Constraining effective field theories with machine learning

Johann Brehmer
New York University

HEFT 2019, Louvain-la-Neuve


Kyle Cranmer


Gilles Louppe


Juan Pavez


Felix Kling


Irina Espejo

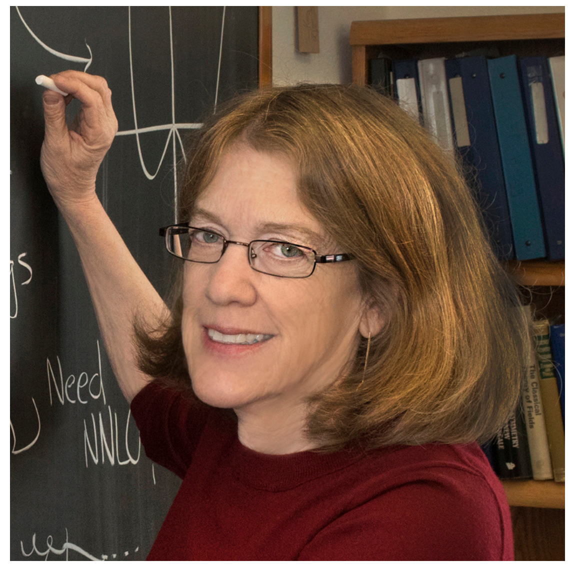

Zubair Bhatti


Markus Stoye


Sid Mishra-Sharma


Joeri Hermans

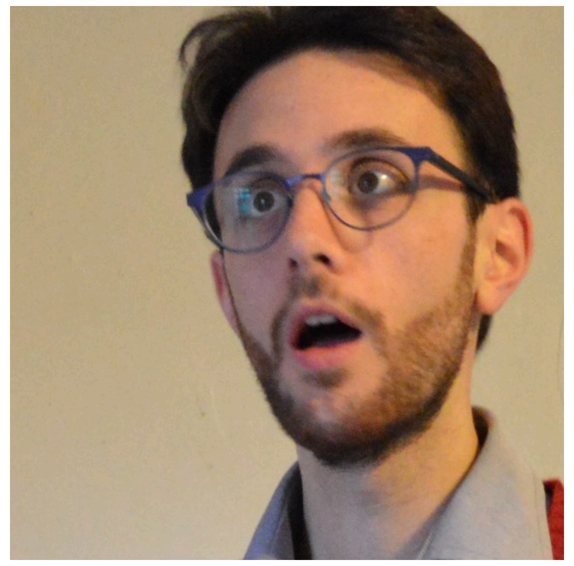

Tilman Plehn


Sally Dawson


Sam Homiller


Josh Ruderman


Duccio Pappadopulo


Marco Farina

Thanks to Kyle, Gilles, Felix, Irina, and Sam for material and inspiration for slides!

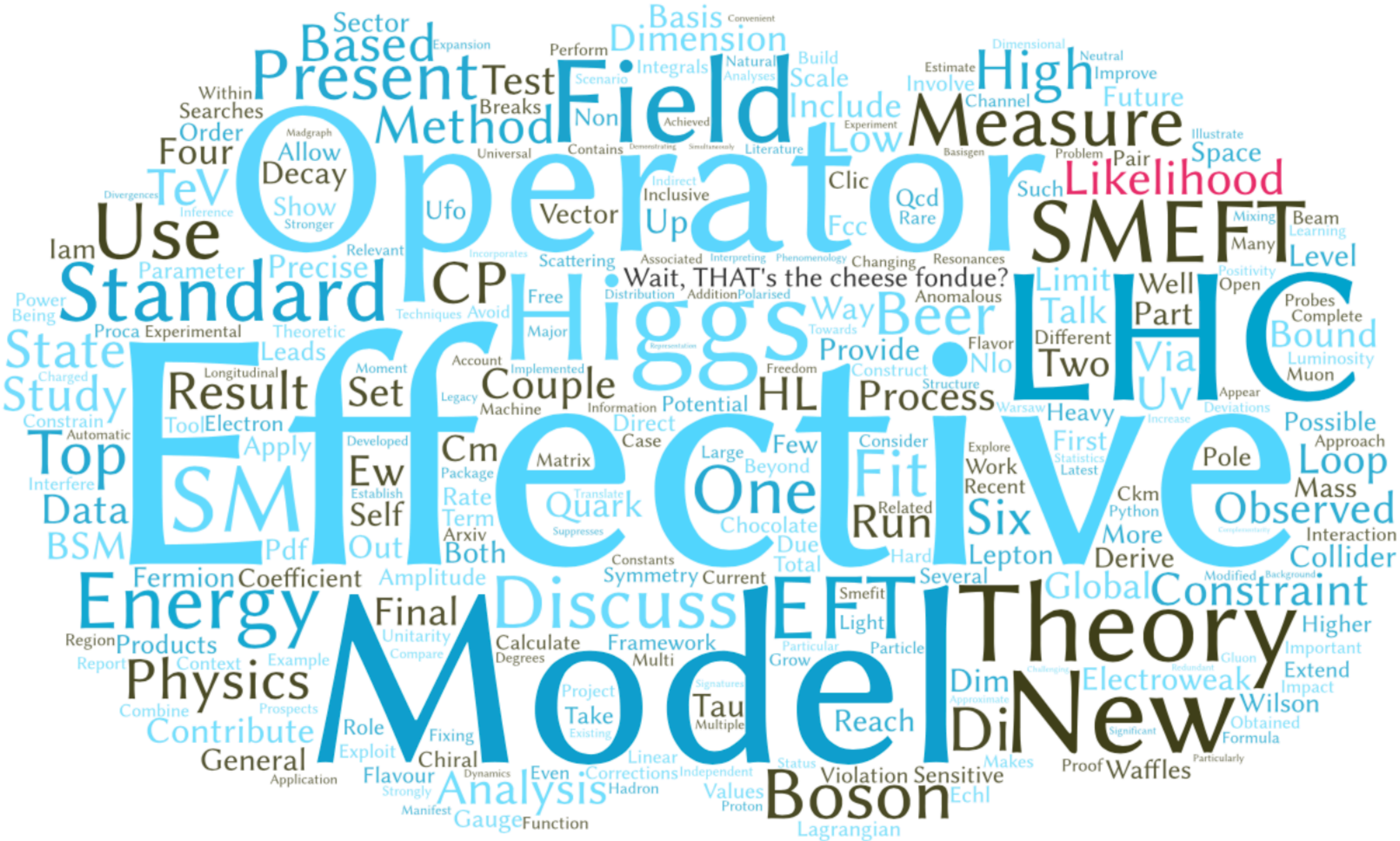
NYU

MSDSE
Moore - Sloan
Data Science Environments

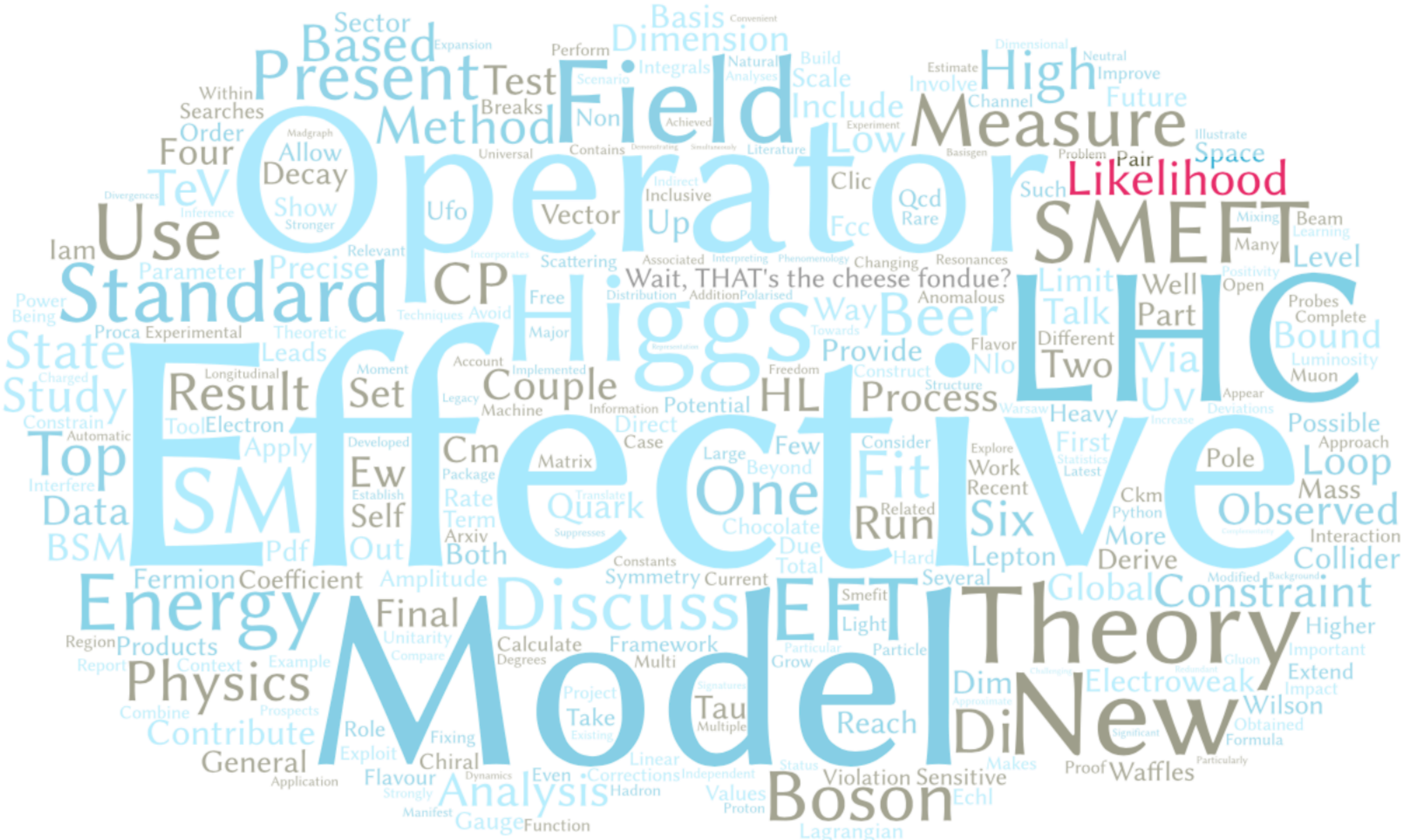
dianahep

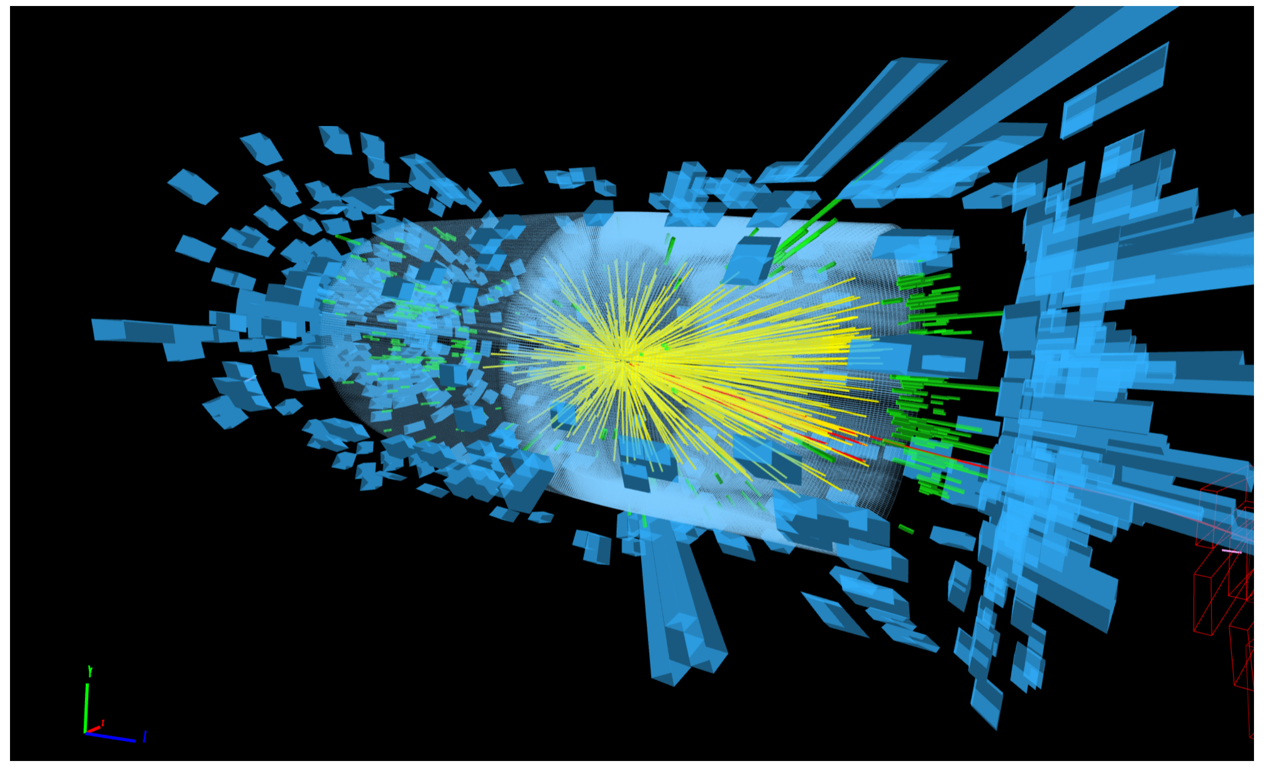
iris hep
Institute for Research and Innovation
in Software for High Energy Physics

The SCAILFIN Project
scailfin.github.io

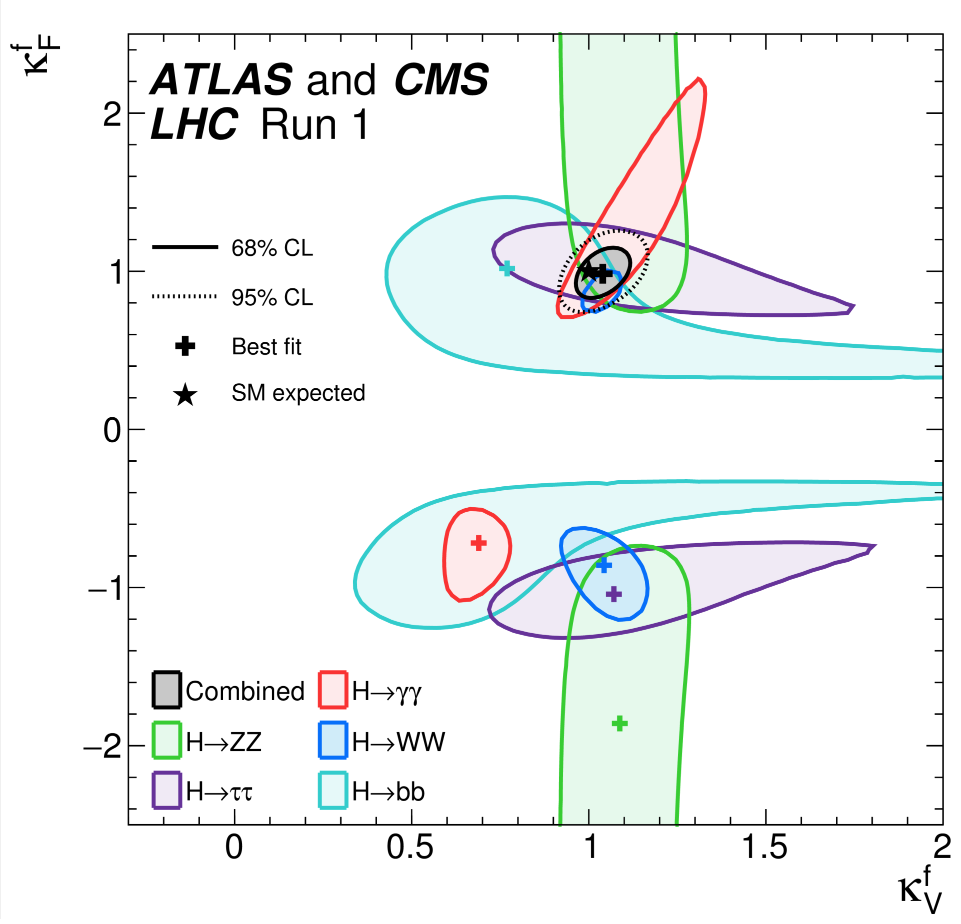


Effective
Model
Operator
Higgs
LHC
SMEFT
Theory
New
Boson
Field
Present
Measure
High
Standard
Use
Energy
Discuss
EFT
SM
Physics
Likelihood
Wait, THAT's the cheese fondue?
Beer
Chocolate
Waffles
Based
Sector
Basis
Dimension
Method
Test
Include
Low
Scale
Integrals
Natural
Analyses
Build
Involve
Channel
Future
Improve
Neutral
Estimate
Dimensional
Expansion
Perform
Within
Searches
Order
Four
Allow
Decay
TeV
Show
Stronger
Inference
Divergences
Madgraph
Universal
Contains
Breaks
Non
Achieved
Experiment
Literature
Illustrate
Space
Pair
Problem
Basisgen
Such
Clic
Qcd
Rare
Fcc
Up
Inclusive
Indirect
Vector
Ufo
Iam
CP
Relevant
Incorporates
Scattering
Associated
Interpreting
Phenomenology
Changing
Resonances
Mixing
Beam
Learning
Many
Level
Positivity
Open
Limit
Well
Parameter
Precise
Power
Being
Free
Distribution
Addition
Polarised
Anomalous
Way
Talk
Part
Probes
Complete
Bound
Techniques
Avoid
Theoretic
Major
Towards
Provide
Flavor
Different
Two
Via
Luminosity
Muon
Proca
Experimental
State
Leads
Study
Charged
Longitudinal
Moment
Account
Implemented
Freedom
Construct
Nlo
Structure
Uv
Appear
Result
Set
Couple
HL
Process
Legacy
Machine
Information
Potential
Warsaw
Heavy
Increase
Deviations
Constrain
Automatic
Tool
Electron
Direct
Possible
Approach
Top
Apply
Developed
Cm
Case
Matrix
Large
Few
Consider
Explore
First
Statistics
Latest
Pole
Loop
Interfere
Ew
Package
Establish
Beyond
One
Fit
Work
Recent
Mass
Ckm
Data
Rate
Term
Translate
Quark
Related
Run
Six
Python
Observed
Self
BSM
Pdf
Out
Arxiv
Suppresses
Both
Due
Total
Hard
Lepton
More
Interaction
Derive
Collider
Constants
Fermion
Coefficient
Amplitude
Symmetry
Current
Several
Modified
Background
Global
Constraint
Smefit
Light
Higher
Final
Unitarity
Calculate
Framework
Particular
Particle
Important
Region
Products
Compare
Degrees
Multi
Grow
Gluon
Extend
Report
Context
Example
Electroweak
Impact
Dim
Project
Signatures
Approximate
Wilson
Combine
Prospects
Take
Existing
Tau
Multiple
Reach
Di
Obtained
Significant
Contribute
Role
Fixing
Formula
General
Exploit
Chiral
Linear
Status
Makes
Particularly
Proof
Application
Flavour
Dynamics
Even
Corrections
Independent
Violation
Sensitive
Strongly
Analysis
Hadron
Values
Echl
Manifest
Gauge
Function
Proton
Lagrangian

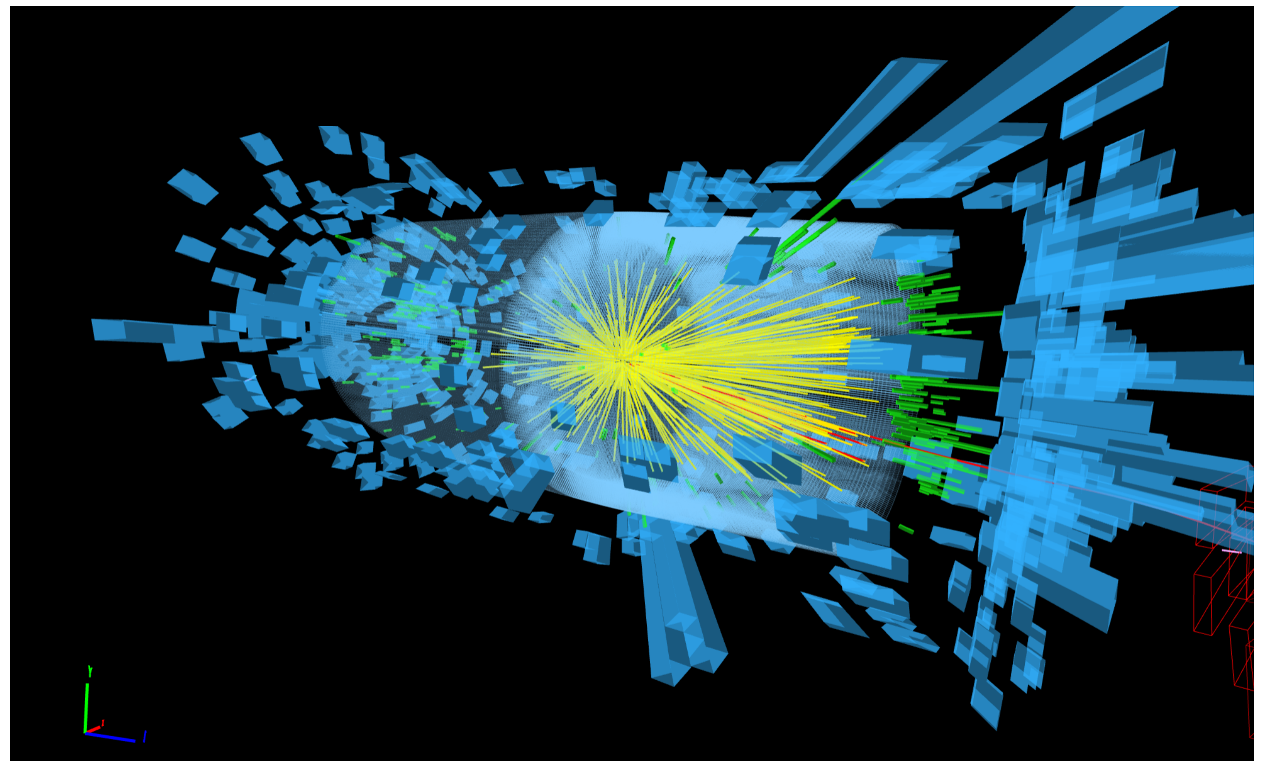

Basis
Convenient
Sector
Based
Expansion
Dimension
Perform
Present
Test
Field
Integrals
Natural
Build
Scale
Estimate
High
Dimensional
Neutral
Improve
Within
Searches
Breaks
Scenario
Analyses
Involve
Channel
Future
Order
Method
Non
Achieved
Include
Experiment
Measure
Illustrate
Madgraph
Allow
Four
Universal
Contains
Literature
Low
Basisgen
Problem
Pair
Space
Decay
Likelihood
TeV
Divergences
Operator
Indirect
Inclusive
Clic
Such
Qcd
Rare
Show
Inference
Stronger
Ufo
Vector
Up
Fcc
SMEFT
Mixing
Beam
Learning
Use
Iam
Relevant
Incorporates
Scattering
Associated
Interpreting
Phenomenology
Changing
Resonances
Many
Level
Parameter
Precise
CP
Wait, THAT's the cheese fondue?
Limit
Positivity
Well
Open
Standard
Power
Being
Free
Distribution
Addition
Polarised
Anomalous
Higgs
Way
Beer
LHC
Probes
Complete
Techniques
Avoid
Talk
Part
Proca
Experimental
Theoretic
Major
Towards
Different
Bound
State
Leads
Flavor
Provide
Two
Via
Luminosity
Account
Moment
Implemented
Freedom
Construct
Nlo
Longitudinal
Charged
Result
Set
Couple
Muon
Study
Legacy
Machine
Information
Potential
HL
Structure
Process
Appear
Uv
Warsaw
Heavy
Constrain
Deviations
Increase
Tool
Electron
Direct
Possible
Automatic
Developed
Cm
Case
Few
Consider
Explore
First
Approach
Top
Apply
Matrix
Large
Beyond
Statistics
Latest
Pole
Loop
Effective
SM
Ew
Package
One
Fit
Work
Interfere
Establish
Translate
Recent
Ckm
Mass
Rate
Quark
Related
Data
Self
Run
Six
Python
Observed
Term
Chocolate
Suppresses
Arxiv
More
Interaction
BSM
Pdf
Out
Both
Due
Total
Hard
Lepton
Derive
Collider
Constants
Amplitude
Symmetry
Current
Several
Global
Modified
Background
Fermion
Coefficient
Energy
Discuss
EFT
Smefit
Constraint
Final
Light
Theory
Higher
Unitarity
Region
Products
Compare
Calculate
Framework
Particular
Particle
Important
Report
Context
Example
Degrees
Multi
Grow
Gluon
Extend
Physics
Electroweak
Model
Dim
Impact
Project
Signatures
Approximate
Tau
Wilson
Combine
Prospects
Take
Multiple
Reach
Obtained
Contribute
Role
Existing
Di
New
Significant
Formula
Fixing
Exploit
Linear
General
Chiral
Status
Makes
Particularly
Application
Flavour
Dynamics
Even
Corrections
Independent
Violation
Sensitive
Proof
Waffles
Strongly
Analysis
Hadron
Values
Echl
Manifest
Proton
Boson
Gauge
Function
Lagrangian



High-dimensional
event data x



Constraints on
parameters θ

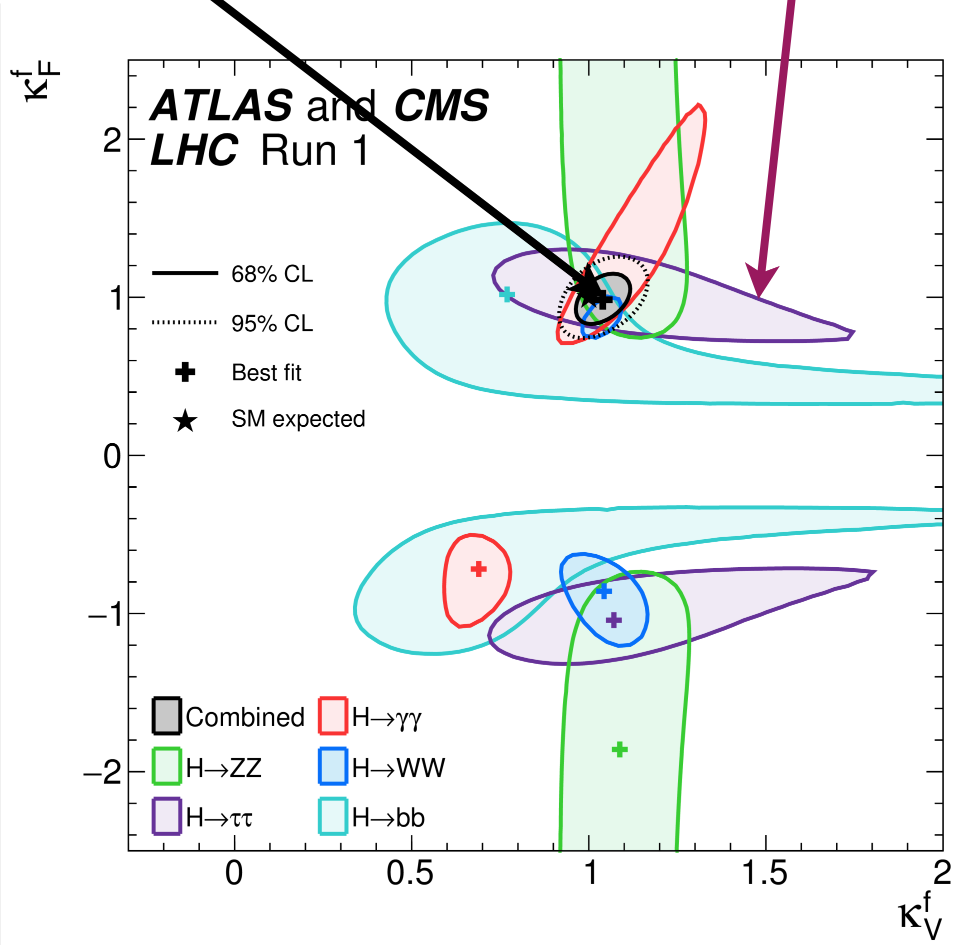


High-dimensional event data x

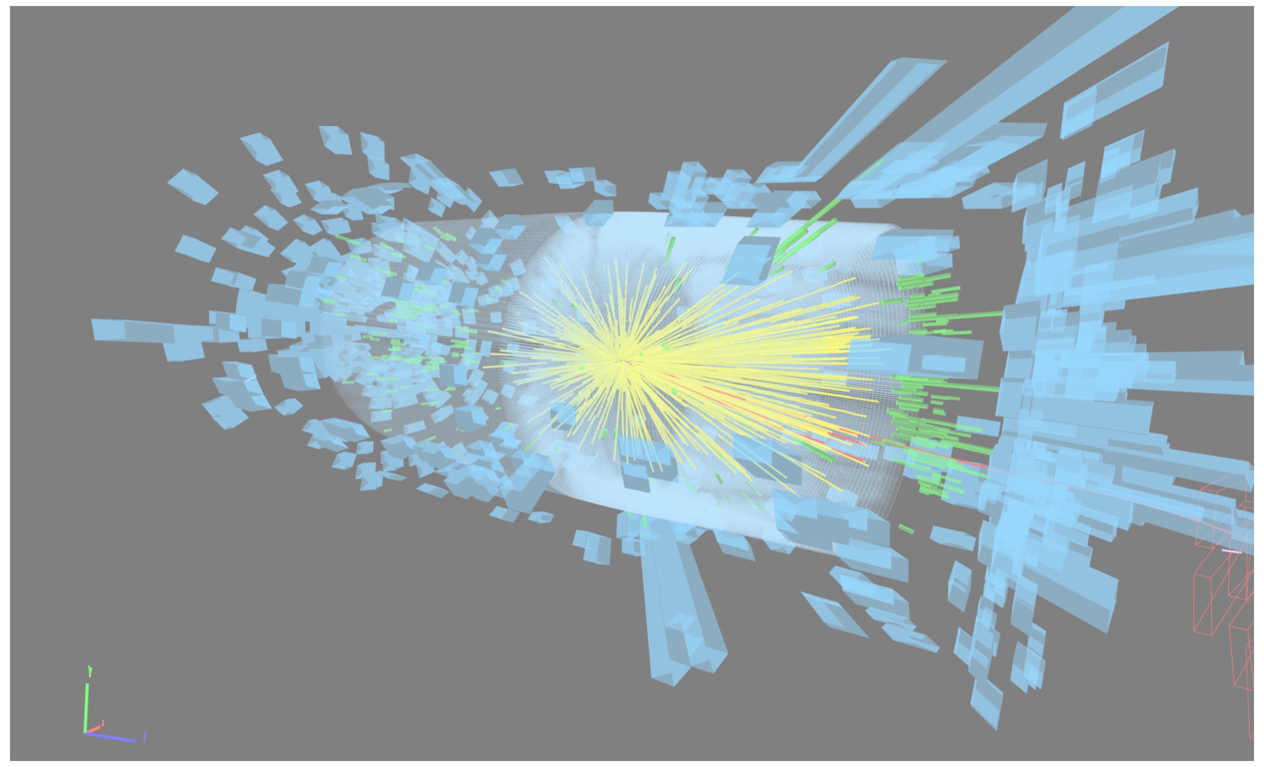
Likelihood function
$p(x|\theta)$

Maximum-likelihood estimator

Confidence limits based on likelihood ratio tests

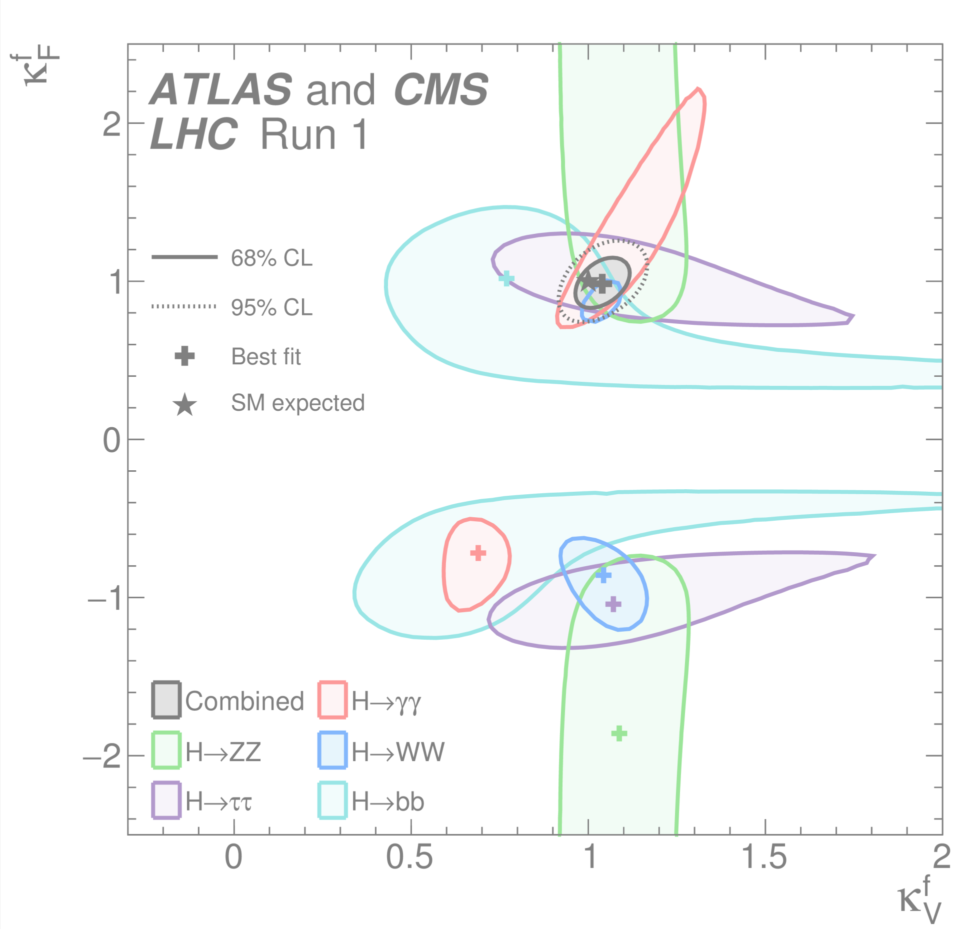


Constraints on parameters θ



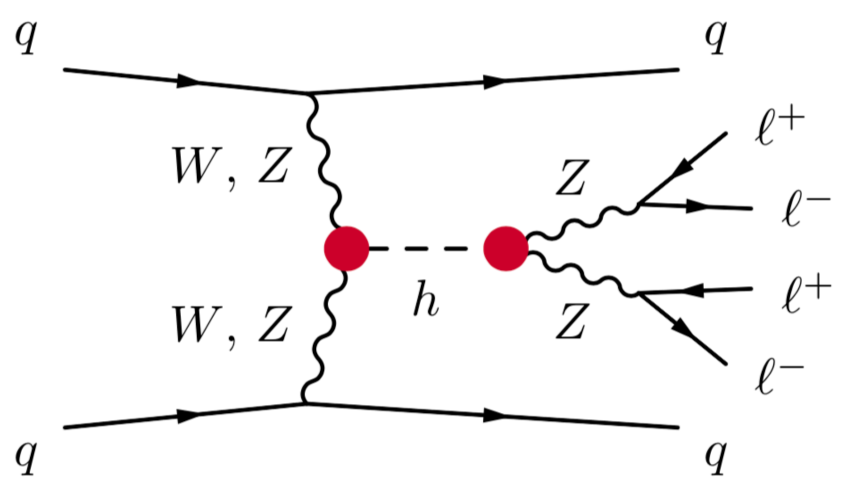
High-dimensional event data x

Likelihood function $p(x|\theta)$

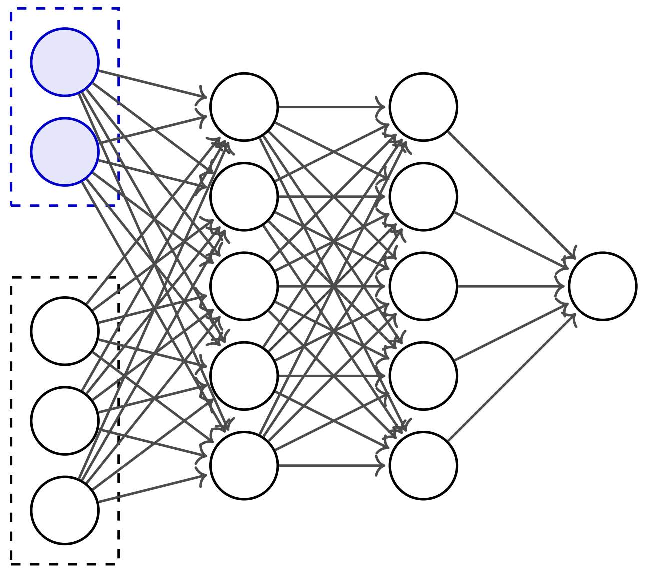


Constraints on parameters θ

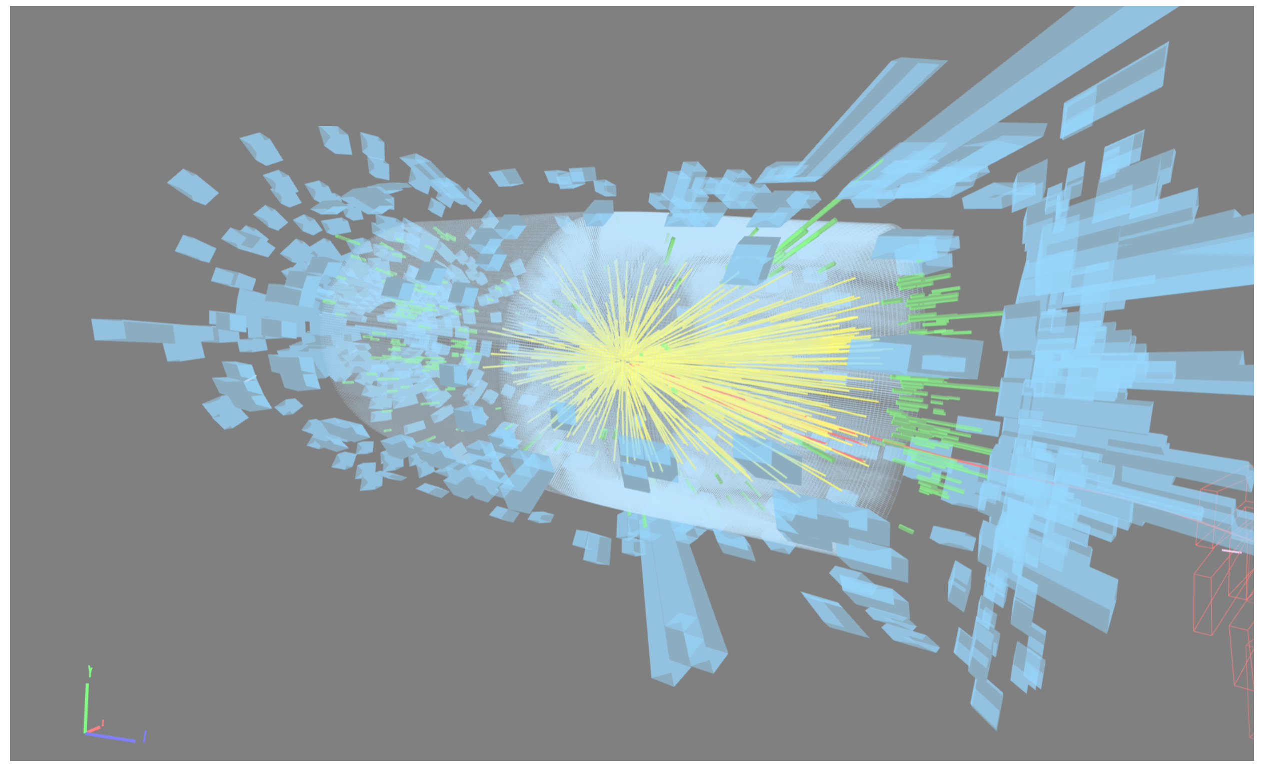
Surprisingly, when we want to use high-dimensional data and have to deal with the detector response, we do not have a good way to calculate the likelihood.



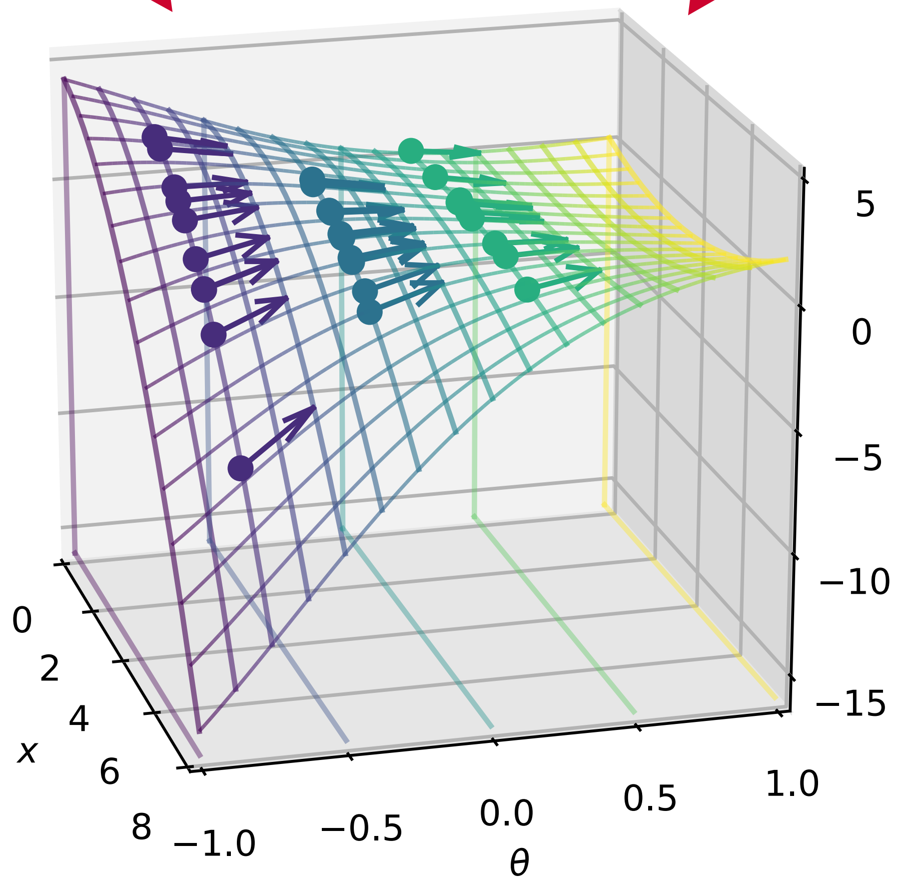
Physics insight:
matrix element information



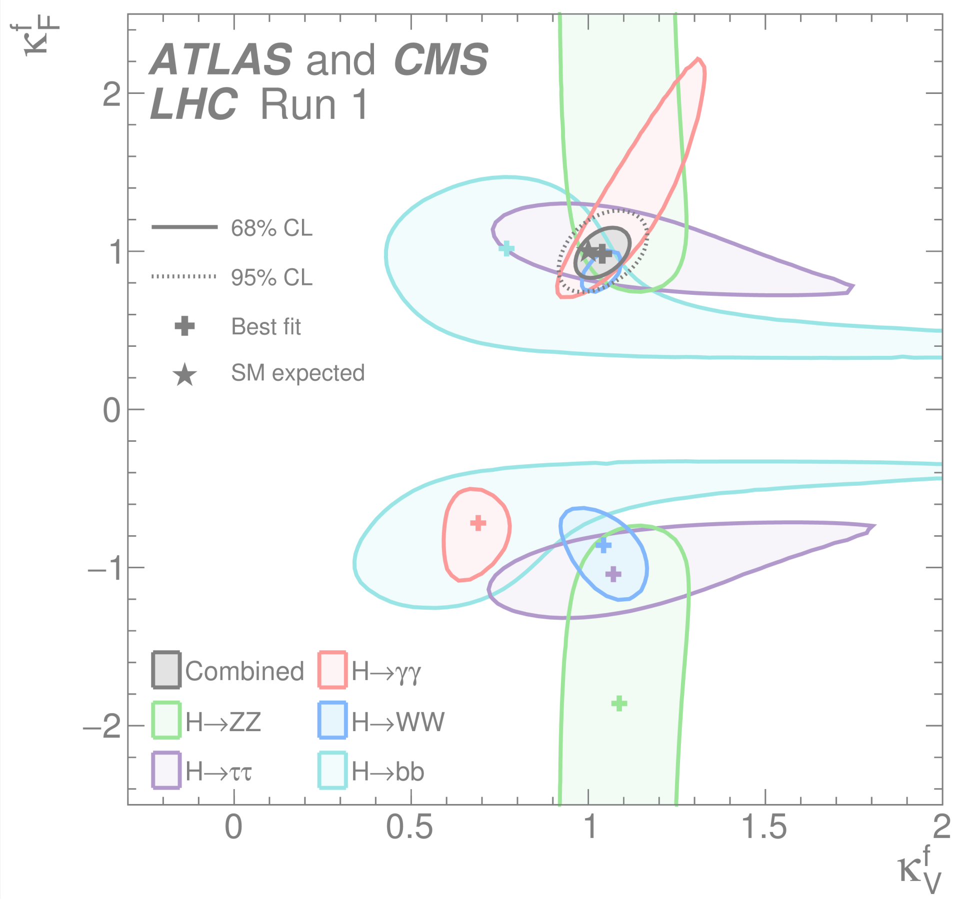
Machine learning



High-dimensional
event data x



Estimator of the
likelihood $p(x|\theta)$



Constraints on
parameters θ

LHC measurements as a likelihood-free inference problem

Modelling particle physics processes

Theory
parameters
θ

Evolution

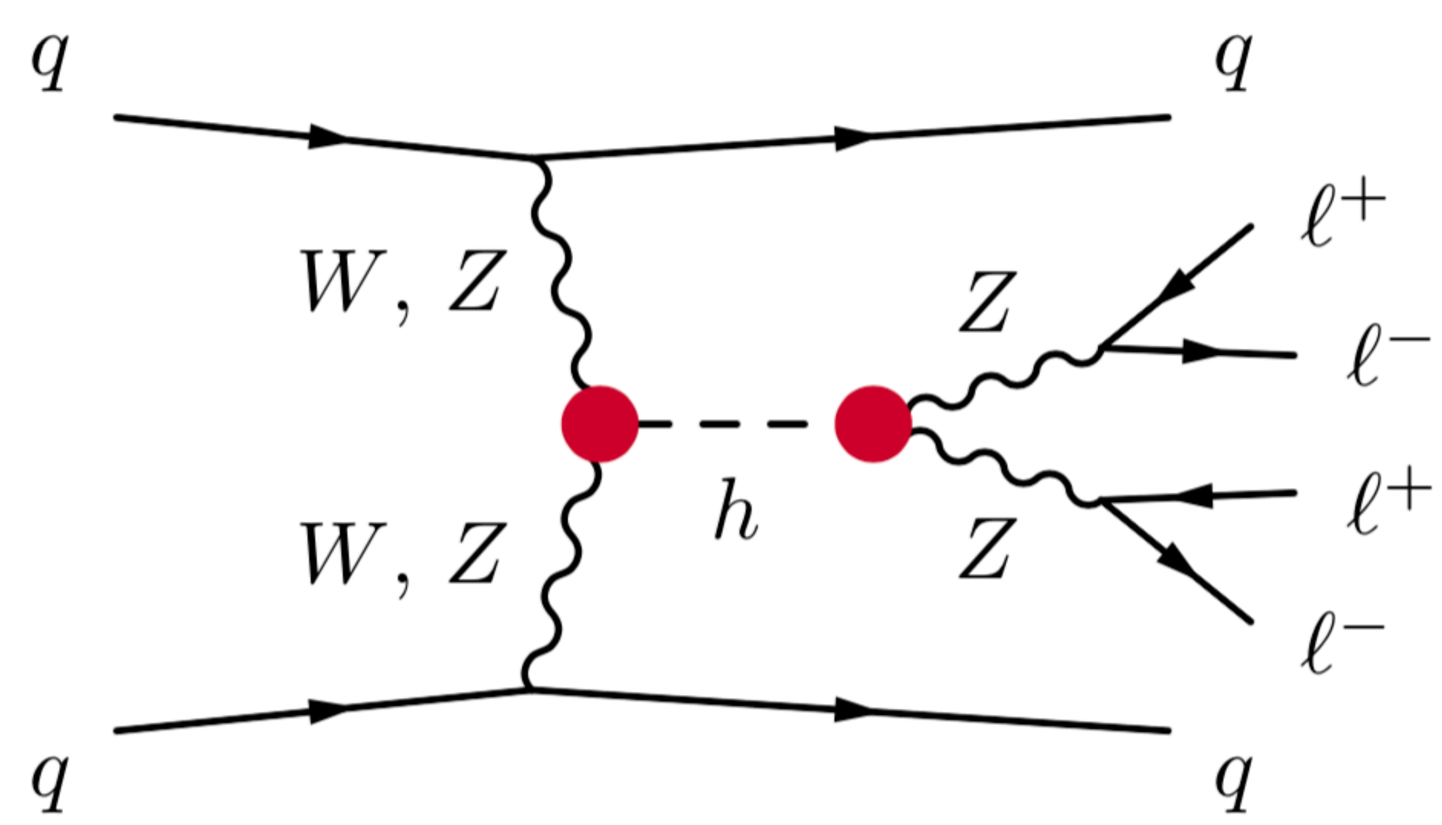
Modelling particle physics processes

Latent variables

Parton-level momenta

Theory parameters

$z_p \longleftarrow \theta$

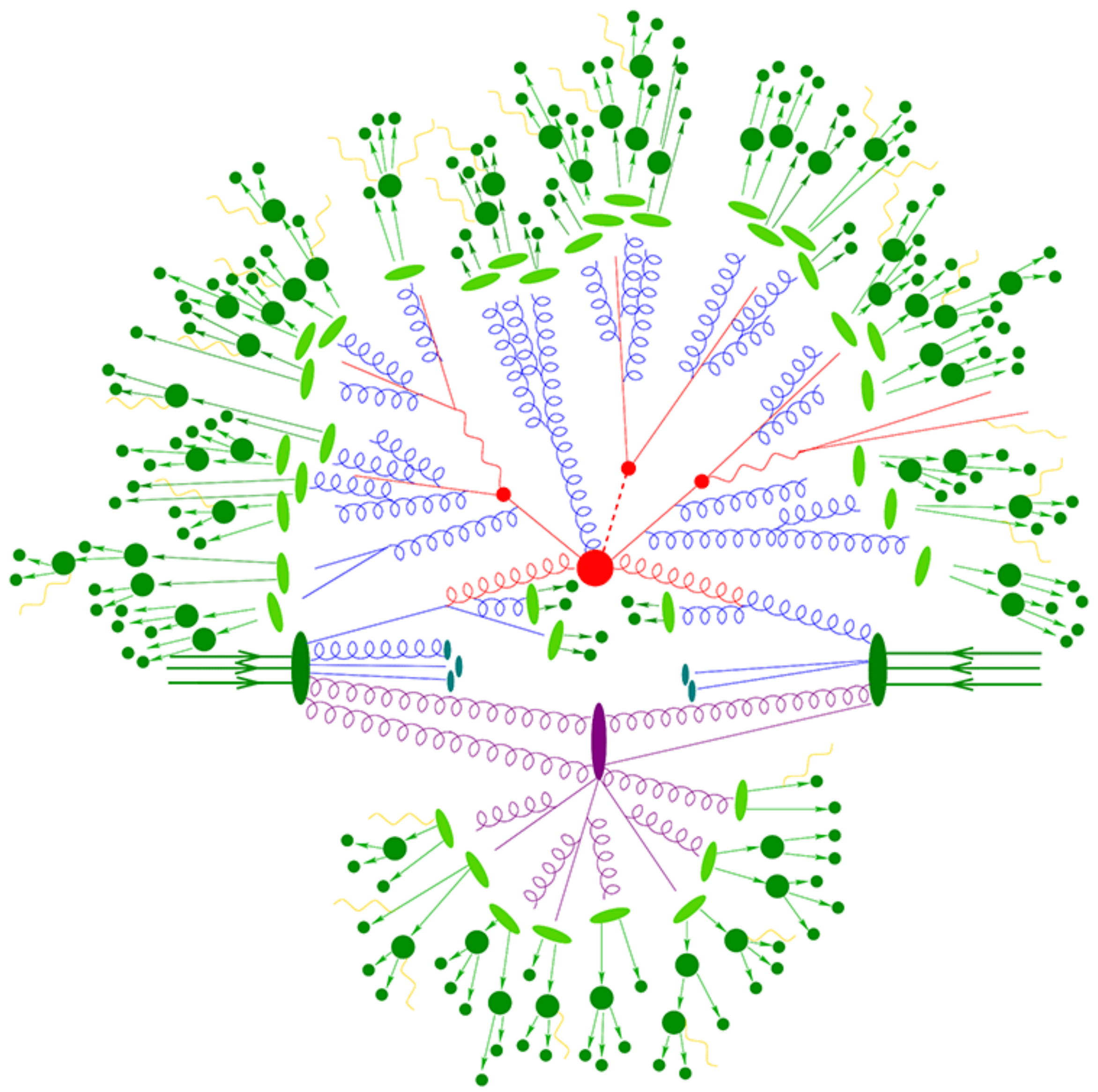


Evolution

Modelling particle physics processes

Latent variables

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



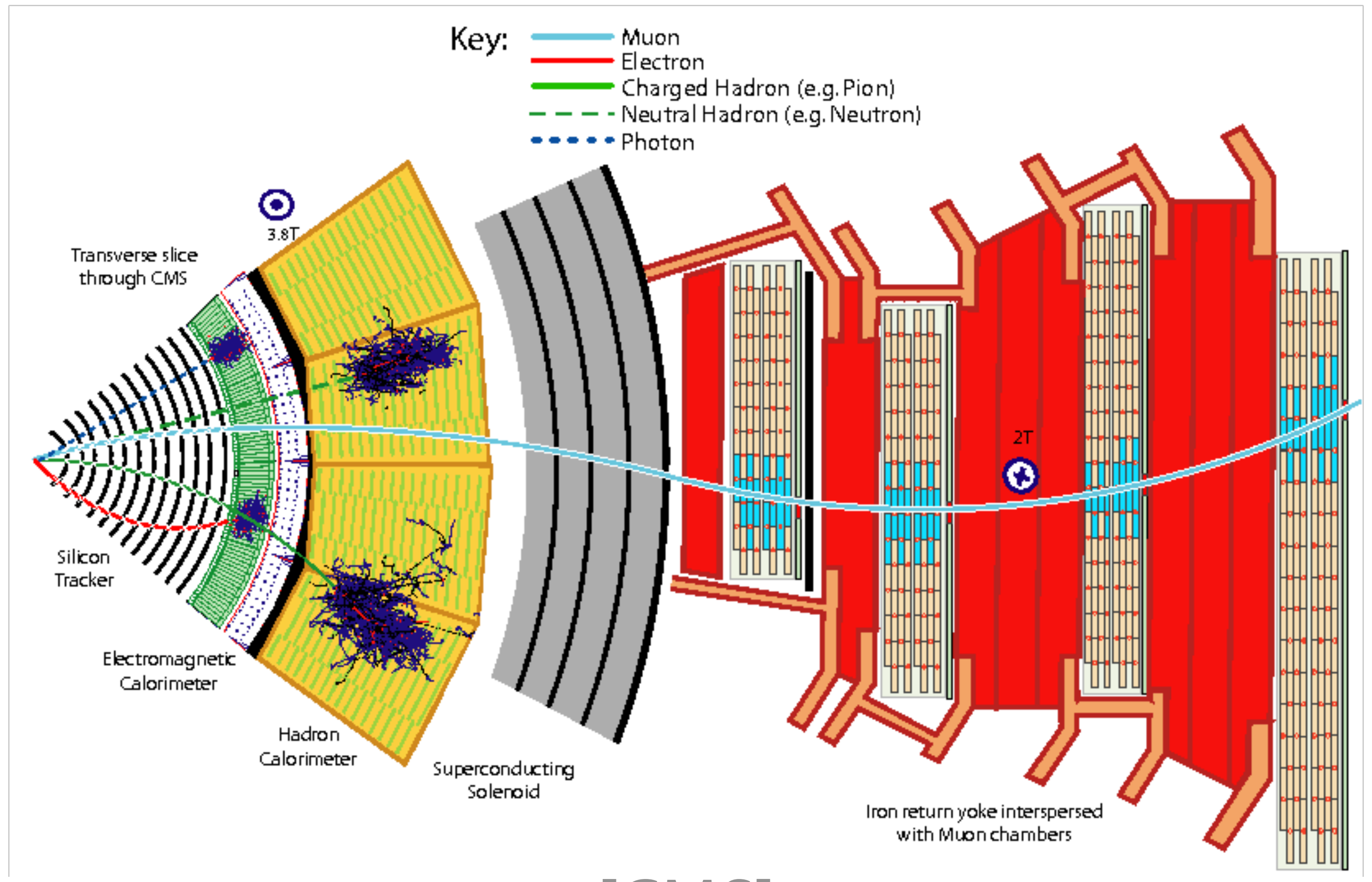
[F. Krauss]

Evolution

Modelling particle physics processes

Latent variables

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



[CMS]

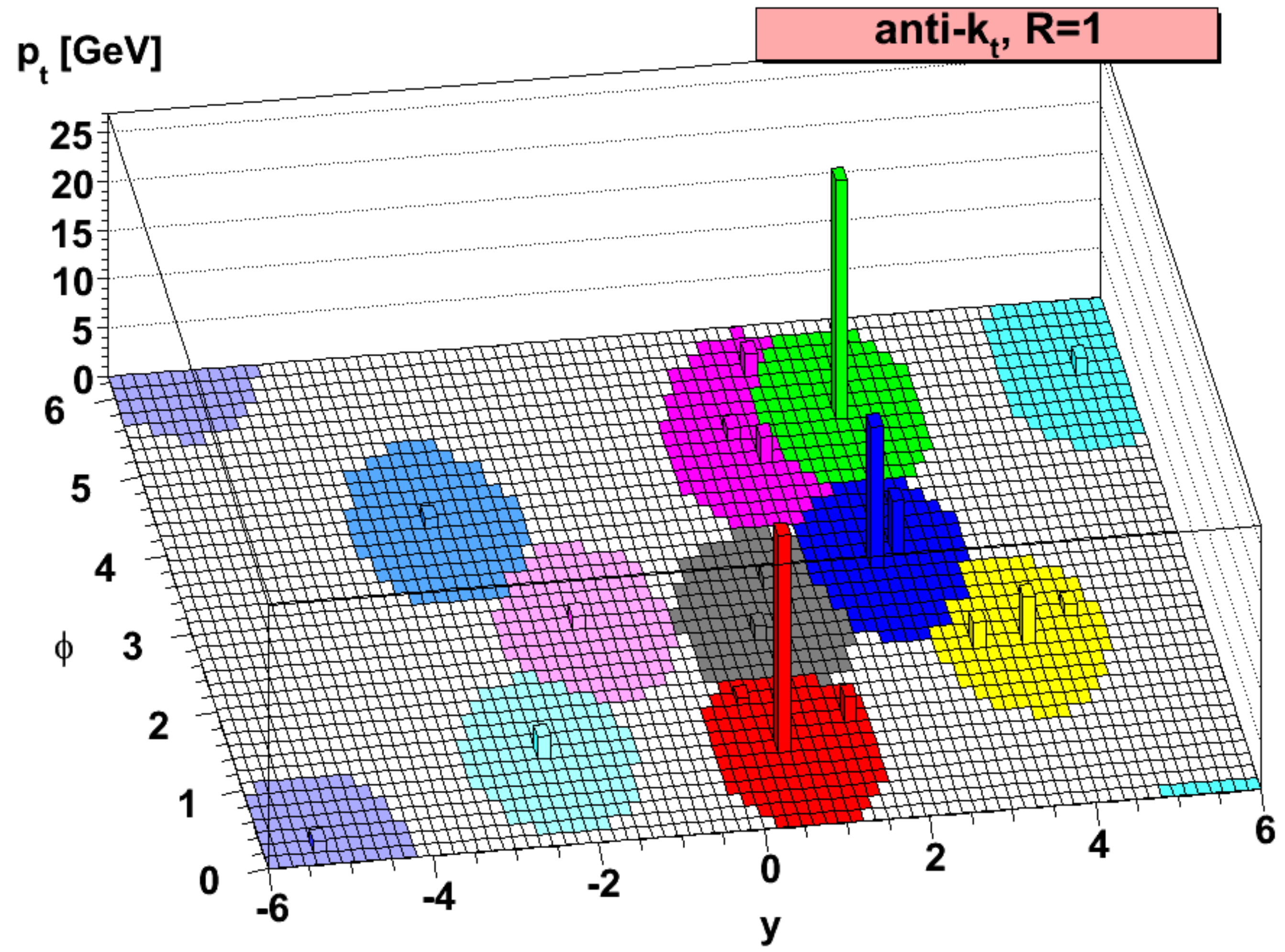
$\longleftarrow$

Evolution

Modelling particle physics processes

Latent variables

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



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

Evolution

Modelling particle physics processes

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)

Modelling particle physics processes

Latent variables

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

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

$\longrightarrow$

Inference

Modelling particle physics processes

Latent variables

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

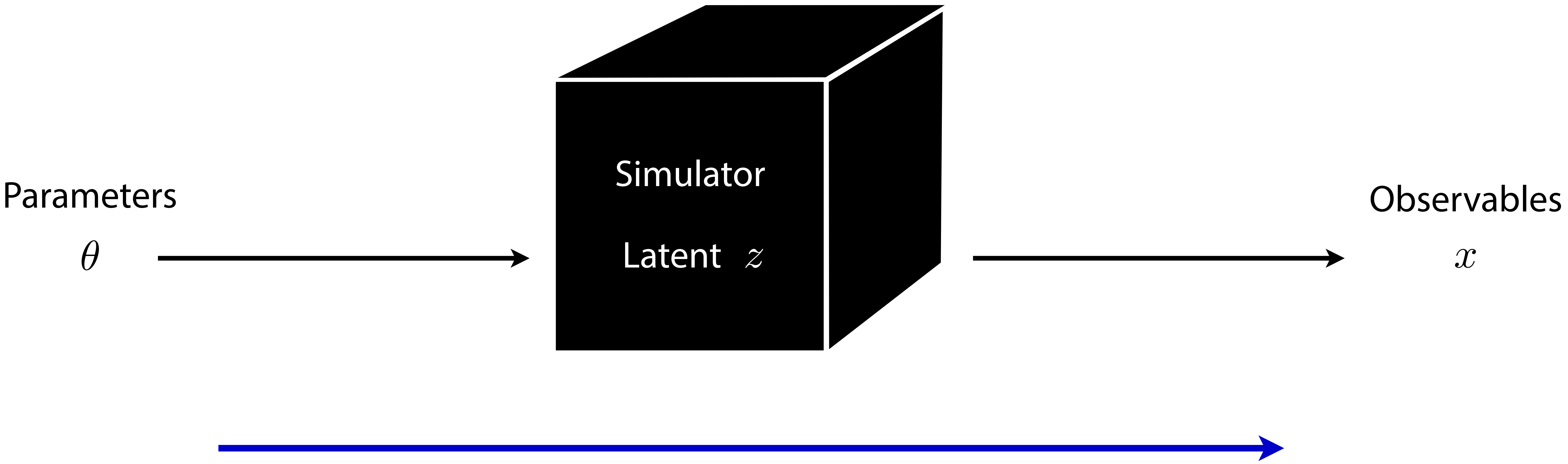
$$p(x|\theta) = \boxed{\int \mathrm{d}z_d \int \mathrm{d}z_s \int \mathrm{d}z_p} \; p(x|z_d) \quad p(z_d|z_s) \quad p(z_s|z_p) \quad p(z_p|\theta)$$

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

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

Inference

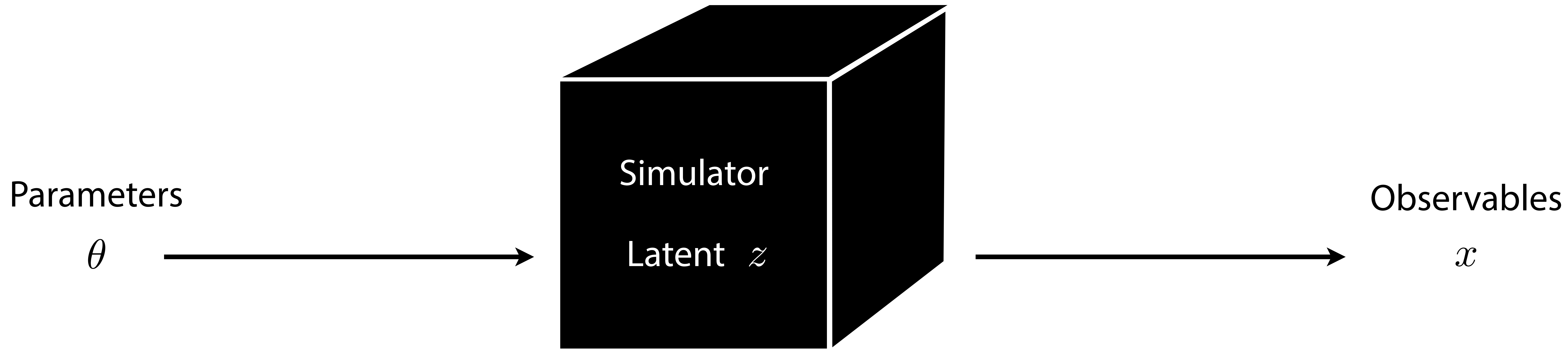
Likelihood-free inference / implicit models



Prediction:

- Well-understood mechanistic model
- Simulator can generate samples

Likelihood-free inference / implicit models

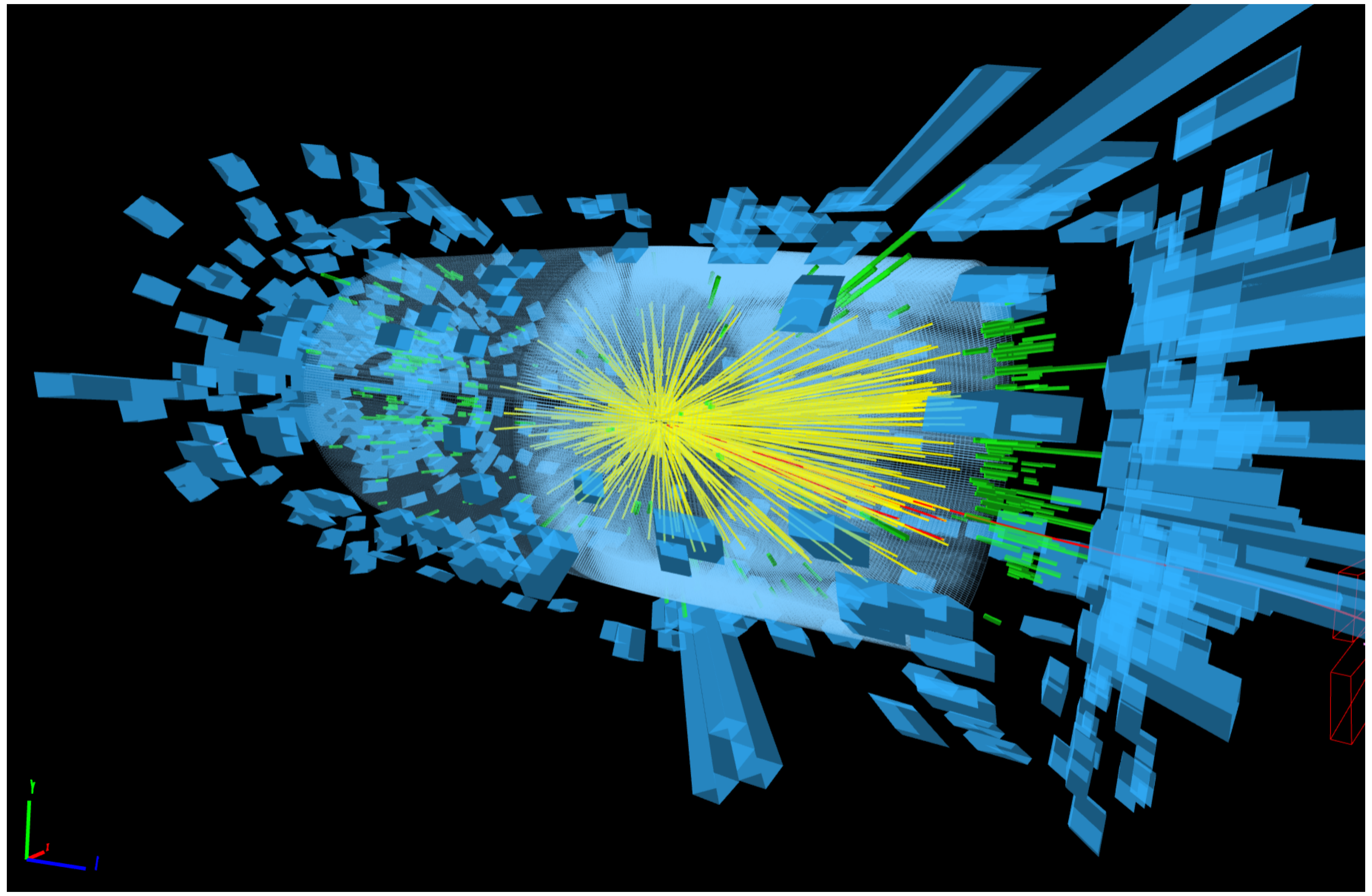


Prediction:
- Well-understood mechanistic model
- Simulator can generate samples

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

Why has that not stopped us before?

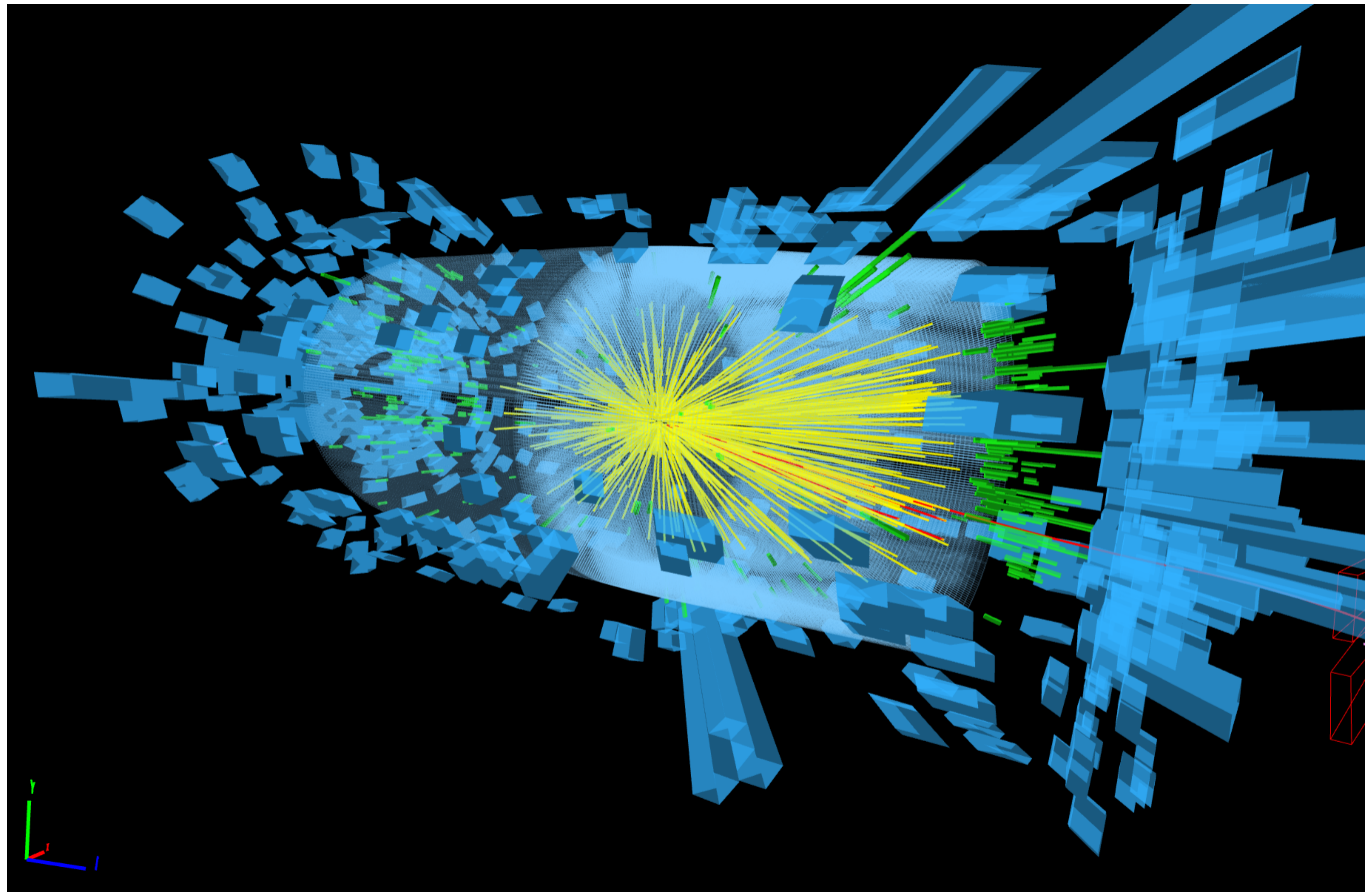
Solve it with summary statistics

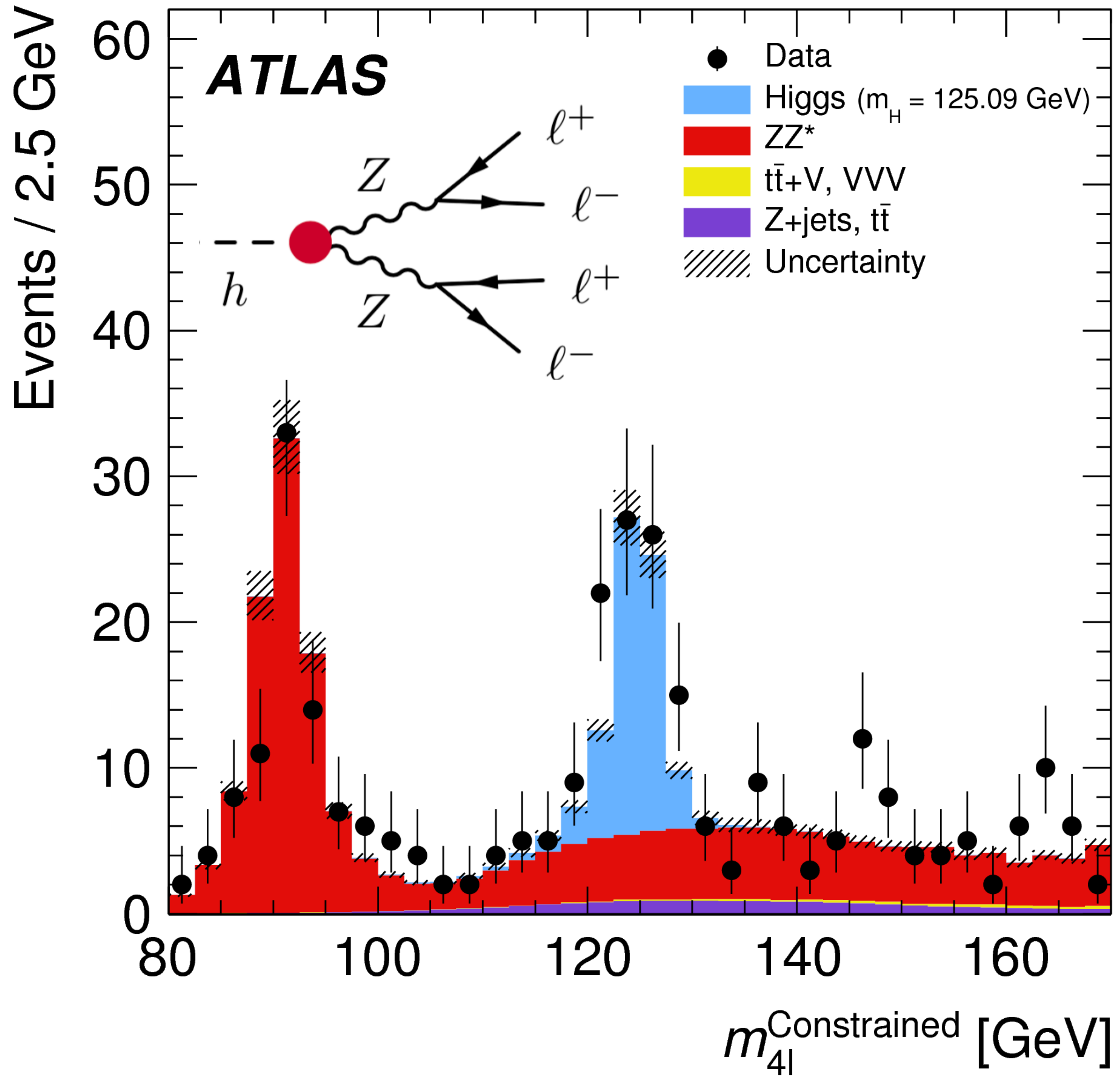


High-dimensional event data x

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

Solve it with summary statistics





[ATLAS 1712.02304]

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 EFT measurements?

- Choosing summary statistics x' is difficult and problem-dependent

- Often there is no single good standard variable — compressing to any x' loses information!
  [JB, K. Cranmer, F. Kling, T. Plehn 1612.05261; JB, F. Kling, T. Plehn , T. Tait 1712.02350]

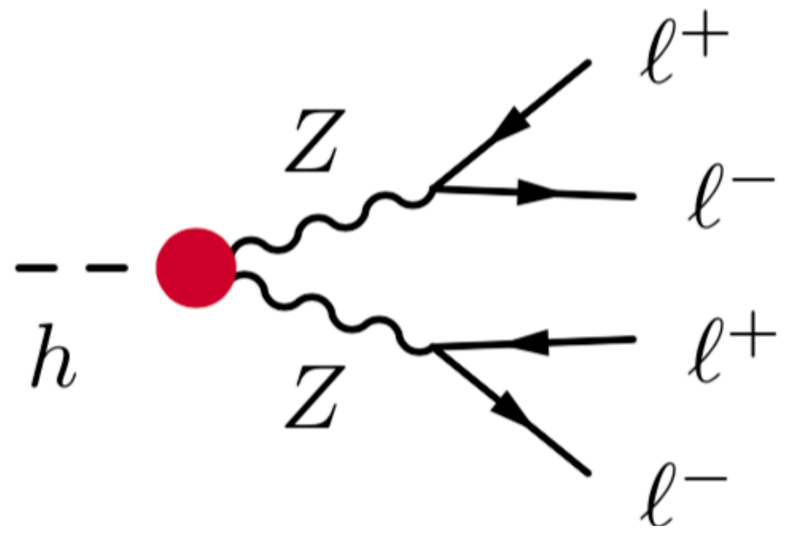
- Ideally: analyze high-dimensional x including all correlations
  ("fully differential cross section")

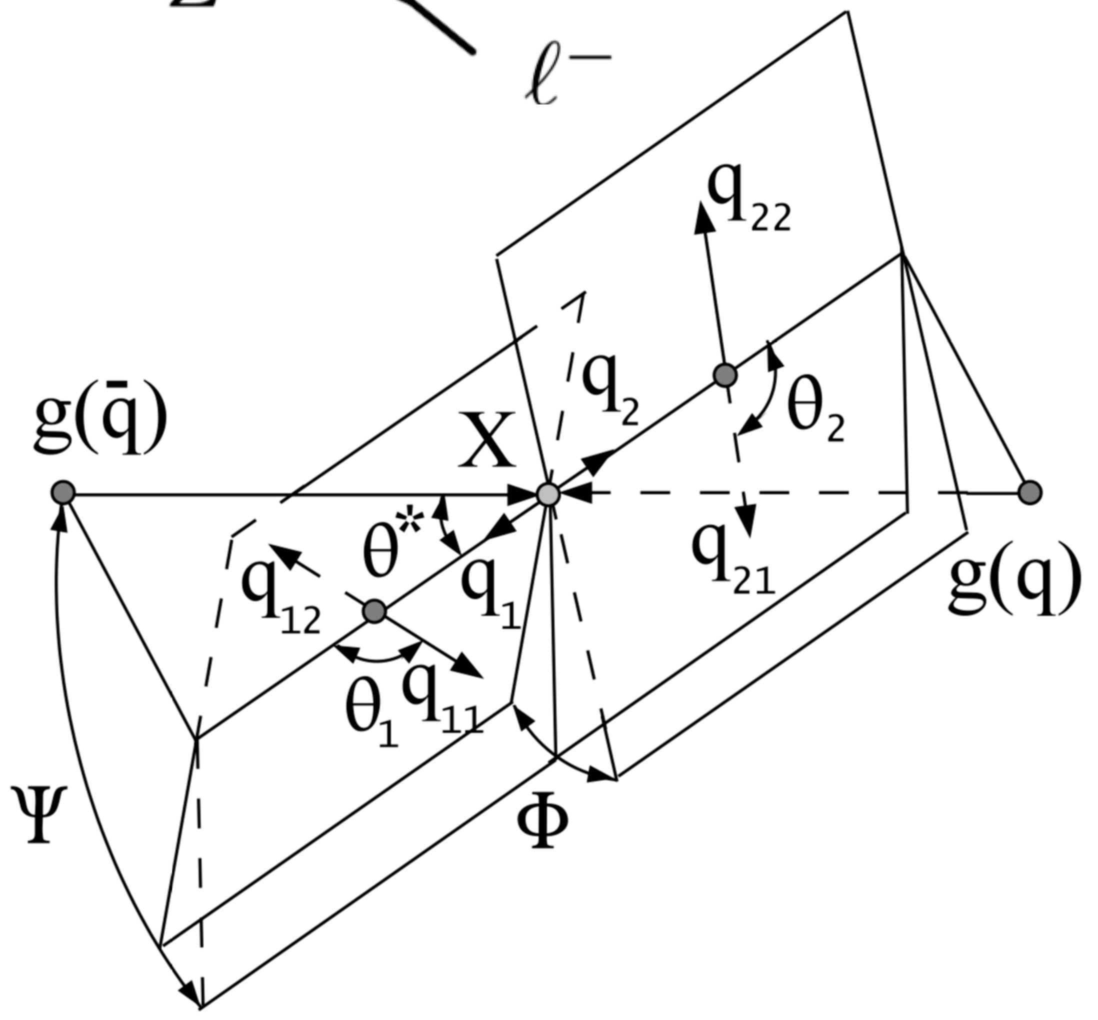
Summary statistics for EFT measurements?

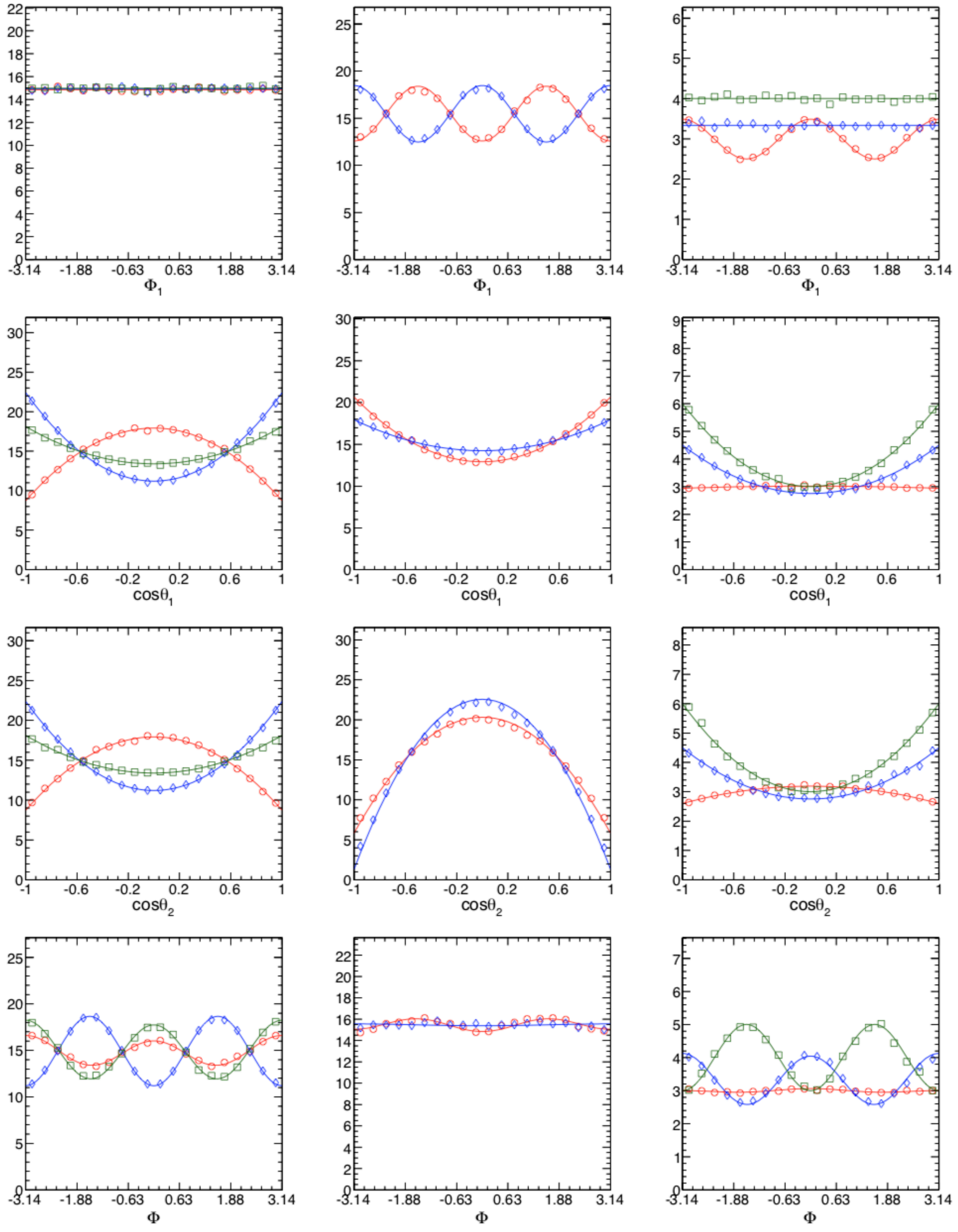
- Choosing summary statistics x' is difficult and problem-dependent

- Often there is no single good standard variable — 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 high-dimensional x including all correlations
  ("fully differential cross section")







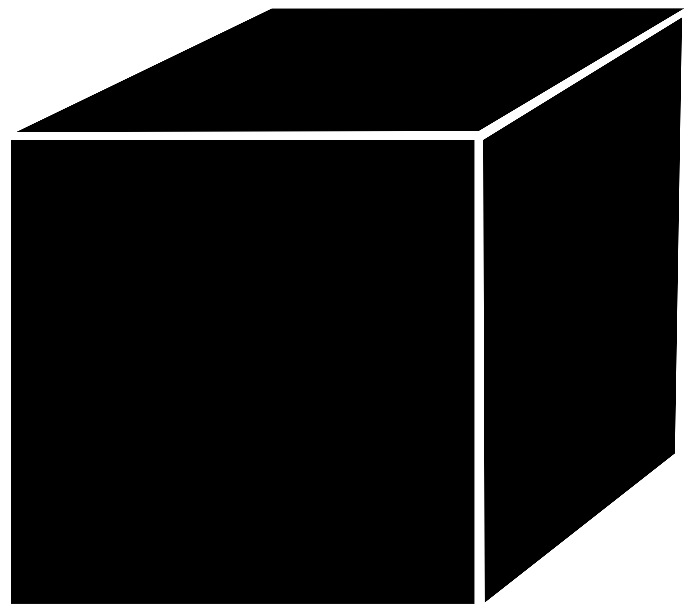
[Bolognesi et al. 1208.4018]

An incomplete list of 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, …

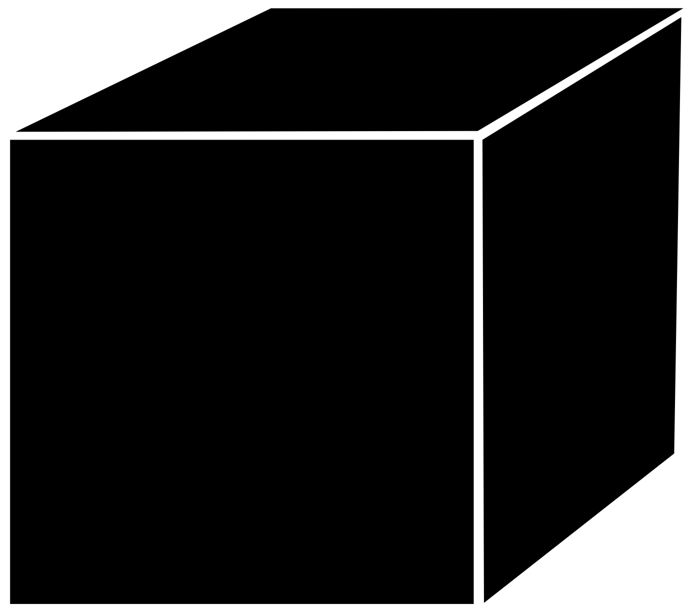


An incomplete list of likelihood-free inference methods

Treat simulator as black box:

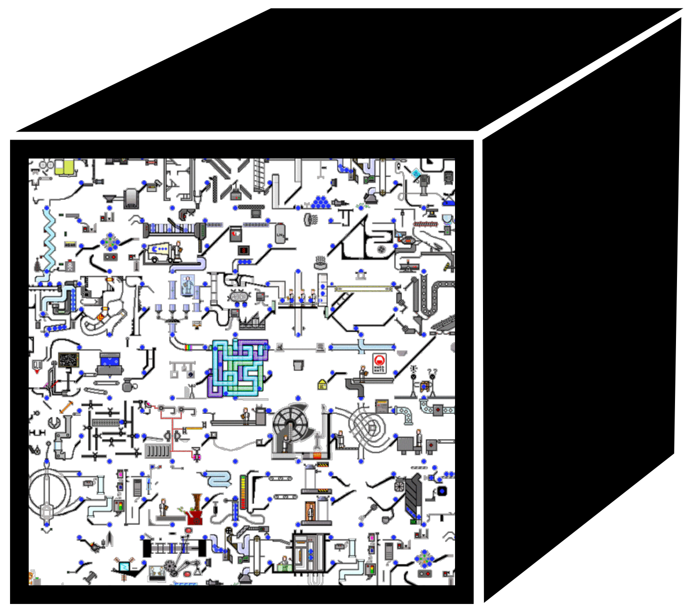
- Histograms of observables, Approximate Bayesian Computation
  Rely on summary statistics

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



Use physics insights:

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

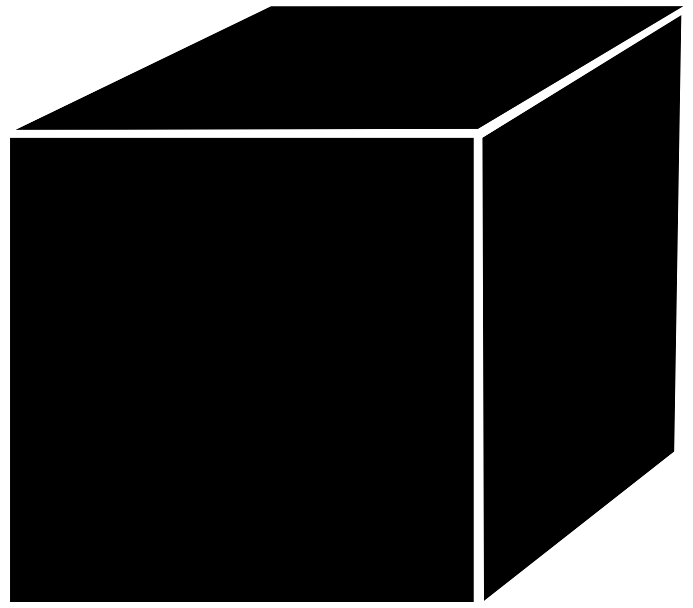


An incomplete list of likelihood-free inference methods

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, …

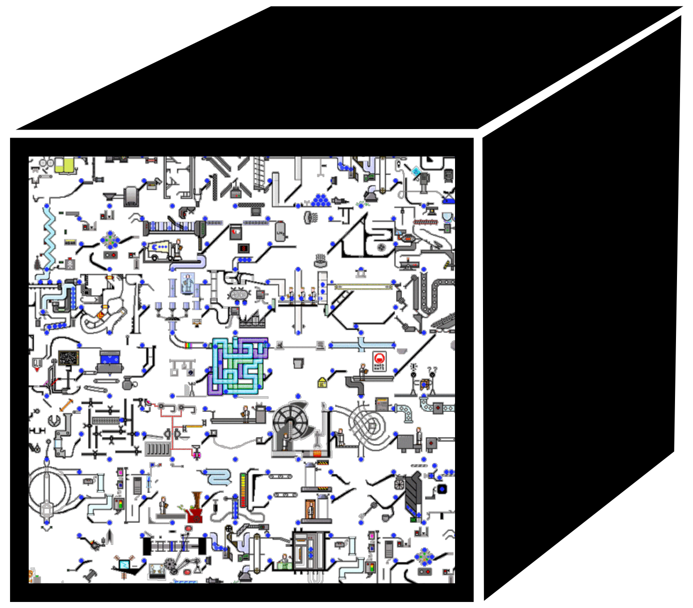


Use physics insights:

- Matrix Element Method, Optimal Observables, Shower Deconstruction, Event 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]

What if we could estimate the likelihood…

- for high-dimensional observables, including correlations?

  like MEM: no need to pick summary statistics

- including state-of-the-art shower and detector models?

  allowing for extra radiation, no need for transfer functions

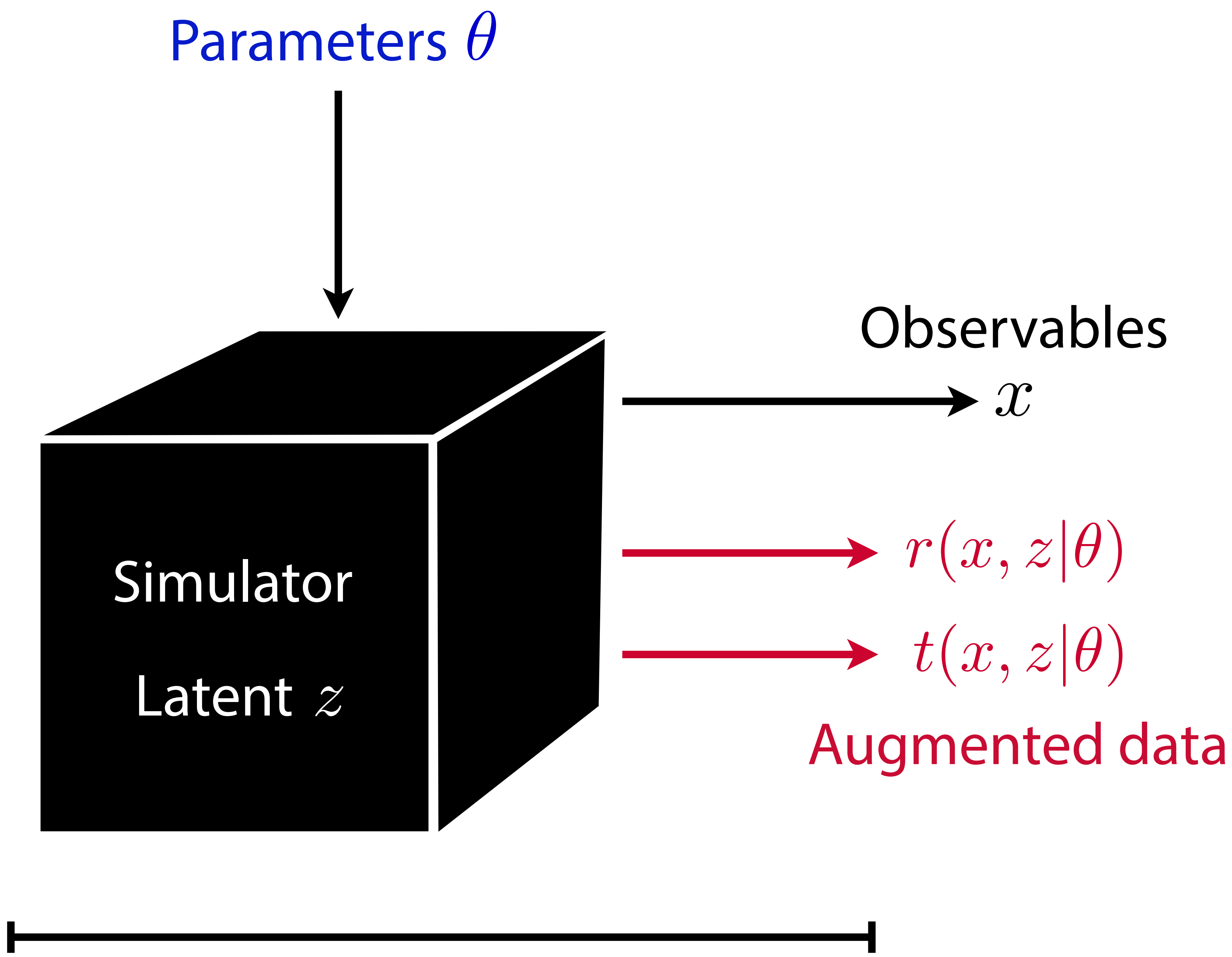
- in microseconds?

  amortized inference: train once, then always evaluate fast

- requiring less training examples than established machine learning methods?

  using matrix element information: "ML version of MEM"

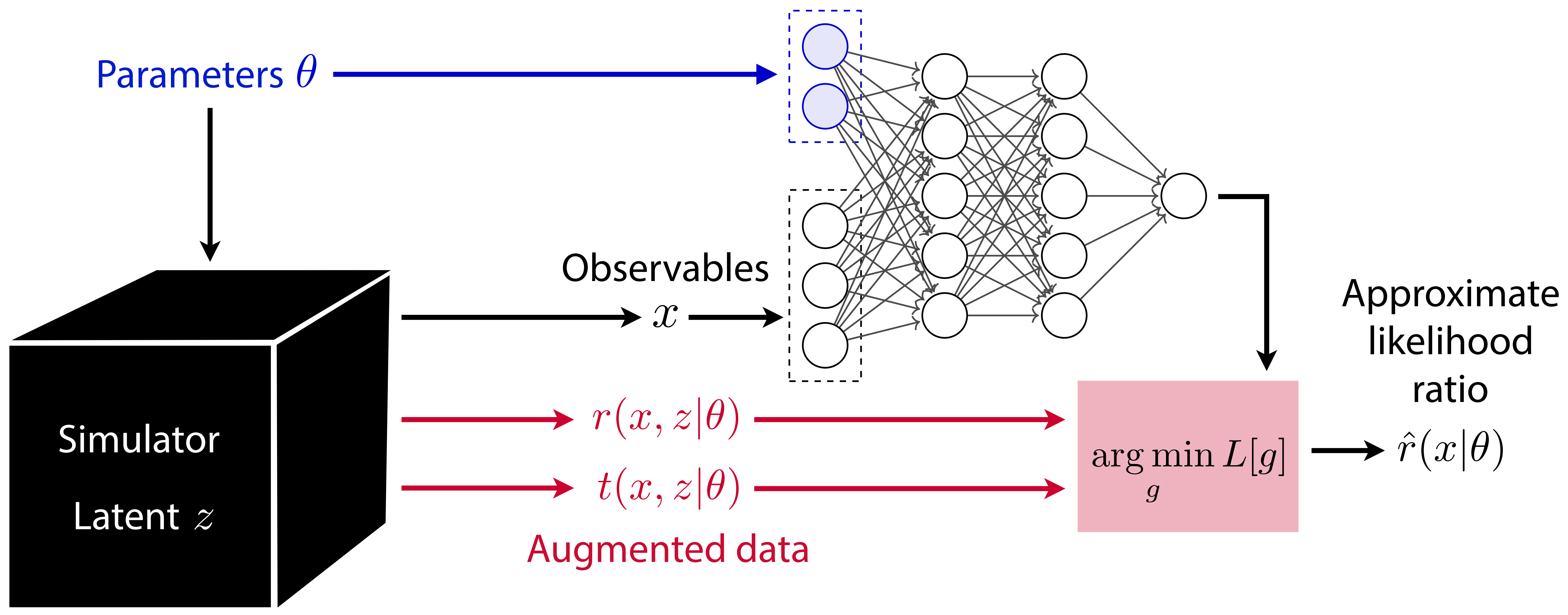
Bird’s-eye view



1. Simulation

“Mining gold”: Extract additional information from simulator

Bird's-eye view



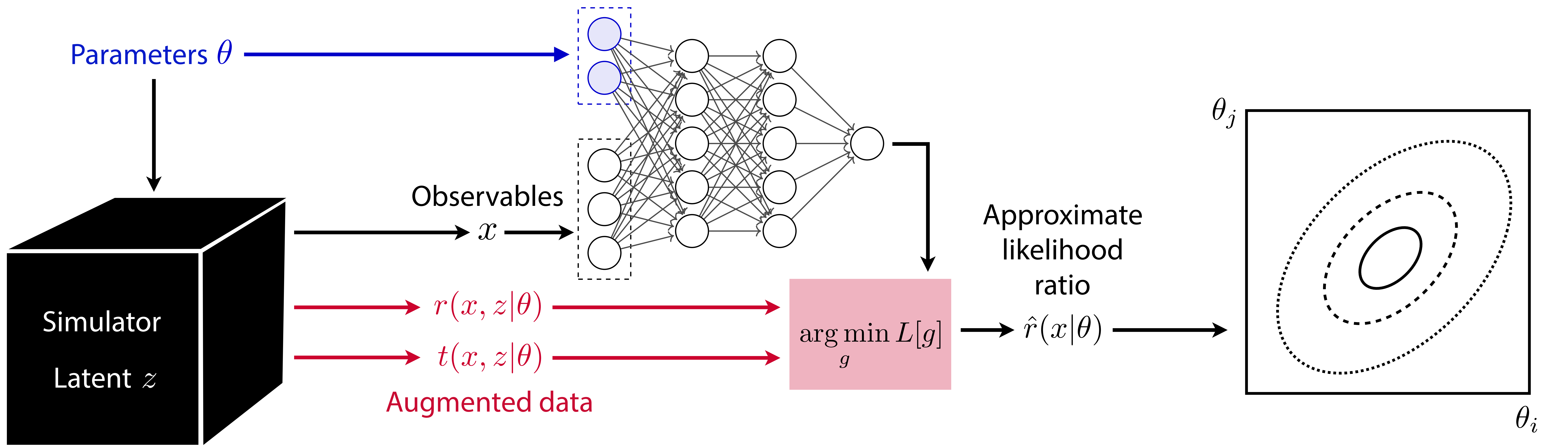
1. Simulation

"Mining gold": Extract additional information from simulator

2. Machine Learning

Use this information to train estimator for likelihood ratio

Bird's-eye view



1. Simulation

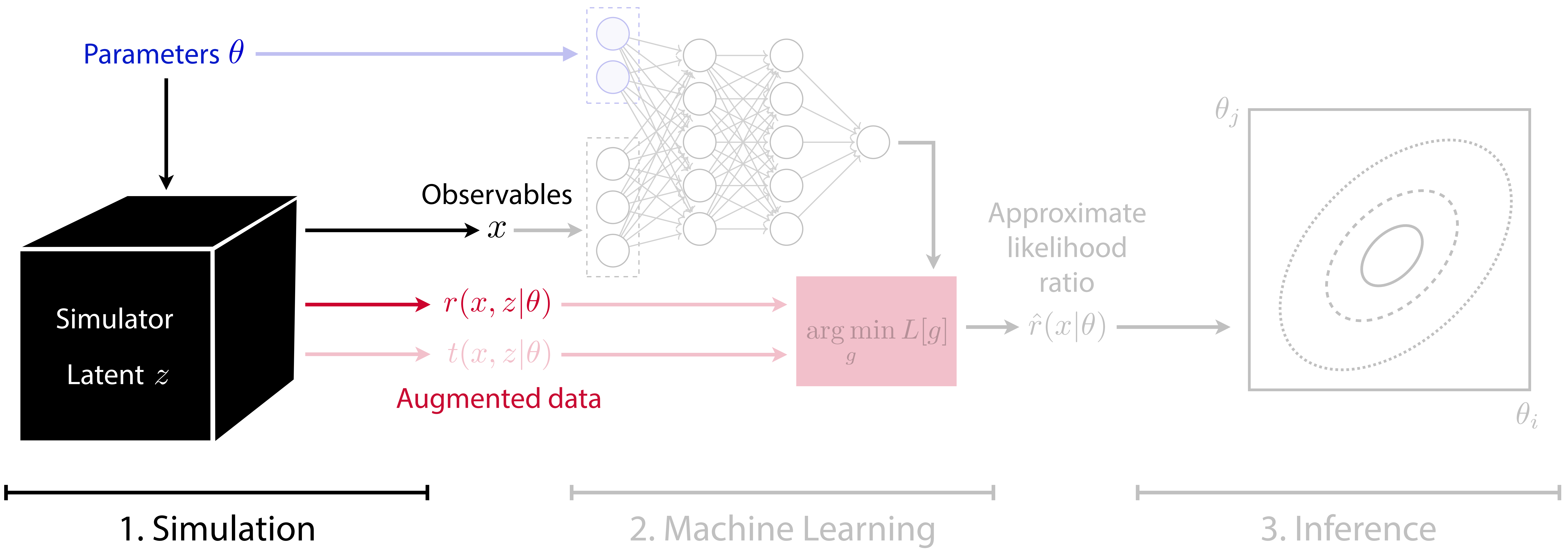
"Mining gold": Extract additional information from simulator

2. Machine Learning

Use this information to train estimator for likelihood ratio

3. Inference

Limit setting with standard hypothesis tests


Parameters θ
Observables
x
Simulator
Latent z
$r(x, z|\theta)$
$t(x, z|\theta)$
Augmented data
$\arg\min_g L[g]$
Approximate likelihood ratio
$\hat{r}(x|\theta)$
θ_j
θ_i
1. Simulation
2. Machine Learning
3. Inference

Mining gold from the simulator

Latent variables

Observables	Detector interactions	Shower splittings	Parton-level momenta	Theory parameters

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

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

Intractable integrals

Mining gold from the simulator

Latent variables

Observables	Detector interactions	Shower splittings	Parton-level momenta	Theory parameters

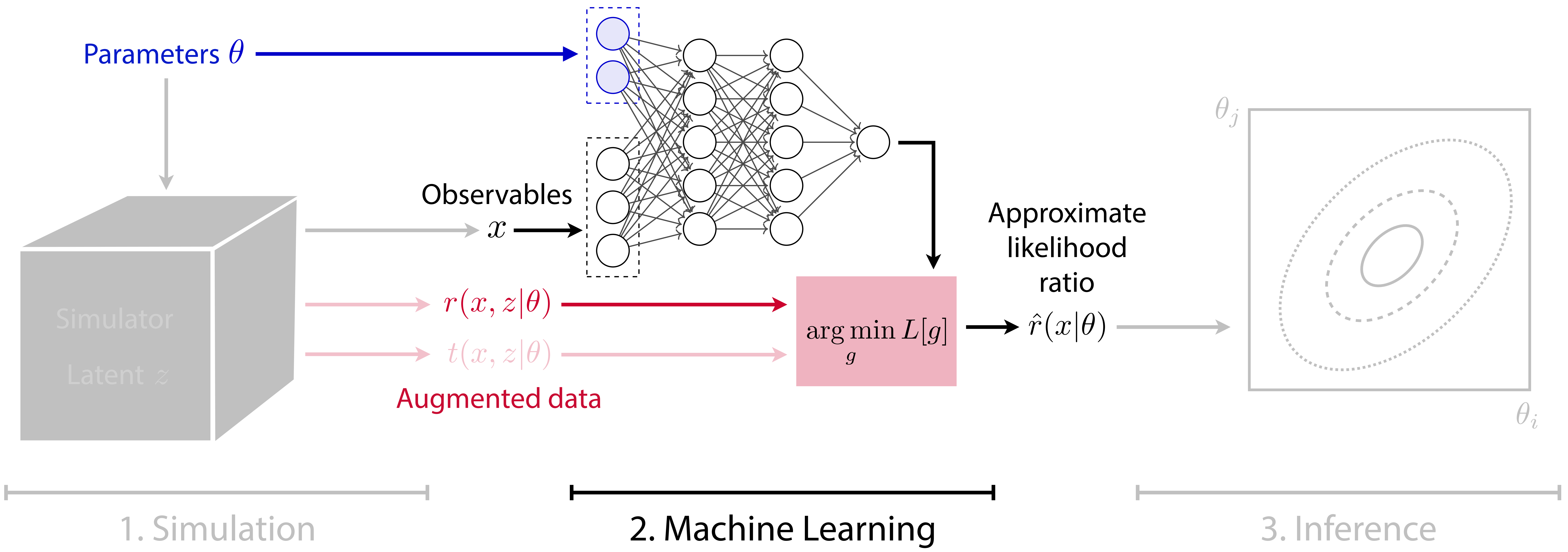
$$x \longleftarrow z_d \longleftarrow z_s \longleftarrow z_p \longleftarrow \theta$$

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

Parton-level likelihood
is given by matrix element
and can be evaluated!

⇒ For each simulated event, we can calculate the joint likelihood ratio which depends on the specific evolution of the simulation:

$$r(x, z|\theta_0, \theta_1) \equiv \frac{p(x, z_d, z_s, z_p|\theta_0)}{p(x, z_d, z_s, z_p|\theta_1)} = \frac{p(x|z_d)}{p(x|z_d)} \frac{p(z_d|z_s)}{p(z_d|z_s)} \frac{p(z_s|z_p)}{p(z_s|z_p)} \boxed{\frac{p(z_p|\theta_0)}{p(z_p|\theta_1)} \sim \frac{|\mathcal{M}(z_p|\theta_0)|^2}{|\mathcal{M}(z_p|\theta_1)|^2}}$$


Parameters θ
Observables
x
Simulator
Latent z
$r(x, z|\theta)$
$t(x, z|\theta)$
Augmented data
$\arg\min_g L[g]$
Approximate
likelihood
ratio
$\hat{r}(x|\theta)$
θ_j
θ_i
1. Simulation
2. Machine Learning
3. Inference

The value of gold

We can calculate the joint likelihood ratio

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

("How much more likely is this simulated event, including all intermediate states, for θ_0 compared to θ_1?")

We want the likelihood ratio function

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

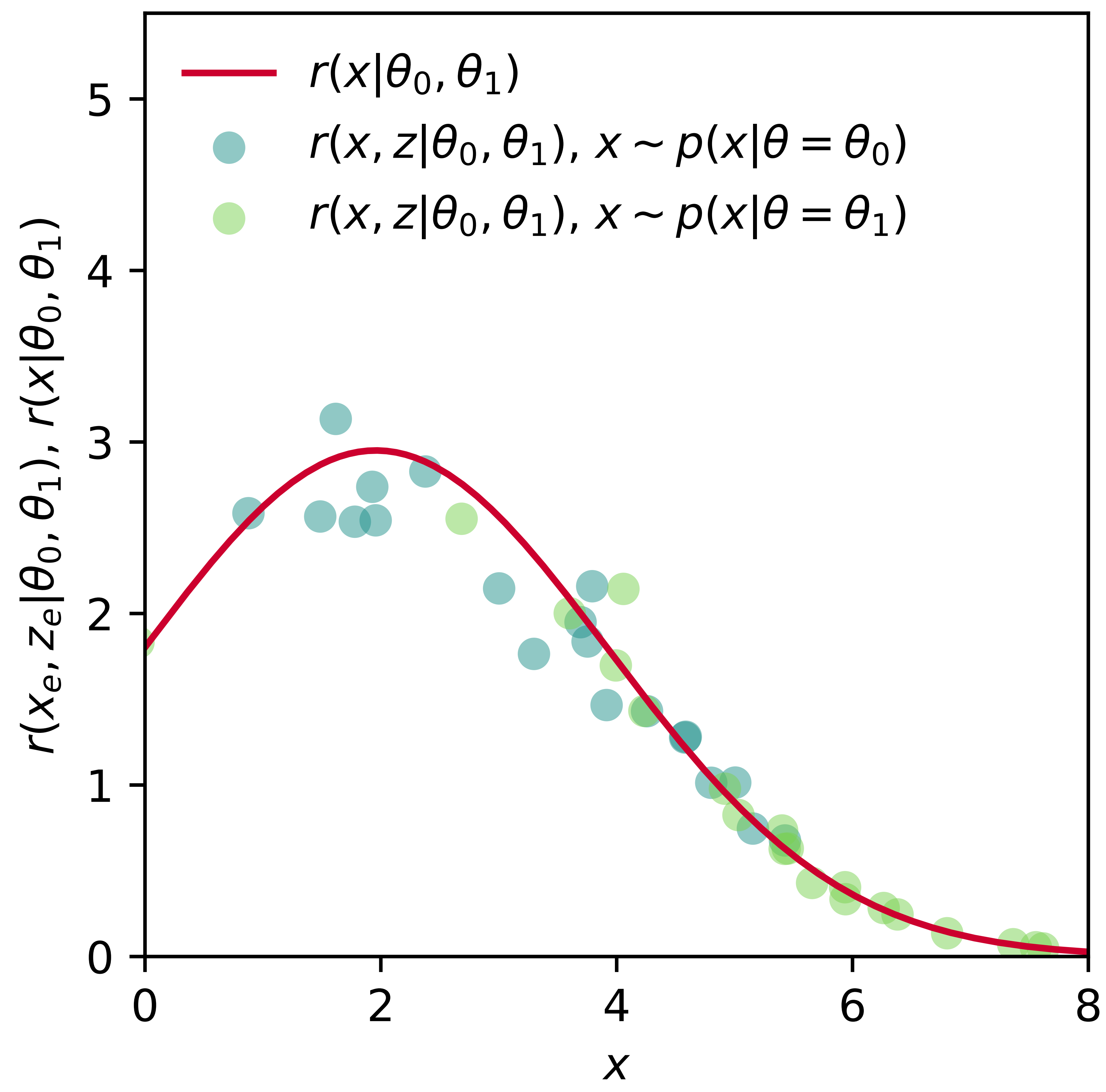
("How much more likely is the observation x for θ_0 compared to θ_1?")

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 a functional like

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

It is minimized by

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

(And we can sample from $p(x, z|\theta)$ by running the simulator.)

We want the likelihood ratio function

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

Machine learning = applied calculus of variatios

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

Variational family

Extremization

Functional with integral

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

Machine learning = applied calculus of variatios

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

Variational family

Extremization

Functional with integral

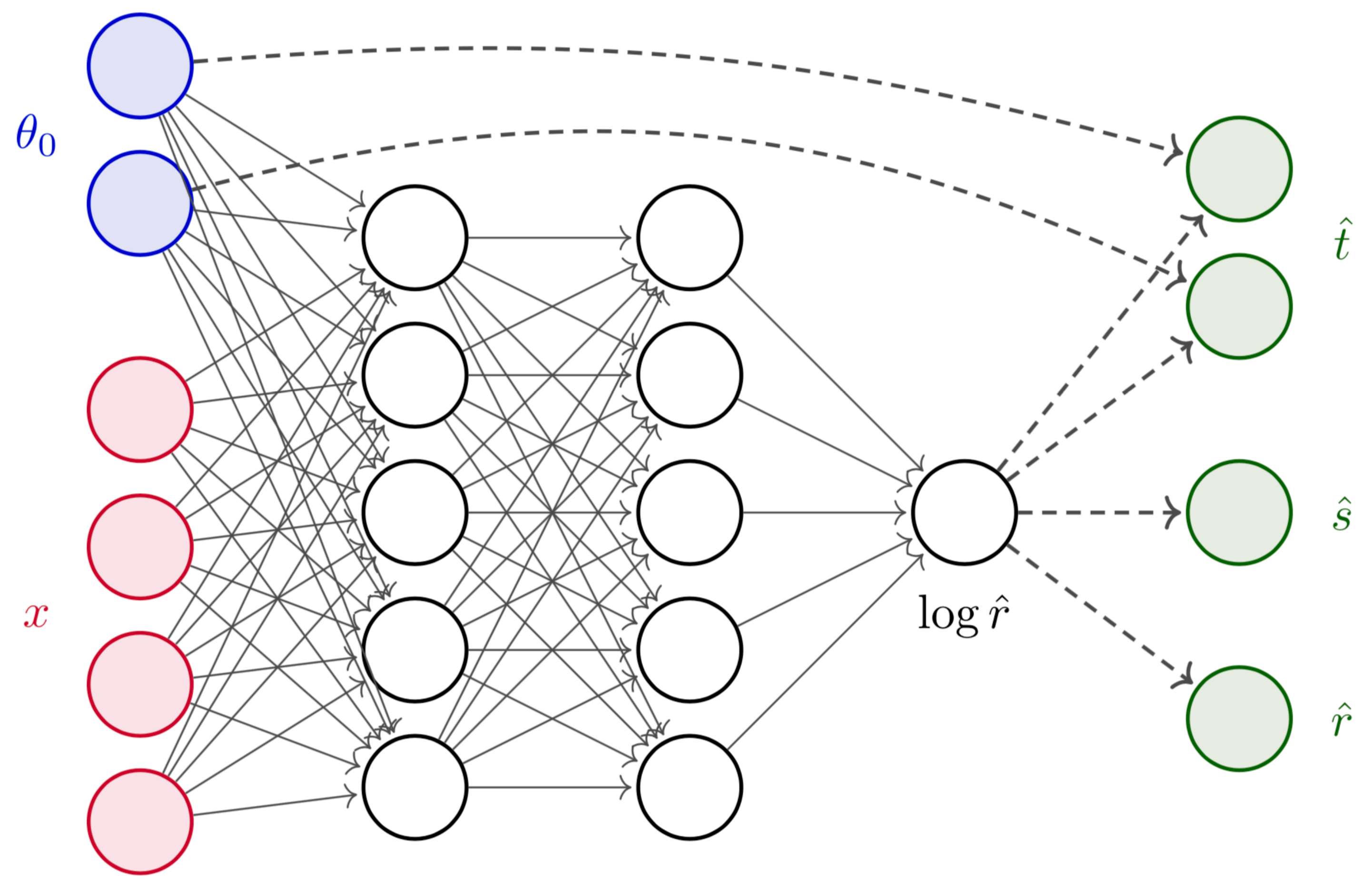
$$r(x|\theta_0, \theta_1) = \arg\min_{\hat{r}(x|\theta_0,\theta_1)} L_r[\hat{r}(x|\theta_0, \theta_1)]$$

This is where machine learning comes in!

Loss function with
finite sum over samples

Neural network

Stochastic gradient descent



Machine learning = applied calculus of variatios

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

Variational family

Extremization

Functional with integral

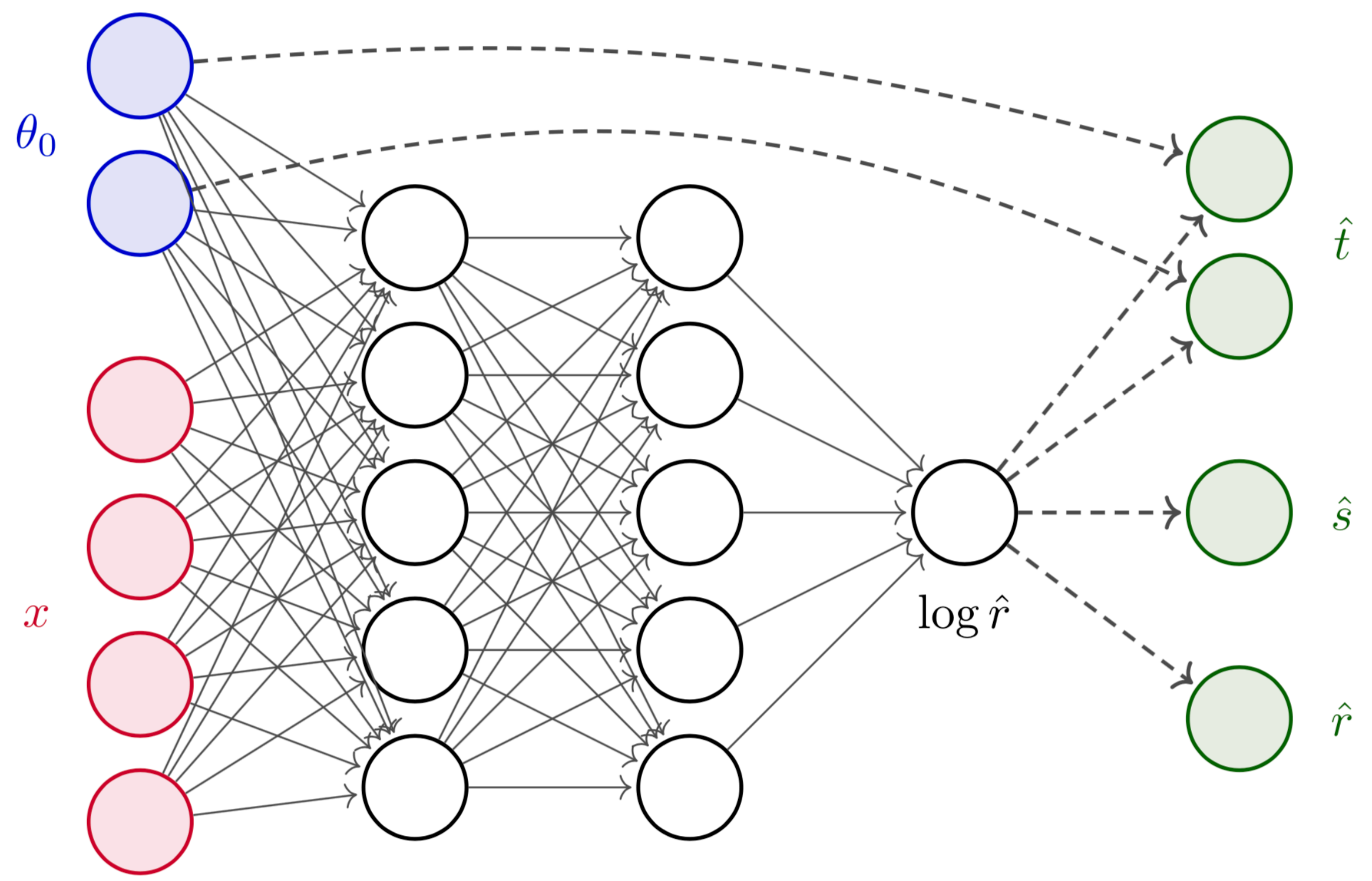
$$r(x|\theta_0, \theta_1) = \arg\min_{\hat{r}(x|\theta_0,\theta_1)} L_r[\hat{r}(x|\theta_0, \theta_1)]$$

Loss function with
finite sum over samples

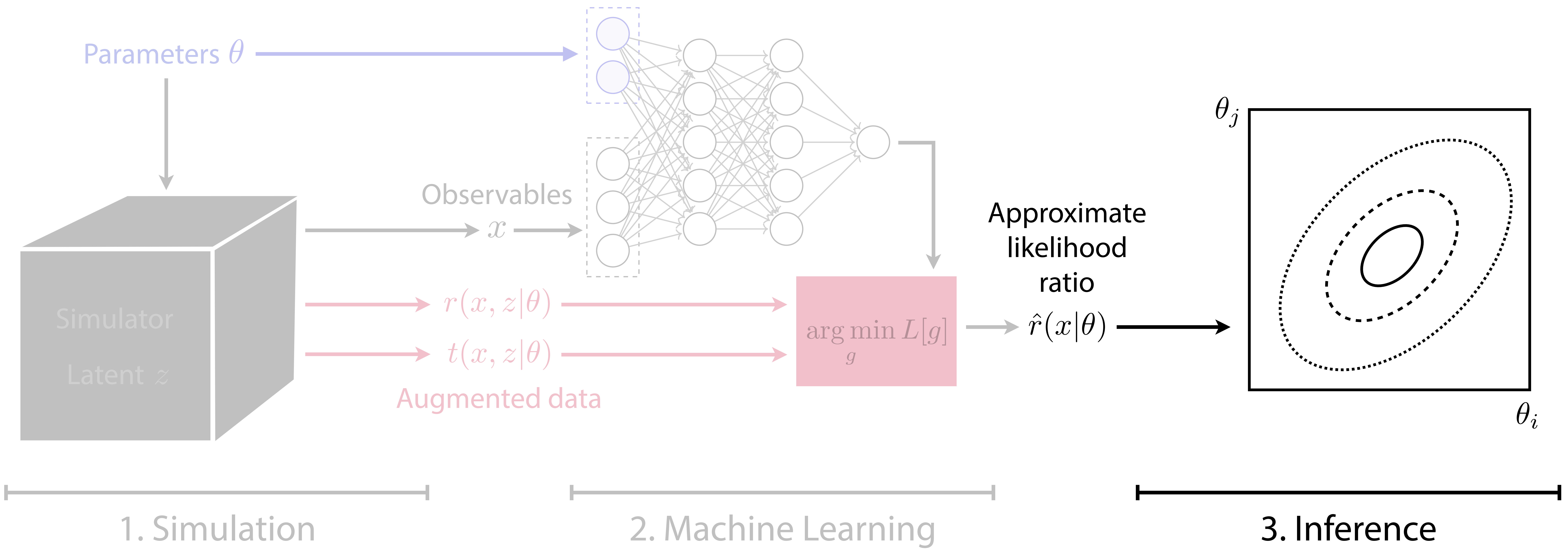
Stochastic gradient descent

This is where machine learning comes in!

Neural network

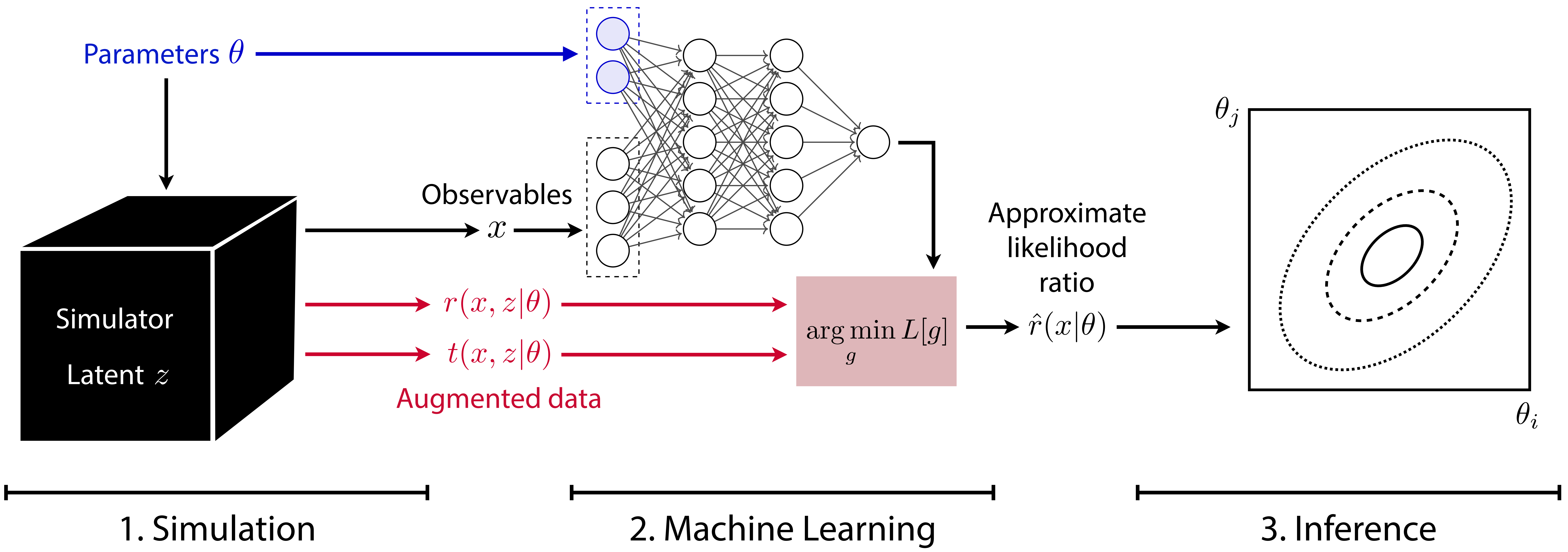


A sufficiently expressive neural network
efficiently trained in this way
with enough data will learn
the likelihood ratio function $r(x|\theta_0, \theta_1)$!



Parameters θ
Observables
x
Simulator
Latent z
$r(x, z|\theta)$
$t(x, z|\theta)$
Augmented data
$\arg\min_g L[g]$
Approximate likelihood ratio
$\hat{r}(x|\theta)$
θ_j
θ_i
1. Simulation
2. Machine Learning
3. Inference

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



"The machine learning version of the Matrix Element Method"

Learning optimal observables

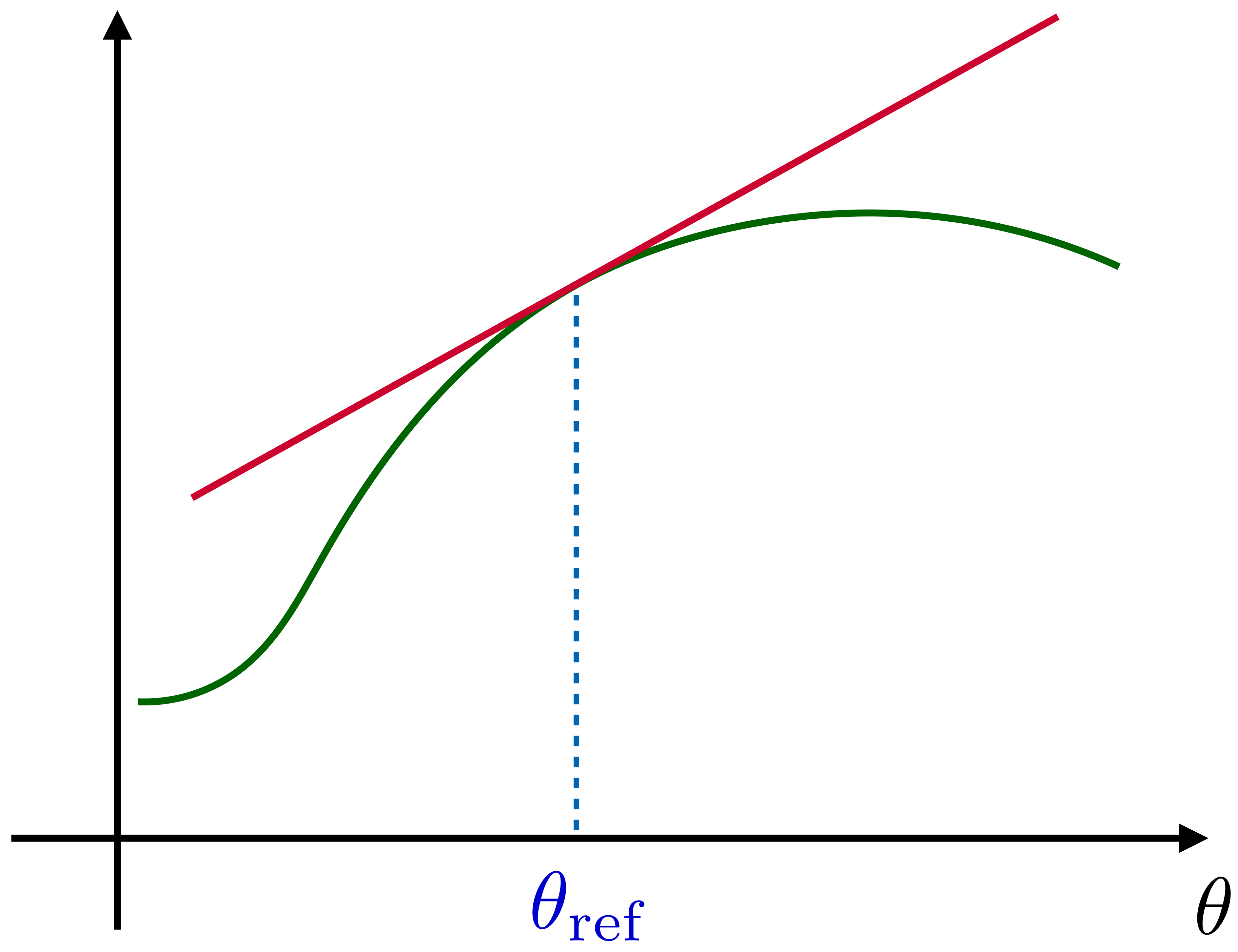
[JB, K. Cranmer, G. Louppe, J. Pavez 1805.00013, 1805.00020, 1805.12244]

The local model

[see also J. Alsing, B. Wandelt 1712.00012; J. Alsing, B. Wandelt. S. Freeney 1801.01497; P. de Castro, T. Dorigo 1806.04743; J. Alsing, B. Wandelt 1903.01473]

Taylor expansion of $\log p(x|\theta)$ around θ_{ref}:

$$\begin{aligned}\log p(x|\theta) = \log p(x|\theta_{\mathrm{ref}}) \\ + \underbrace{\nabla_\theta \log p(x|\theta)\Big|_{\theta_{\mathrm{ref}}}}_{\equiv t(x|\theta_{\mathrm{ref}})} \cdot (\theta - \theta_{\mathrm{ref}}) \\ + \mathcal{O}\left((\theta - \theta_{\mathrm{ref}})^2\right)\end{aligned}$$



The local model

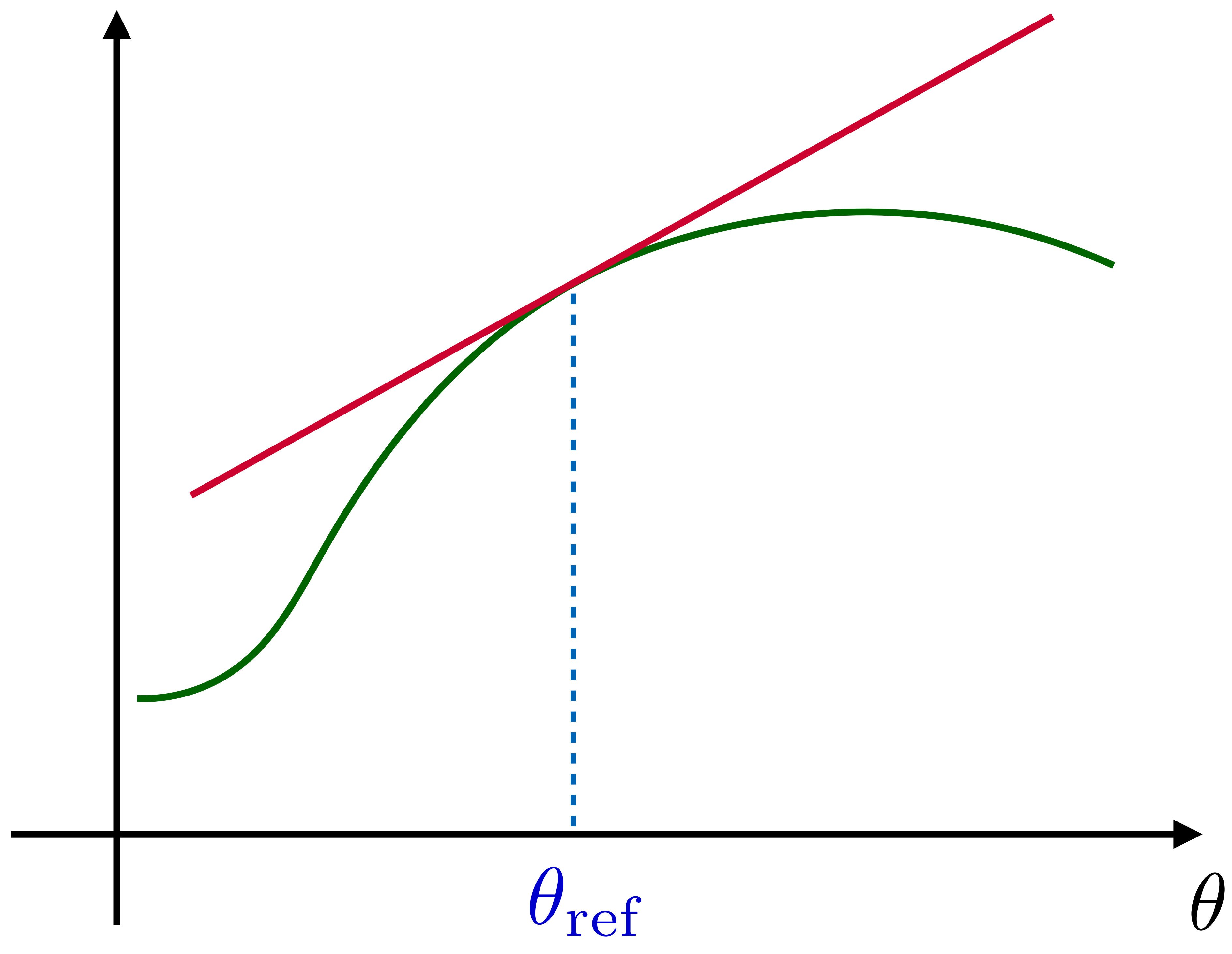
[see also J. Alsing, B. Wandelt 1712.00012; J. Alsing, B. Wandelt. S. Freeney 1801.01497; P. de Castro, T. Dorigo 1806.04743; J. Alsing, B. Wandelt 1903.01473]

Taylor expansion of $\log p(x|\theta)$ around θ_{ref}:

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

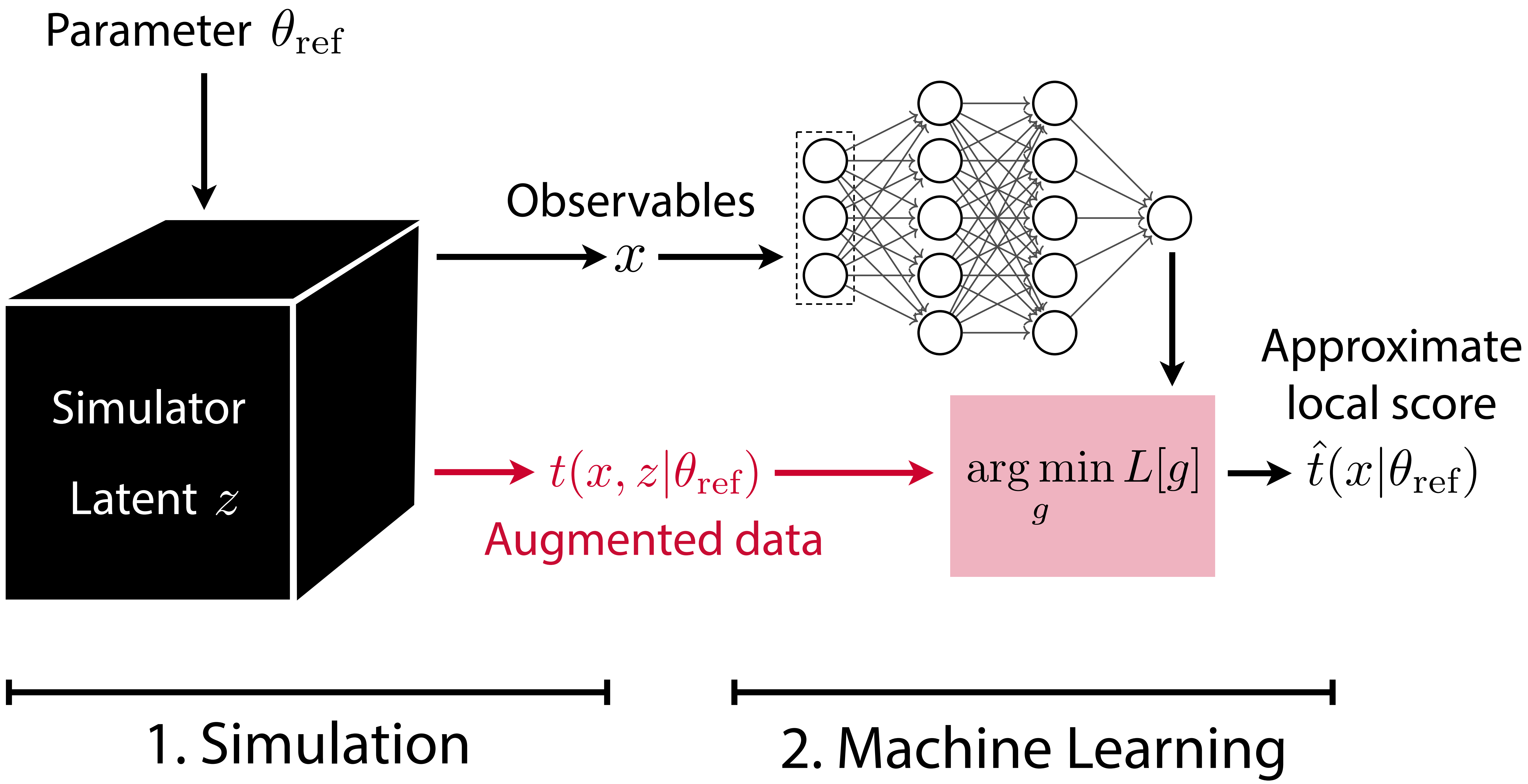
In the neighborhood of θ_{ref} (e.g. close to the SM):

- the score vector $t(x|\theta_{\text{ref}})$ is the sufficient statistics
- knowing $t(x|\theta_{\text{ref}})$ is just as powerful as knowing the full function $\log p(x|\theta)$
- $t(x|\theta_{\text{ref}})$ is the most powerful observable

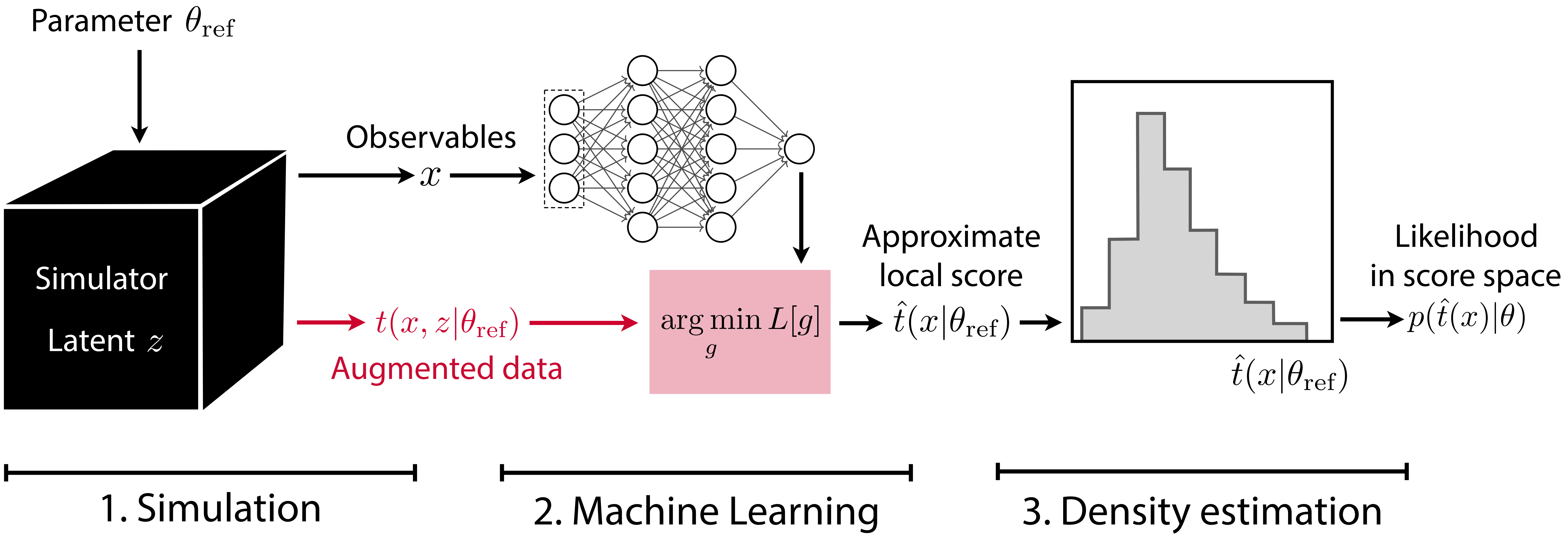


The score itself is intractable. But we can use the same trick as for the likelihood ratio!

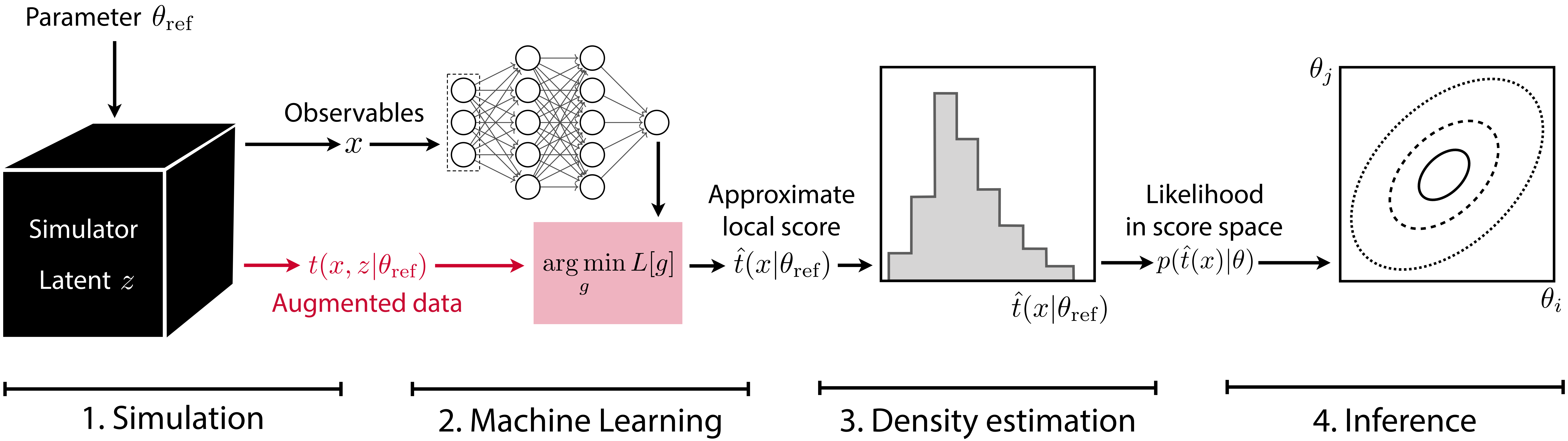
SALLY (Score approximates likelihood locally)



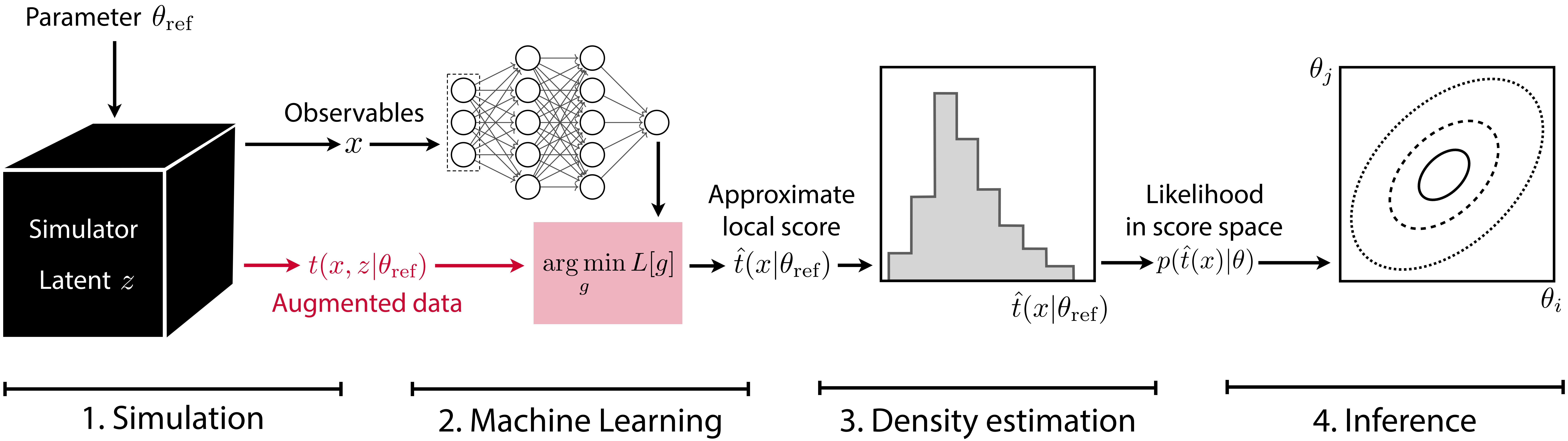
SALLY (Score approximates likelihood locally)



SALLY (Score approximates likelihood locally)



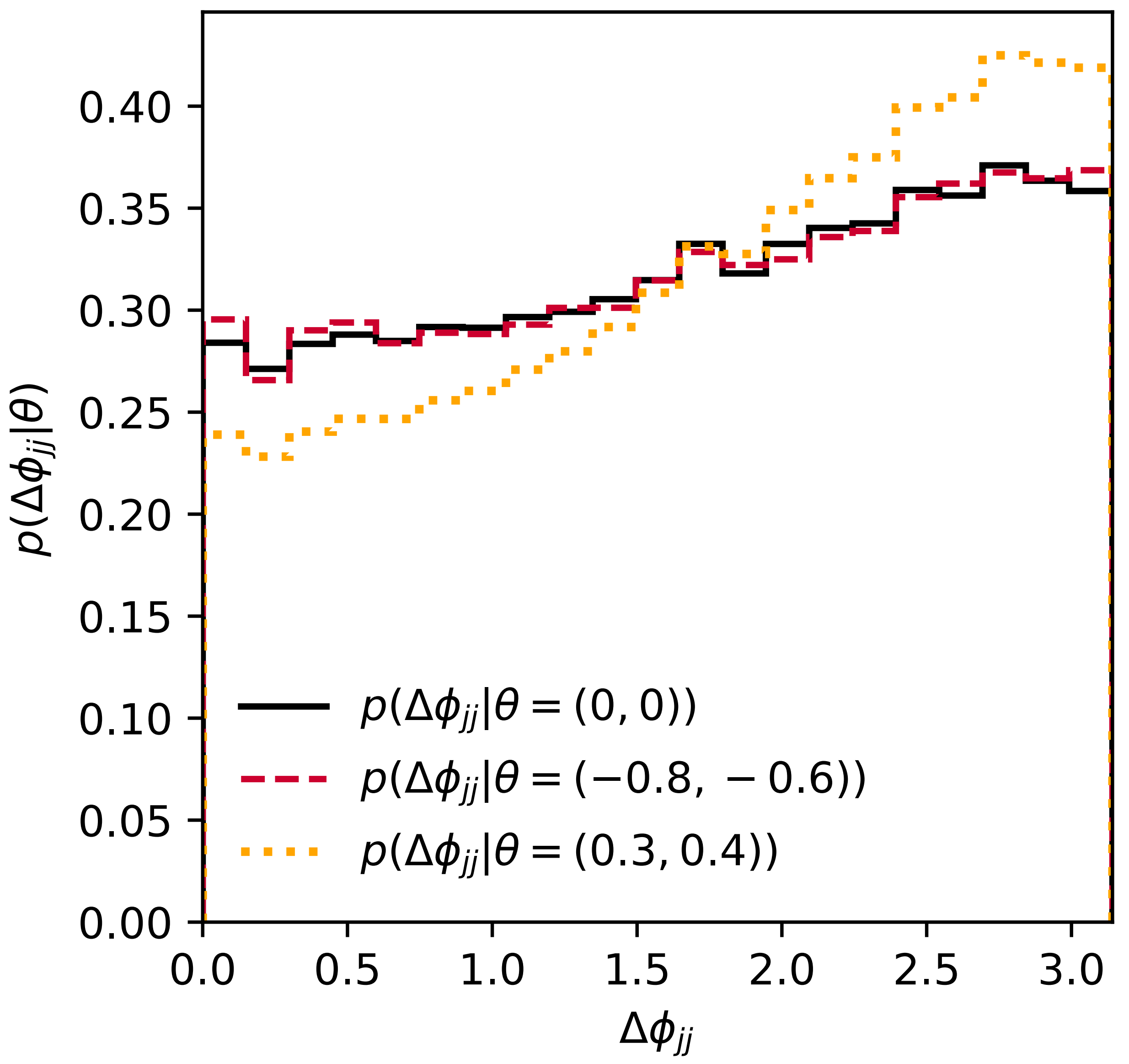
SALLY (Score approximates likelihood locally)



"The machine learning version of Optimal Observables":

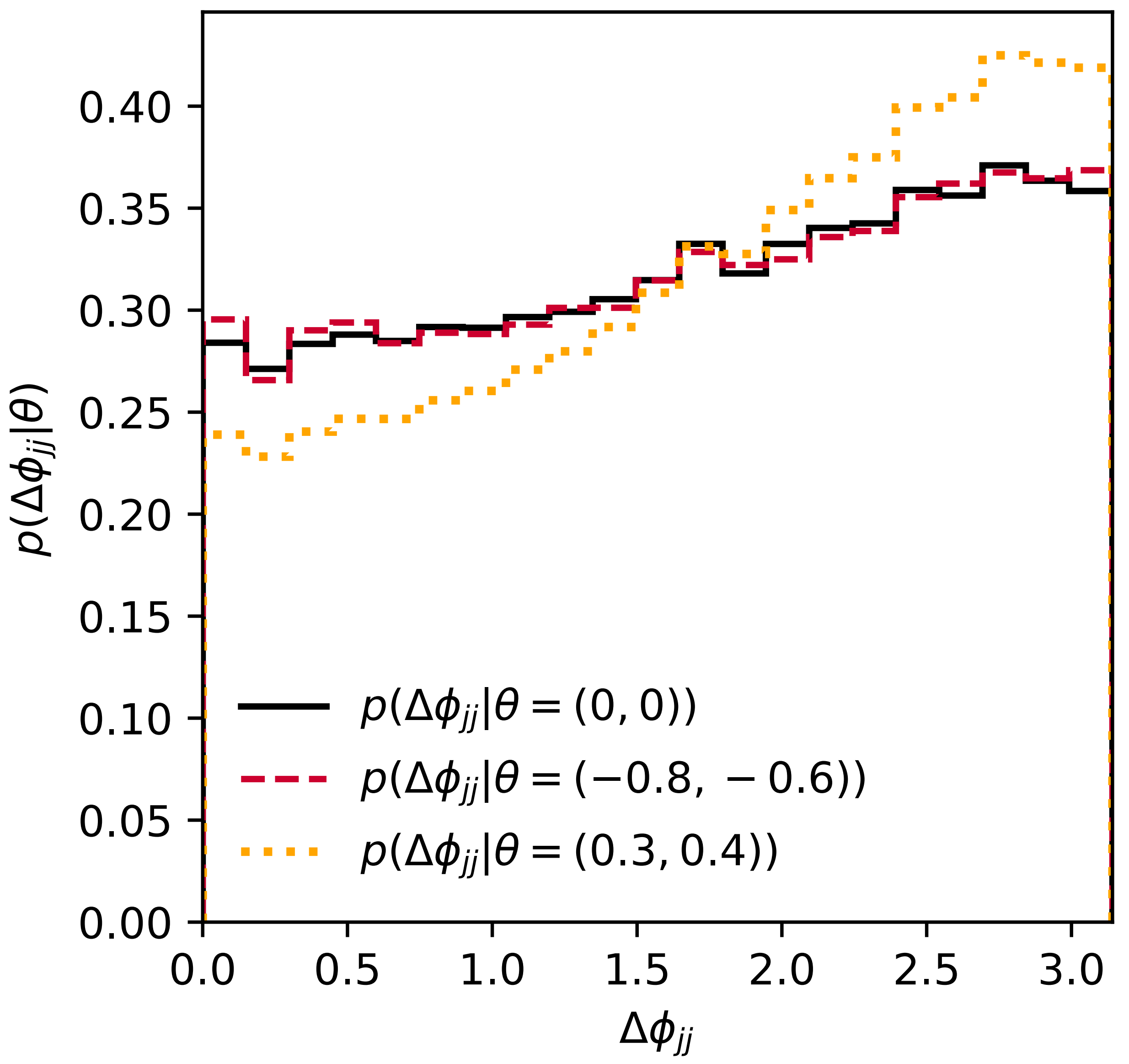
- Simpler & more robust than RASCAL
- Just as powerful close to θ_{ref}, but can lead to suboptimal limits further away

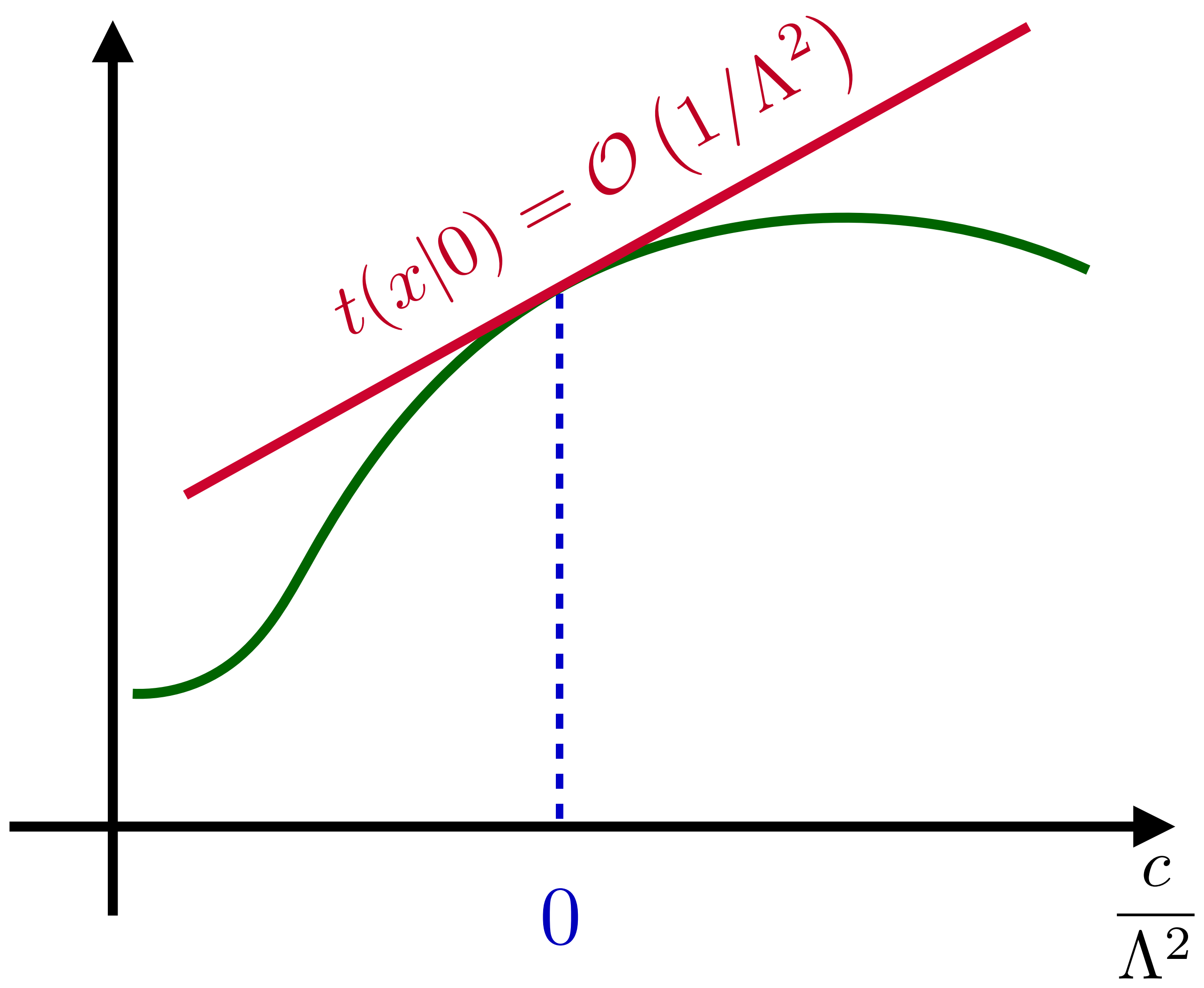
Perfect match for EFT measurements



- Good for subtle kinematic effects
  (Subtle point: Large overlap of kinematic distributions reduces variance of joint likelihood ratio / joint score)

Perfect match for EFT measurements

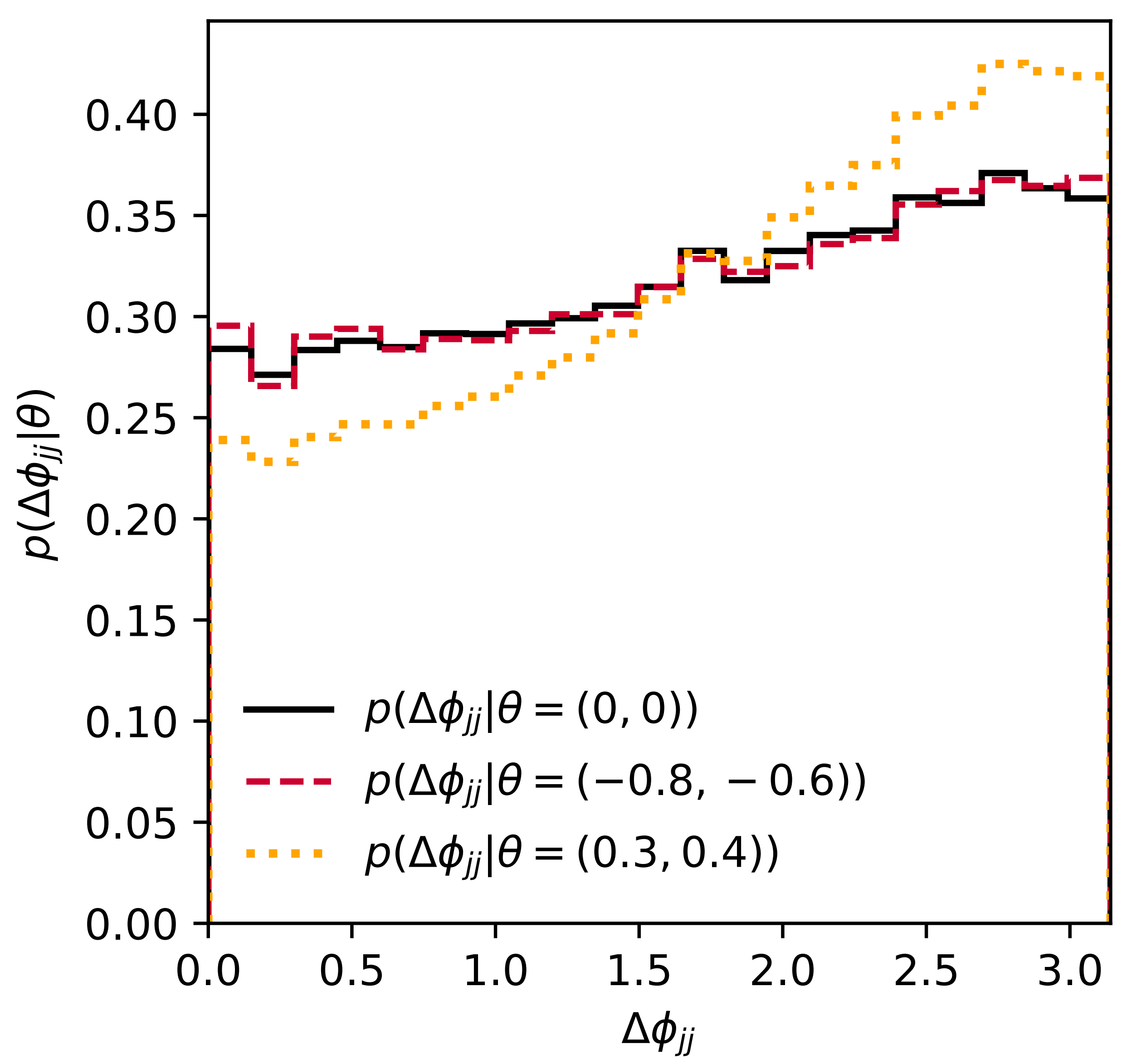


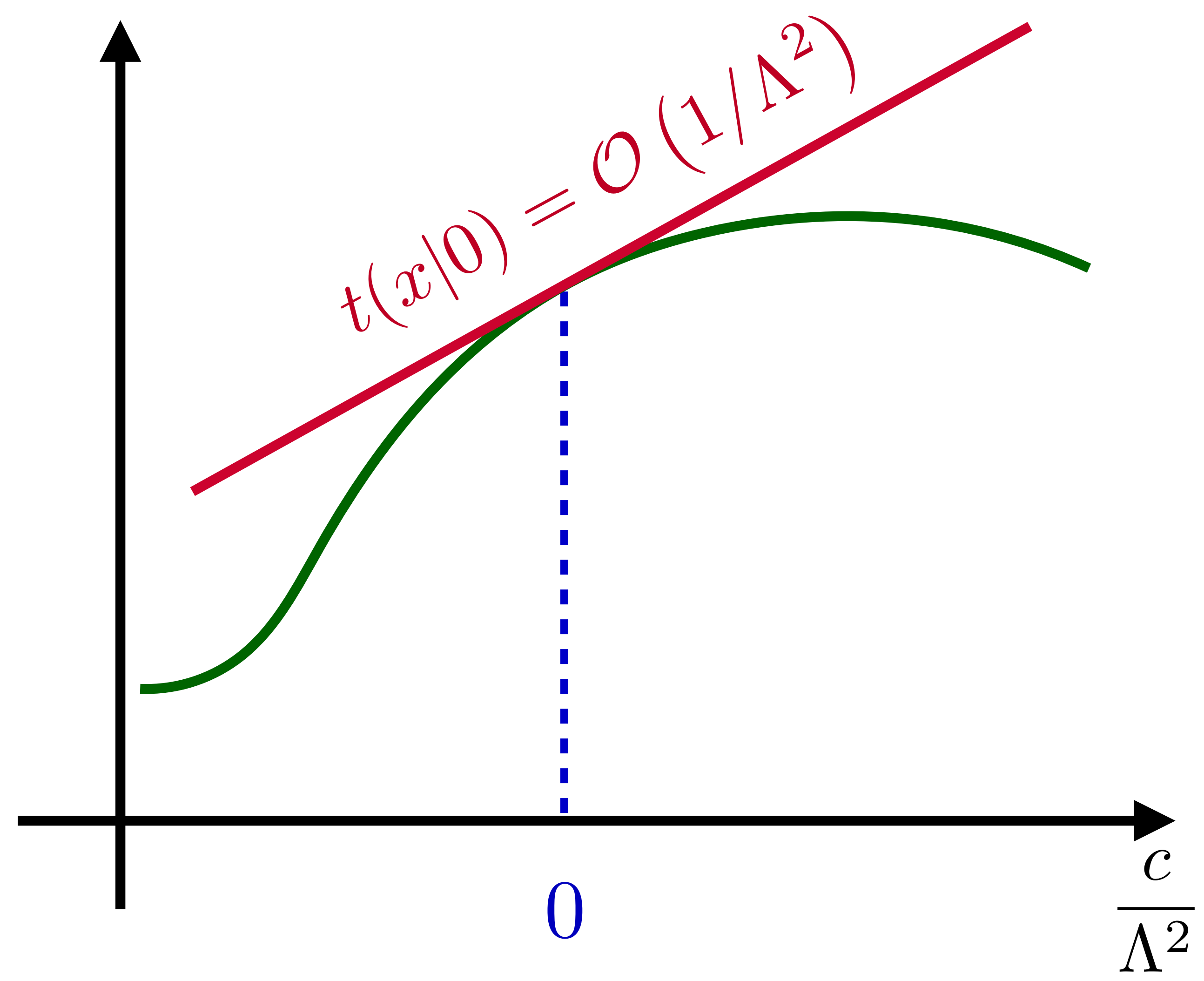


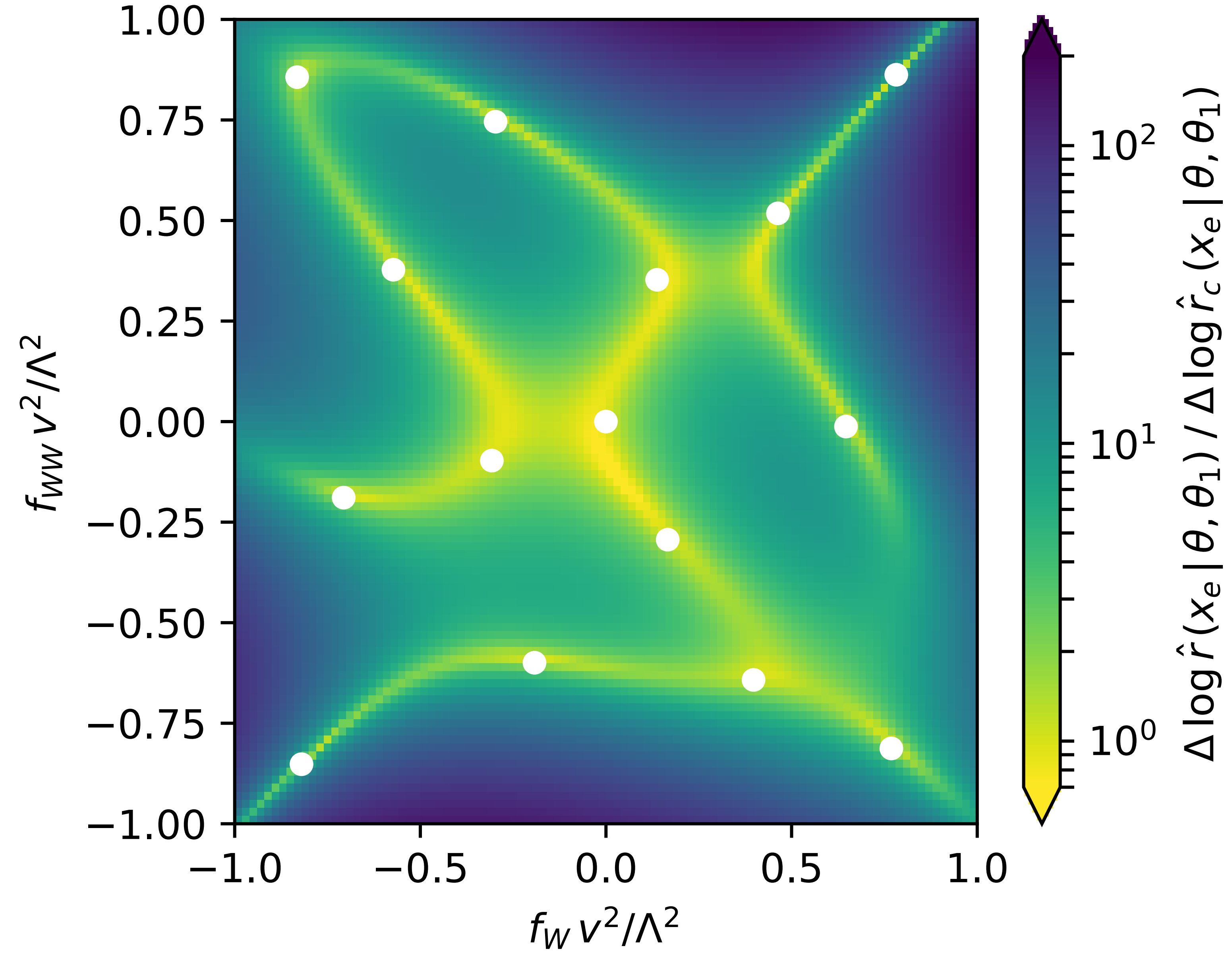
- Good for subtle kinematic effects
  (Subtle point: Large overlap of kinematic distributions reduces variance of joint likelihood ratio / joint score)

- Interference effects can be isolated using SALLY at the SM
  (SMALLY?)

Perfect match for EFT measurements







- Good for subtle kinematic effects
  (Subtle point: Large overlap of kinematic distributions reduces variance of joint likelihood ratio / joint score)
- Interference effects can be isolated using SALLY at the SM
  (SMALLY?)
- Morphing techniques allow fast reweighting to any parameter points
  [e.g. ATL-PHYS-PUB-2015-047]

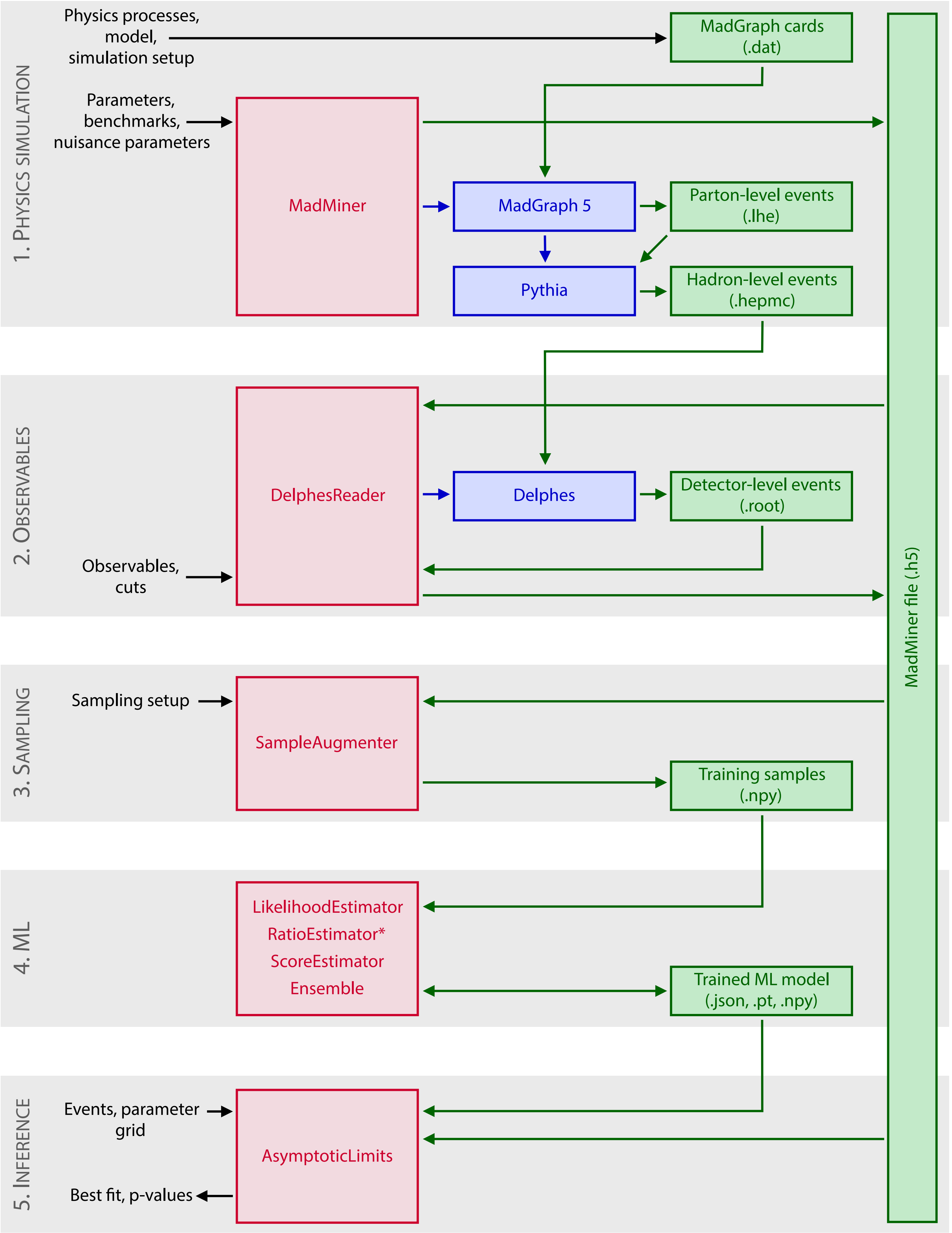
In practice

Automization

[JB, F. Kling, I. Espejo, K. Cranmer doi: 10.5281/zenodo.1489147]

We are developing MadMiner, which makes it straightforward to apply the new techniques to LHC problems

- Out of the box: Pheno-level analyses
  - MadGraph, Pythia, Delphes
  - Systematic uncertainties from PDF / scale variation
- Scalable to state-of-the-art experimental tools
  - Mostly requires bookkeeping of fully differential cross sections
- Modular interface
  - Extensive documentation
  - Embedded into Python / ML ecosystem

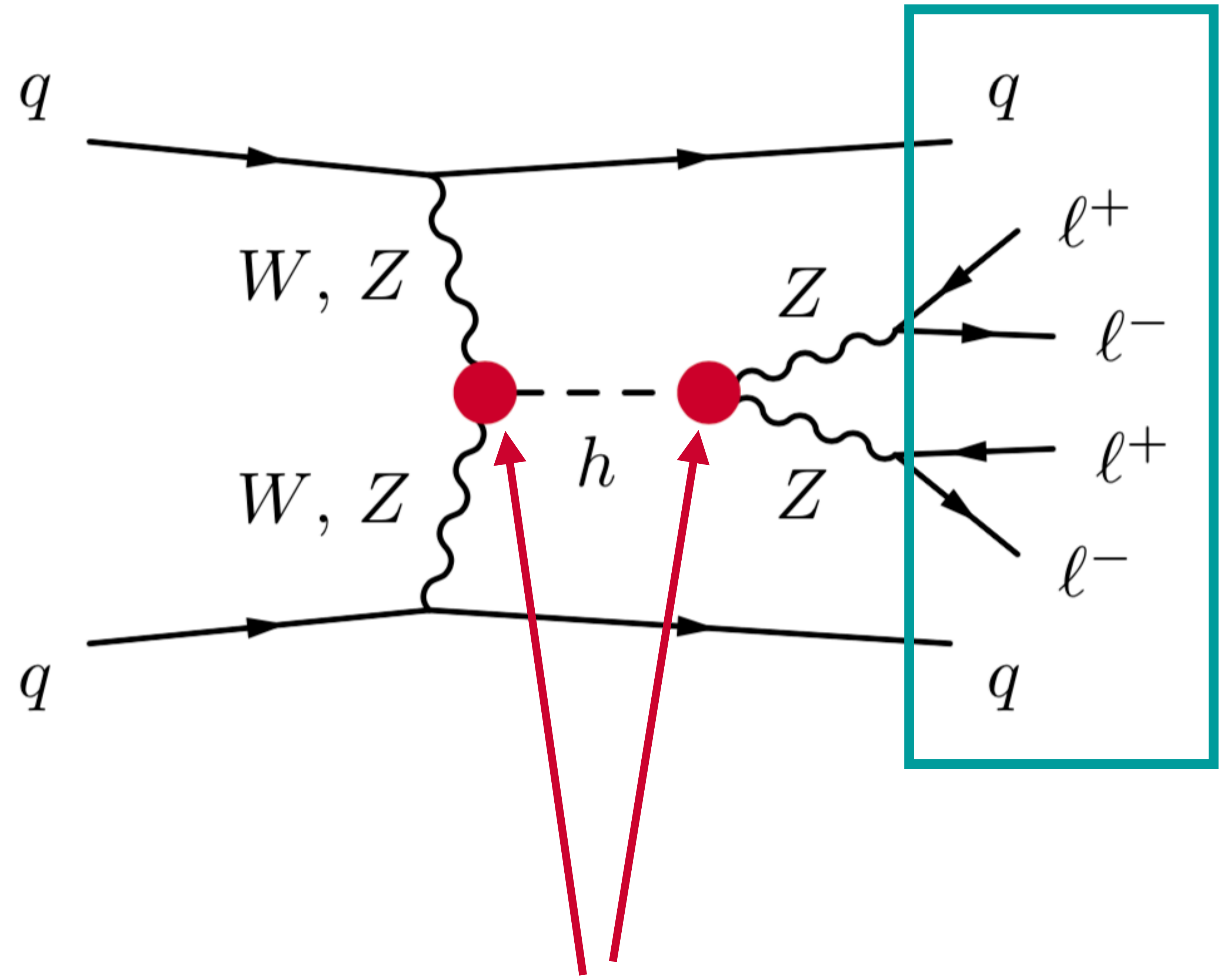


github.com/johannbrehmer/madminer madminer.readthedocs.io `pip install madminer`

Proof of concept: Higgs production in weak boson fusion

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

at least 16-dimensional
observable space



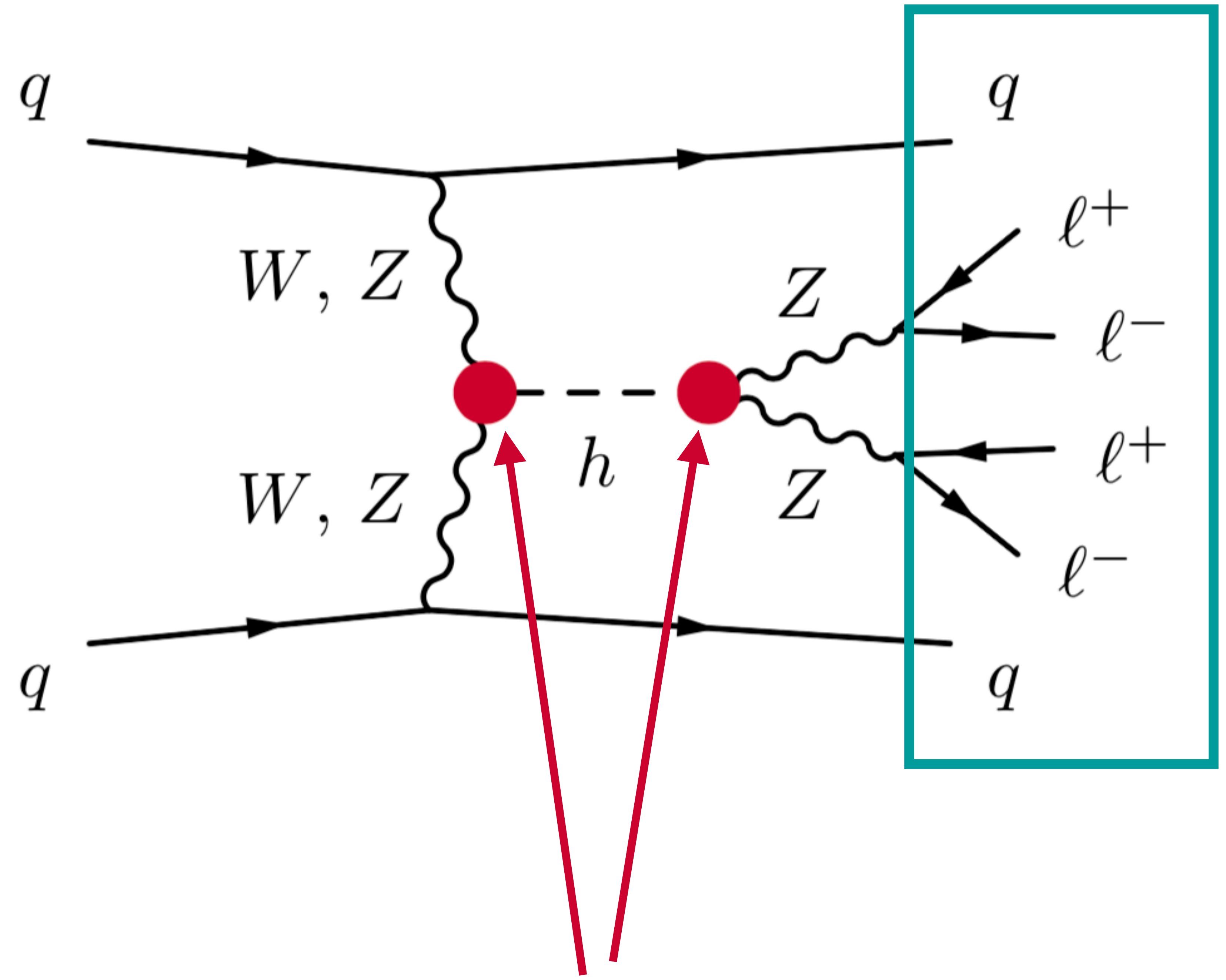
Exciting new physics might hide here!
We parameterize it with two EFT coefficients:

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

Proof of concept: Higgs production in weak boson fusion

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

at least 16-dimensional
observable space



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

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

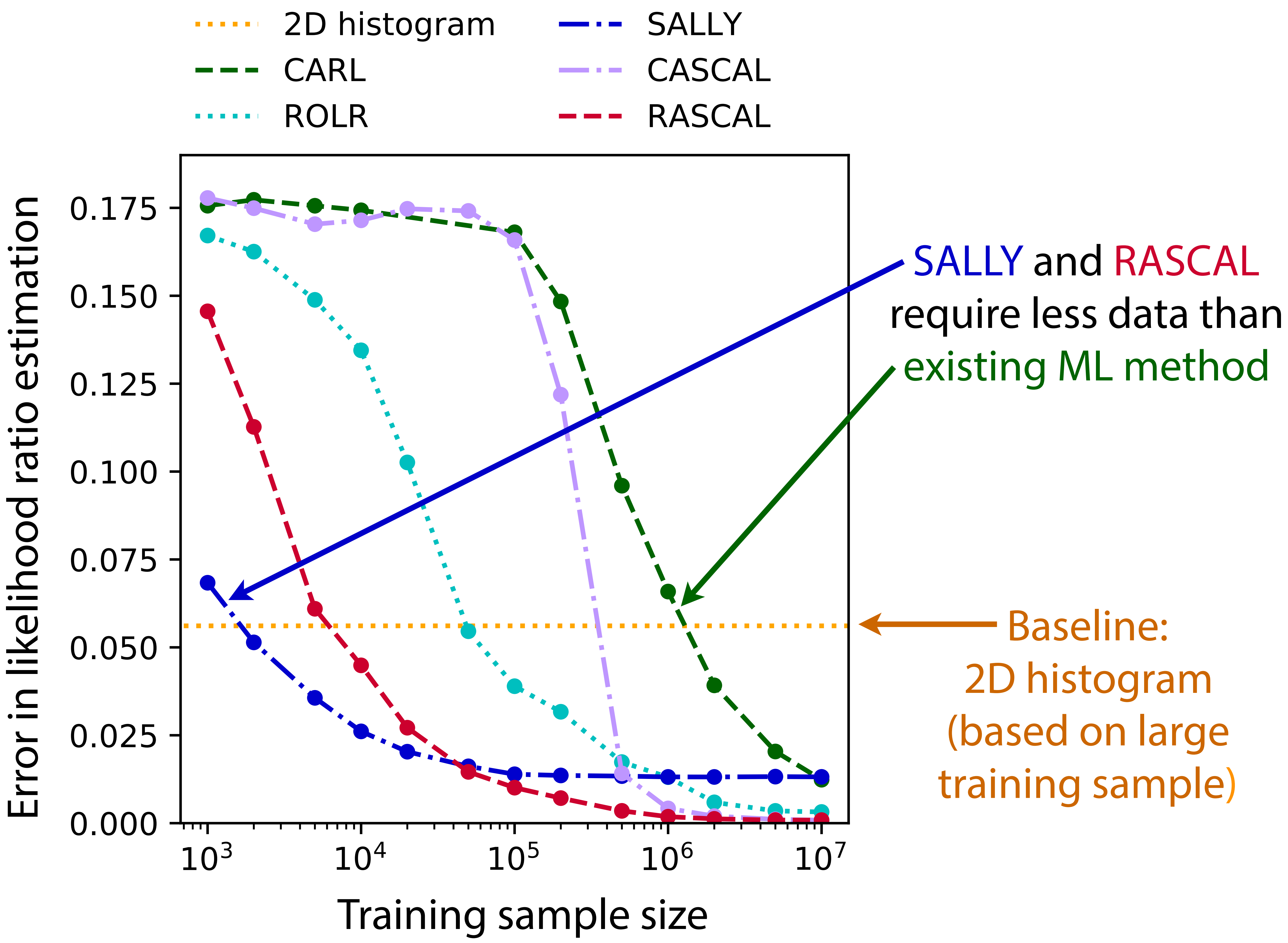
Goal: constrain the two EFT parameters

- new inference methods
- baseline: 2d histogram analysis of jet momenta & angular correlations

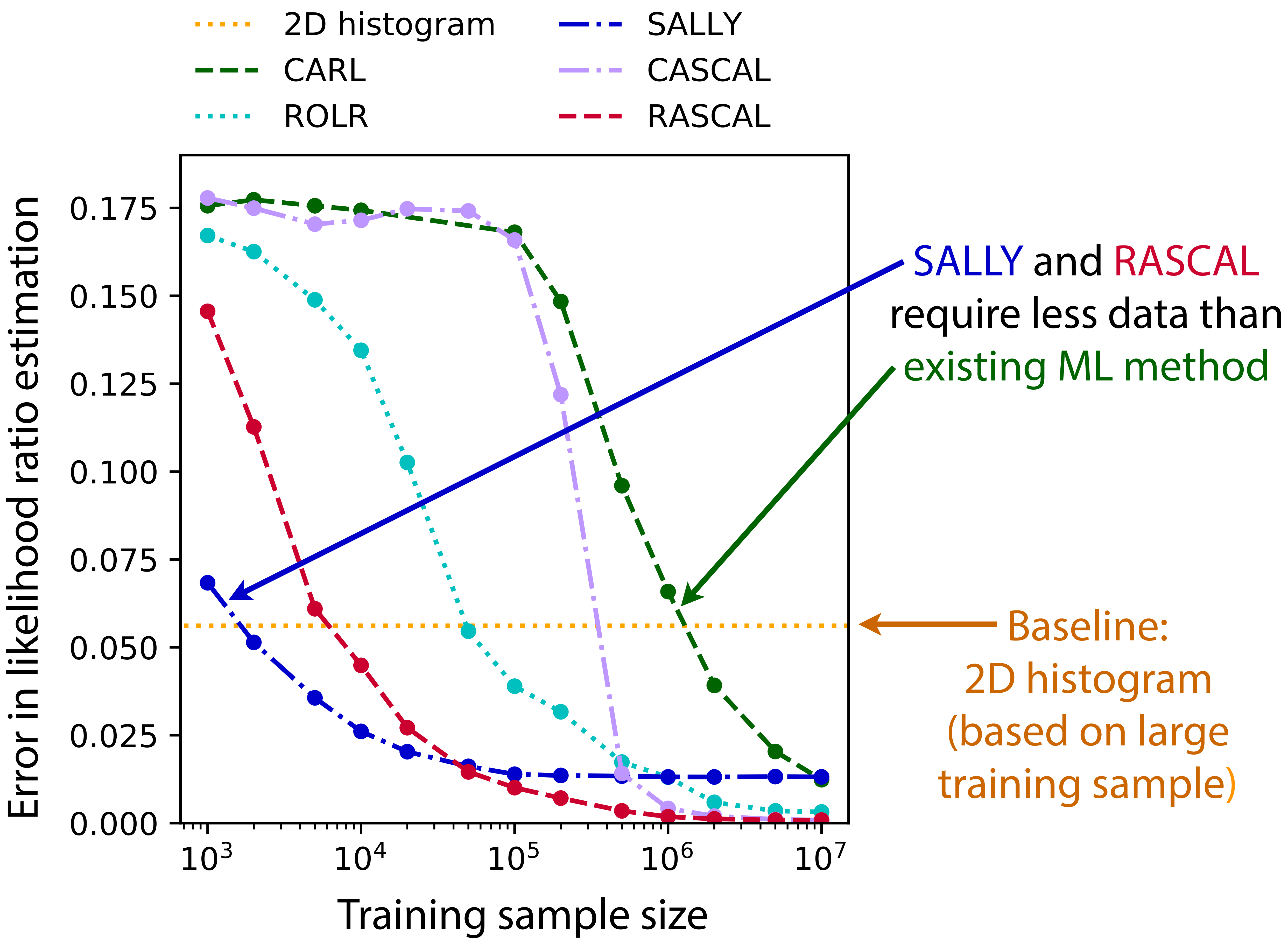
Two scenarios:

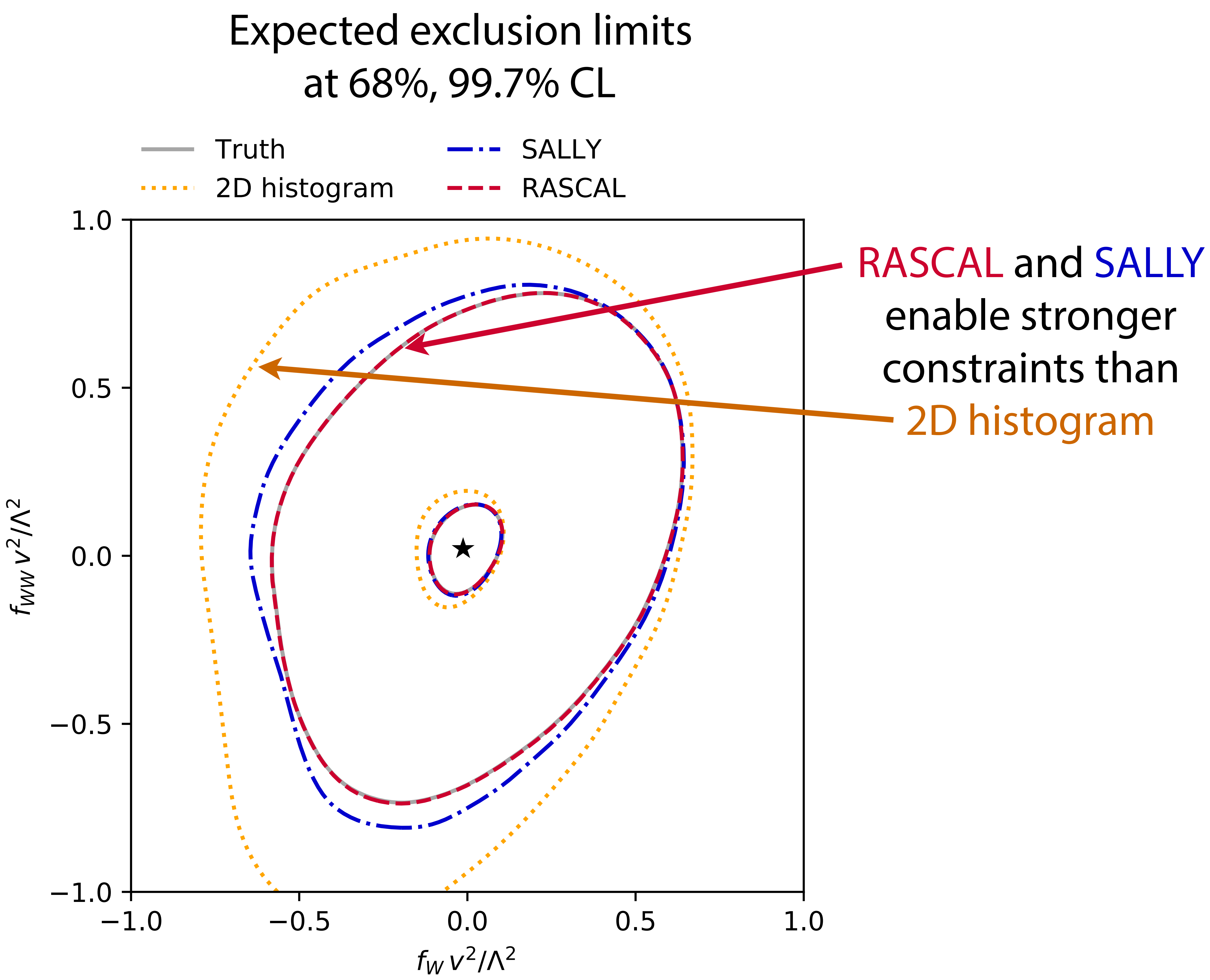
- Simplified setup in which we can compare to true likelihood
- "Realistic" simulation with approximate detector effects

Stronger constraints with less training data

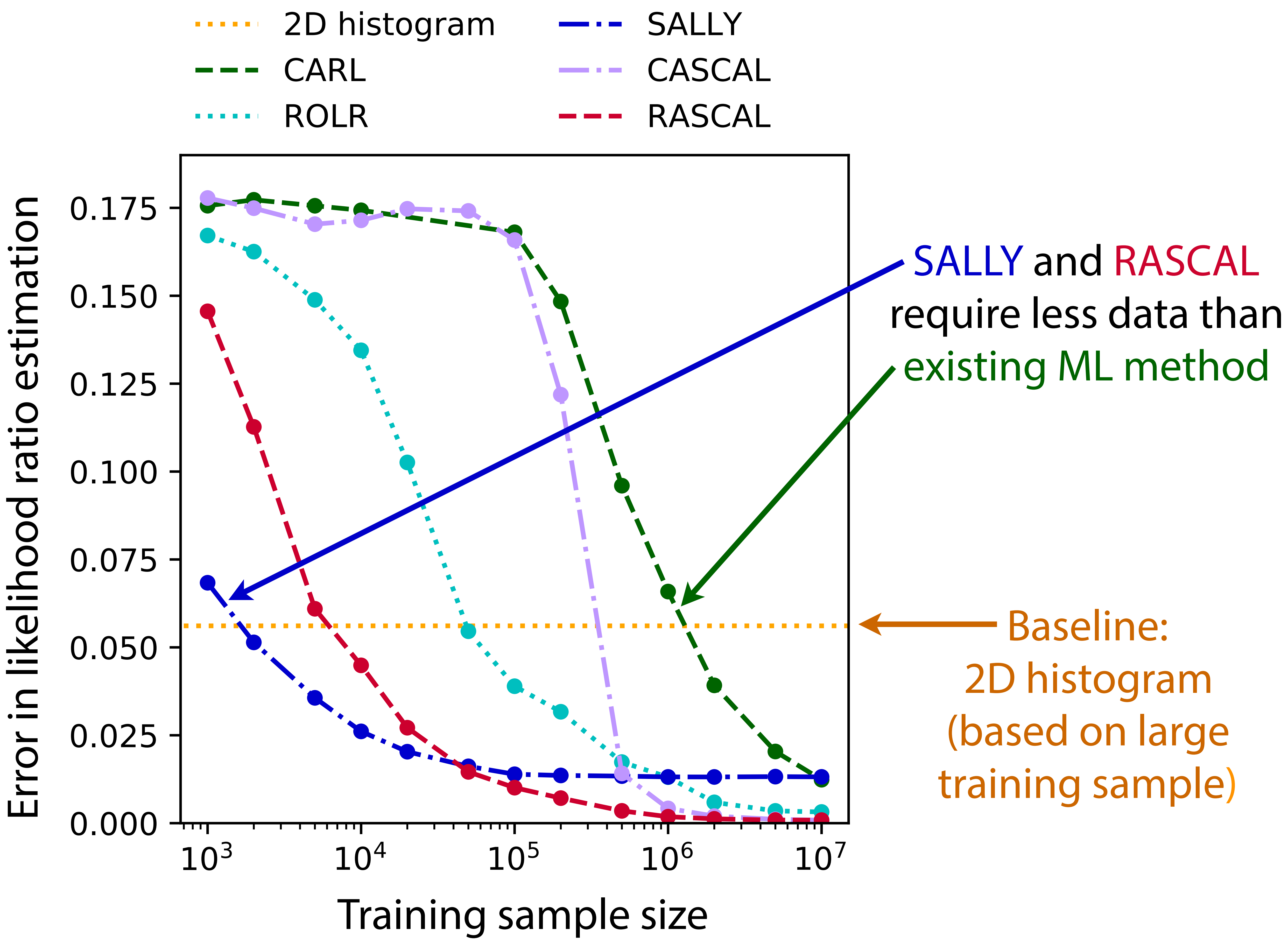


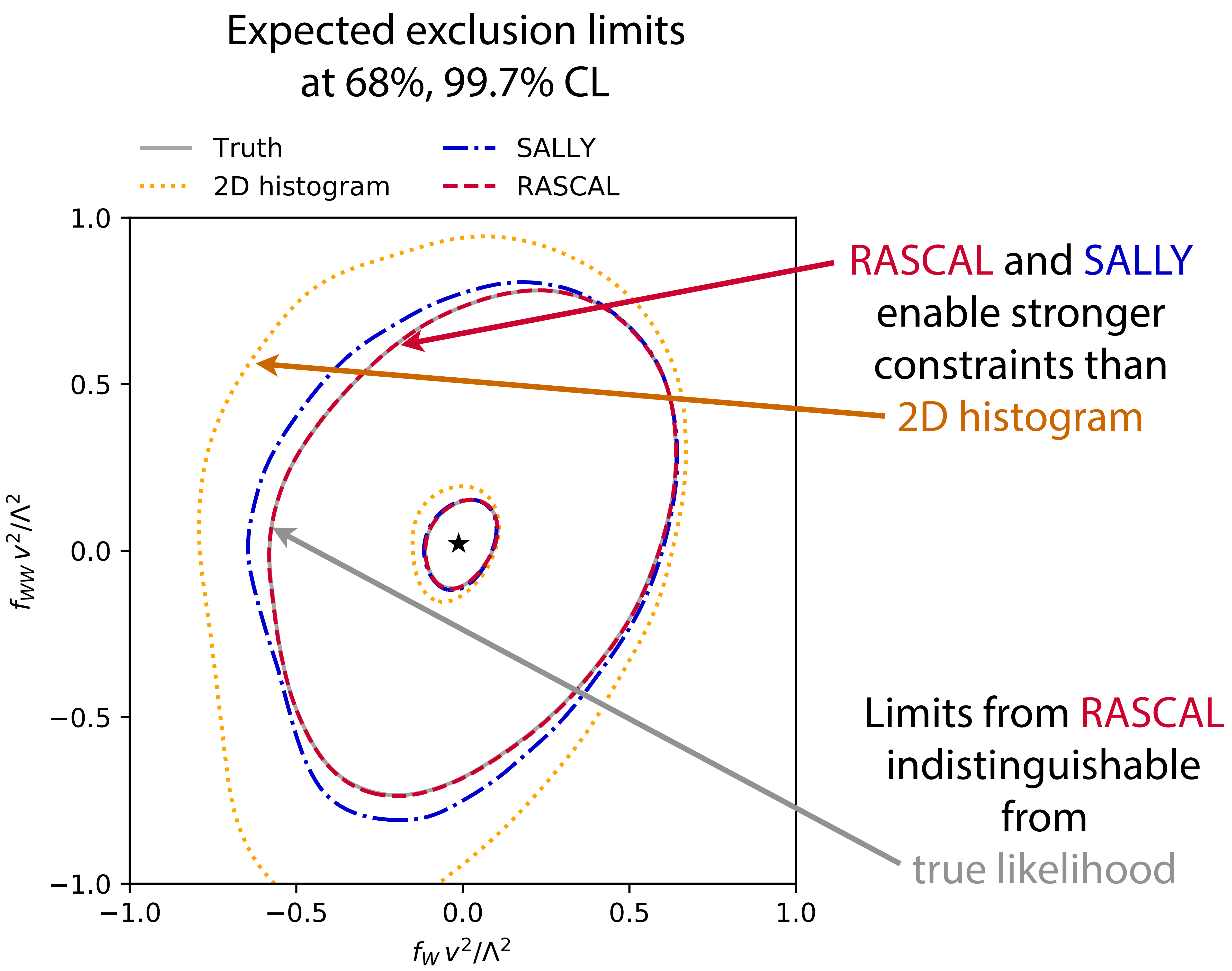
Stronger constraints with less training data





Stronger constraints with less training data



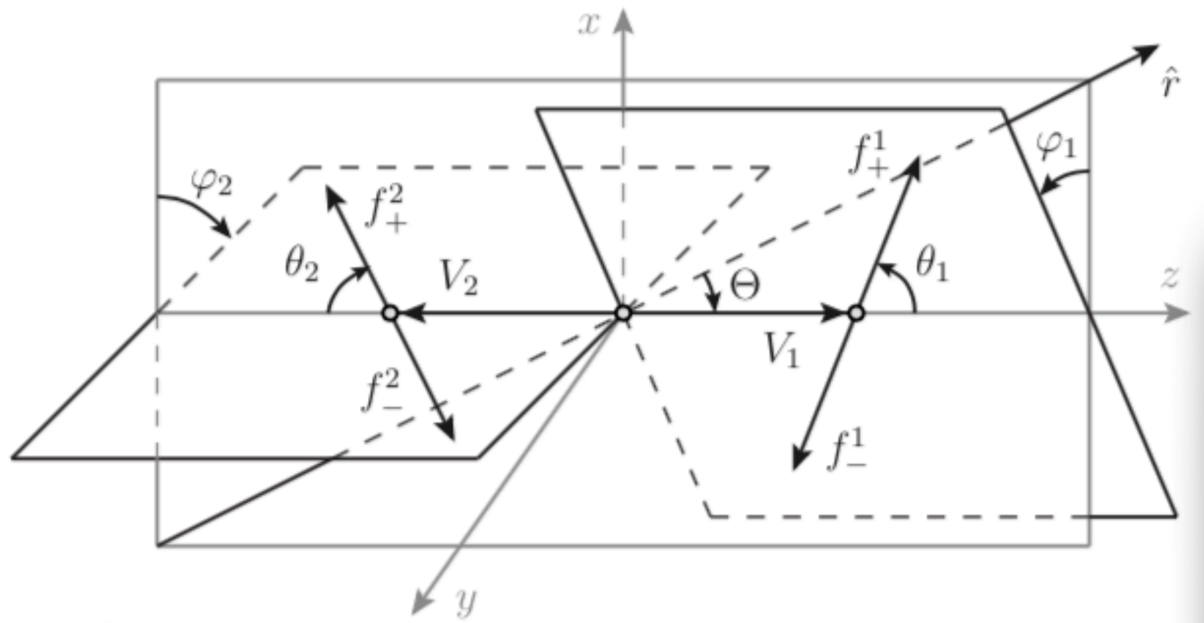


Dibosons

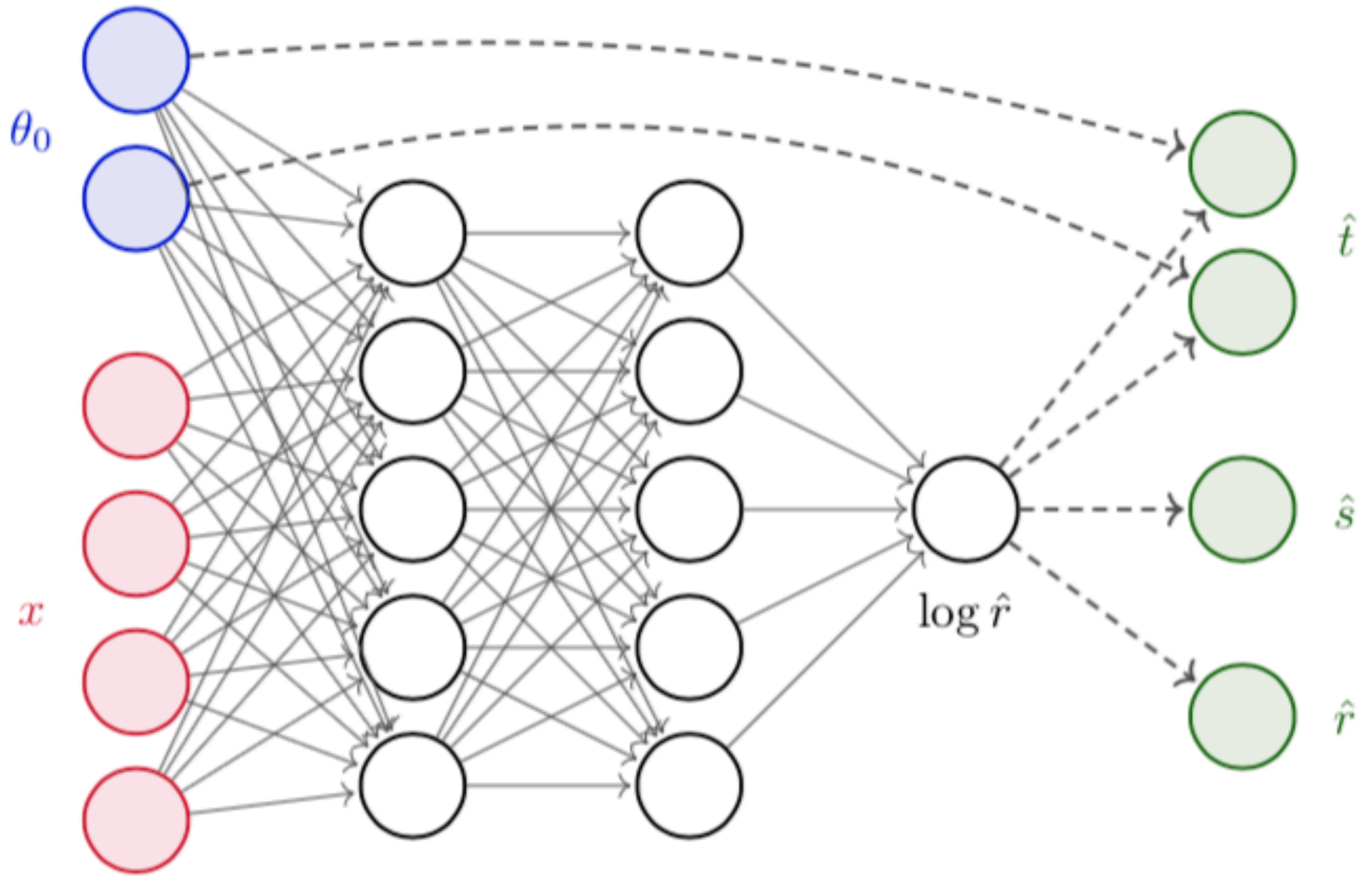
Transverse DiBosons

[Panico, Riva, AW, 2017]

Measuring diboson diff. cross-sections is not enough



Can we get this done by a Machine? (in real world)



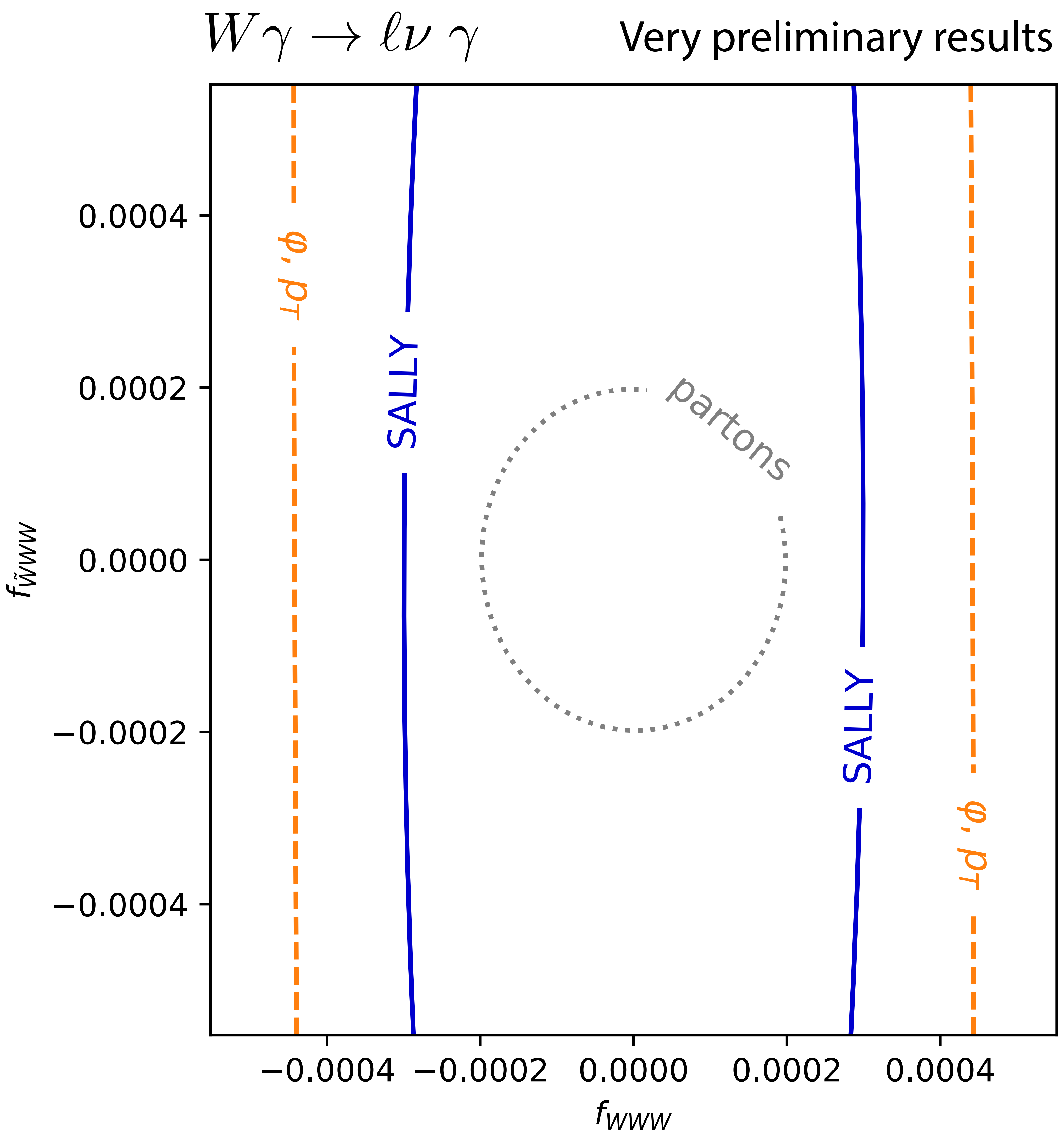
Cranmer et.al., 2018

[from A. Wulzer's talk]

Dibosons

[JB, K. Cranmer, M. Farina, F. Kling, D. Pappadopulo, J. Ruderman in progress]

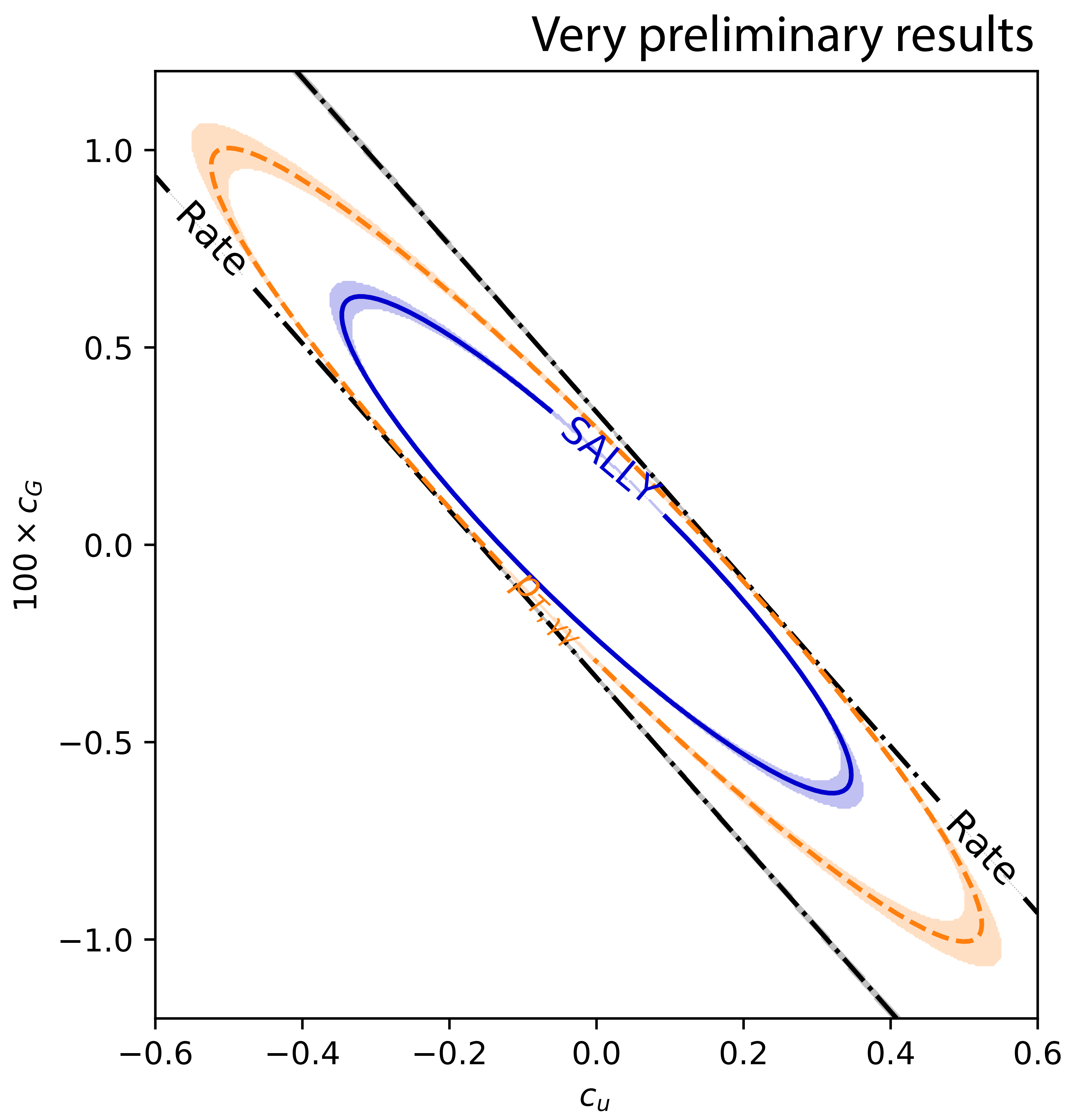
Yes.



Higgs measurements

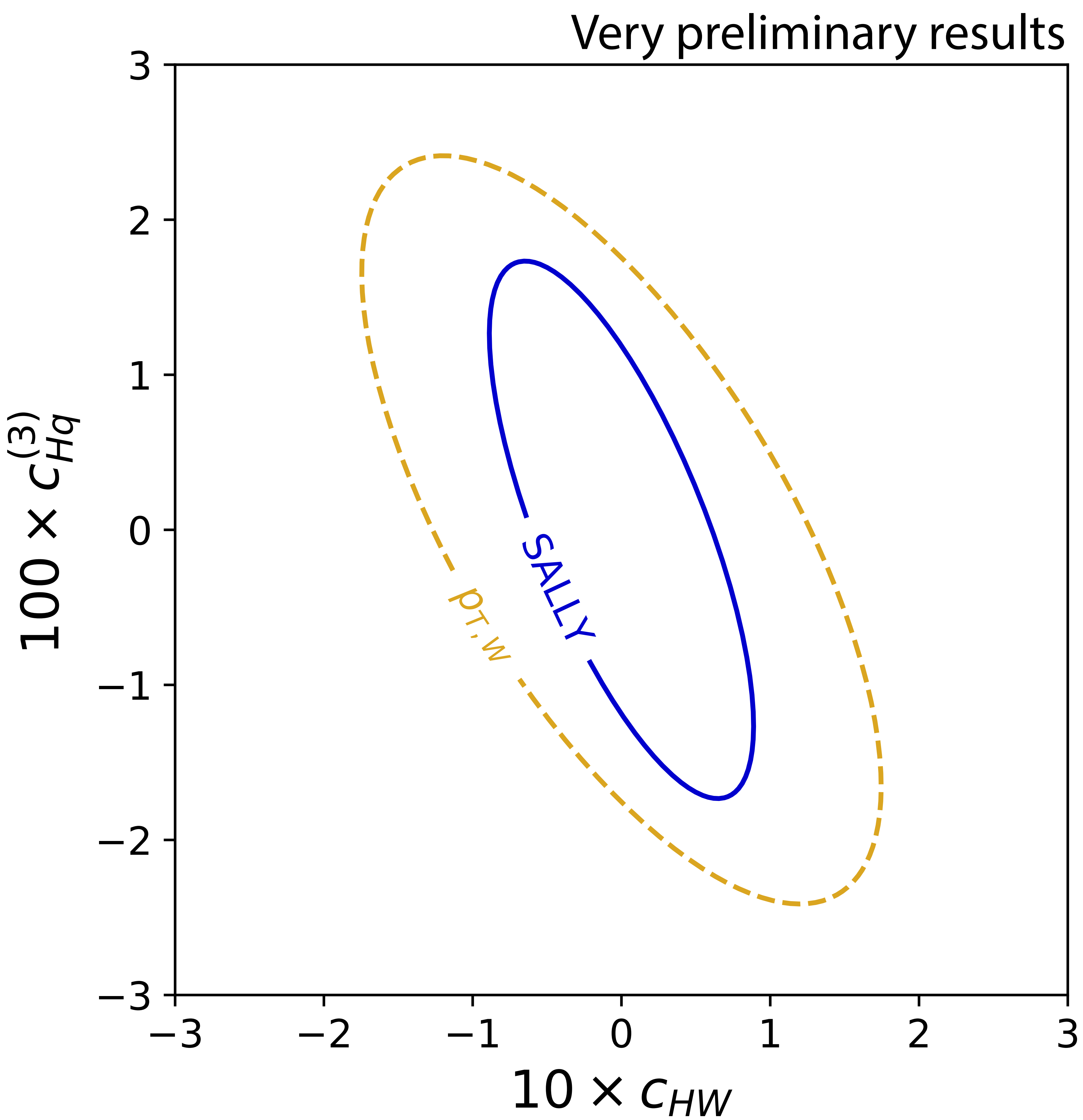
$$t\bar{t}\,h \to (b\ell^+\nu)\,(b\ell^-\bar{\nu})\,(\gamma\gamma)$$

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

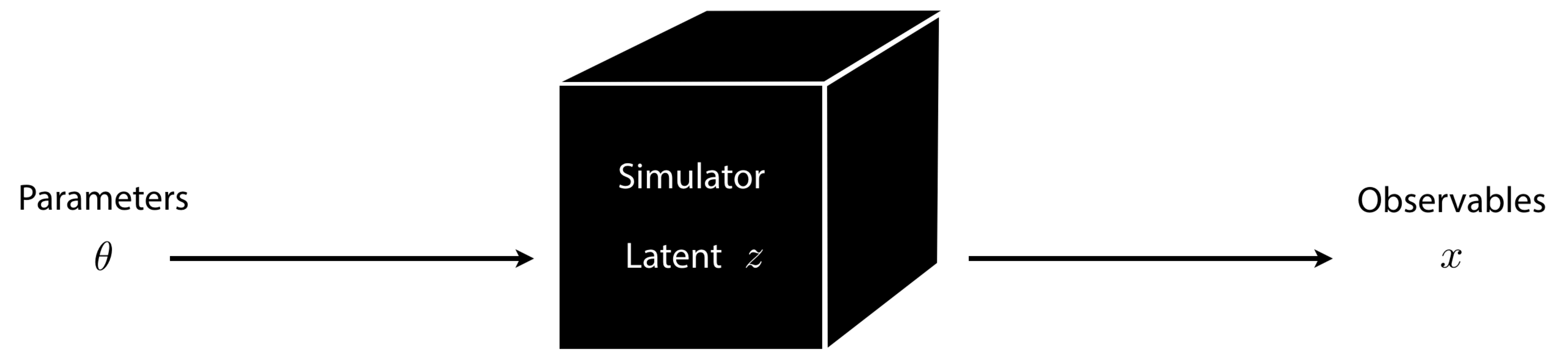


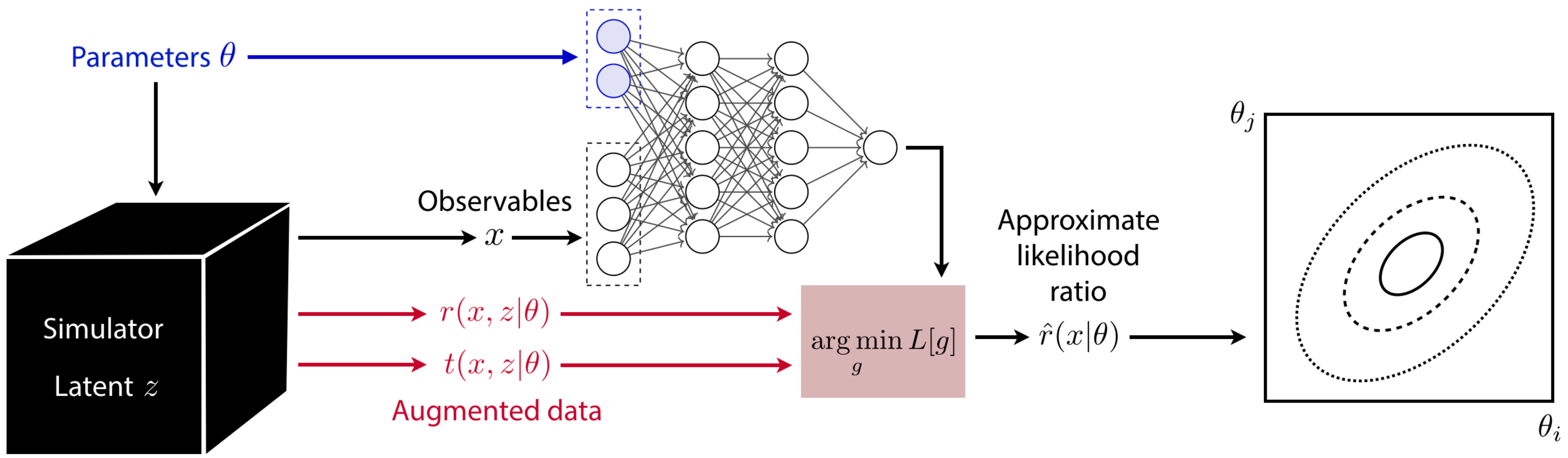
$$Wh \to (\ell\nu)(b\bar{b})$$

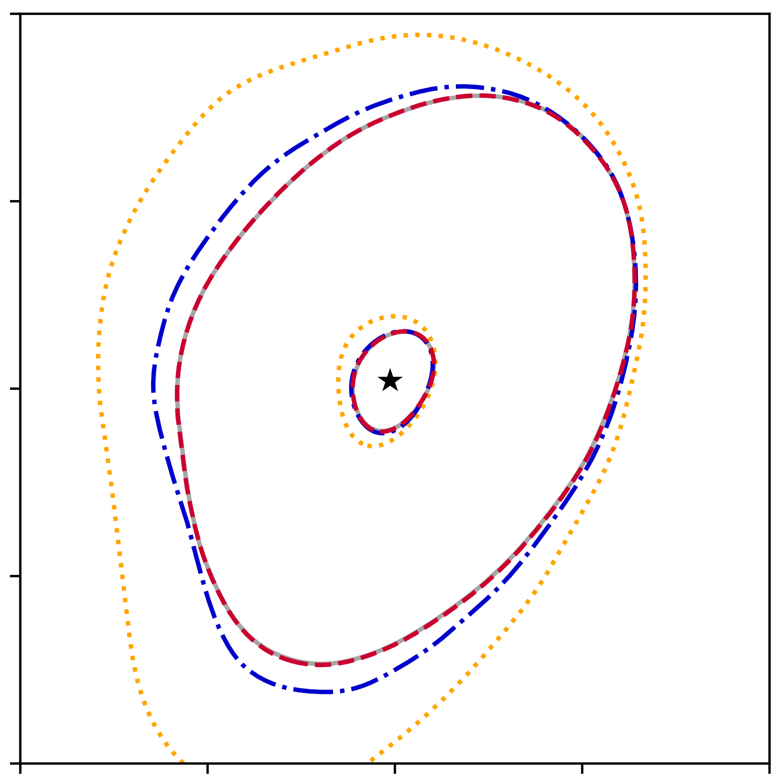
[JB, S. Dawson, S. Homiller, F. Kling, T. Plehn in progress]



A new approach to LHC measurements







- LHC measurements with high-dimensional observations have "likelihood-free" structure
- New multivariate inference techniques: Leverage information in matrix elements + power of machine learning to…
  - estimate the full likelihood function
  - learn optimal summary statistics
- First tests show potential to substantially increase sensitivity to new physics
  - Ideal for EFT measurements

References


Kyle Cranmer


Gilles Louppe


Juan Pavez


Markus Stoye


Felix Kling


Irina Espejo

JB, KC, GL, JP:	Constraining Effective Field Theories with Machine Learning	[PRL, 1805.00013]
JB, KC, GL, JP:	A Guide to Constraining Effective Field Theories with Machine Learning	[PRD, 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, FK, IE, KC:	MadMiner: An inference toolkit for particle physics	[doi: 10.5281/zenodo.1489147]

Bonus material

Solve it by approximating the integral

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

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

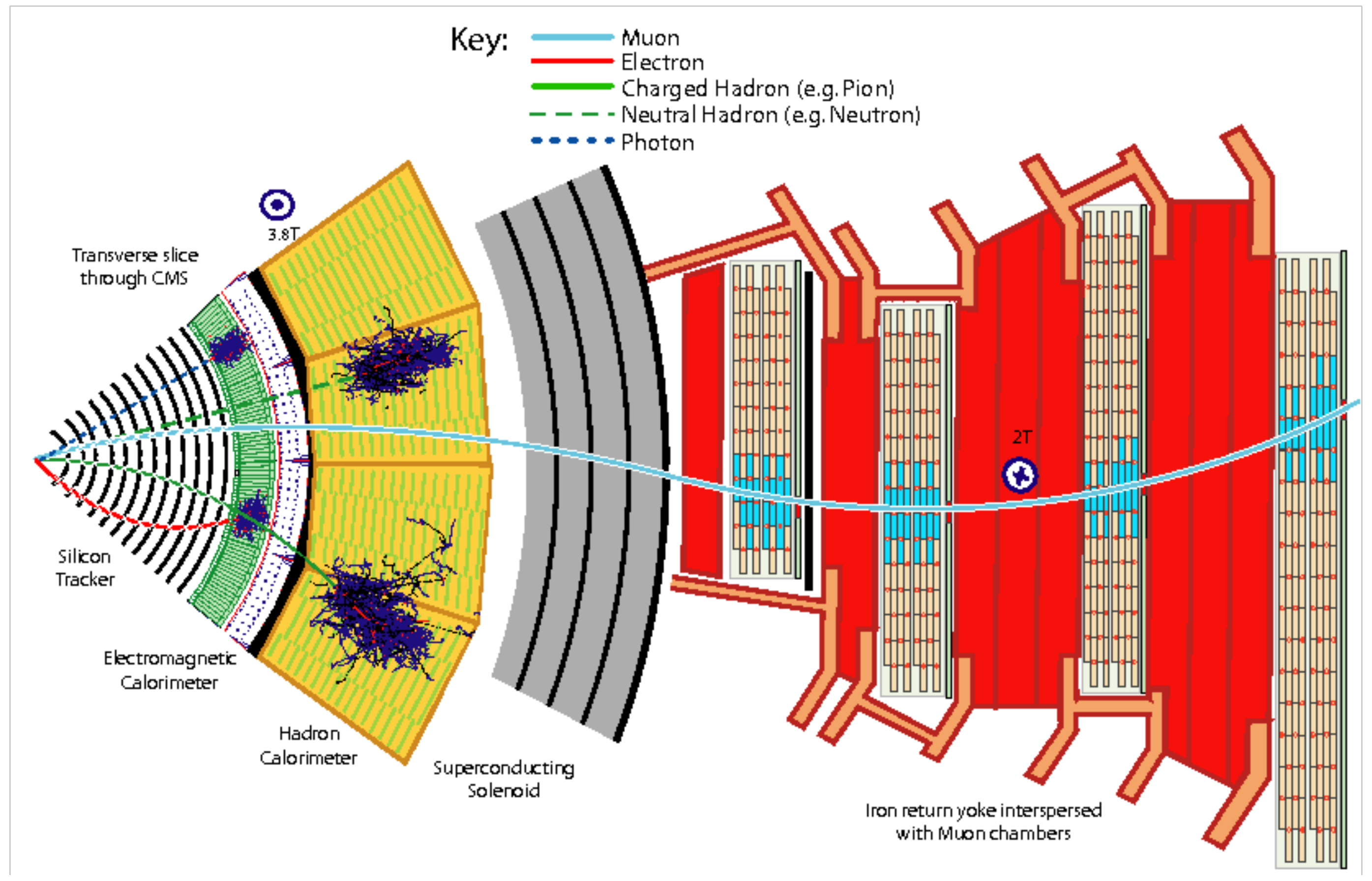
- Matrix Element Method: [K. Kondo 1988]
  - approximate shower + detector effects into transfer function $\hat{p}(x|z_p)$
  - explicitly calculate remaining integral

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

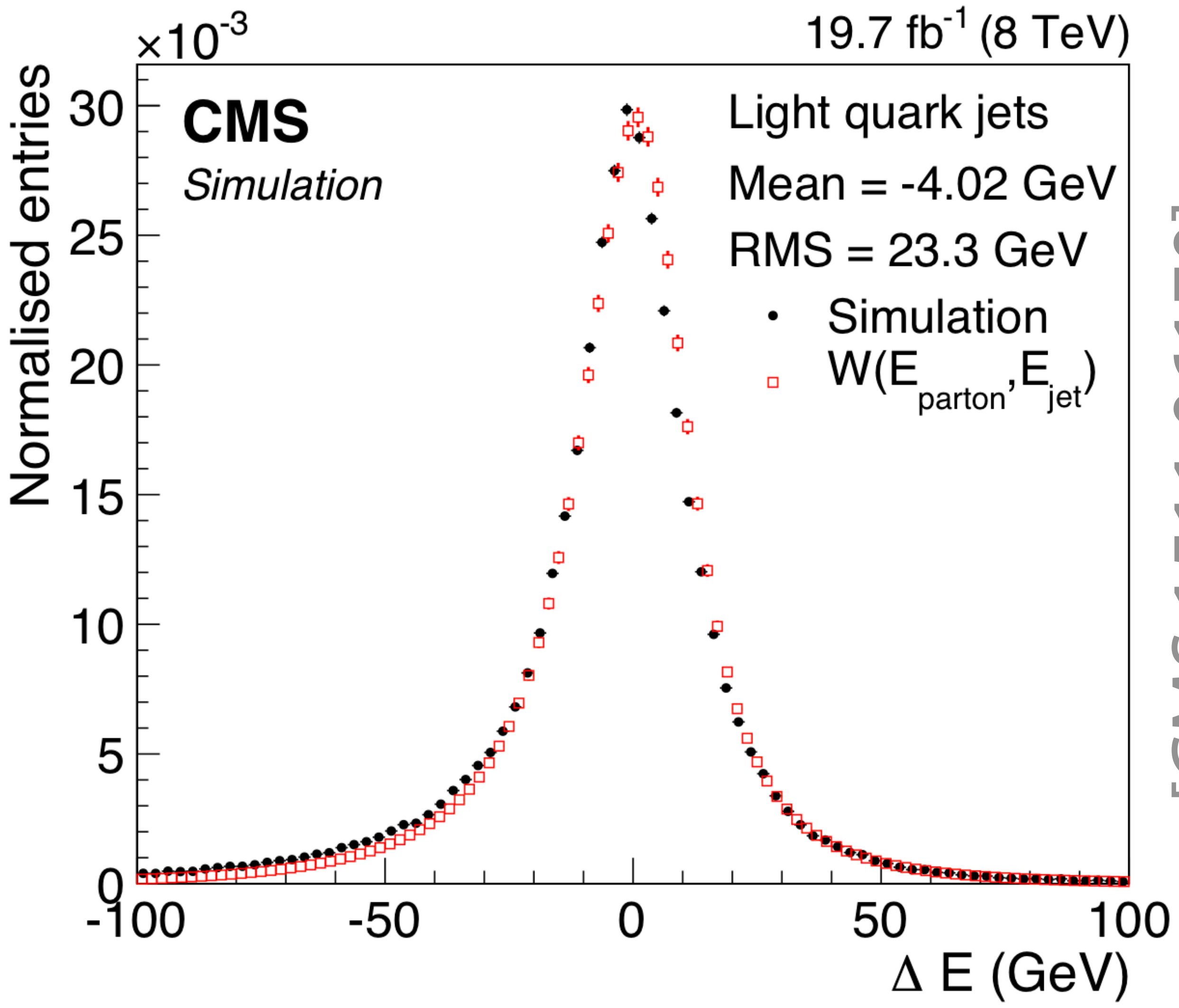
  - Shower / Event Deconstruction [D. E. Soper, M. Spannowsky 1102.3480, 1402.1189] extend explicit calculation to the shower

⇒ Uses matrix-element information, no summary statistics necessary, but:

- ad-hoc transfer functions (what about extra radiation?)
- evaluation still requires calculating an expensive integral



[CMS]



[CMS 1511.06170]

Solve it with machine learning classifiers

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

Solve it with machine learning classifiers

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

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

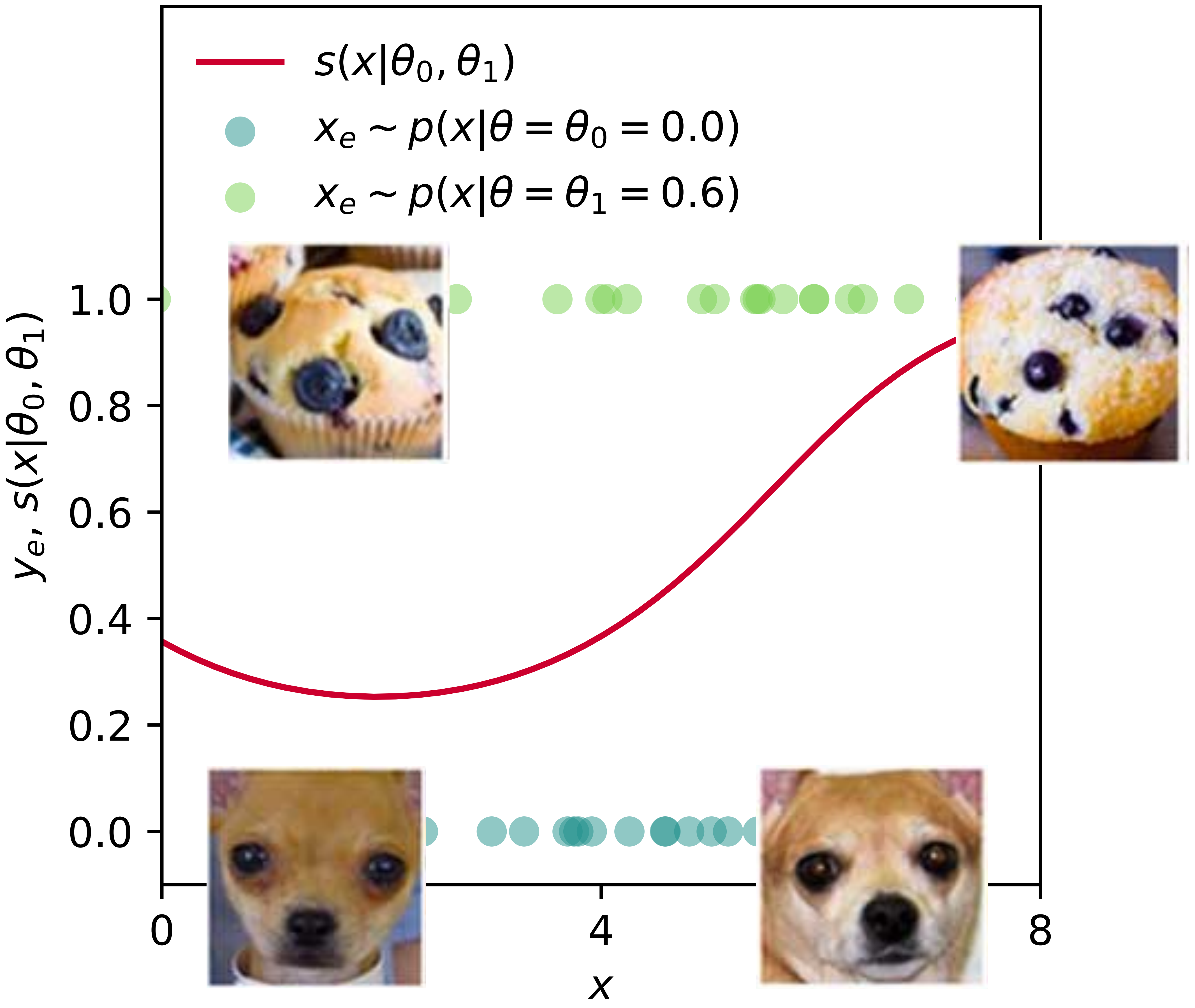
Solve it with machine learning classifiers

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

- Train neural network (BDT, …) to tell $x \sim p(x|\theta_0)$ from $x \sim p(x|\theta_1)$
- Classifier output $\hat{s}(x)$ is closer to 0 for θ_0-like events (closer to 1 for θ_1-like events)
- CARL: Transform classifier output function $\hat{s}(x)$ into estimator for the likelihood ratio

$$r(x) \equiv p(x|\theta_0)/p(x|\theta_1)$$

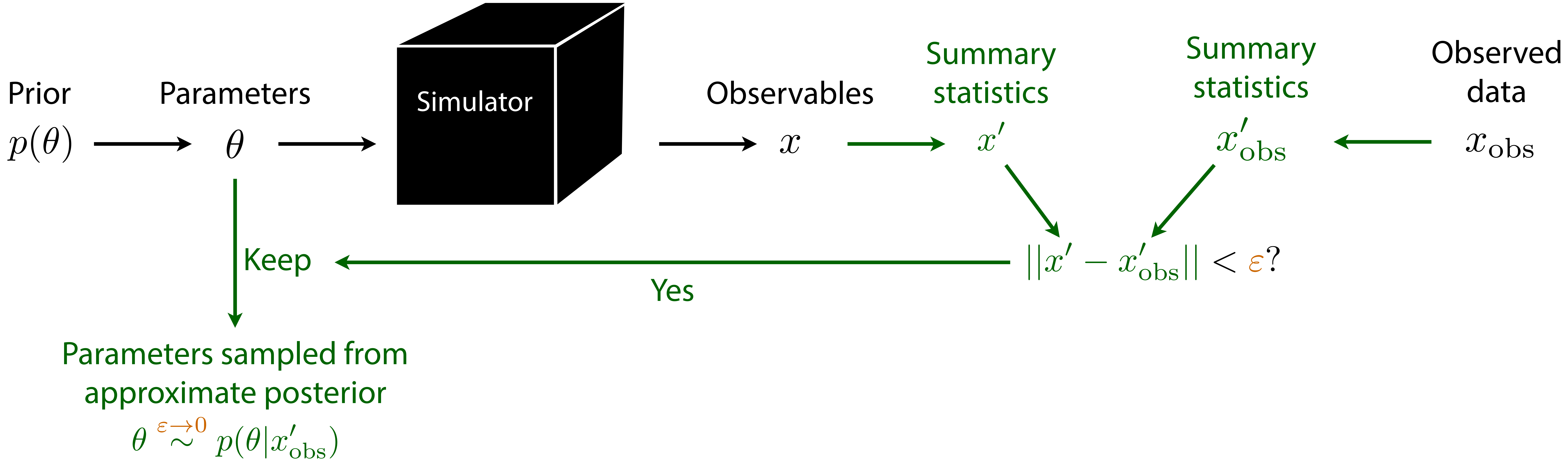
(calibrate estimator with histograms of NN output)



⇒ No summary statistics necessary, very fast evaluation… but may require large training samples

Approximate Bayesian Computation (ABC)

[D. Rubin 1984]



- How to choose x'?
- How to choose ε?
- No tractable posterior
- Need to run new simulations for new data or new prior

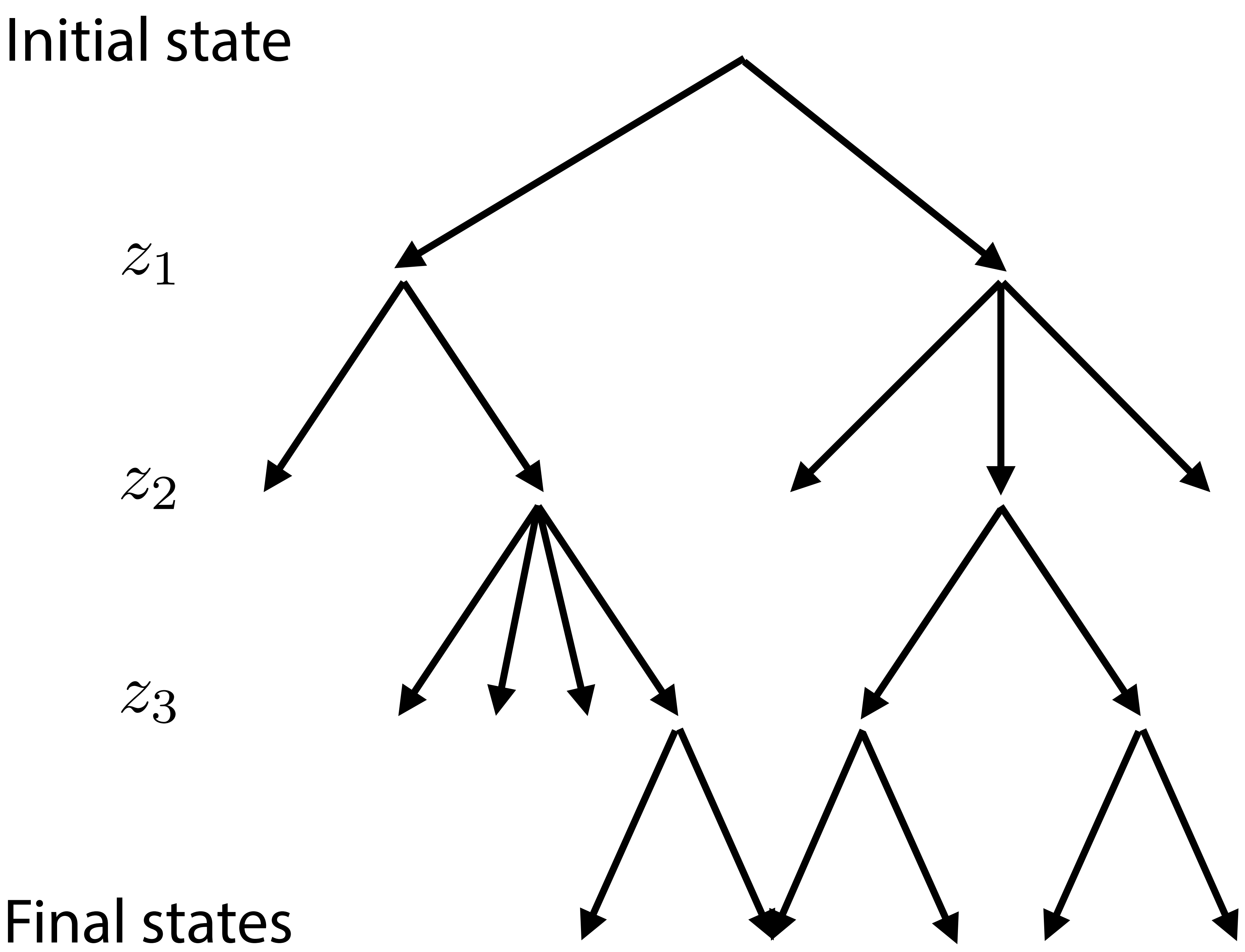
"Curse of dimensionality"

Precision vs efficiency tradeoff

Extracting the joint likelihood ratio from any simulation

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

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



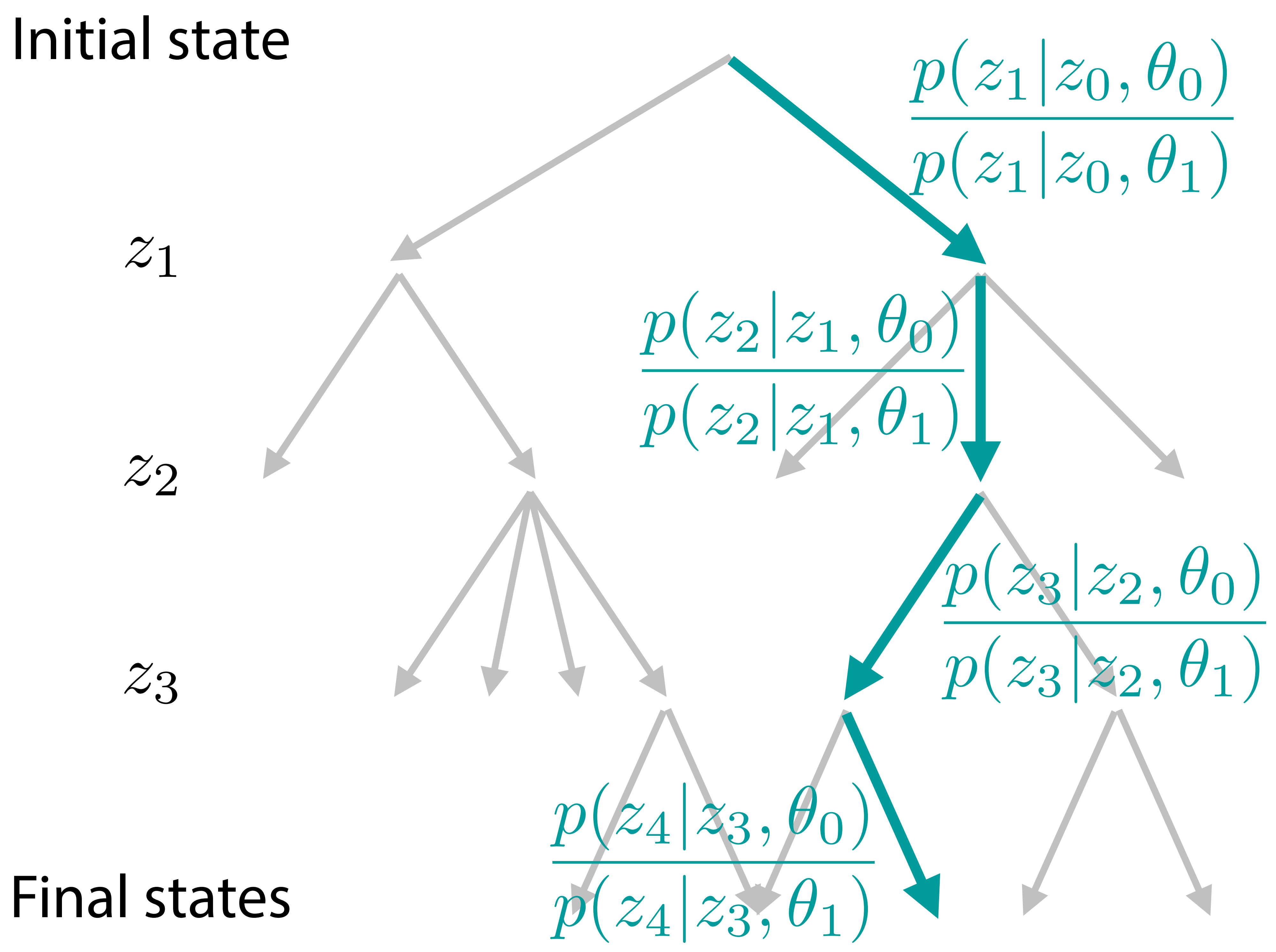
Extracting the joint likelihood ratio from any simulation

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

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

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

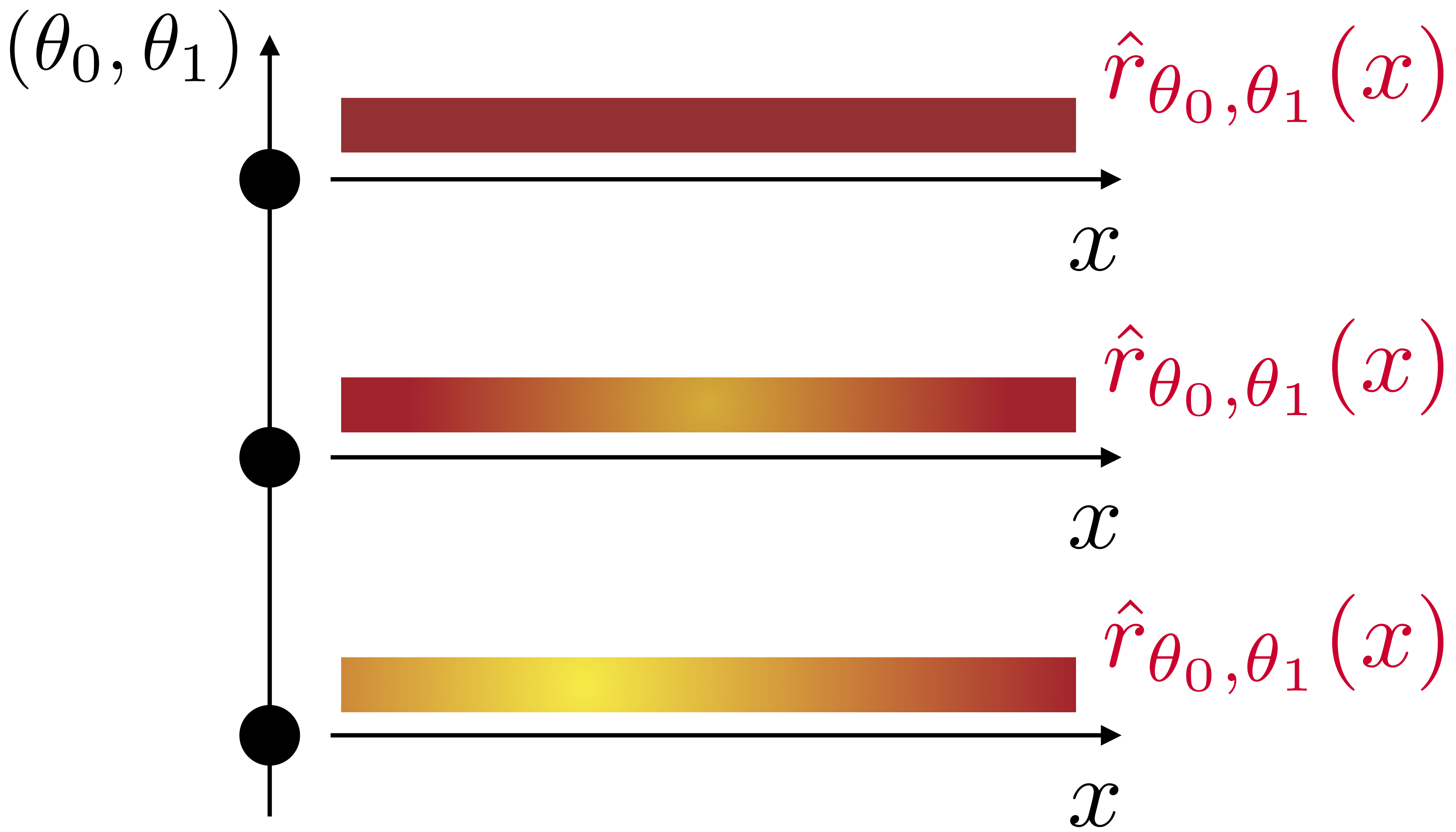
$$r(x,z|\theta_0,\theta_1) = \frac{p(x,z|\theta_0)}{p(x,z|\theta_1)} = \prod_i \frac{p(z_i|z_{i-1},\theta_0)}{p(z_i|z_{i-1},\theta_1)}$$



Two types of likelihood ratio estimators

A) Point by point:

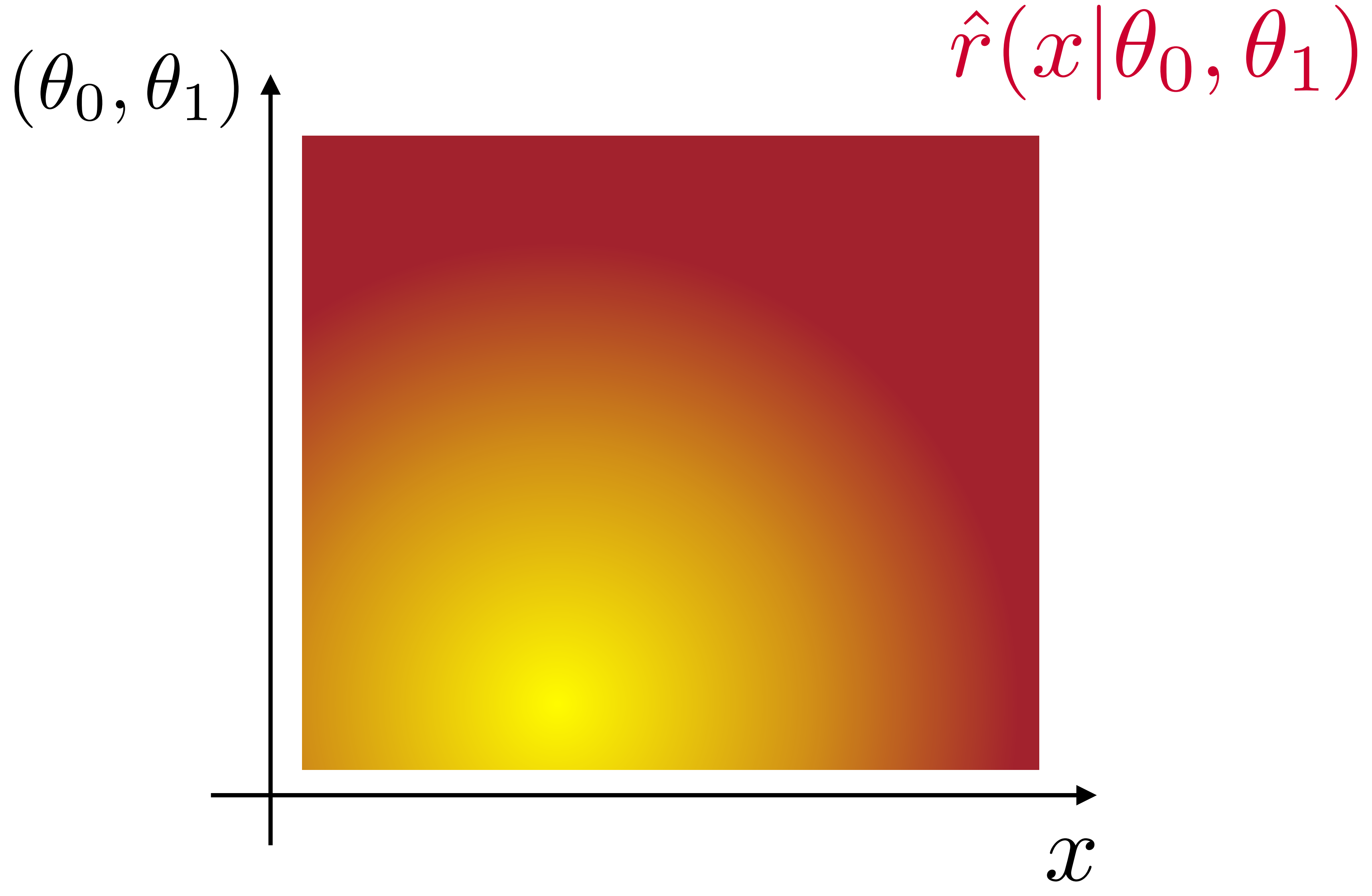
- first, define grid of parameter points $\{(\theta_0, \theta_1)\}$
- for each combination (θ_0, θ_1) , create separate estimator $\hat{r}_{\theta_0,\theta_1}(x)$
- final results can be interpolated between grid points

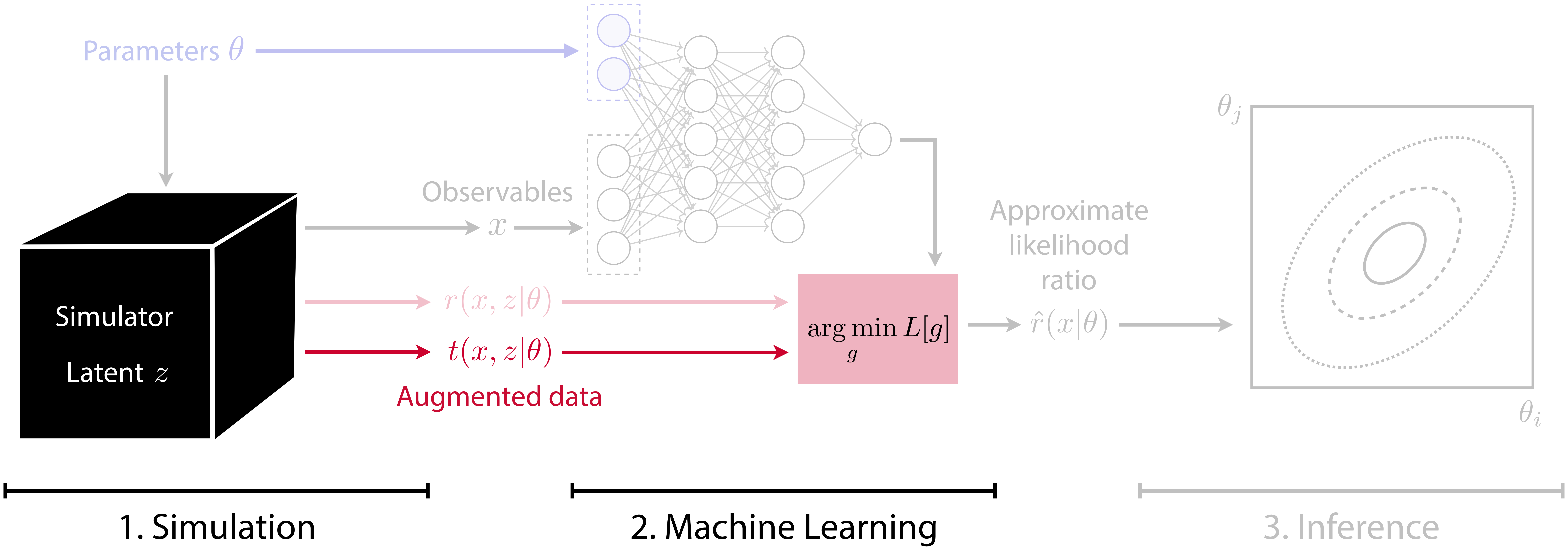


B) Parameterized:

[K. Cranmer, J. Pavez, G. Louppe 1506.02169; P. Baldi et al. 1601.07913]

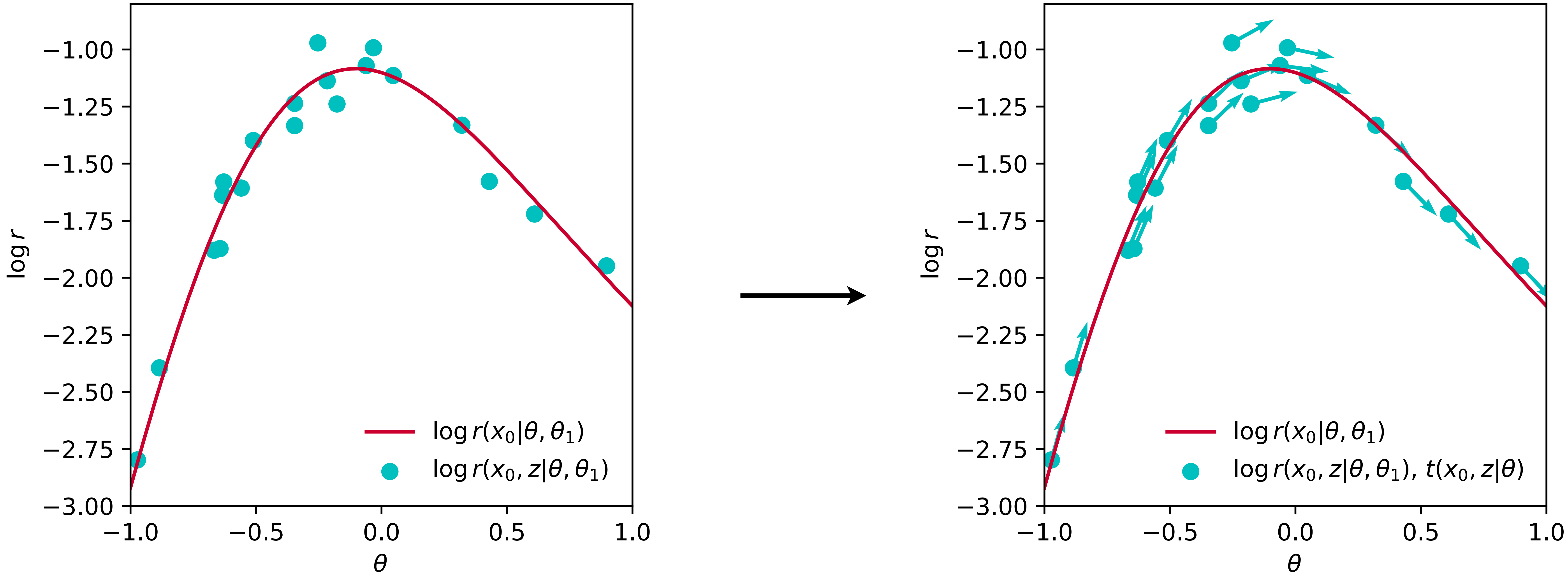
- create one estimator $\hat{r}(x|\theta_0, \theta_1)$ that is a function of θ_0 and θ_1
- no further interpolation necessary
- “borrows information” from close points




Parameters θ
Observables x
Simulator
Latent z
$r(x, z|\theta)$
$t(x, z|\theta)$
Augmented data
$\arg\min_g L[g]$
Approximate likelihood ratio
$\hat{r}(x|\theta)$
θ_j
θ_i
1. Simulation
2. Machine Learning
3. Inference

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 itself is intractable. But…

Learning the score

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

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

We want the score

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

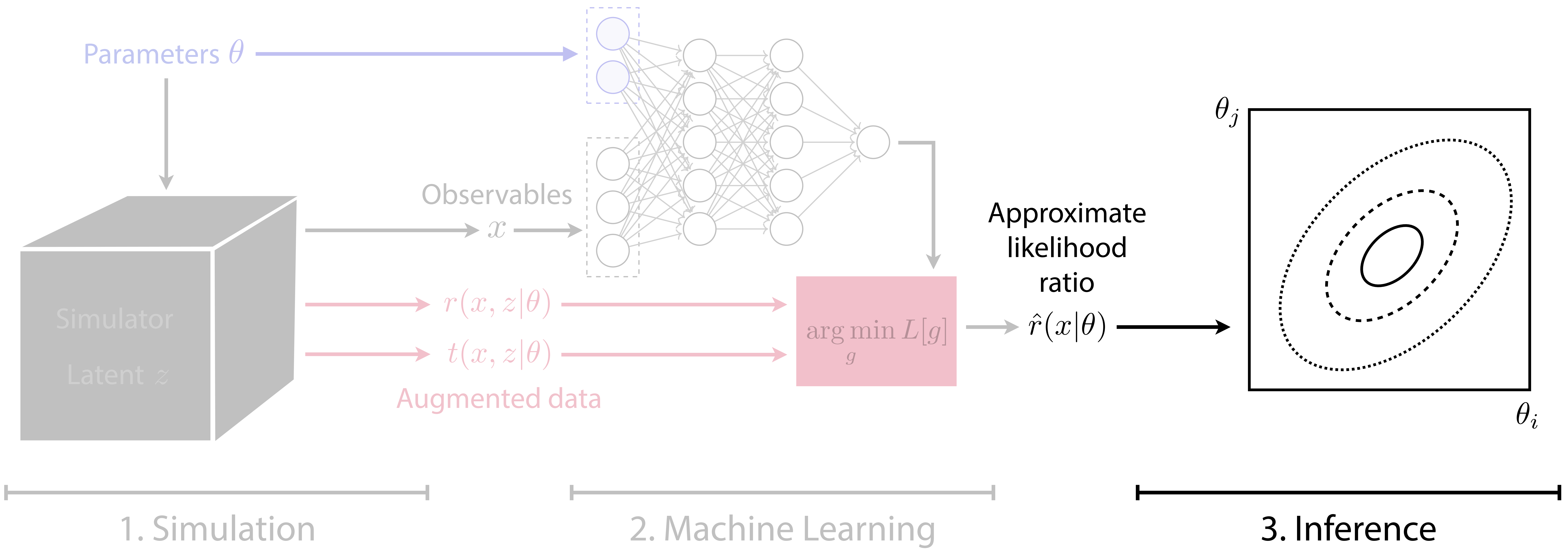
Given $t(x, z|\theta_0)$,
we define the functional

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

One can show it is minimized by

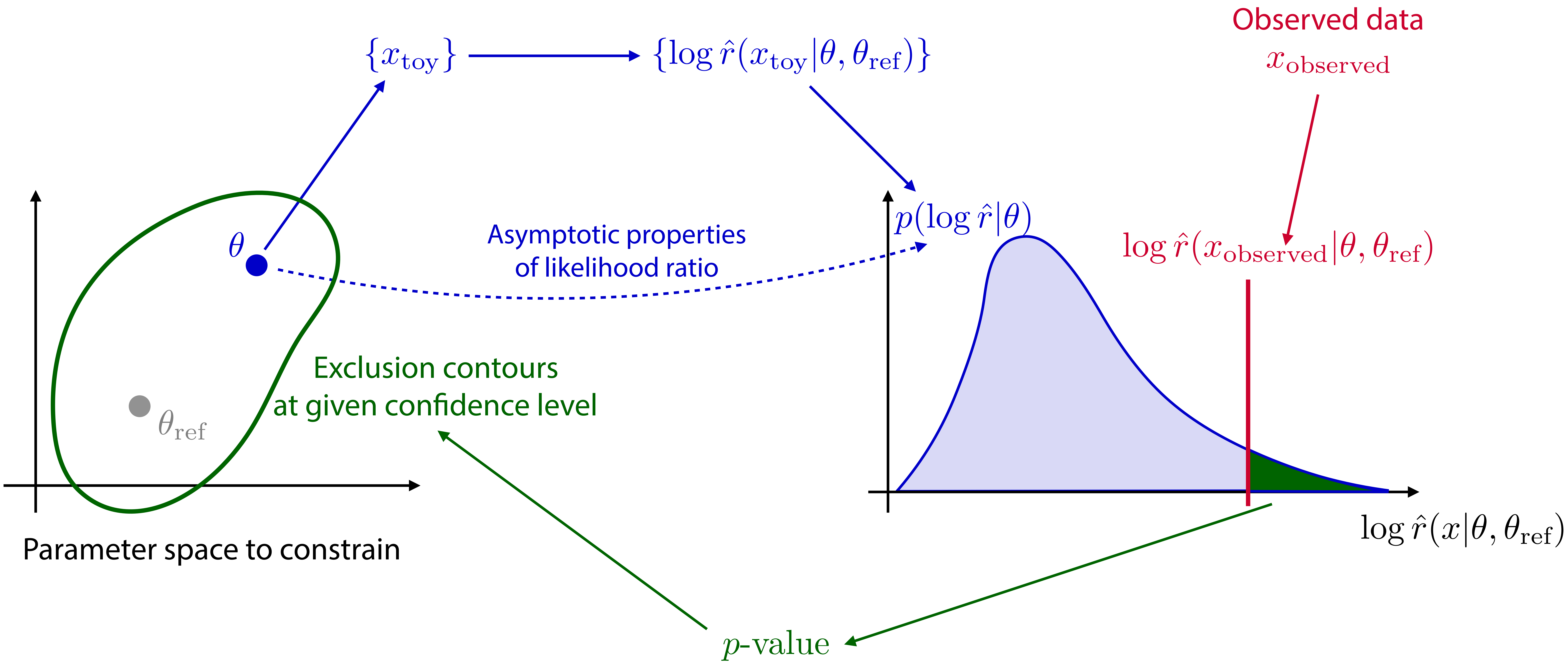
$$t(x|\theta_0) = \underset{\hat{t}(x|\theta_0)}{\arg\min}\, L_t[\hat{t}(x|\theta_0)]\,.$$

Again, we implement this minimization through machine learning.



Parameters θ
Observables
x
Simulator
Latent z
$r(x, z|\theta)$
$t(x, z|\theta)$
Augmented data
$\arg\min_g L[g]$
Approximate likelihood ratio
$\hat{r}(x|\theta)$
θ_j
θ_i
1. Simulation
2. Machine Learning
3. Inference

Limit setting (frequentist)



A family of new inference techniques

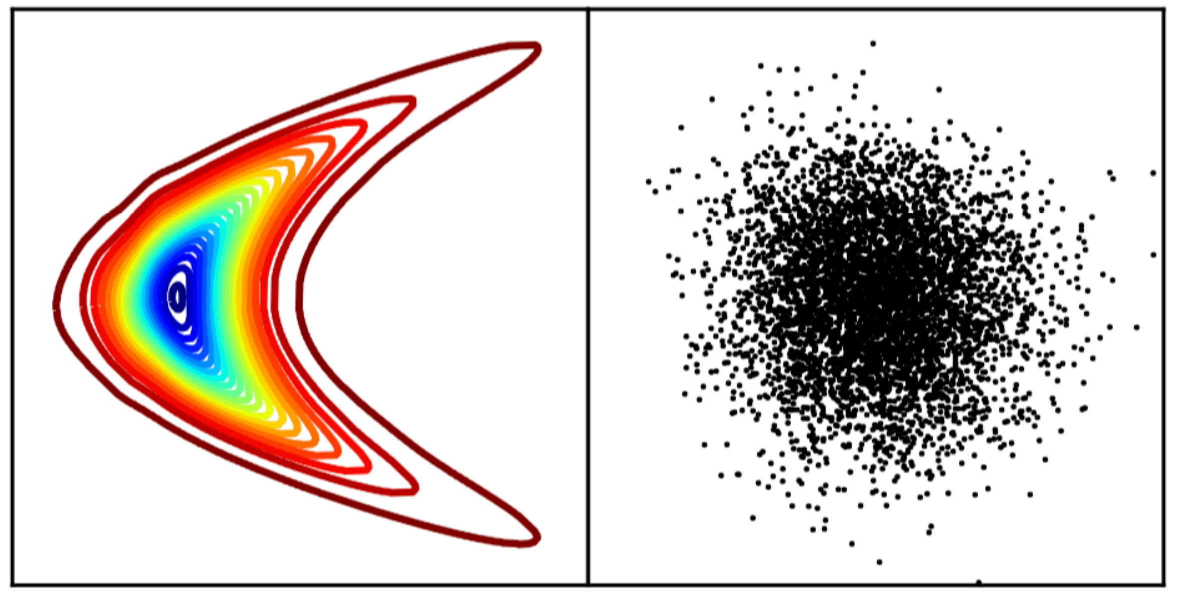
Method	Simulate	Extract $r(x,z)$	Extract $t(x,z)$	NN estimates	Asympt. exact	Generative
ROLR	$\theta_0 \sim \pi(\theta), \theta_1$	✓		$\hat{r}(x\|\theta_0,\theta_1)$	✓	
CASCAL	$\theta_0 \sim \pi(\theta), \theta_1$		✓	$\hat{r}(x\|\theta_0,\theta_1)$	✓	
ALICE	$\theta_0 \sim \pi(\theta), \theta_1$		✓	$\hat{r}(x\|\theta_0,\theta_1)$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

Performance gains with cross-entropy-based loss

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

A family of new inference techniques

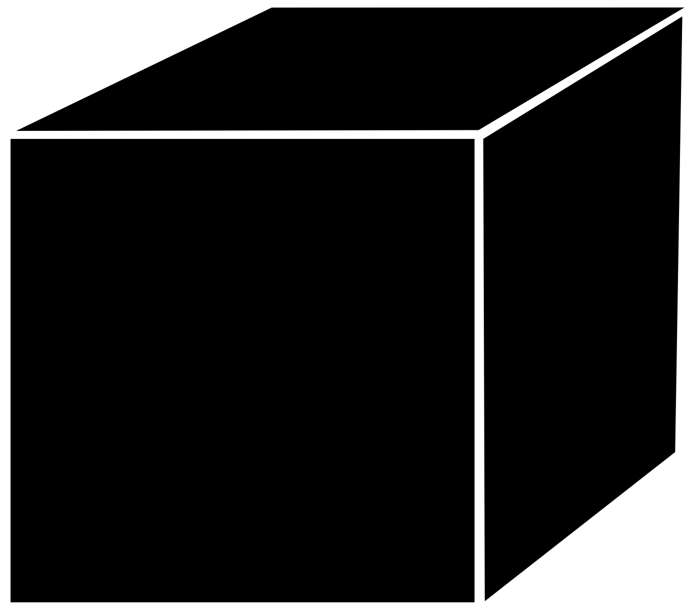
Method	Simulate	Extract $r(x,z)$	Extract $t(x,z)$	NN estimates	Asympt. exact	Generative
ROLR	$\theta_0 \sim \pi(\theta), \theta_1$	✓		$\hat{r}(x\|\theta_0,\theta_1)$	✓	
CASCAL	$\theta_0 \sim \pi(\theta), \theta_1$		✓	$\hat{r}(x\|\theta_0,\theta_1)$	✓	
ALICE	$\theta_0 \sim \pi(\theta), \theta_1$		✓	$\hat{r}(x\|\theta_0,\theta_1)$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



Combination with state-of-the-art conditional neural density estimators, e.g. normalizing flows

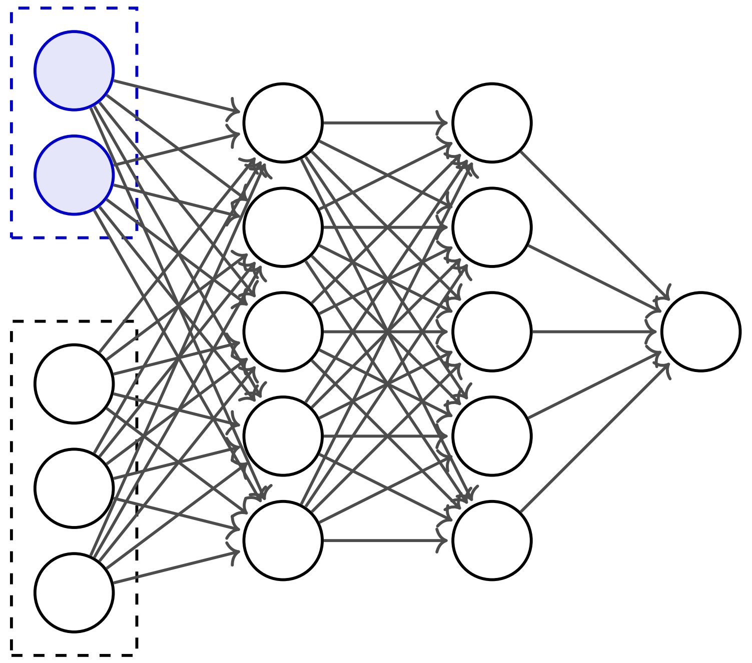
[everything by G. Papamakarios: G. Papamakarios, T. Pavlakou, I. Murray 1705.07057; G. Papamakarios, D. Sterratt, I. Murray 1805.07226; …]

Systematics



Don't fully trust the simulator?

- Nuisance parameters to model systematic uncertainties
- Methods learn dependence both on parameters of interest and nuisance parameters. Then we can construct profile likelihood and "nuisance-hardened" score
  [P. de Castro, T. Dorigo 1806.04743;
  J. Alsing, B. Wandelt 1903.01473]
- Alternatively: Robustness to nuisance with adversarial training
  [G. Louppe, M. Kagan, K. Cranmer 1611.01046]



Don't blindly trust the neural network?

- Diagnostic cross checks: known expectation values, "critic" tests
- Calibration / Neyman construction with toys:
  badly trained network can lead to suboptimal limits, but not to wrong limits

WBF Higgs to four leptons with detector effects