

Impressions

PHYSTAT: Stats Meets ML

11 Dec 2023

Bayesian Methodologies with pyhf

Matthew Feickert^{1,*}, *Lukas Heinrich*^{2,**}, and *Malin Horstmann*^{2,***}

¹University of Wisconsin-Madison, Madison, Wisconsin, USA

²Technical University of Munich, Munich, Germany

Abstract. `bayesian_pyhf` is a Python package that allows for the parallel Bayesian and frequentist evaluation of multi-channel binned sta-

Profile Likelihoods in Cosmology: When, Why and How illustrated with Λ CDM, Massive Neutrinos and Dark Energy

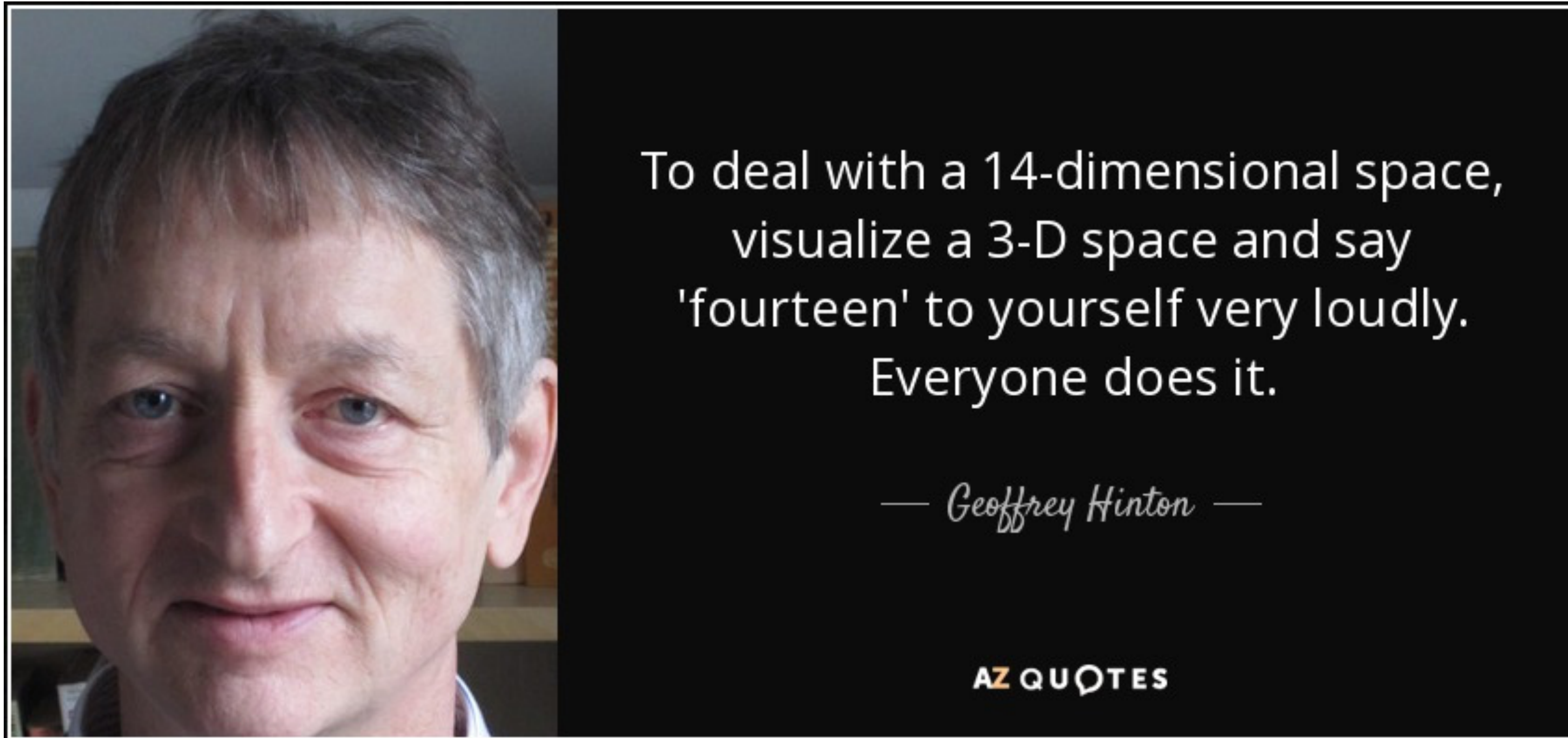
Laura Herold,^{1,*} Elisa G. M. Ferreira,² and Lukas Heinrich³

¹Department of Physics and Astronomy, Johns Hopkins University,
3400 North Charles Street, Baltimore, Maryland 21218, USA

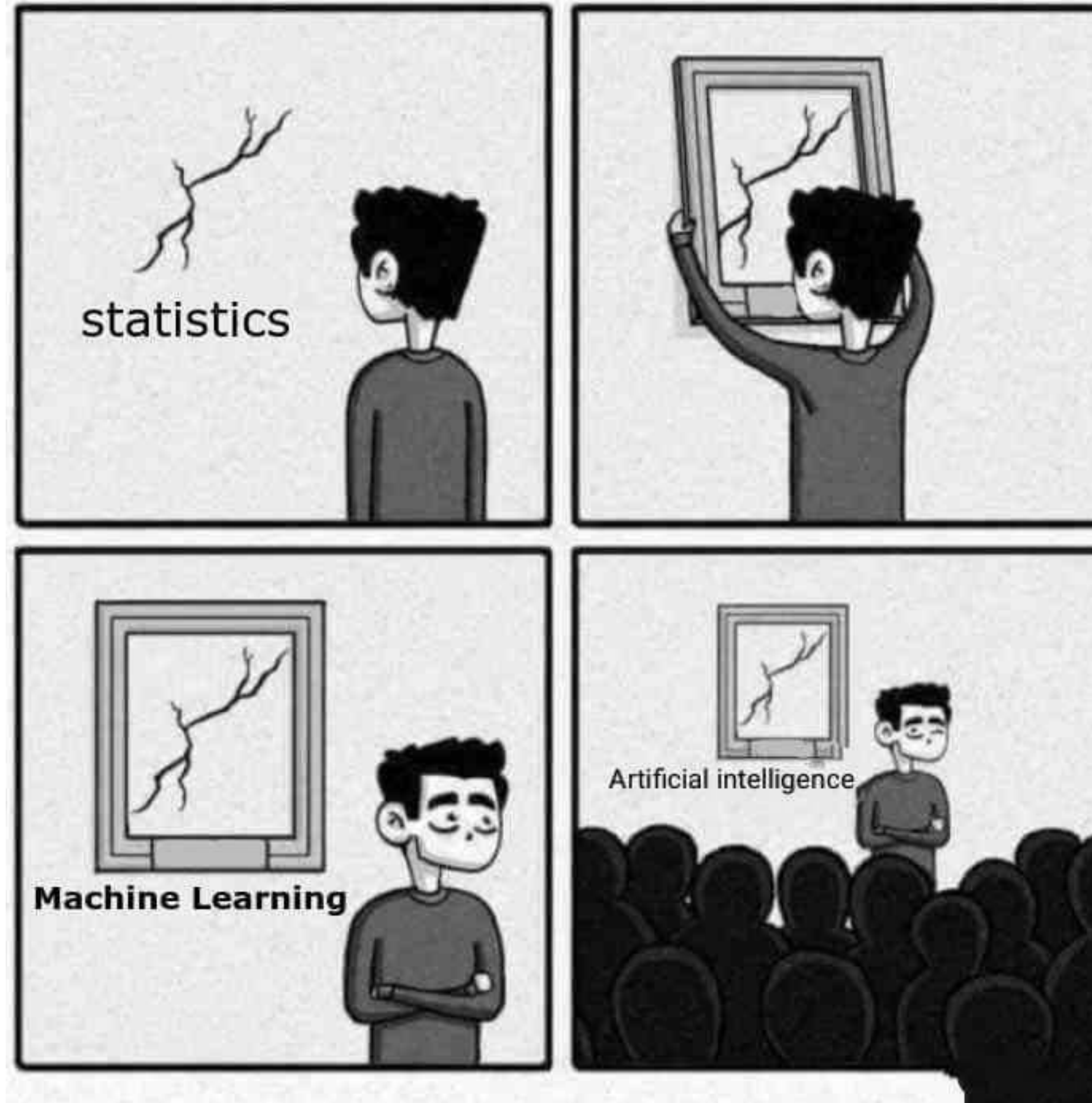
²Kavli Institute for the Physics and Mathematics of the Universe (WPI),

Lukas Heinrich, 12. Sept. 2024


My ~~5-D~~ 14-Dimensional Outline




Stats Meets ML?



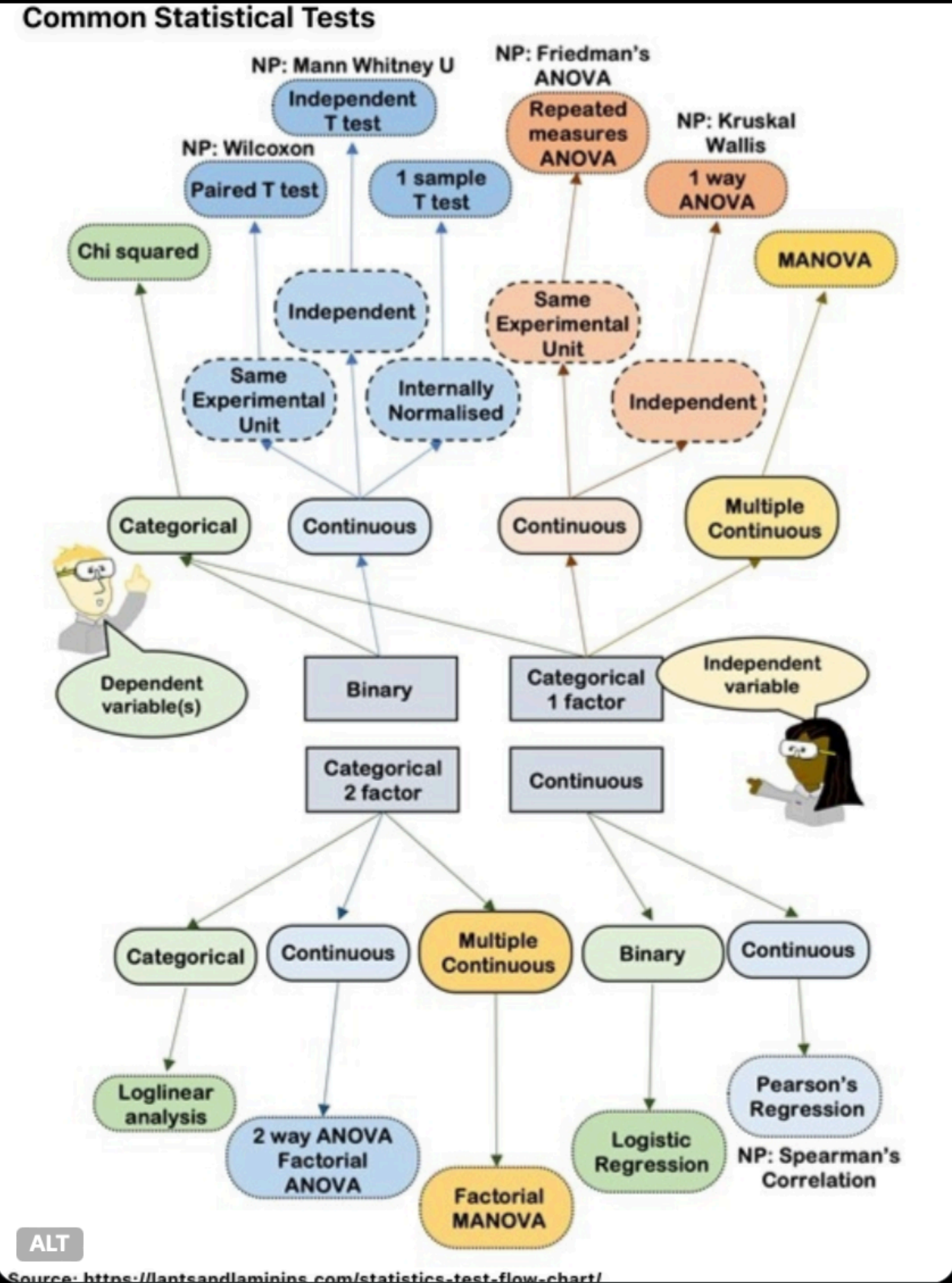
Stats Meets ML?

 **David Pfau**
@pfau

Whenever people say "statistics and machine learning are the same thing" just know that not a single person in machine learning knows (or cares about) any of this.

 **Prof Lennart Nacke, PhD** @acagamic · Aug 15
How I pick the right statistical test for my experimental variables

Common Statistical Tests



The flowchart starts with 'Dependent variable(s)' and 'Independent variable'. It branches into 'Categorical' and 'Continuous' for both. For categorical dependent variables, it leads to 'Loglinear analysis'. For categorical independent variables, it leads to 'Categorical 1 factor' (1 way ANOVA, Kruskal Wallis) and 'Categorical 2 factor' (2 way ANOVA, Factorial ANOVA). For continuous dependent variables, it leads to '1 sample T test' (Mann Whitney U, Paired T test, Wilcoxon) and 'Multiple Continuous' (MANOVA). For continuous independent variables, it leads to 'Continuous' (Pearson's Regression, Spearman's Correlation) and 'Multiple Continuous' (Factorial MANOVA). 'Same Experimental Unit' and 'Internally Normalised' are also noted as conditions for certain tests.

ALT

Source: <https://lantsandlaminins.com/statistics-test-flow-chart/>

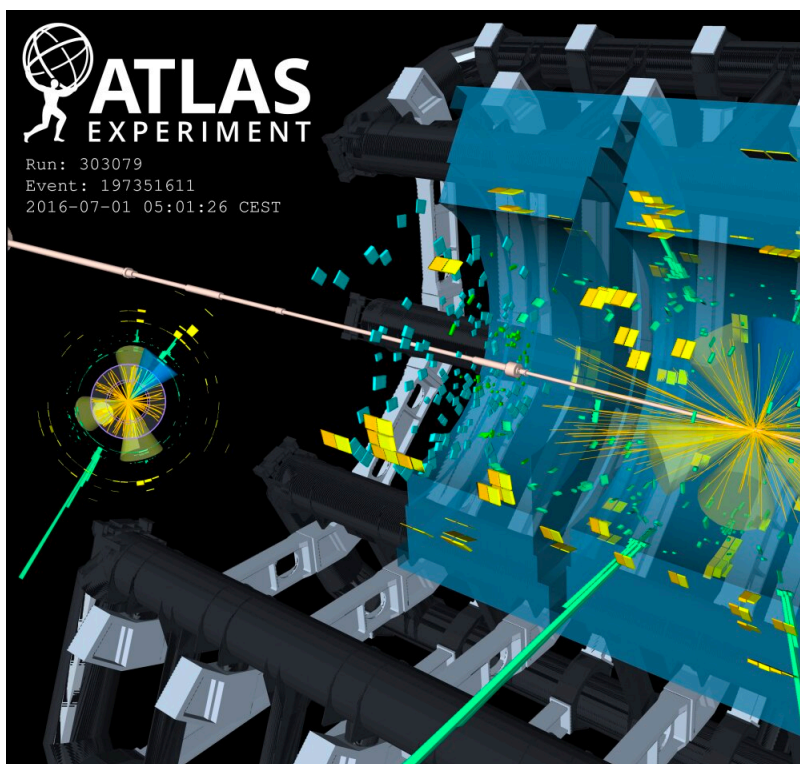
3:26 PM · Aug 17, 2024 · 151.2K Views

HEP (and others) & ML: deeply related

$$\mathcal{L} = -\frac{1}{4} F_{\mu\nu} F^{\mu\nu} + i\bar{\psi}\not{D}\psi + h.c. + \bar{\psi}_i y_{ij} \psi_j \phi + h.c. + \frac{1}{2} D_\mu \phi^\dagger D^\mu \phi - V(\phi)$$



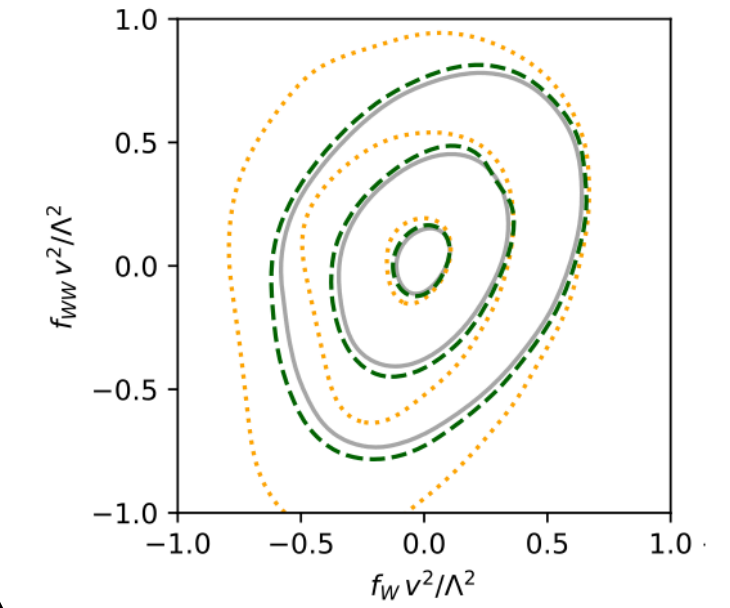
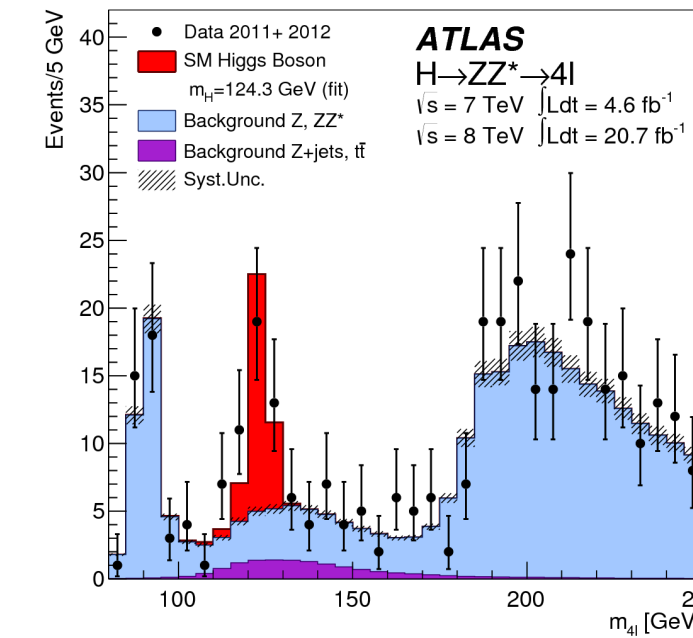
generate low-level, high-dim data from high-level concepts



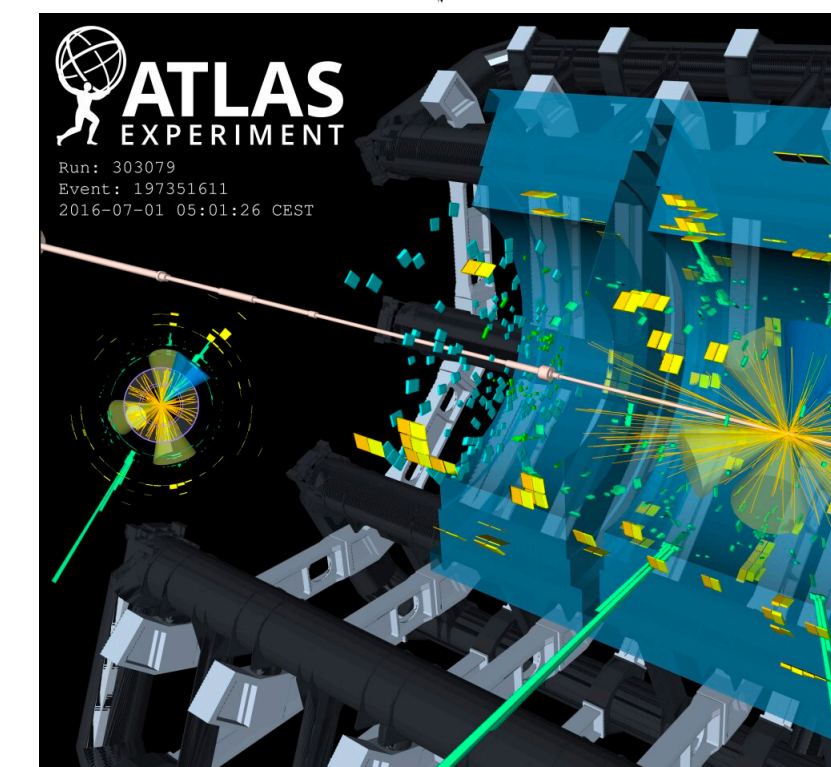
High-Level Concept



Low-Level Data



reconstruct high level concepts from low-level, high-dim data



HEP (and others) & ML: deeply related

street style photo of a woman selling pho at a Vietnamese street market, sunset, shot on fujifilm

generate low-level, high-dim data from high-level concepts



This is a picture of Barack Obama. His foot is positioned on the right side of the scale. The scale will show a higher weight.

reconstruct high level concepts from low-level, high-dim data



High-Level Concept

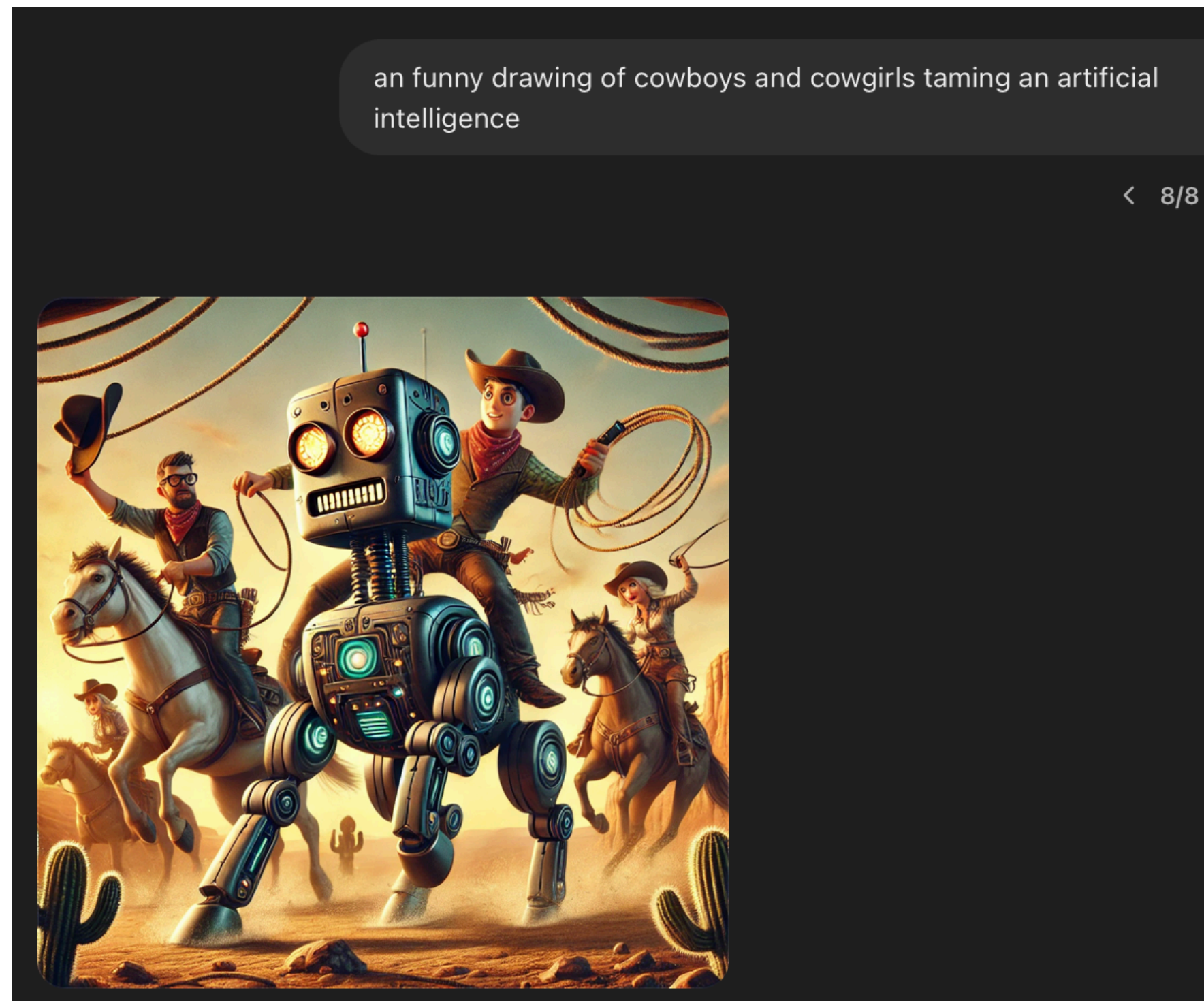
Low-Level Data



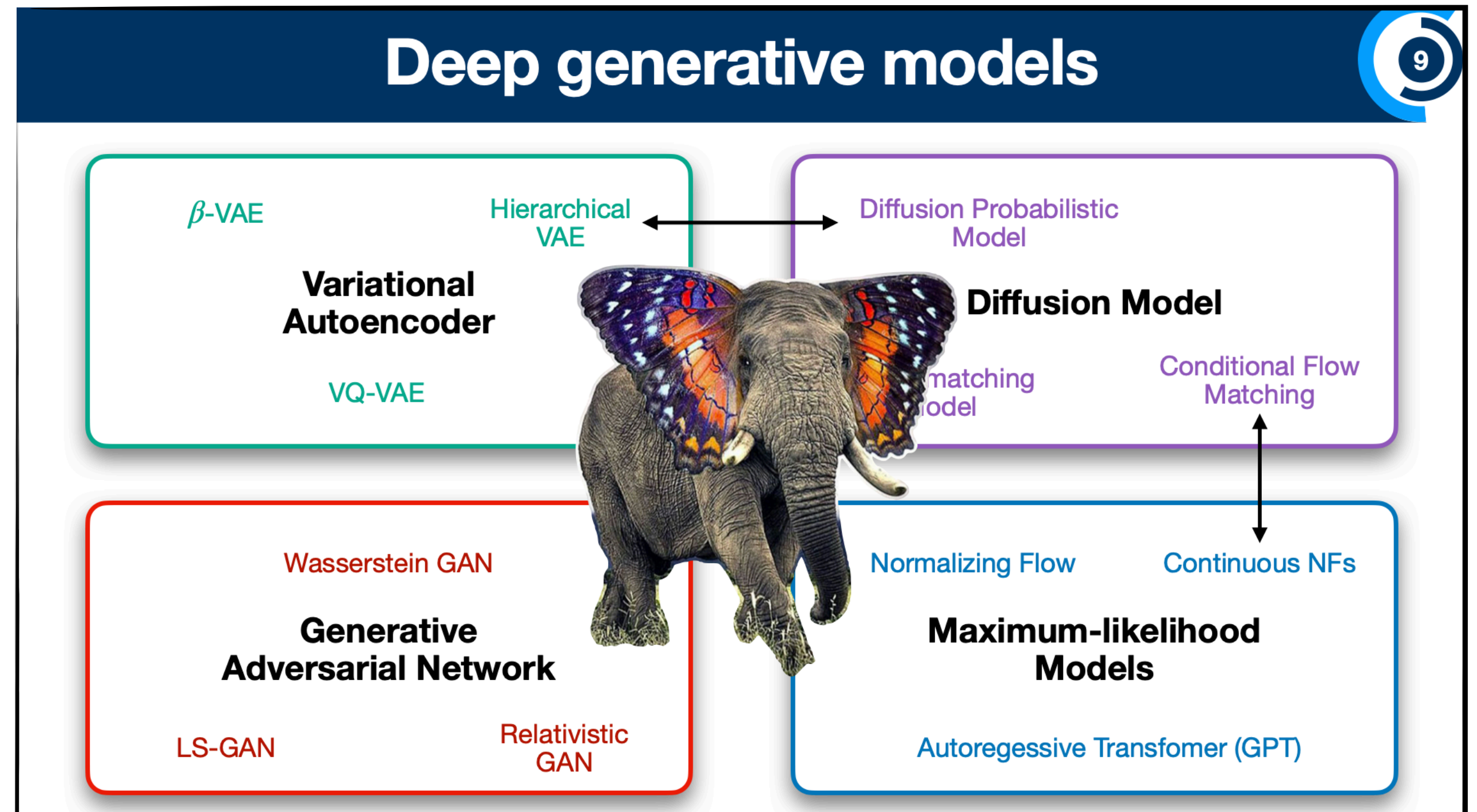
ML Wild West

A focus of the last years is to “to learn the technology” - build up technical capability around this new set of “new numerical tools” (Tilman)

We have gotten very good at this



(Gregor)



(Ramon)

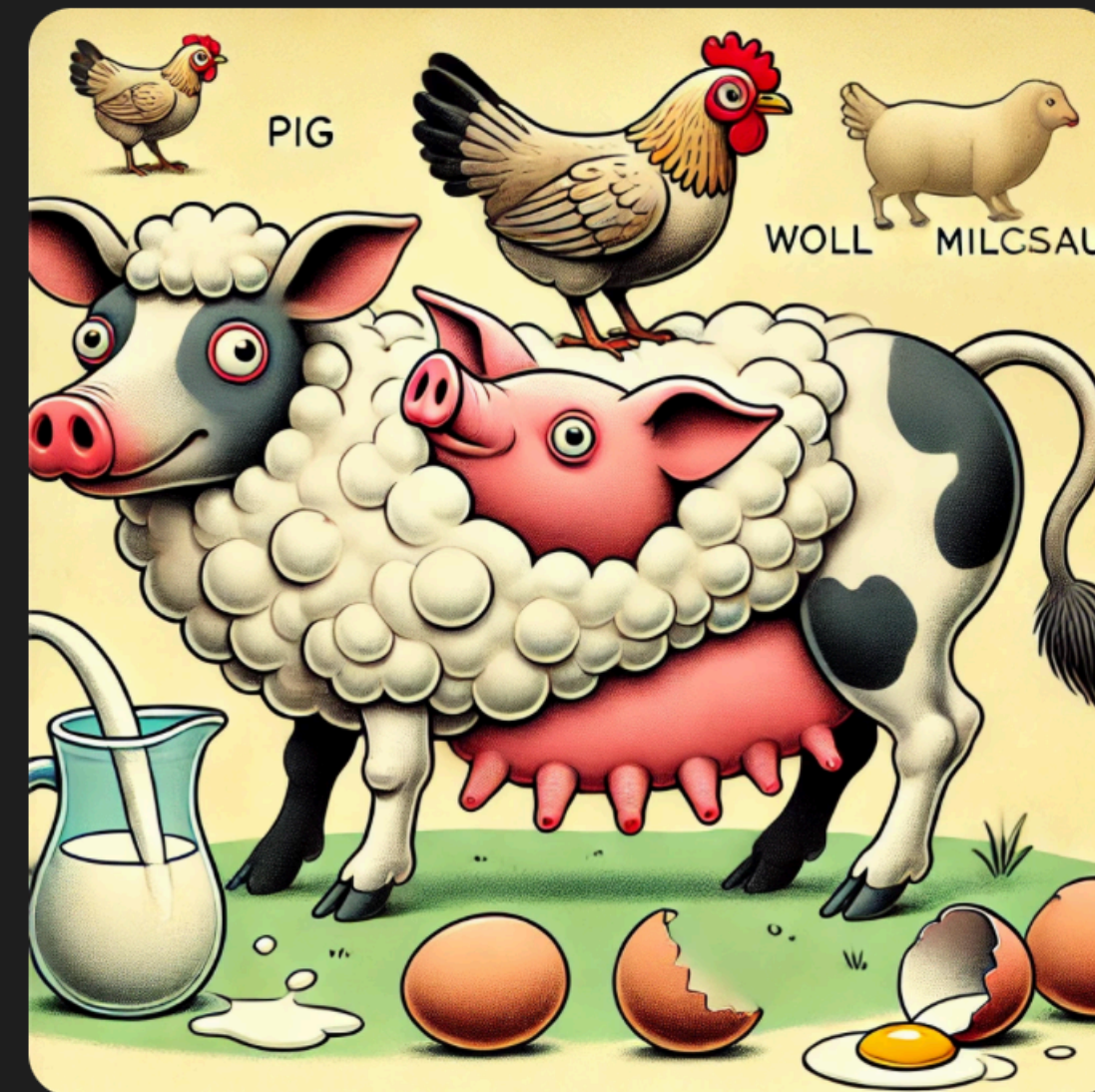
ML: Eierlegende Wollmichsau?

A picture of a Eierlegende Wollmichsau

The "Eierlegende Wollmilchsau" (German for "egg-laying wool-milk-sow") is a humorous metaphor used to describe something that is supposed to do everything, typically in an exaggerated and impossible way. It represents a creature that lays eggs, produces wool, gives milk, and functions as a pig.

I'll create a whimsical and surreal image of this fictional creature combining elements of a pig, sheep (for wool), cow (for milk), and hen (for eggs), in a humorous and exaggerated style.

Generating this now!



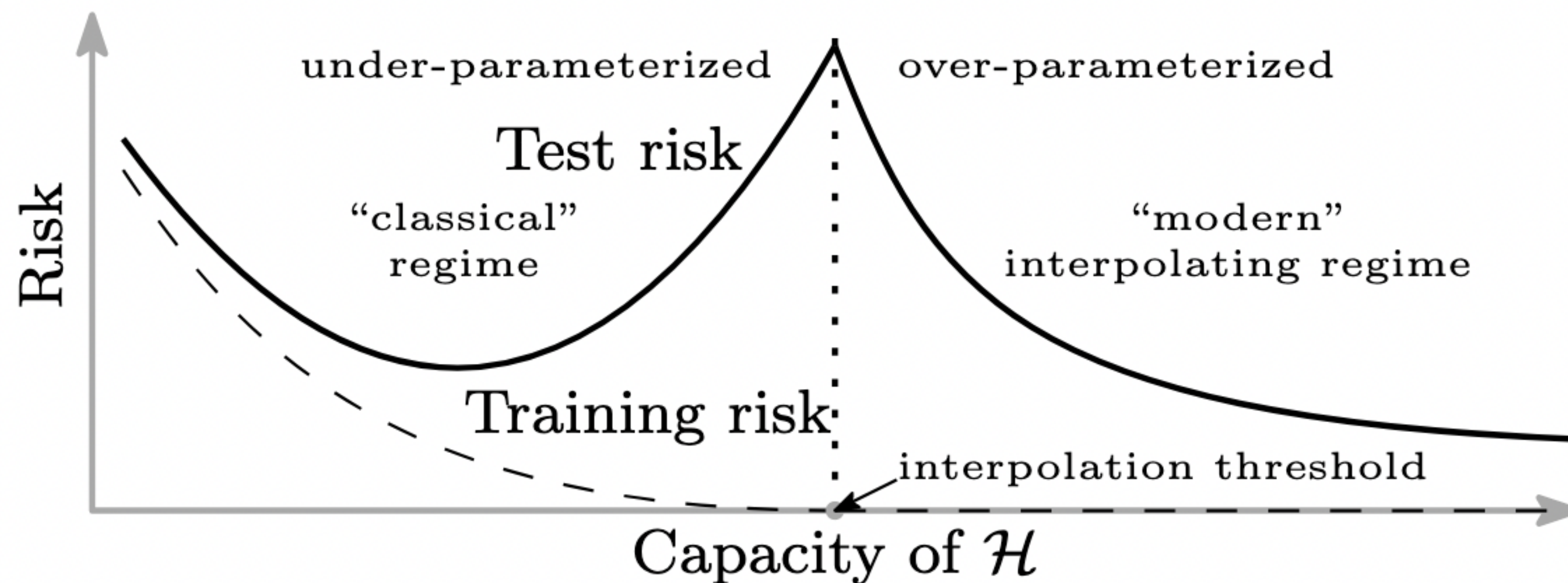
Here is a whimsical representation of the "Eierlegende Wollmilchsau," capturing the fantastical nature of this humorous creature. Let me know if you'd like any changes!

(Ramon)

Where can a Statistical Point of View help?

ML is an extremely empirical science. In many ways it's successful because it ignored theory to some extent

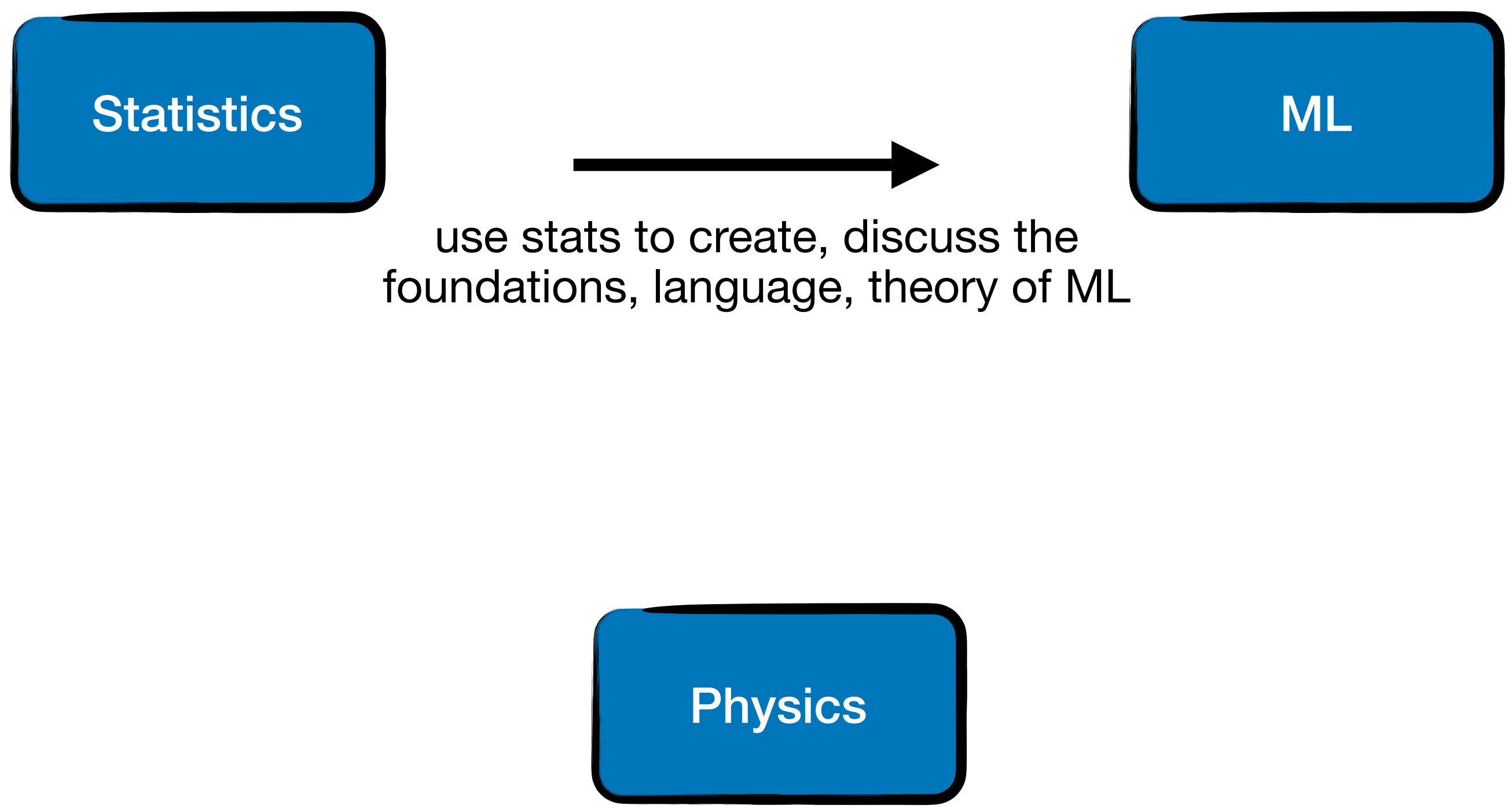
But once we tame this new technology maybe stats can in some ways help us tame ourselves.



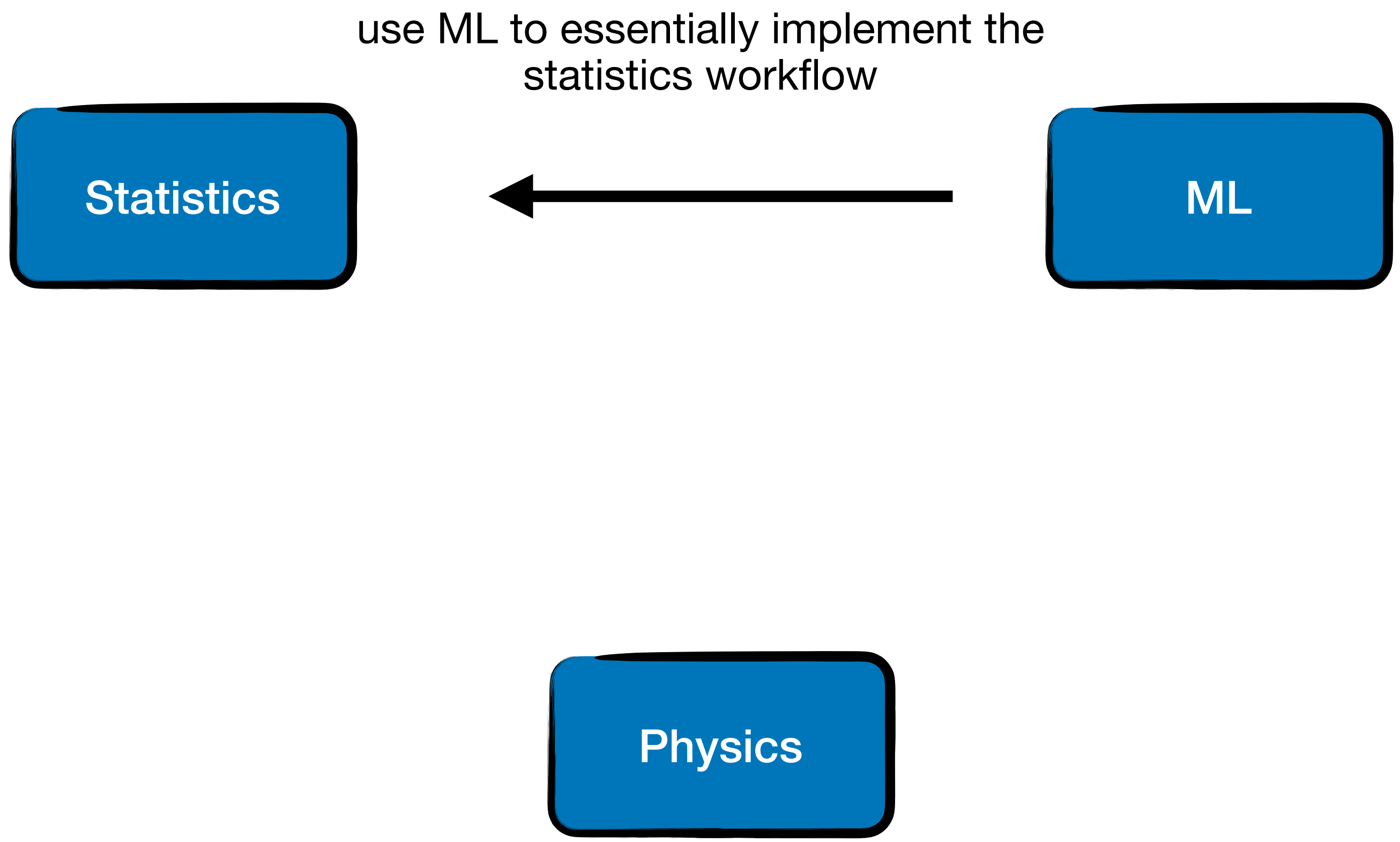
but e.g. interestingly, this just re-discovered some old statistics
- is this the Feldman-Cousins of ML? ;)

(Pierre)

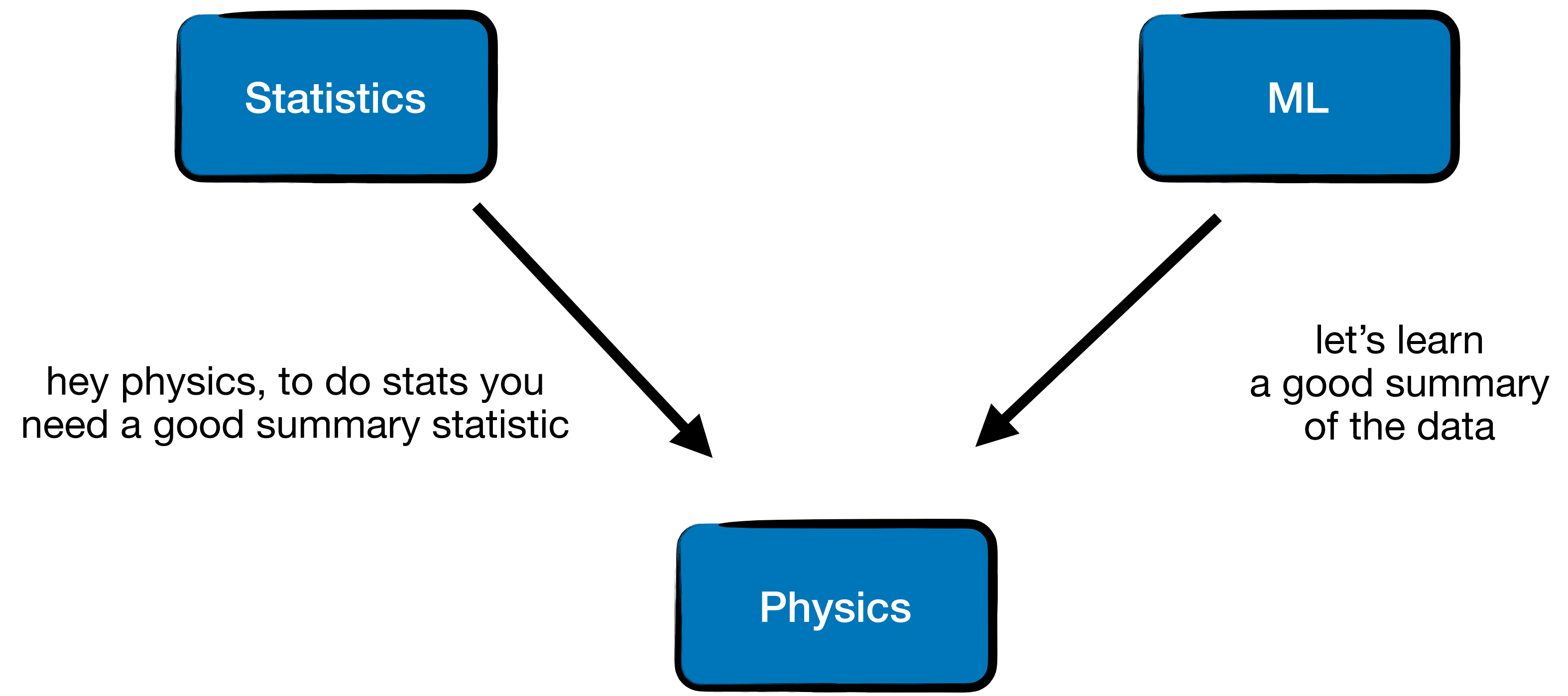
Where is the Statistics ?



Where is the Statistics ?

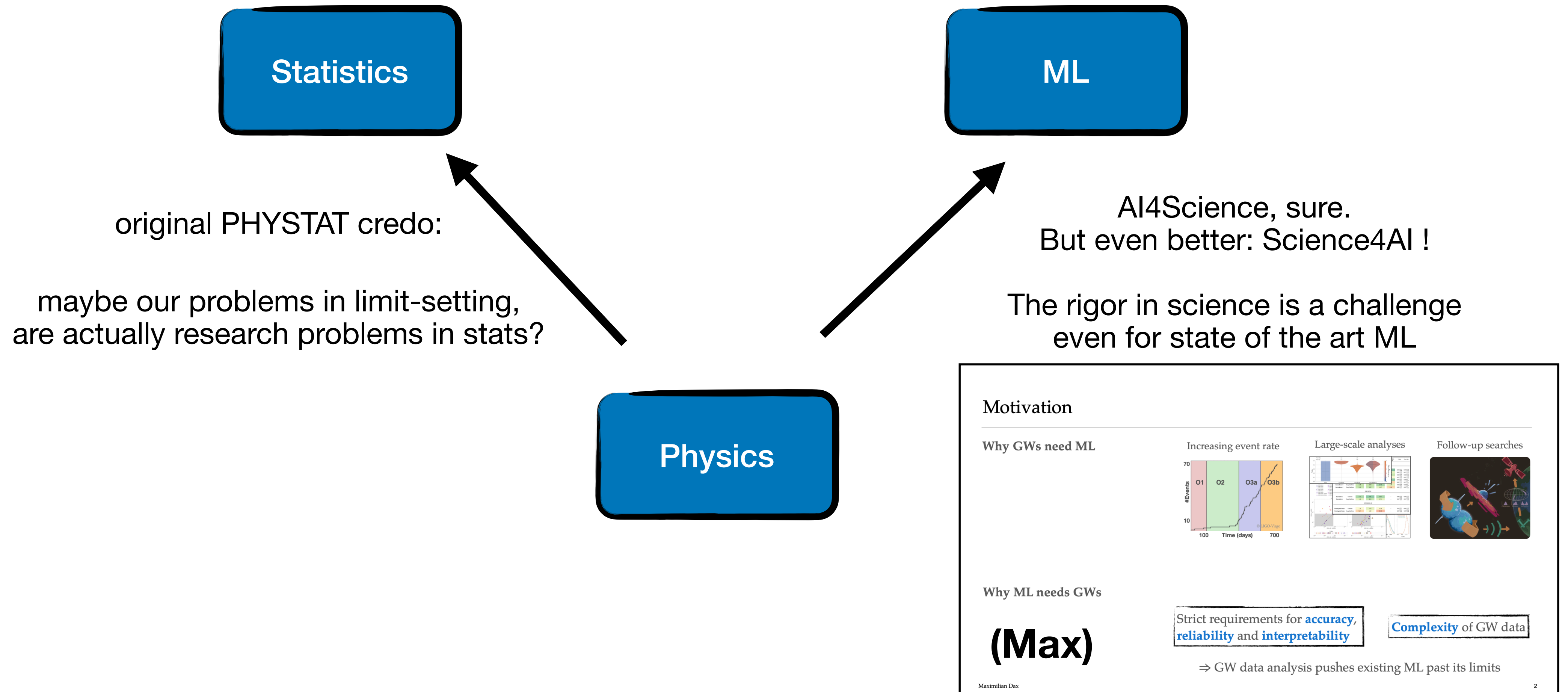


Where is the Statistics ?



Where is the Statistics ?

Our Delusion of Grandeur (or not?)



Anomaly Detection

Statistics can help bring clarity to how we frame ML use-cases

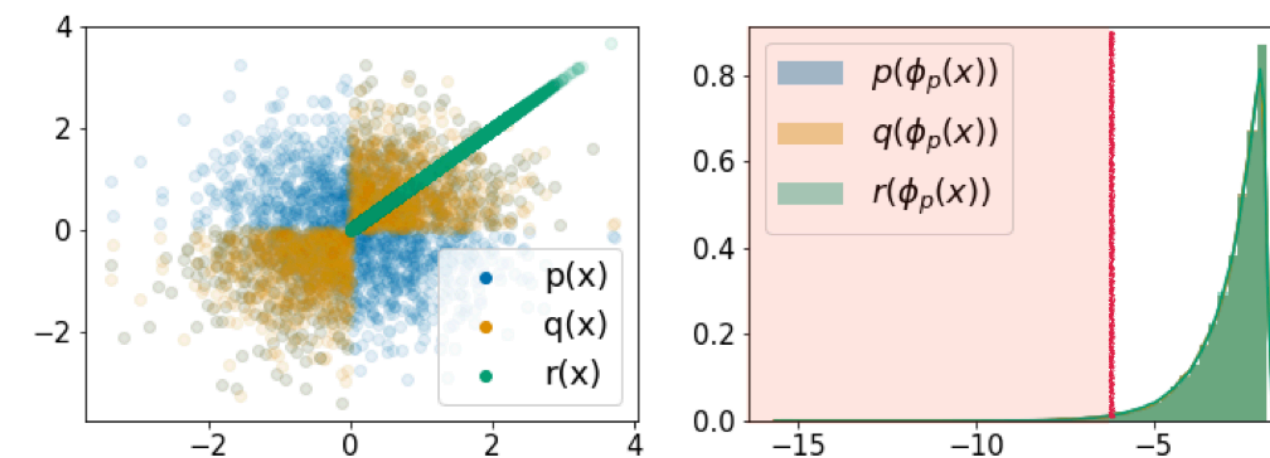
where do we want statistical power ?

This is the key in AD:

Are we ready to do it?
If yes in what language?

What's the right way to perform anomaly detection?

Proposition (informal): *No method can guarantee performance better than random guessing without assumptions on the out-distributions.*



(Lily)

Need to specify out-distributions of interest!

Lily H. Zhang, Mark Goldstein, Rajesh Ranganath.
"Understanding Failures in Out-of-distribution
Detection with Deep Generative Models." *ICML 2021*.

Ultimately goes back to what we learn in Stats Intro:
there no universal most powerful test

Anomaly Detection

Statistics can help bring clarity to how we frame ML use-cases
where do we want statistical power ?

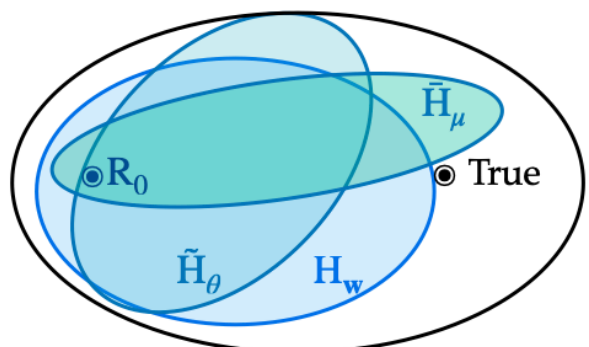
This is the key in AD:

**Are we ready to do it?
If yes in what language?**


September 10th, 2024

The problem of *model selection*

How to define the family of universal approximants?

$$f(x, w) = \log \left[\frac{n(x|D)}{n(x|R)} \right]$$


- Trade-off between *expressivity* and *specificity* is required
 - ▶ *Model selection*: hyper-parameters choice poses hard constraints.
 - ▶ *Regularization* is a powerful form of *inductive bias* (e.g. smoothness) affecting the learning dynamics

(Gaia) 

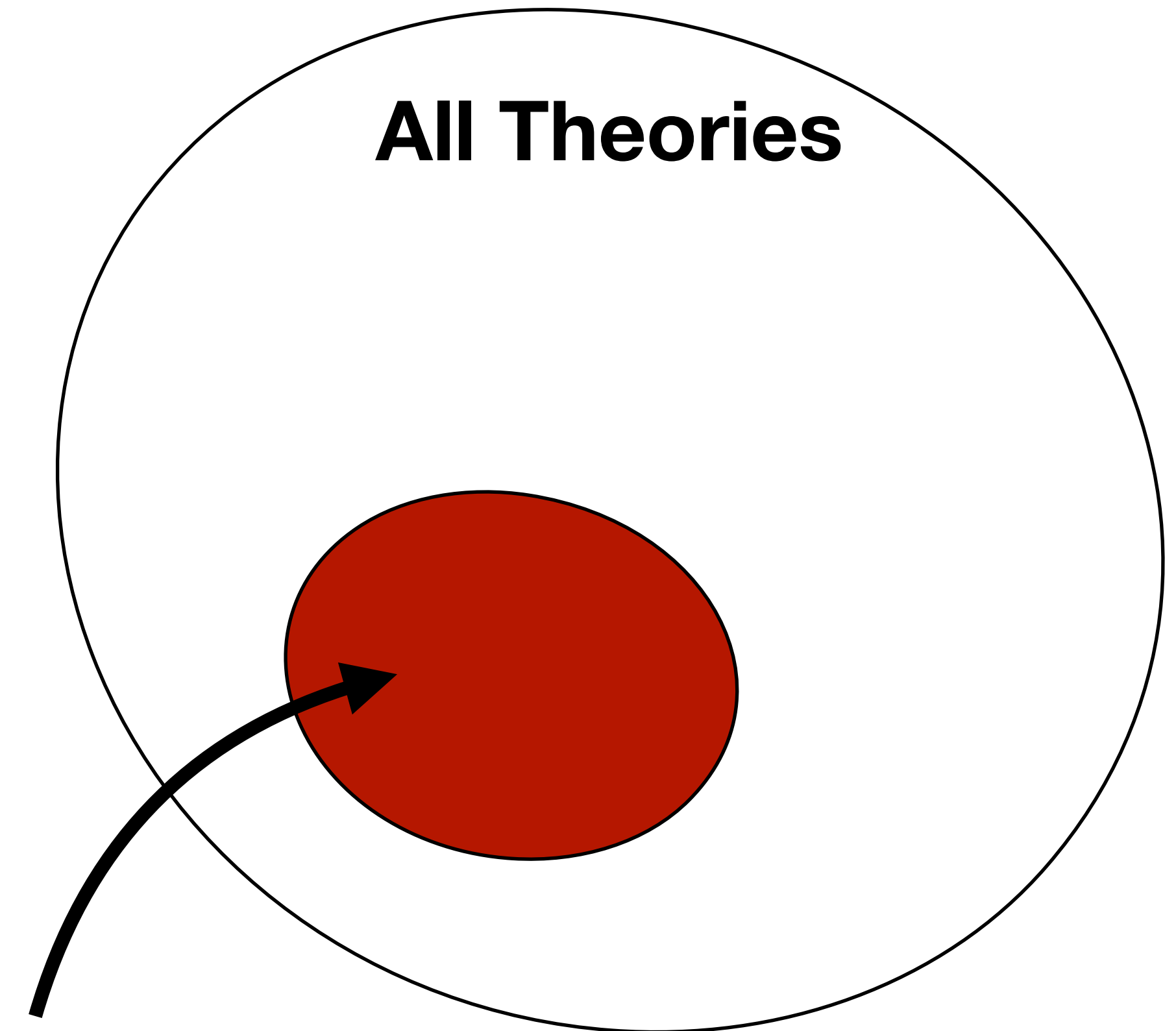
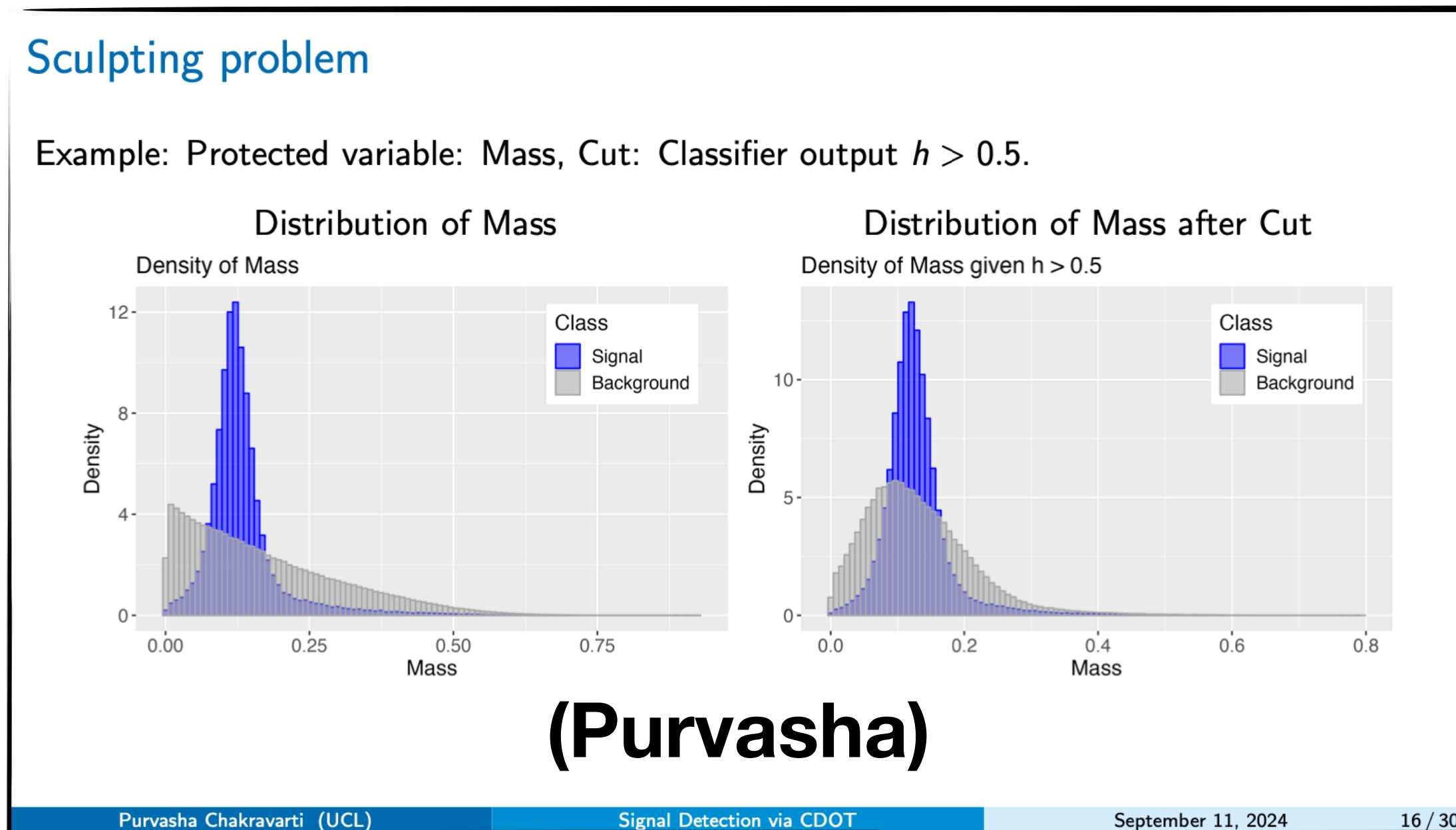
G. Grosso | gaia.grosso@cern.ch 17

Challenge to Theorists:

**How do we specify theories we care about without specifying the Lagrangian ?
(History: GUT Theories → Simplified Models → ???)**

Bump Hunts

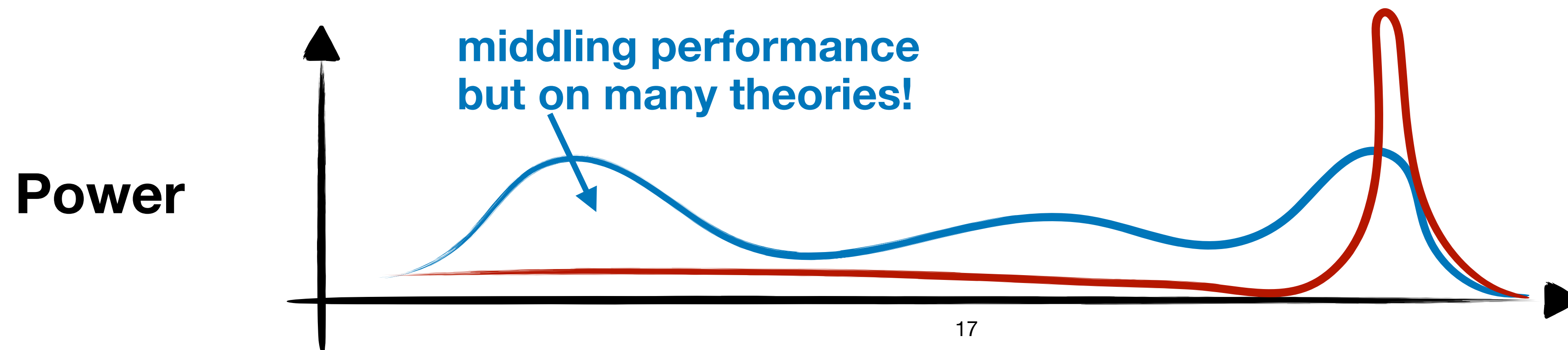
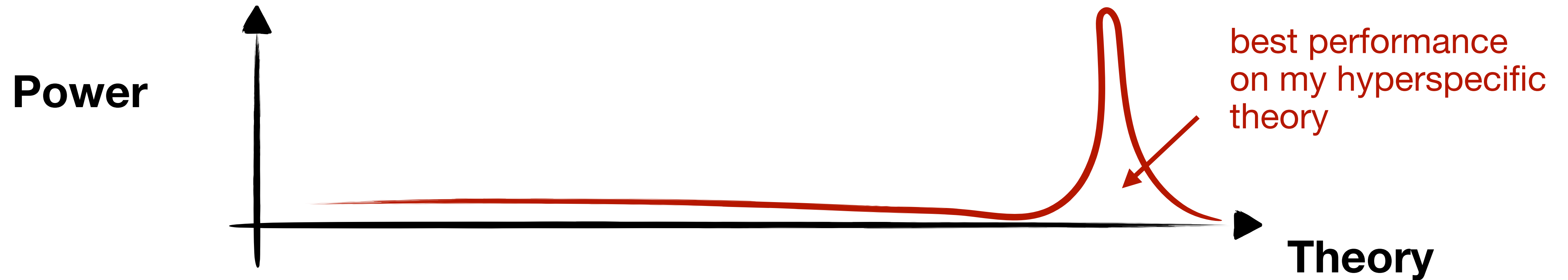
“Bump hunts” are doing exactly this: We specify a equivalence class of theories that share a very vaguely defined feature



**Theories with a resonance
at mass M (irrespective of the rest)**

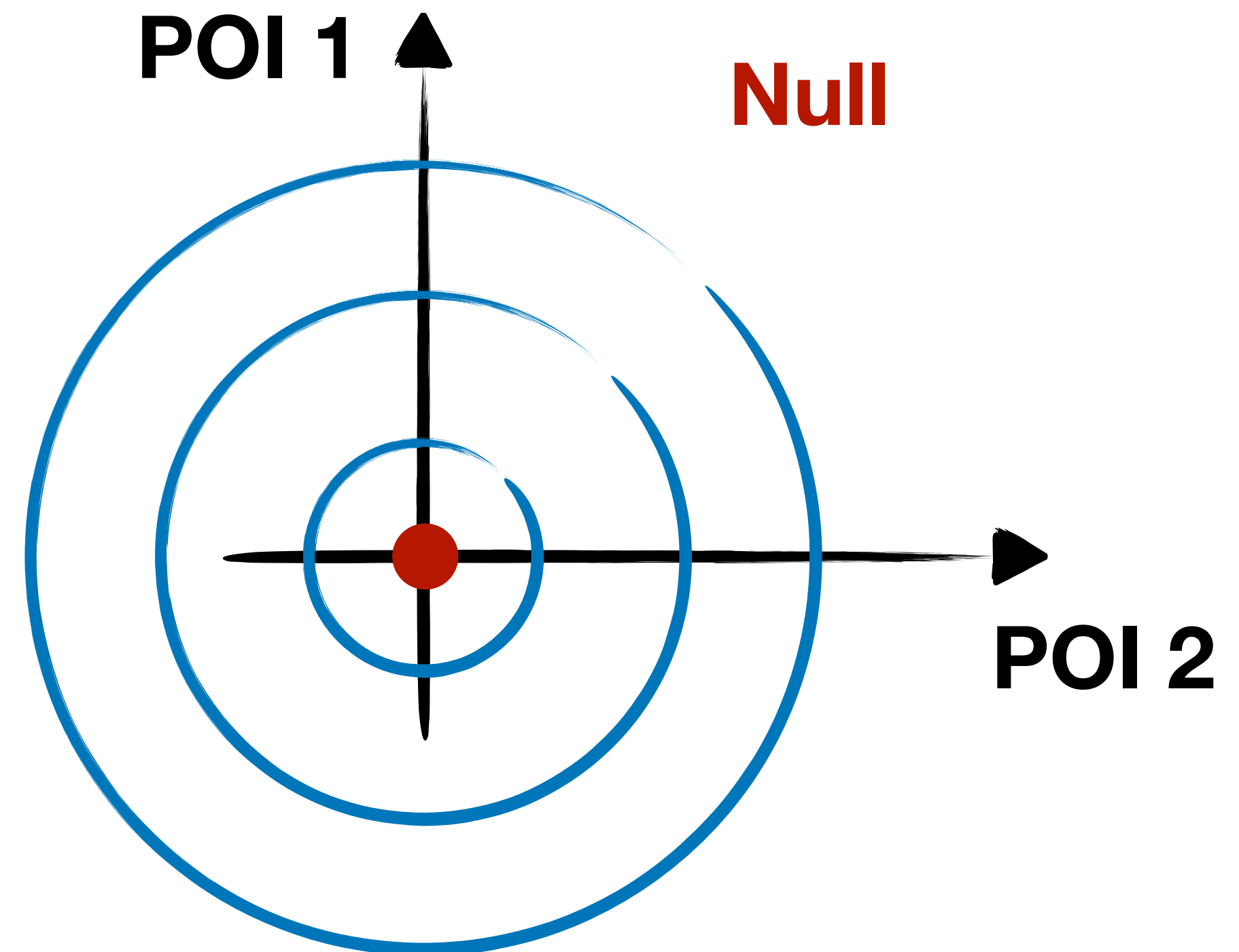
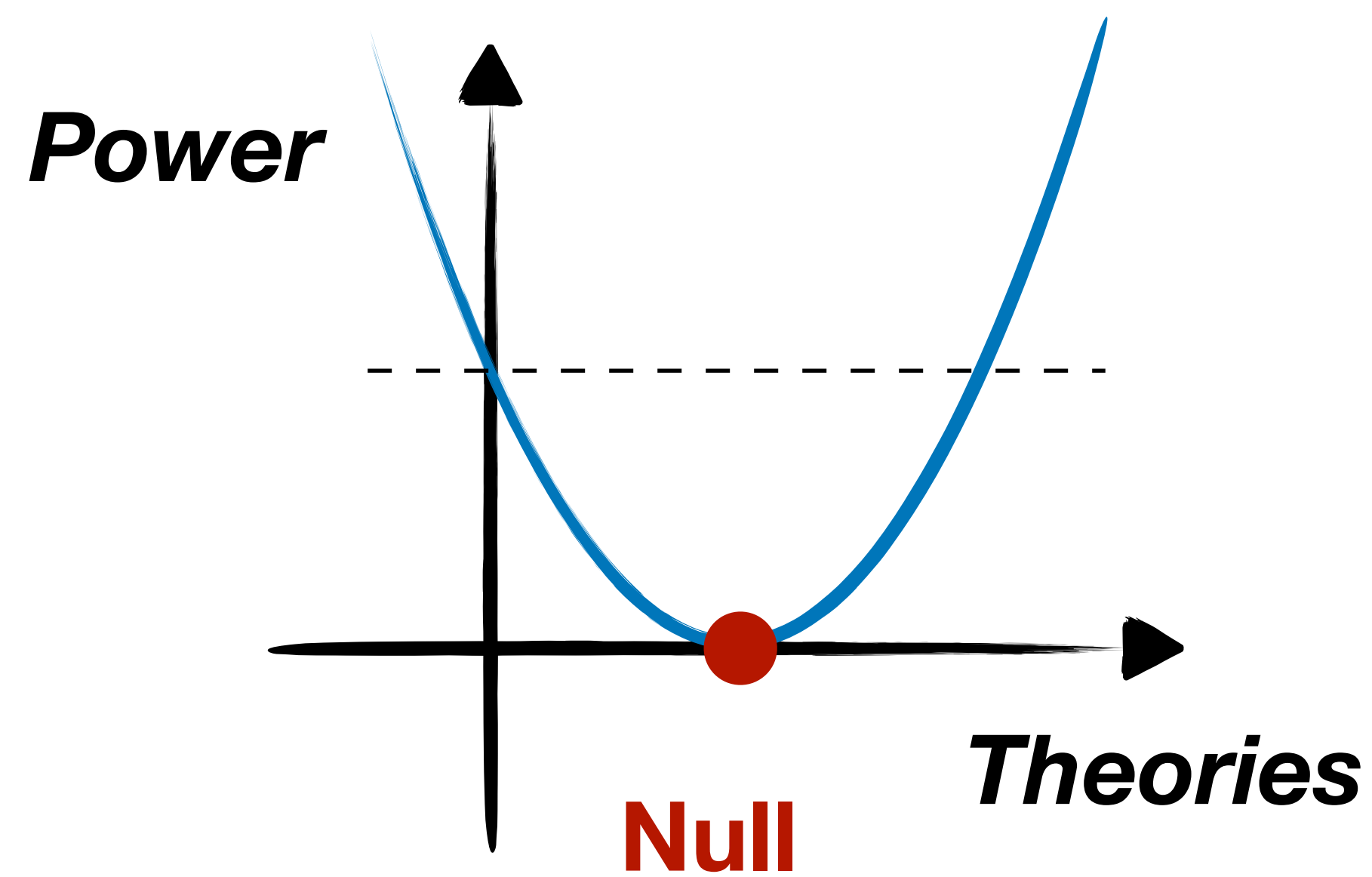
Anomaly Detection + Reinterpretation

The Story from Simplified Models Repeat: we give up power for any one specific theory: Effective in Multiple Testing Scenarios



Reminder of Motivation of Profile L'hood

Profile Likelihood is designed to have ~ roughly equal power for all alternatives that are “equally far away” from the null. It’s a specific choice and we could make other once



Anomaly Detection

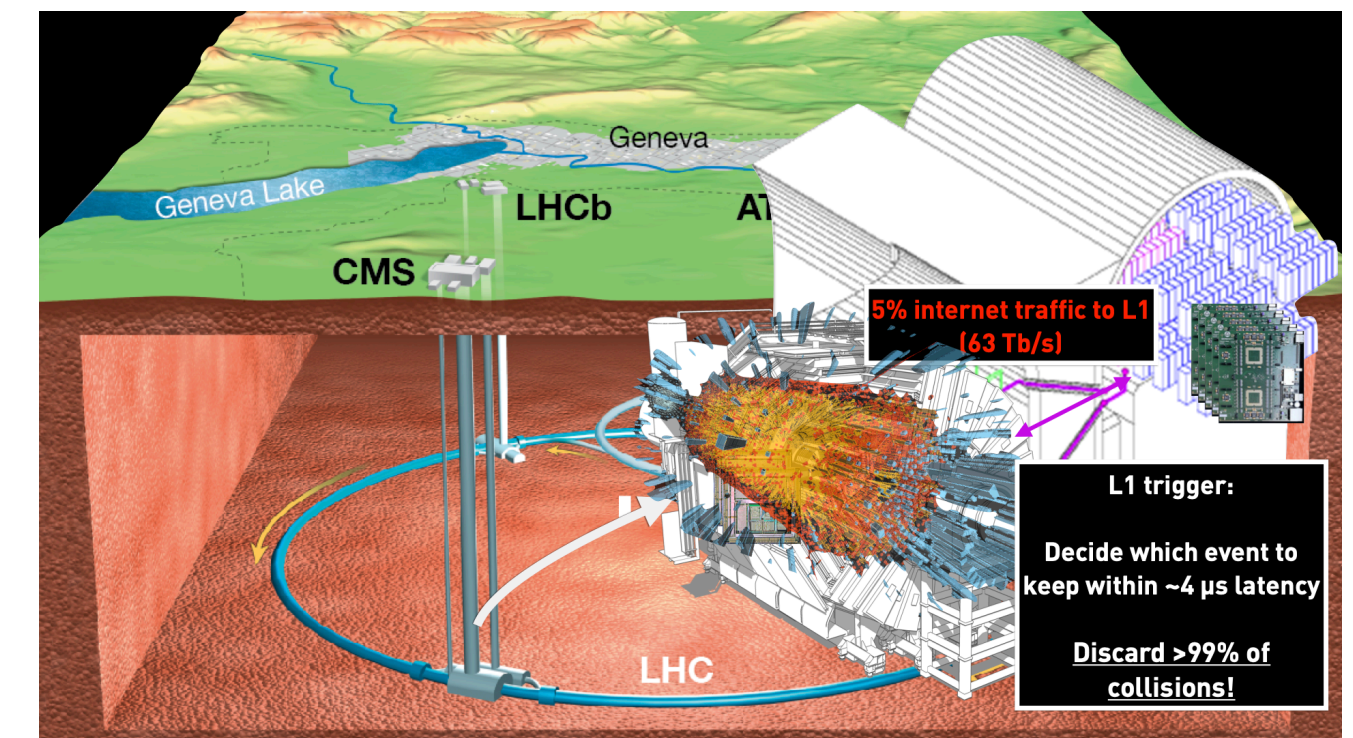
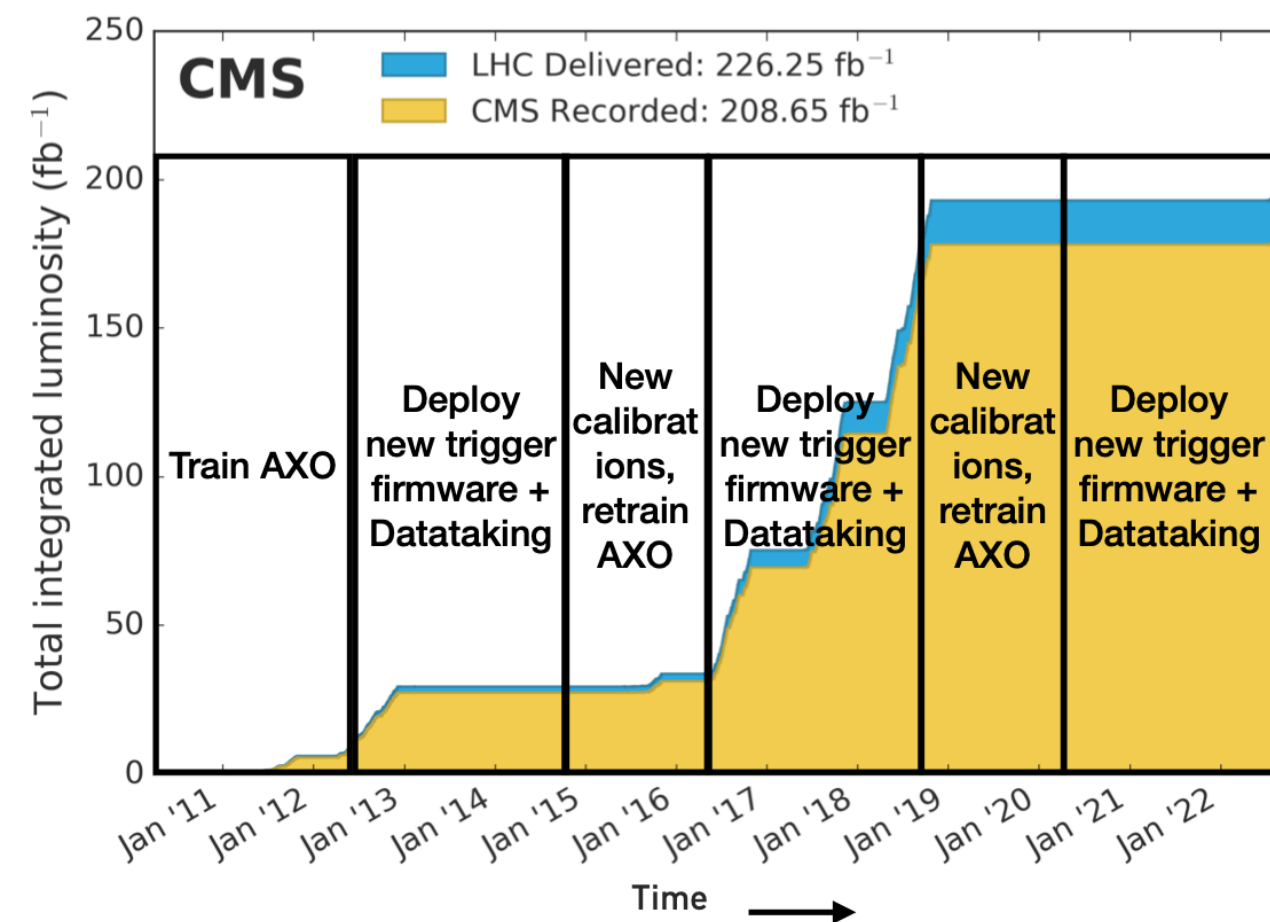
Statistics can help bring clarity to how we frame ML use-cases

To me a useful experimentalist framing
(and a Q I ask often)

You have 5 AD algorithms, but only
100 Hz of Bandwidth in the Trigger

How do you decide which one to deploy?

If the answer involves any reference
to performance on simulation, we
essentially made a choice in theory space



(Thea)

**Big jump in technical readiness in extreme environments:
Confident we can implement any answer we come up with**

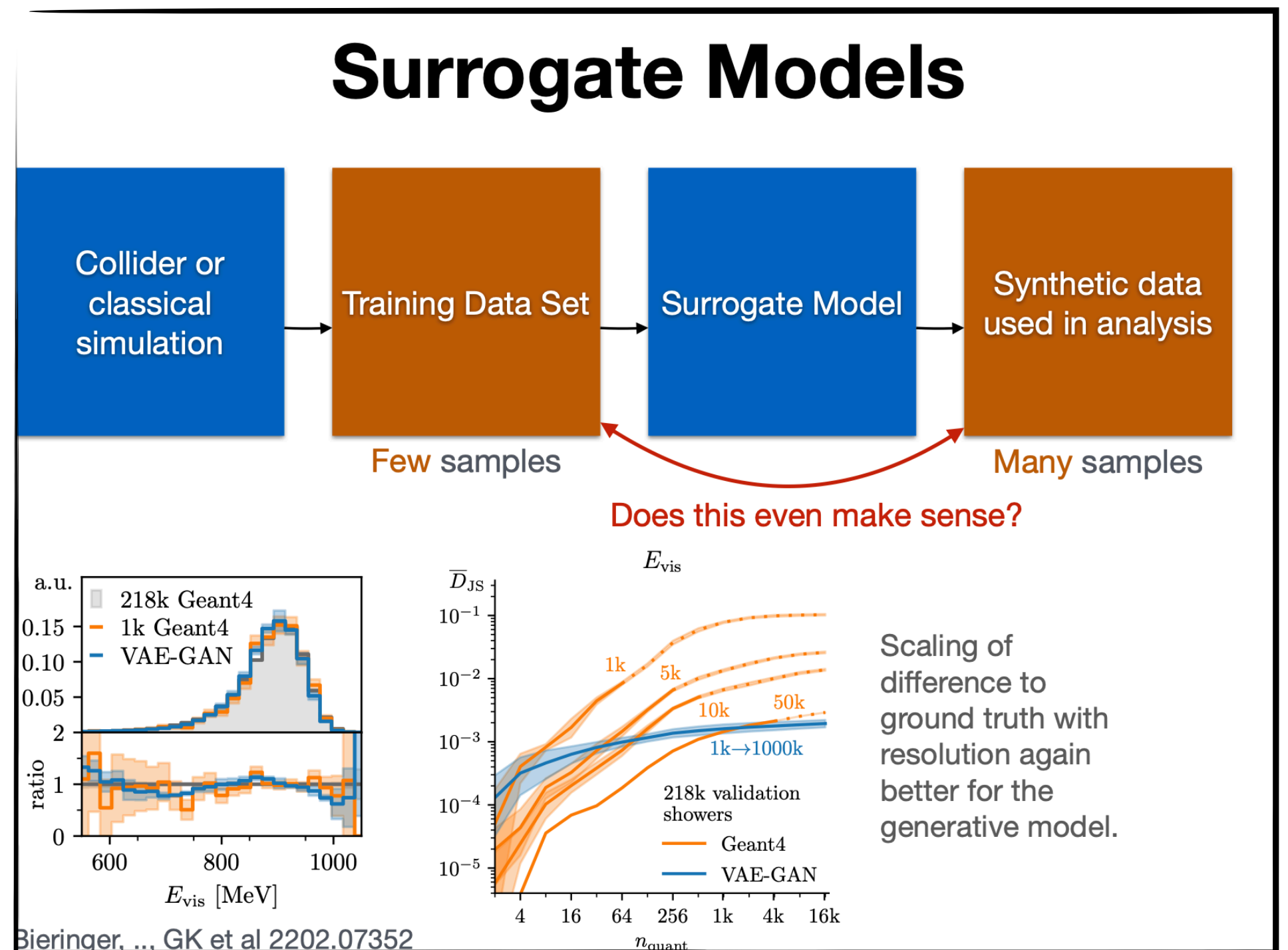
Amplification?

Can you really create information out of nothing? No, so what's going on?

2. Generative models for many fast simulations from few full ones.

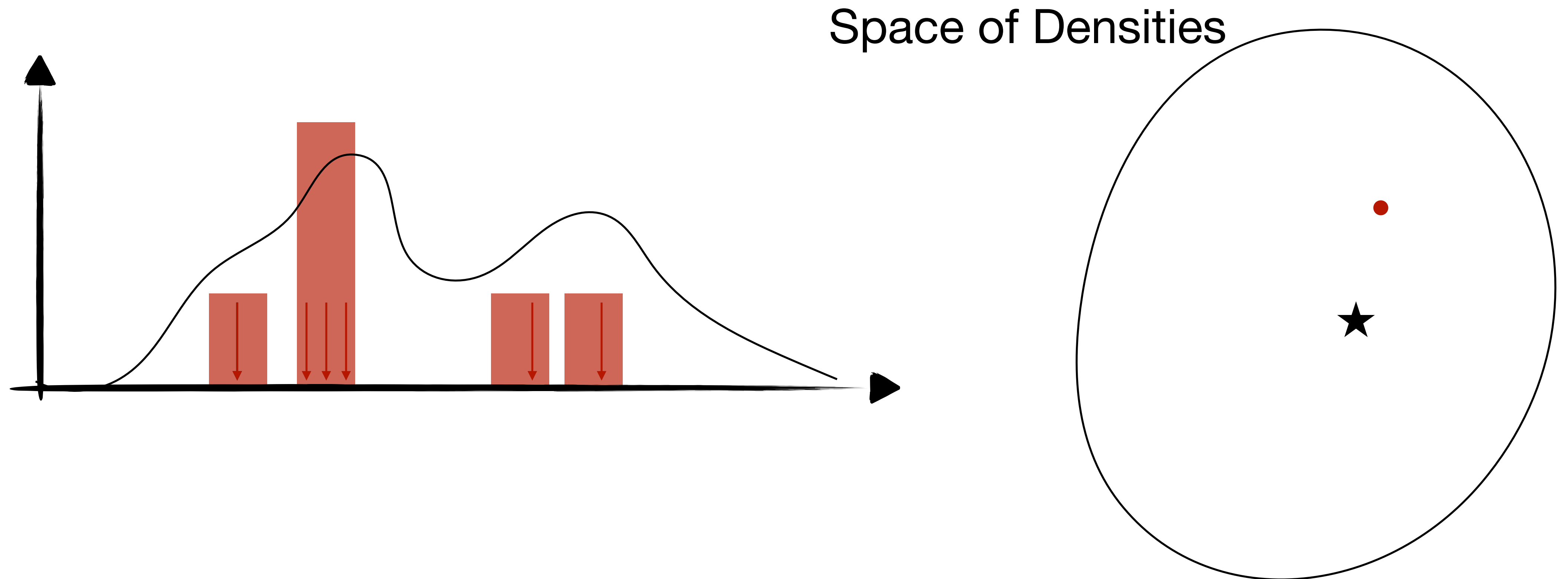
What do we gain by using ML to learn from a few fully simulated events how to generate a large number of events quickly?

For example, we believe some (x,y) data should lie on a straight line and are interested in the gradient. With difficulty we do a full simulation of 4 (x,y) points. Our ML procedure learns from these 4 points how to generate new data, and produces 1000 new (x,y) pairs. The statistical uncertainty on the gradient is greatly reduced, but there is a large systematic related to the particular choice of the initial 4 points. Is there anything that we can usefully learn from the larger sample? Are generative methods different from this in some subtle way?



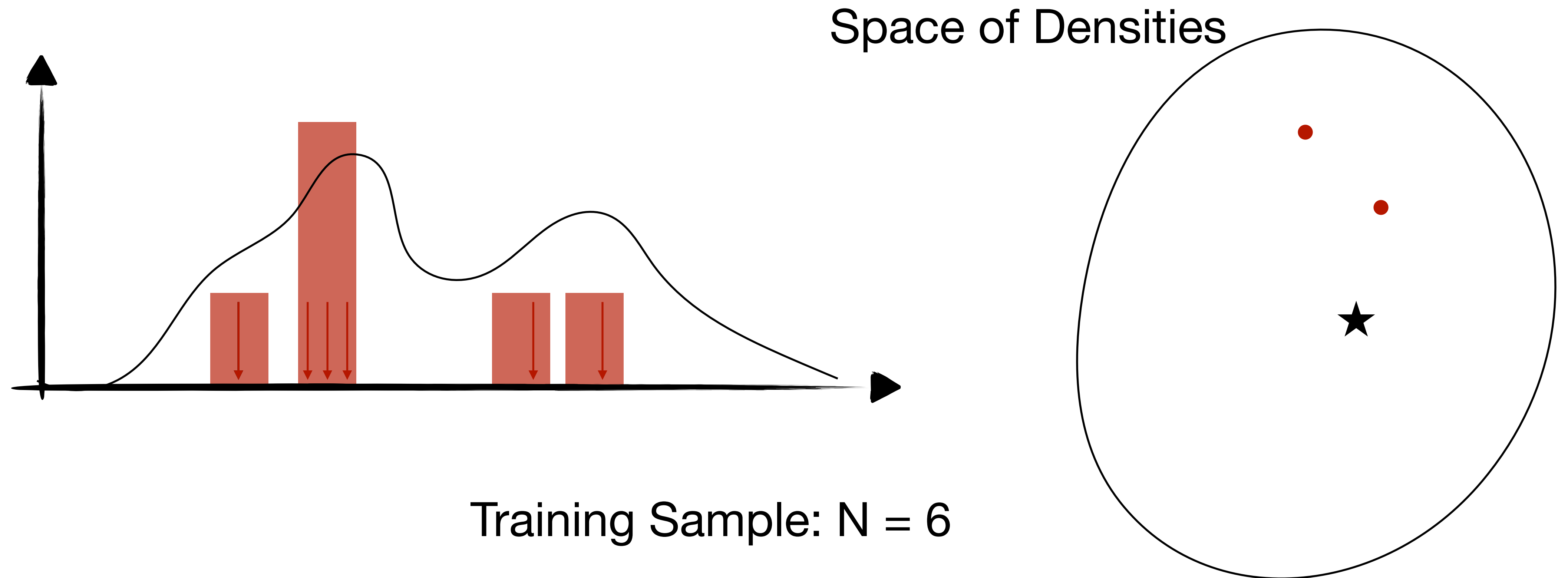
Amplification?

Can you really create information out of nothing? No, so what's going on?



Amplification?

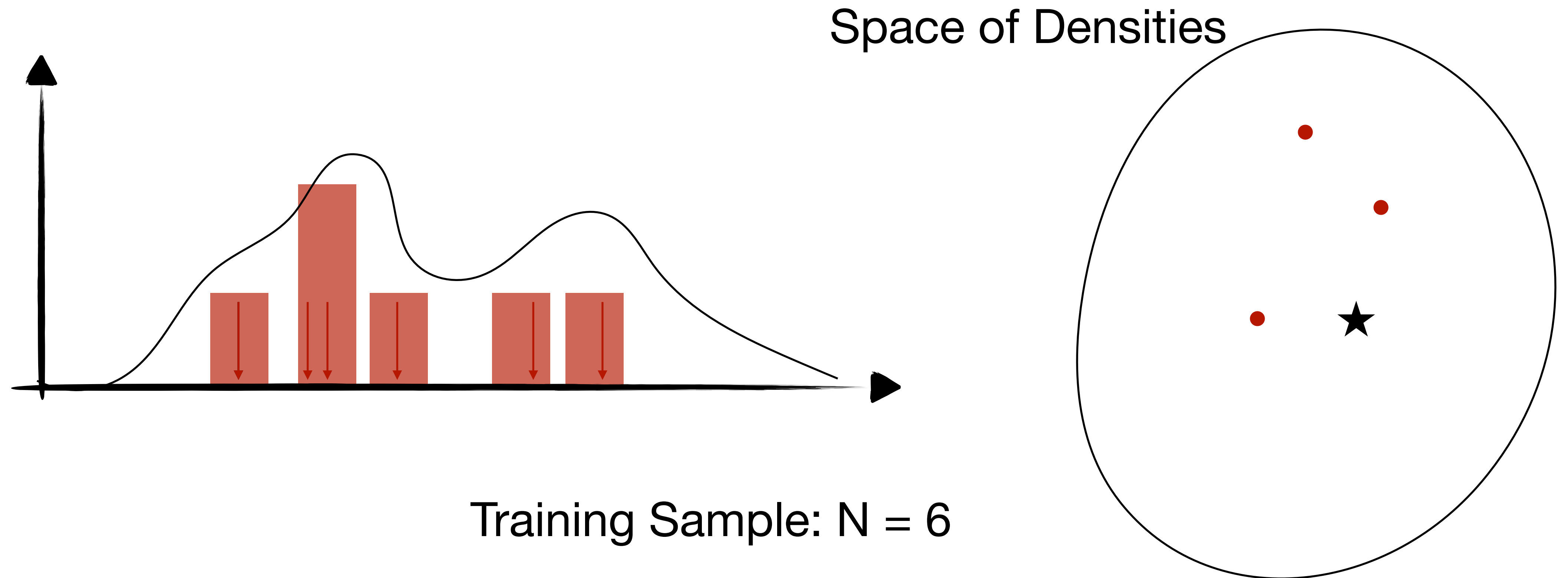
Can you really create information out of nothing? No, so what's going on?



A simple density estimation technique: histogram

Amplification?

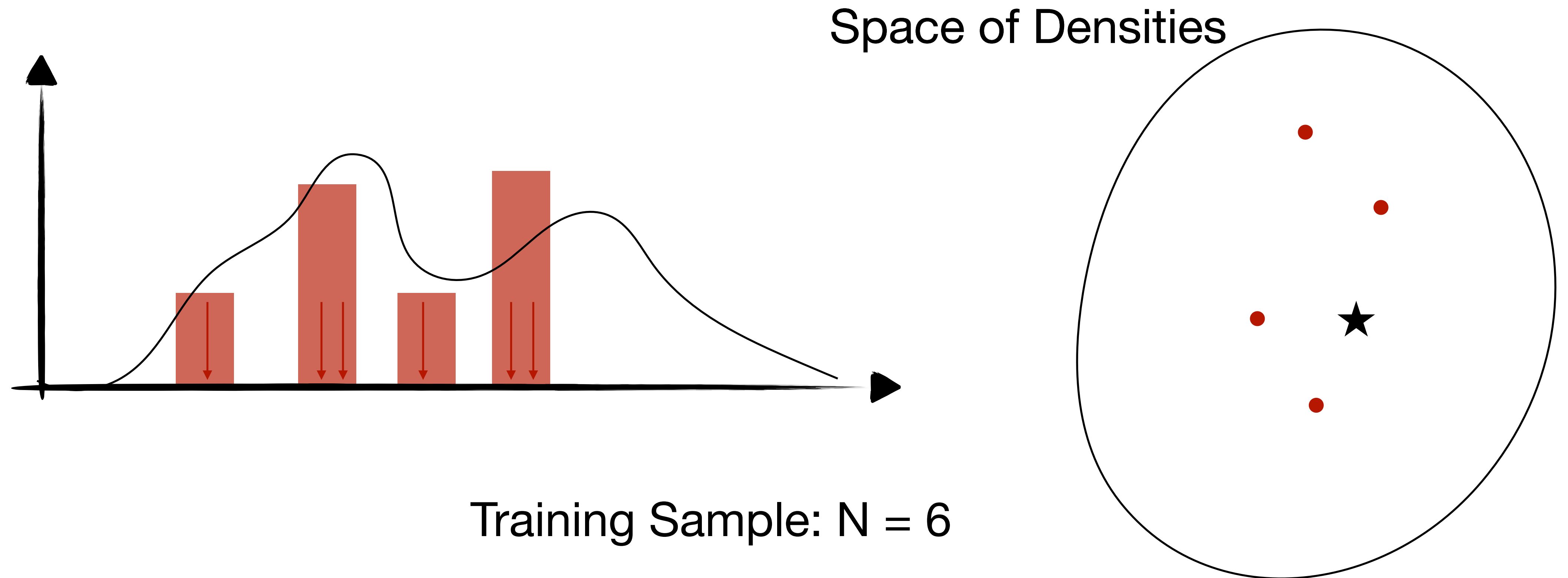
Can you really create information out of nothing? No, so what's going on?



A simple density estimation technique: histogram

Amplification?

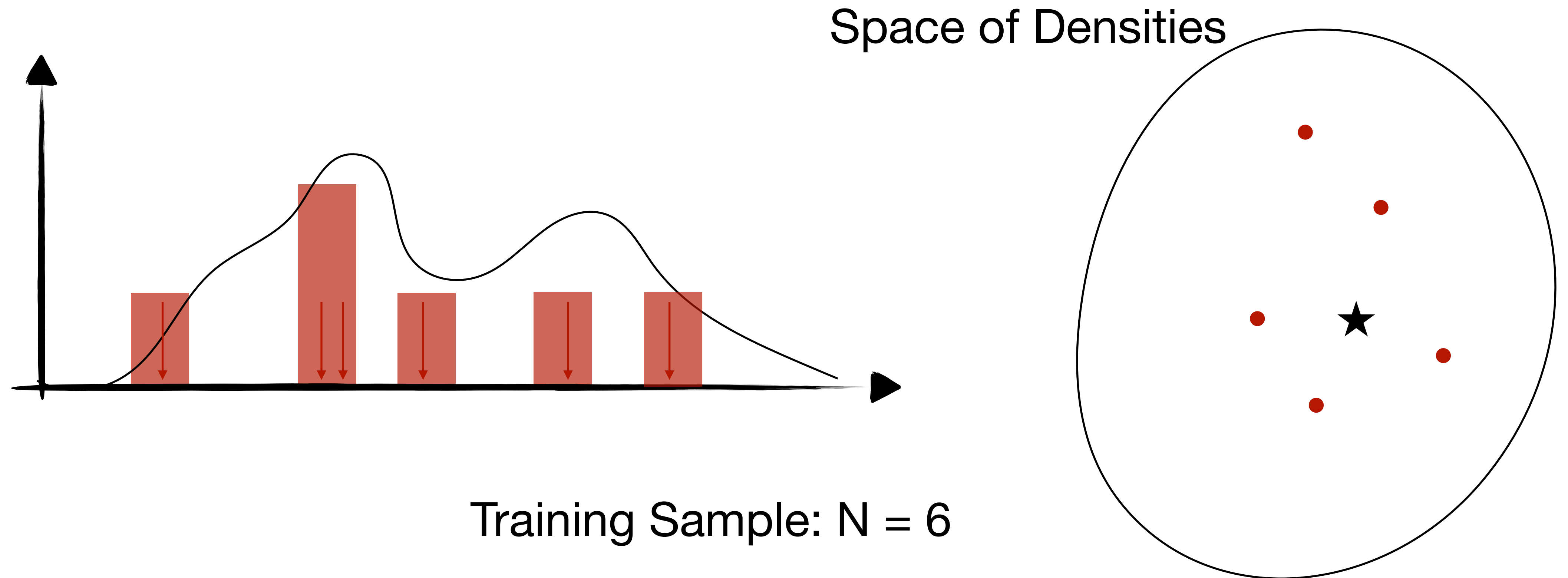
Can you really create information out of nothing? No, so what's going on?



A simple density estimation technique: histogram

Amplification?

Can you really create information out of nothing? No, so what's going on?



A simple density estimation technique: histogram

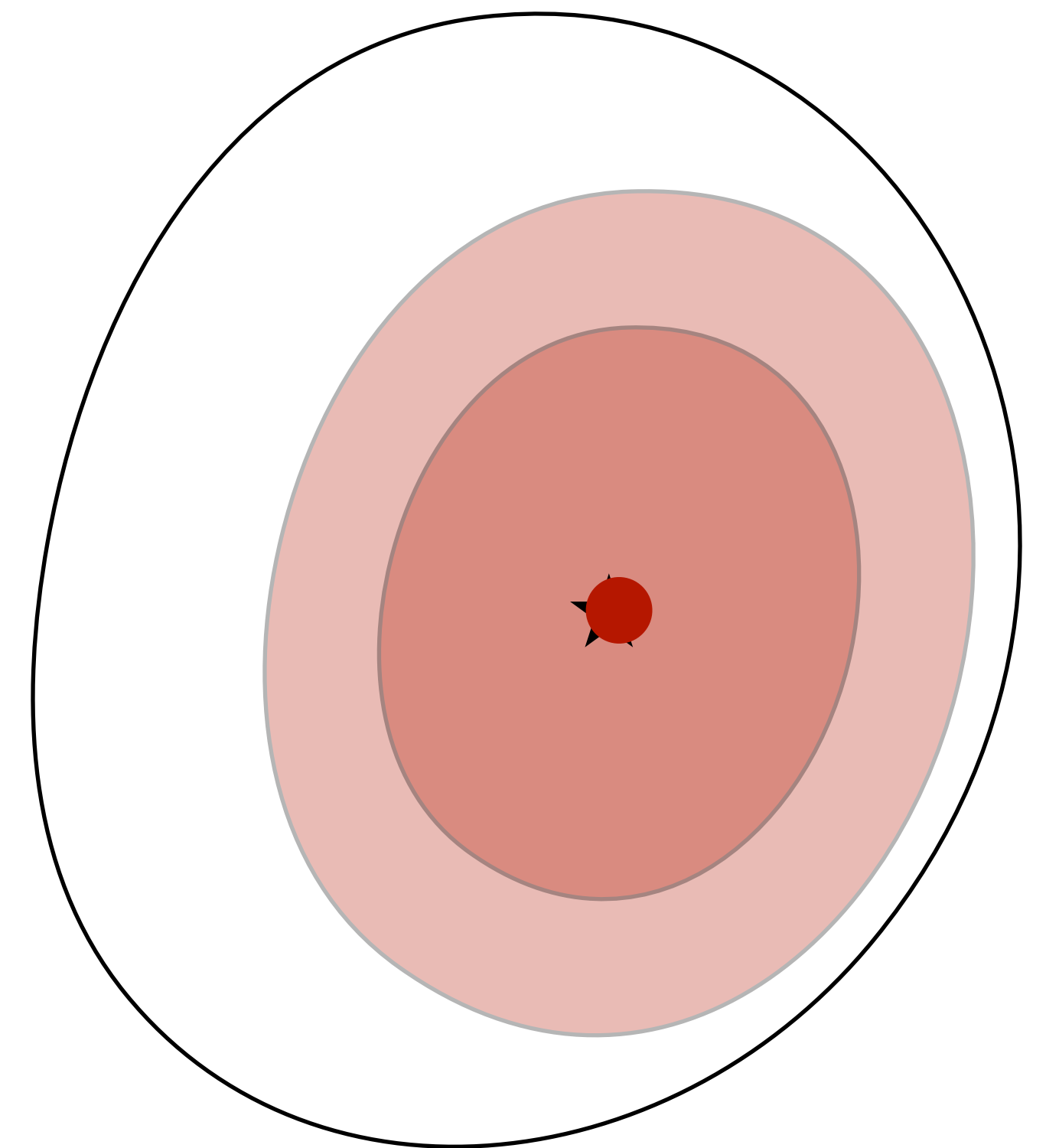
Amplification?

If we repeat this many times we can see how this density estimator fares

Unbiased: We sample from the real thing

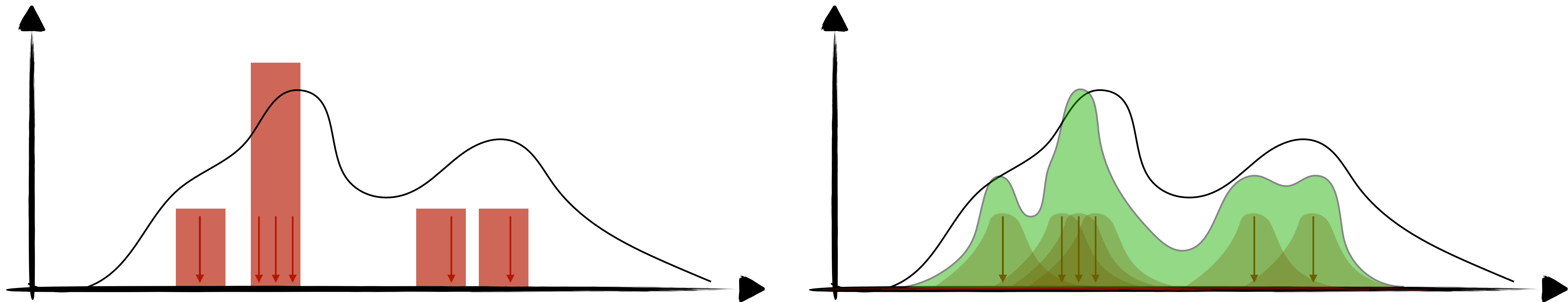
High-Variance: with few samples, the estimate is all over the place from sampling variance

What we usually call “MC Stat Error”



Amplification?

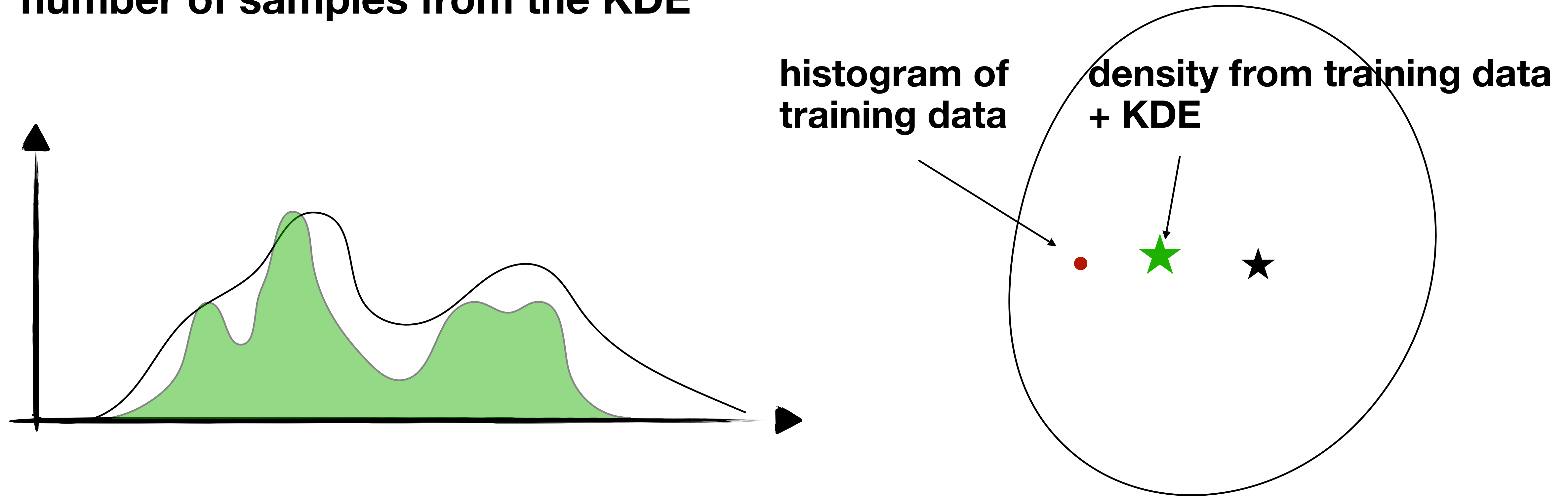
Nobody forces us to use mere histograms. A frequent idea that comes up: couldn't we do some other type of density estimate?



**KDE of 6 samples: This is a density model and generative model!
(like a pre-historic normalizing flow)**

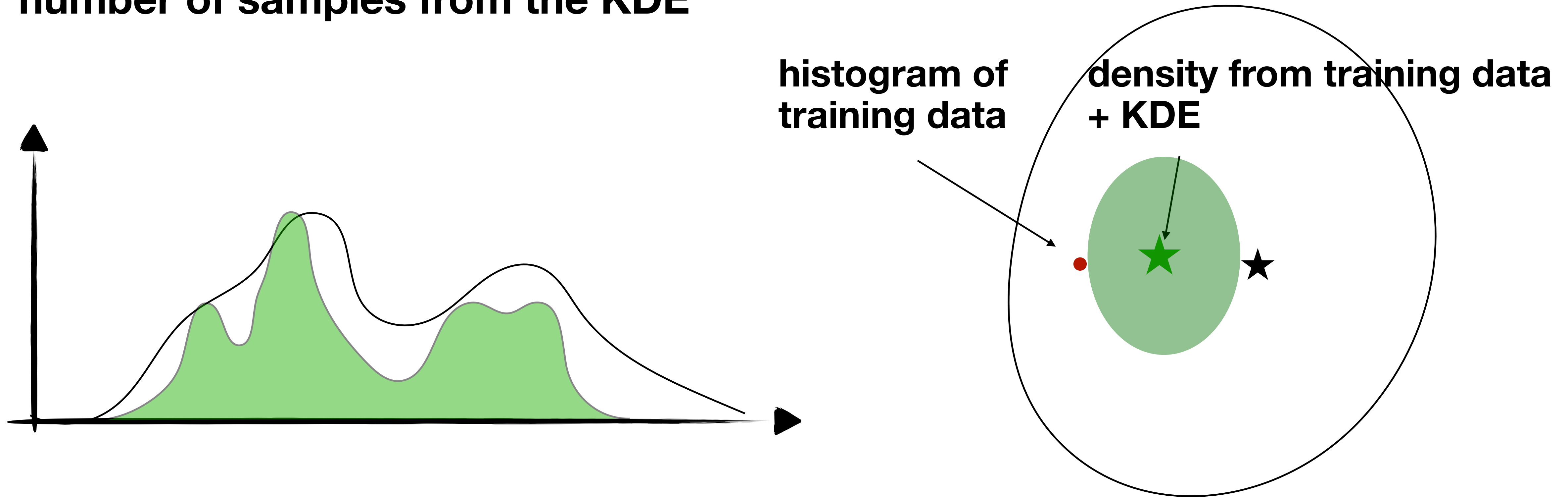
Amplification?

You can now sample quickly (call it “fast simulation”) basically an infinite number of samples from the KDE



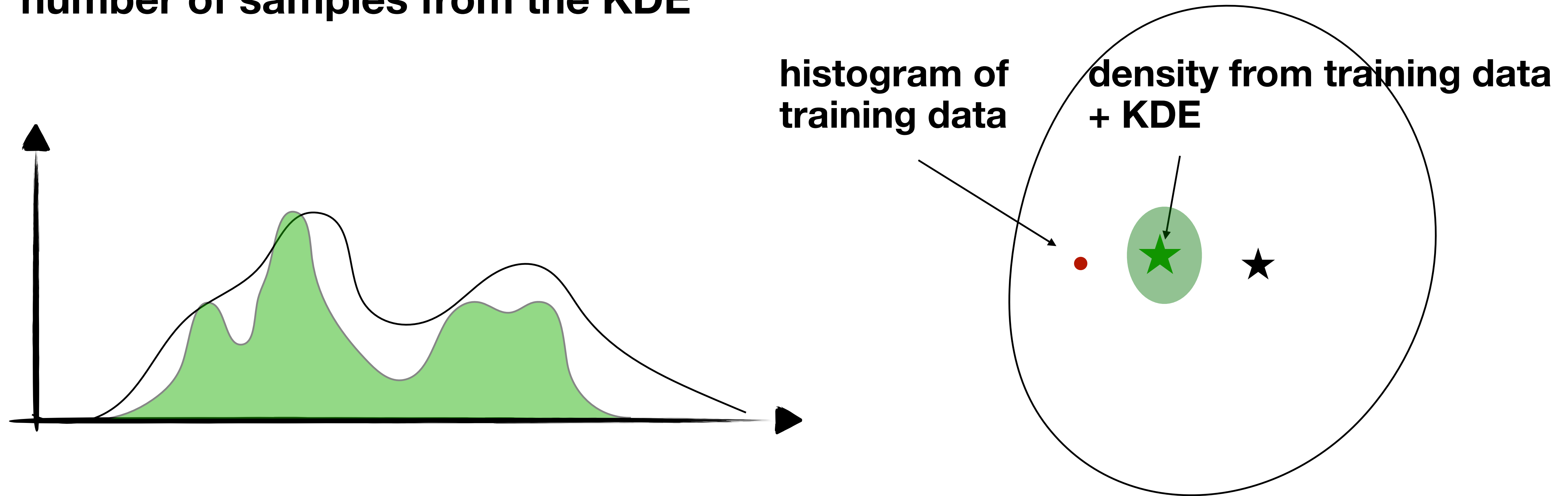
Amplification?

You can now sample quickly (call it “fast simulation”) basically an infinite number of samples from the KDE



Amplification?

You can now sample quickly (call it “fast simulation”) basically an infinite number of samples from the KDE

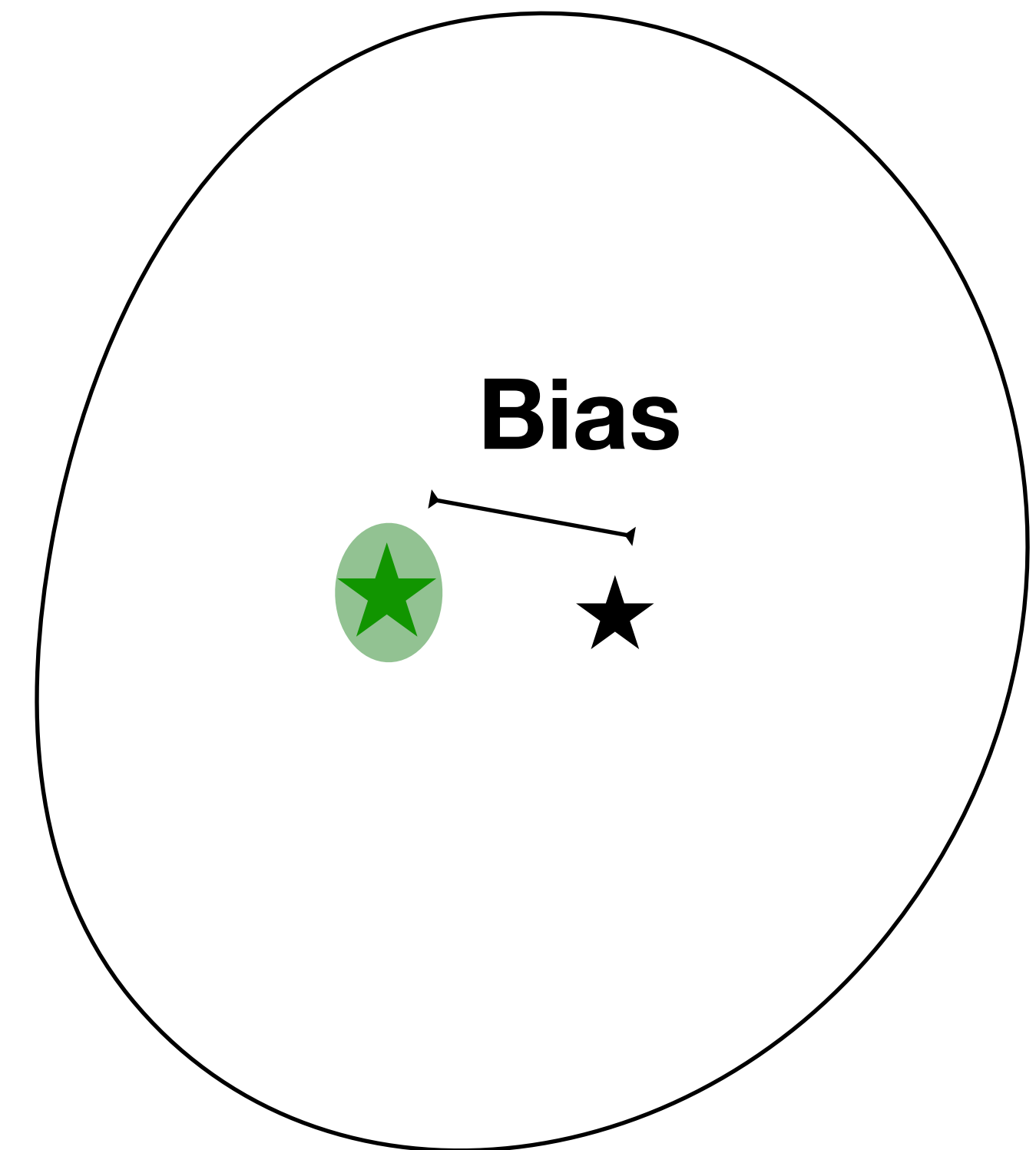


Amplification?

How do we characterize this estimator?

Biased: this depends basically depends on which training data + hyperparameters (bandwidth, etc)

Zero-Variance: if we can draw an infinite amount from this we can make it arbitrarily small

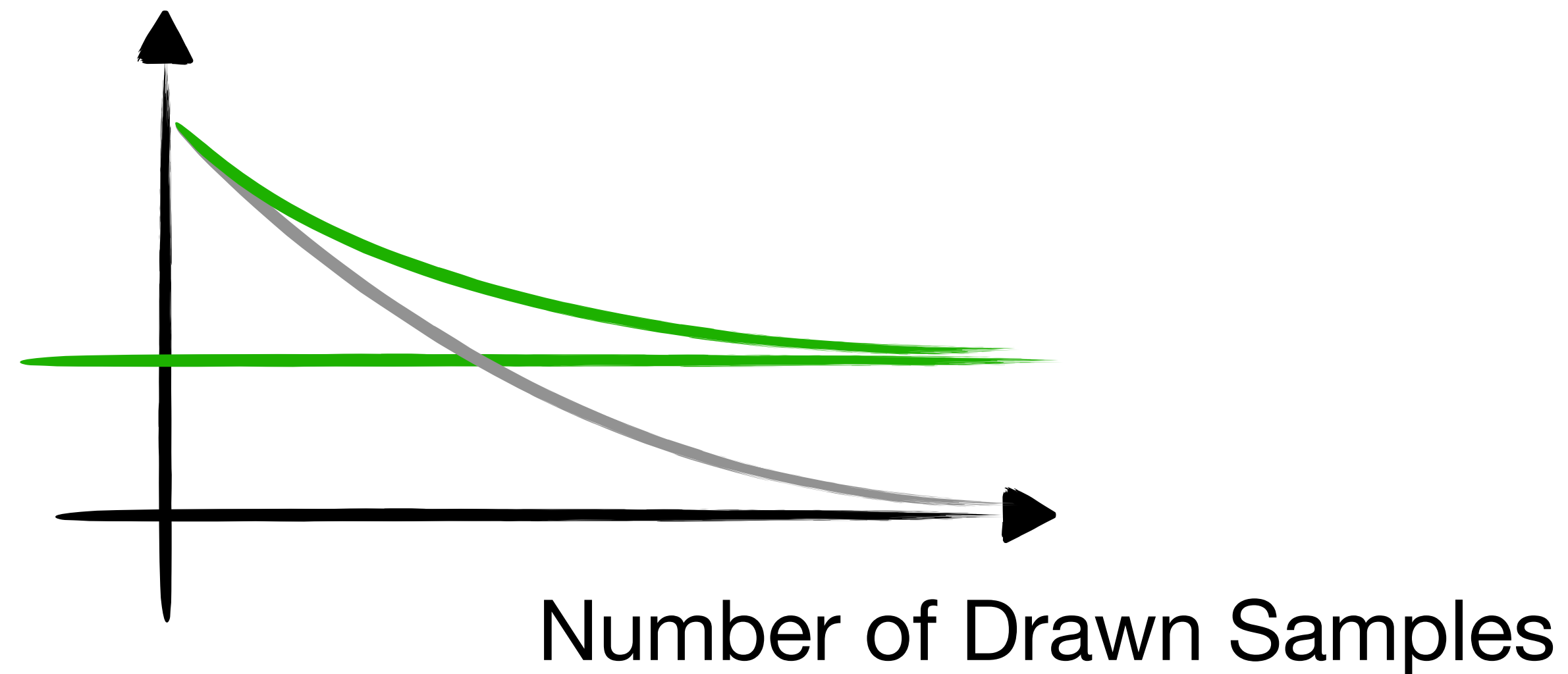


Amplification?

So how do the two compare?

sampling from Geant
converges to true zero MSE

sampling from generative model
converges to its inherent bias



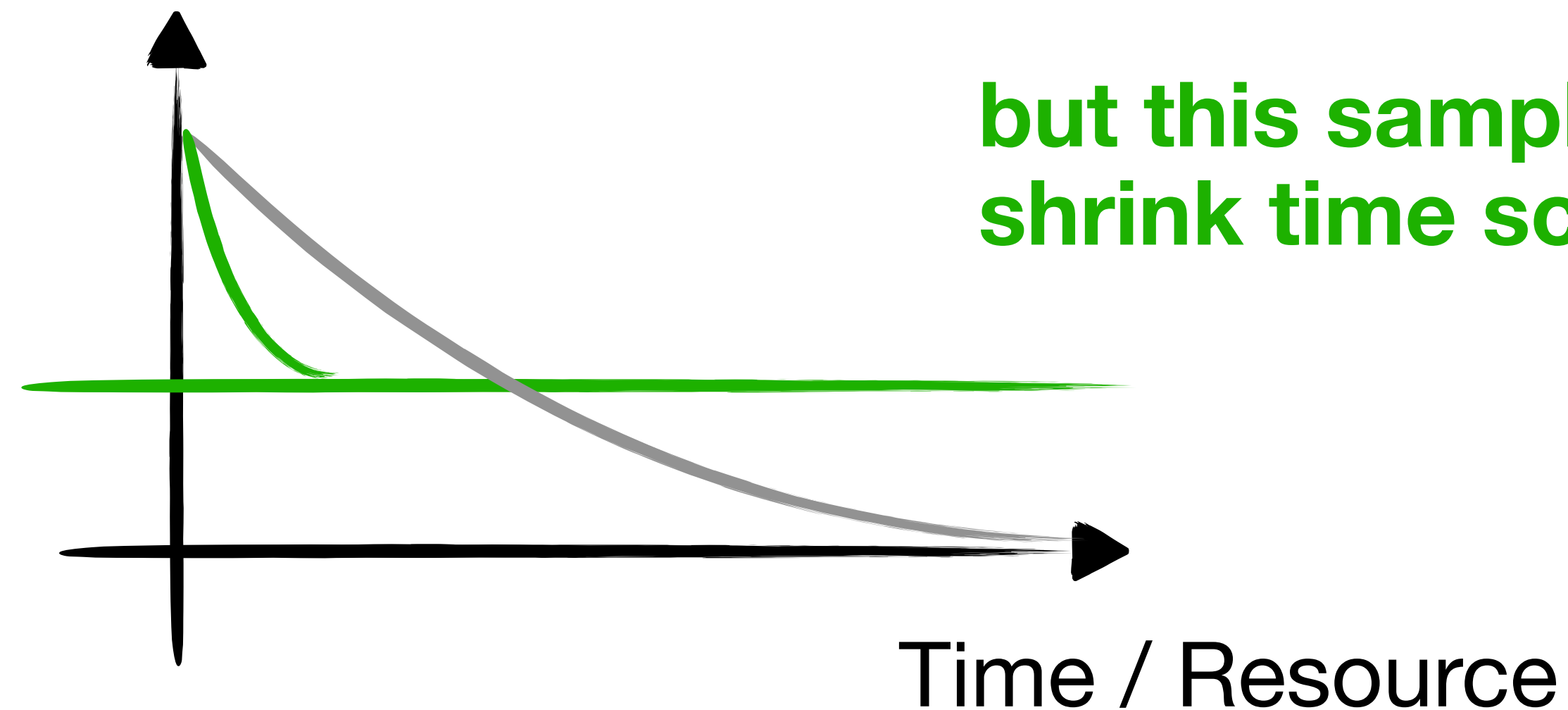
Amplification?

So how do the two compare?

sampling from Geant
converges to true zero MSE

sampling from generative model
converges to its inherent bias

**but this sampling is much faster!
shrink time scale!**



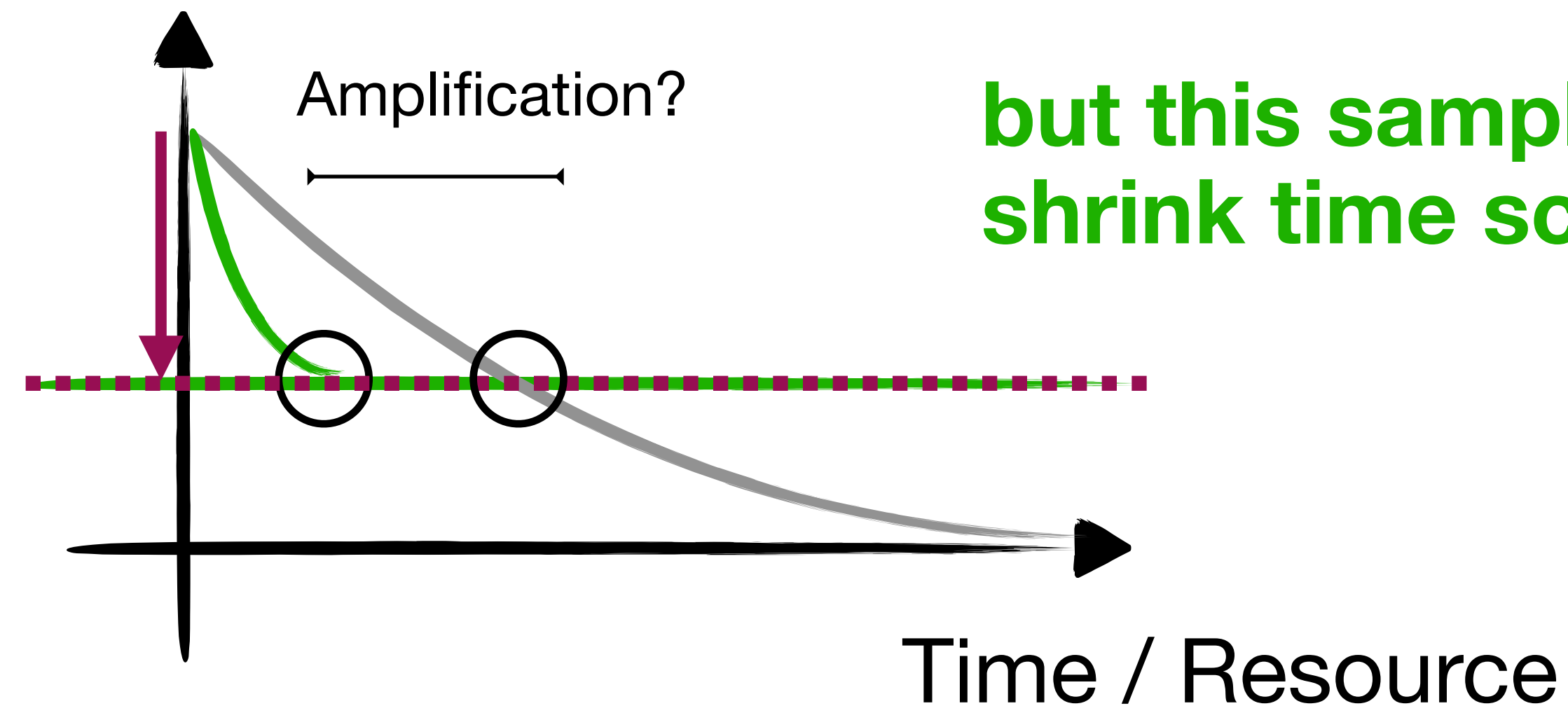
Amplification?

So how do the two compare?

sampling from Geant
converges to true zero MSE

sampling from generative model
converges to its inherent bias

With a real density model
I don't even need to sample
(Flow, KDE, ...)



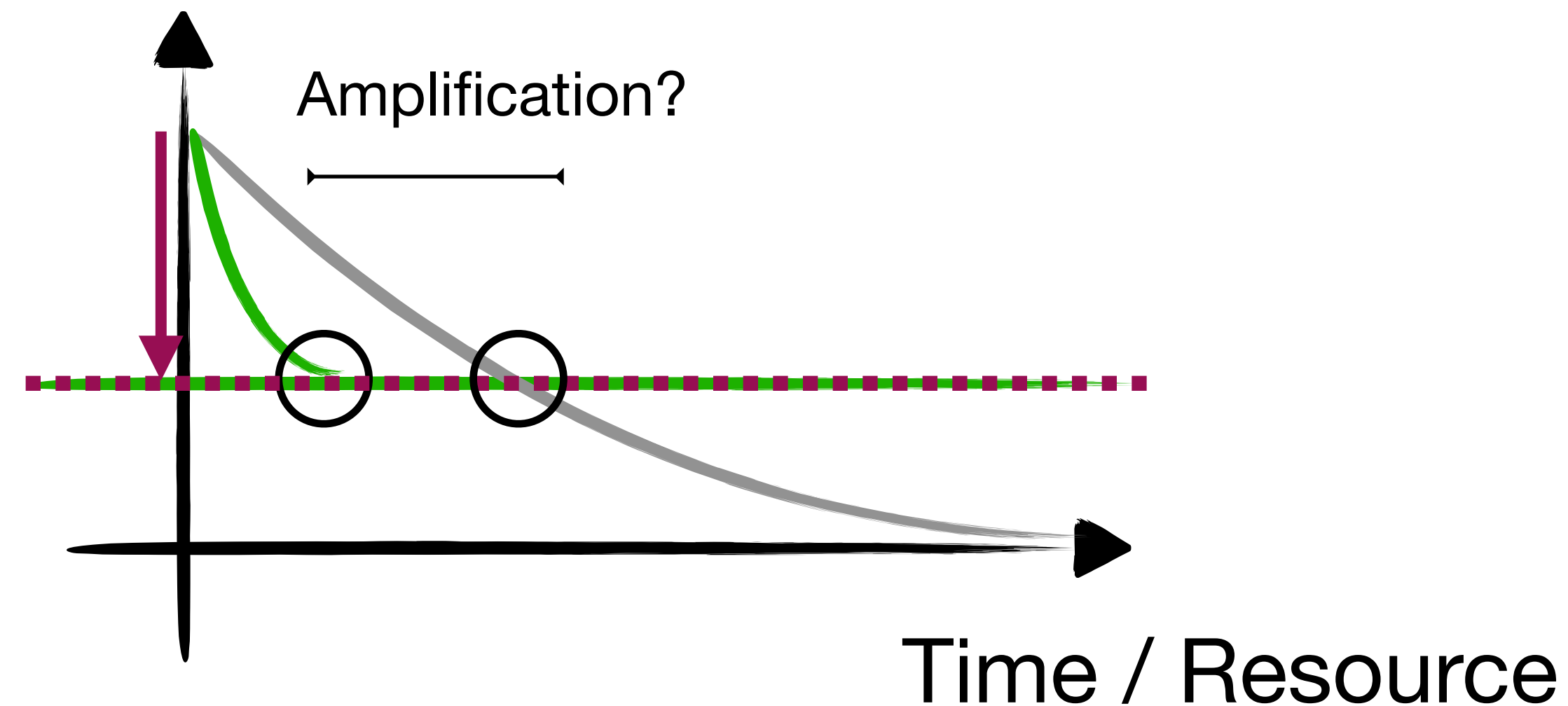
**but this sampling is much faster!
shrink time scale!**

You can reach same "MSE" in shorter (or zero) time...

Amplification?

In a way this just the bias-variance tradeoff. We trade off

- a zero-bias ~high-variance strategy (samples from G4 + naive histograms)
- biased, ~zero-variance strategy (density estimate trained on few samples e.g. KDE, Flows, GANs, ...)



You can reach same “MSE” in shorter (or zero) time...

Amplification?

In a way this just the bias-variance tradeoff. We trade off

- a zero-bias ~high-variance strategy (samples from G4 + naive histograms)
- biased, ~zero-variance strategy (density estimate trained on few samples e.g. KDE, Flows, GANs, ...)

But it's a bit apples to oranges

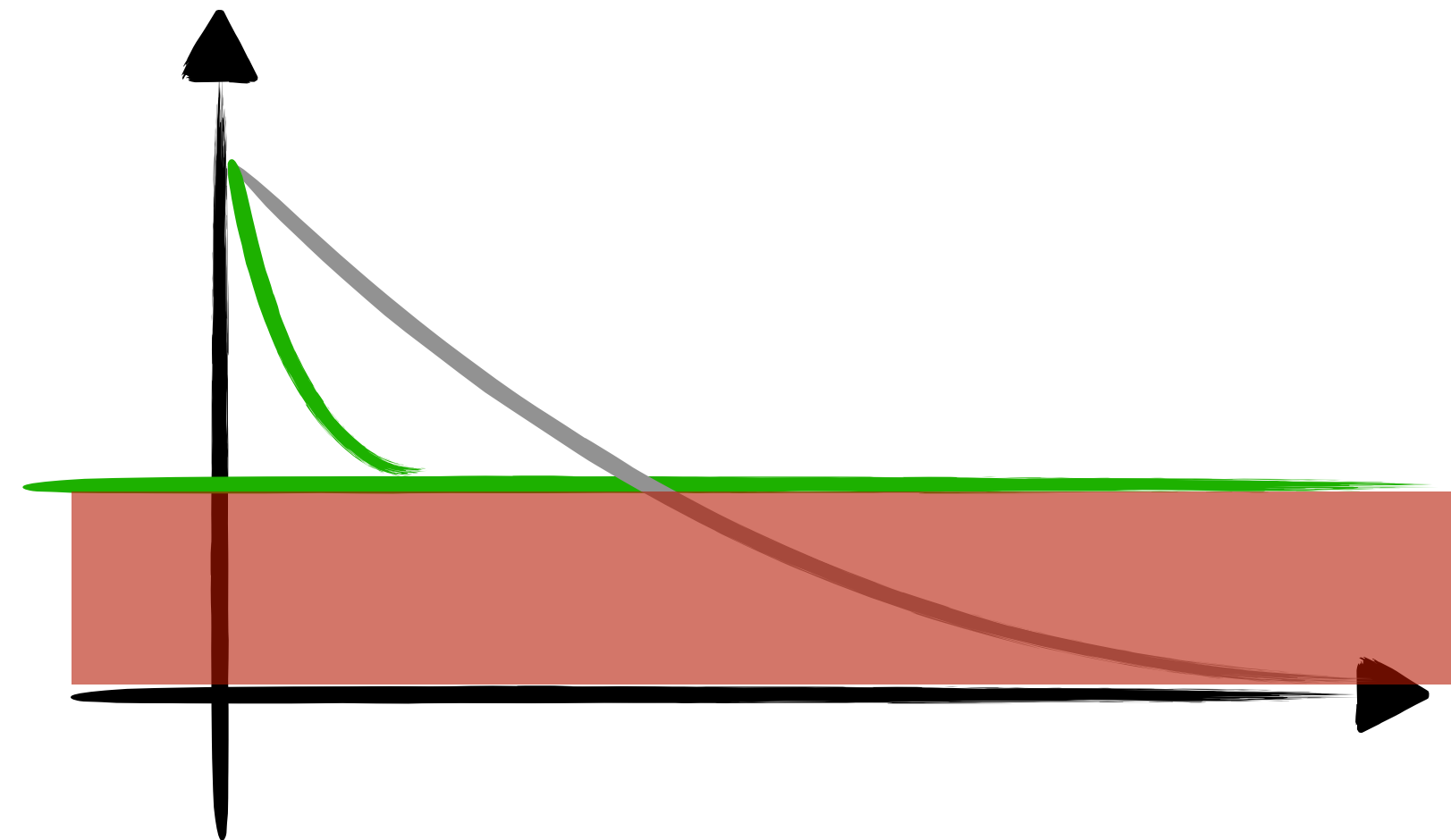


Amplification only happens if you use a fancy density estimate on few samples to compare to a dumb density estimate with many samples

Amplification?

This is a one-time gain. You can benefit from regularization once. But **there is no general rule where 1 GenAI event \approx 1/100 Geant Event** (i.e. if GenAI were 100x faster you would gain)

With a fixed generative model, you will never reach the true distribution no matter how many samples you draw. With Geant4 you will.



Interpretability & Control

We don't only want to tame us but also the networks - or "understand" them.

But what are they learning??

Why might we want ML to be "Interpretable"? **(Jesse)**
Or explainable, trustworthy, safe, robust, aligned, helpful, transparent, ...

Scientific Reasons:

- Could be working in **non-asymptotic** regime
- Training data might be **biased** in some way
- Result could depend on **poorly modeled** features
- Limited ability to perform independent **validation**
- Need for compact **symbolic** expressions
- Desire to **generalize** away from specific context
- ...

Sociological Reasons:

- Skeptical** of algorithmic/statistical/computational reasoning
- Need to explain decisions to external **stakeholders**
- Desire to **manage risks** from unforeseen outcomes
- ...

*All valid reasons, but suggest **imperfect specification** of our initial goals!*

Jesse Thaler (MIT, IAIFI) — Interpretable Machine Learning for Particle Physics 15



It's a lot about retaining control in an uncertain world, when you don't trust the process

Not new: "Bayesian Workflow" / Iterative Model Building etc is a lot about understanding a system. If we'd trust the model / process we would just run MCMC and be done

Interpretability & Control

When you trust the process & underlying tools we're fine w/o interpretability

Example: Likelihood-Ratio Estimation when you have a simulator you trust

“What is the machine learning?”

For this **loss function**, an estimate of the **likelihood ratio** derived from **sampled data** and regularized by the **network architecture** and **training paradigm**

“But I want to understand what it has learned!”

Do you really expect the **likelihood ratio** to take on a particularly **nice functional form**?

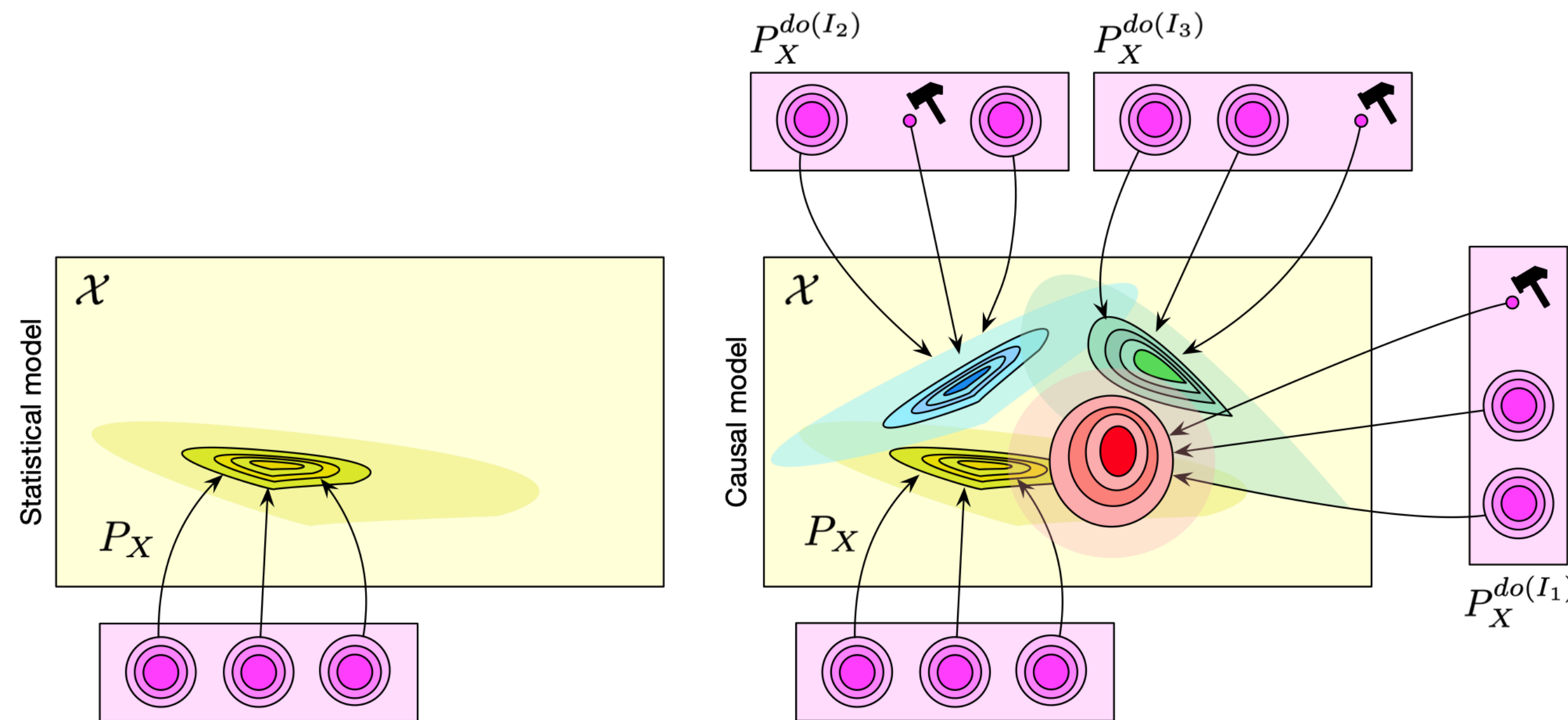
N.B. QFT calculations often involve special functions that have no elementary representation

“ ... ”

Performance under Intervention

What we do want to understand: how does ML react to distribution shift e.g. from interventions (Sherpa → Herwig, etc)

A lot of literature (Causal Inference) that we don't use



Does a New Drug Improve Health Outcomes?

Causal Inference:

- Split subjects: treatment ($A = 1$) and control ($A = 0$) group.
- What if treatment group differs systematically from control group, e.g., in terms of x .

$$p_{\text{treatment}}(x) \stackrel{?}{=} p_{\text{control}}(x)$$

- Randomization is the gold standard, not always possible.

Propensity Scores:

- Rosenbaum and Rubin (1983) define propensity scores:

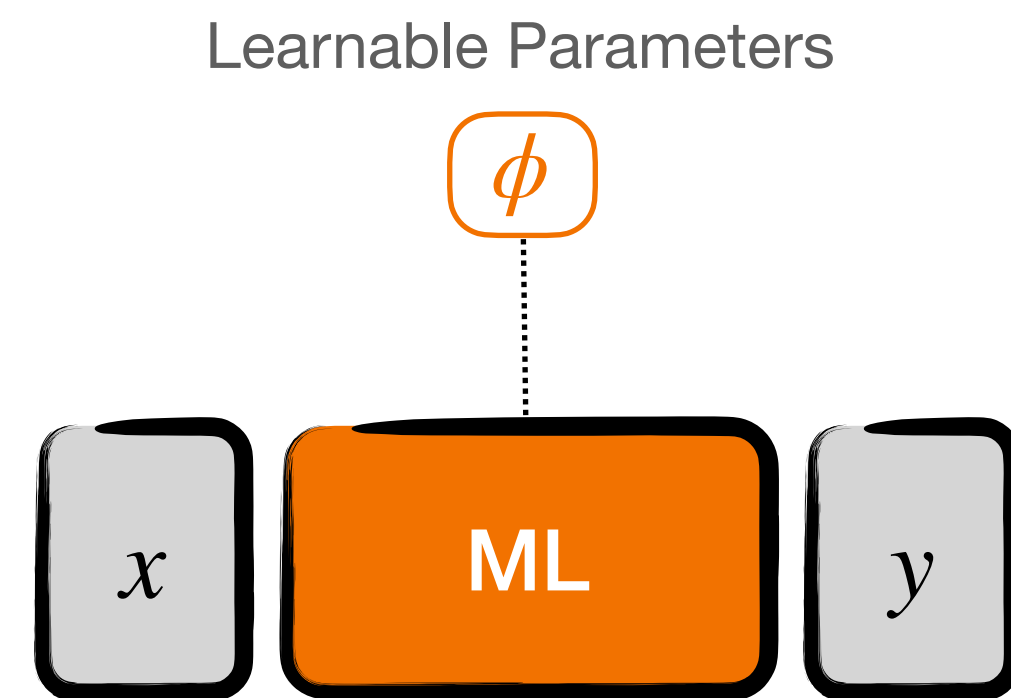
$$e(x) = P(A = 1|x).$$

- Demonstrate that $e(x)$ is a balancing score:

$$p_{\text{treatment}}(x|e(x)) = p_{\text{control}}(x|e(x)).$$

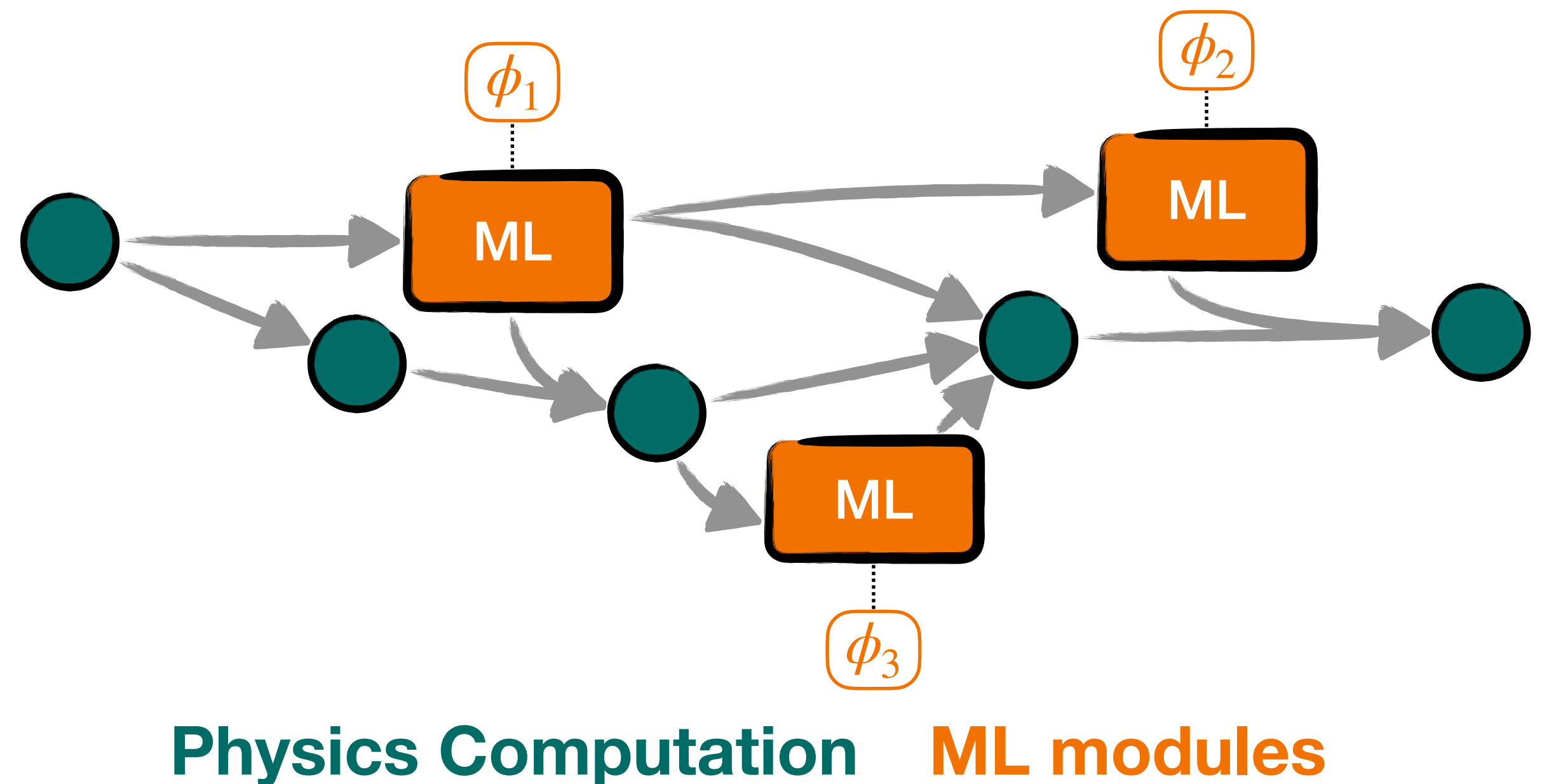
Inductive Bias: A Physicists' Love Affair

Two ways to think about it



$$f_{\phi}(r_x(g)x) = r_y(g)f_{\phi}(x)$$

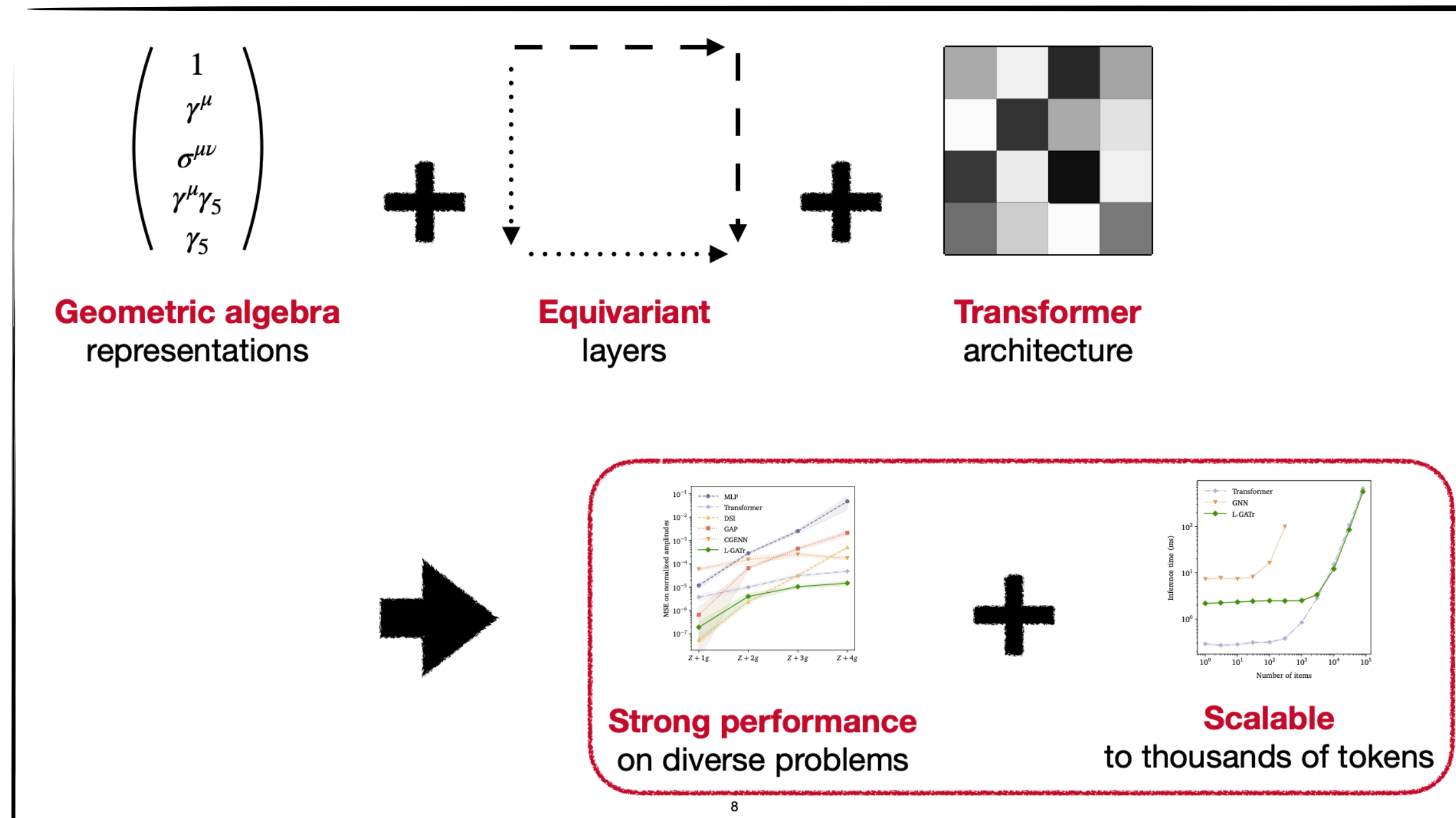
Statics
(enforce symmetries etc)



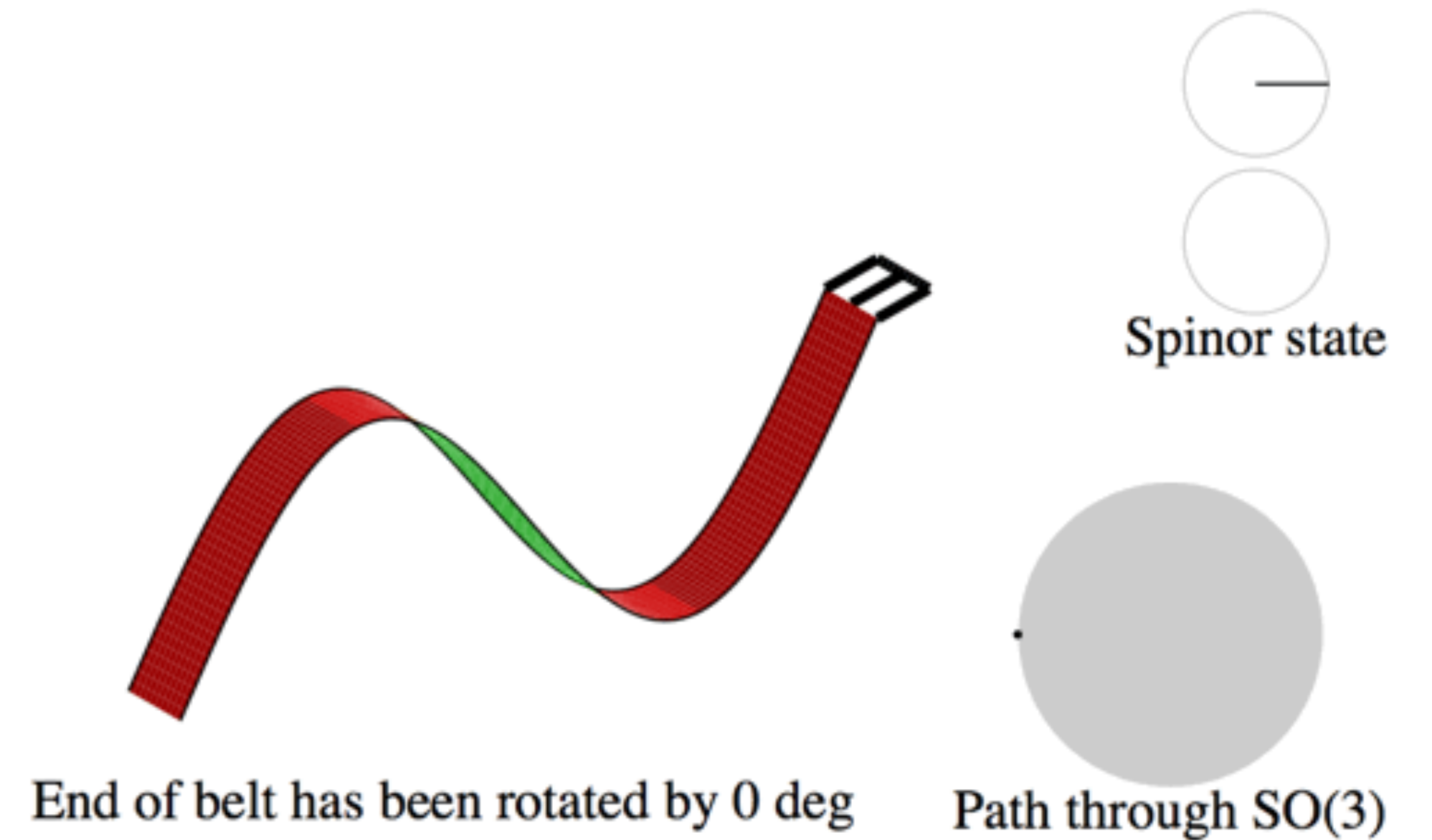
Physics Computation ML modules

Dynamics
mix physics + ML workflow,
keep control over data flow

Inductive Bias: A Physicists' Love Affair



First, give the belt two full twists.



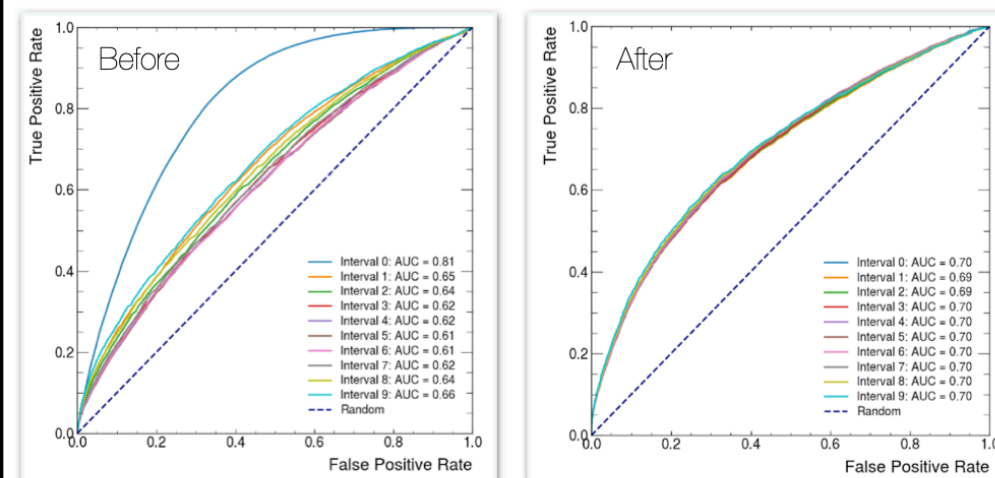
Soft Symmetries

Another version of Control / Interpretability: Force Behavior onto ML

Enforcing fairness on a classifier

ROC Split (during training of ML model)

- Iterative algorithm trains classifier to satisfy EOP
 - Divide Mass into bins and determine AUC
 - Sample events from with $p_i = 2(1-AUC_i)$
 - Train model on new sample and repeat



Post integration methods

- Train classifier R with mass as input
- Integrate out mass

$$R_p(x) = \int R(M_{\mu\mu}, x) P(M_{\mu\mu}) dM_{\mu\mu}$$

- Effective, but strong impact on performance
- Helpful, if the correlation is small from the beginning
- See Purvasha's talk for more sophisticated methods!

15

Nikhef

(Oliver)

Learning to pivot with adversarial networks

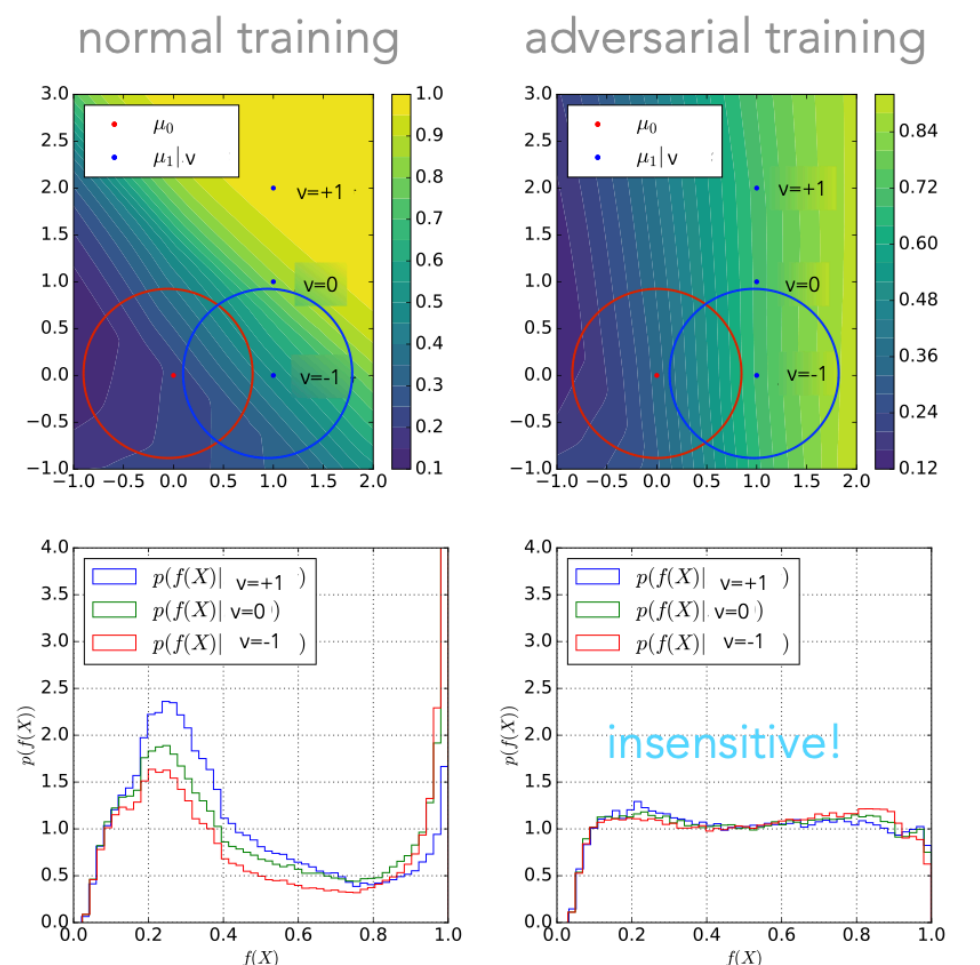
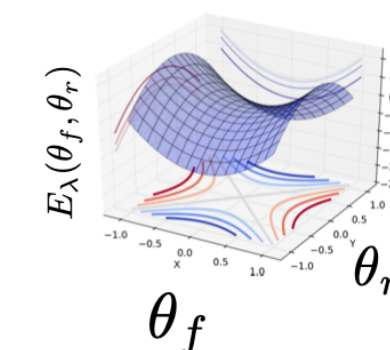
G. Louppe, M. Kagan, K. Cranmer, arXiv:1611.01046

Typically classifier $f(x)$ trained to minimize loss \mathcal{L}_f .

- want classifier output to be insensitive to systematics (nuisance parameter \mathbf{v})
- introduce an **adversary** \mathbf{r} that tries to predict \mathbf{v} based on f .
- setup as a minimax game:

$$\hat{\theta}_f, \hat{\theta}_r = \arg \min_{\theta_f} \max_{\theta_r} E(\theta_f, \theta_r).$$

$$E_\lambda(\theta_f, \theta_r) = \mathcal{L}_f(\theta_f) - \lambda \mathcal{L}_r(\theta_f, \theta_r)$$

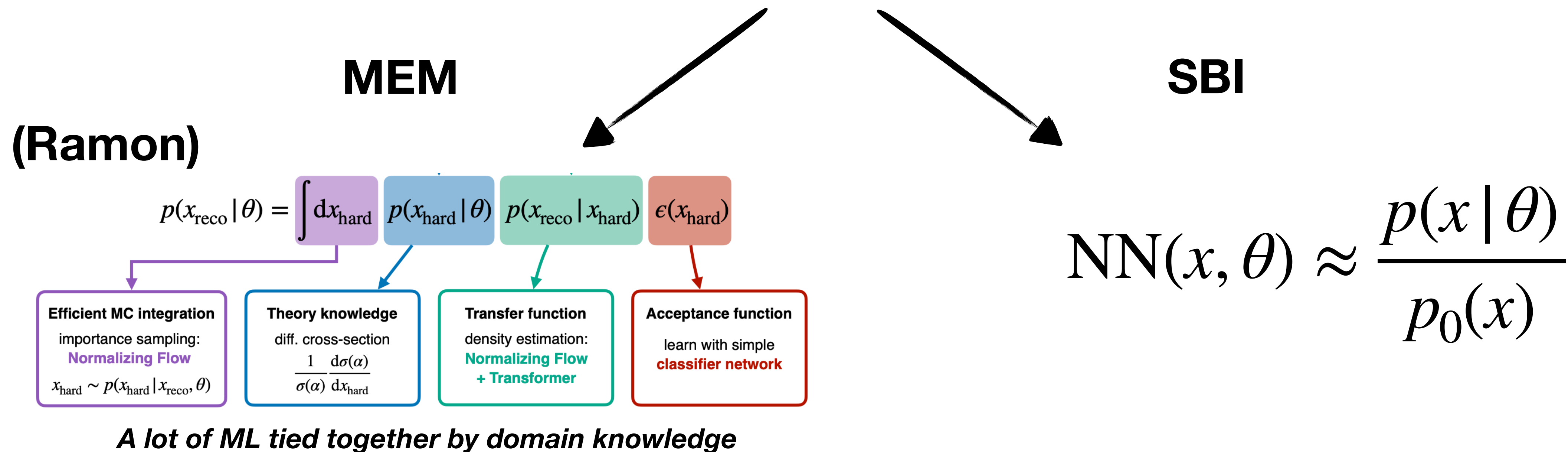


33

(Kyle)

Dynamics: Control over the Data Flow

Two ways to do optimal observables




why should you ever do MEM after we got SBI?

- maybe to have a fuzzy feeling of control
- to separately debug each piece
- the fuzzy feeling is very expensive - how much do we care?

Extreme Version of Dynamics

end-to-end gradient based optimization: SANNT, neos, Inferno,

Interesting parallel to foundation models: pretrain a initial analysis w/o systematics, finetune later e2e in-situ w/ full physics context



KIT
Karlsruhe Institute of Technology

Changes to the training procedure summarized

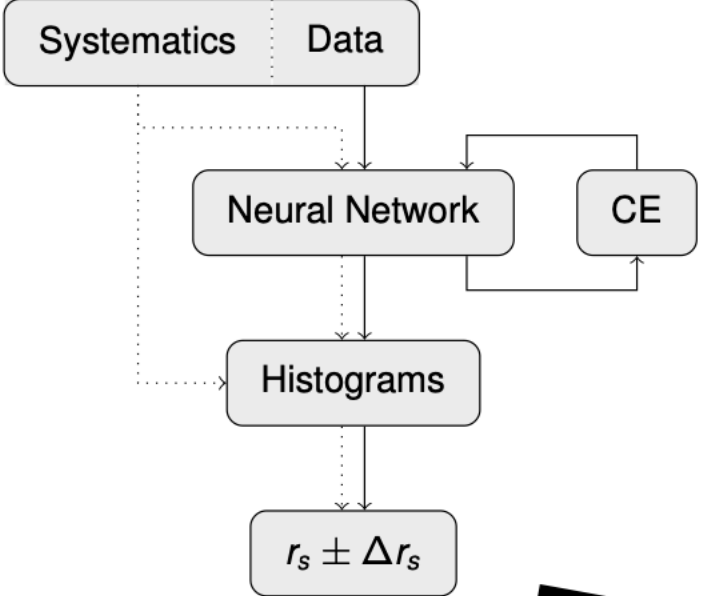
Use CE pretraining achieving:

- Process separation
- Good starting point for SANNT

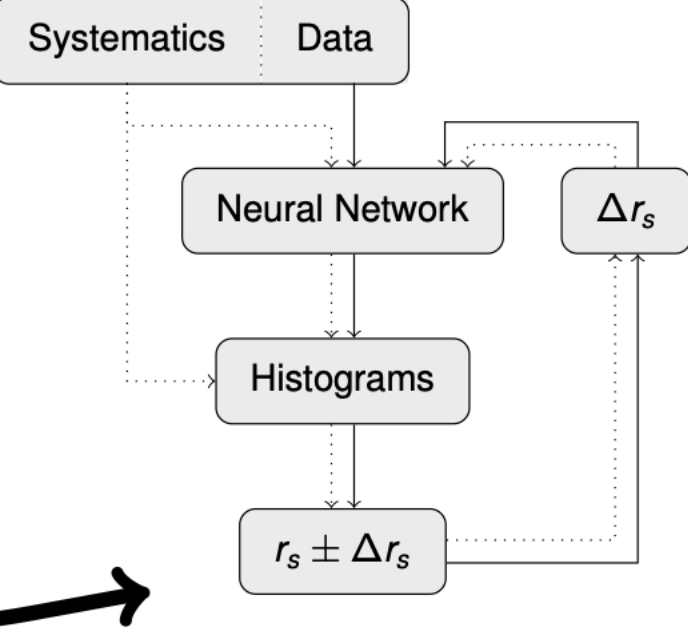
Main difference: SANNT vs. CENNT :

- Changed training objective
- Added information about systematic variations to the training

CENNT



SANNT



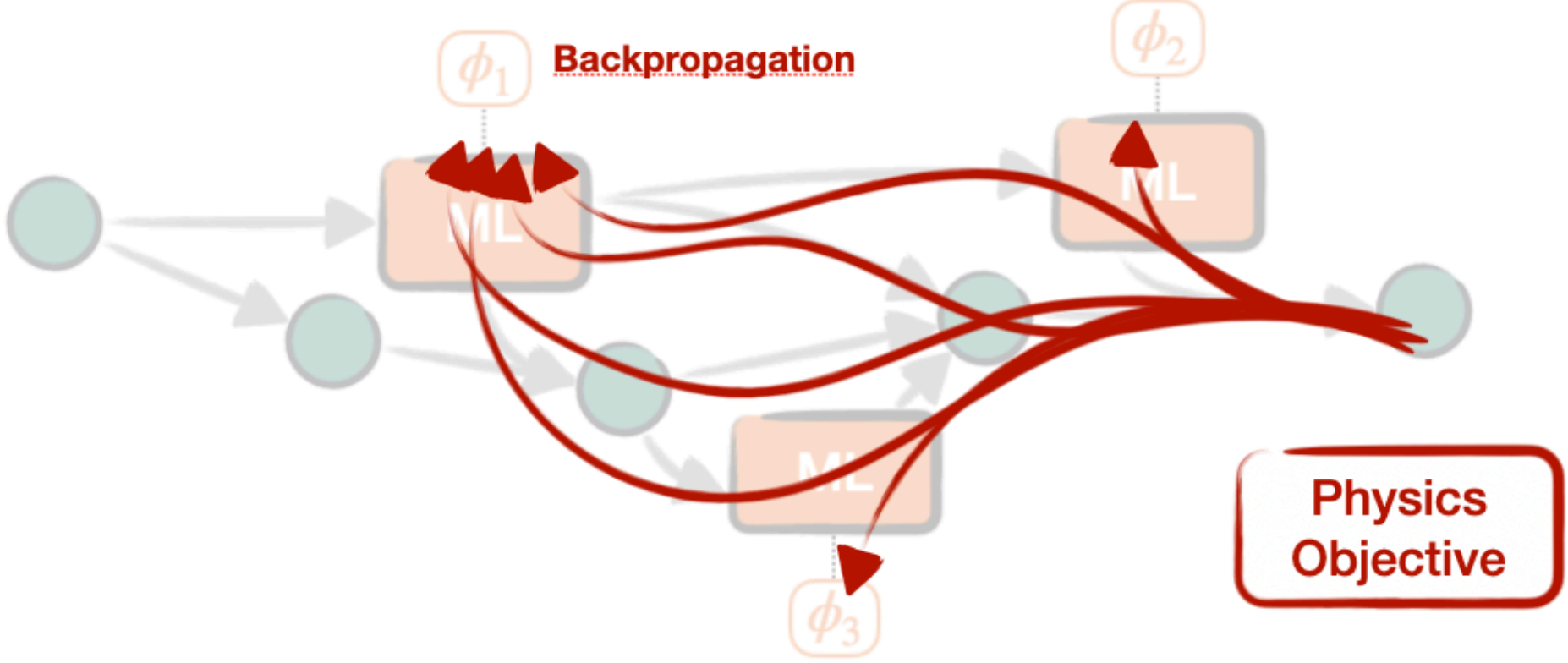
Conventional analyses
○○

Δ information
○○●○○

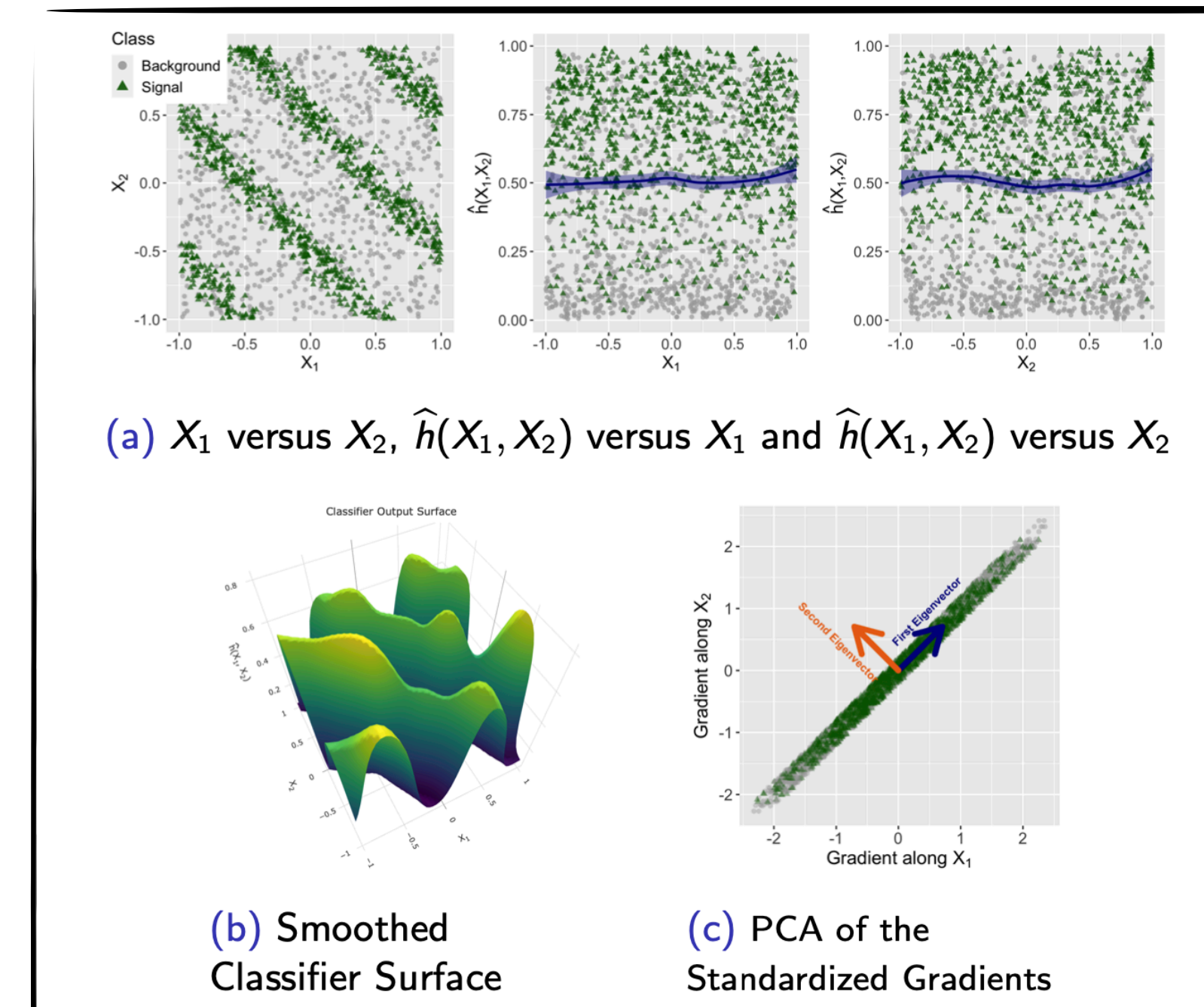
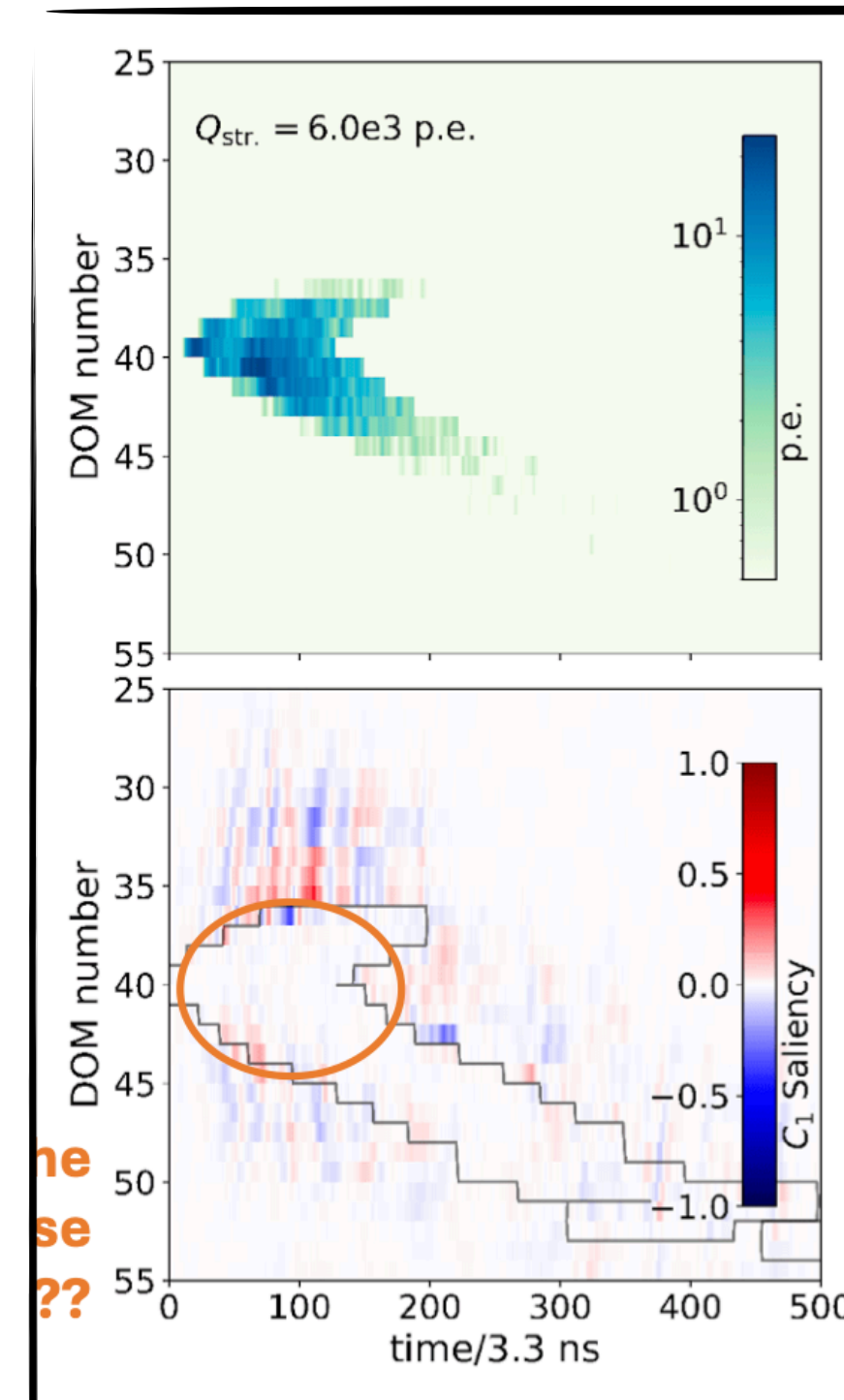
Effects from added Δ
○○○

$H(125) \rightarrow \tau\tau$ application
○○○○○○○

Summary
○○○



Intuitive & Interpretable is what you are used to

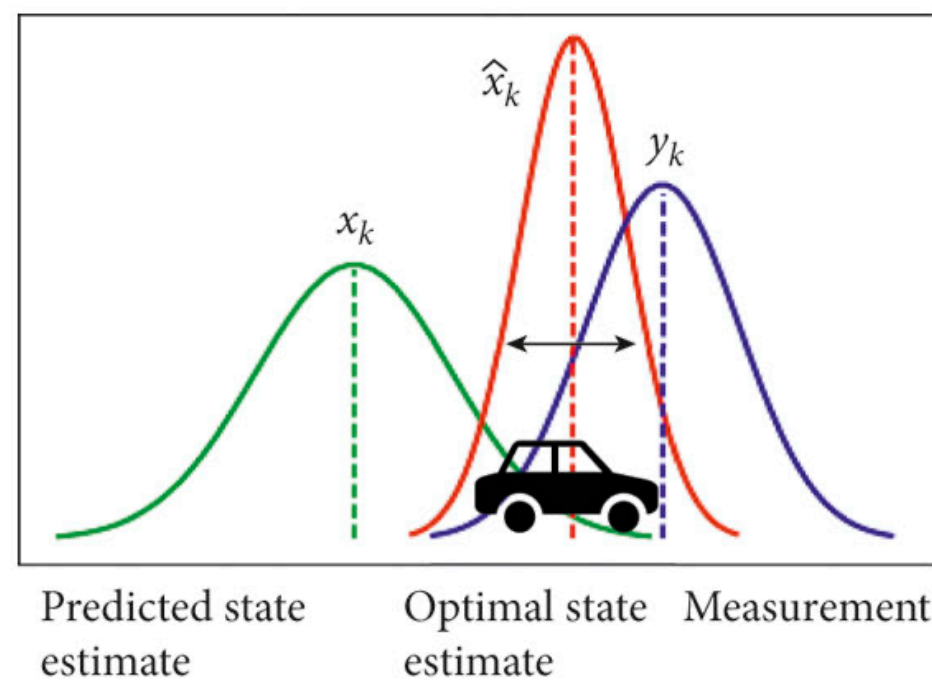
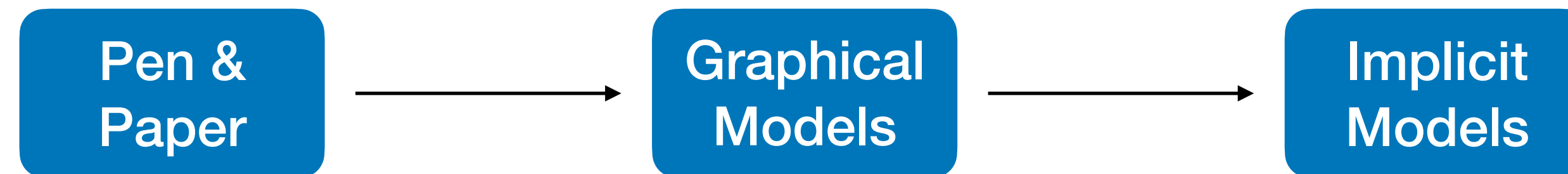


(Tobias, Mikael, Philipp, Pierre, etc)

Simulation-Based Inference

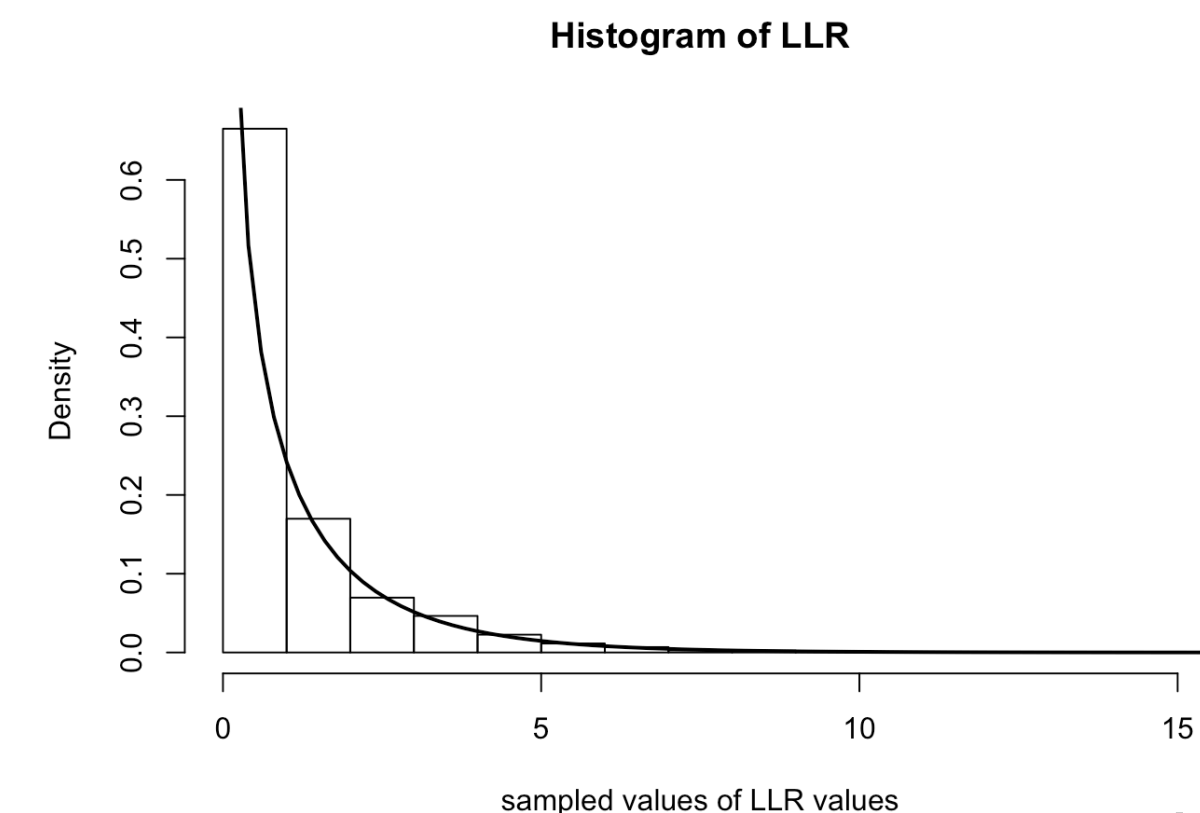
Statistics always had a nice quality: a common language to tie together many different fields that are driven by data, irrespective of the details

Version 1: “Pen & Paper” Statistics



Conjugate Priors, Kalman Filters, ...

Bayes



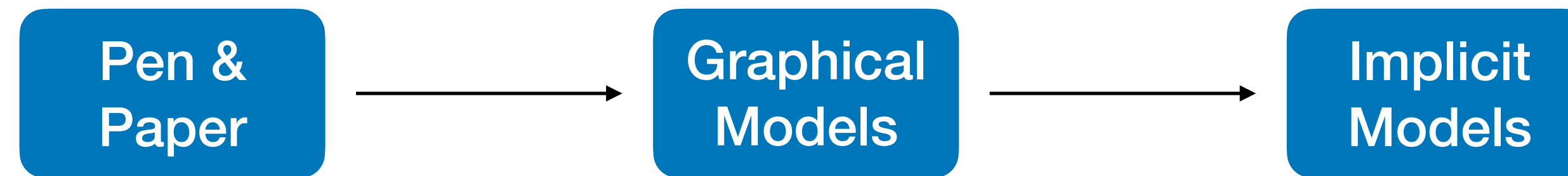
Distribution-free Tests, Asymptotic (Wilks'), etc..

Frequentist

Simulation-Based Inference

Statistics always had a nice quality: a common language to tie together many different fields that are driven by data, irrespective of the details

Version 2: Principled, Heavy Compute Stats



Bayesian Hierarchical Modelling of Conditional Densities

- **Population level** distribution within bin b :

$$z_i | \mu_b, \sigma_b \stackrel{\text{indep.}}{\sim} N(\mu_b, \sigma_b^2), \quad (6)$$
 with redshift population variance σ_b^2 .
- **Object level:**

$$z_i | x_i \stackrel{\text{indep.}}{\sim} N(\hat{\zeta}_i, \hat{\tau}_i^2), \quad (7)$$
 Gaussian replacement of conditional density estimates.
- With $\hat{X}_{\text{nb}} := \{\hat{\zeta}_i, \hat{\tau}_i\}_{i=1}^{n_{\text{nb}}}$, obtain the (joint) marginal posterior

$$p(\mu_b, \sigma_b | \hat{X}_{\text{nb}}) \propto p(\mu_b, \sigma_b) \prod_i N(\hat{\zeta}_i | \mu_b, \hat{\tau}_i^2 + \sigma_b^2). \quad (8)$$

PHYSTAT 2024 - Statistics meets ML September 9, 2024 9 / 24

(Maximilian)

The HistFactory model

- **HistFactory** is a statistical model for **binned template fits** (CERN-OPEN)
- prescription for constructing probability density functions (pdfs) from
- covers a **wide range of use cases** (and can be extended if needed)
- here: primary observables are \vec{n} , auxiliary observables are \vec{a}

observed data \rightarrow primary term prediction (sum over samples)

unconstrained parameters, e.g. POI \rightarrow $\vec{k}, \vec{\theta}$

auxiliary data, e.g. from calibration measurement \rightarrow \vec{a}

constrained nuisance parameters \rightarrow $\vec{\nu}, \vec{c}$

product over all bins

$$p(\vec{n}, \vec{a} | \vec{k}, \vec{\theta}) = \prod_i \text{Pois}(n_i | \nu_i(\vec{k}, \vec{\theta})) \cdot \prod_j c_j(a_j | \theta_j)$$

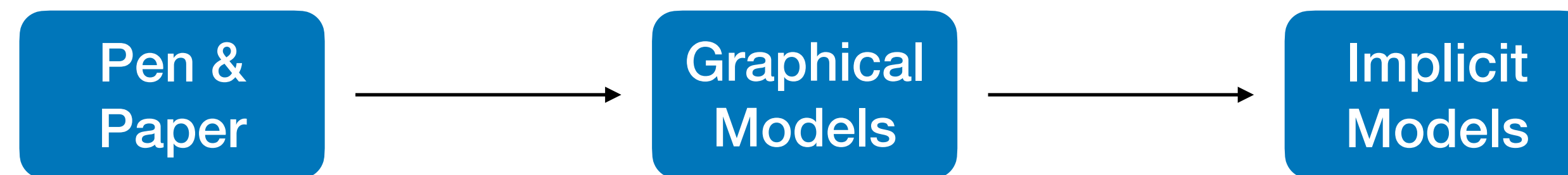
Alexander Held

(Alex, Kyle)

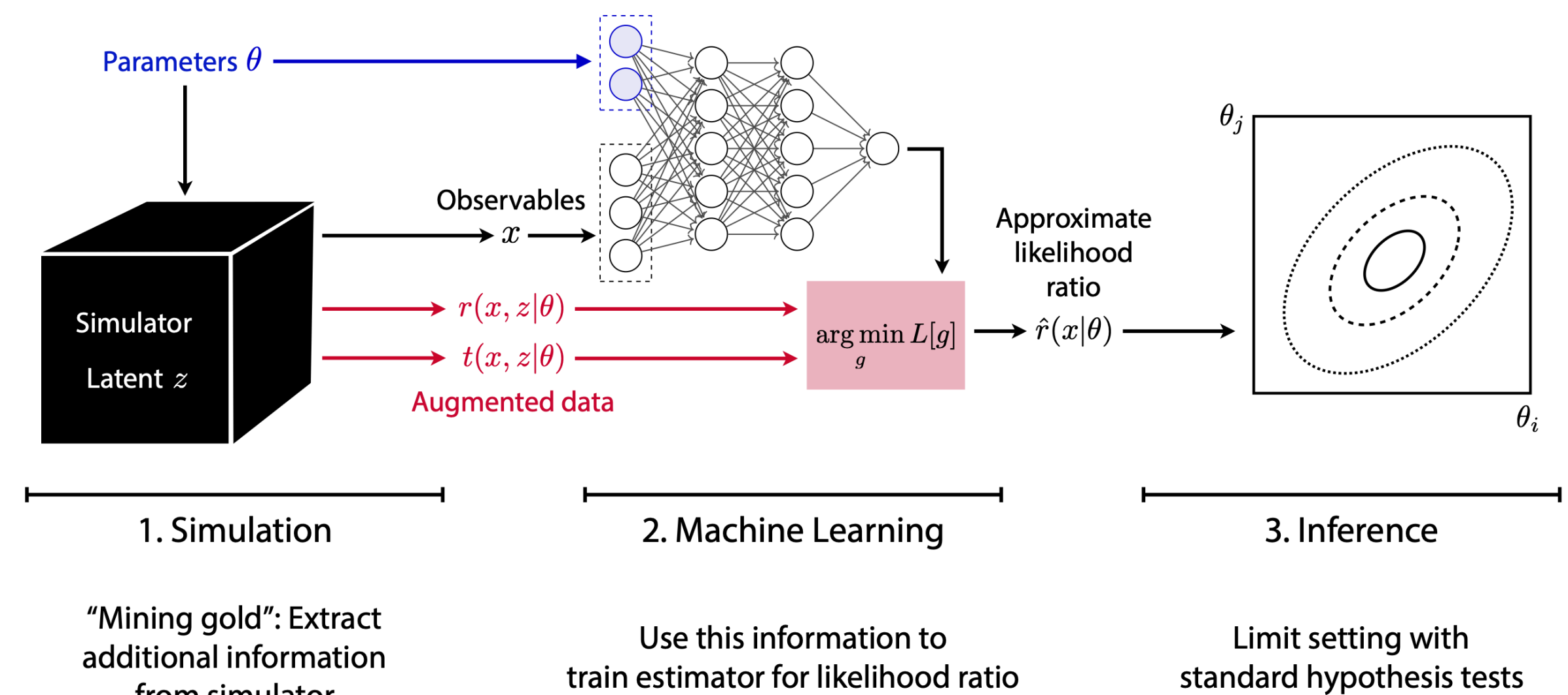
Simulation-Based Inference

Statistics always had a nice quality: a common language to tie together many different fields that are driven by data, irrespective of the details

Version 3: Implicit Models aka Simulation-based Inference



- Model is defined implicitly as a black box sampler
- Fast, approximate amortized Inference



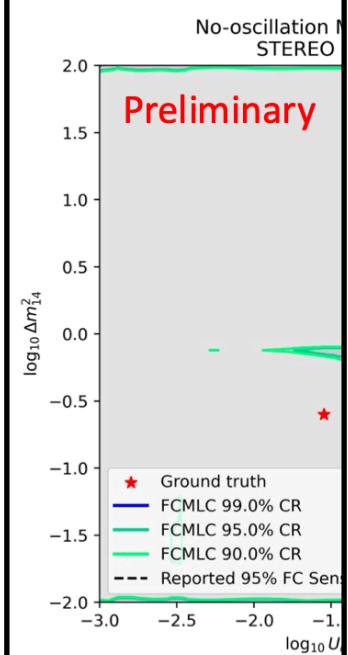
Simulation-Based Inference

(also see Gilles' Talk)

Results

Fits on Simulated Experimental Data

Sensitivities



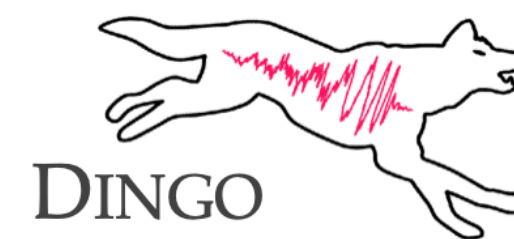
Discussion

FCMLC Pros and Cons

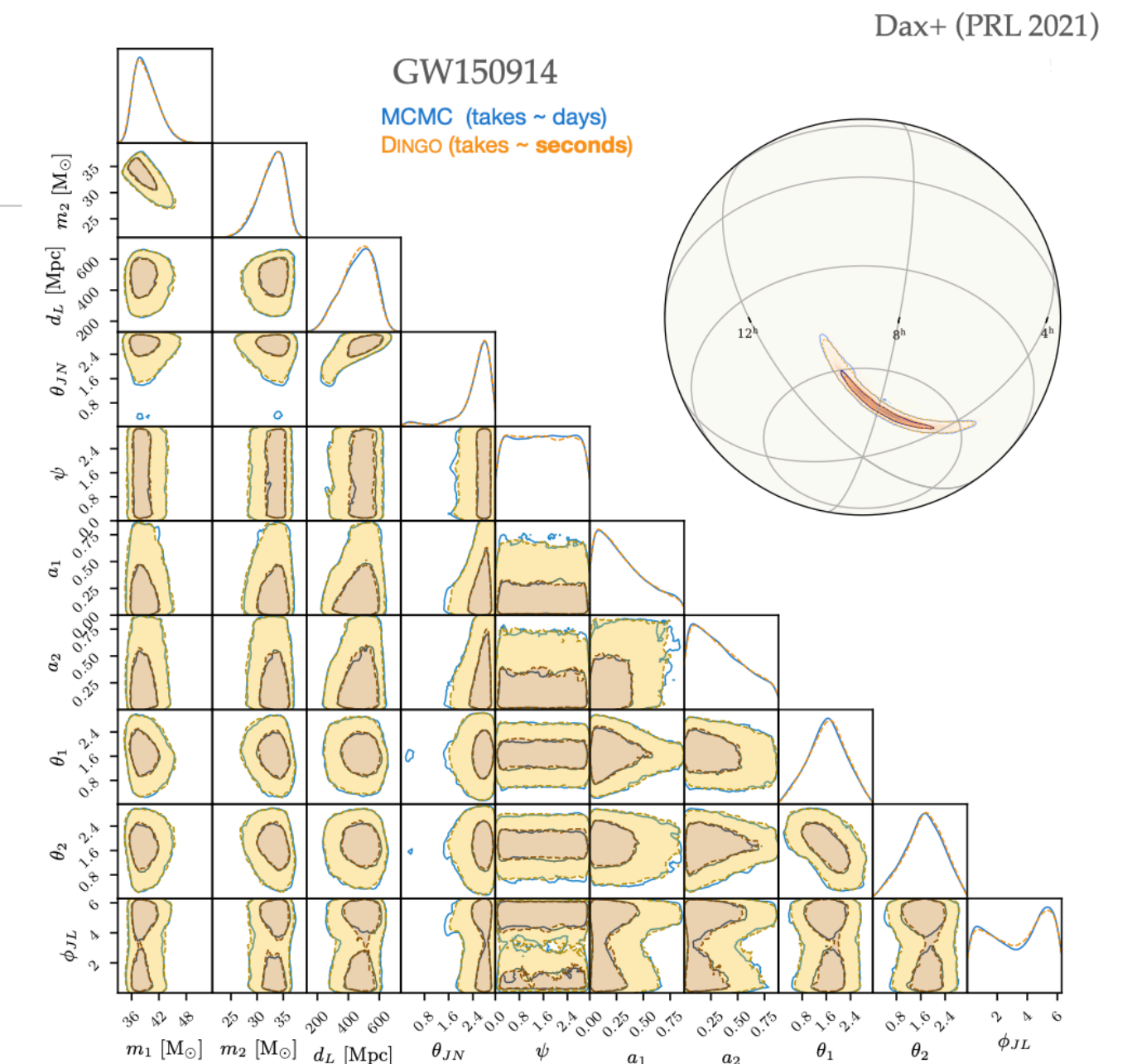
- Pro: The SBI approach is much faster than traditional fitting methods.
 - With a NN and a NF trained, posterior density estimation takes ~5 minutes and can run in a Jupyter notebook.
 - Together with generation of MC data, training, and FCMLC evaluation, ~2.5K CPU hours
 - By comparison, Feldman-Cousins takes ~510K CPU for the experiments considered in this work.
- Pro: Runtime of FCMLC is ~constant with respect to the inclusion of more experiments.
- Pro: SBI does not assume Gaussian uncertainties, and it is straightforward to set up simulations to capture this.
- Con: Hyperparameter optimization can further extend the overhead of FCMLC.
- Con: FCMLC is sensitive to the way in which you generate your MC data.
- Con: FCMLC's posterior density and Feldman-Cousins answer slightly different statistical questions.

(Josh)

Binary black holes



- Inference in seconds to minutes using pre-trained networks (1000x speed up)
- Extremely good agreement with standard samplers
- Likelihood-free



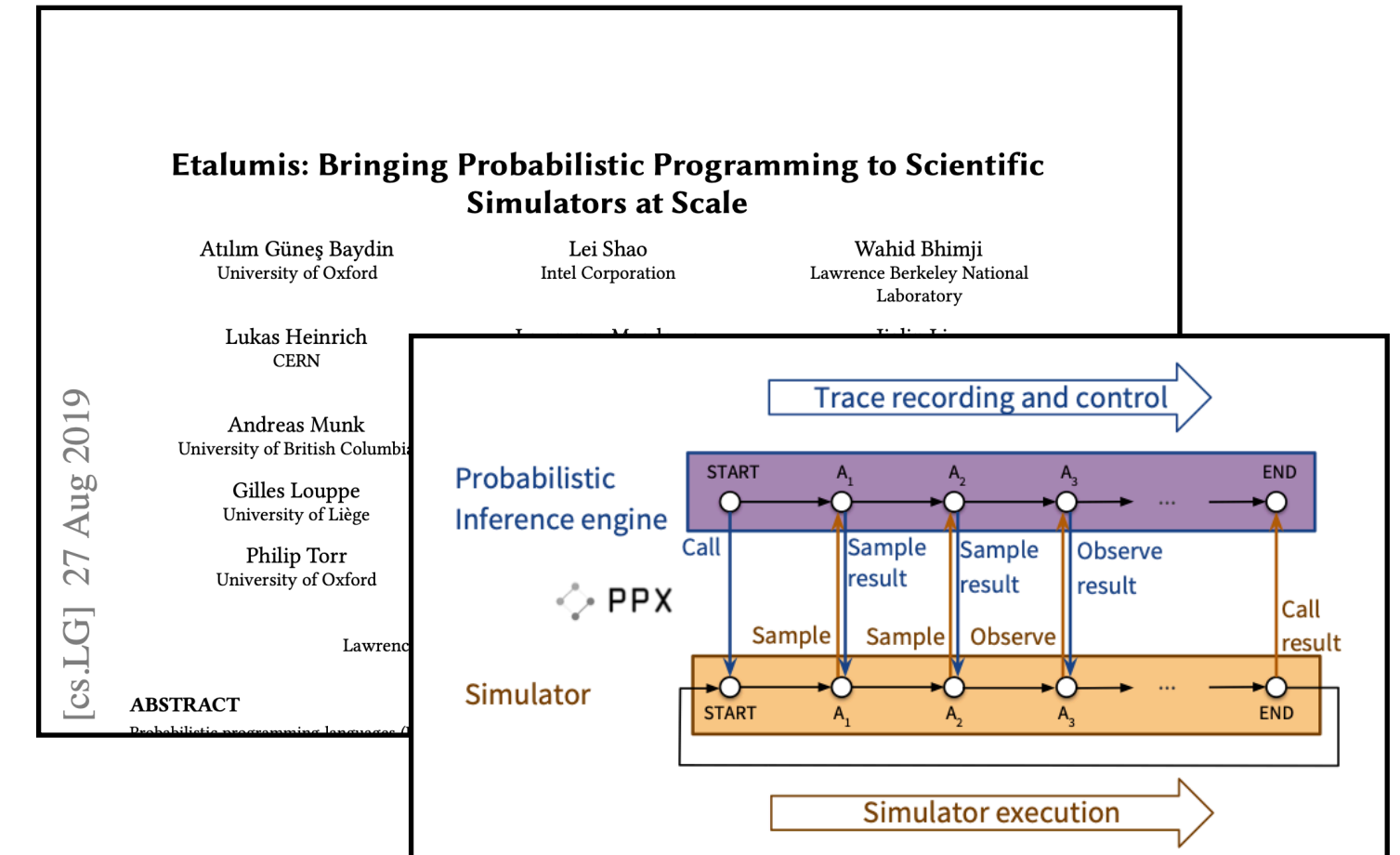
(Max)

Not better, but much much faster. Interesting Observation:
 SBI useful even in not likelihood-free settings, just as a fast amortized inference
 → are there principled ways to do L'hood-full SBI? (Gradients, L'hood, etc..)

Tooling, Tooling Tooling

1907.03382

Differentiable & Probabilistic Programming bridge the gap between “black box” simulators and graphical models



High-dimensional model comparison for cosmology

Piras and Spurio Mancini, 2023

Emulation (CosmoPower-JAX)
+ Differentiable and probabilistic programming
+ Scalable sampling (NUTS)
+ Decoupled and scalable evidence (*harmonic*)
=

The future of cosmological likelihood-based inference...
(Piras et al., 2024) arXiv:2405.12965

Alicja Polanska

(Alicja)

Speeding things up with neural emulators

Emulated spectrum: $\hat{L}_\lambda(\theta; \mathbf{w}) = \sum_{i=1}^{N_{\text{PCA}}} \hat{\alpha}_i(\theta; \mathbf{w}) q_{\lambda, i}$

Emulating spectra | Emulating photometry

16-parameter SPS model | sub-percent accuracy | factors x 10⁴ speed-up | differentiable

ALSING, PEIRIS, LEJA, HAHN, TOJEIRO, MORTLOCK, LEISTEDT, JOHNSON, CONROY (APJ, 2020)

(Hiranya)

MADNIS — Neural importance sampling

Improvement wrt VEGAS: 2.5, 2.0, 1.5, 1.0, 0.5, 0

unweighting efficiency ϵ [%]: 20, 15, 10, 5, 0

VEGAS: fixed α , trained α , fixed α , VEGAS-unit, VEGAS-unit, stratified, stratified, trained α , trained α , dropping, dropping, buffered, buffered, $R_B = 3$, $R_B = 5$

① excellent results with all features

Fully differentiable version available

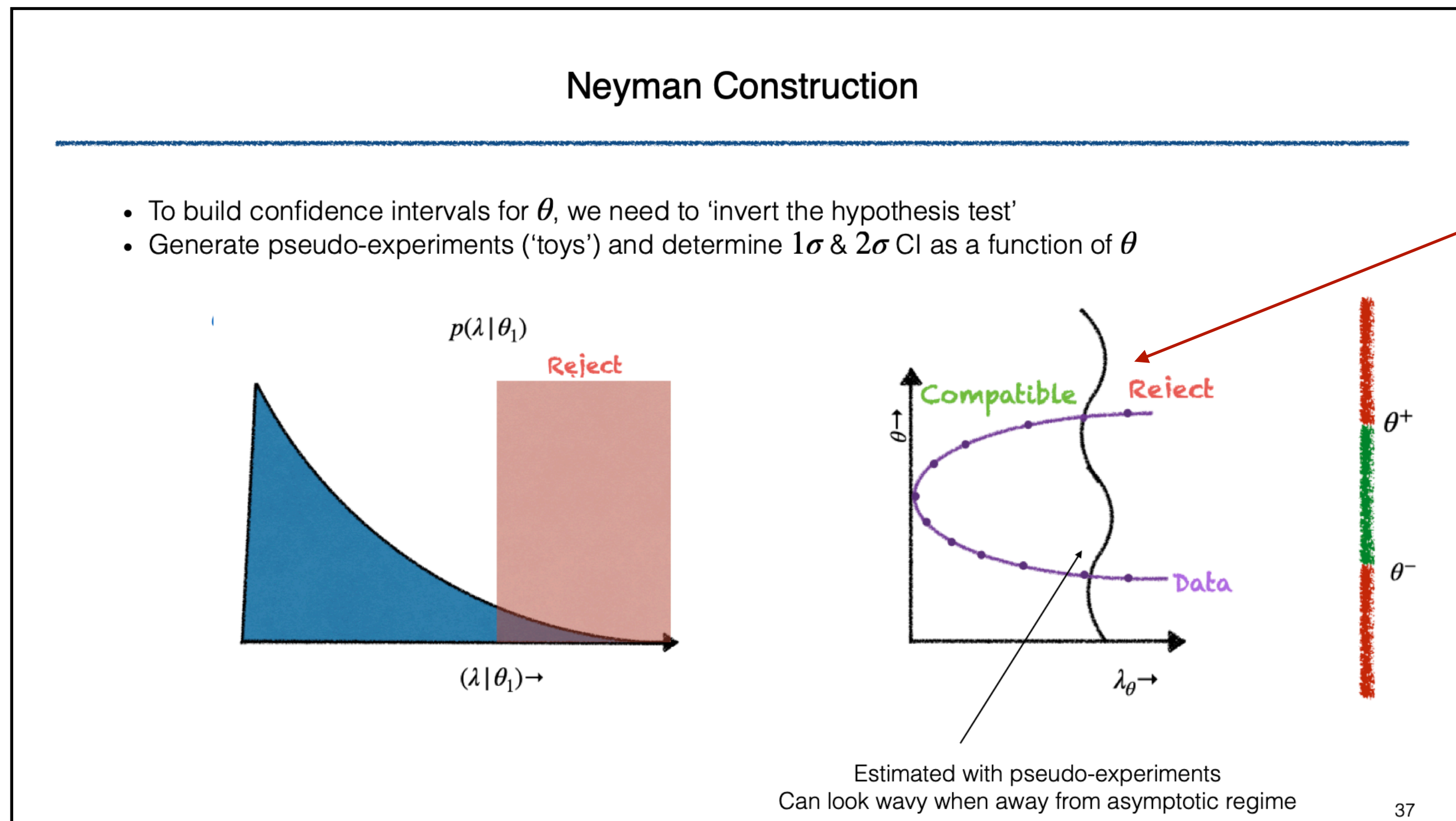
2212.06172, 2311.01548, 2408.01486

(Ramon)

Simulation-Based Inference

Slowly, but surely: we are actually doing this in HEP. Step change soon?

#SBI analyses at LHC: $0 \rightarrow 1$. How will we ensure $1 \rightarrow N$ is easier?

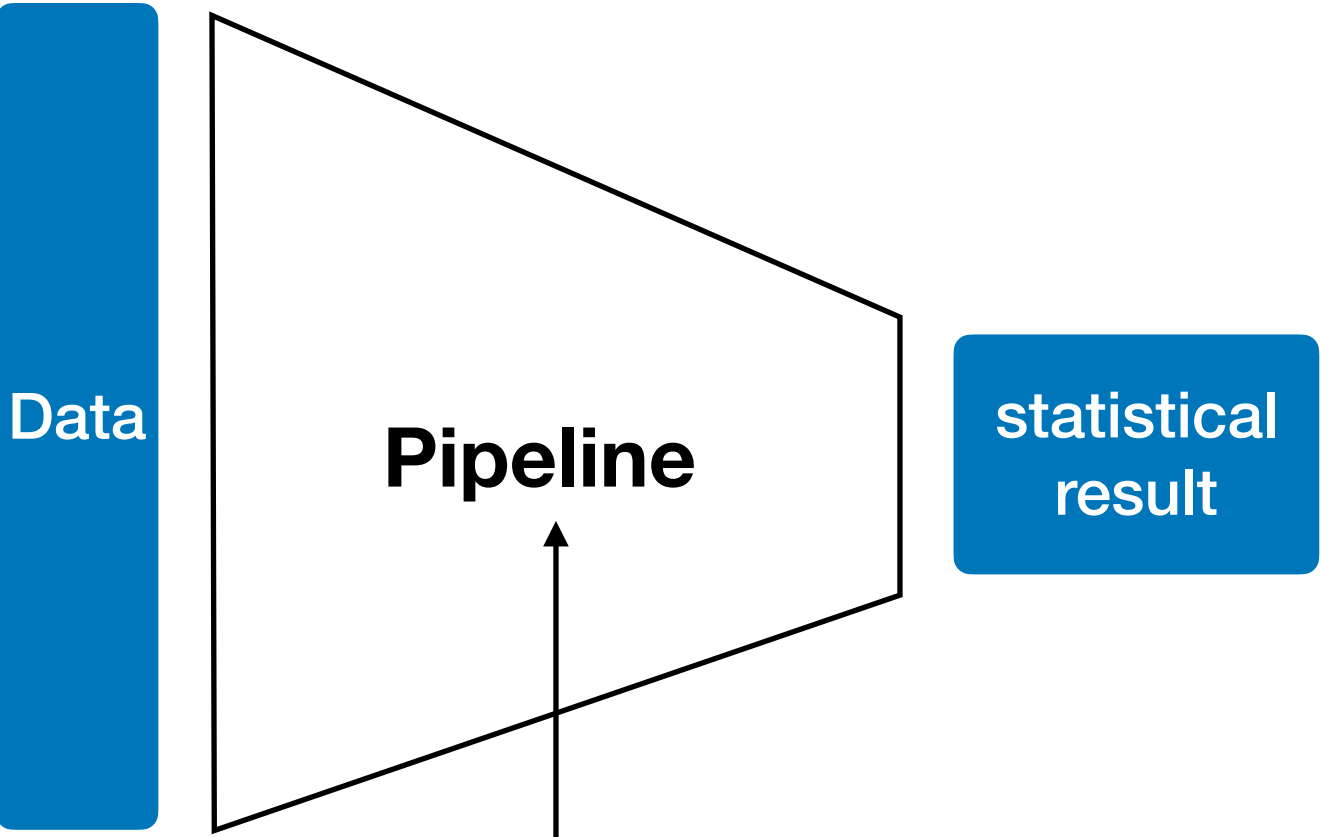


**a frequentist's dream:
a non-boring Neyman
Construction**

(Aishik)

How much Statistics can we cram into ML

Statistics: Data → Insight

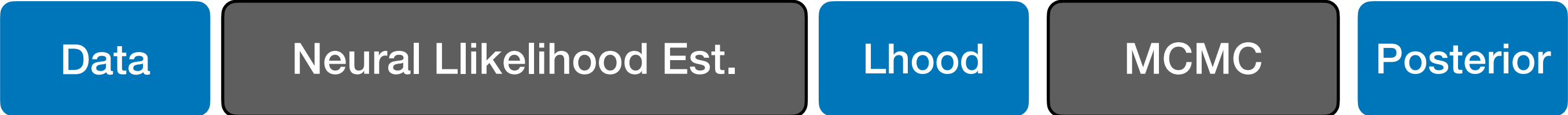


how much of this is ML, how much is Stats

(hard codes prior)



(can adjust prior but expensive)



(high-level parametrization)



(Learning to Profile)

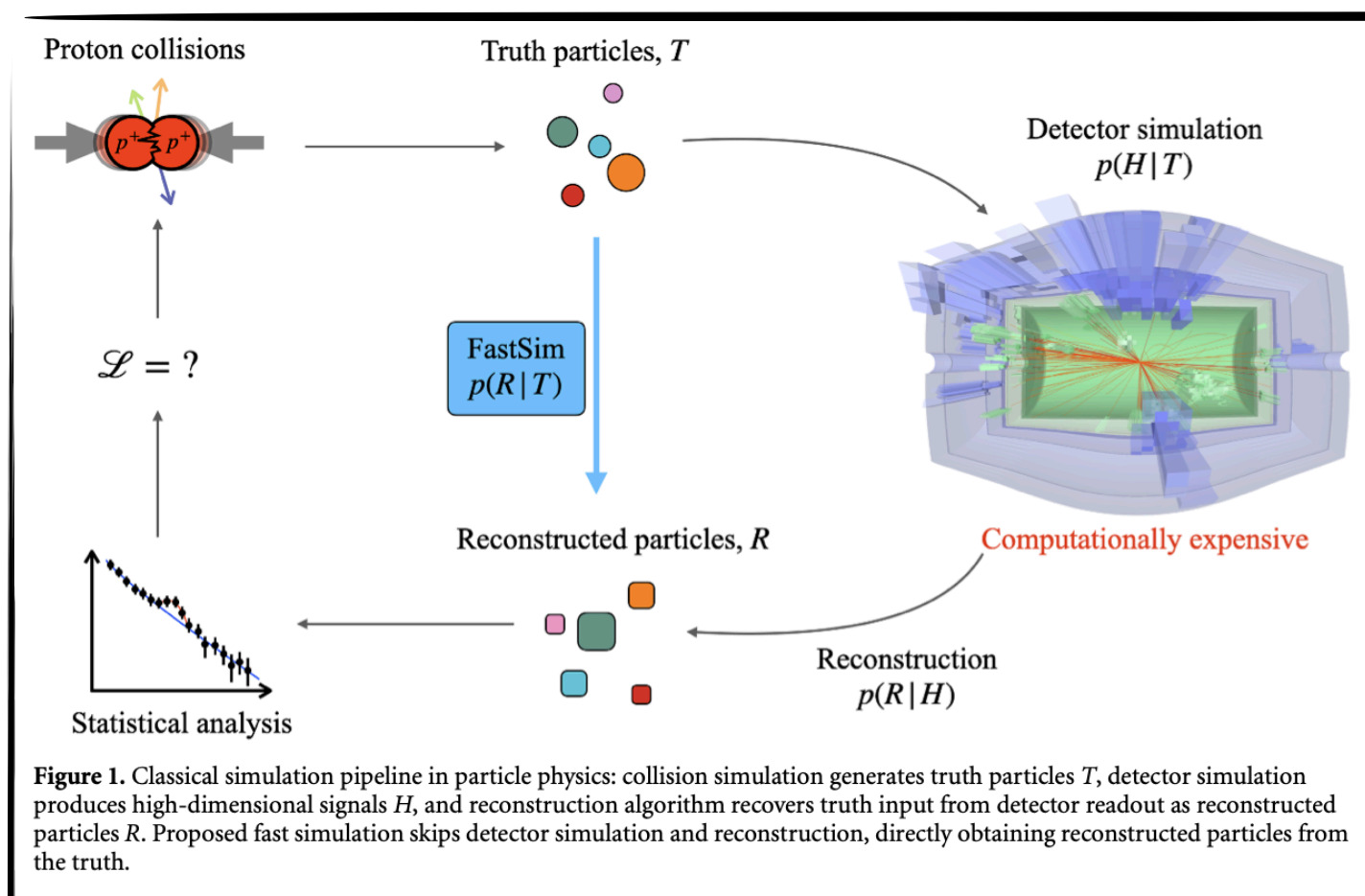


**How much do we trust the process ?
Where, how many inspectable intermediate steps do we want?**

Getting Rid of the Simulator: Unfolding

Reminder that HEP is always also a social enterprise. Core question: what's the format. ML allows is to go beyond histograms and into full phasespace but in which direction? If folding becomes trivial, what does this mean?

Example:



2211.06406

(Vini)

➔
➔

Unfolding

Benefits of Unfolding:

- Facilitate the comparison between theory predictions and measurements
- Facilitate the comparison between different theory predictions

Drawbacks of Unfolding:

- Unfolding is a lossy procedure and may lead to larger uncertainties to the final measurement

See also: [GVD 11.6.11](#)

$$\begin{array}{ccc}
 \rho_{\text{sim}}(x_{\text{part}}) & \xleftrightarrow{\text{unfolding inference}} & \rho_{\text{unfold}}(x_{\text{part}}) \\
 \downarrow p(x_{\text{reco}} | x_{\text{part}}) & & \uparrow p(x_{\text{part}} | x_{\text{reco}}) \\
 \rho_{\text{sim}}(x_{\text{reco}}) & \xleftrightarrow{\text{forward inference}} & \rho_{\text{data}}(x_{\text{reco}})
 \end{array}$$

[Unfolding with combine](#)

[roduction to unfolding](#)

➔

(Tilman)

Summary

We're very good at learning new ML methods and finding a way to move them closer to production

Classic Statistics can give us guidance towards *what's possible in principle*

Interpretability is largely about trusting a process & convincing ourselves

Domain Knowledge helps until it doesn't (bitter lesson?)

SBI as a 3rd-wave of statistical methodology / common language between sciences

