- Histograms of observables, Approximate Bayesian Computation
  Rely on summary statistics

- Machine learning techniques
  Density networks, CARL, autoregressive models, normalizing flows, …



Use latent structure:

- Matrix Element Method, Optimal Observables, Shower Deconstruction
  Neglect or approximate shower + detector, explicitly calculate z integral

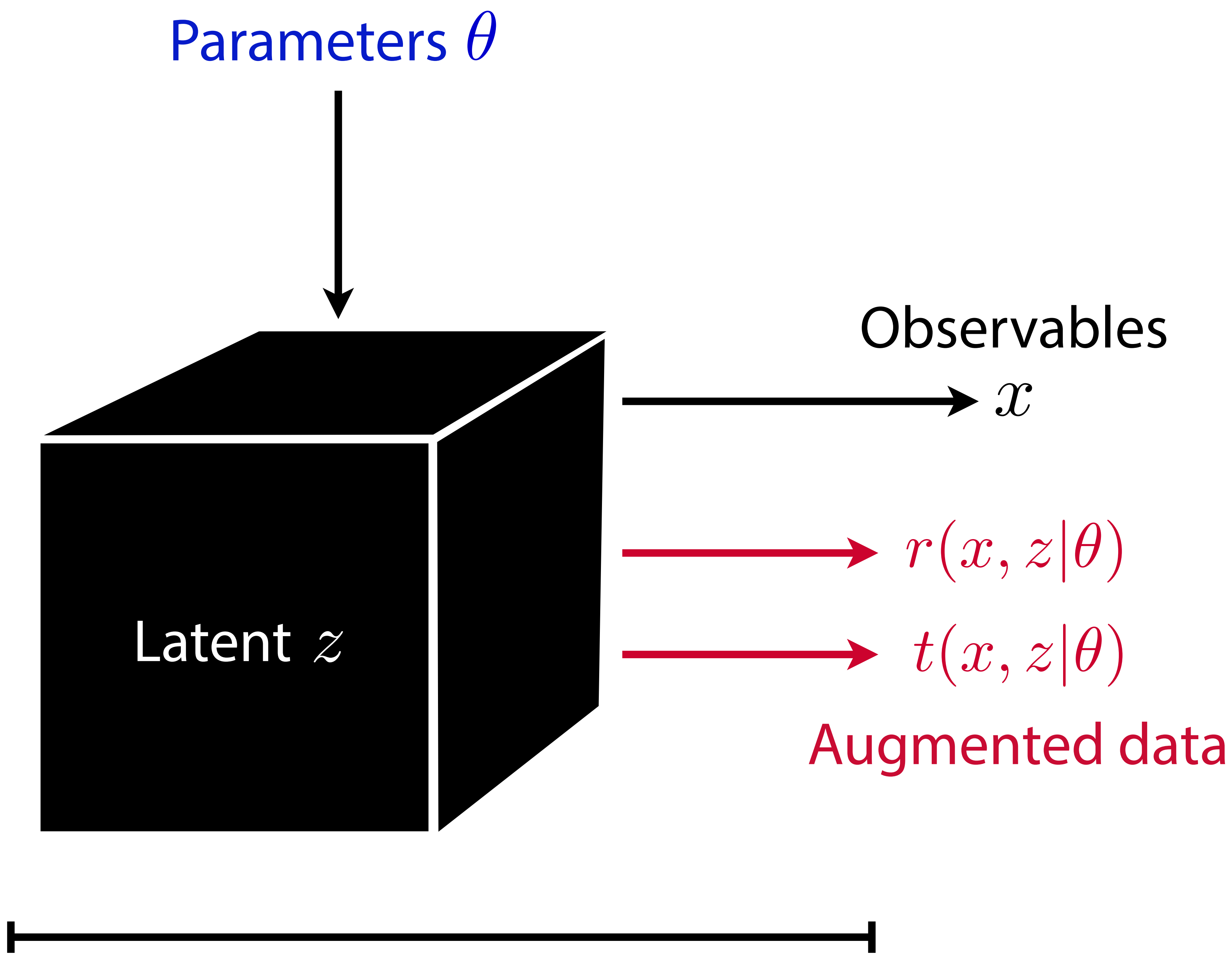
New!

- Mining gold from the simulator
  Leverage matrix-element information + machine learning

A new approach:
Mining gold from the simulator

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013, 1805.00020, 1805.12244;
with M. Stoye 1808.00973; with F. Kling in progress]

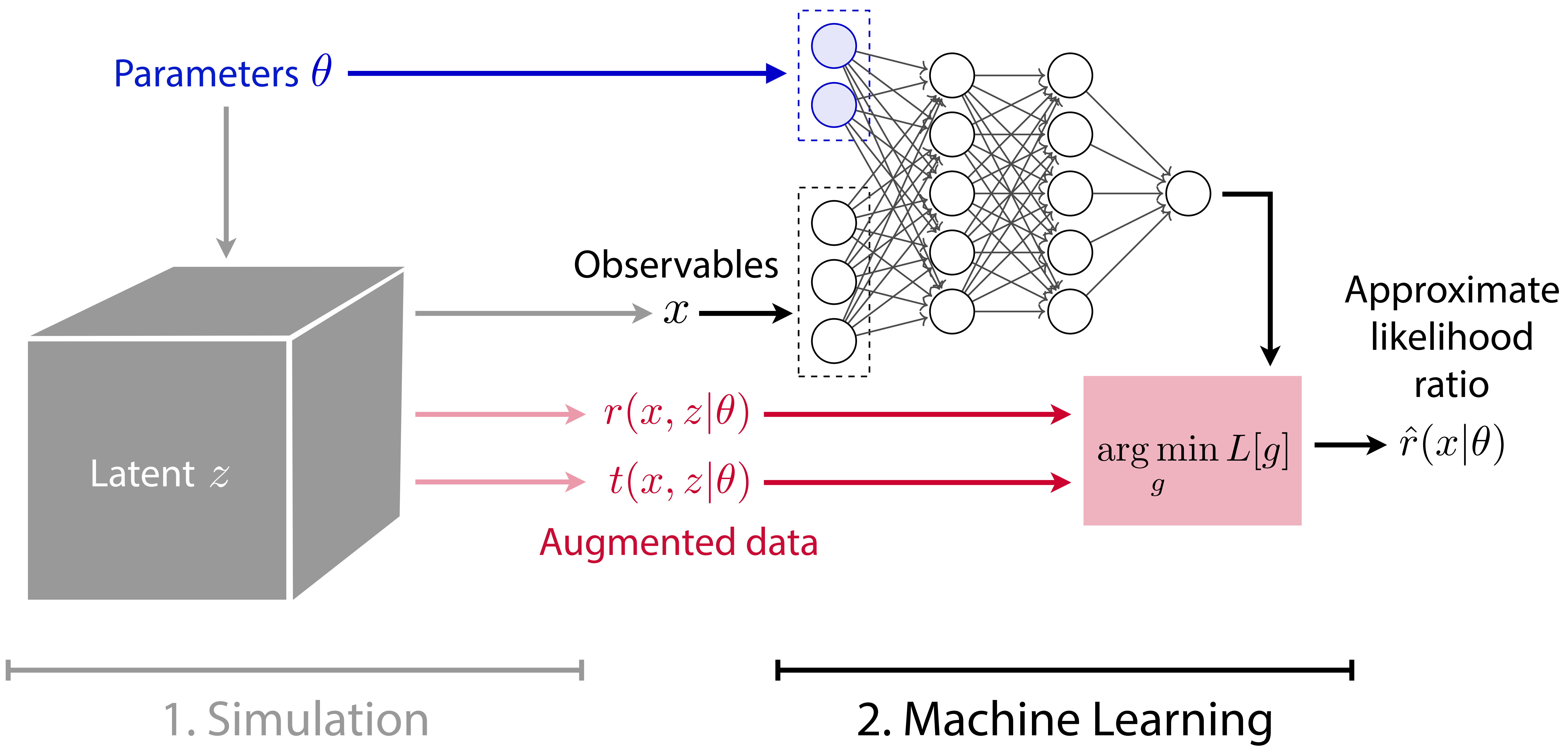
Bird's-eye view



1. Simulation

"Mining gold": Extract additional information from simulator

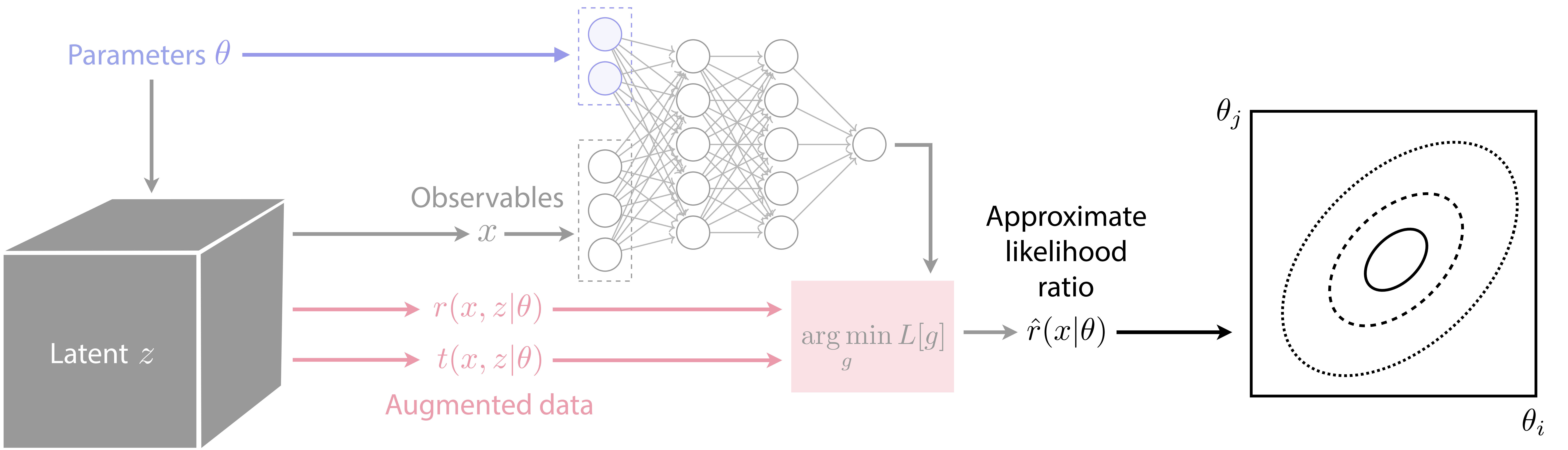
Bird's-eye view



"Mining gold": Extract additional information from simulator

Use this information to train estimator for likelihood ratio

Bird's-eye view



1. Simulation

"Mining gold": Extract additional information from simulator

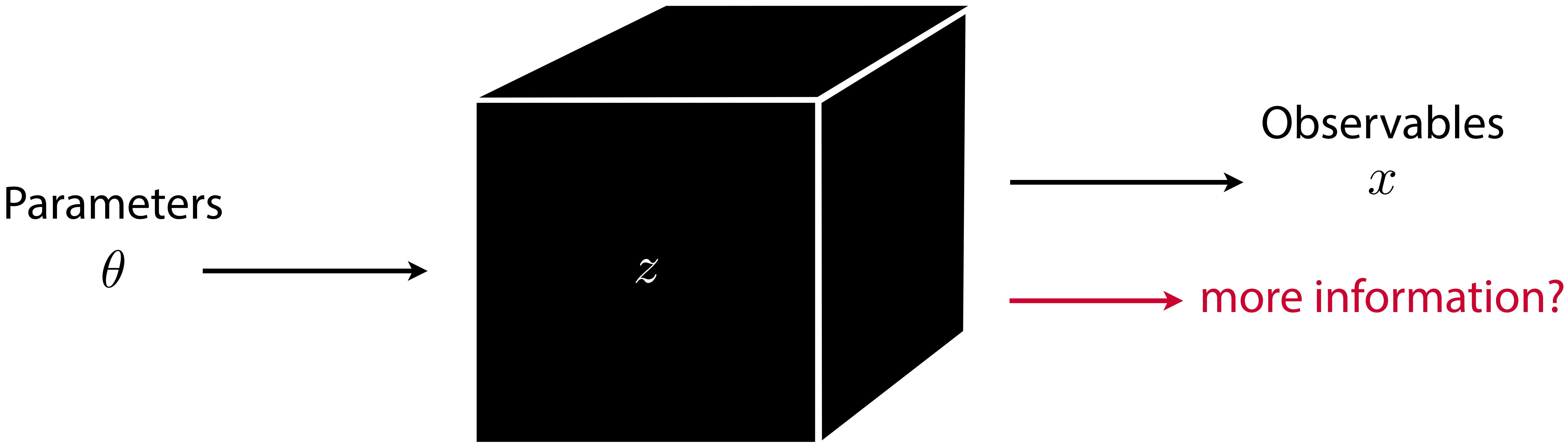
2. Machine Learning

Use this information to train estimator for likelihood ratio

3. Inference

Limit setting with standard hypothesis tests

Mining gold from the simulator



Mining gold from the simulator

Latent variables

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

$$p(x|\theta) = \boxed{\int \mathrm{d}z_d \int \mathrm{d}z_s \int \mathrm{d}z_p} \; p(x|z_d) \qquad p(z_d|z_s) \qquad p(z_s|z_p) \qquad p(z_p|\theta)$$

Intractable integrals

Mining gold from the simulator

Latent variables

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

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

$$p(x|\theta) = \int \mathrm{d}z_d \int \mathrm{d}z_s \int \mathrm{d}z_p \quad p(x|z_d) \quad p(z_d|z_s) \quad p(z_s|z_p) \quad \boxed{p(z_p|\theta)}$$

Parton-level likelihood can be evaluated!

⇒ For each generated event, we can calculate the joint likelihood ratio conditional on its specific evolution:

$$r(x, z|\theta_0, \theta_1) \equiv \frac{p(x, z_d, z_s, z_p|\theta_0)}{p(x, z_d, z_s, z_p|\theta_1)} = \frac{p(x|z_d)}{p(x|z_d)} \frac{p(z_d|z_s)}{p(z_d|z_s)} \frac{p(z_s|z_p)}{p(z_s|z_p)} \boxed{\frac{p(z_p|\theta_0)}{p(z_p|\theta_1)}}$$

The value of gold

We can calculate the joint likelihood ratio

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

$$\big\downarrow$$

We want the likelihood ratio function

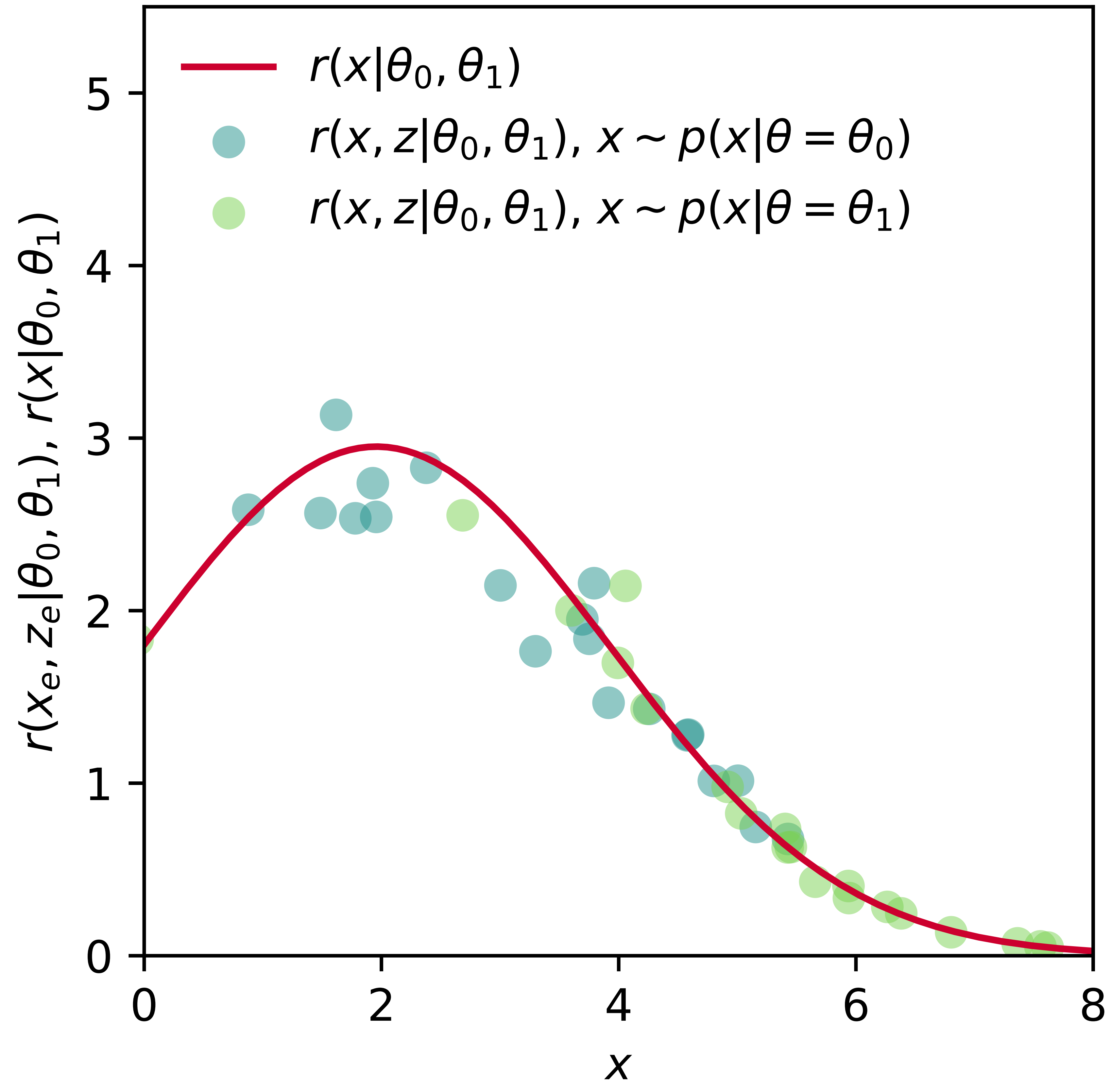
$$r(x|\theta_0, \theta_1) \equiv \frac{p(x|\theta_0)}{p(x|\theta_1)}$$

The value of gold

We can calculate the joint likelihood ratio

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

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



We want the likelihood ratio function

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

The value of gold

We can calculate the joint likelihood ratio

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

With $r(x, z|\theta_0, \theta_1)$, we define the functional

$$L_r[\hat{r}(x|\theta_0, \theta_1)] = \int \mathrm{d}x \int \mathrm{d}z \; p(x, z|\theta_1) \left[\left(\hat{r}(x|\theta_0, \theta_1) - r(x, z|\theta_0, \theta_1) \right)^2 \right].$$

One can show it is minimized by

$$r(x|\theta_0, \theta_1) = \underset{\hat{r}(x|\theta_0, \theta_1)}{\arg\min} \, L_r[\hat{r}(x|\theta_0, \theta_1)]!$$

We want the likelihood ratio function

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

Enter machine learning

So to get a good estimator of the likelihood ratio, we need to mimimize a functional numerically:

Variational family

Extremization

Functional with integral

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

Enter machine learning

So to get a good estimator of the likelihood ratio, we need to mimimize a functional numerically:

Variational family

Extremization

Functional with integral

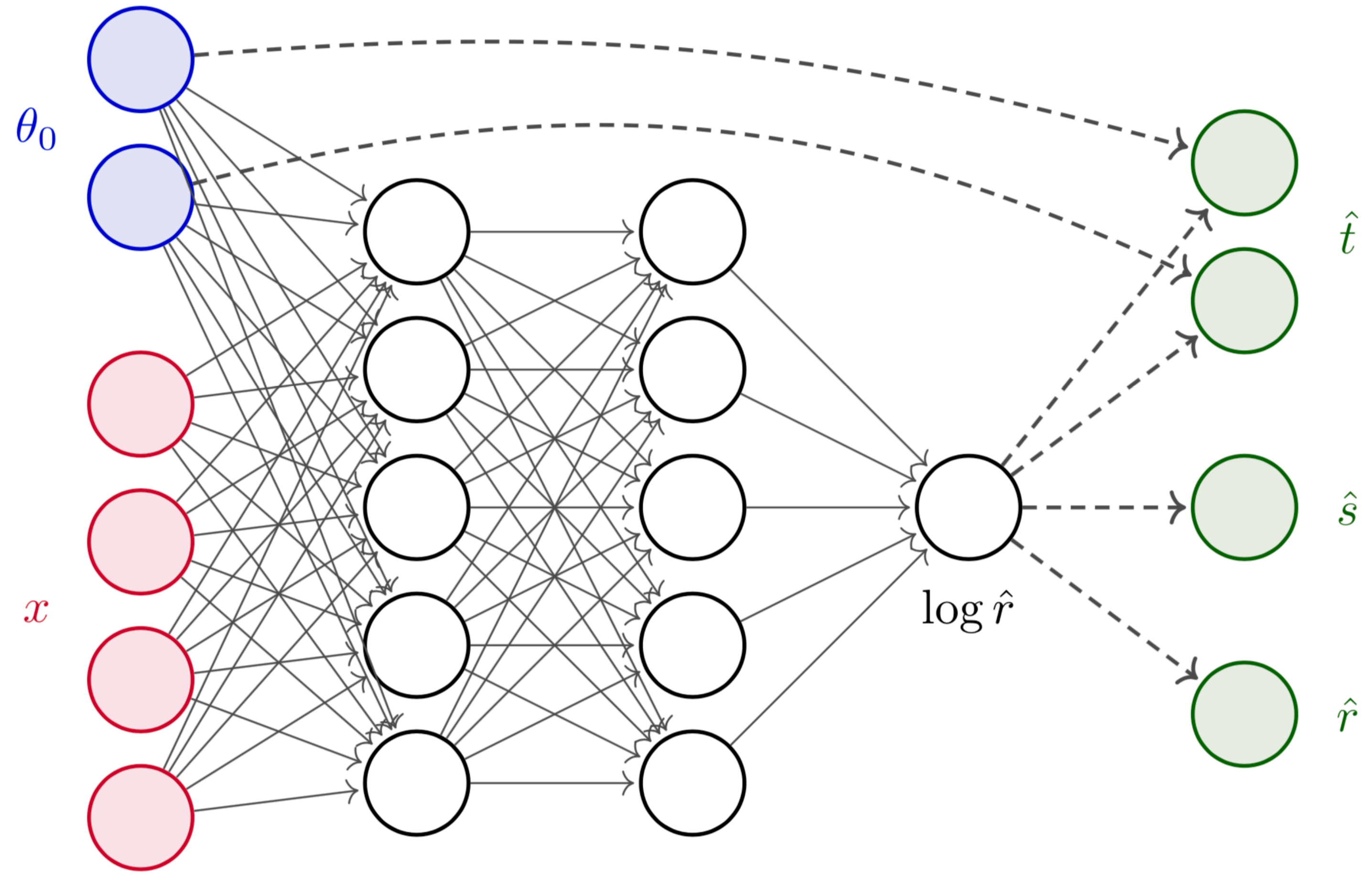
$$r(x|\theta_0,\theta_1) = \arg\min_{\hat{r}(x|\theta_0,\theta_1)} L_r[\hat{r}(x|\theta_0,\theta_1)]$$

Loss function with finite sum over samples

Stochastic gradient descent

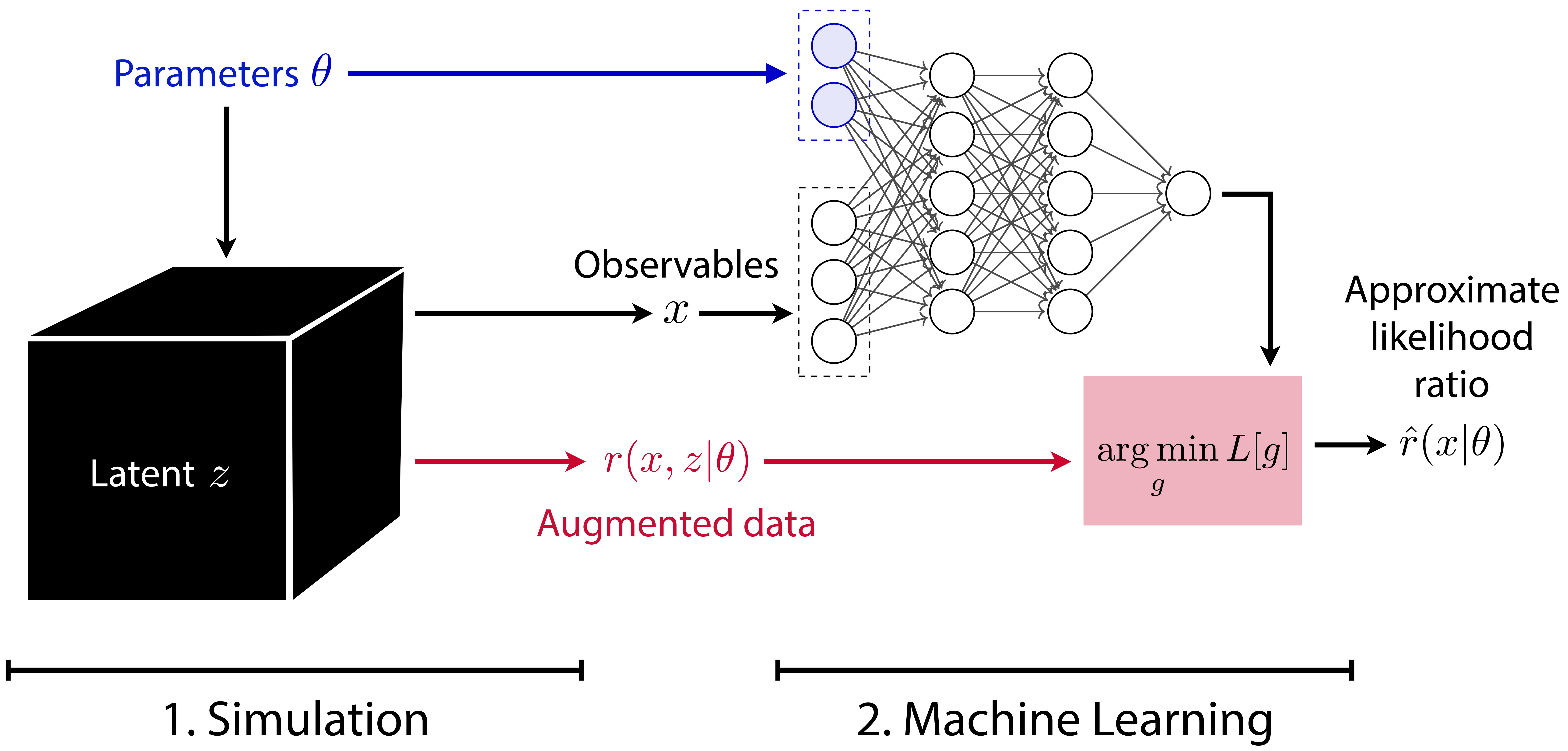
This is where machine learning comes in!

Neural network



$\Rightarrow$ We implement $\hat{r}(x|\theta_0,\theta_1)$ as a neural network trained on the data available from the simulator

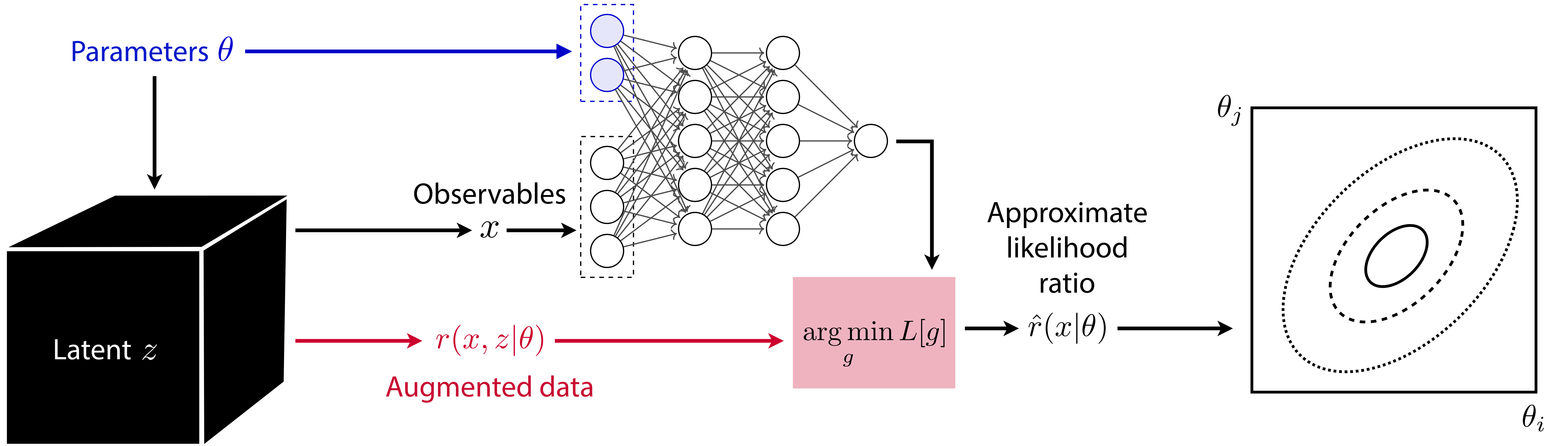
What we have so far



"Mining gold": Extract additional information from simulator

Use this information to train estimator for likelihood ratio

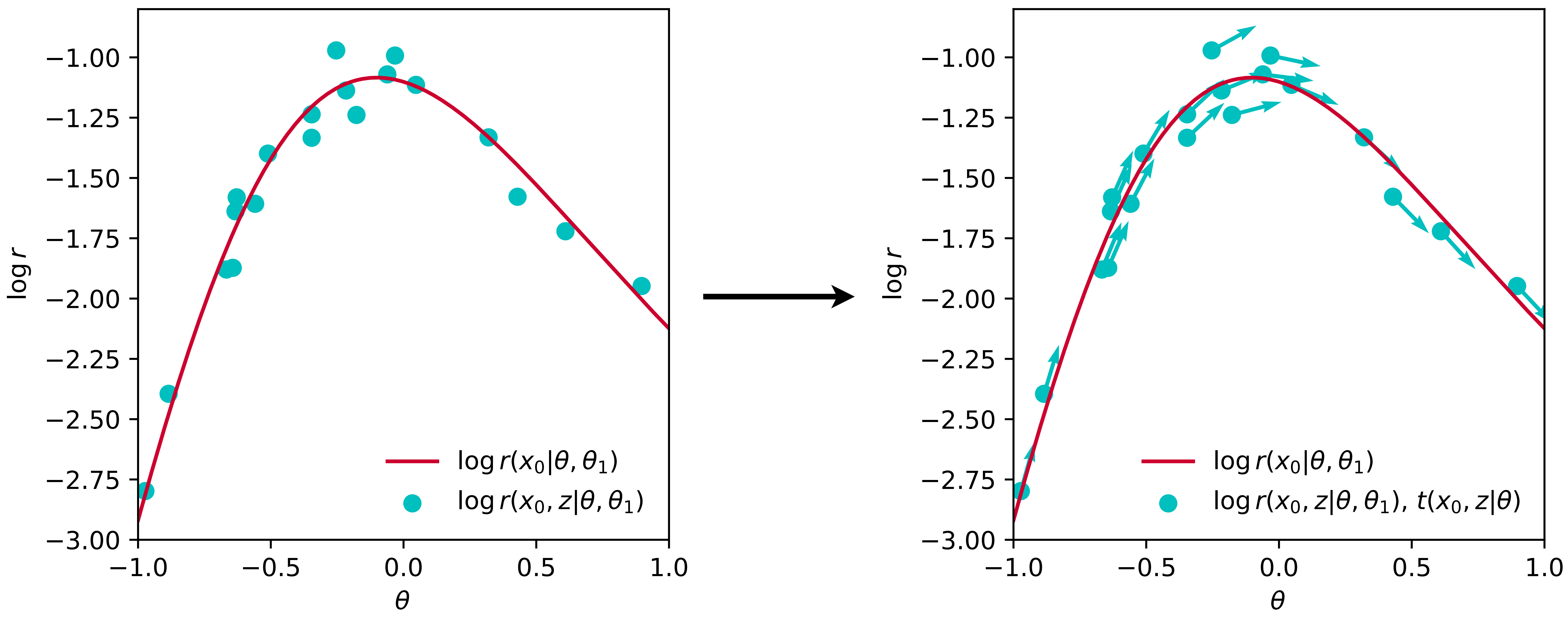
What we have so far



1. Simulation	2. Machine Learning	3. Inference
"Mining gold": Extract additional information from simulator	Use this information to train estimator for likelihood ratio	Limit setting with standard hypothesis tests

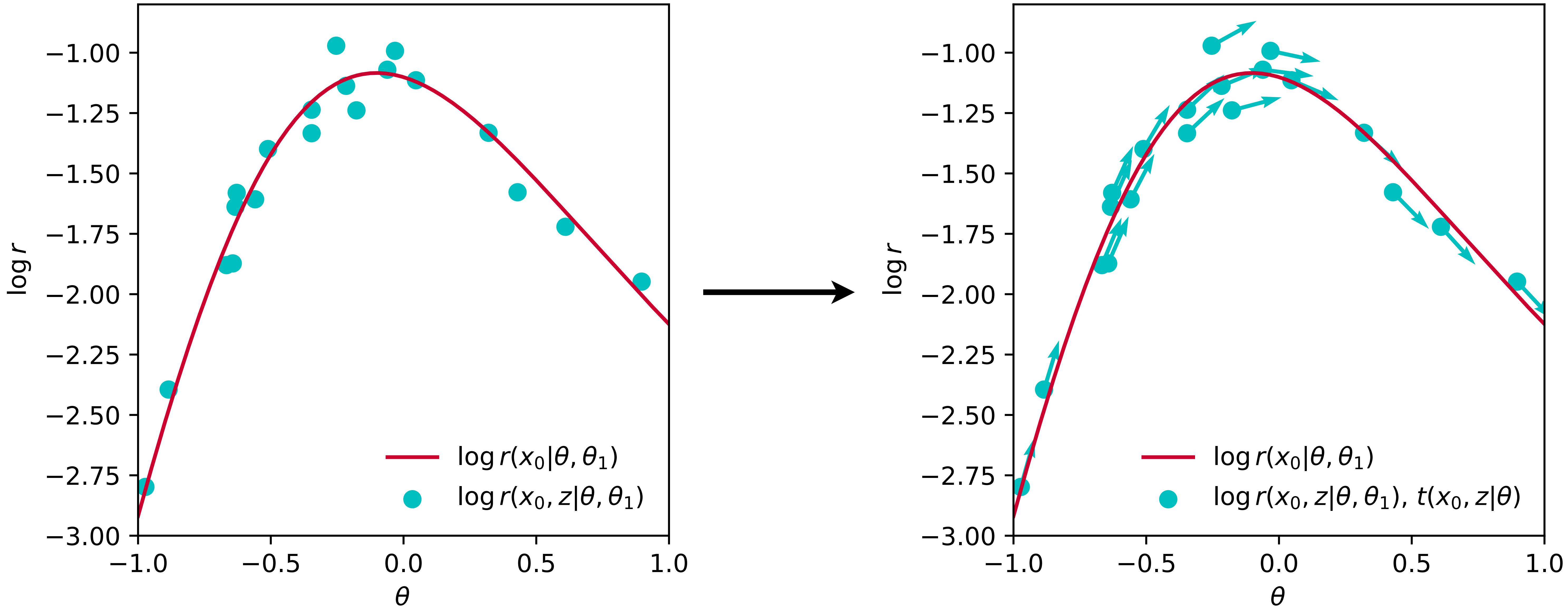
One more piece: the score

- Knowing derivative often helps fitting:



One more piece: the score

- Knowing derivative often helps fitting:



- In our case, the relevant quantity is the score $t(x|\theta_0) \equiv \nabla_\theta \log p(x|\theta)\Big|_{\theta_0}$.
  - The score fully characterizes the likelihood function in the neighborhood of θ_0
  - The score itself is intractable. But…

Learning the score

Similar to the joint likelihood ratio, from the simulator we can extract the joint score

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

$$\downarrow$$

We want the score

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

Learning the score

Similar to the joint likelihood ratio, from the simulator we can extract the joint score

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

We want the score

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

Given $t(x, z|\theta_0)$,
we define the functional

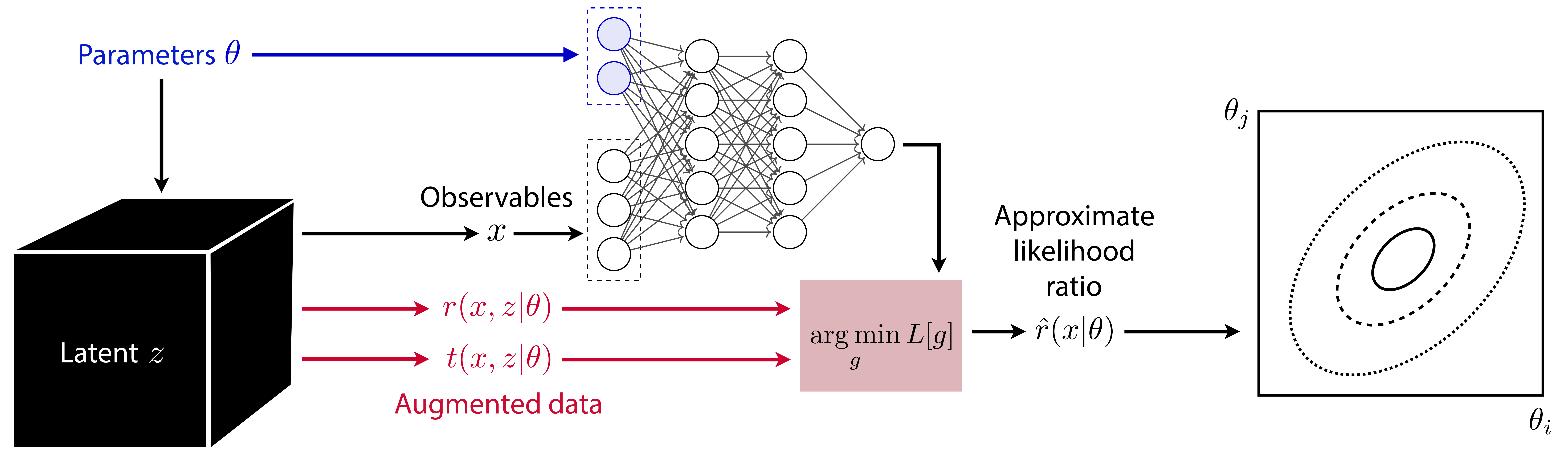
$$L_t[\hat{t}(x|\theta_0)] = \int \mathrm{d}x \int \mathrm{d}z \;\; p(x, z|\theta_0) \; \Big[\big(\hat{t}(x|\theta_0) - t(x, z|\theta_0)\big)^2\Big].$$

One can show it is minimized by

$$t(x|\theta_0) = \underset{\hat{t}(x|\theta_0)}{\arg\min}\, L_t[\hat{t}(x|\theta_0)] \;\; !$$

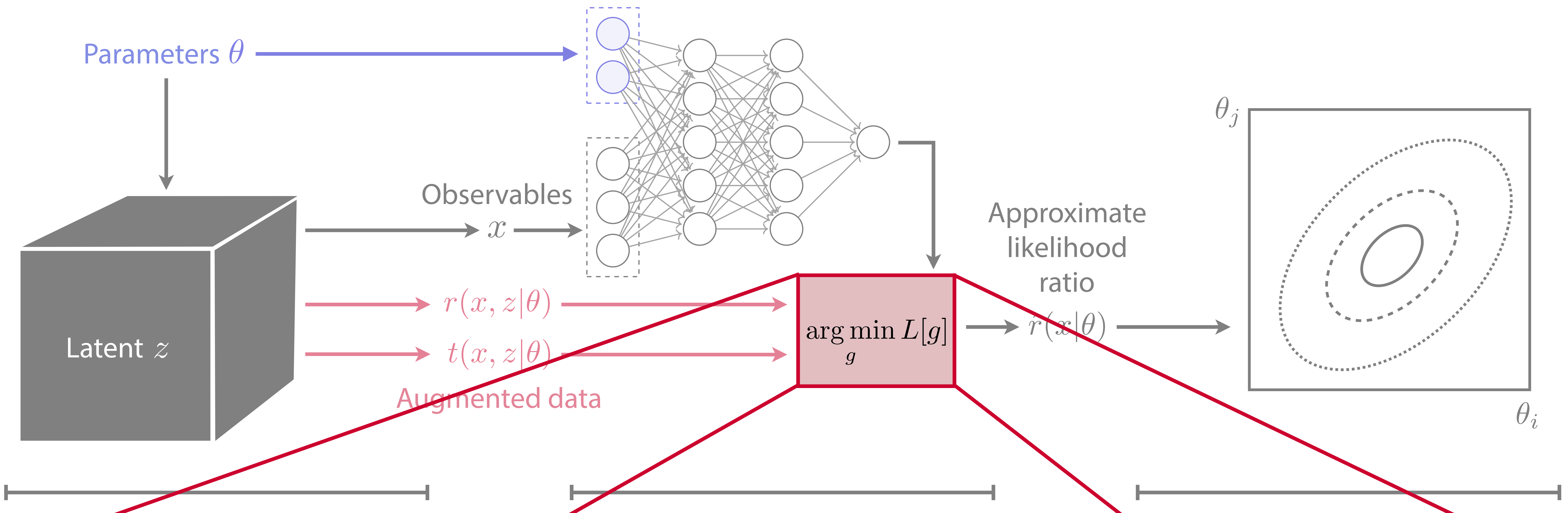
Again, we implement this minimization through machine learning

Putting the pieces together: RASCAL (Ratio and score approximate likelihood ratio)



1. Simulation	2. Machine Learning	3. Inference
"Mining gold": Extract additional information from simulator	Use this information to train estimator for likelihood ratio	Limit setting with standard hypothesis tests

Putting the pieces together: RASCAL (Ratio and score approximate likelihood ratio)

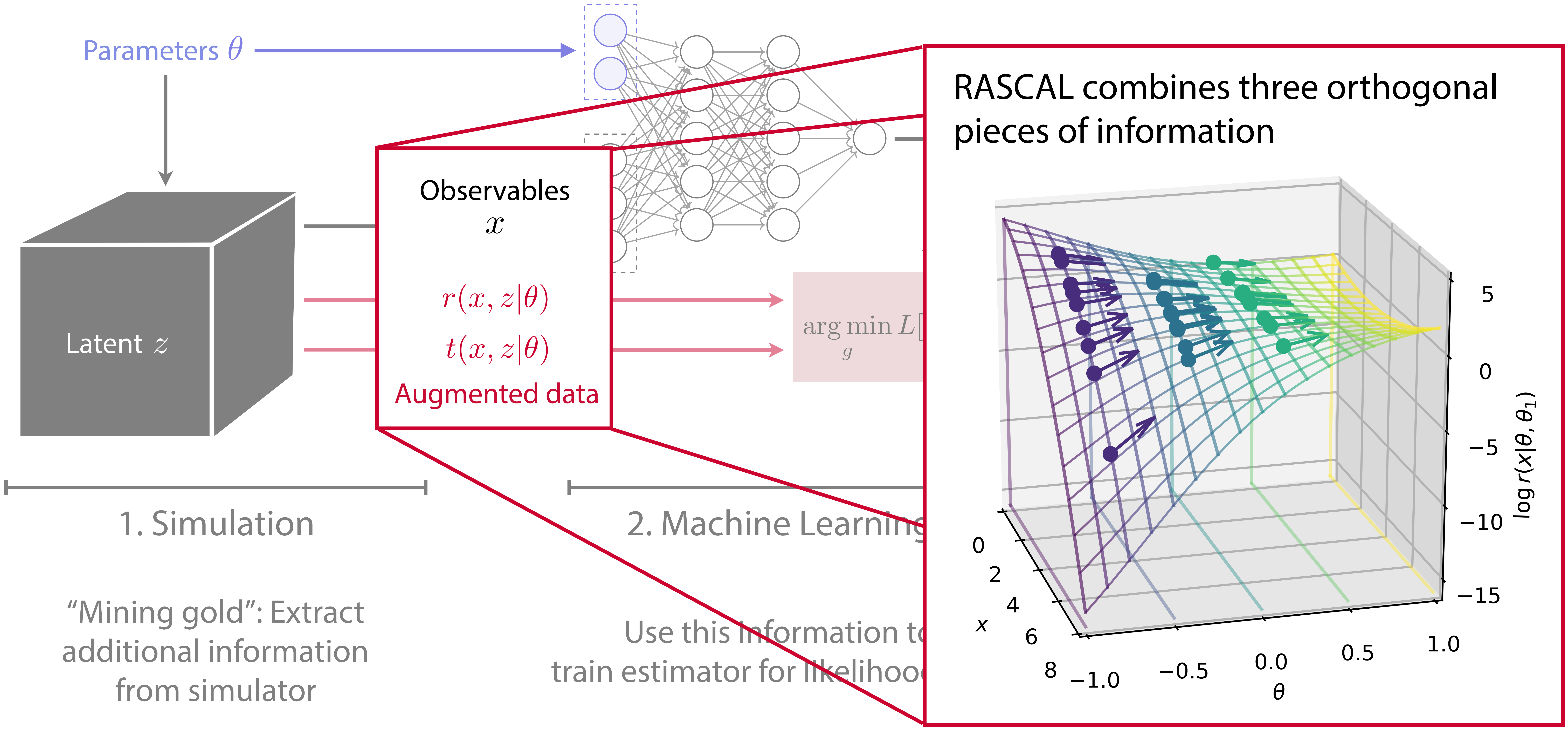


Combining ratio and score information:

- If ratio estimator is differentiable, we can calculate the estimated score $\hat{t}(x|\theta_0) = \nabla_{\theta_0} \log \hat{r}(x|\theta_0, \theta_1)$
- Minimize combined loss function:

$$L_{\mathrm{RASCAL}}[\hat{r}(x|\theta_0, \theta_1)] = L_r[\hat{r}(x|\theta_0, \theta_1)] + \alpha\, L_t[\nabla_{\theta_0} \log \hat{r}(x|\theta_0, \theta_1)]$$

Putting the pieces together: RASCAL (Ratio and score approximate likelihood ratio)

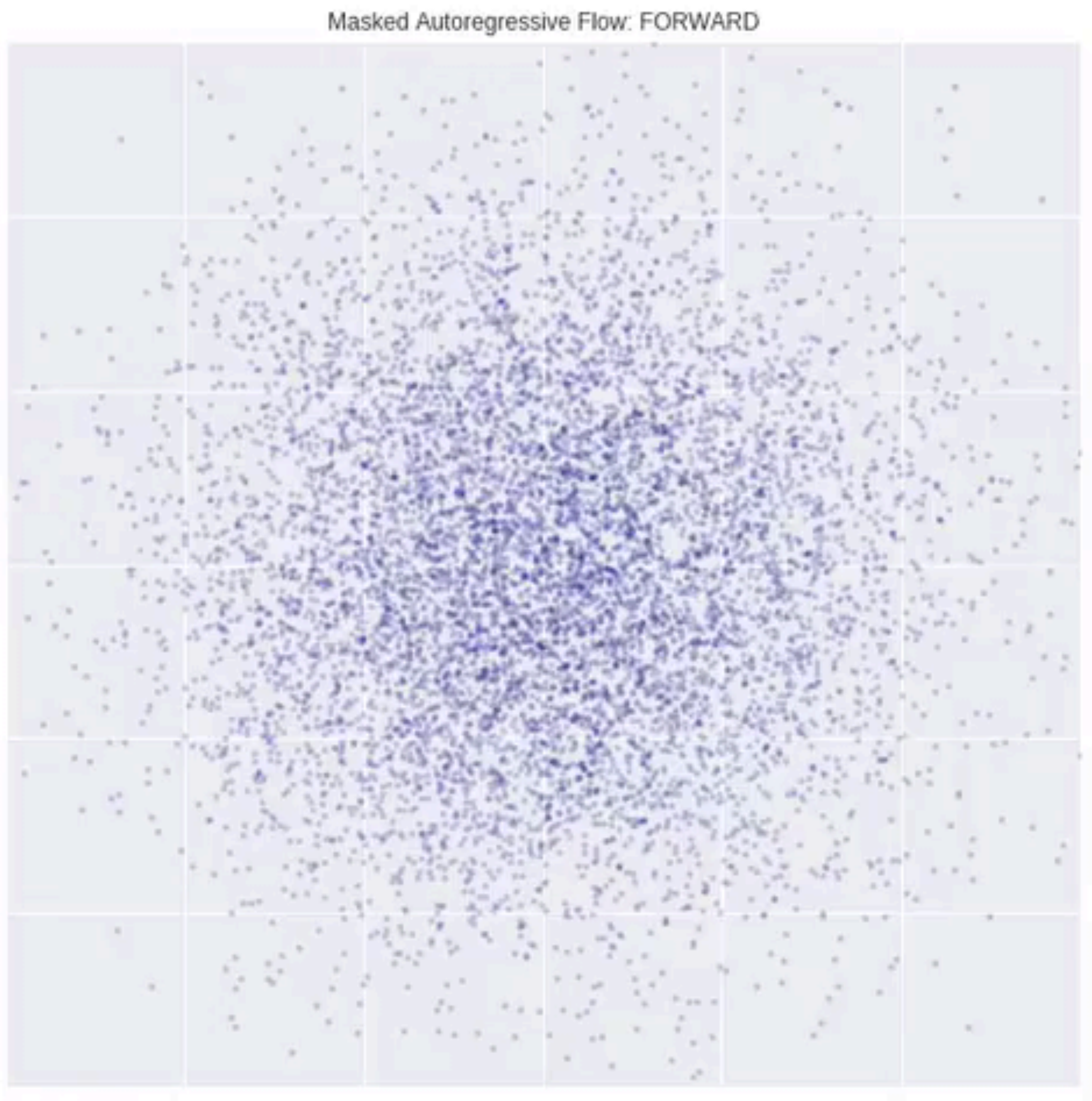


"Mining gold": Extract additional information from simulator

Use this information to train estimator for likelihood

Alternatives and extensions

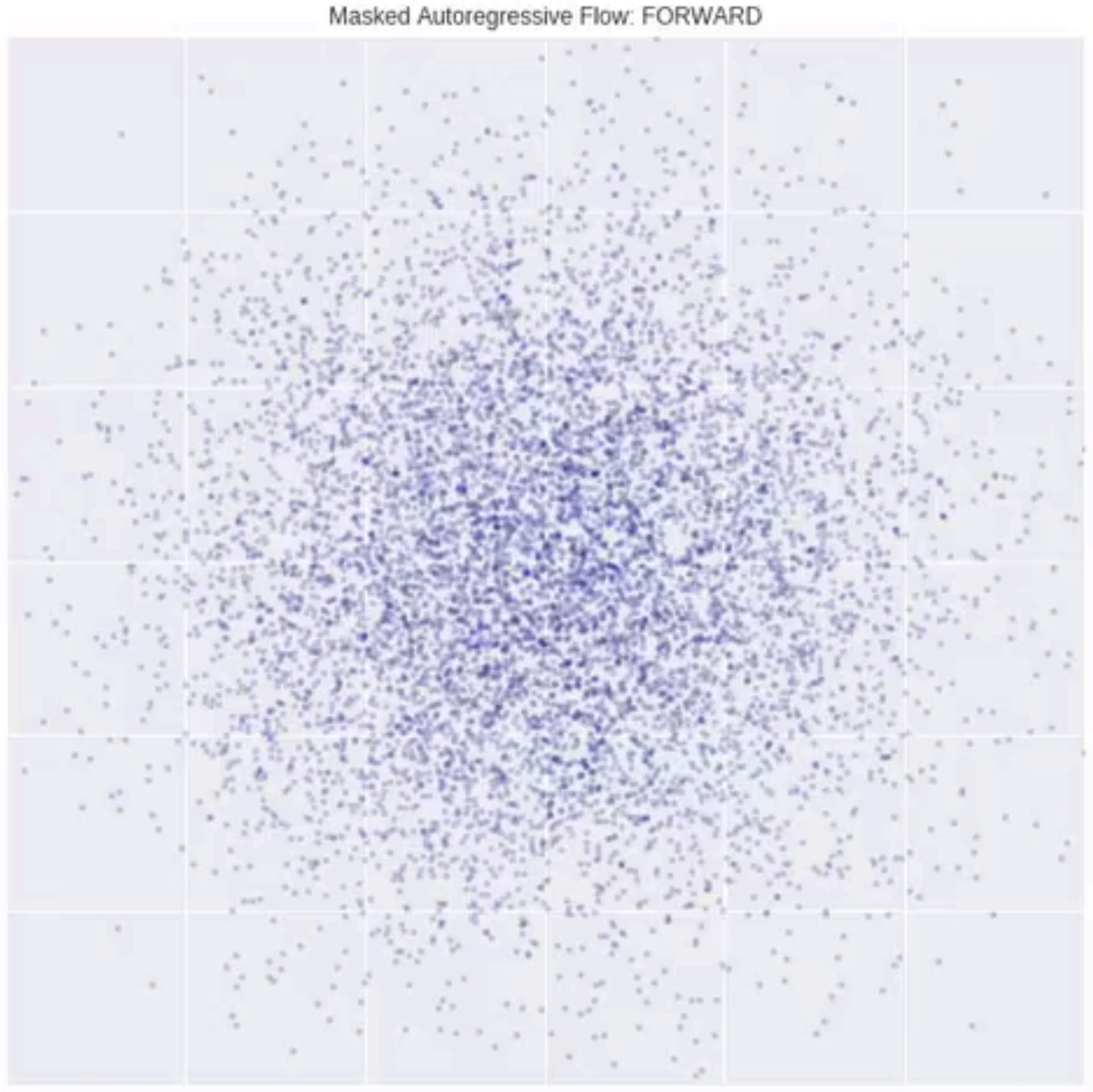
- More than one way to the likelihood (ratio)!
  - SCANDAL: combine with neural density estimators, e.g. Masked Autoregressive Flows
    [G. Papamakarios, T. Pavlakou, I. Murray 1705.07057]
  - SALLY / SALLINO: use estimated score as "optimal observable"



[Alex Mordvintsev]

Alternatives and extensions

- More than one way to the likelihood (ratio)!
  - SCANDAL: combine with neural density estimators, e.g. Masked Autoregressive Flows
    [G. Papamakarios, T. Pavlakou, I. Murray 1705.07057]
  - SALLY / SALLINO: use estimated score as "optimal observable"

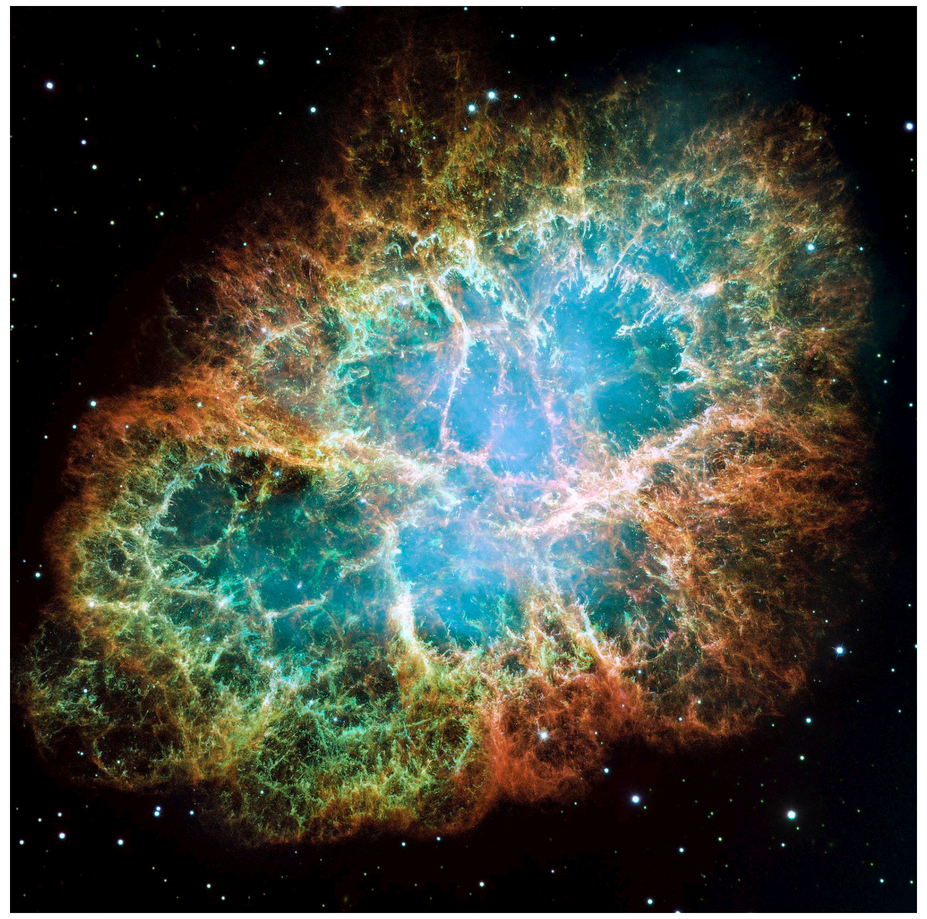


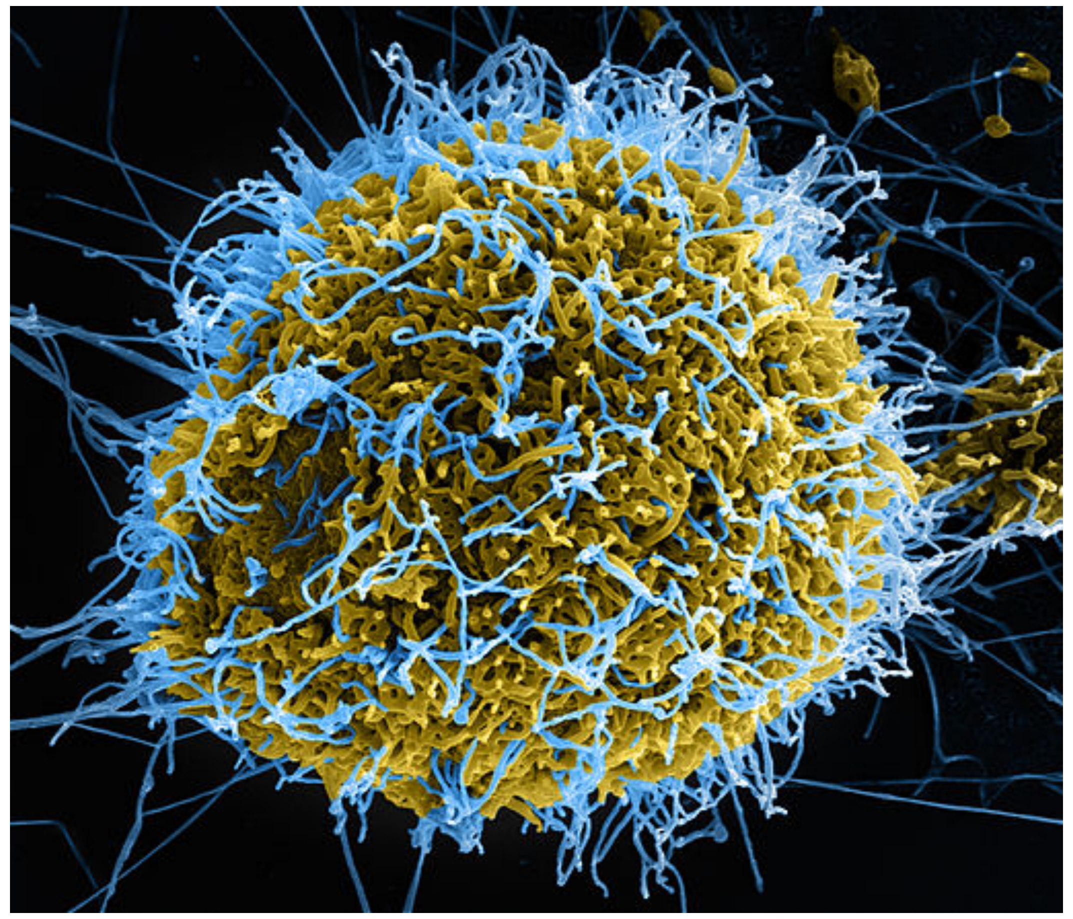
[Alex Mordvintsev]

- What if we don't fully trust the simulator?

  ⇒ Nuisance parameters / systematic uncertainties

- More general than particle physics
  - Currently being adapted to cosmology and epidemiology
  - Bayesian inference





[NASA, NIAID]

Comparison with established methods

Treat simulator as black box:

- Histograms of observables, Approximate Bayesian Computation
  Rely on summary statistics
- Machine learning techniques
  Density networks, CARL, autoregressive models, normalizing flows, …

Use latent structure:

- Matrix Element Method, Optimal Observables, Shower Deconstruction
  Neglect or approximate shower + detector, explicitly calculate $\approx$ integral
- New! Mining gold from the simulator
  Leverage matrix-element information + machine learning

	Histograms, ABC	Neural density est.	Matrix-Element Method	RASCAL etc
High-dimensional observables		✓	✓	✓
Realistic shower, detector sim.	✓	✓	transfer fns.	✓
Uses matrix element information			✓	✓
Evaluation	fast	fast	expensive	fast


DEGUSSA
FEINGOLD
999.9

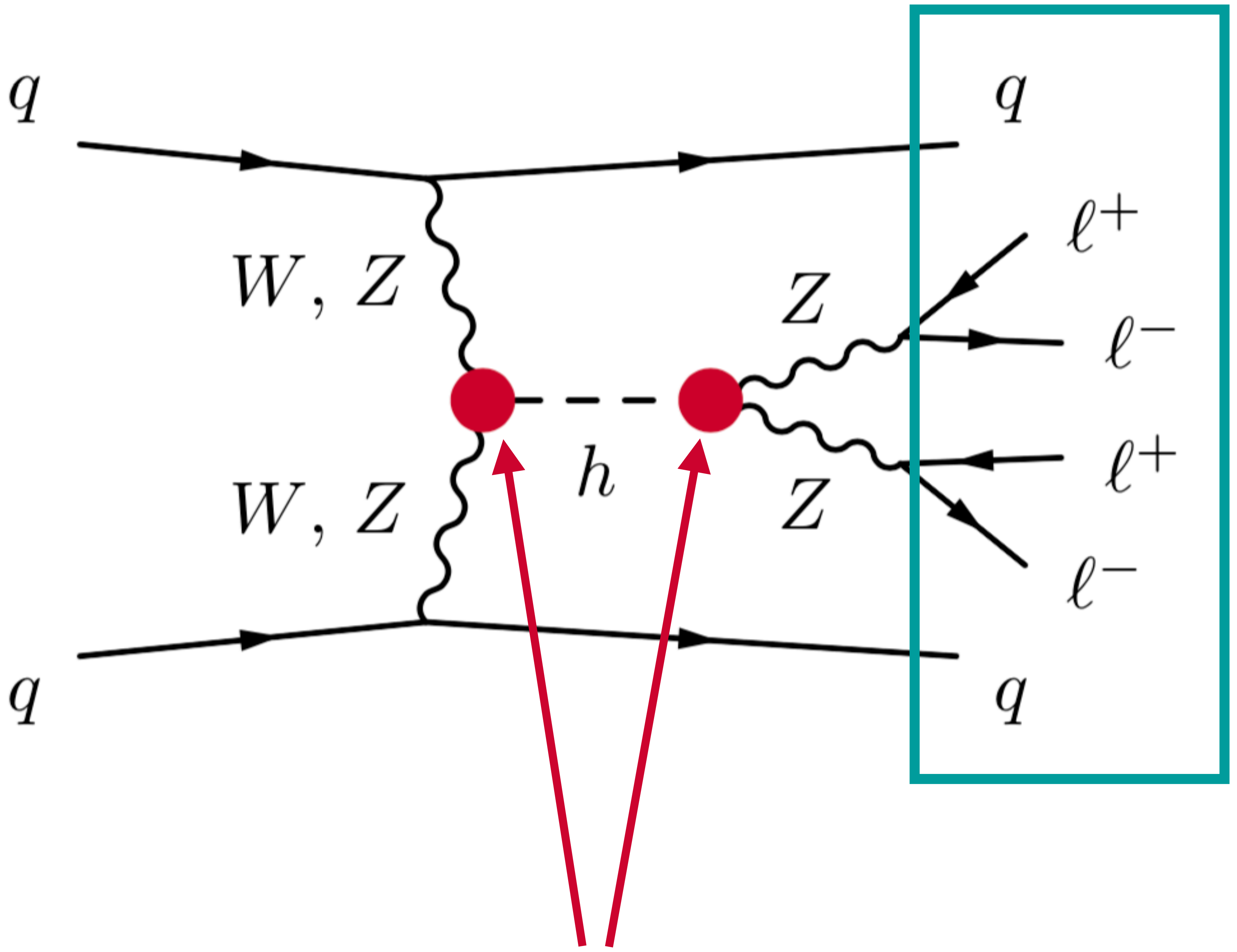
EFT example

[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013, 1805.00020, 1805.12244;
with M. Stoye 1808.00973]

Proof of concept

Higgs production in weak boson fusion:

at least 16-dimensional
observable space

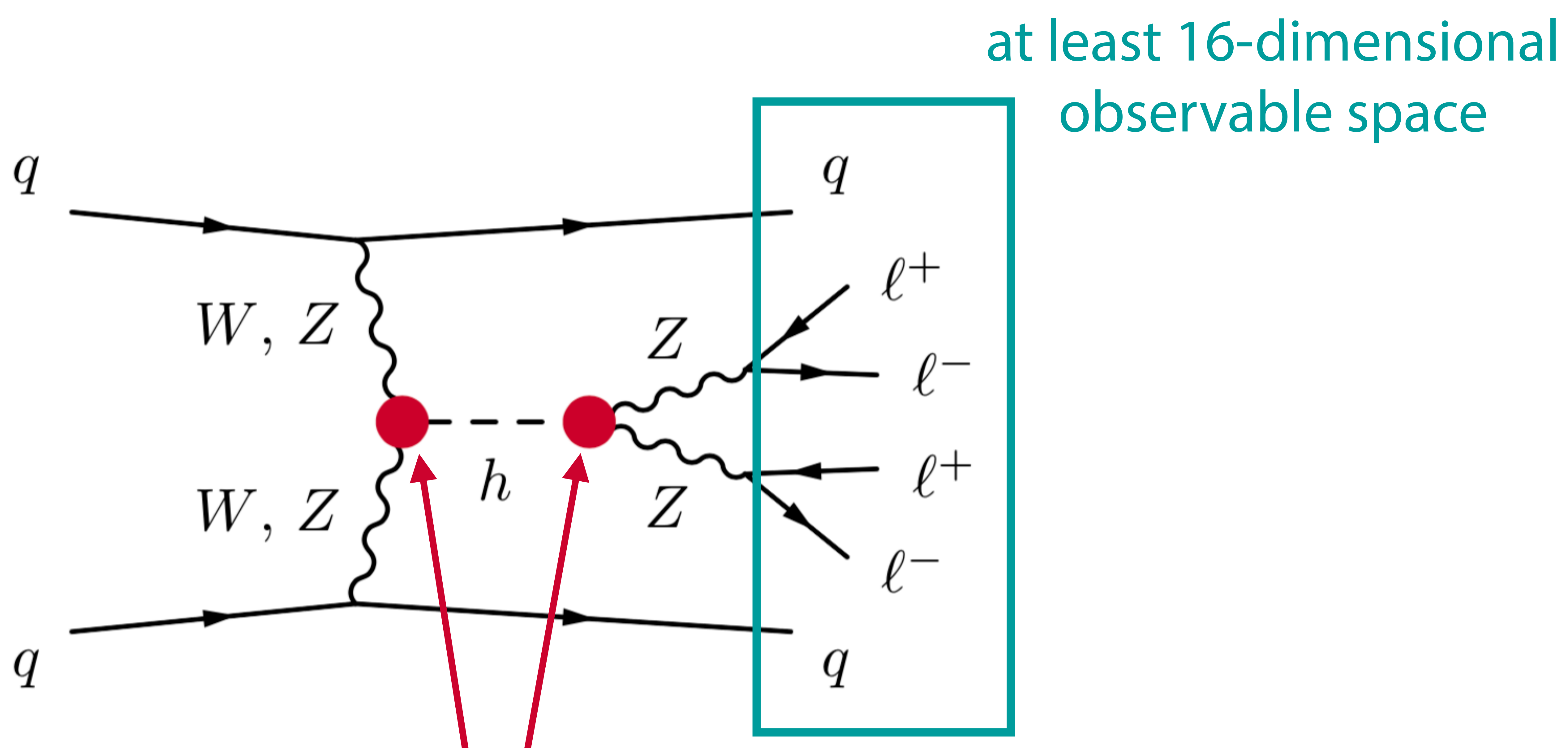


Exciting new physics might hide here!
We parameterize it with two EFT coefficients:

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

Proof of concept

Higgs production in weak boson fusion:



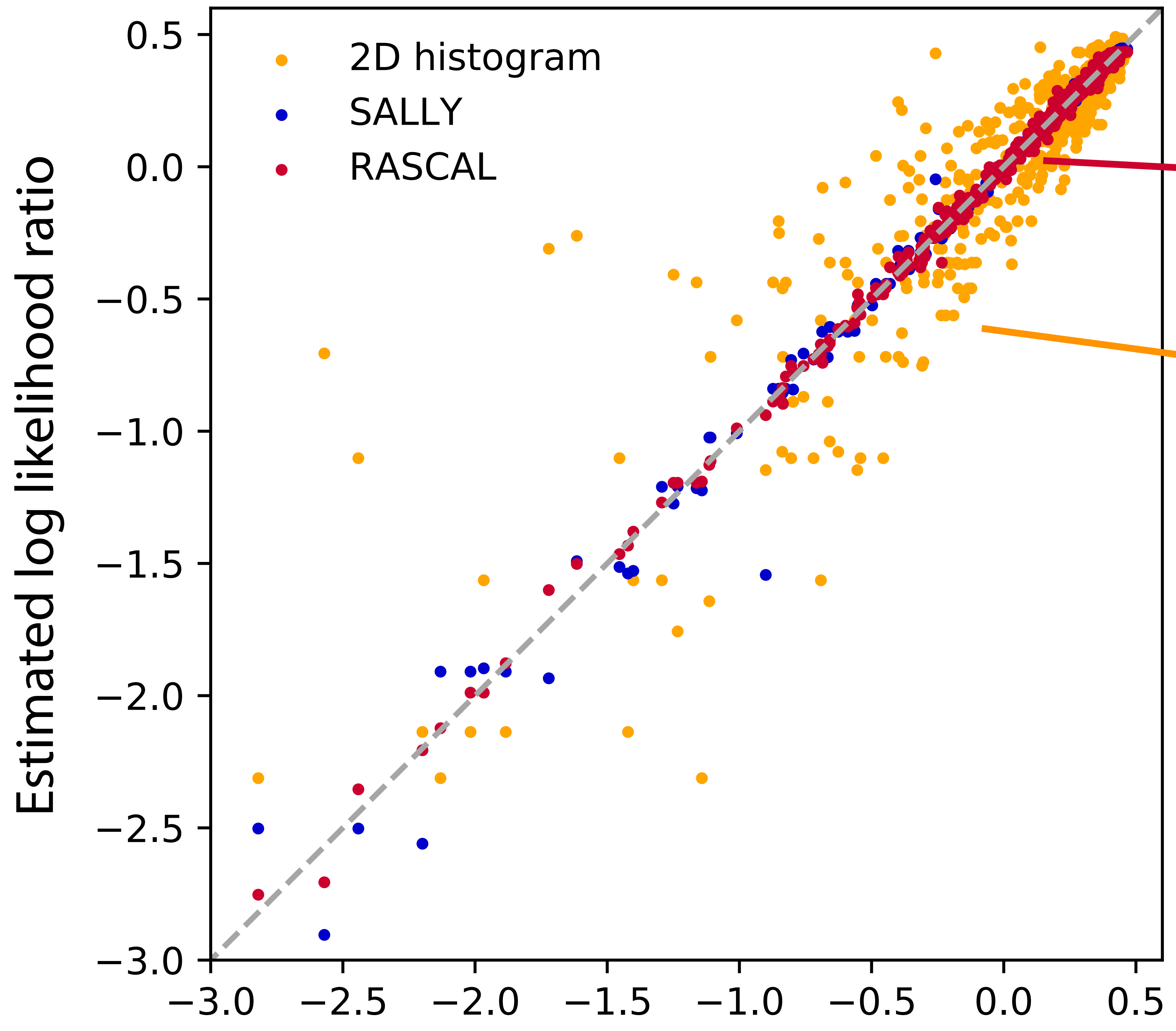
Exciting new physics might hide here!
We parameterize it with two EFT coefficients:

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

- Goal: constrain the two EFT parameters
  - new inference methods
  - baseline: 2d histogram analysis of jet momenta & angular correlations
- Two scenarios:
  - Simplified setup in which we can compare to true likelihood
  - "Realistic" simulation with approximate detector effects
- Simulation: MadGraph + MadMax
  [J. Alwall et al. 1405.0301; K. Cranmer, T. Plehn hep-ph/0605268; T. Plehn, P. Schichtel, D. Wiegand 1311.2591]

More precise likelihood ratio estimates with less training data

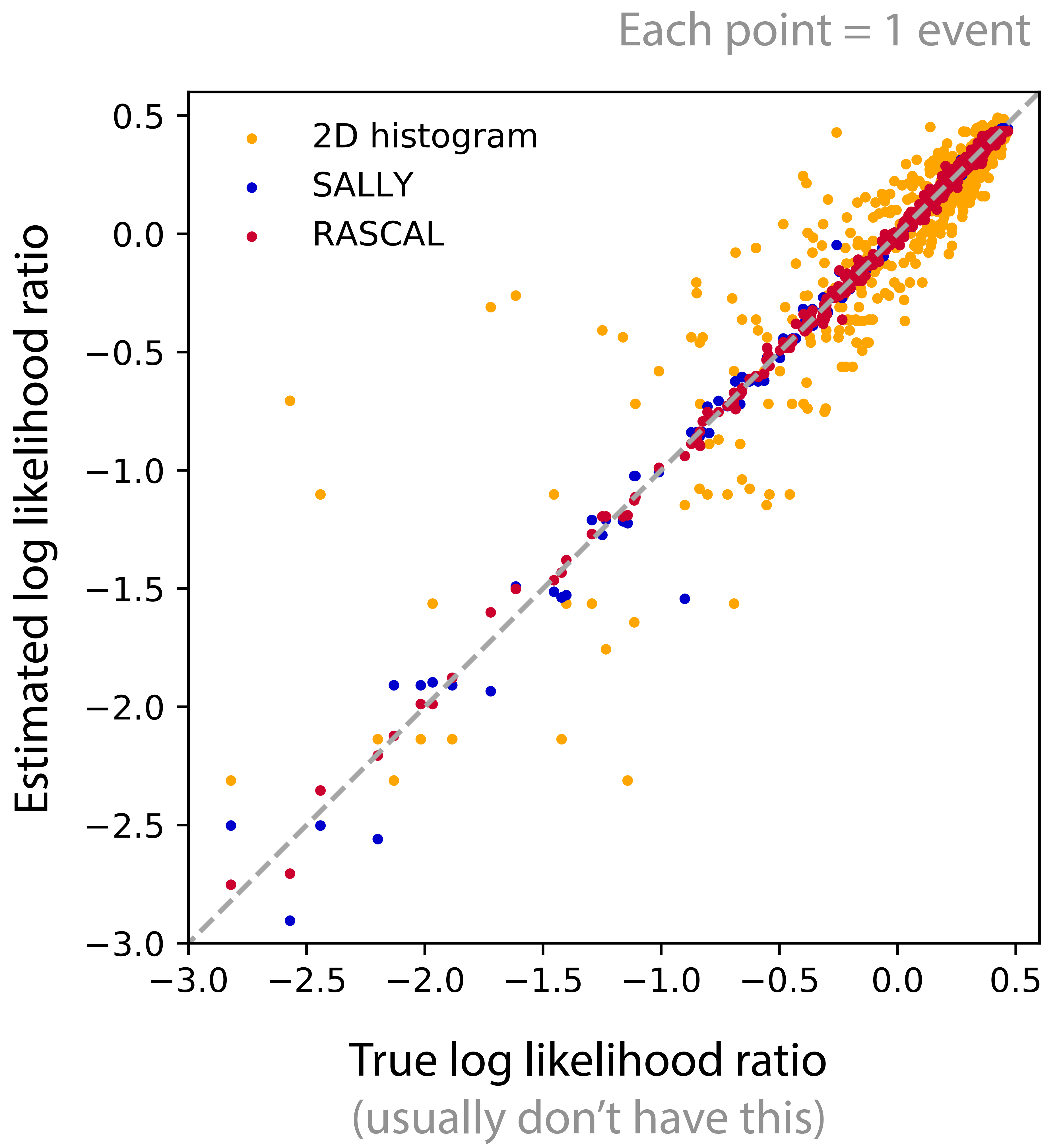
Each point = 1 event

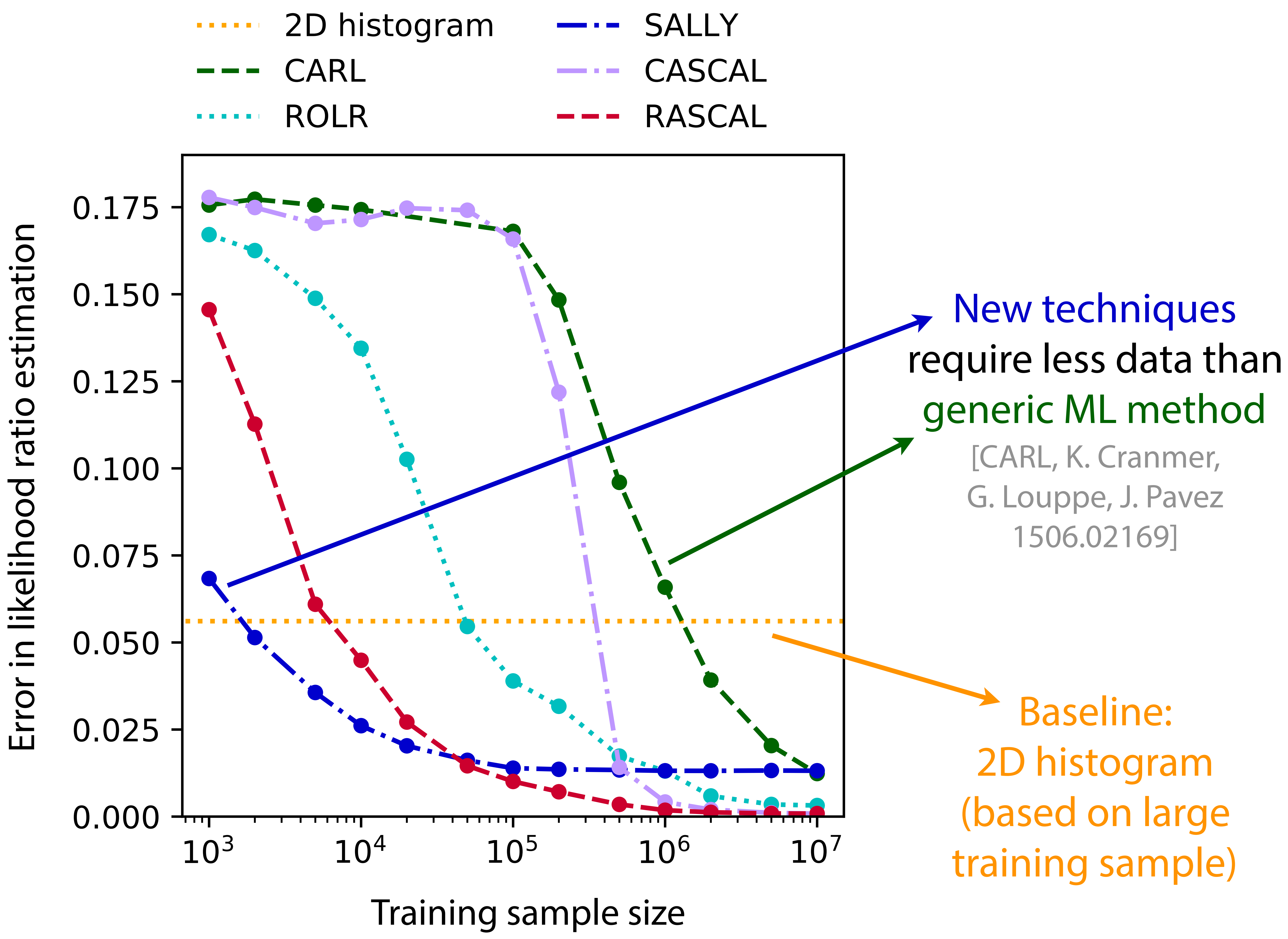


True log likelihood ratio
(usually don't have this)

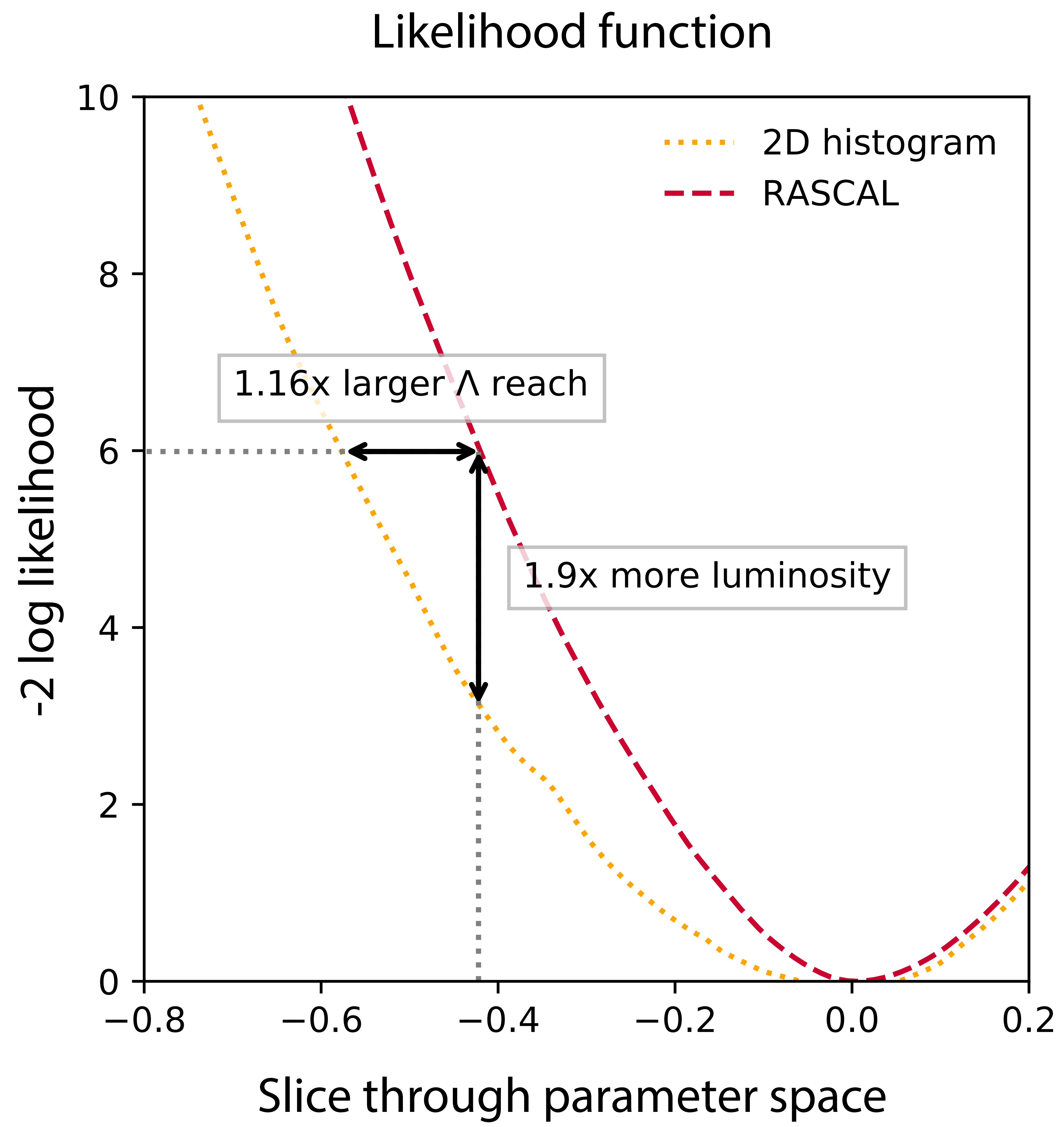
RASCAL
gives more precise likelihood ratio estimates than the
2D histogram

More precise likelihood ratio estimates with less training data



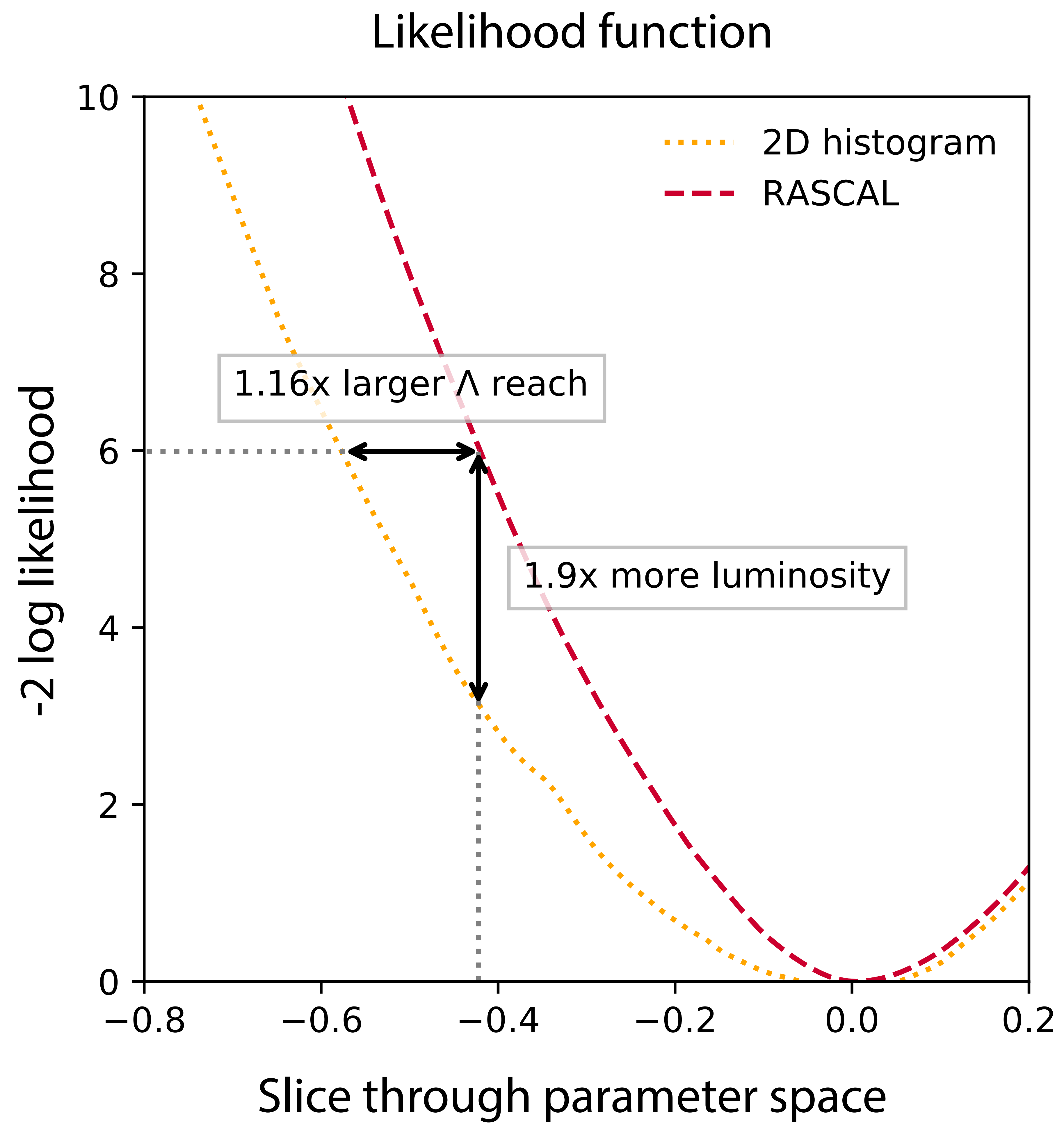


Better sensitivity to new physics

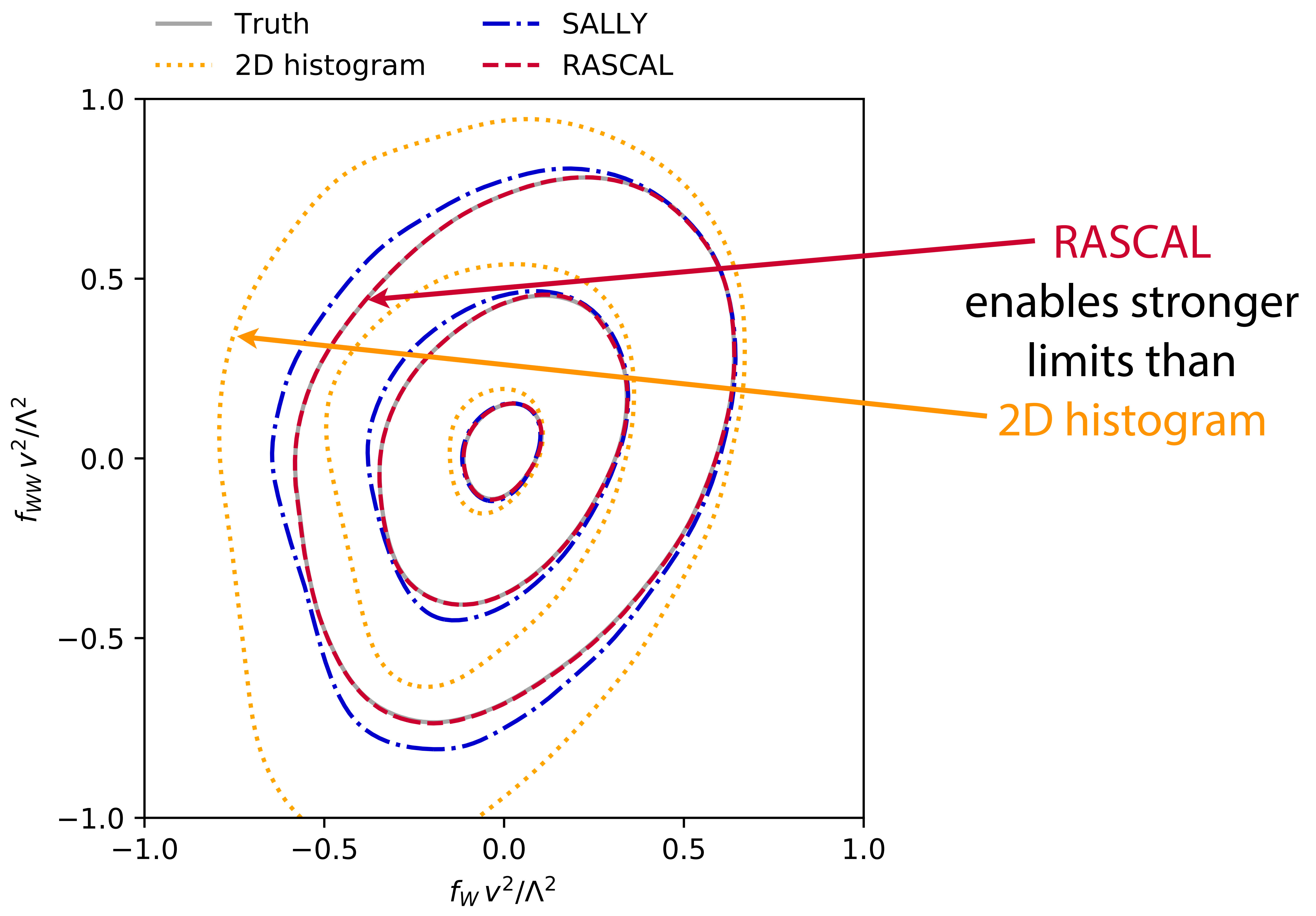


Results are based on 36 observed events, assuming SM

Better sensitivity to new physics

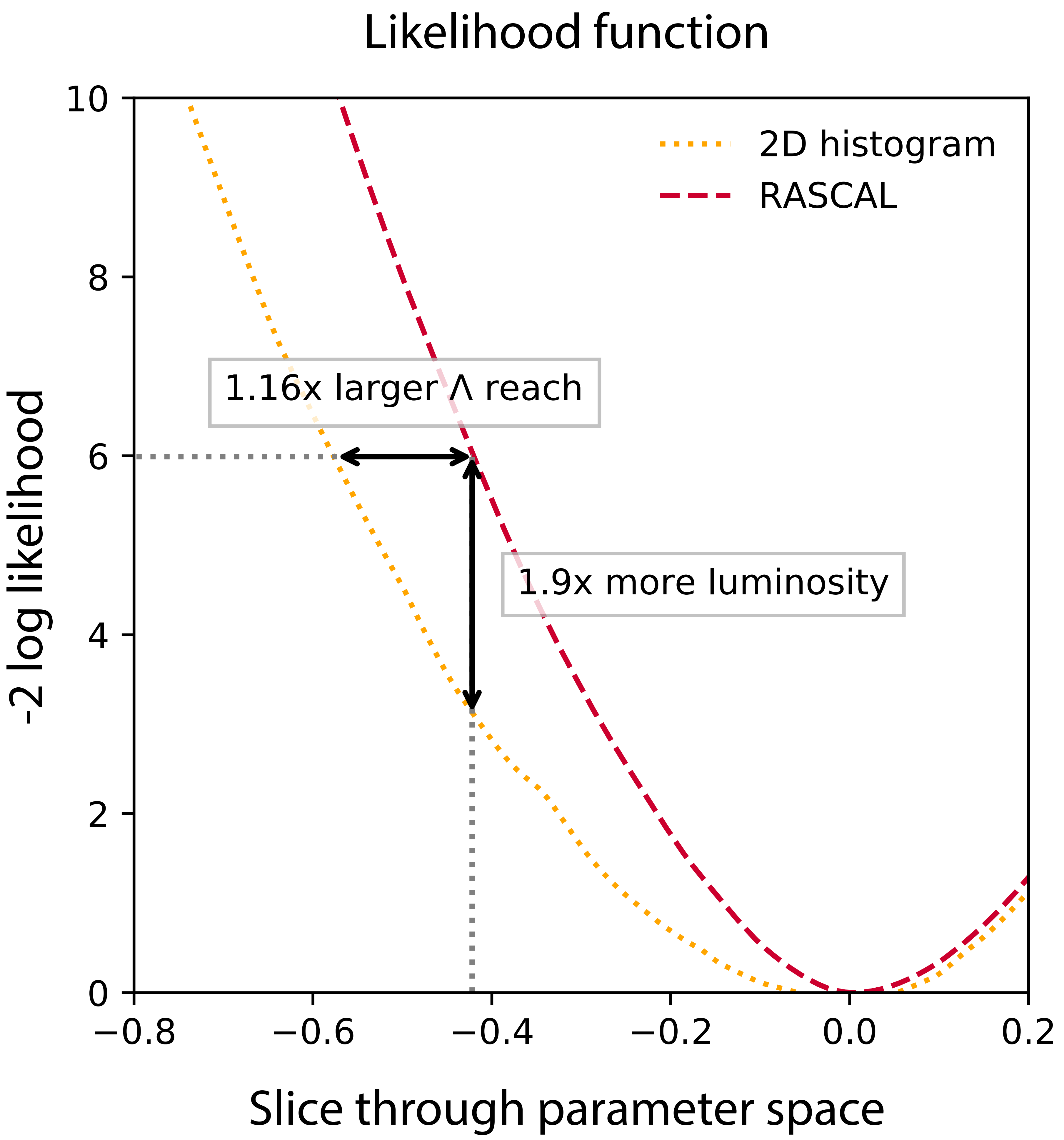


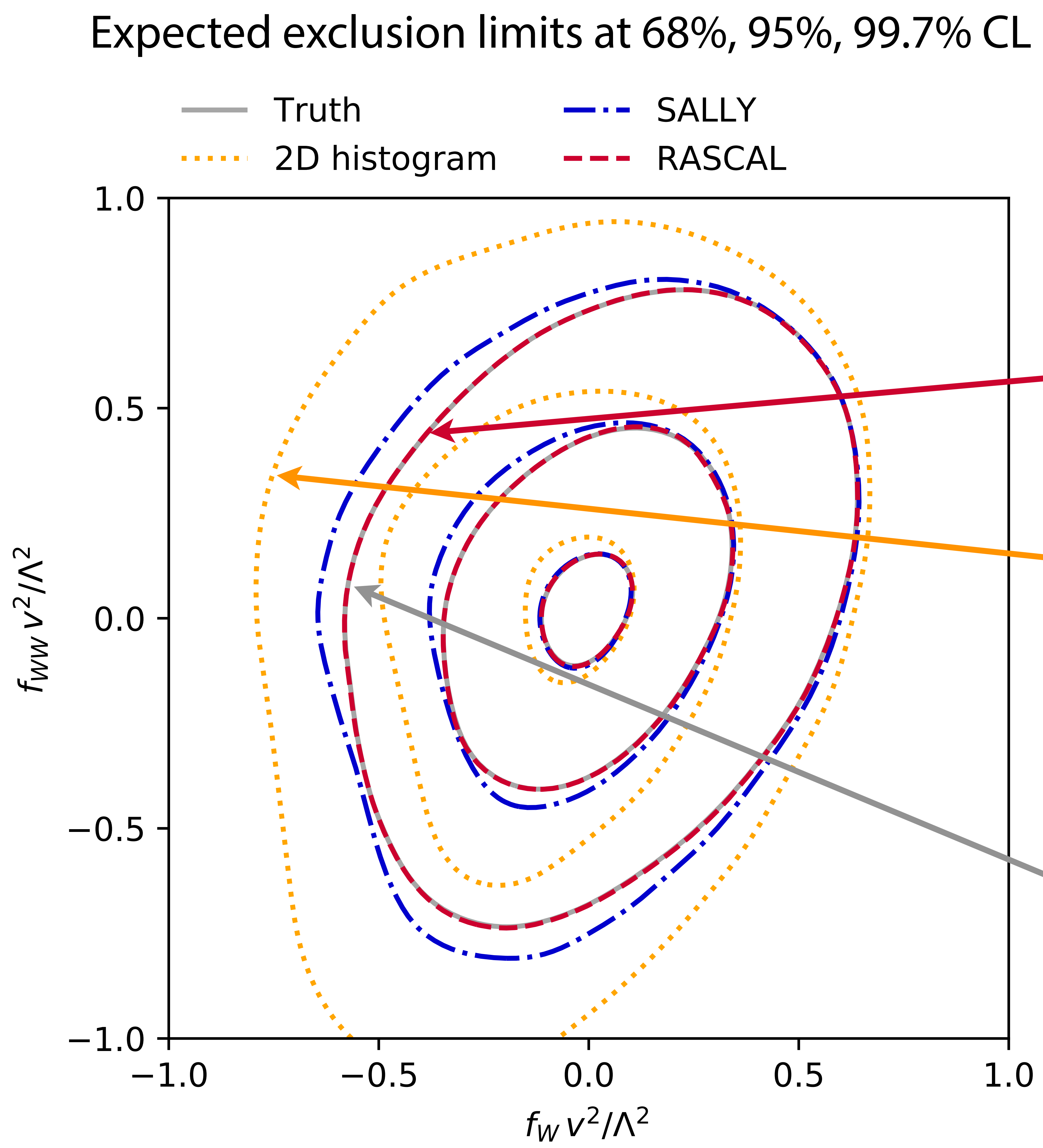
Expected exclusion limits at 68%, 95%, 99.7% CL



Results are based on 36 observed events, assuming SM

Better sensitivity to new physics





RASCAL enables stronger limits than 2D histogram

Limits from RASCAL indistinguishable from true likelihood

(usually we don't have that)

Results are based on 36 observed events, assuming SM

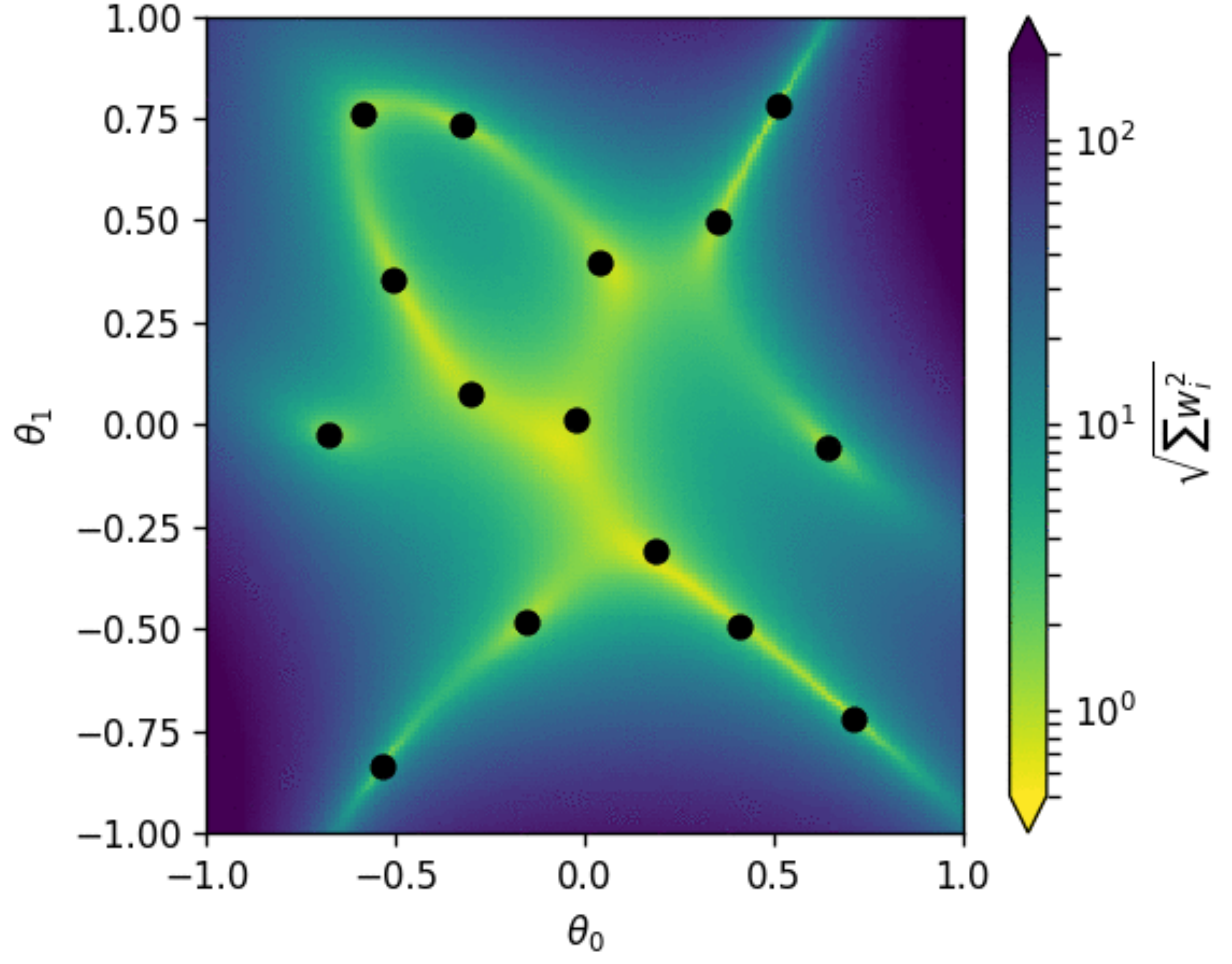
MadMiner

[JB, K. Cranmer, F. Kling in progress]

Can I use any of this?

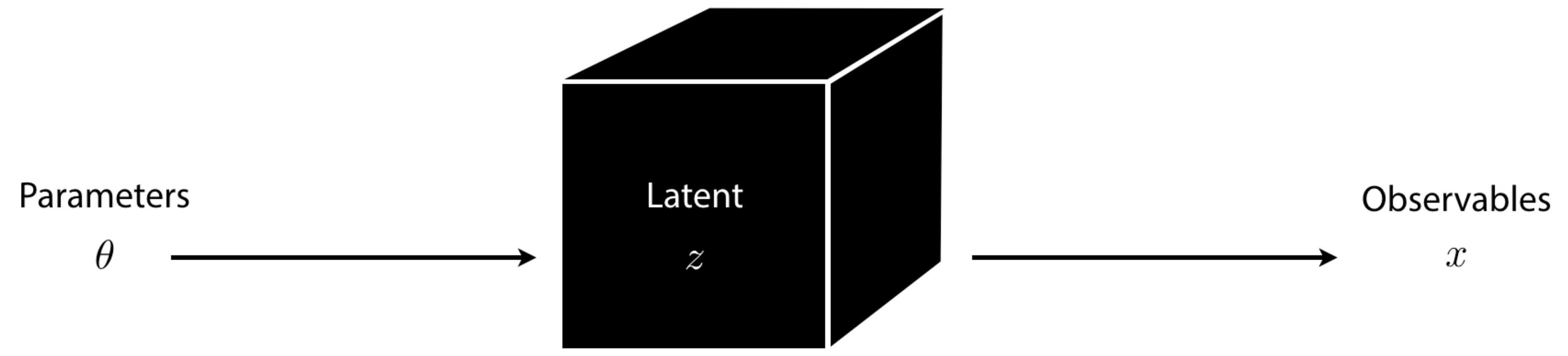
Yes! To make that as painless as possible, we're working on the python package MadMiner:

- "Mining gold" from MadGraph + Pythia + detector simulation
- Morphing: reconstruct full dependence on model parameters from few MC runs
- Likelihood ratio estimation with RASCAL and friends
- Calculate Fisher information (truth or reco level)



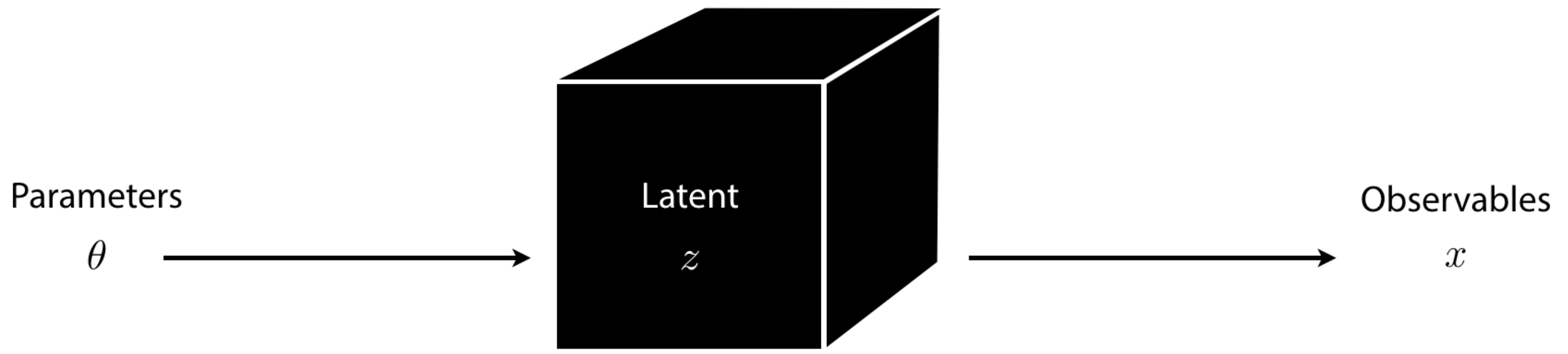
Come visit us soon at github.com/johannbrehmer/madminer!

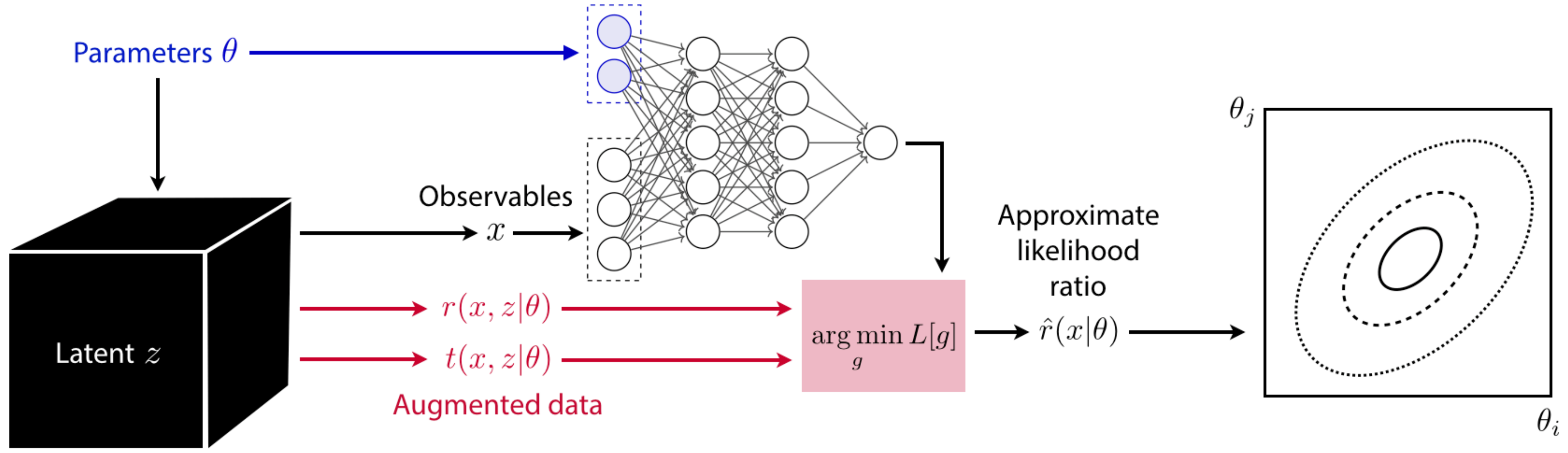
A new approach to simulator-based inference



- Many LHC analyses (and much of modern science) are based on simulations, “likelihood-free”

A new approach to simulator-based inference

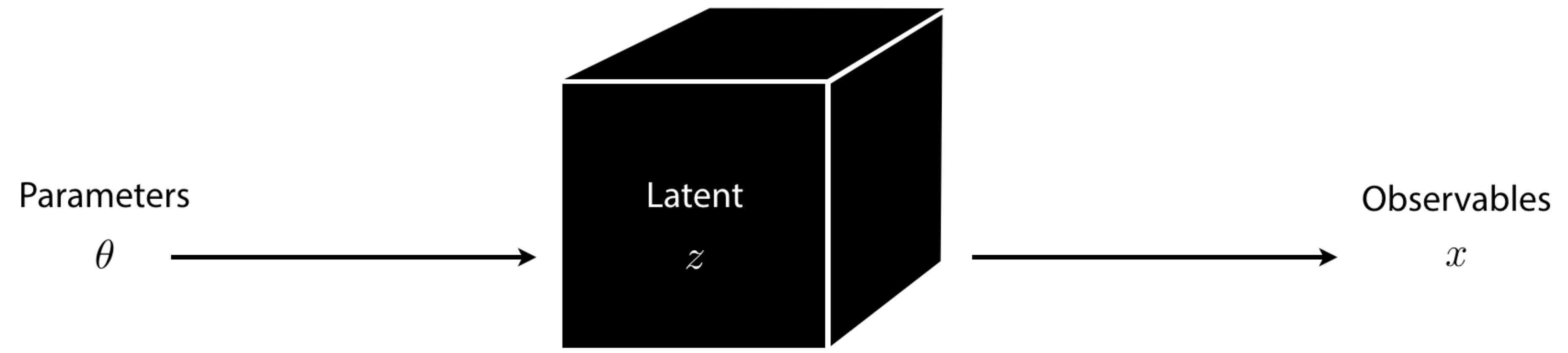


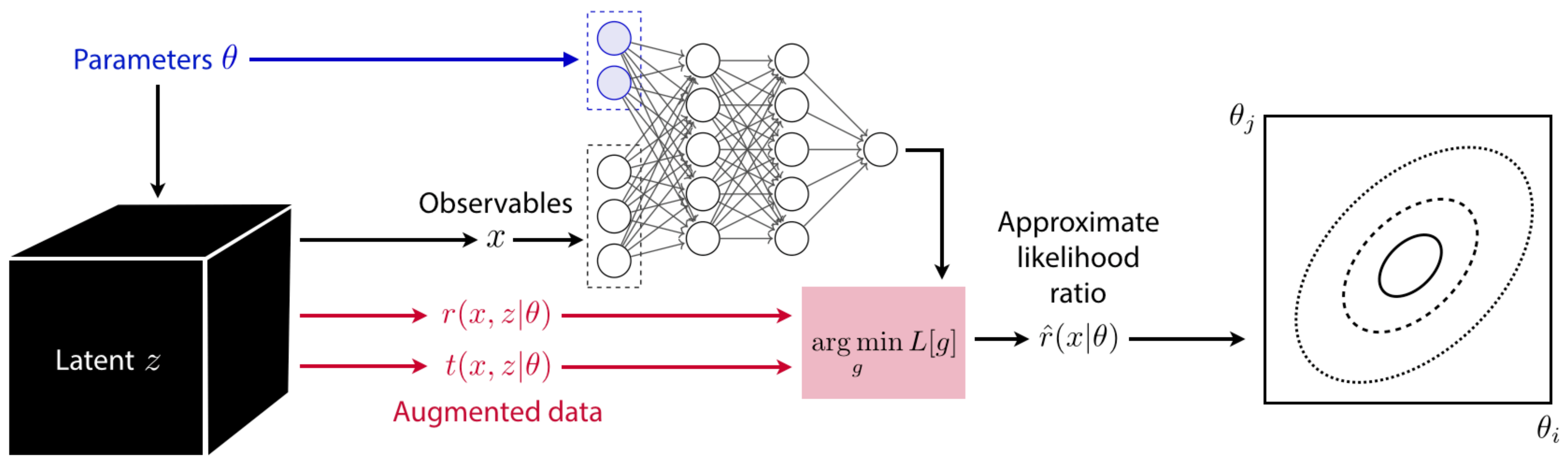


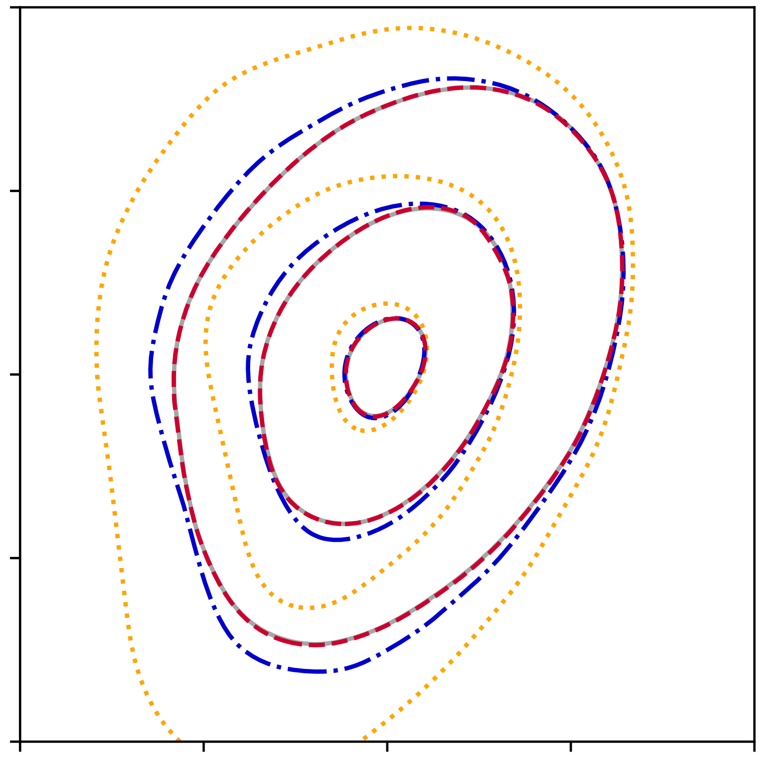
- Many LHC analyses (and much of modern science) are based on simulations, “likelihood-free”

- New multivariate inference techniques: Leverage more information from simulator + power of machine learning

A new approach to simulator-based inference







- Many LHC analyses (and much of modern science) are based on simulations, "likelihood-free"

- New multivariate inference techniques: Leverage more information from simulator + power of machine learning

- First application to LHC physics: Stronger EFT constraints with less simulations

References

Kyle Cranmer

Gilles Louppe

Juan Pavez

Markus Stoye

Felix Kling

Tilman Plehn

Tim Tait

KC, GL, JP:	Approximating Likelihood Ratios with Calibrated Discriminative Classifiers	[1506.02169]
JB, KC, FK, TP:	Better Higgs Measurements Through Information Geometry	[1612.05261]
JB, FK, TP, TT:	Better Higgs-CP Measurements Through Information Geometry	[1712.02350]
JB, KC, GL, JP:	**Constraining Effective Field Theories with Machine Learning**	[1805.00013]
JB, KC, GL, JP:	**A Guide to Constraining Effective Field Theories with Machine Learning**	[1805.00020]
JB, GL, JP, KC:	Mining gold from implicit models to improve likelihood-free inference	[1805.12244]
MS, **JB**, GL, JP, KC:	Likelihood-free inference with an improved cross-entropy estimator	[1808.00973]
JB, KC, FK:	MadMiner	In preparation

Thanks to Kyle and Gilles for inspiring many slides!

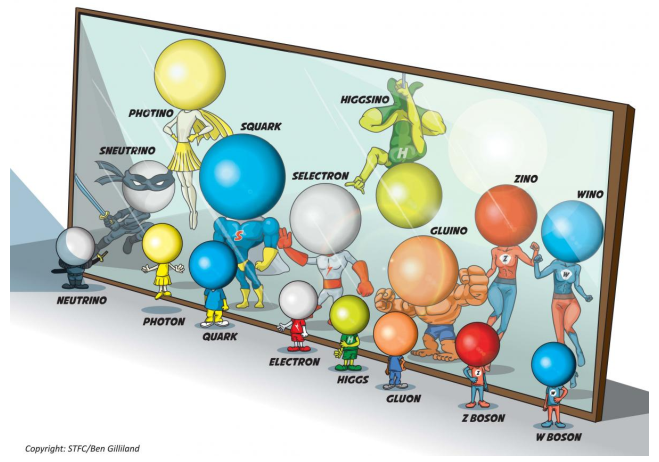
Bonus material

[STFC / Ben Gililand, Sean Carroll, Friedrich Justin Bertuch 1806, symmetry]

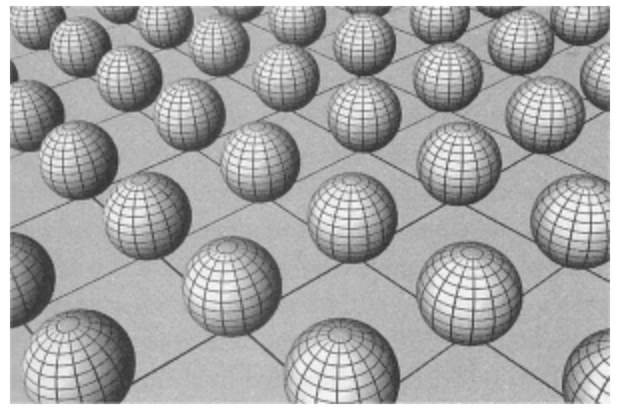
Effective field theory

Energy

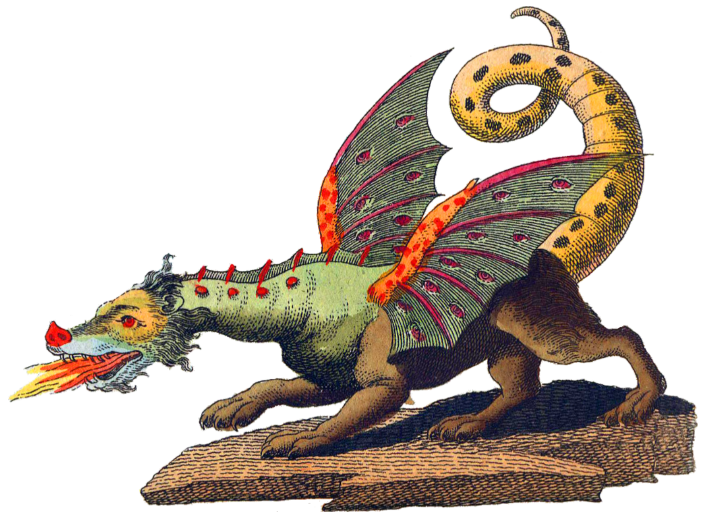
Λ



Supersymmetry?



Extra dimensions?



Dragons?

Composite Higgs?

…

Matching

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

Effective field theory

Measurements

$E \ll \Lambda$

LHC Higgs

Other LHC measurements

LEP

Flavour

…

Variational calculus

$$L[\hat{g}(x)] = \int \mathrm{d}x\, \mathrm{d}z\ p(x,z|\theta)\ |g(x,z) - \hat{g}(x)|^2$$

$$= \int \mathrm{d}x\ \underbrace{\left[\hat{g}^2(x) \int \mathrm{d}z\ p(x,z|\theta) - 2\hat{g}(x) \int \mathrm{d}z\ p(x,z|\theta)\, g(x,z) + \int \mathrm{d}z\ p(x,z|\theta)\, g^2(x,z)\right]}_{F(x)}$$

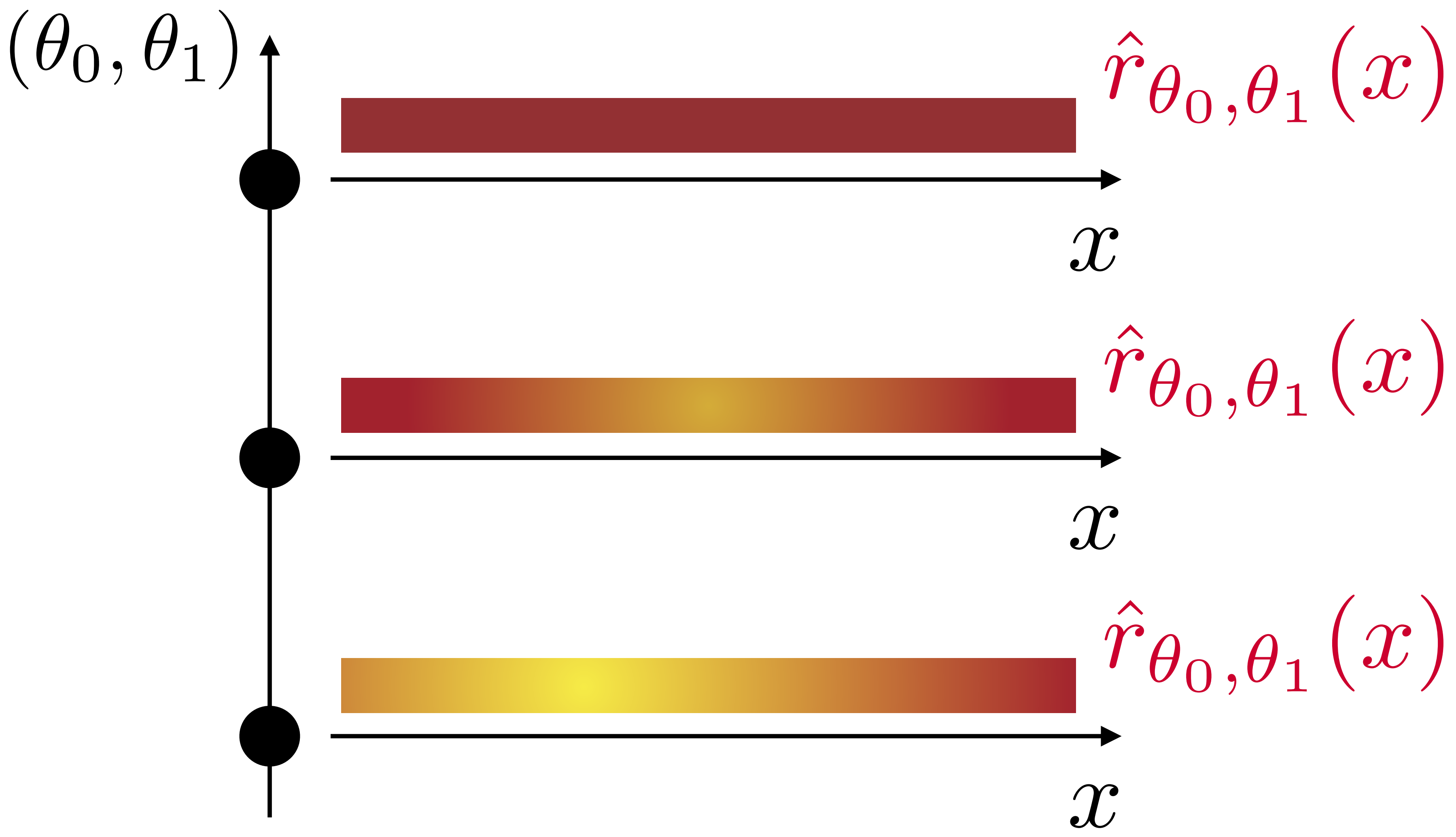
$$0 = \left.\frac{\delta F}{\delta \hat{g}}\right|_{g^*} = 2\hat{g} \underbrace{\int \mathrm{d}z\ p(x,z|\theta)}_{=p(x|\theta)} - 2 \int \mathrm{d}z\ p(x,z|\theta)\, g(x,z)$$

$$g^*(x) = \frac{1}{p(x|\theta)} \int \mathrm{d}z\ p(x,z|\theta)\, g(x,z)$$

Two types of likelihood ratio estimators

A) Point by point:

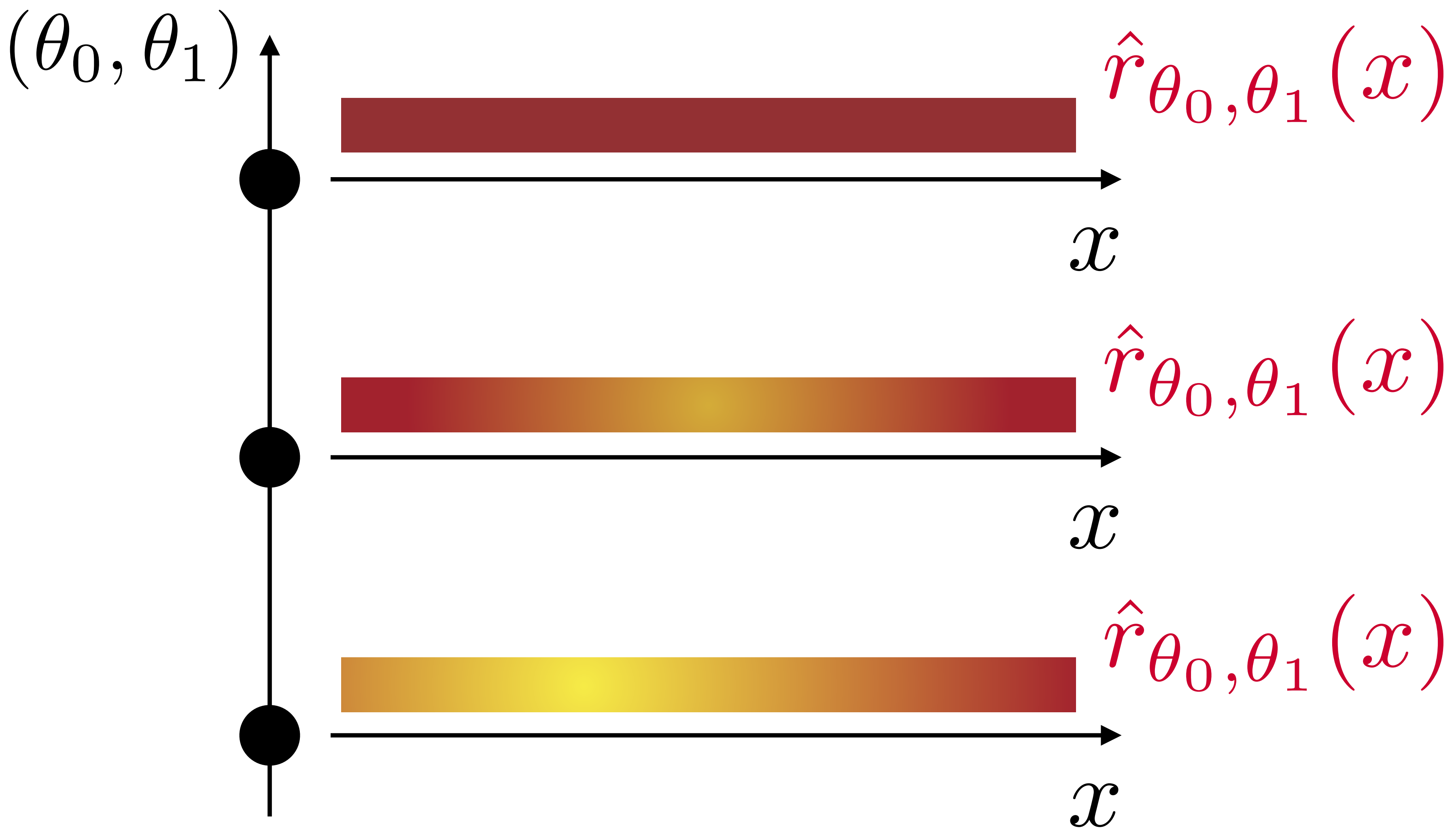
- first, define grid of parameter points $\{(\theta_0, \theta_1)\}$
- for each combination (θ_0, θ_1),
  create separate estimator $\hat{r}_{\theta_0,\theta_1}(x)$
- final results can be interpolated between grid points



Two types of likelihood ratio estimators

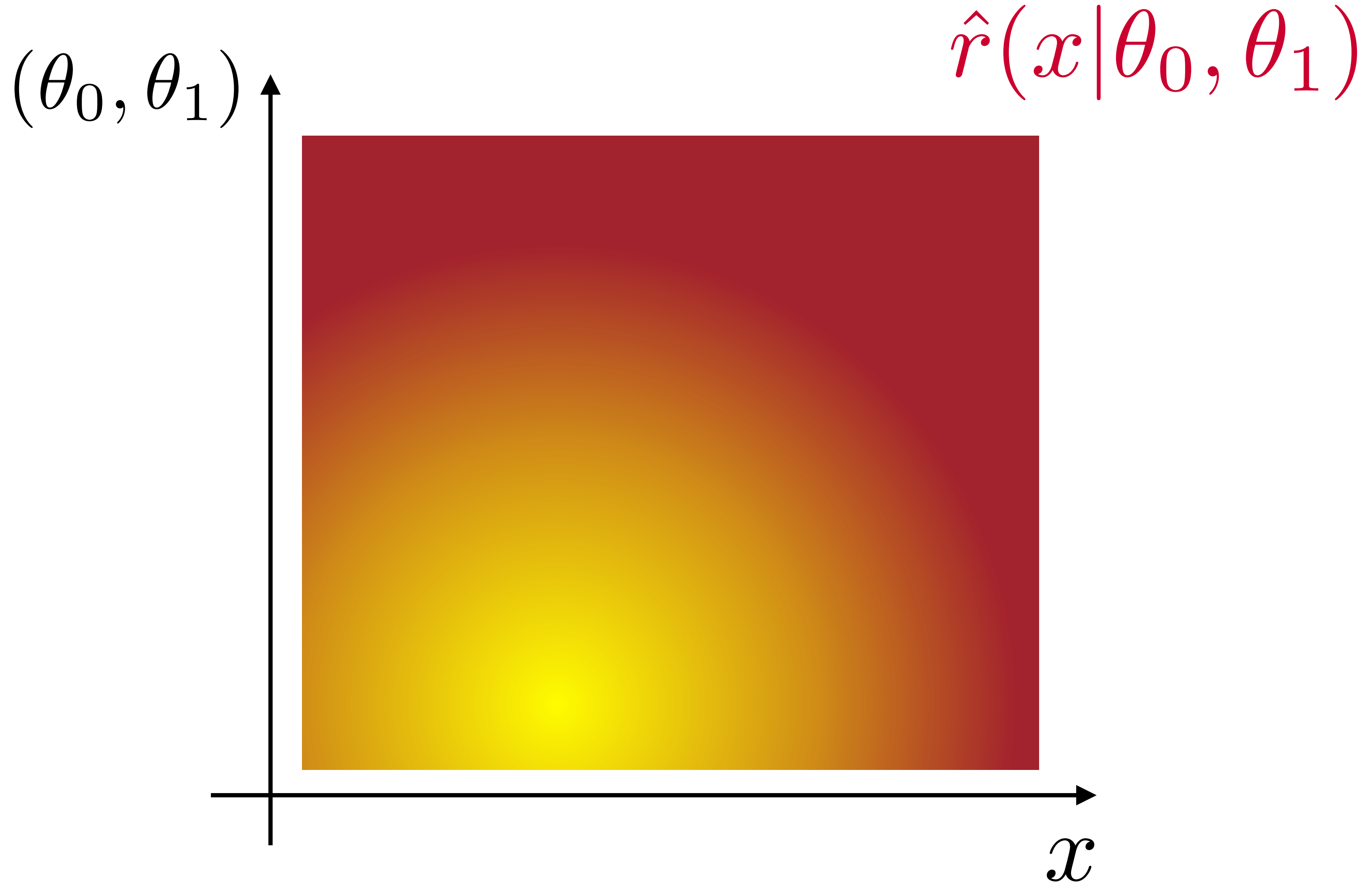
A) Point by point:

- first, define grid of parameter points $\{(\theta_0, \theta_1)\}$
- for each combination (θ_0, θ_1), create separate estimator $\hat{r}_{\theta_0,\theta_1}(x)$
- final results can be interpolated between grid points

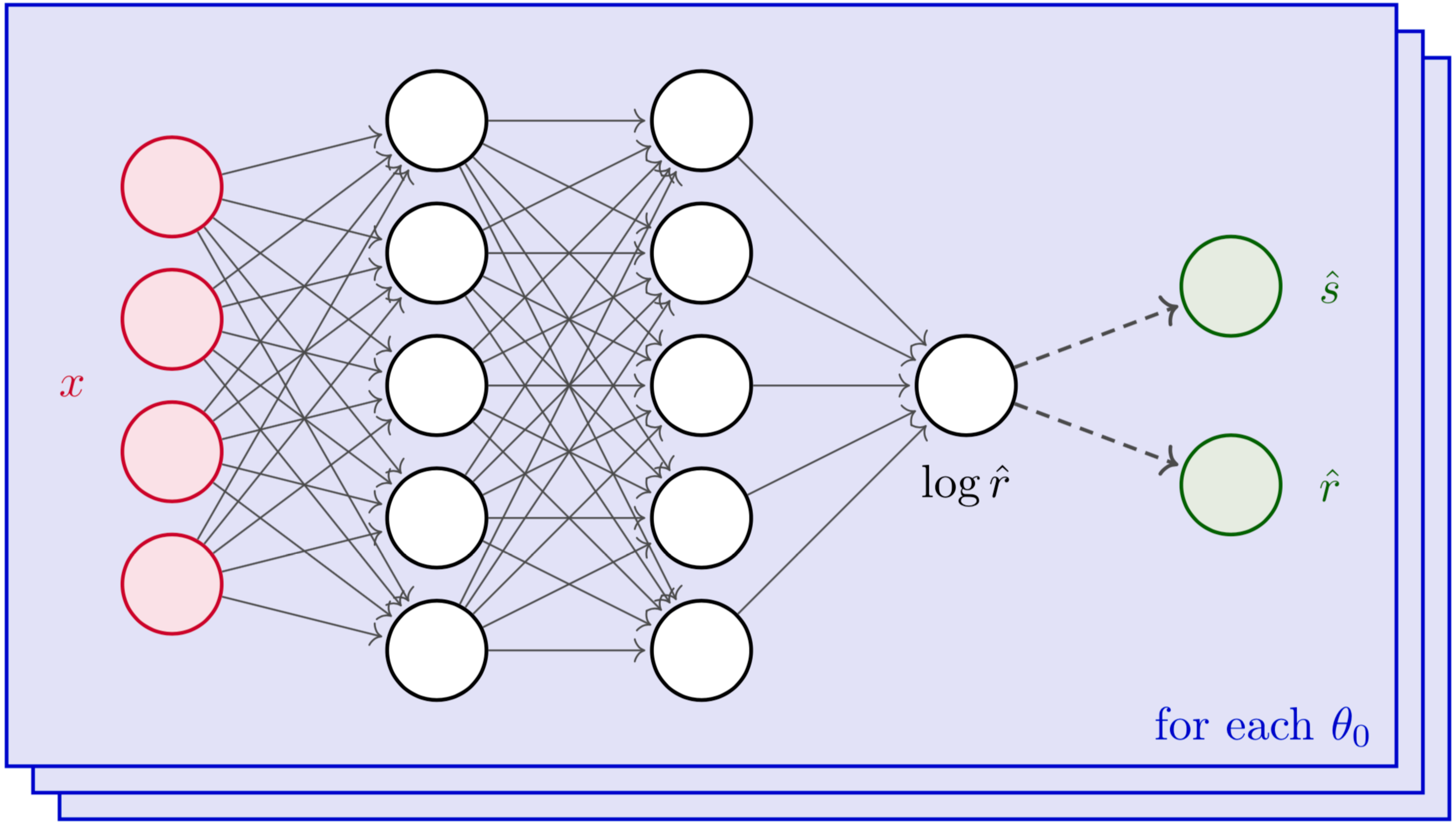


B) Parameterized: [P. Baldi et al. 1506.02169]

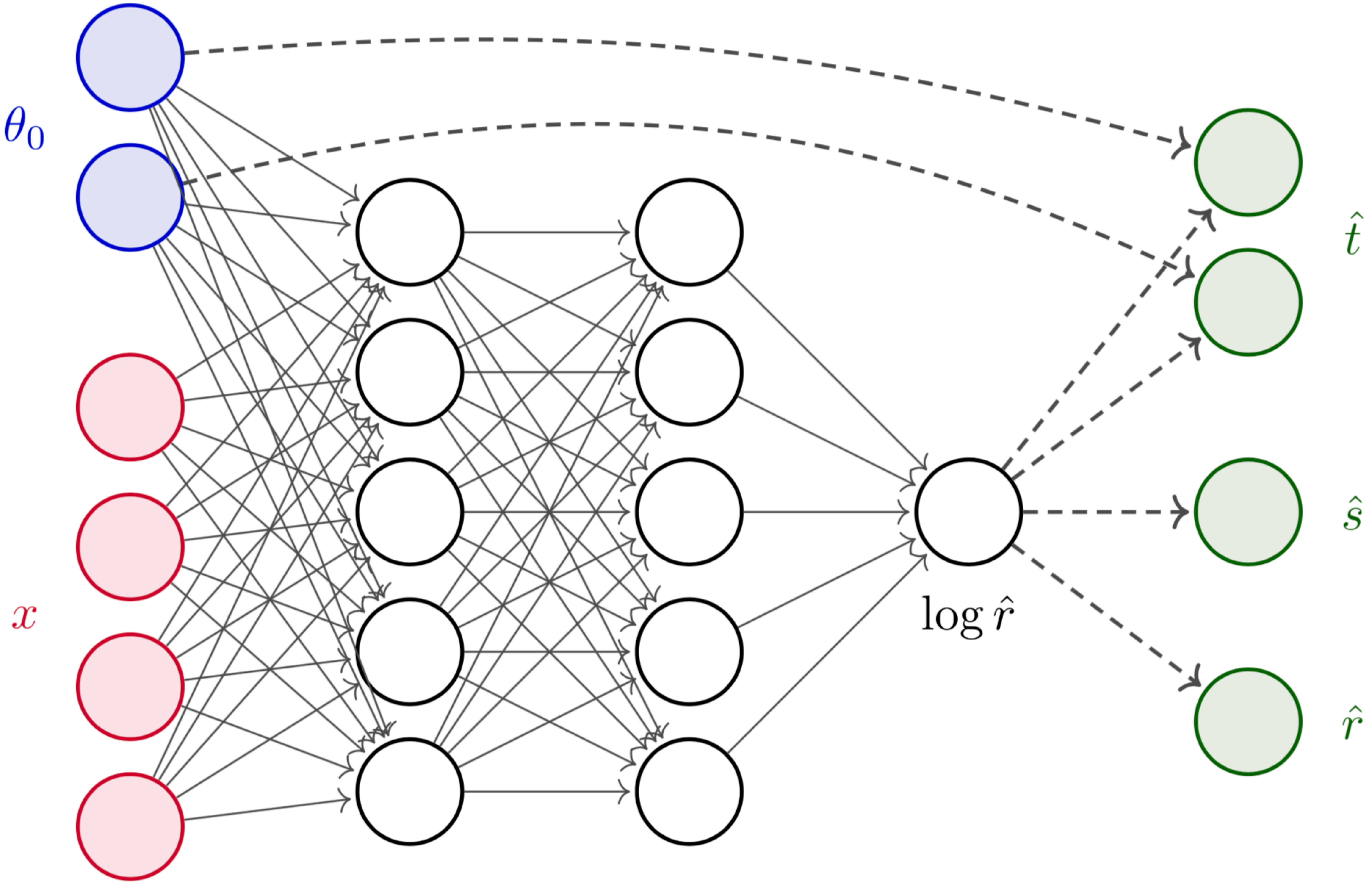
- create one estimator $\hat{r}(x|\theta_0, \theta_1)$ that is a function of θ_0 and θ_1
- no further interpolation necessary
- "borrows information" from close points



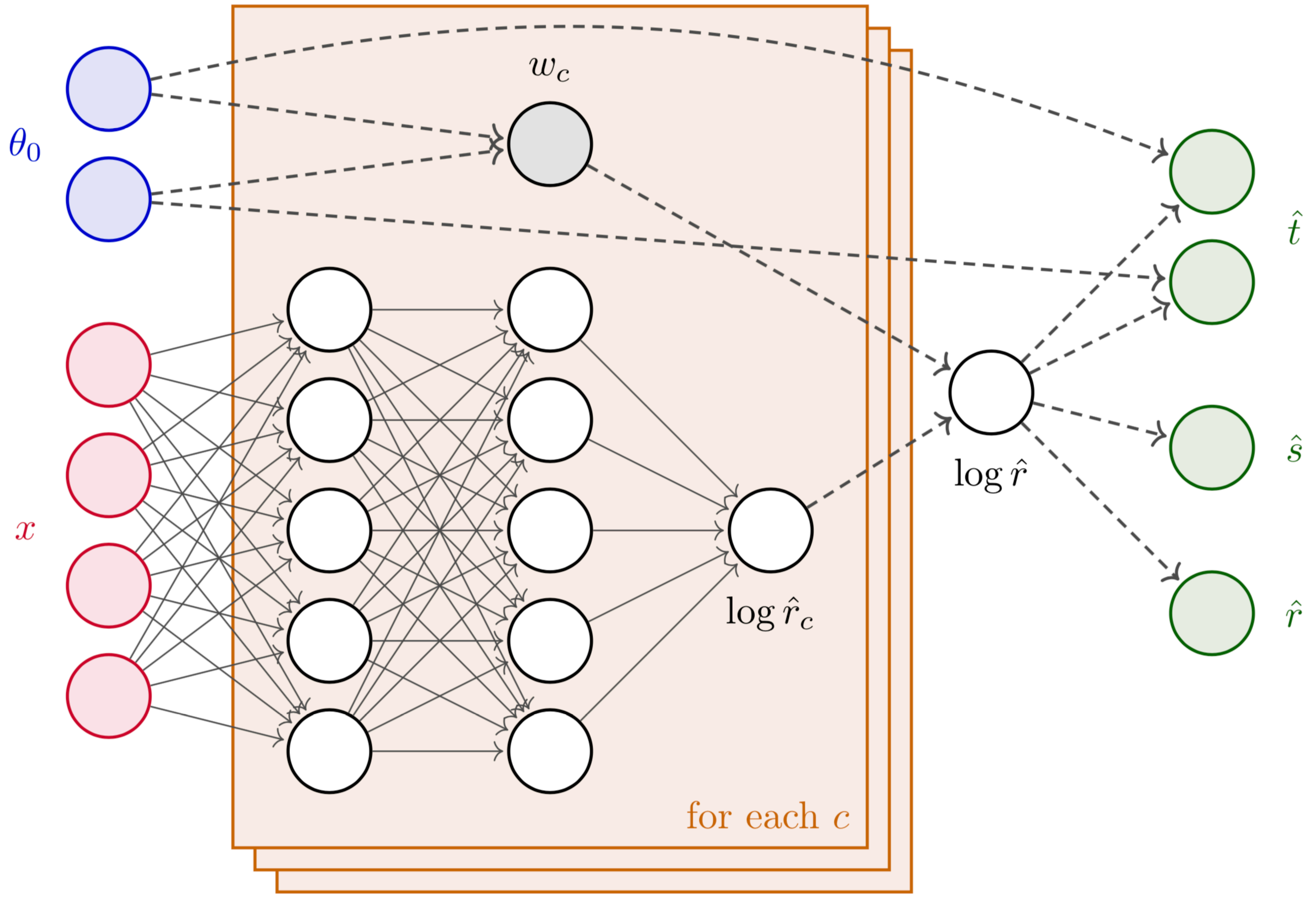
Point by point



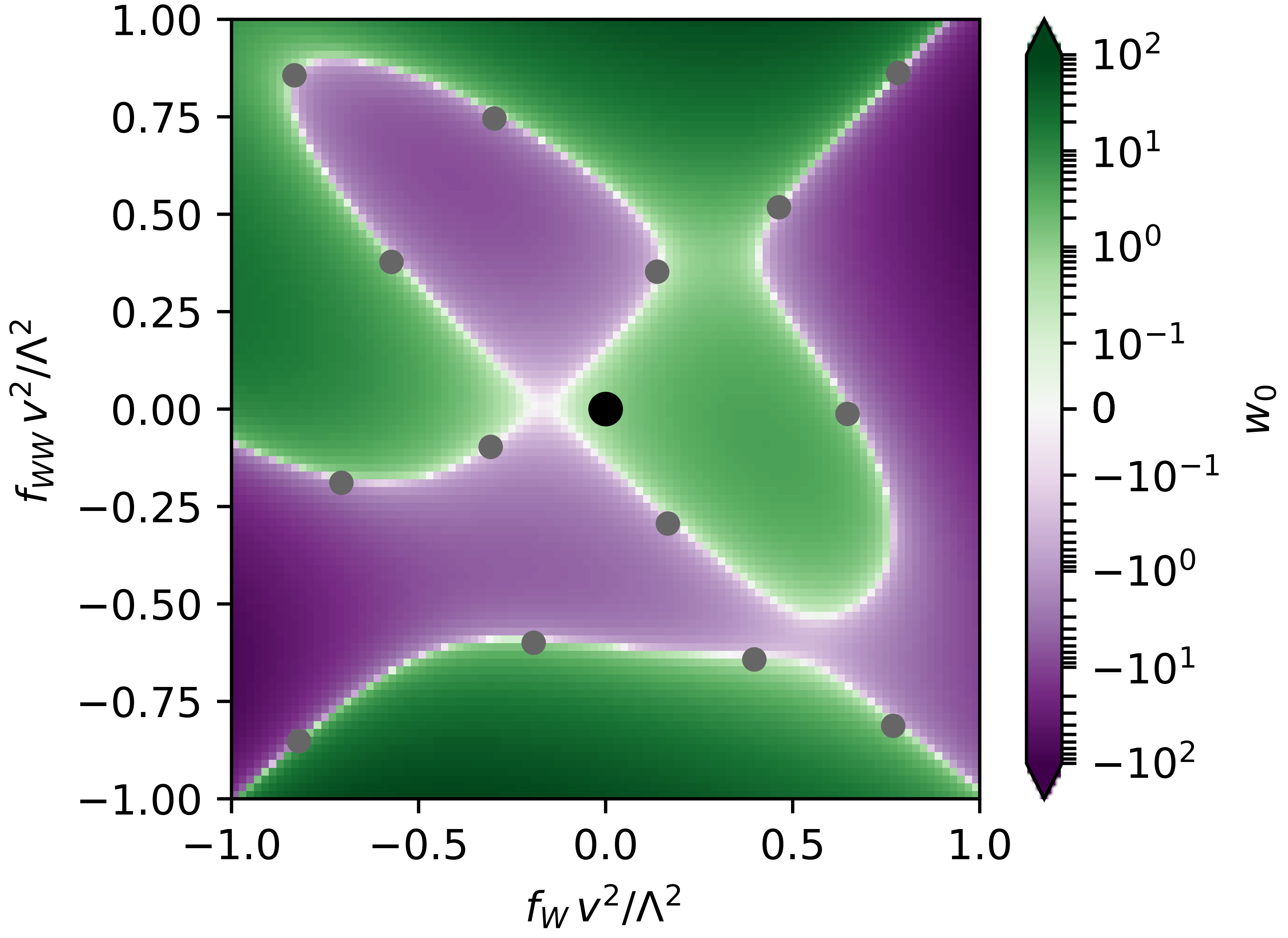
(Agnostic) parameterized estimators



Morphing-aware parameterized estimators



$$\hat{r}(x|\theta_0,\theta_1) = \sum_c w_c(\theta_0)\,\hat{r}_c(x)$$



Detector effects