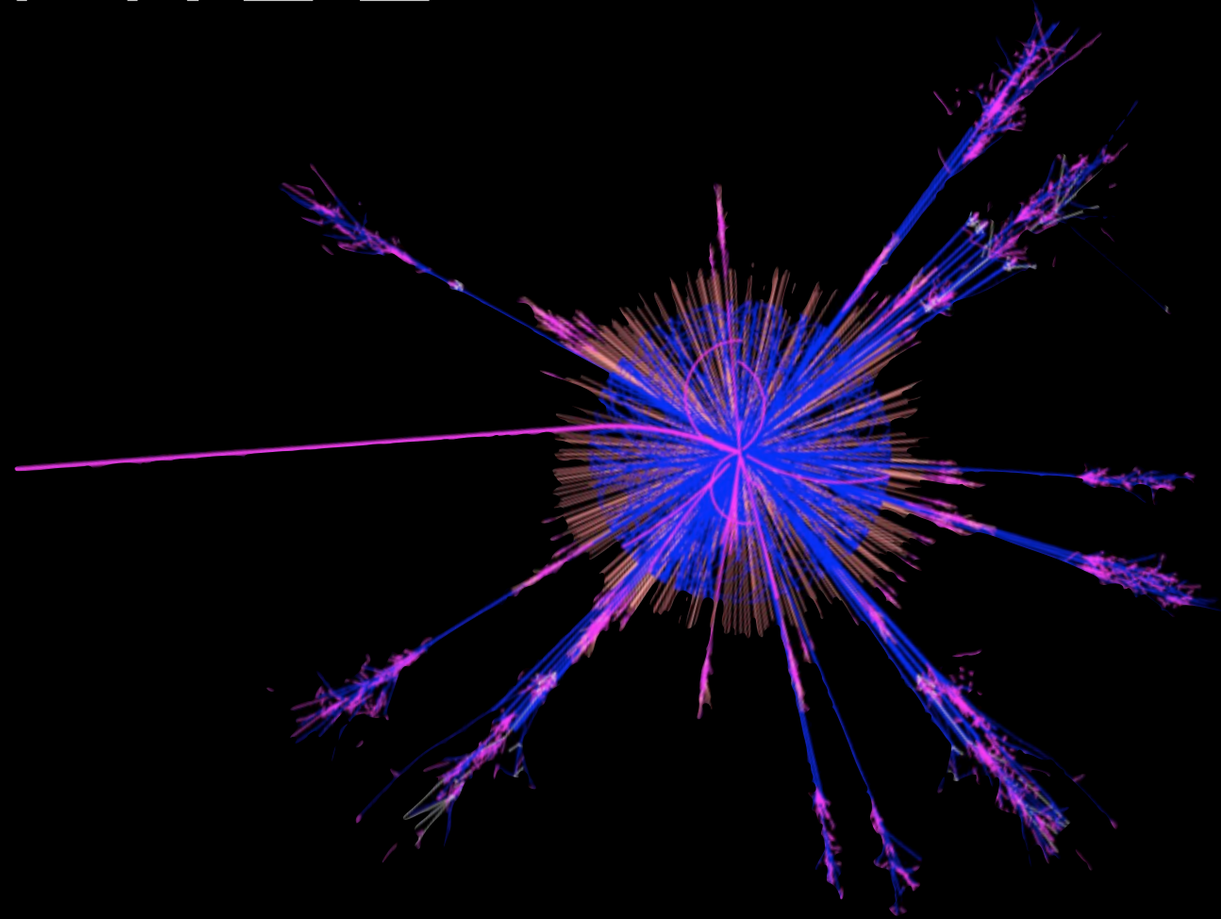
NYU Center for Data Science

Center for Cosmology and particle physics

LIKELIHOOD FREE INFERENCE



@KyleCranmer
New York University
Department of Physics
Center for Data Science

Fundamental Scientific Pipeline

- How to construct such models to explain large systems?
  - Representation Challenge: How to parametrize ?

$$p(x_1, \ldots, x_n) = f_\theta(x_1, \ldots, x_n)$$

- How to adjust the parameters of such models to best explain the data?
  - Learning Challenge: How to fit θ to the data?

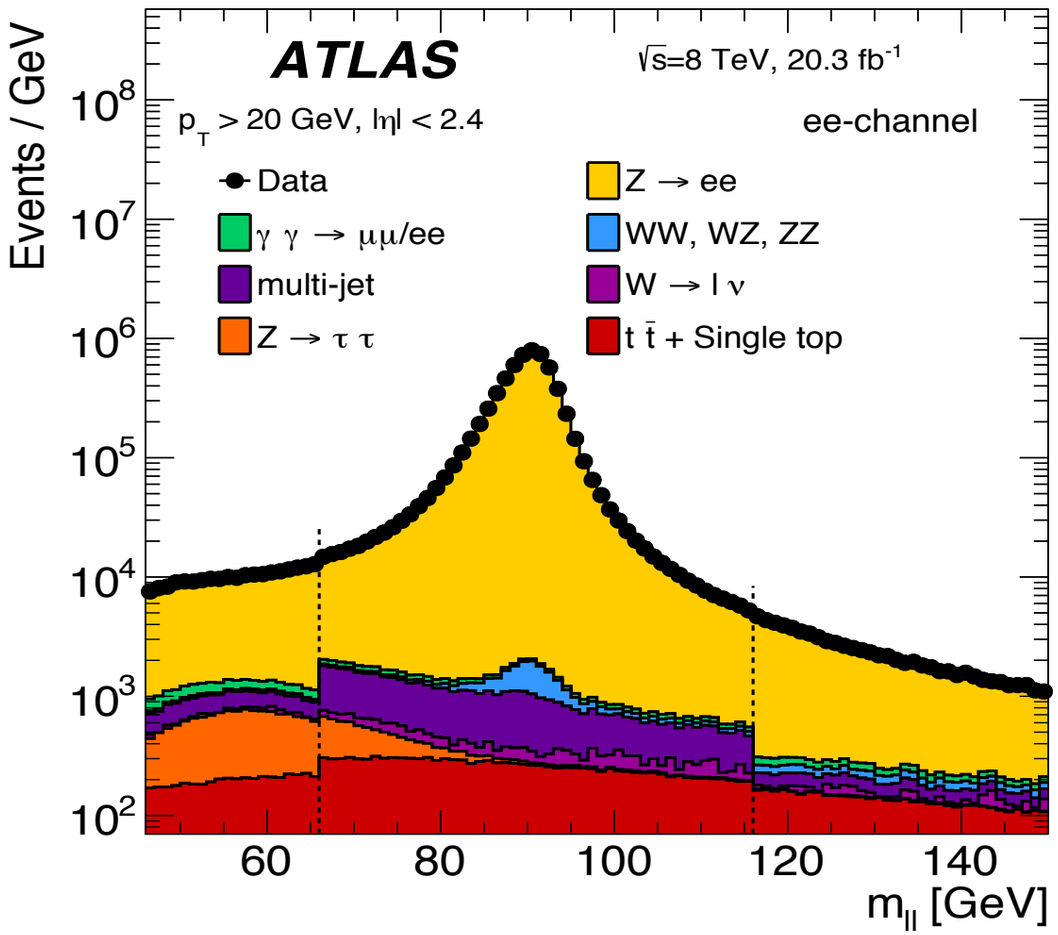
- How to evaluate likelihoods under the model?
  - Inference Challenge
  - In most interesting cases, exact inference will be computationally intractable.
  - Major Topic of the course: develop computationally efficient approximate inference.

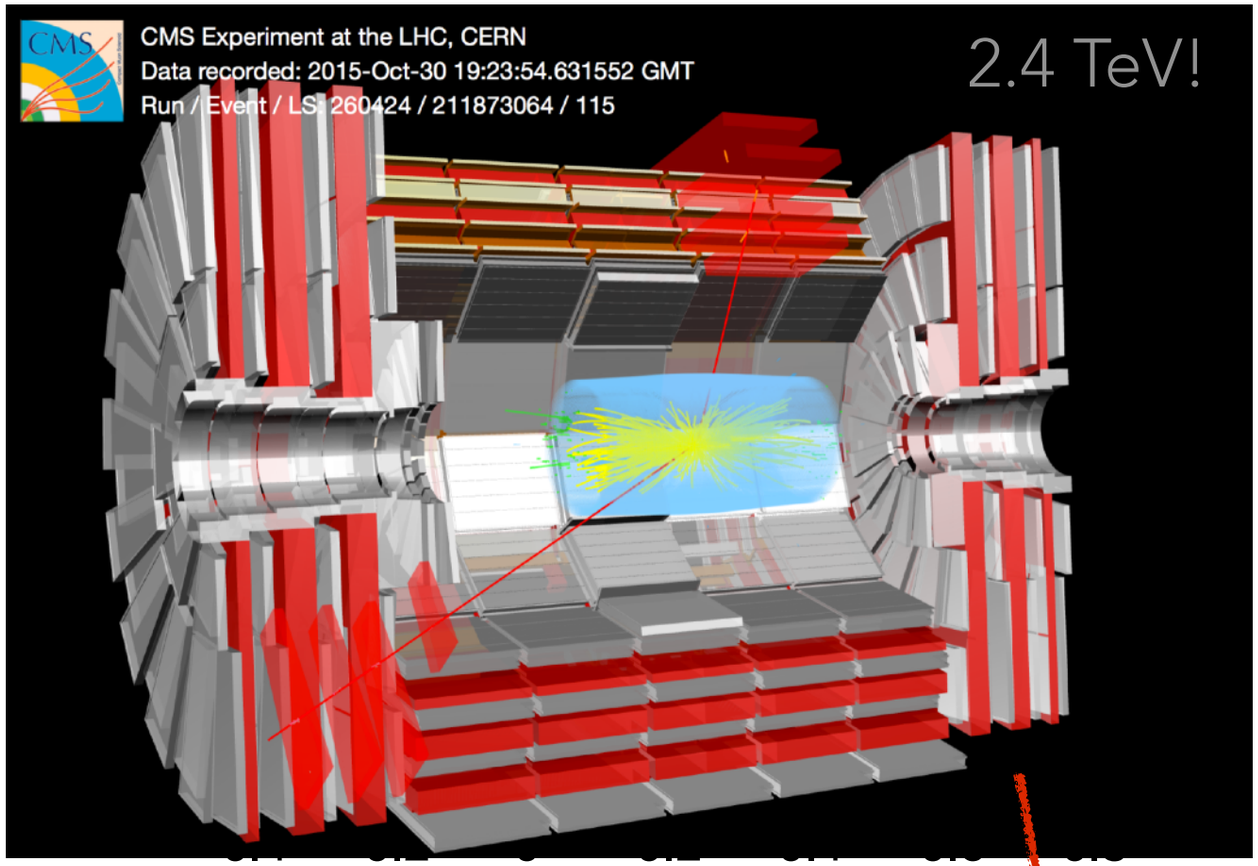
Task-driven inference

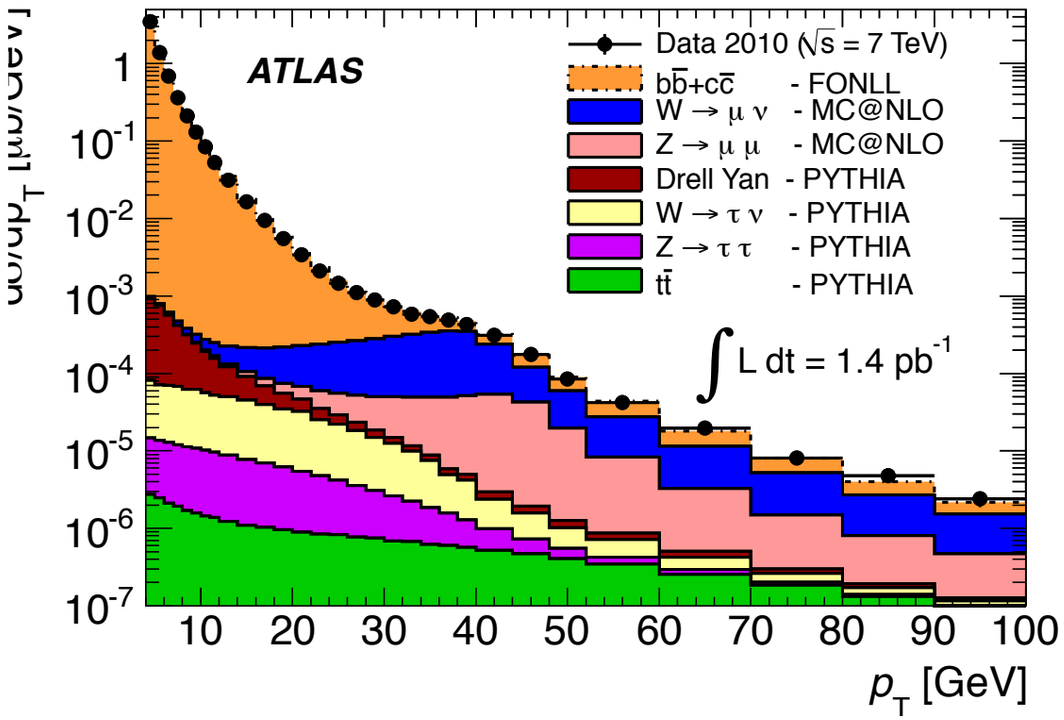
- Depending on the task, we might want to perform different kinds of estimation.
    1. Density Estimation: we are interested in the joint distribution, which can be subsequently used to perform any inference query.
    2. Prediction: we are only interested in a specific set of conditional distribution, e.g classification, or output prediction.
    3. Structural discovery: We are interested in the graph itself (not so much the parameters), e.g. determining dependencies between genes.

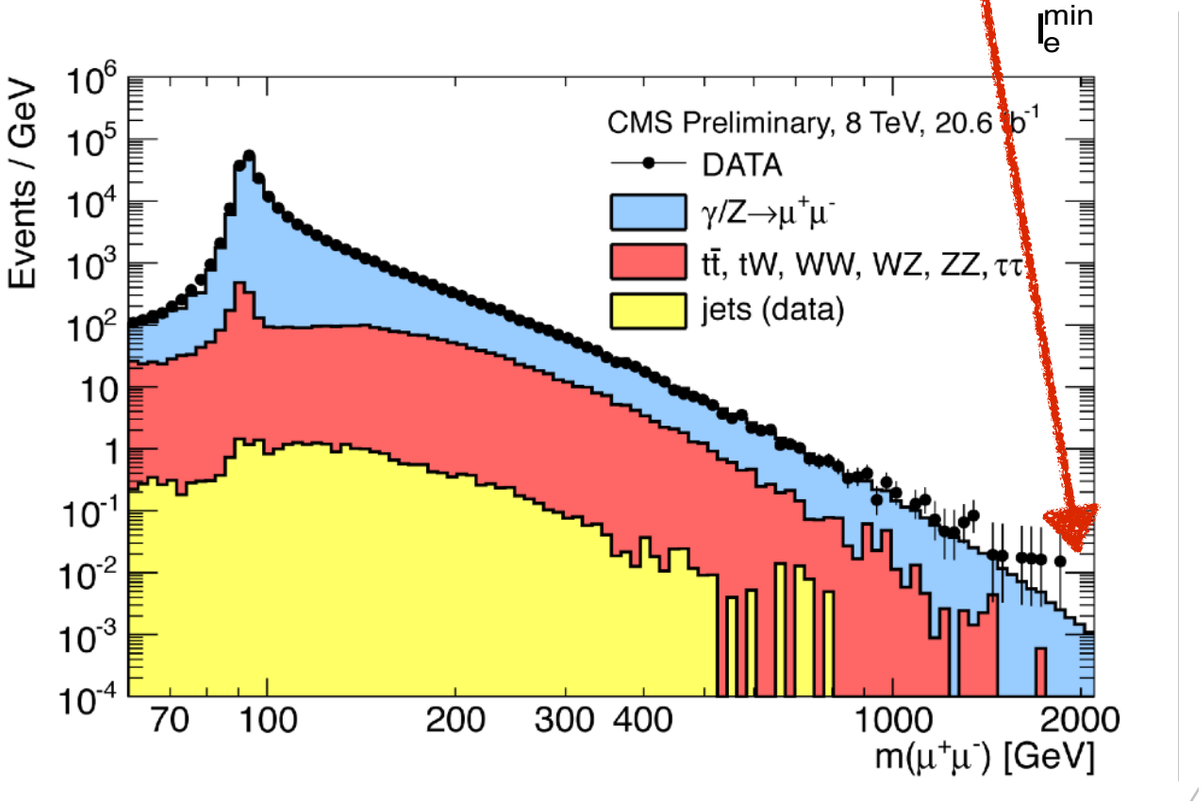
- (1) is typically harder than (2). (3) is typically harder than (2) and (1).

Applications: Particle Physics

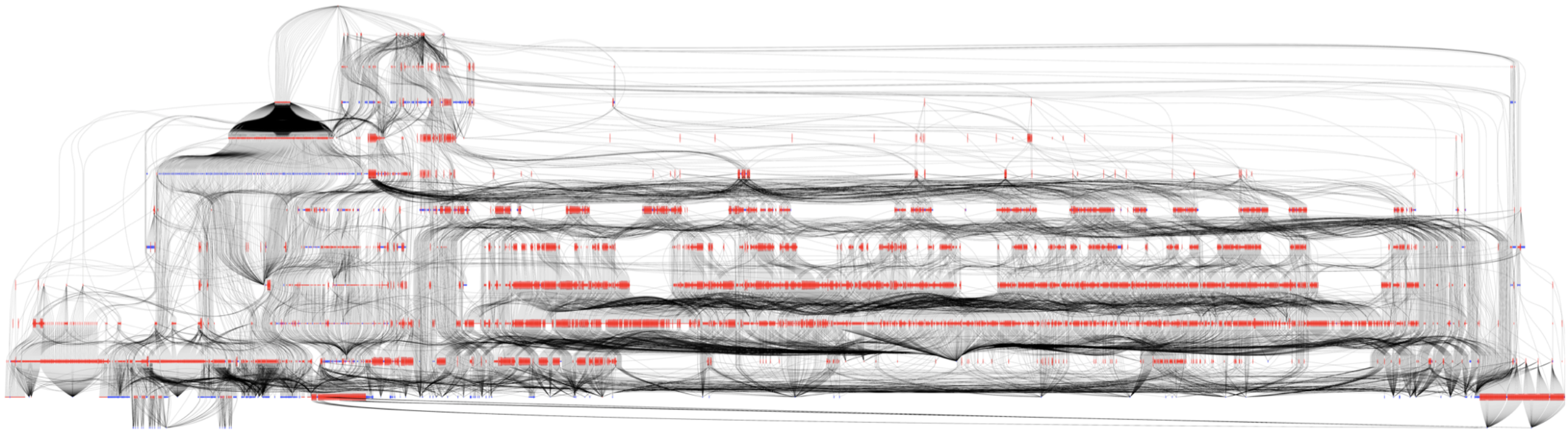






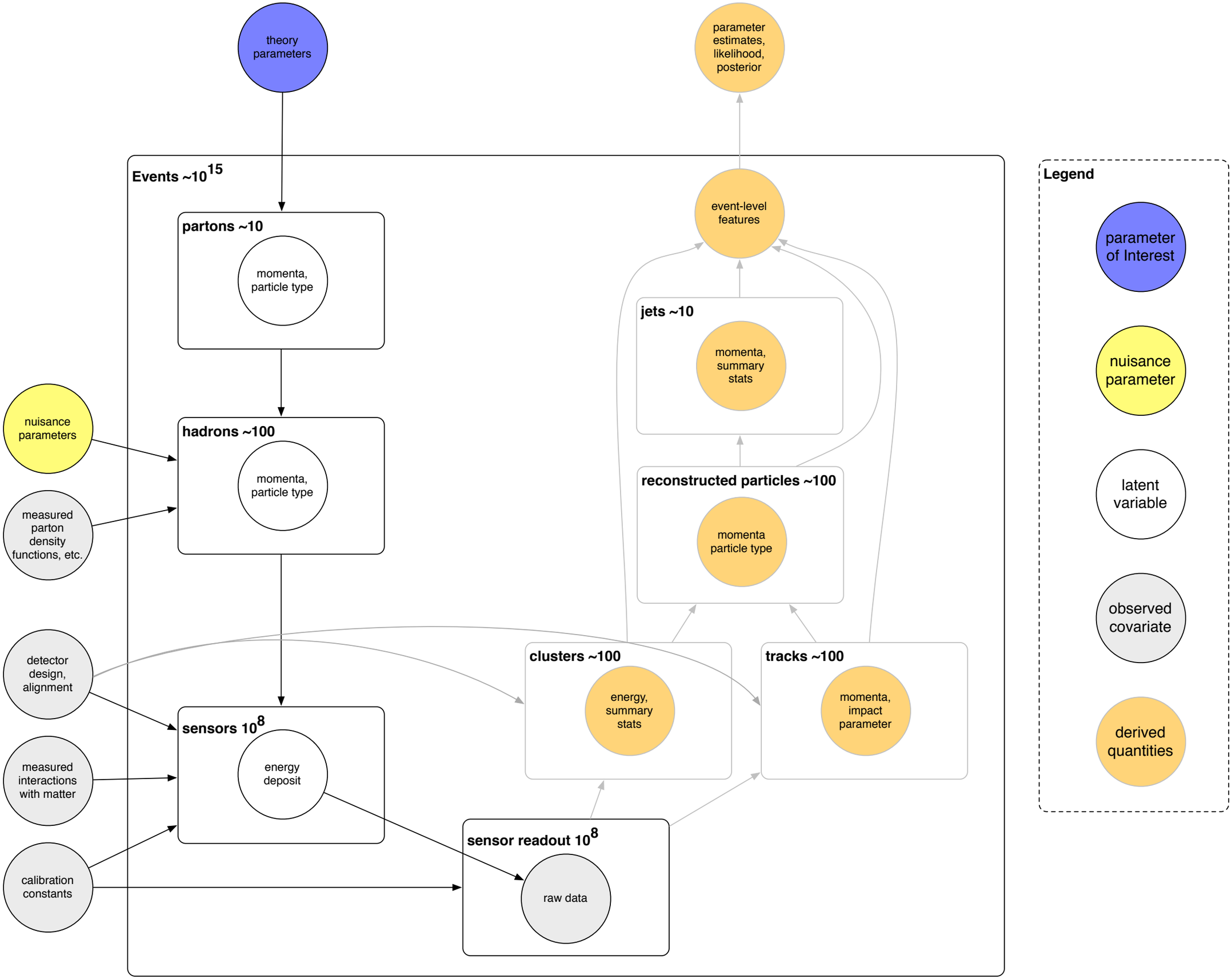


Applications: Particle Physics

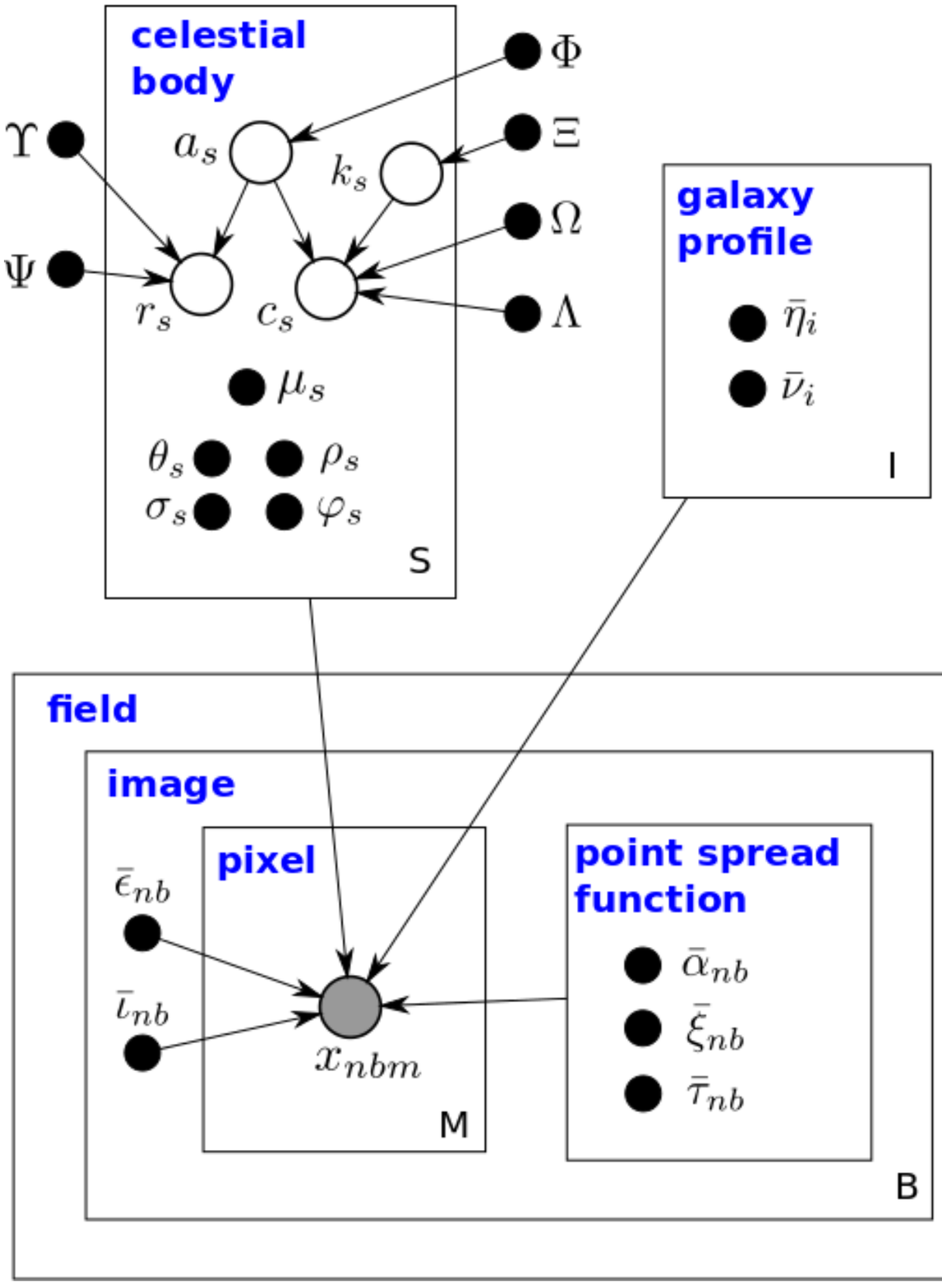


$$\mathbf{f}_{\text{tot}}(\mathcal{D}_{\text{sim}}, \mathcal{G}|\boldsymbol{\alpha}) = \prod_{c \in \text{channels}} \left[\text{Pois}(n_c|\nu_c(\boldsymbol{\alpha})) \prod_{e=1}^{n_c} f_c(x_{ce}|\boldsymbol{\alpha}) \right] \cdot \prod_{p \in \mathbb{S}} f_p(a_p|\alpha_p)$$

HIERARCHICAL GRAPHICAL MODELS IN PARTICLE PHYSICS



HIERARCHICAL GRAPHICAL MODELS IN ASTRONOMY



Celeste: Variational inference for a generative model of astronomical images

Today's topic:
Likelihood-free inference
&
Implicit Models

A COMMON THEME

ABC

resources on approximate Bayesian computational methods

Search

Home

Home

This website keeps track of developments in approximate Bayesian computation (ABC) (a.k.a. likelihood-free), a class of computational statistical methods for Bayesian inference under intractable likelihoods. The site is meant to be a resource both for biologists and statisticians who want to learn more about ABC and related methods. Recent publications are under Publications 2012. A comprehensive list of publications can be found under Literature. If you are unfamiliar with ABC methods see the Introduction. Navigate using the menu to learn more.

ABC in Montreal | ABC in Montreal (2014)

ABC in Montreal

Approximate Bayesian computation (ABC) or likelihood-free (LF) methods have developed mostly beyond the radar of the machine learning community, but are important tools for a large and diverse segment of the scientific community. This is particularly true for systems and population biology, computational neuroscience, computer vision, healthcare sciences, but also many others.

Interaction between the ABC and machine learning community has recently started and contributed to important advances. In general, however, there is still significant room for more intense interaction and collaboration. Our workshop aims at being a place for this to happen.

ICML 2017 Workshop on Implicit Models

Workshop Aims

Probabilistic models are an important tool in machine learning. They form the basis for models that generate realistic data, uncover hidden structure, and make predictions. Traditionally, probabilistic models in machine learning have focused on prescribed models. Prescribed models specify a joint density over observed and hidden variables that can be easily evaluated. The requirement of a tractable density simplifies their learning but limits their flexibility --- several real world phenomena are better described by simulators that do not admit a tractable density. Probabilistic models defined only via the simulations they produce are called implicit models.

Arguably starting with generative adversarial networks, research on implicit models in machine learning has exploded in recent years. This workshop's aim is to foster a discussion around the recent developments and future directions of implicit models.

Implicit models have many applications. They are used in ecology where models simulate animal populations over time; they are used in phylogeny, where simulations produce hypothetical ancestry trees; they are used in physics to generate particle simulations for high energy processes. Recently, implicit models have been used to improve the state-of-the-art in image and content generation. Part of the workshop's focus is to discuss the commonalities among applications of implicit models.

Of particular interest at this workshop is to unite fields that work on implicit models. For example:

- **Generative adversarial networks** (a NIPS 2016 workshop) are implicit models with an adversarial training scheme.
- Recent advances in **variational inference** (a NIPS 2015 and 2016 workshop) have leveraged implicit models for more accurate approximations.
- **Approximate Bayesian computation** (a NIPS 2015 workshop) focuses on posterior inference for models with implicit likelihoods.
- Learning implicit models is deeply connected to **two sample testing, density ratio and density difference** estimation.

We hope to bring together these different views on implicit models, identifying their core challenges and combining their innovations.

THE PLAYERS

forward modeling
generation
simulation

(**z:** latent variables)

PREDICTION

θ
parameters of interest

p(x|θ, ν)

x
observed data
covariates
simulated data

INFERENCE

ν
nuisance parameters

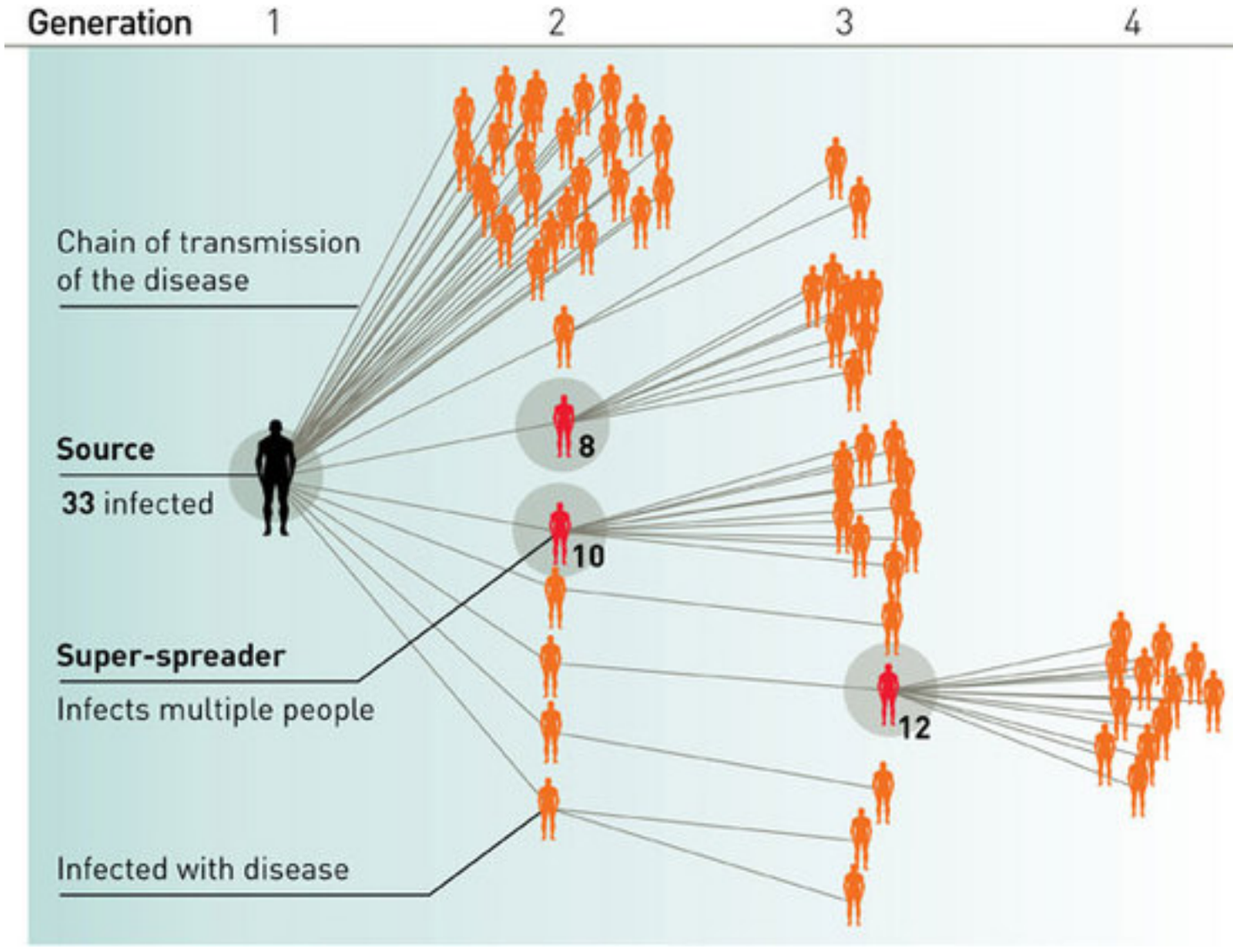
inverse problem
measurement
parameter estimation

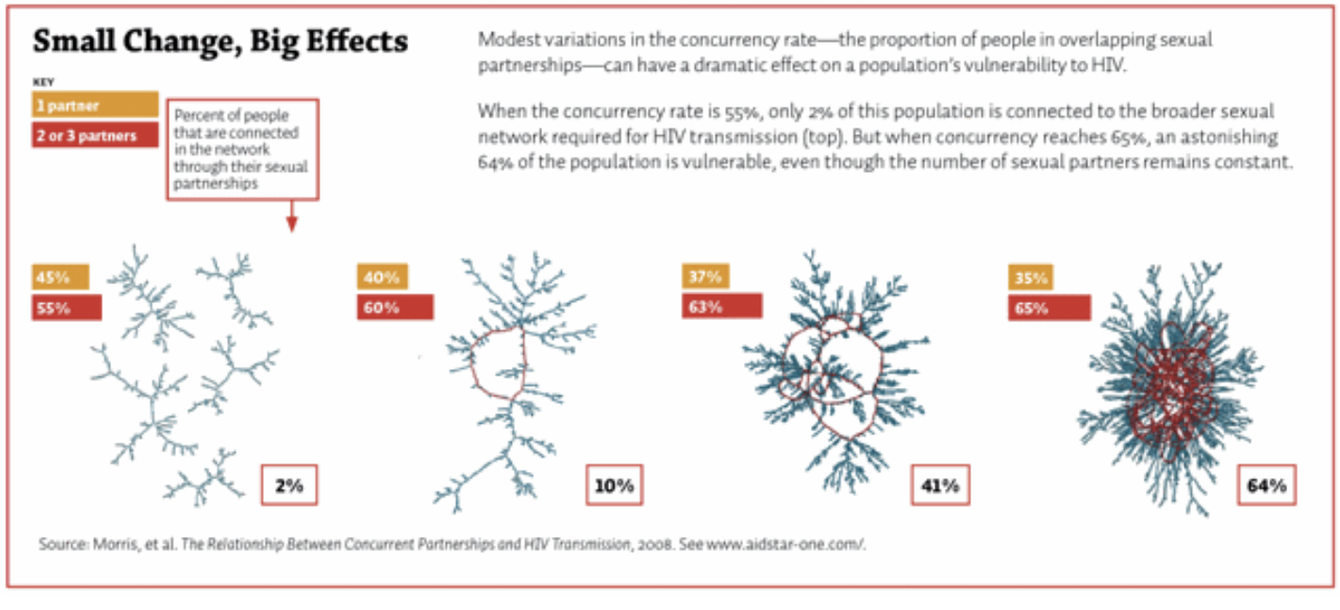
Syllabus Overview

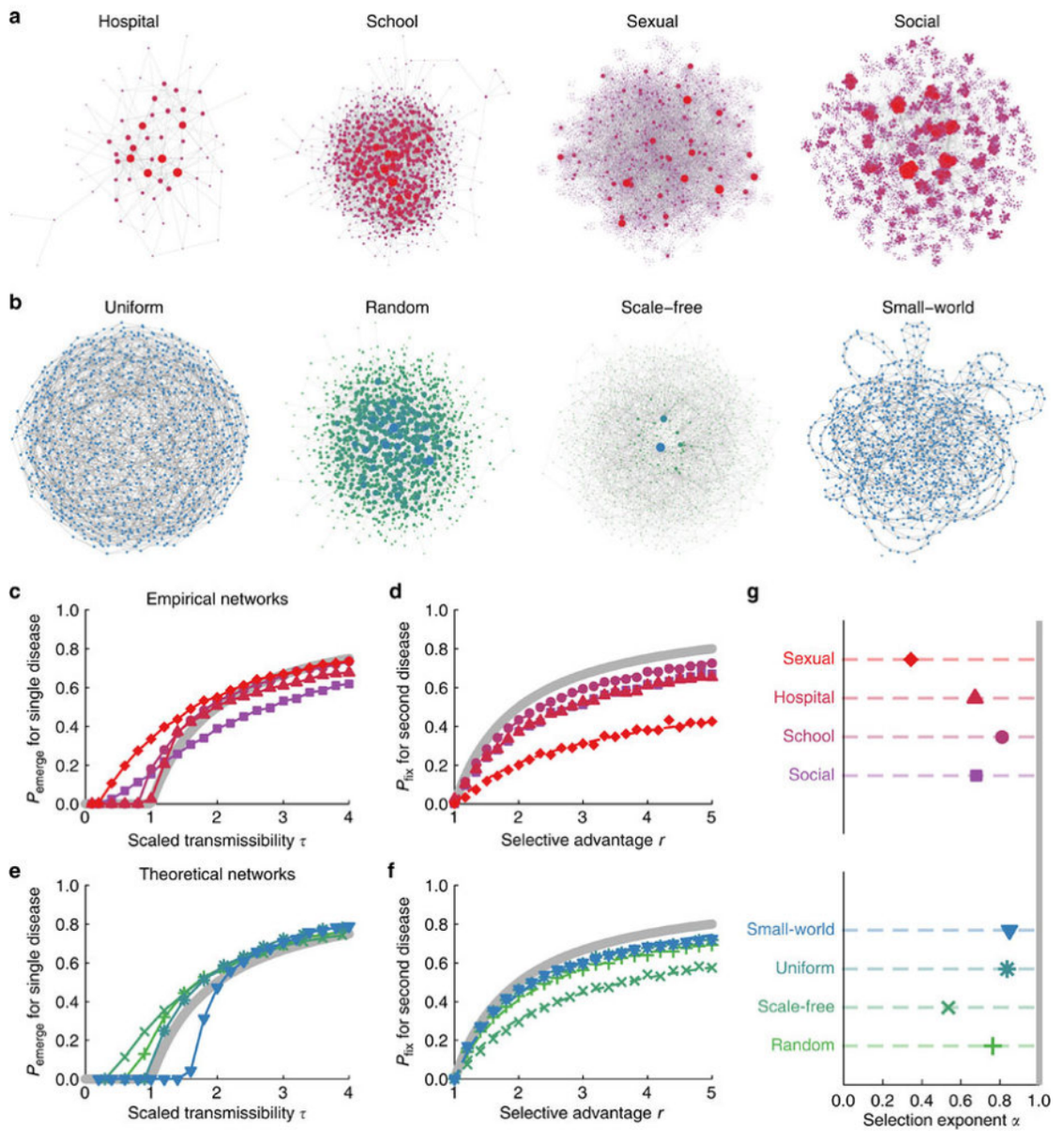
- Definitions, Directed Graphical Models and Bayesian Networks (lect1)
- Undirected Graphical Models. Markov Random Fields (lect2)
- Topic Models and Bayesian Non-parametrics (lect 3)
- Principal Component Analysis with Applications (lect 4)
- Expectation-Maximization. MCMC (lect 5)
- Variational Inference (lect 6)
- Structured Output Prediction (lect 7)
- Sequential Models (lect 8)
- Boltzmann Machines, Variational Autoencoders (lect 9)
- Modeling images. GANs, Flows and Autoregressive (lect 10)
- Modern unsupervised learning with GANs and VAEs (lect 11)

Examples of Simulators and Domain Specific Generative Models

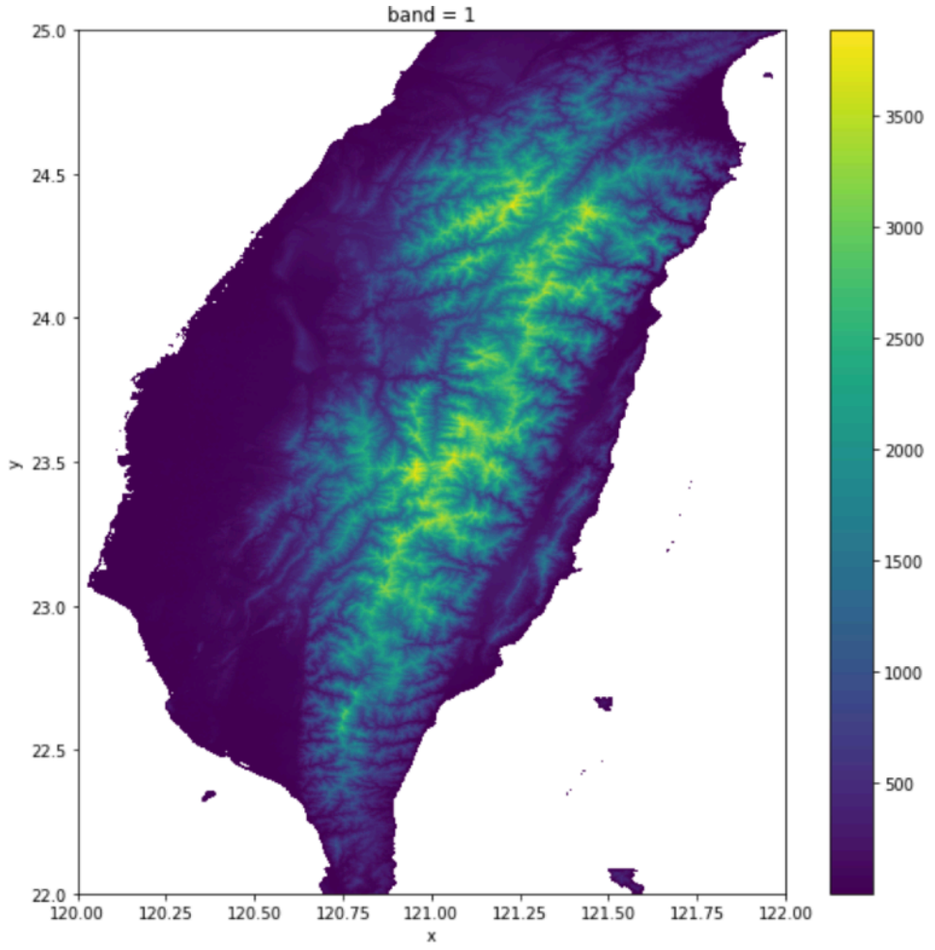
EPIDEMIOLOGY & POPULATION GENETICS

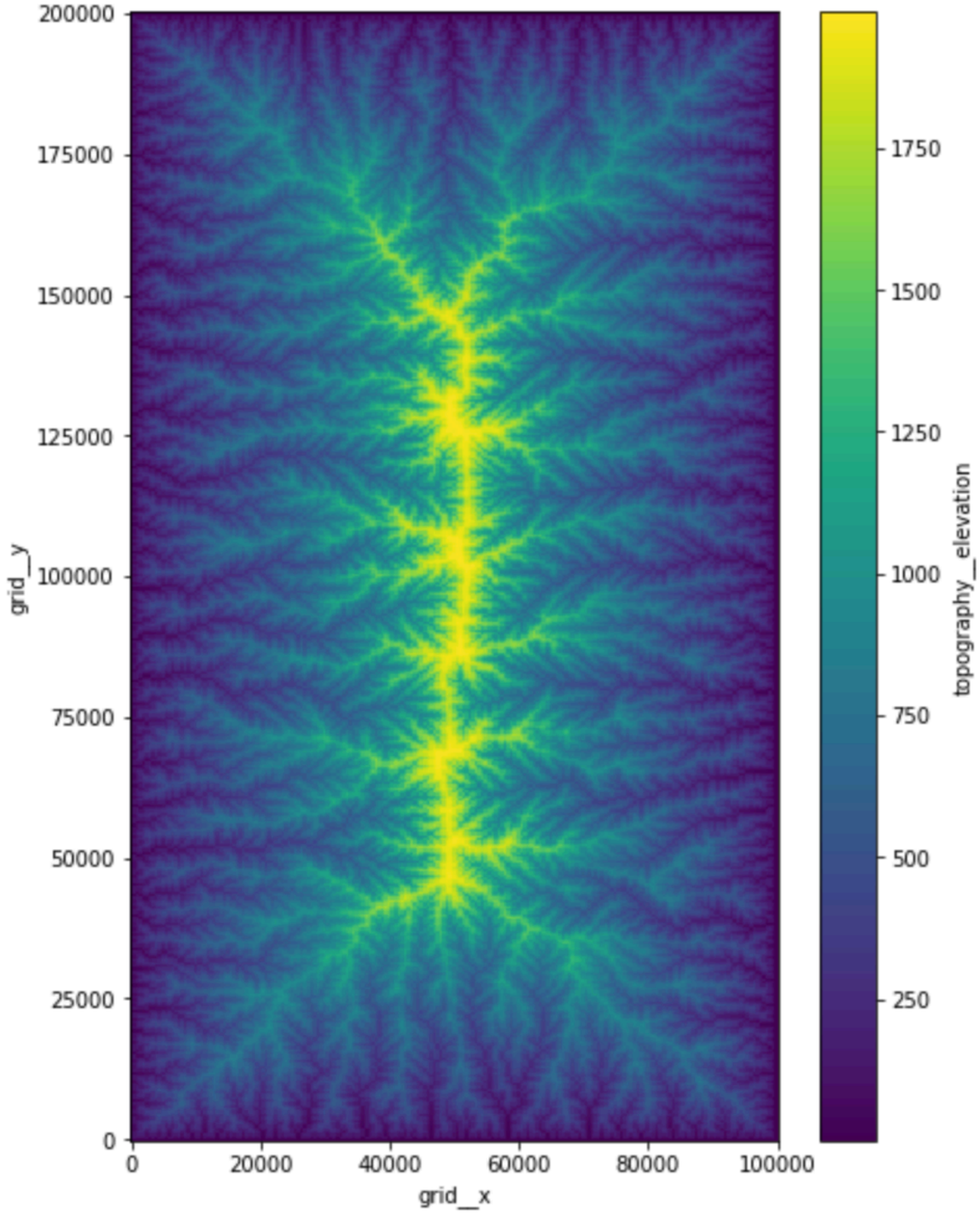


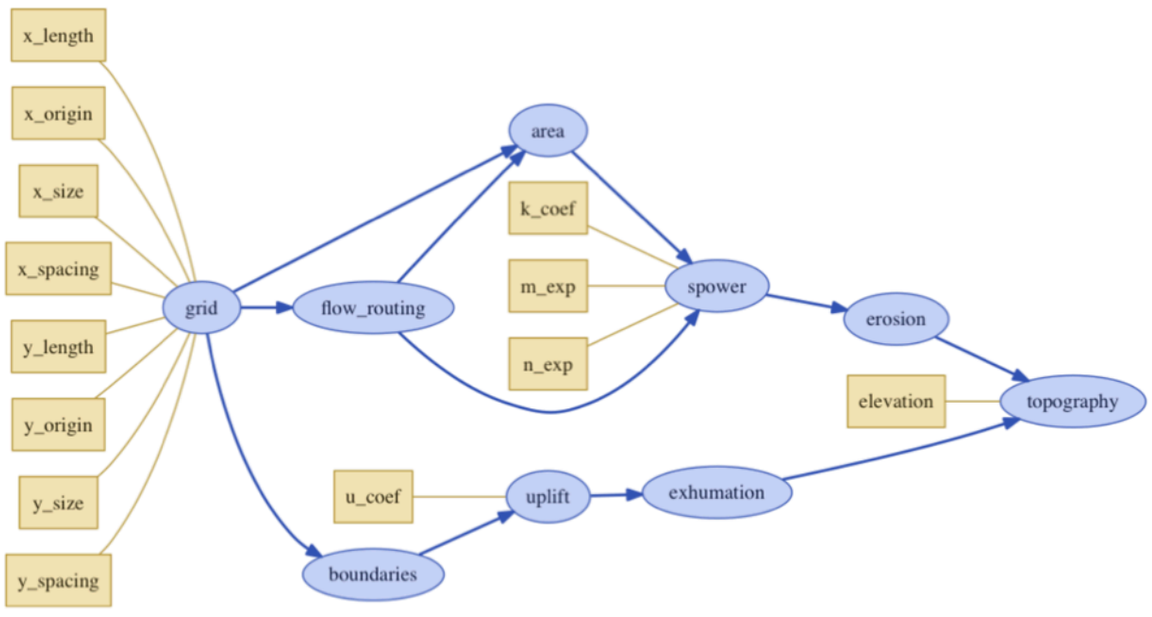




COMPUTATIONAL TOPOGRAPHY







We create a simulation setup for this model, run it, and then plot the final topography (after 1 million years of simulation).

The Forward Model

$$\mathcal{L}_{SM} = \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^{a}_{\mu\nu}G^{\mu\nu}_{a}}_{\text{kinetic energies and self-interactions of the gauge bosons}}$$

$$+ \underbrace{\bar{L}\gamma^{\mu}(i\partial_{\mu} - \frac{1}{2}g\tau\cdot\mathbf{W}_{\mu} - \frac{1}{2}g'YB_{\mu})L \;+\; \bar{R}\gamma^{\mu}(i\partial_{\mu} - \frac{1}{2}g'YB_{\mu})R}_{\text{kinetic energies and electroweak interactions of fermions}}$$

$$+ \underbrace{\frac{1}{2}\left|(i\partial_{\mu} - \frac{1}{2}g\tau\cdot\mathbf{W}_{\mu} - \frac{1}{2}g'YB_{\mu})\,\phi\right|^{2} - V(\phi)}_{W^{\pm},Z,\gamma,\text{and Higgs masses and couplings}}$$

$$+ \underbrace{g''(\bar{q}\gamma^{\mu}T_{a}q)\,G^{a}_{\mu}}_{\text{interactions between quarks and gluons}} + \underbrace{(G_1\bar{L}\phi R + G_2\bar{L}\phi_c R + h.c.)}_{\text{fermion masses and couplings to Higgs}}$$

1) We begin with Quantum Field Theory

The Forward Model

$$\mathcal{L}_{SM} = \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^a_{\mu\nu}G^{\mu\nu}_a}_{\text{kinetic energies and self-interactions of the gauge bosons}}$$

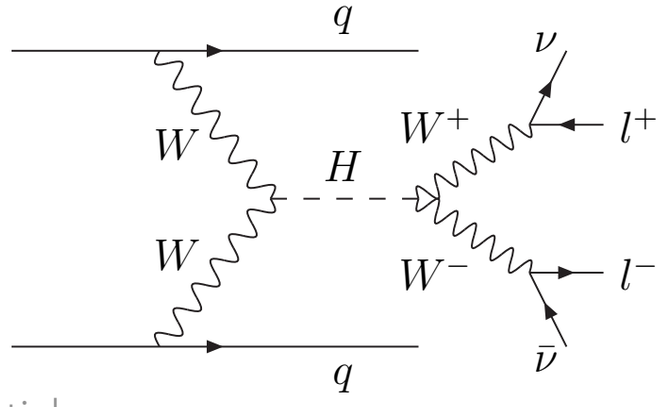
$$+ \underbrace{\bar{L}\gamma^\mu(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)L + \bar{R}\gamma^\mu(i\partial_\mu - \frac{1}{2}g'YB_\mu)R}_{\text{kinetic energies and electroweak interactions of fermions}}$$

$$+ \underbrace{\frac{1}{2}\left|(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)\,\phi\right|^2 - V(\phi)}_{W^\pm, Z, \gamma, \text{and Higgs masses and couplings}}$$

$$+ \underbrace{g''(\bar{q}\gamma^\mu T_a q)\,G^a_\mu}_{\text{interactions between quarks and gluons}} + \underbrace{(G_1\bar{L}\phi R + G_2\bar{L}\phi_c R + h.c.)}_{\text{fermion masses and couplings to Higgs}}$$

1) We begin with Quantum Field Theory

2) Theory gives detailed prediction for high-energy collisions



hierarchical: 2 → O(10) → O(100) particles

The Forward Model

$$\mathcal{L}_{SM} = \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^a_{\mu\nu}G^{\mu\nu}_a}_{\text{kinetic energies and self-interactions of the gauge bosons}}$$

$$+ \underbrace{\bar{L}\gamma^\mu(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)L + \bar{R}\gamma^\mu(i\partial_\mu - \frac{1}{2}g'YB_\mu)R}_{\text{kinetic energies and electroweak interactions of fermions}}$$

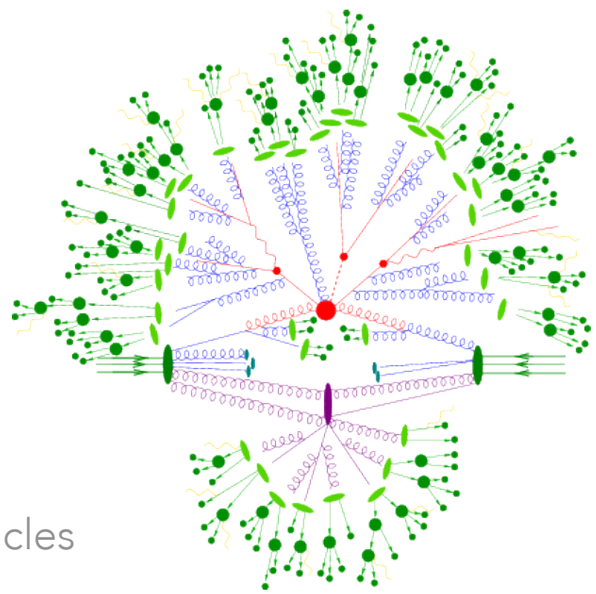
$$+ \underbrace{\frac{1}{2}\left|(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)\,\phi\right|^2 - V(\phi)}_{W^\pm, Z, \gamma, \text{and Higgs masses and couplings}}$$

$$+ \underbrace{g''(\bar{q}\gamma^\mu T_a q)\,G^a_\mu}_{\text{interactions between quarks and gluons}} + \underbrace{(G_1\bar{L}\phi R + G_2\bar{L}\phi_c R + h.c.)}_{\text{fermion masses and couplings to Higgs}}$$

1) We begin with Quantum Field Theory

2) Theory gives detailed prediction for high-energy collisions

hierarchical: 2 → O(10) → O(100) particles



The Forward Model

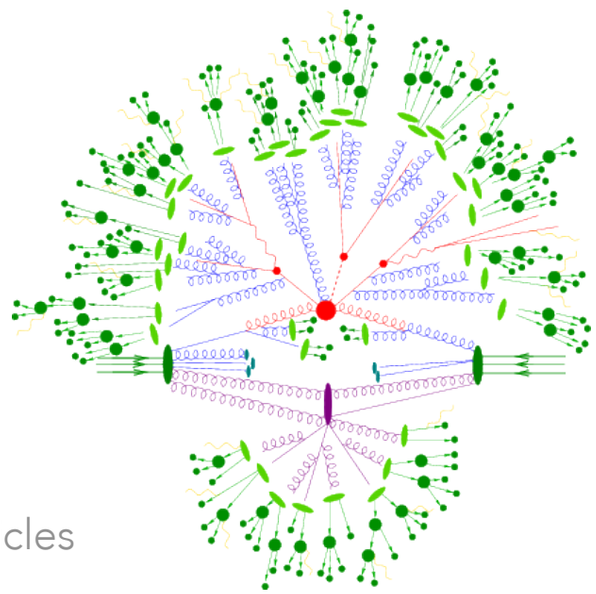
$$\mathcal{L}_{SM} = \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^a_{\mu\nu}G^{\mu\nu}_a}_{\text{kinetic energies and self-interactions of the gauge bosons}}$$

$$+ \underbrace{\bar{L}\gamma^\mu(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)L + \bar{R}\gamma^\mu(i\partial_\mu - \frac{1}{2}g'YB_\mu)R}_{\text{kinetic energies and electroweak interactions of fermions}}$$

$$+ \underbrace{\frac{1}{2}\left|(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)\,\phi\right|^2 - V(\phi)}_{W^\pm, Z, \gamma, \text{and Higgs masses and couplings}}$$

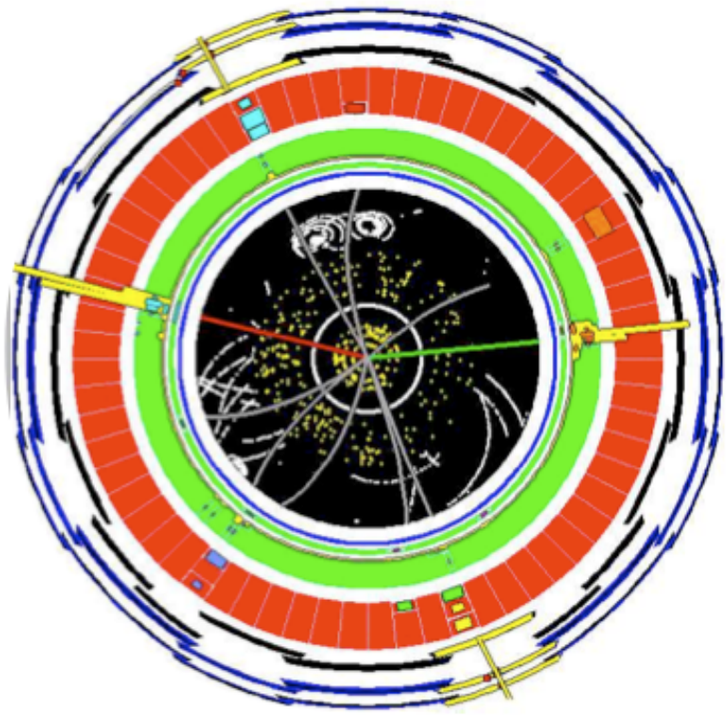
$$+ \underbrace{g''(\bar{q}\gamma^\mu T_a q)\,G^a_\mu}_{\text{interactions between quarks and gluons}} + \underbrace{(G_1\bar{L}\phi R + G_2\bar{L}\phi_c R + h.c.)}_{\text{fermion masses and couplings to Higgs}}$$

1) We begin with Quantum Field Theory



2) Theory gives detailed prediction for high-energy collisions

hierarchical: 2 → O(10) → O(100) particles



3) The interaction of outgoing particles with the detector is simulated.

>100 million sensors

The Forward Model

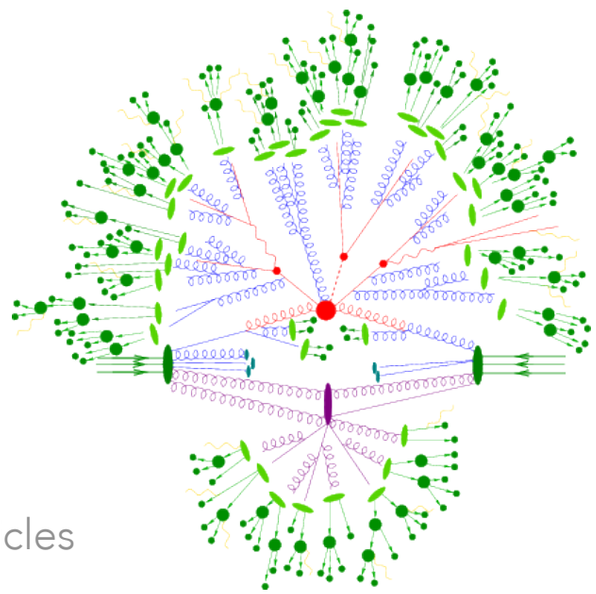
$$\mathcal{L}_{SM} = \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^a_{\mu\nu}G^{\mu\nu}_a}_{\text{kinetic energies and self-interactions of the gauge bosons}}$$

$$+ \underbrace{\bar{L}\gamma^\mu(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)L + \bar{R}\gamma^\mu(i\partial_\mu - \frac{1}{2}g'YB_\mu)R}_{\text{kinetic energies and electroweak interactions of fermions}}$$

$$+ \underbrace{\frac{1}{2}\left|(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)\,\phi\right|^2 - V(\phi)}_{W^\pm, Z, \gamma, \text{and Higgs masses and couplings}}$$

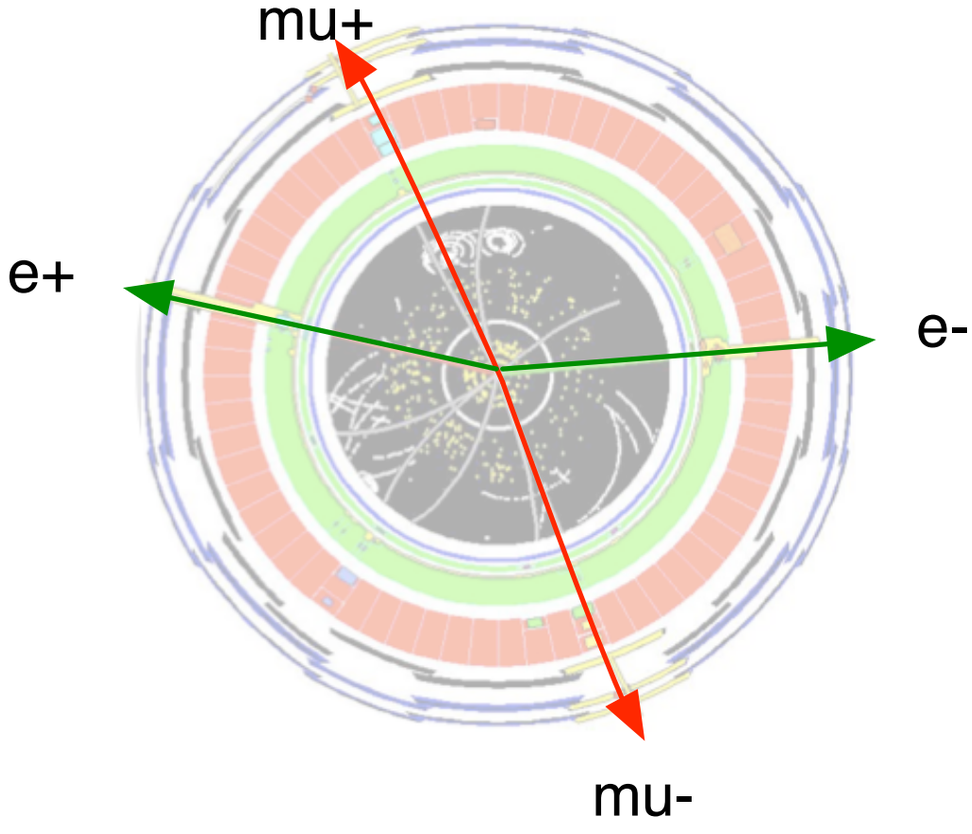
$$+ \underbrace{g''(\bar{q}\gamma^\mu T_a q)\,G^a_\mu}_{\text{interactions between quarks and gluons}} + \underbrace{(G_1\bar{L}\phi R + G_2\bar{L}\phi_c R + h.c.)}_{\text{fermion masses and couplings to Higgs}}$$

1) We begin with Quantum Field Theory



2) Theory gives detailed prediction for high-energy collisions

hierarchical: 2 → O(10) → O(100) particles



3) The interaction of outgoing particles with the detector is simulated.

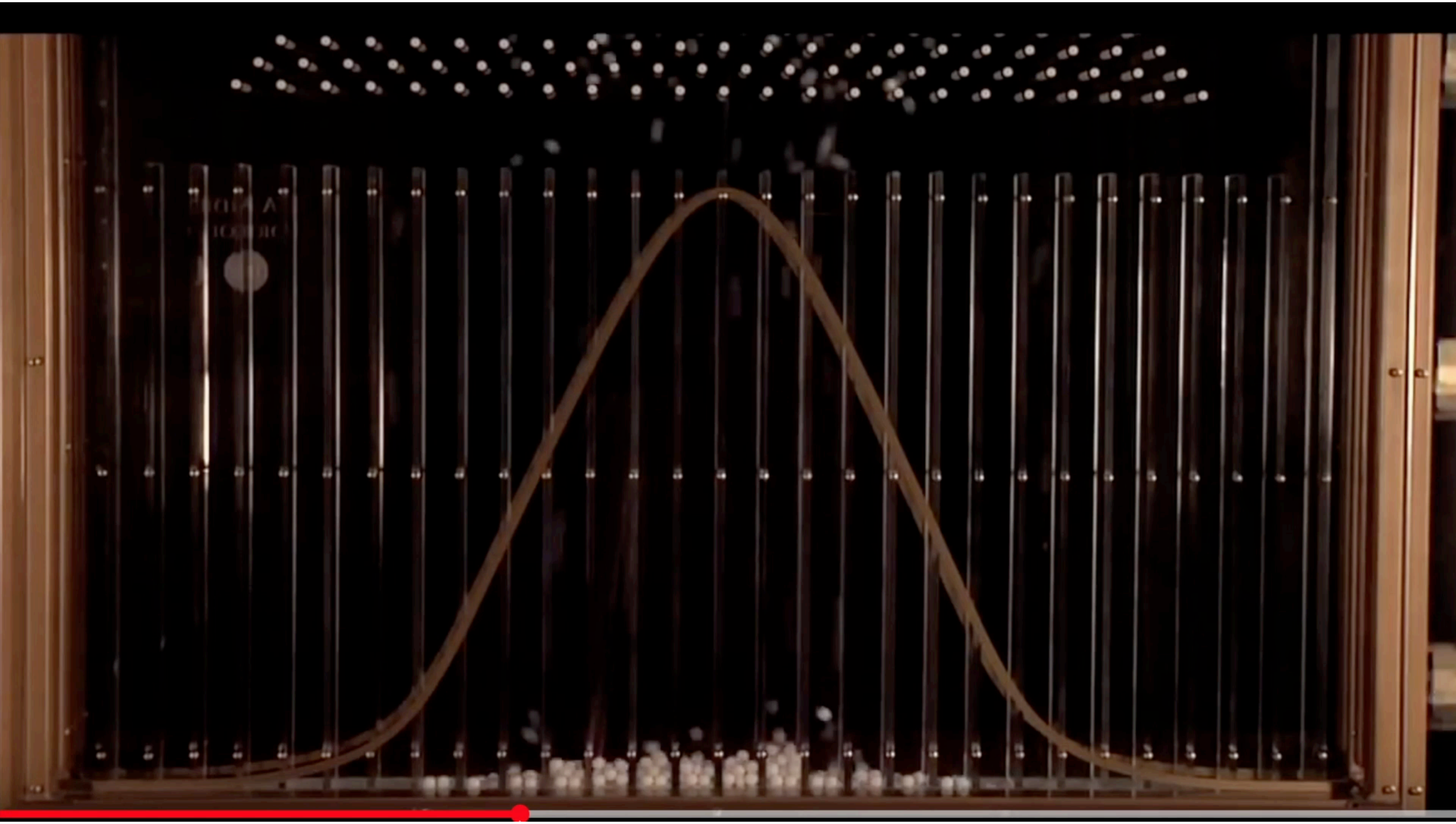
>100 million sensors

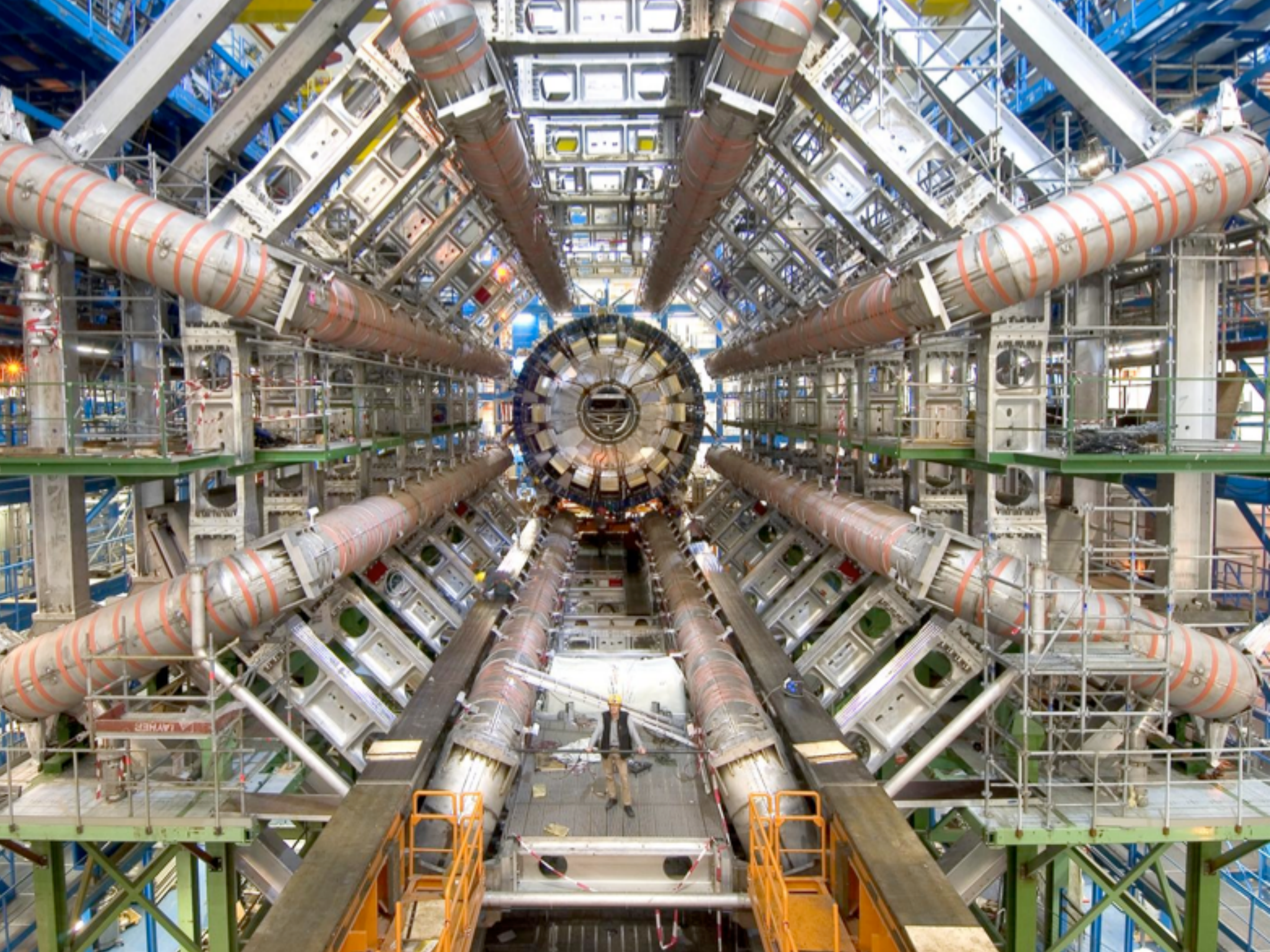
4) Finally, we run particle identification and feature extraction algorithms on the simulated data as if they were from real collisions.

~10-30 features describe interesting part

PLINKO

If all the nails are centered, we can solve it analytically,
but what if each of the nails were slightly misplaced?





FUNDAMENTAL PARTICLES & INTERACTIONS

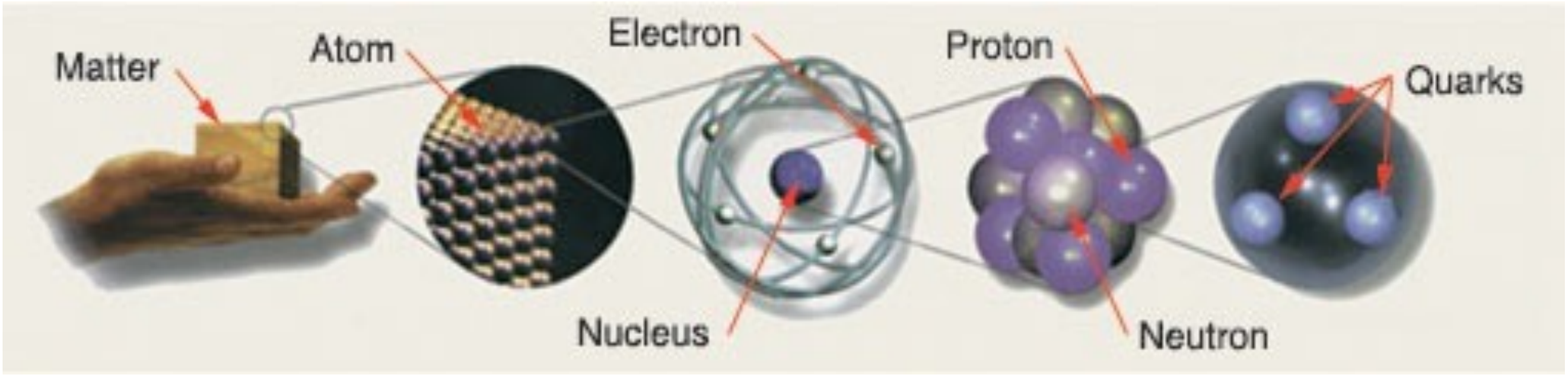
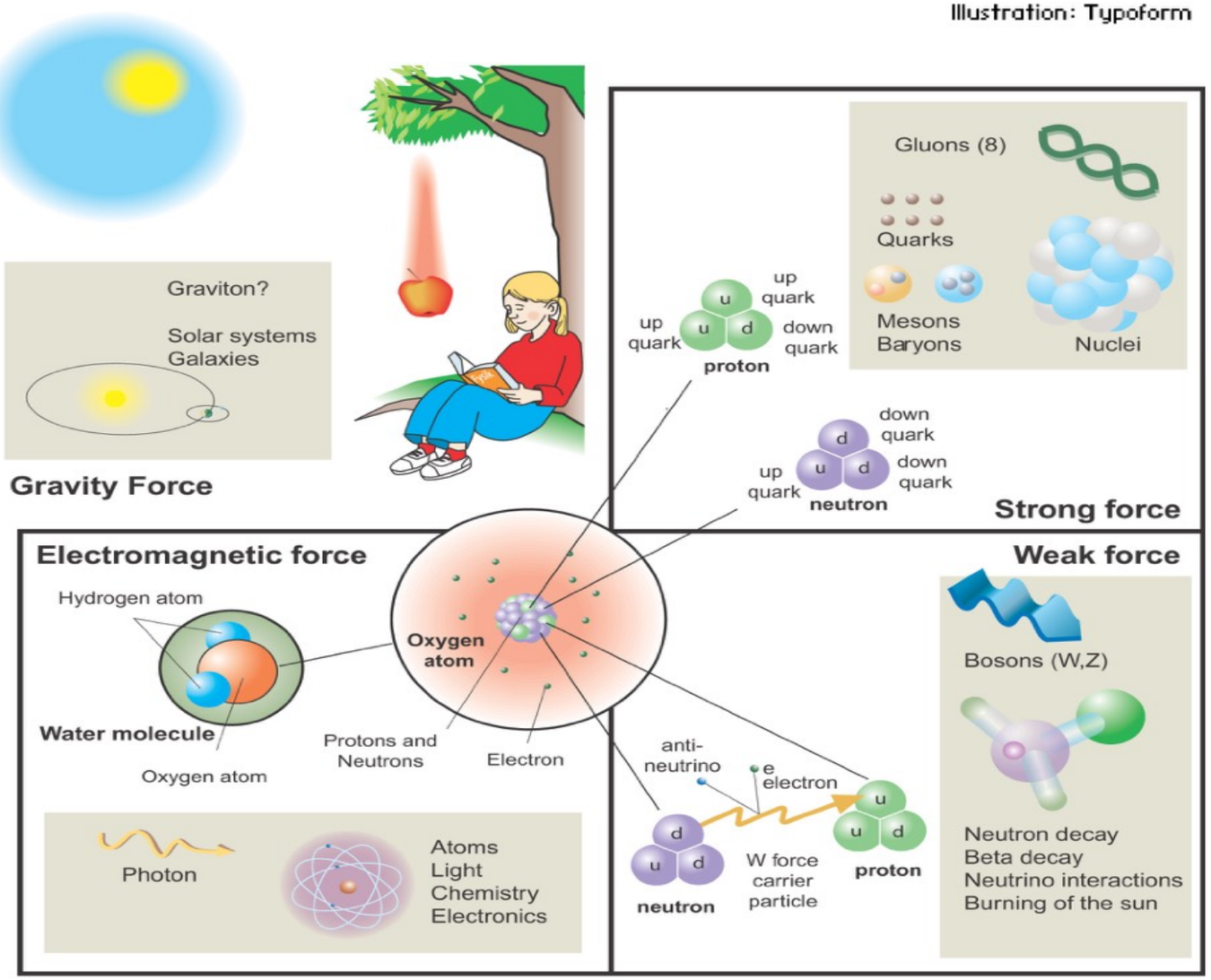
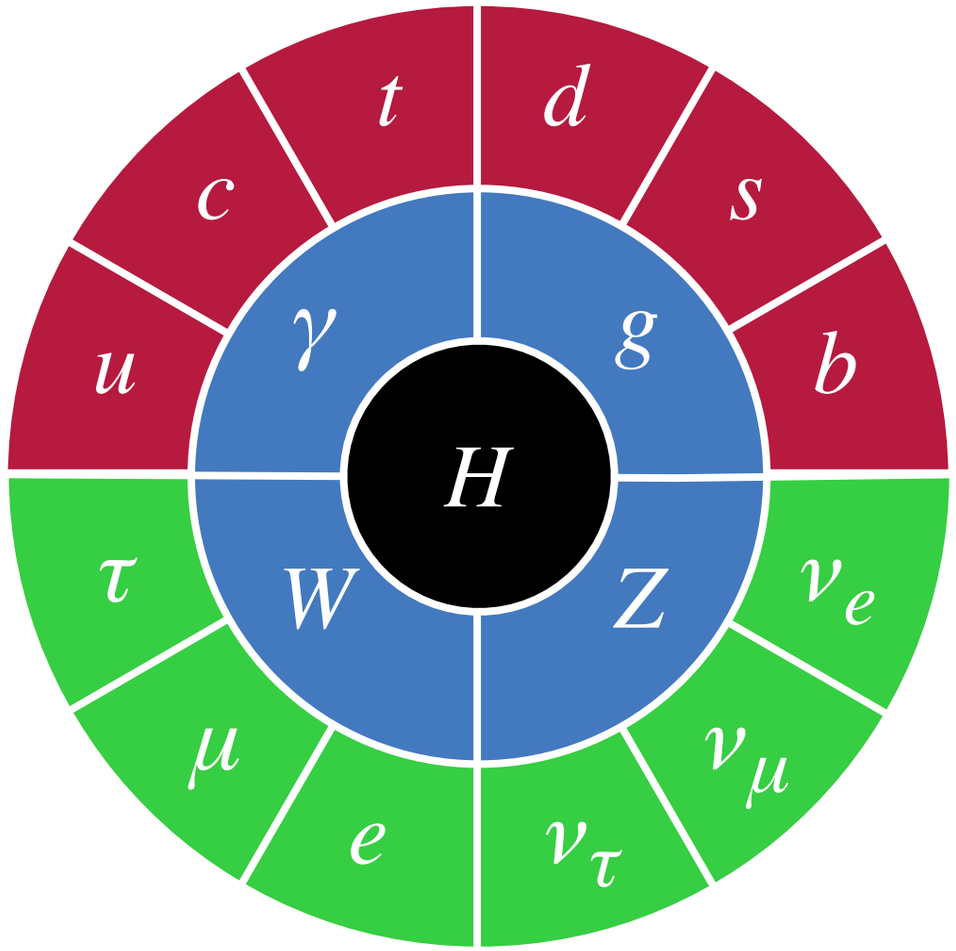
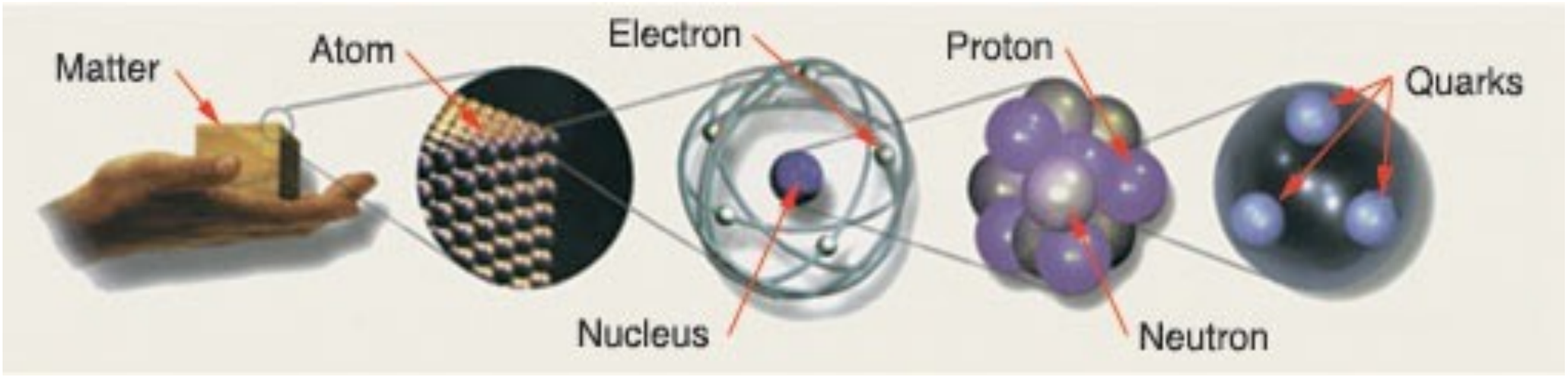


Illustration: Typoform





FUNDAMENTAL PARTICLES & INTERACTIONS

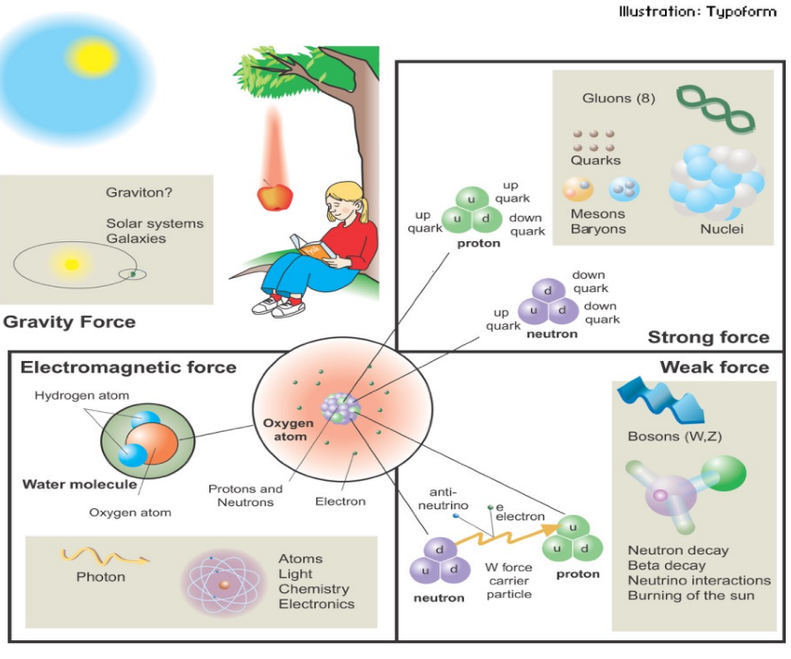


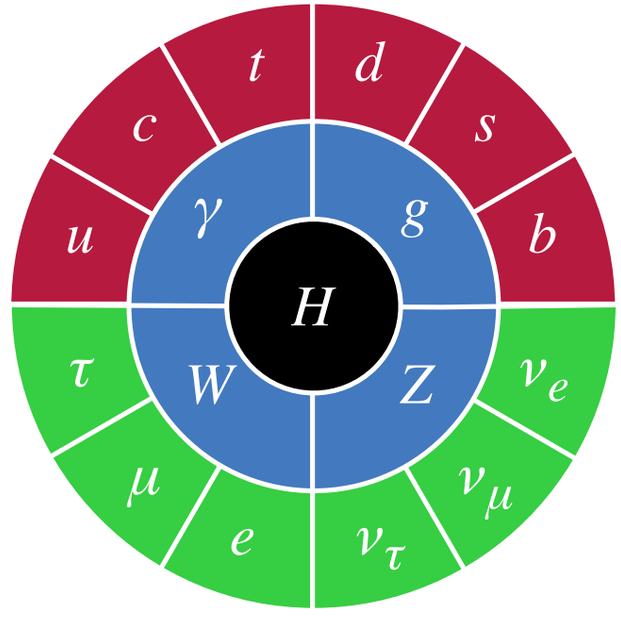
$$\mathcal{L}_{SM} = \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^{a}_{\mu\nu}G^{\mu\nu}_{a}}_{\text{kinetic energies and self-interactions of the gauge bosons}}$$

$$+ \underbrace{\bar{L}\gamma^{\mu}(i\partial_{\mu} - \frac{1}{2}g\tau\cdot\mathbf{W}_{\mu} - \frac{1}{2}g'YB_{\mu})L + \bar{R}\gamma^{\mu}(i\partial_{\mu} - \frac{1}{2}g'YB_{\mu})R}_{\text{kinetic energies and electroweak interactions of fermions}}$$

$$+ \underbrace{\frac{1}{2}\left|(i\partial_{\mu} - \frac{1}{2}g\tau\cdot\mathbf{W}_{\mu} - \frac{1}{2}g'YB_{\mu})\,\phi\right|^{2} - V(\phi)}_{W^{\pm},Z,\gamma,\text{and Higgs masses and couplings}}$$

$$+ \underbrace{g''(\bar{q}\gamma^{\mu}T_{a}q)\,G^{a}_{\mu}}_{\text{interactions between quarks and gluons}} + \underbrace{(G_{1}\bar{L}\phi R + G_{2}\bar{R}\phi_{c}L + h.c.)}_{\text{fermion masses and couplings to Higgs}}$$





Examples of Learning a Generative Model

THE PLAYERS

forward modeling
generation
simulation

PREDICTION

θ
parameters of interest

$p(x, z \mid \theta, \nu)$

x
observed data
simulated data

z
latent variables
Monte Carlo truth

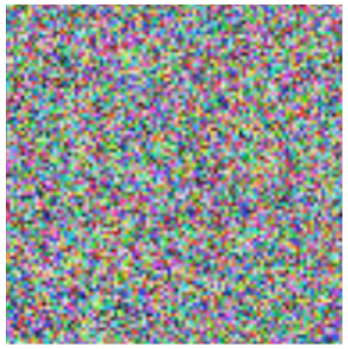
ν
nuisance parameters

INFERENCE

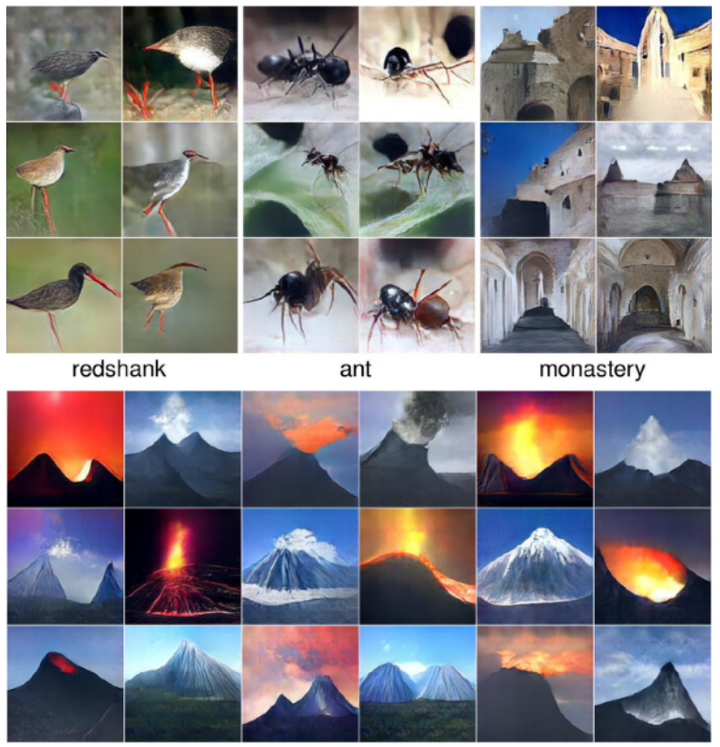
inverse problem
measurement
parameter estimation

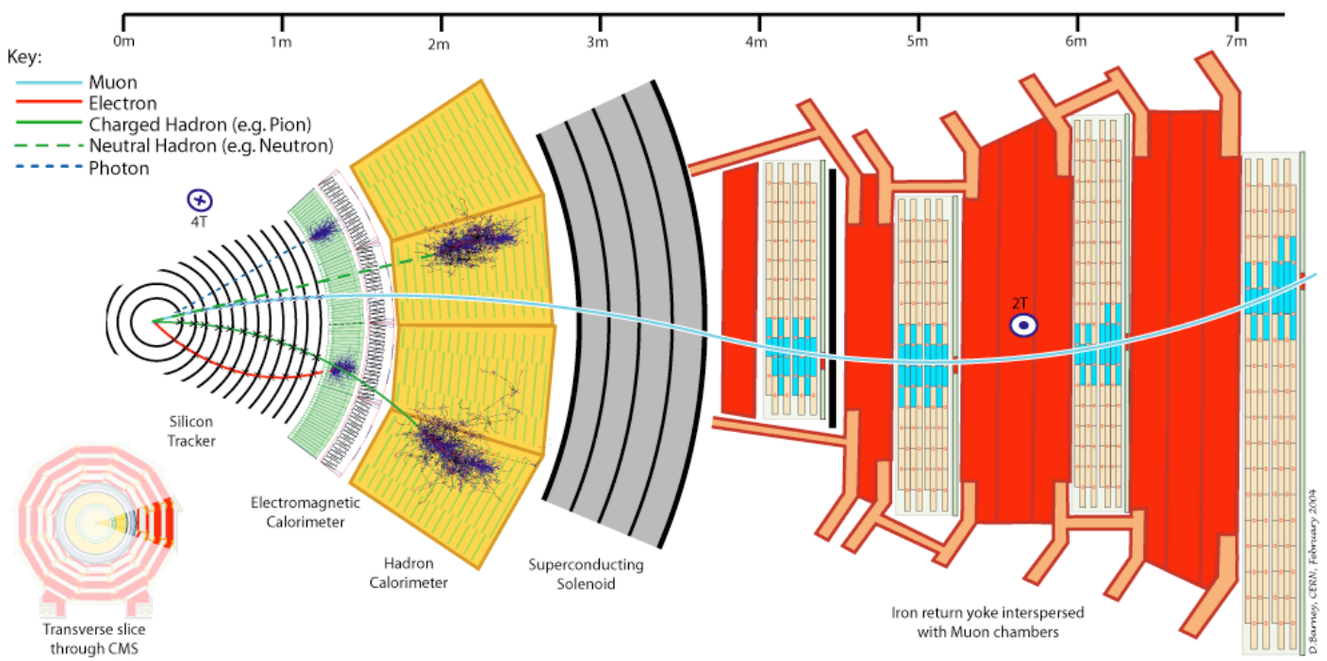
LEARNING THE GENERATIVE MODEL

Noise ~ N(0,1)



Generative Model

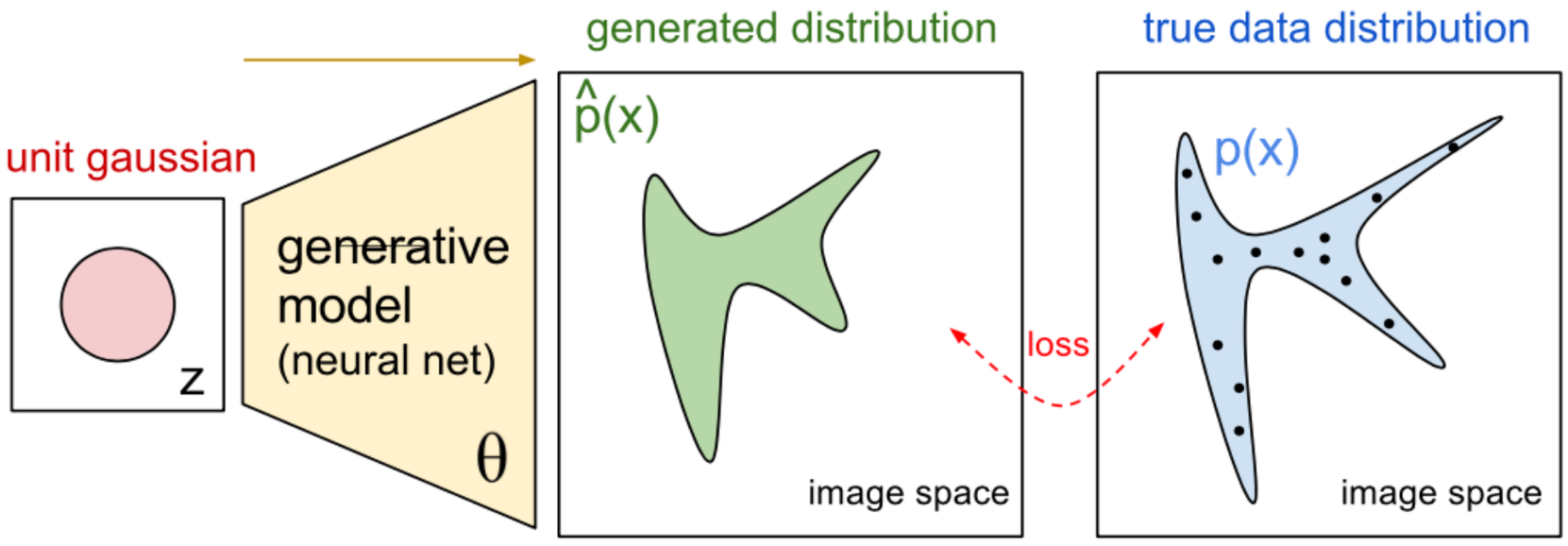




http://torch.ch/blog/2015/11/13/gan.html

GENERATIVE ADVERSARIAL NETWORKS

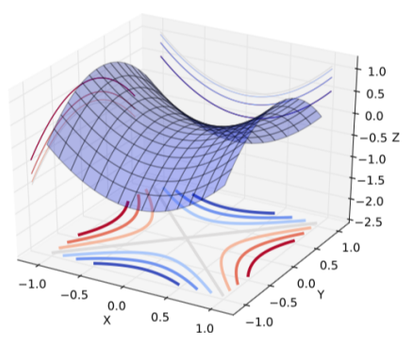
Goodfellow, et al arXiv:1406.2661
slide Gilles Louppe



- Two-player game:
  - a discriminator D,
  - a generator G;
- D is a classifier $\mathcal{X} \mapsto \{0, 1\}$ that tries to distinguish between
  - a sample from the data distribution ($D(\mathbf{x}) = 1$, for $\mathbf{x} \sim p_{\text{data}}$),
  - and a sample from the model distribution ($D(G(\mathbf{z})) = 0$, for $\mathbf{z} \sim p_{\text{noise}}$);
- G is a generator $\mathcal{Z} \mapsto \mathcal{X}$ trained to produce samples $G(\mathbf{z})$ (for $\mathbf{z} \sim p_{\text{noise}}$) that are difficult for D to distinguish from data.

$$(D^*, G^*) = \max_D \min_G V(D, G).$$

Leo is G Tom is D



GANs FOR PHYSICS

CaloGAN: Simulating 3D High Energy Particle Showers in Multi-Layer Electromagnetic Calorimeters with Generative Adversarial Networks

Michela Paganini[a,b], **Luke de Oliveira**[a], **and Benjamin Nachman**[a]

[a] *Lawrence Berkeley National Laboratory, 1 Cyclotron Rd, Berkeley, CA, 94720, USA*
[b] *Department of Physics, Yale University, New Haven, CT 06520, USA*

E-mail: michela.paganini@yale.edu, lukedeoliveira@lbl.gov, bnachman@cern.ch

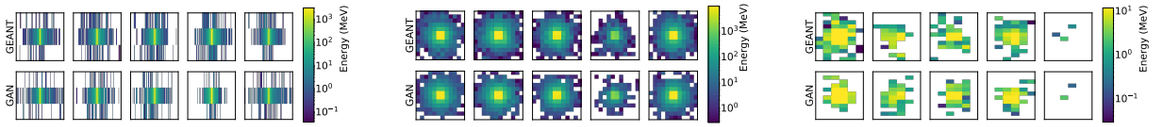


Figure 9: Five randomly selected e^+ showers per calorimeter layer from the training set (top) and the five nearest neighbors (by euclidean distance) from a set of CaloGAN candidates.

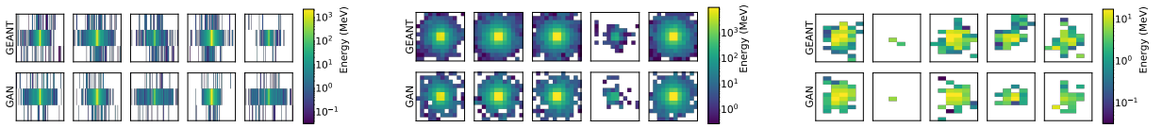


Figure 10: Five randomly selected γ showers per calorimeter layer from the training set (top) and the five nearest neighbors (by euclidean distance) from a set of CaloGAN candidates.

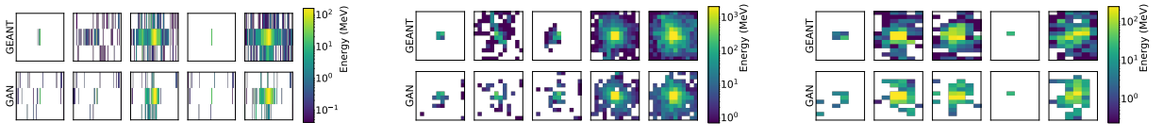
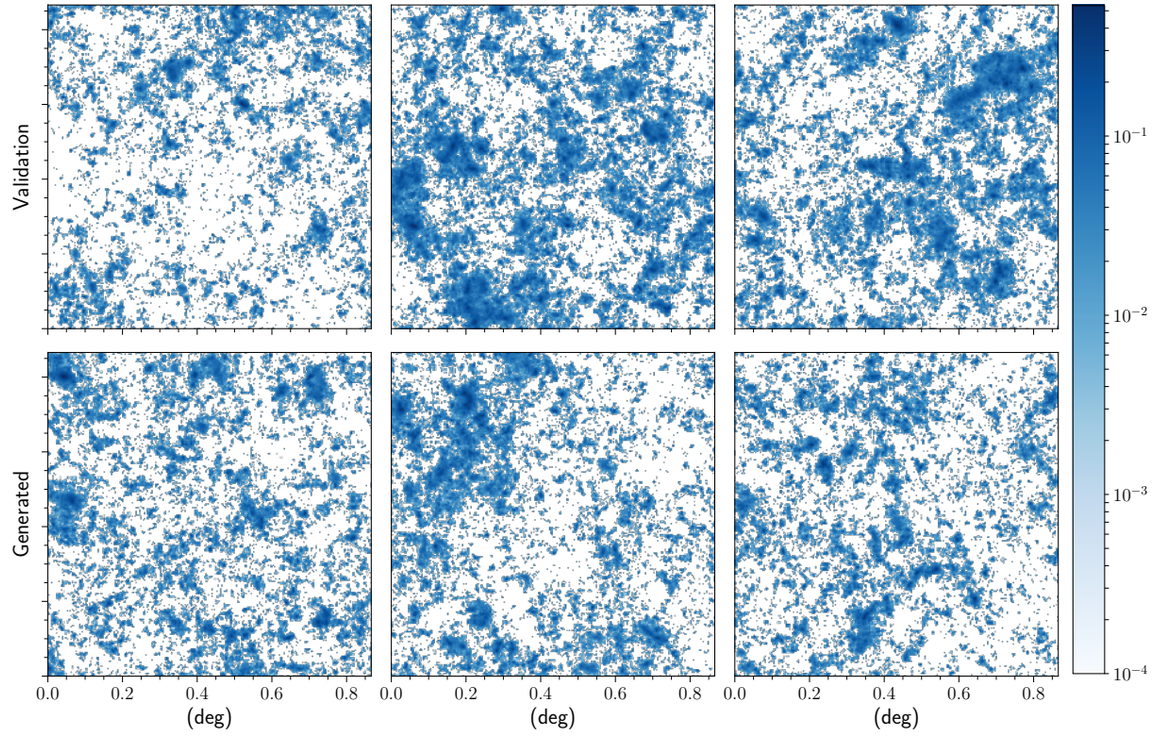


Figure 11: Five randomly selected π^+ showers per calorimeter layer from the training set (top) and the five nearest neighbors (by euclidean distance) from a set of CaloGAN candidates.

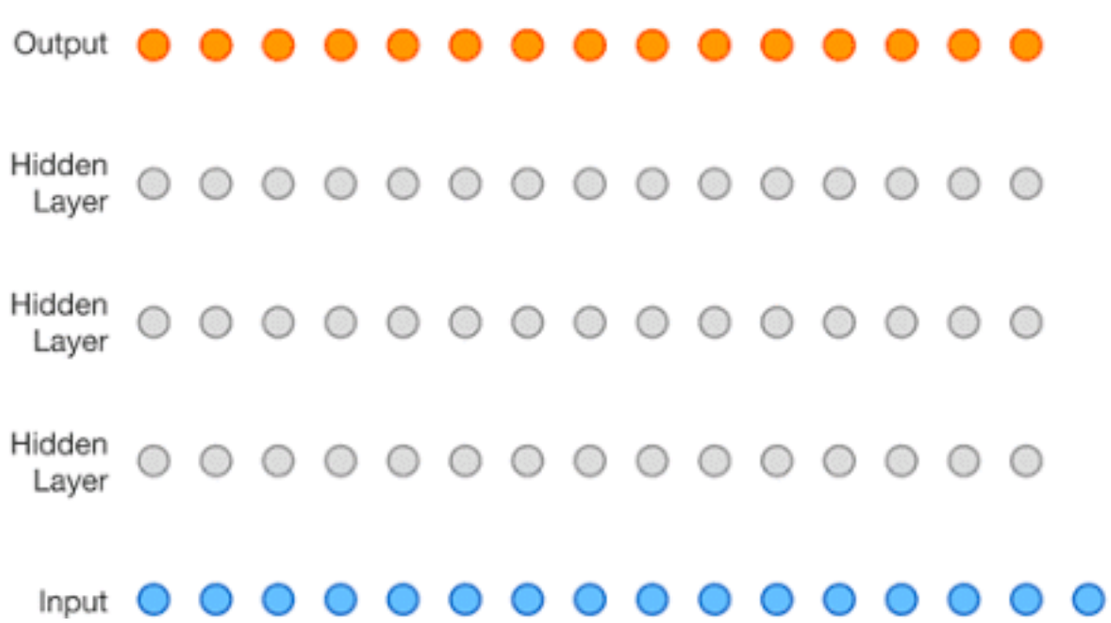
Creating Virtual Universes Using Generative Adversarial Networks

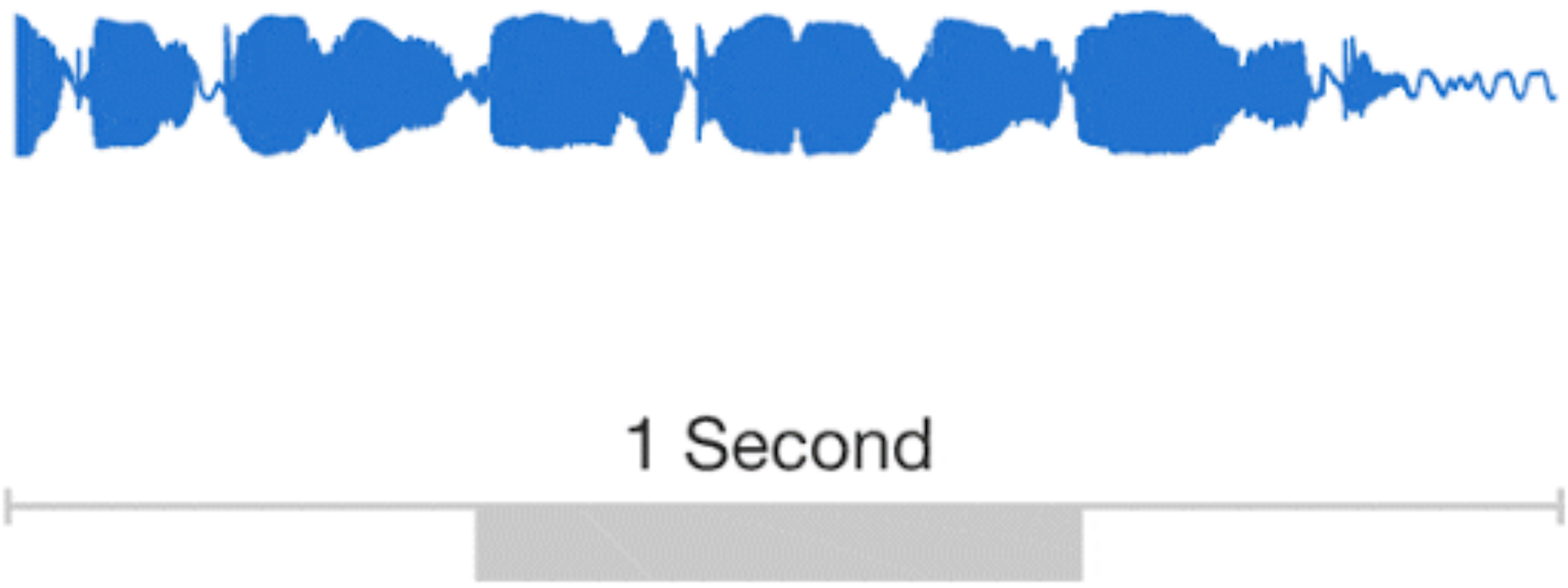
Mustafa Mustafa*[1], Deborah Bard[1], Wahid Bhimji[1], Rami Al-Rfou[2], and Zarija Lukić[1]

[1]Lawrence Berkeley National Laboratory, Berkeley, CA 94720
[2]Google Research, Mountain View, CA 94043



WAVENET: A GENERATIVE MODEL FOR RAW AUDIO





An example of Inference

The Higgs Boson

THE PLAYERS

forward modeling
generation
simulation

PREDICTION

θ
parameters of interest

$p(x, z \mid \theta, \nu)$

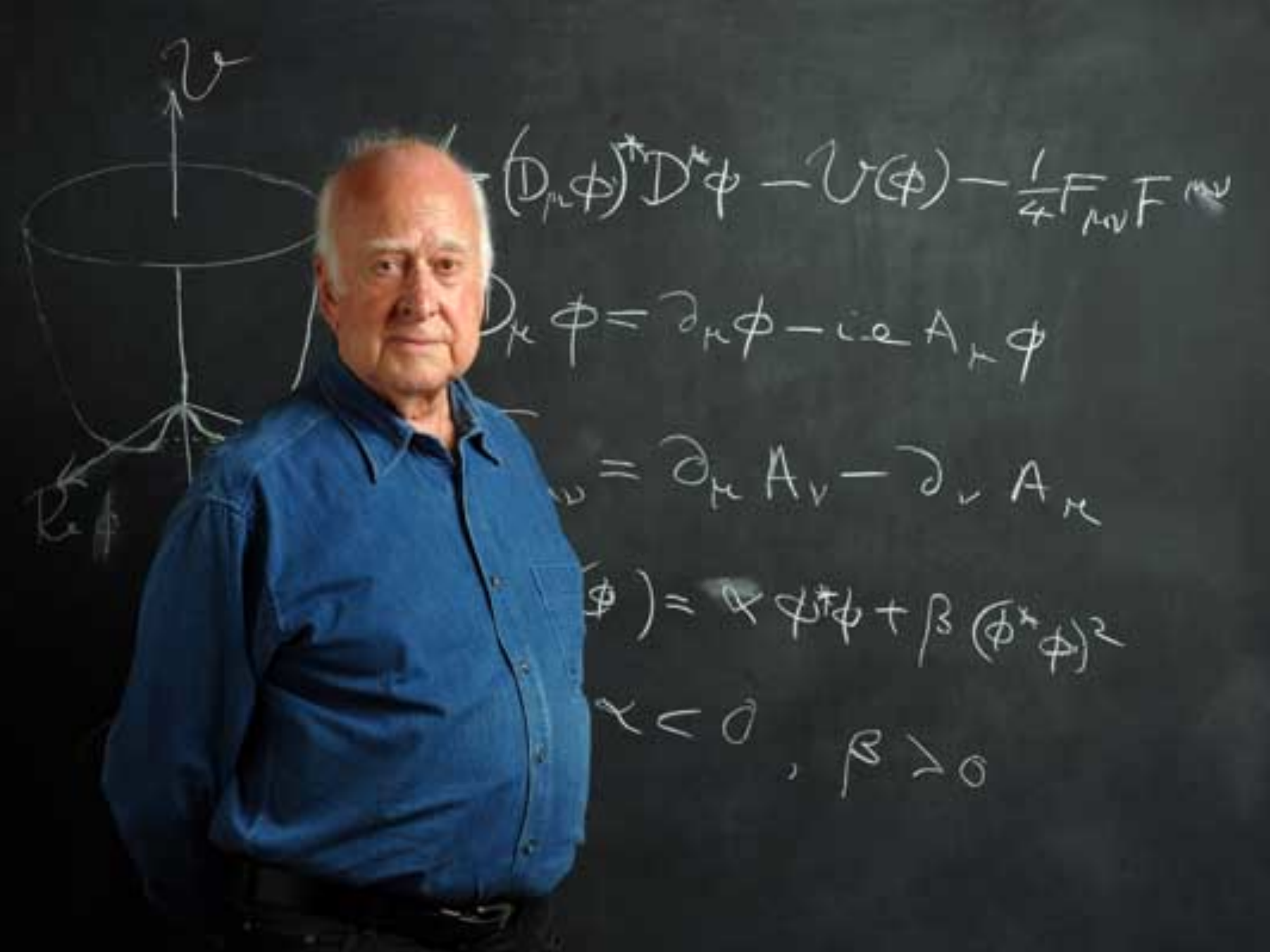
z
latent variables
Monte Carlo truth

x
observed data
simulated data

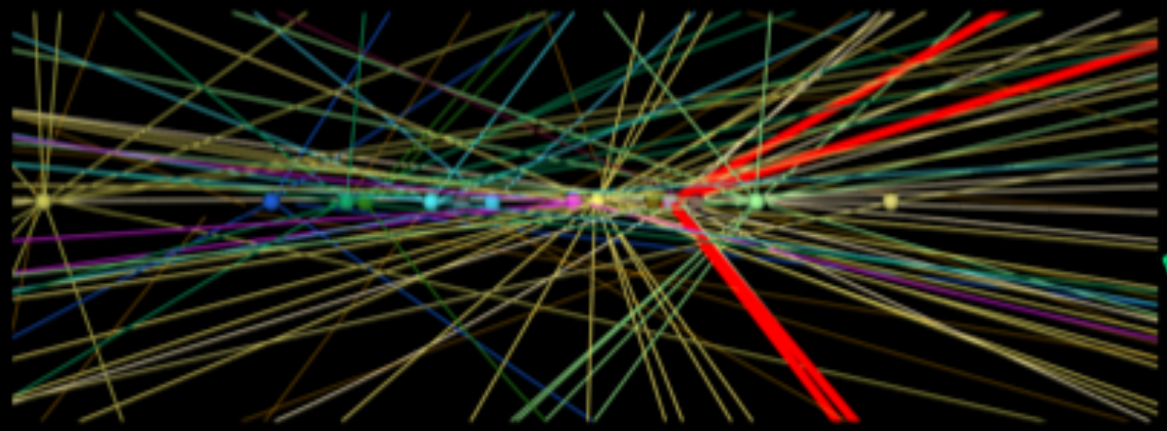
ν
nuisance parameters

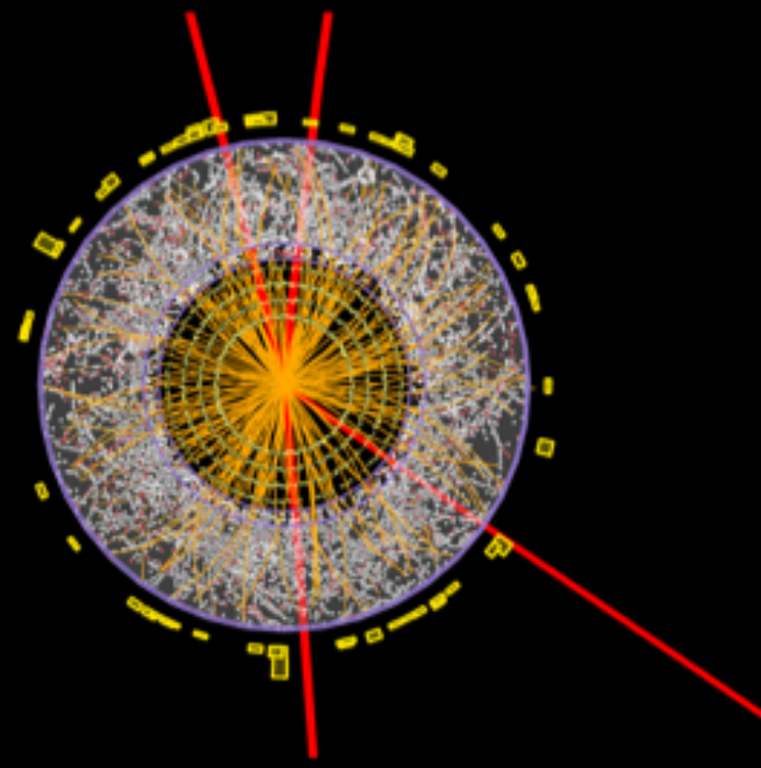
INFERENCE

inverse problem
measurement
parameter estimation


$(D_\mu\phi)^* D^\mu\phi - V(\phi) - \frac{1}{4}F_{\mu\nu}F^{\mu\nu}$
$D_\mu\phi = \partial_\mu\phi - ieA_\mu\phi$
$= \partial_\mu A_\nu - \partial_\nu A_\mu$
$\phi) = \alpha\phi^*\phi + \beta(\phi^*\phi)^2$
$\alpha < 0$
$, \beta > 0$

$H \to ZZ \to 4l$

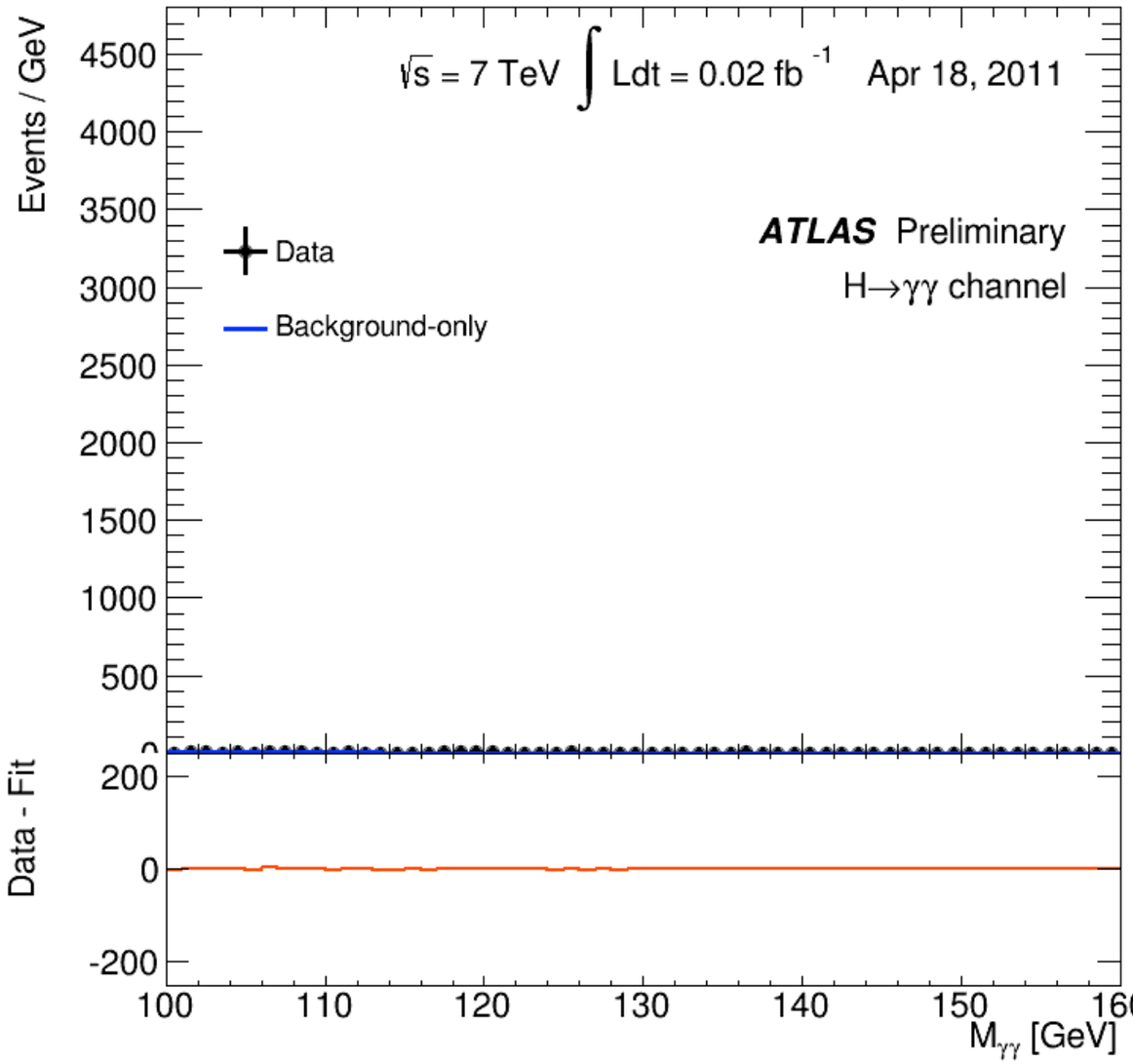


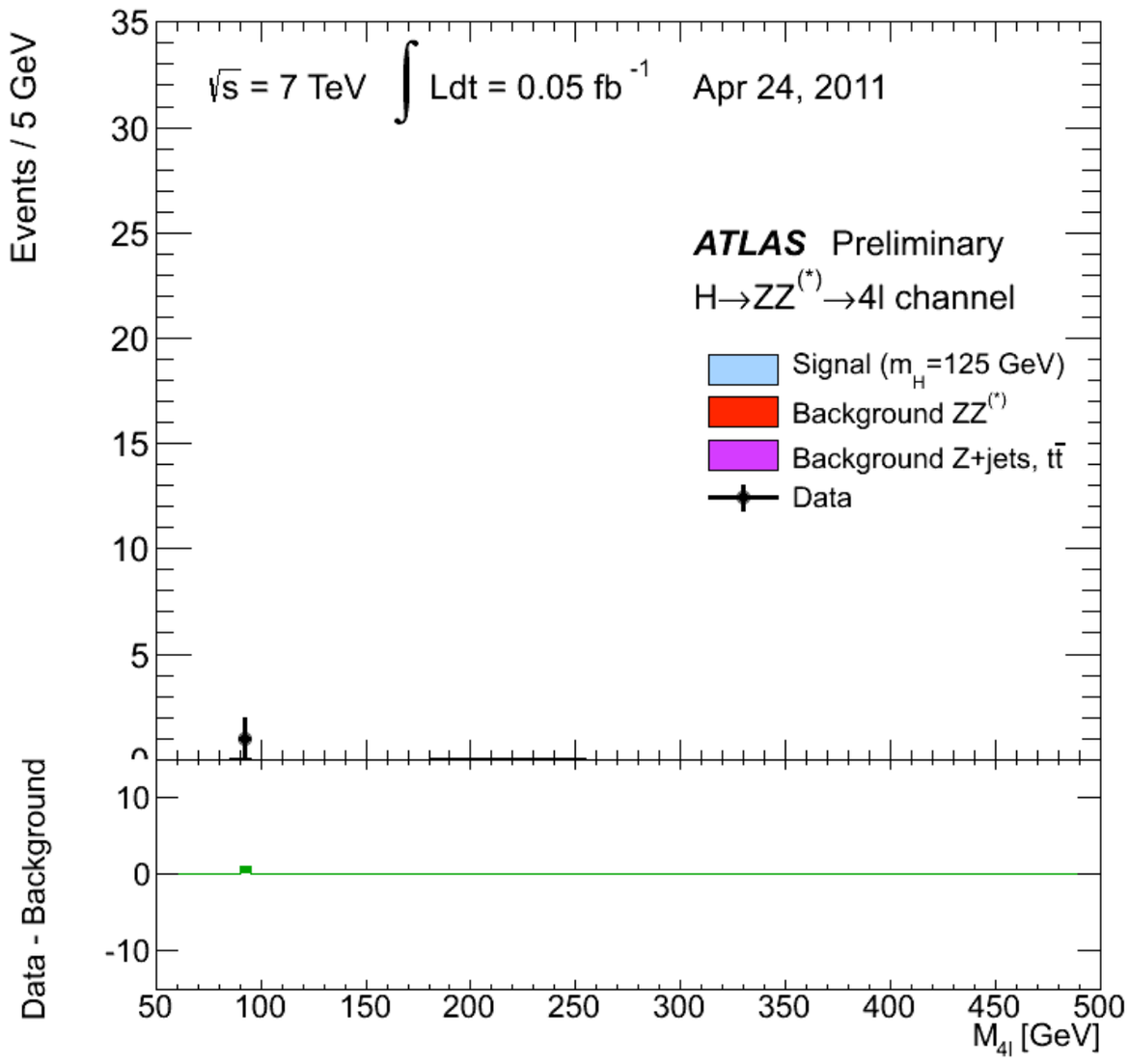


ATLAS EXPERIMENT
http://atlas.ch

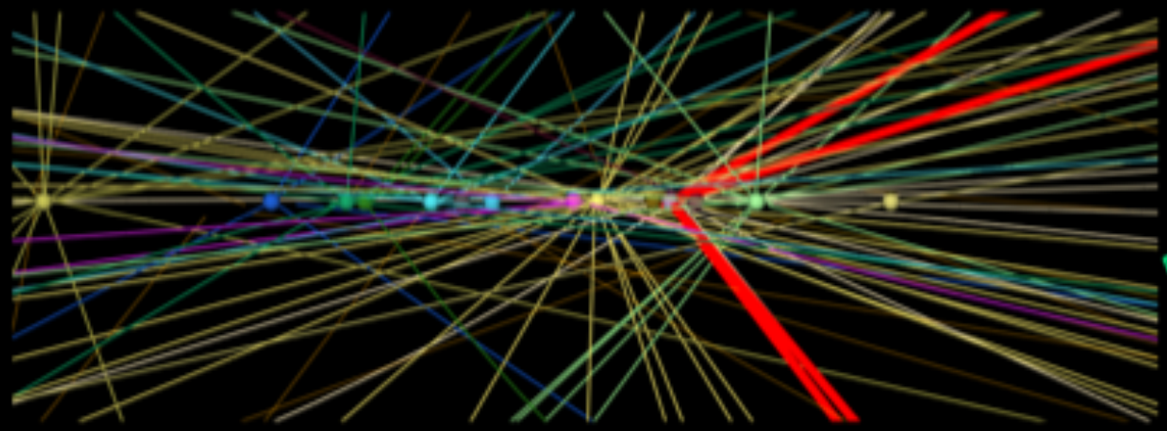
Run: 204769
Event: 71902630
Date: 2012-06-10
Time: 13:24:31 CEST

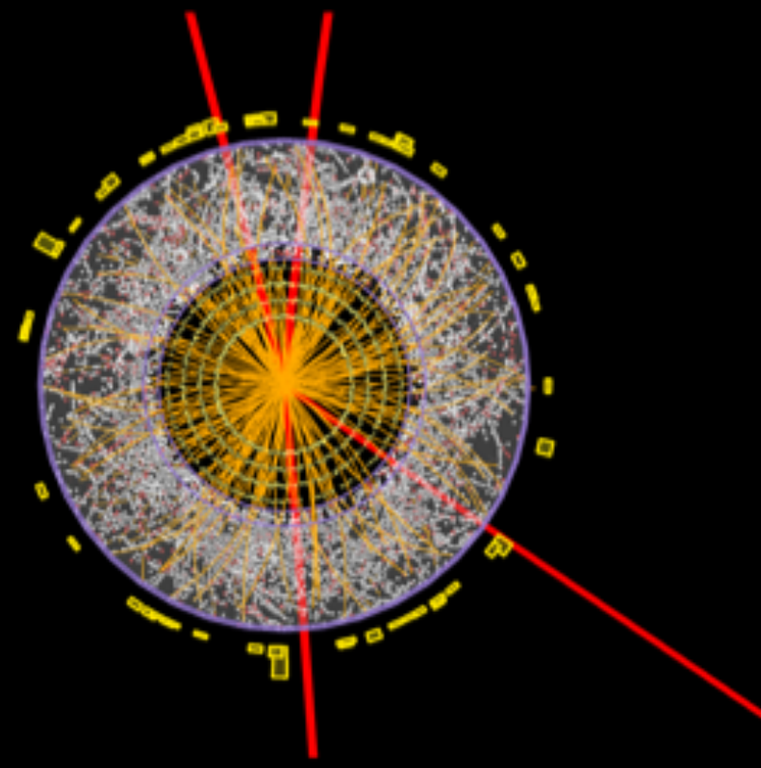
Discovery!


√s = 7 TeV ∫ Ldt = 0.02 fb -1 Apr 18, 2011
ATLAS Preliminary
H→γγ channel
Data
Background-only
Events / GeV
Data - Fit
M_γγ [GeV]


√s = 7 TeV ∫ Ldt = 0.05 fb -1 Apr 24, 2011
ATLAS Preliminary
H→ZZ(*)→4l channel
Signal (m_H=125 GeV)
Background ZZ(*)
Background Z+jets, tt̄
Data
Events / 5 GeV
Data - Background
M_4l [GeV]

$H \to ZZ \to 4l$





ATLAS EXPERIMENT
http://atlas.ch

Run: 204769
Event: 71902630
Date: 2012-06-10
Time: 13:24:31 CEST

A Physically Motivated Feature / Summrary Statistic

Don't believe the media:

$$E \neq mc^2$$

What Einstein really said:

$$E^2 = (mc^2)^2 + (|\vec{p}|c)^2$$

Every physics student knows energy and momentum are conserved

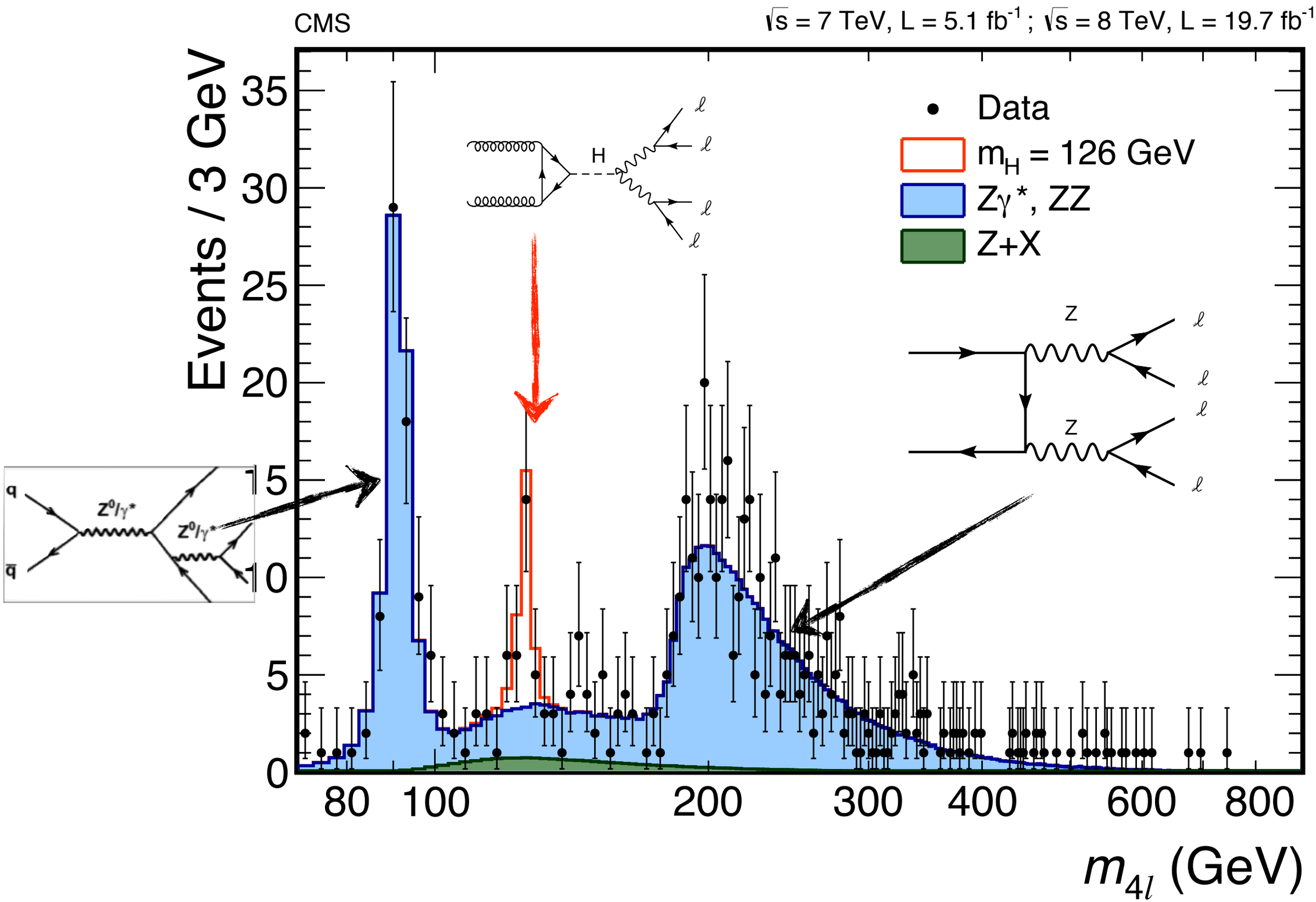
$$E_{\text{Higgs}} = E_{\text{before}} = E_{\text{after}} = \sum_i E_i$$

$$\vec{p}_{\text{Higgs}} = \vec{p}_{\text{before}} = \vec{p}_{\text{after}} = \sum_i \vec{p}_i$$

Thus, we can estimate the mass of the Higgs with

$$m_H = \sqrt{E_{\text{after}}^2/c^4 - |\vec{p}_{\text{after}}|^2/c^2}$$

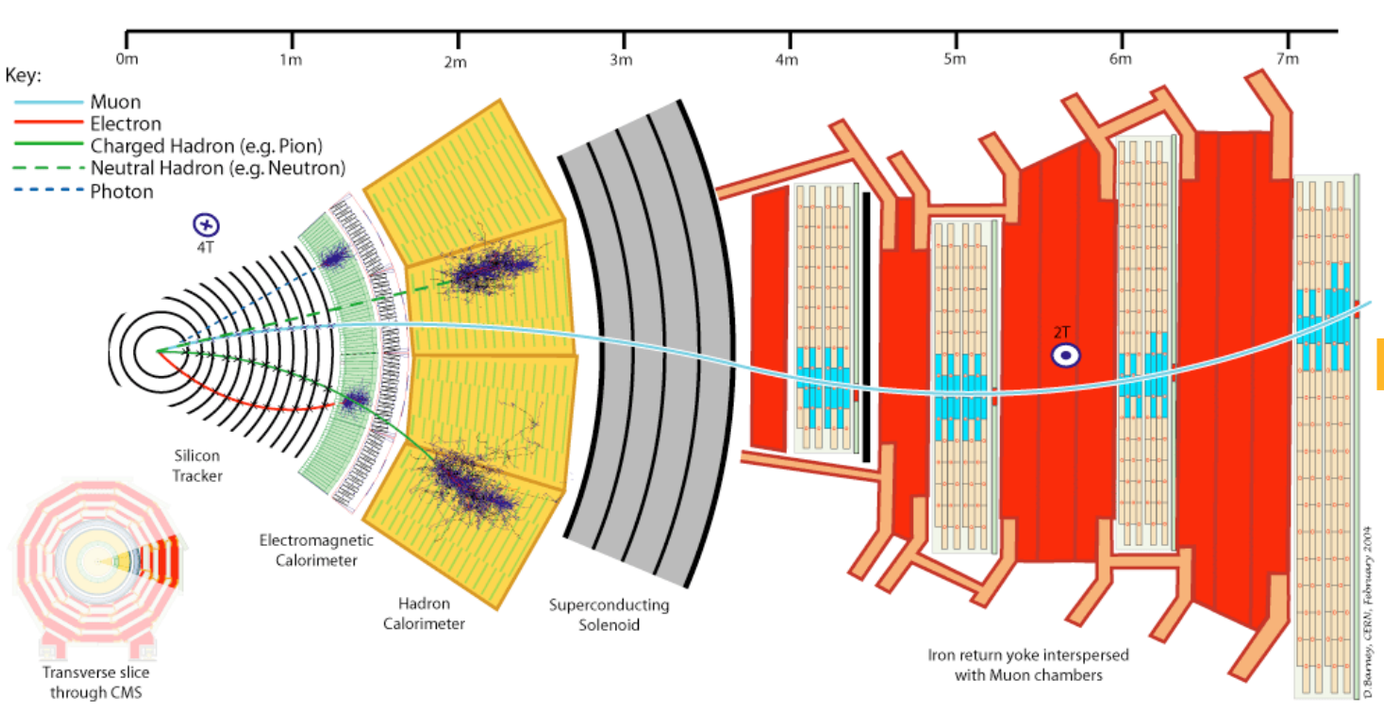
PREDICTIONS FROM SIMULATION

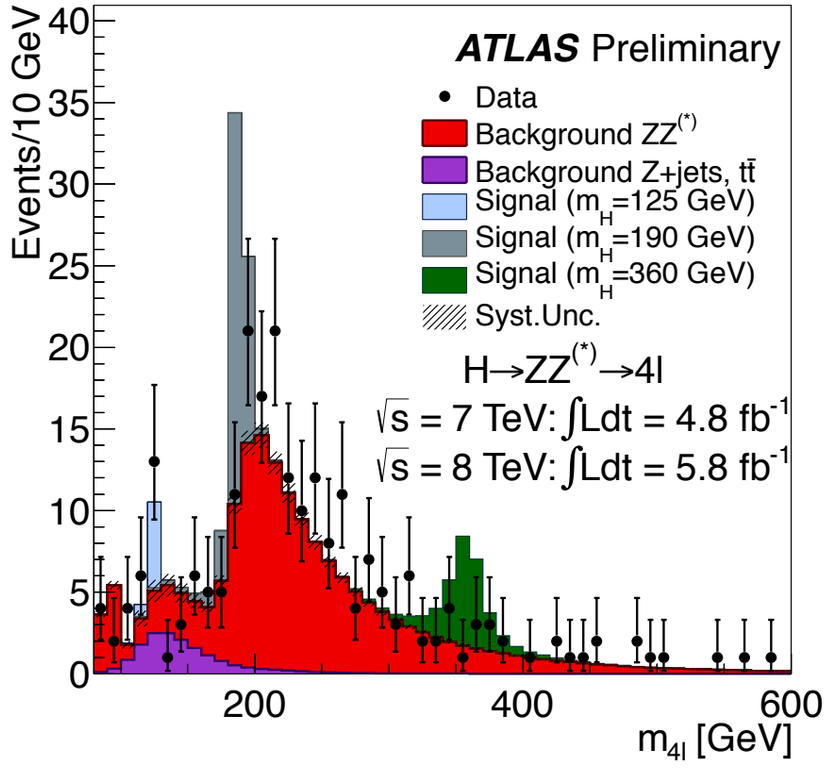


10^8 SENSORS → 1 REAL-VALUED QUANTITY

Most measurements and searches for new particles at the LHC are based on the distribution of a single variable / feature / summary statistic

- choosing a good variable (feature engineering) is a task for a skilled physicist and tailored to the goal of measurement or new particle search
- likelihood $p(x|\theta)$ **approximated** using histograms (univariate density estimation)

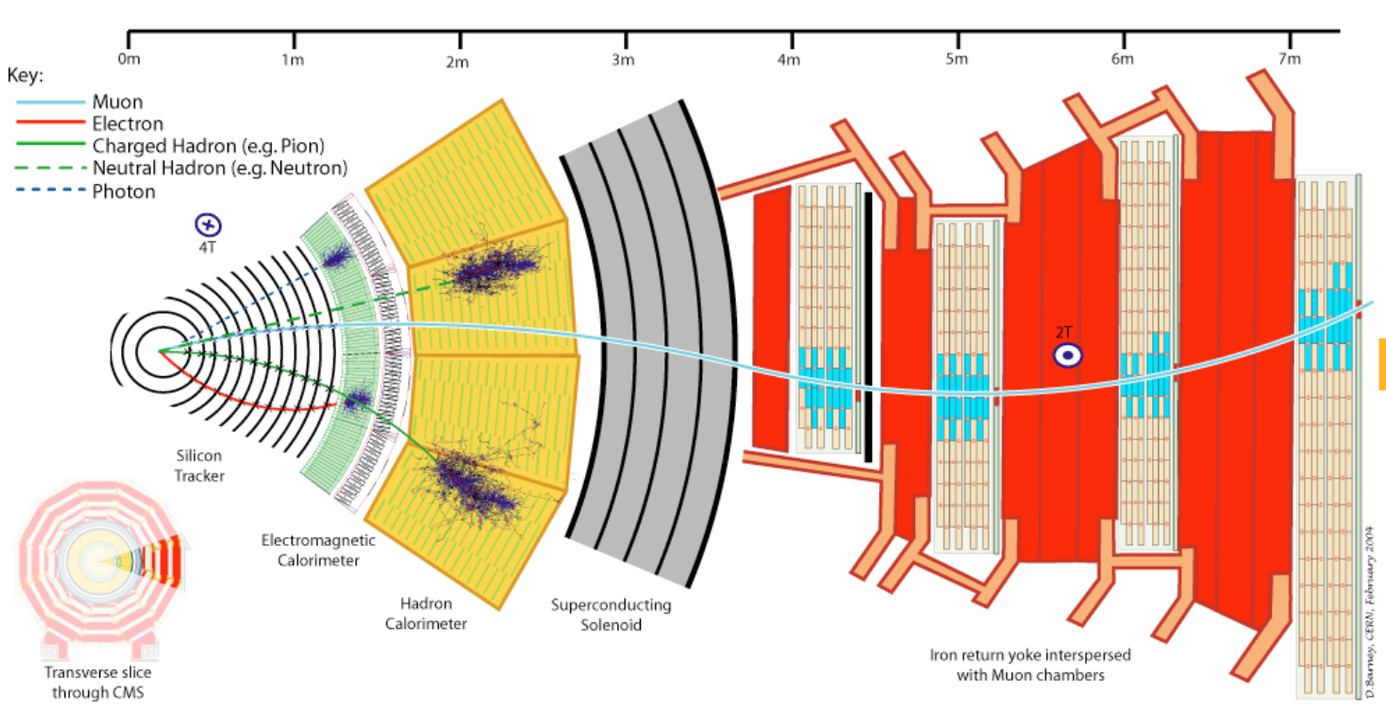


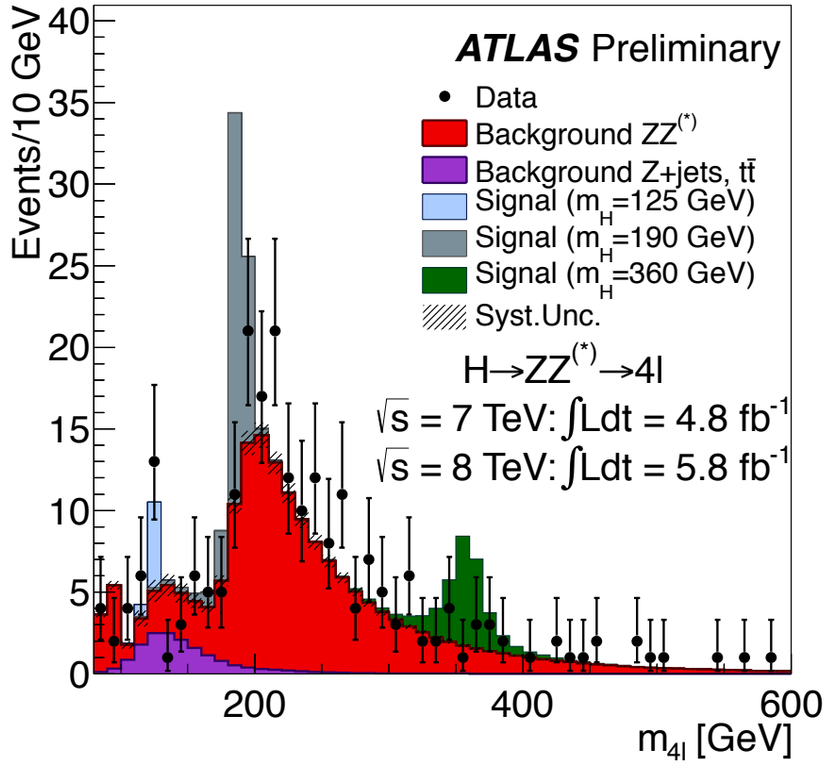


10^8 SENSORS → 1 REAL-VALUED QUANTITY

Most measurements and searches for new particles at the LHC are based on the distribution of a single variable / feature / summary statistic

- choosing a good variable (feature engineering) is a task for a skilled physicist and tailored to the goal of measurement or new particle search
- likelihood $p(x|\theta)$ **approximated** using histograms (univariate density estimation)





This doesn't scale if x is high dimensional!

THE CRUX, AN INTRACTABLE INTEGRAL

Monte Carlo Sampling

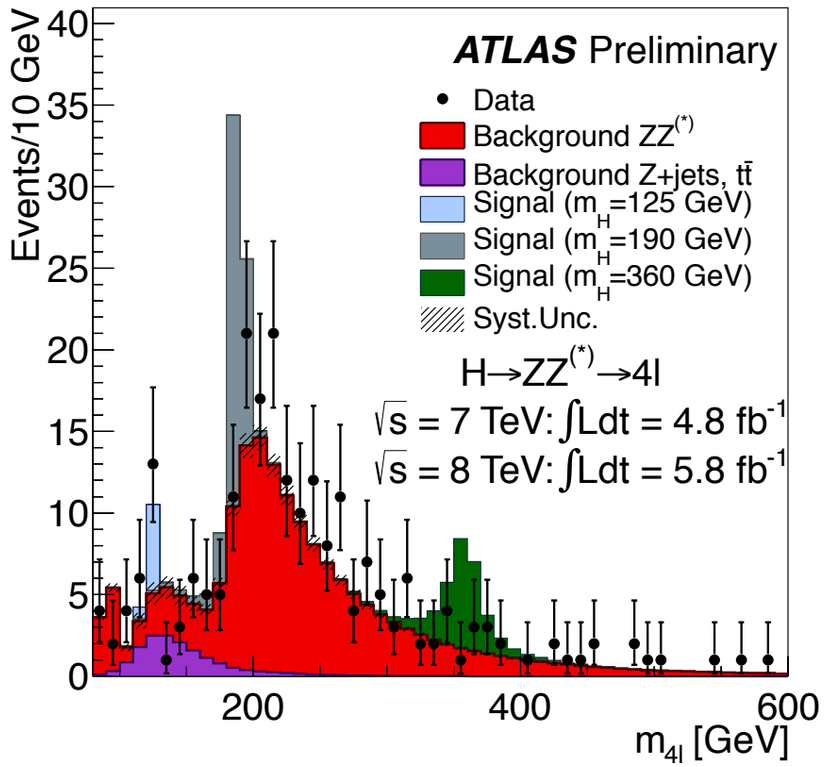
observed

simulation

$$p(x|\theta) = \int dz\, p(x, z|\theta)$$

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

histogram approximation



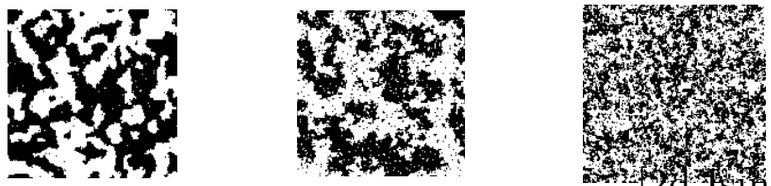
REMINDER: THE PARTITION FUNCTION

Boltzmann/Gibbs distributions

- Boltzmann/Gibbs distribution: [Boltzmann, Gibbs 1900s]

$$p(X) = \frac{1}{Z}\exp(-\beta H(X))$$

- Foundational in the development of statistical mechanics and thermodynamics.
- β: status of a temperature parameter.
  - controls the "order" of the system.
  - at high temperatures, particules tend to behave more independently from each other.
  - at low temperatures, the system "locks" to specific configurations.
  - Phase transitions: study of critical phenomena that goes from "ordered" to "disordered".
- Ex:



[2d Ising increasing β]

Undirected Graphical Models

- Thus, factors only contain nodes that are fully-connected — this is called a *clique*.
- Since a clique of size *m* contains all cliques of smaller sizes, we can reduce ourselves to *maximal cliques* (cliques that cannot be extended while being fully connected).
  - If X_C form a maximal clique, arbitrary functions $\psi(x_C)$ capture all possible dependencies within the clique.
- So, by considering

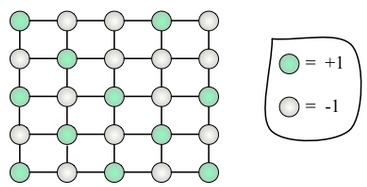
$\mathcal{C} =$ set of maximal cliques of G

$\psi_C(x_C)$: non-negative potential function (not necessarily normalized)

- We have $p(x) = \frac{1}{Z}\prod_{C\in\mathcal{C}}\psi_C(x_C)$, $Z = \int dx \prod_{C\in\mathcal{C}}\psi_C(x_C)$.

partition function

Parameter Estimation in MRFs



$$p(x_1,\cdots,x_n) = \frac{1}{Z}\exp\Big(\sum_{i<j} w_{i,j}x_i x_j - \sum_i u_i x_i\Big)$$

- In a MRF, we also have a factorization into local potentials…

$$p(x_1,\ldots,x_n;\theta) = \frac{1}{Z(\theta)}\prod_{C\in\mathcal{C}}\psi_C(x_C;\theta) .$$

- … but the partition function entangles the estimation!

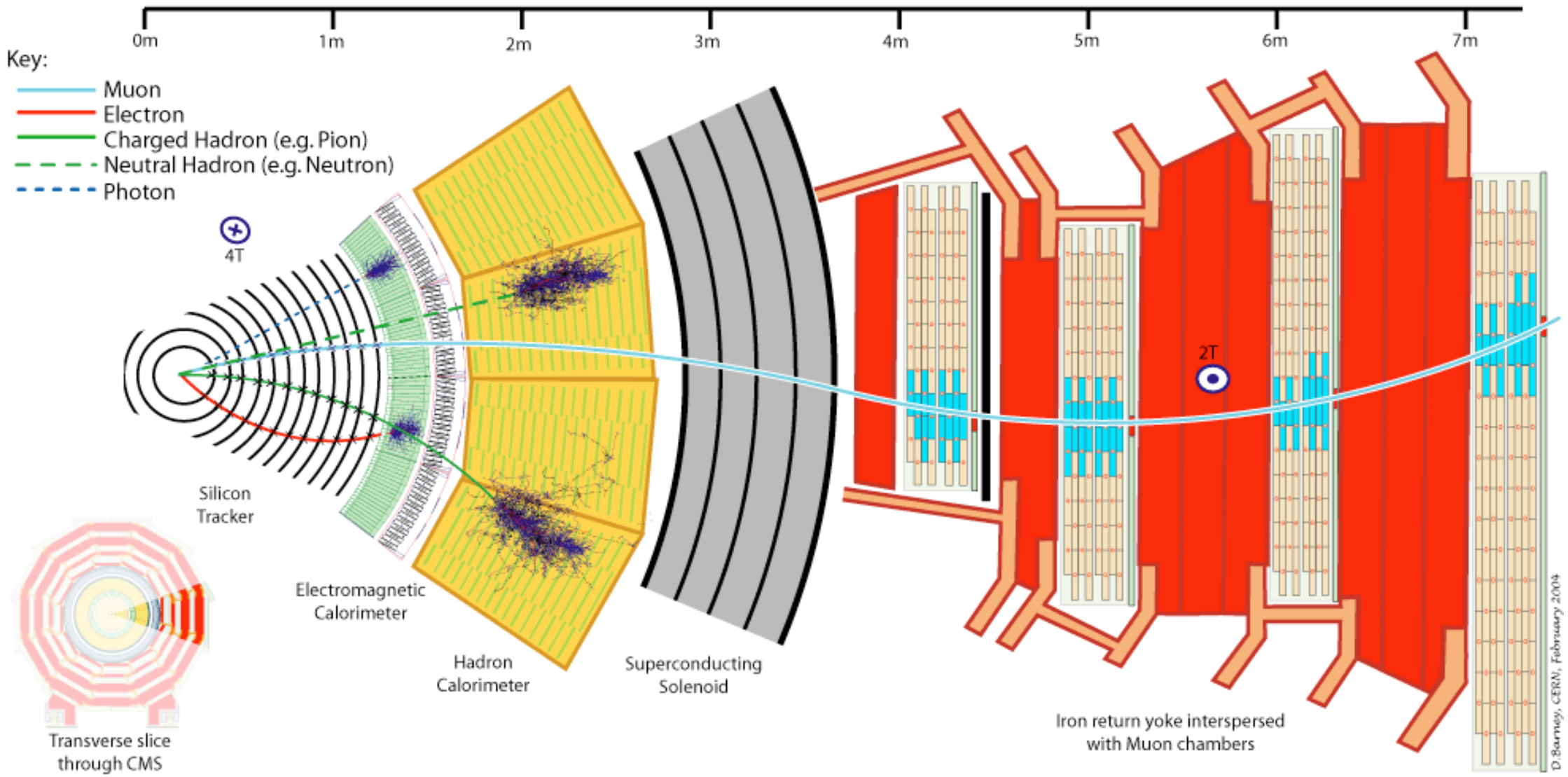
$$\sum_{l\le L}\log p(X^l;\theta) = \sum_{l\le L}\left(\sum_{C\in\mathcal{C}}\log\psi(X^l_C;\theta) - \log Z(\theta)\right) .$$

DETECTOR SIMULATION

Conceptually: Prob(detector response | particles)

Implementation: Monte Carlo integration over micro-physics

Consequence: evaluation of the likelihood is intractable



DETECTOR SIMULATION

Conceptually: Prob(detector response | particles)

Implementation: Monte Carlo integration over micro-physics

Consequence: evaluation of the likelihood is intractable

This motivates a new class of algorithms for what is called **likelihood-free inference**, which only require ability to generate samples from the simulation in the "forward mode"

…But first, some classical statistics

Comment on Bayes theorem

Comment on Bayes vs. Frequentist

Classical statistics when probability model is known:

- Hypothesis Testing
- Parameter Estimation
- Confidence Intervals & Credible Intervals

Then we will be ready to transition to likelihood-free inference

- classification & likelihood ratio estimation
- density estimation
- point estimates & credible intervals

COMMENTS ON BAYES THEOREM

Bayes Rule

- Theorem/definition of conditional probability:

$$p(x_1, x_2) = p(x_1)\ p(x_2 \mid x_1)$$

- Iterating this rule, we obtain

$$\begin{aligned} p(x_1, \ldots, x_n) &= p(x_1)p(x_2, \ldots, x_n \mid x_1) \\ &= p(x_1)p(x_2 \mid x_1)p(x_3, \ldots x_n \mid x_1, x_2) \\ &= \prod_{i=1}^{n} p(x_i \mid x_1 \ldots x_{i-1}) \ . \end{aligned}$$

– the bulk of the terms on the rhs involve $o(n)$ terms.

– variable ordering is arbitrary in general.

– what if we drop some conditioning variables? $(i_j < i)$

$$p_G(x_1, \ldots, x_n) = \prod_{i=1}^{n} p(x_i \mid x_{i_1}, \ldots x_{i_{l_i}}) \ .$$

Bayes' Theorem

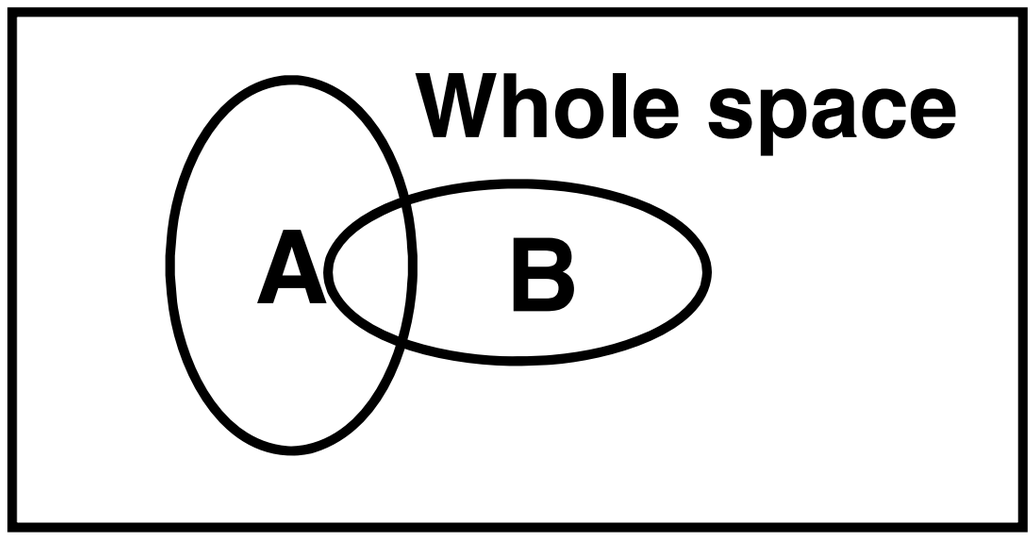
Bayes' theorem relates the conditional and marginal probabilities of events A & B

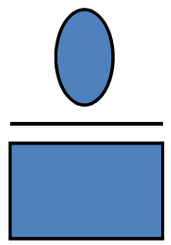
$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

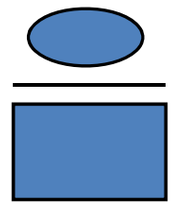
- **P(*A*)** is the prior probability. It is "prior" in the sense that it does not take into account any information about *B*.
- **P(*A* | *B*)** is the conditional probability of *A*, given *B*. It is also called the posterior probability because it is derived from or depends upon the specified value of *B*.
- **P(*B* | *A*)** is the conditional probability of *B* given *A*.
- **P(*B*)** is the prior or marginal probability of *B*, and acts as a normalizing constant.

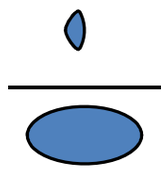
$$\pi(\theta|x) = \frac{f(x|\theta)\pi(\theta)}{\mathcal{N}} \propto L(\theta)\pi(\theta)$$

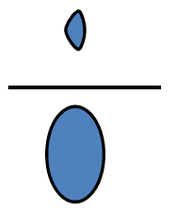
P, Conditional P, and Derivation of Bayes' Theorem in Pictures

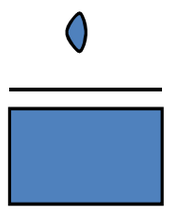


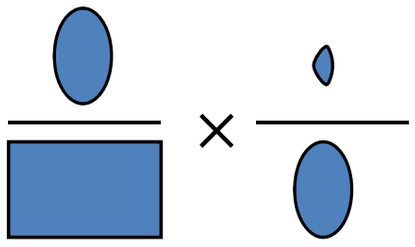
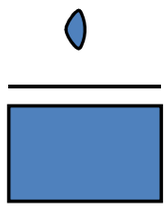
$P(A) =$ 

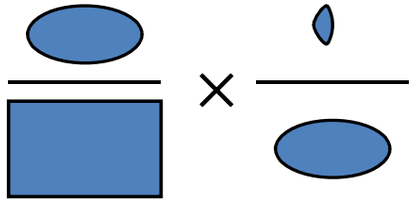
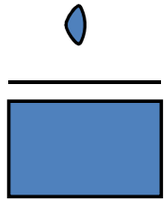
$P(B) =$ 

$P(A|B) =$ 

$P(B|A) =$ 

$P(A \cap B) =$ 

$P(A) \times P(B|A) =$  $=$  $= P(A \cap B)$

$P(B) \times P(A|B) =$  $=$  $= P(A \cap B)$

$\Rightarrow P(B|A) = P(A|B) \times P(B) / P(A)$

A USEFUL EXAMPLE

$$P(\text{Data};\text{Theory}) \neq P(\text{Theory};\text{Data})$$

Theory = male or female

Data = pregnant or not pregnant

P (pregnant ; female) ~ 3%

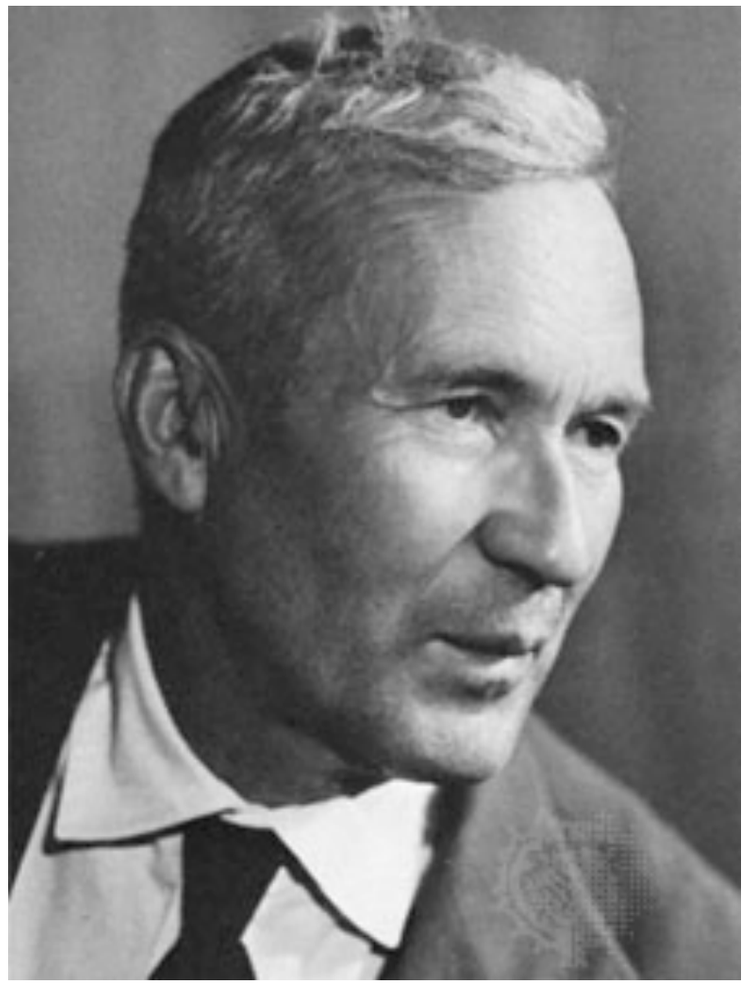
but

P (female ; pregnant) >>>3%

Axioms of Probability

These Axioms are a mathematical starting point for probability and statistics

1. probability for every element, E, is non-negative $P(E) \geq 0 \qquad \forall E \subseteq \mathcal{F} = 2^{\Omega}$
2. probability for the entire space of possibilities is 1 $P(\Omega) = 1.$
3. if elements E_i are disjoint, probability is additive $P(E_1 \cup E_2 \cup \cdots) = \sum_i P(E_i).$



Kolmogorov axioms (1933)

Consequences:

$$P(A \cup B) = P(A) + P(B) - P(A \cap B)$$

$$P(\Omega \setminus E) = 1 - P(E)$$

Different definitions of Probability

Frequentist

- defined as limit of long term frequency
- probability of rolling a 3 := limit of (# rolls with 3 / # trials)
  - you don't need an infinite sample for definition to be useful
  - sometimes ensemble doesn't exist
    - eg. P(Higgs particle exists), P(mass of particle = 125 GeV), P(it will snow tomorrow)
- Intuitive if you are familiar with Monte Carlo methods

Subjective Bayesian

- Probability is a degree of belief (personal, subjective)
  - can be made quantitative based on betting odds
  - most people's subjective probabilities are not ***coherent*** and do not obey laws of probability

http://plato.stanford.edu/archives/sum2003/entries/probability-interpret/#3.1

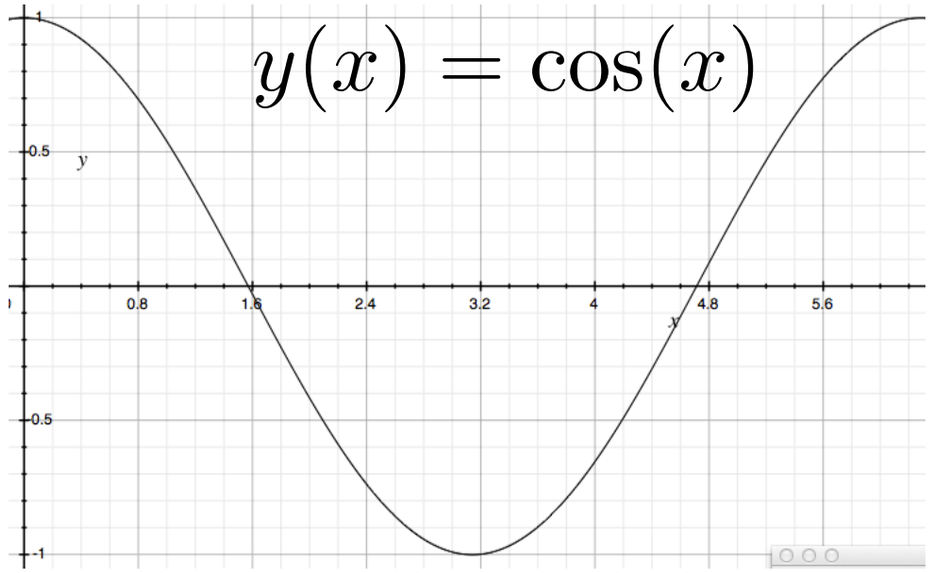
TRANSFORMATION PROPERTIES:

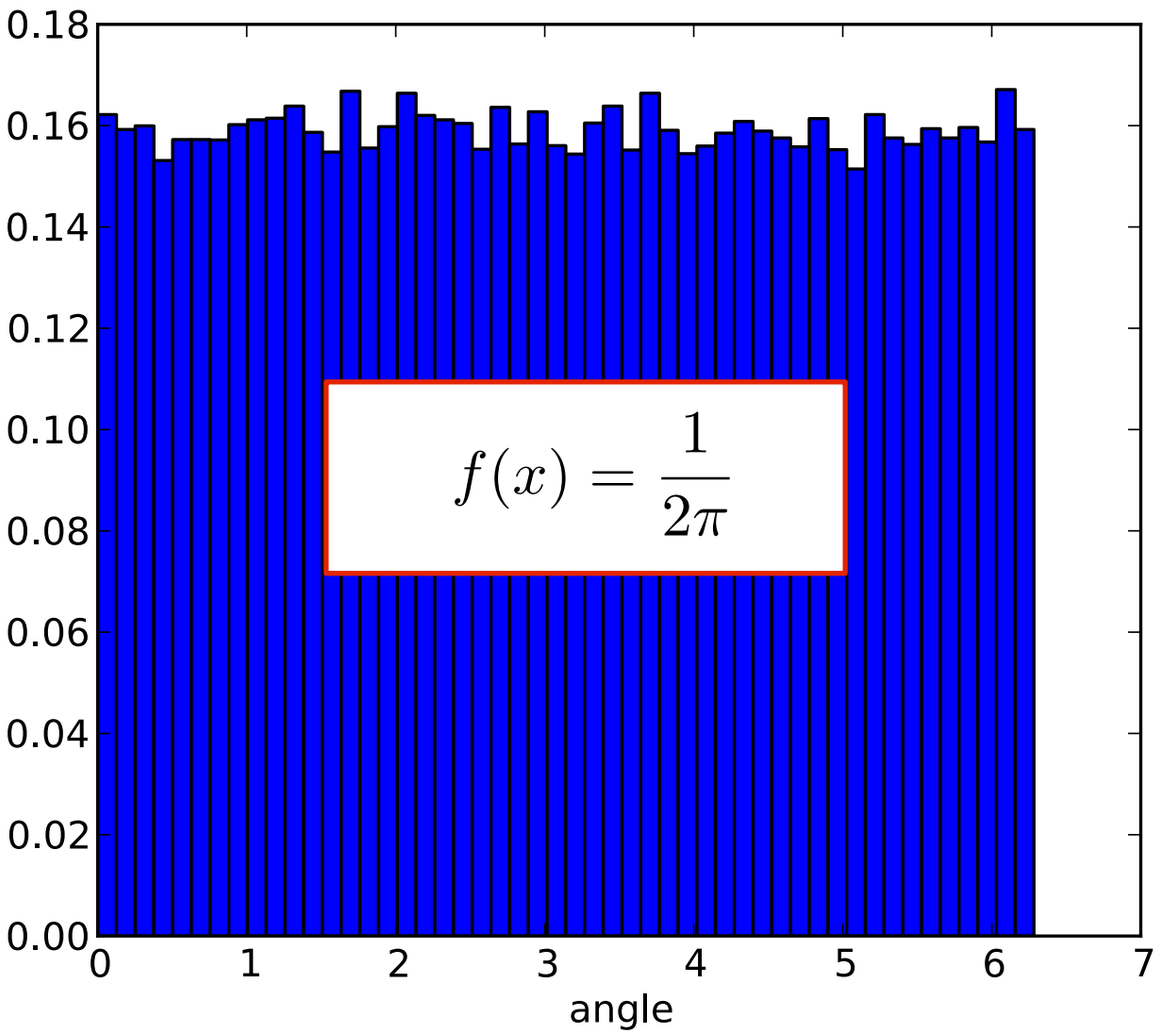
PDF VS. LIKELIHOOD

An example

if x has a uniform distribution [0,2π) ,

what is distribution of y=cos(x)?

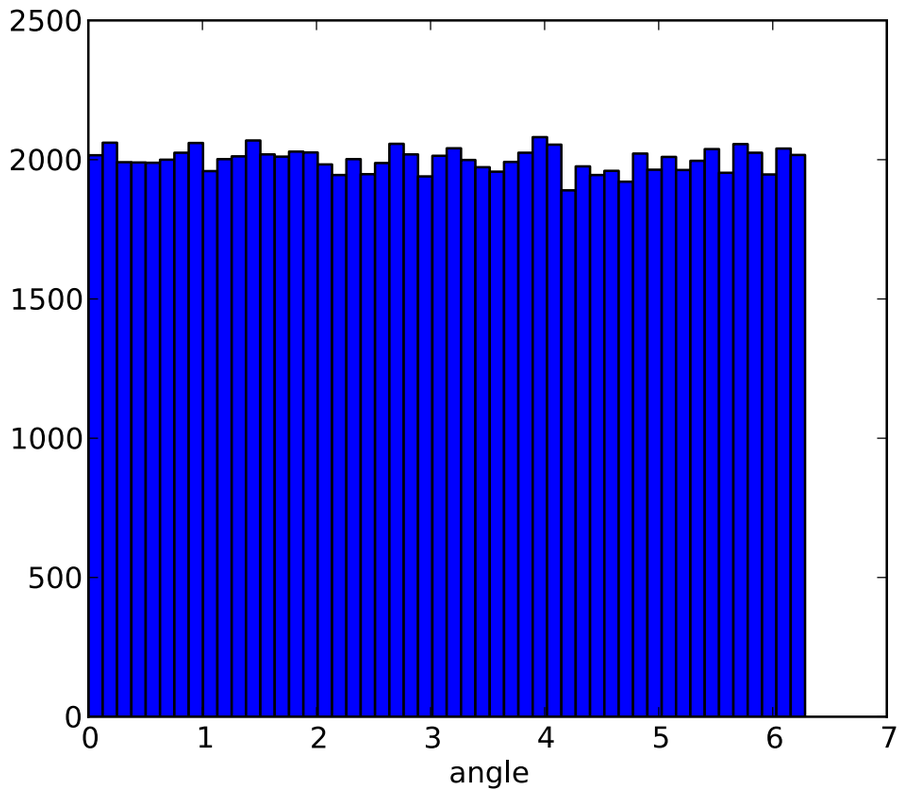




CHANGE OF VARIABLES

What happens with x→ cos(x)

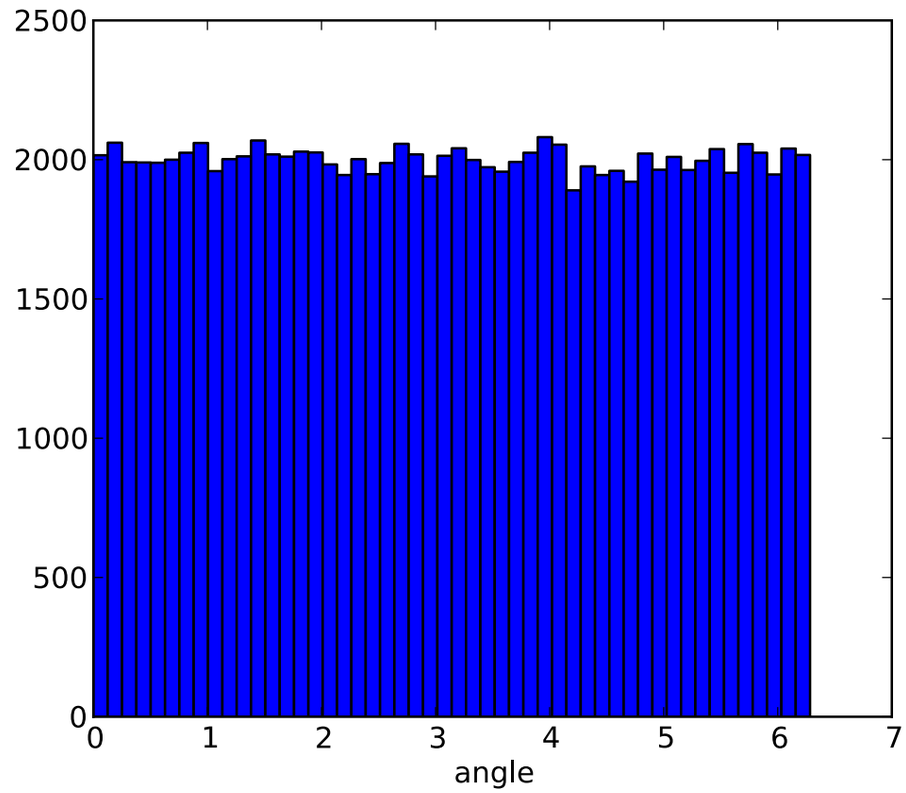
```python
import numpy as np
import matplotlib.pyplot as plt

N_MC=100000  # number of Monte Carlo Experiments
nBins = 50 # number of bins for Histograms

data_x, data_y = [],[]  #lists that will hold x and y

# do experiments
for i in range(N_MC):
    # generate observation for x
    x = np.random.uniform(0,2*np.pi)

    y = np.cos(x)
    data_x.append(x)
    data_y.append(y)

#setup figures
fig = plt.figure(figsize=(13,5))
fig_x = fig.add_subplot(1,2,1)
fig_y = fig.add_subplot(1,2,2)

fig_x.hist(data_x,nBins)
fig_x.set_xlabel('angle')

fig_y.hist(data_y,nBins)
fig_y.set_xlabel('cos(angle)')

plt.show()
```

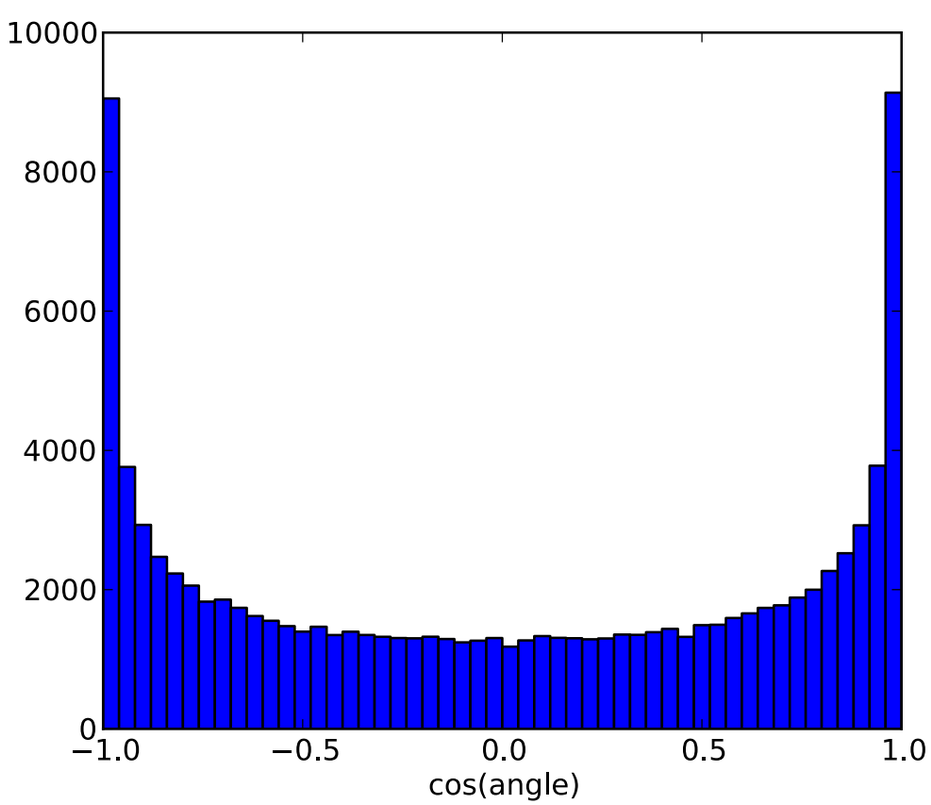


CHANGE OF VARIABLES

What happens with x→ cos(x)

```
1   import numpy as np
2   import matplotlib.pyplot as plt
3
4   N_MC=100000  # number of Monte Carlo Experiments
5   nBins = 50 # number of bins for Histograms
6
7   data_x, data_y = [],[]  #lists that will hold x and y
8
9   # do experiments
10  for i in range(N_MC):
11      # generate observation for x
12      x = np.random.uniform(0,2*np.pi)
13
14      y = np.cos(x)
15      data_x.append(x)
16      data_y.append(y)
17
18  #setup figures
19  fig = plt.figure(figsize=(13,5))
20  fig_x = fig.add_subplot(1,2,1)
21  fig_y = fig.add_subplot(1,2,2)
22
23  fig_x.hist(data_x,nBins)
24  fig_x.set_xlabel('angle')
25
26  fig_y.hist(data_y,nBins)
27  fig_y.set_xlabel('cos(angle)')
28
29  plt.show()
```





CHANGE OF VARIABLES

If *f(x)* is the pdf for *x* and *y(x)* is a change of variables, then the pdf *g(y)* must satisfy

$$P(x_a < x < x_b) \equiv \int_{x_a}^{x_b} f(x)dx = \int_{y(x_a)}^{y(x_b)} g(y)dy \equiv P(y(x_a) < y < y(x_b))$$

We can rewrite the integral on the right

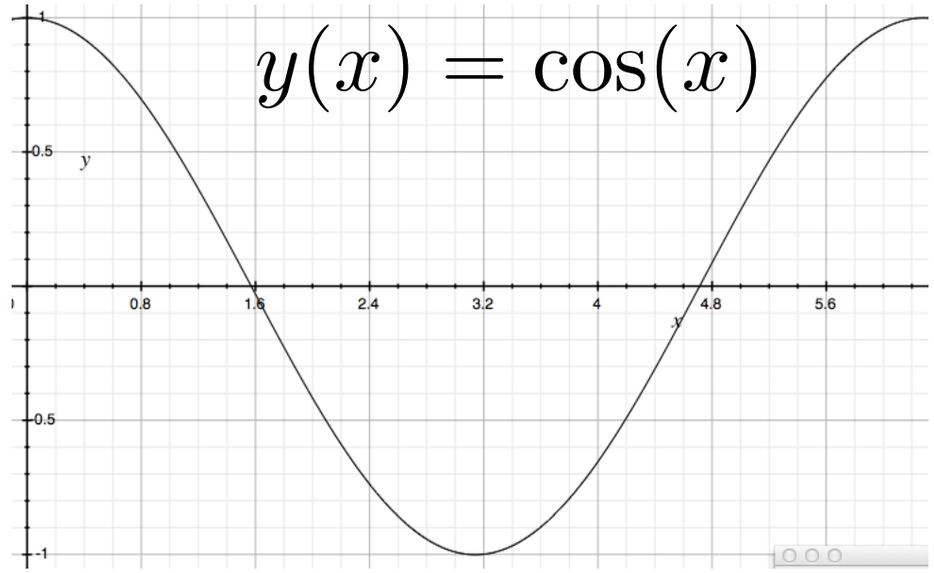
$$\int_{y(x_a)}^{y(x_b)} g(y)dy = \int_{x_a}^{x_b} g(y(x)) \left|\frac{dy}{dx}\right| dx$$

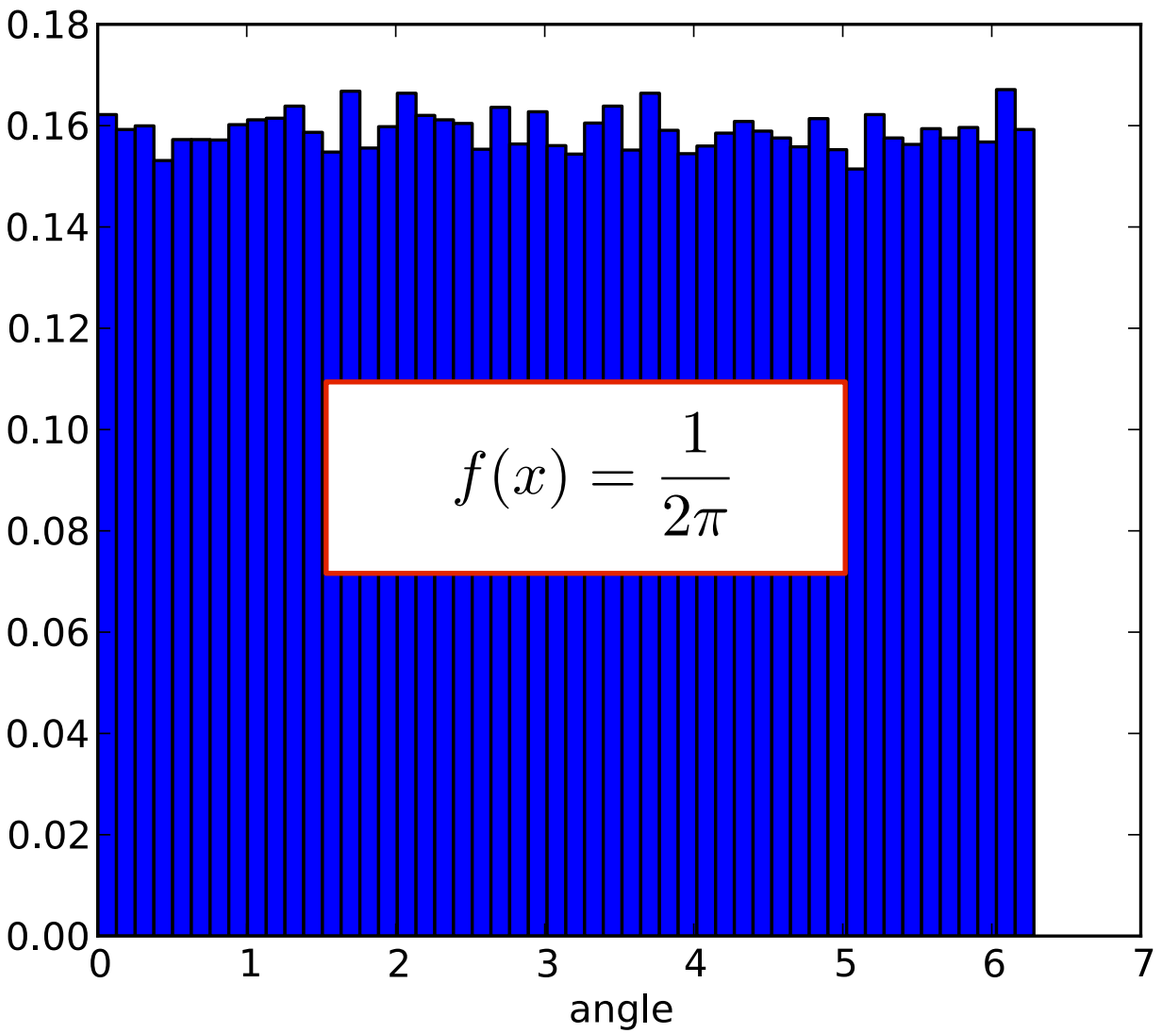
therefore, the two pdfs are related by a Jacobian factor

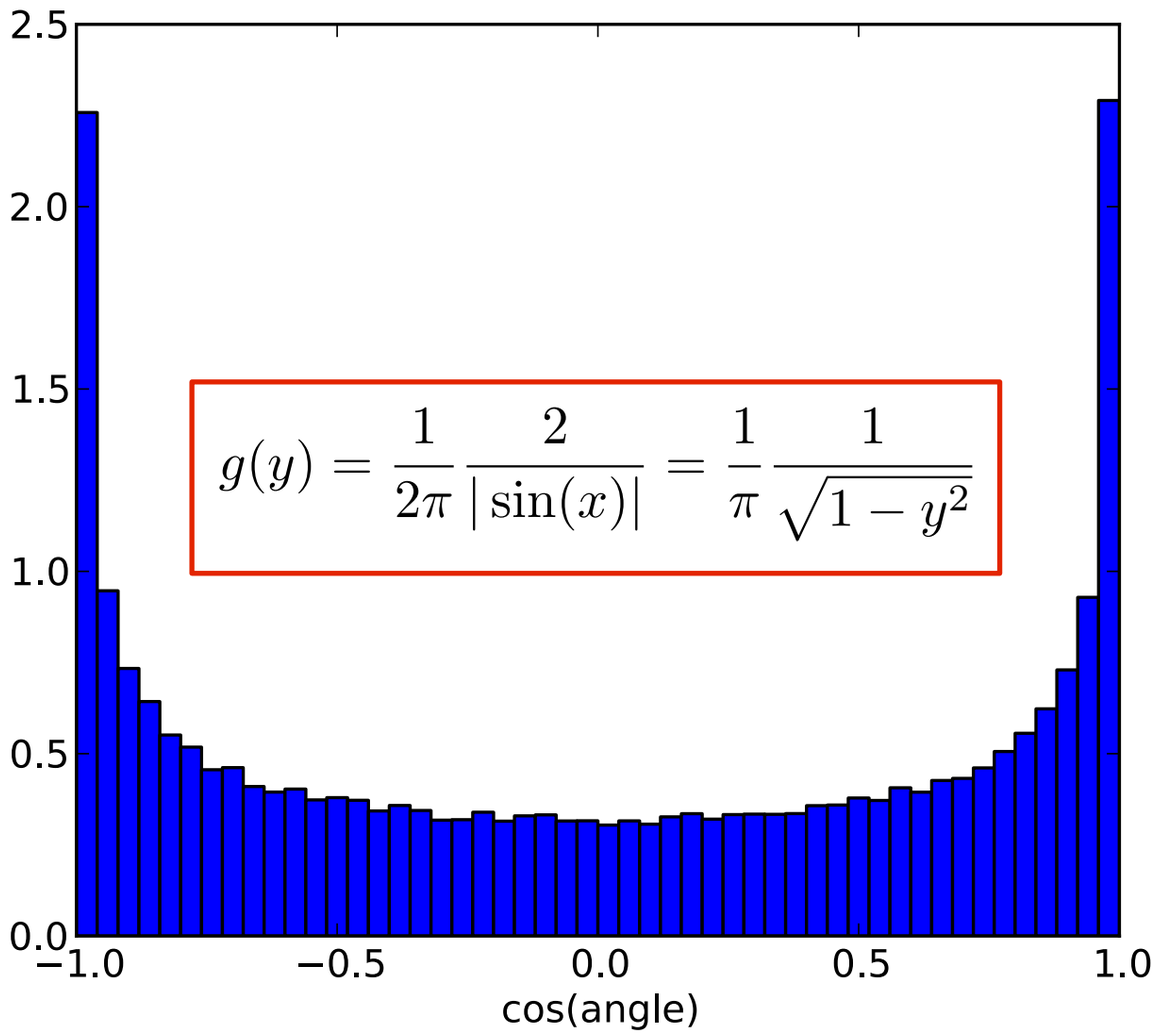
$$f(x) = g(y) \left|\frac{dy}{dx}\right|$$

An example

$$f(x) = g(y)\left|\frac{dy}{dx}\right|$$

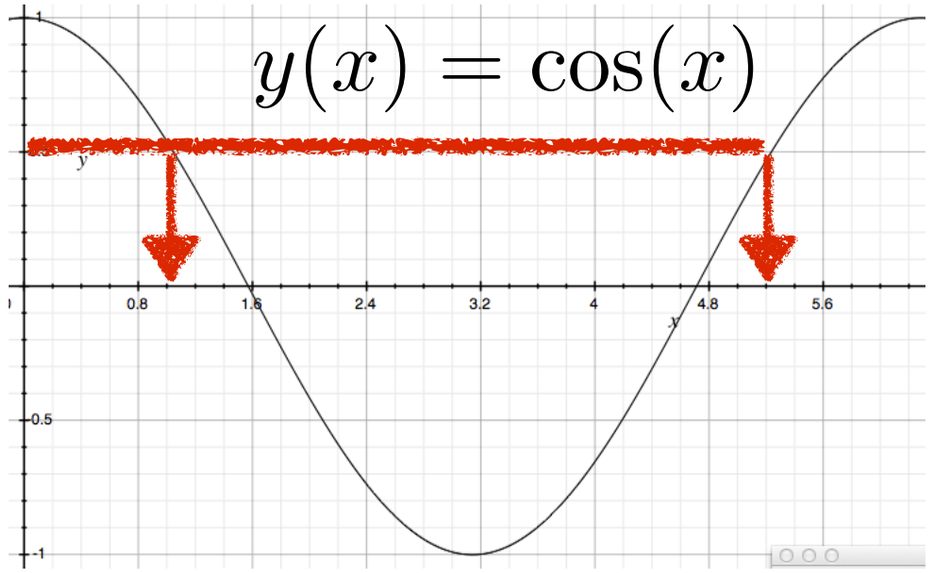


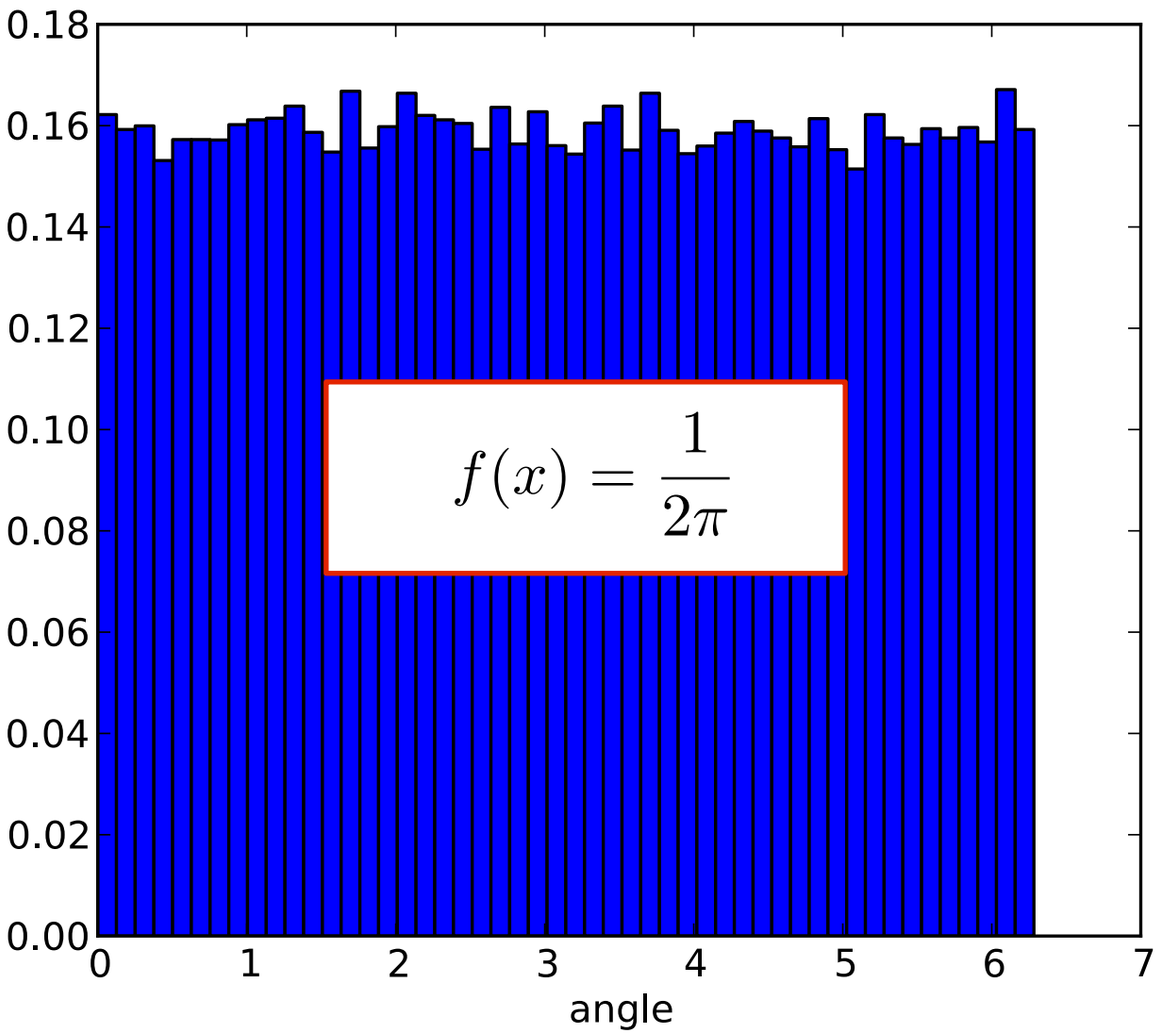


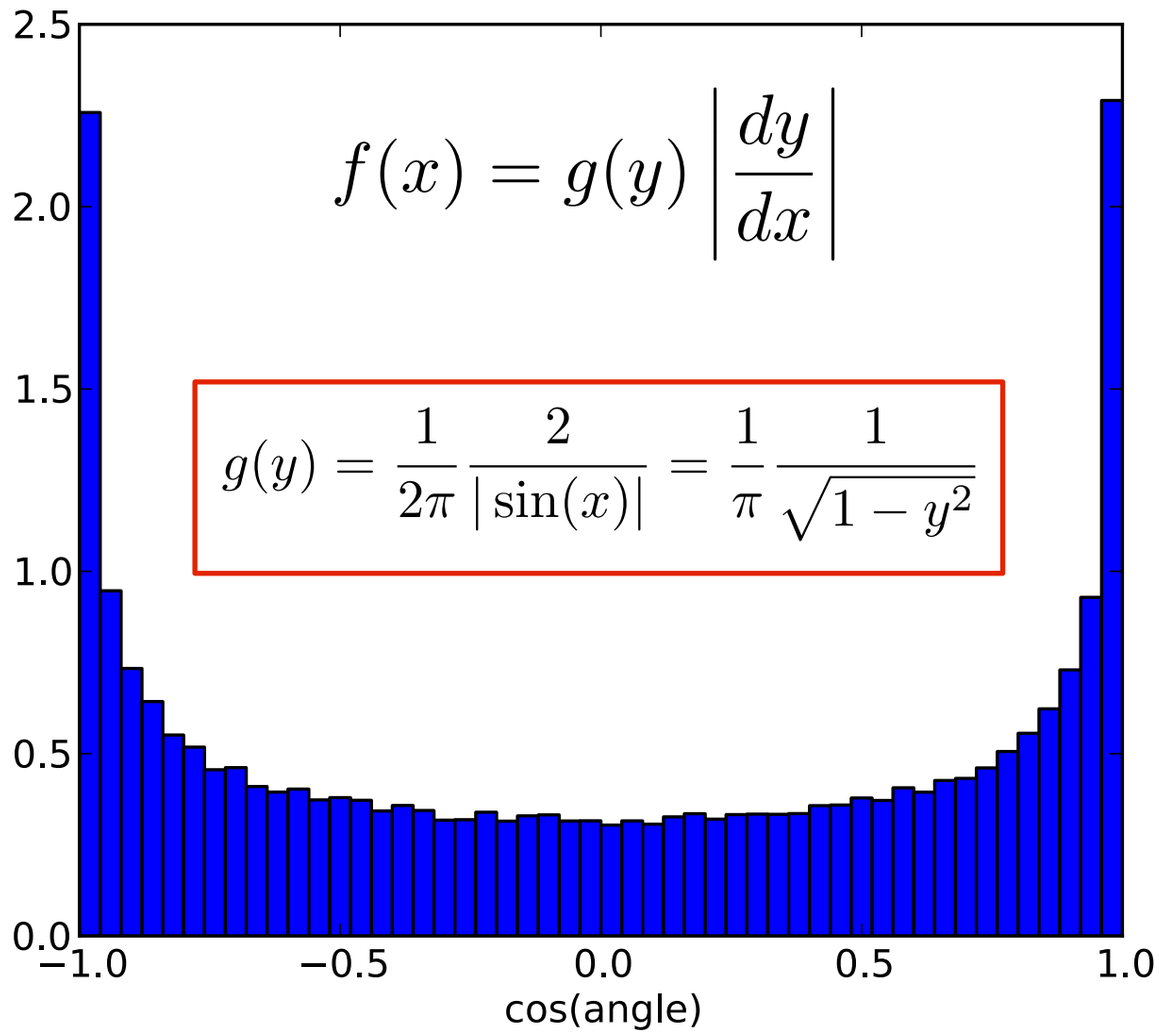


An example

I am glossing over the fact that the map is not 1-to-1. Different values of x, map into same value of y. We will need to sum/integrate over them. Here it is easy, but in general this may become intractable… need inverse map







Change of variable x, change of parameter θ

- **For pdf p(x|θ) and change of variable from x to y(x):**

  p(y(x)|θ) = p(x|θ) / |dy/dx|.

  Jacobian modifies probability *density*, guaranties that

  P($y(x_1) < y < y(x_2)$) = P($x_1 < x < x_2$), i.e., that

  ***Probabilities* are invariant under change of variable x.**

  - **Mode of probability *density* is *not* invariant (so, e.g., criterion of maximum probability density is ill-defined).**
  - **Likelihood *ratio* is invariant under change of variable x. (Jacobian in denominator cancels that in numerator).**

- **For likelihood $\mathcal{L}(\theta)$ and reparametrization from θ to u(θ):**

  $\mathcal{L}(\theta) = \mathcal{L}(u(\theta))$ (!).

  - **Likelihood $\mathcal{L}(\theta)$ is invariant under reparametrization of parameter θ (reinforcing fact that $\mathcal{L}$ is *not* a pdf in θ).**

Probability Integral Transform

Consider a specific change of variables related to the cumulative for some arbitrary *f(x)*

$$y(x) = \int_{-\infty}^{x} f(x')dx'$$

Using our general change of variables formula:

$$f(x) = g(y)\left|\frac{dy}{dx}\right|$$

We find for this case the Jacobian factor is

$$\left|\frac{dy}{dx}\right| = f(x)$$

Thus $g(y) = 1$

Probability Integral Transform

"…seems likely to be one of the most fruitful conceptions introduced into statistical theory in the last few years"
– Egon Pearson (1938)

Given continuous x ∈ (a,b), and its pdf p(x), let

$$y(x) = \int_a^x p(x')\,dx' \ .$$

Then y ∈ (0,1) and p(y) = 1 (uniform) for all y. (!)

So there always exists a metric in which the pdf is uniform.

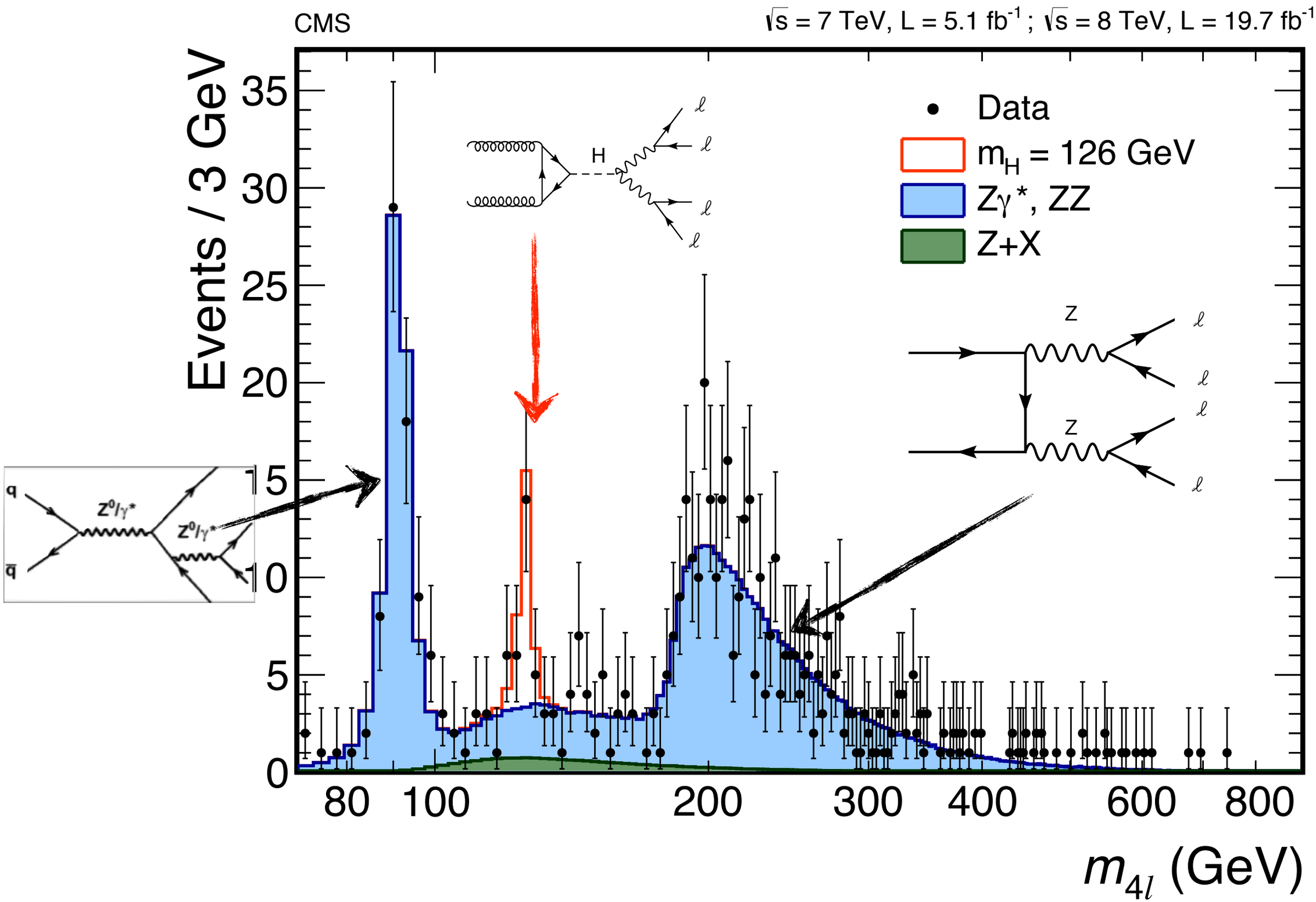
***Many* issues become more clear (or trivial) after this transformation*. (If x is discrete, some complications.)**

The specification of a Bayesian prior pdf p(μ) for parameter μ is equivalent to the choice of the metric f(μ) in which the pdf is uniform. This is a *deep* issue, not always recognized as such by users of flat prior pdf's in HEP!

***And the inverse transformation provides for efficient M.C. generation of p(x) starting from RAN().**

HYPOTHESIS TESTING

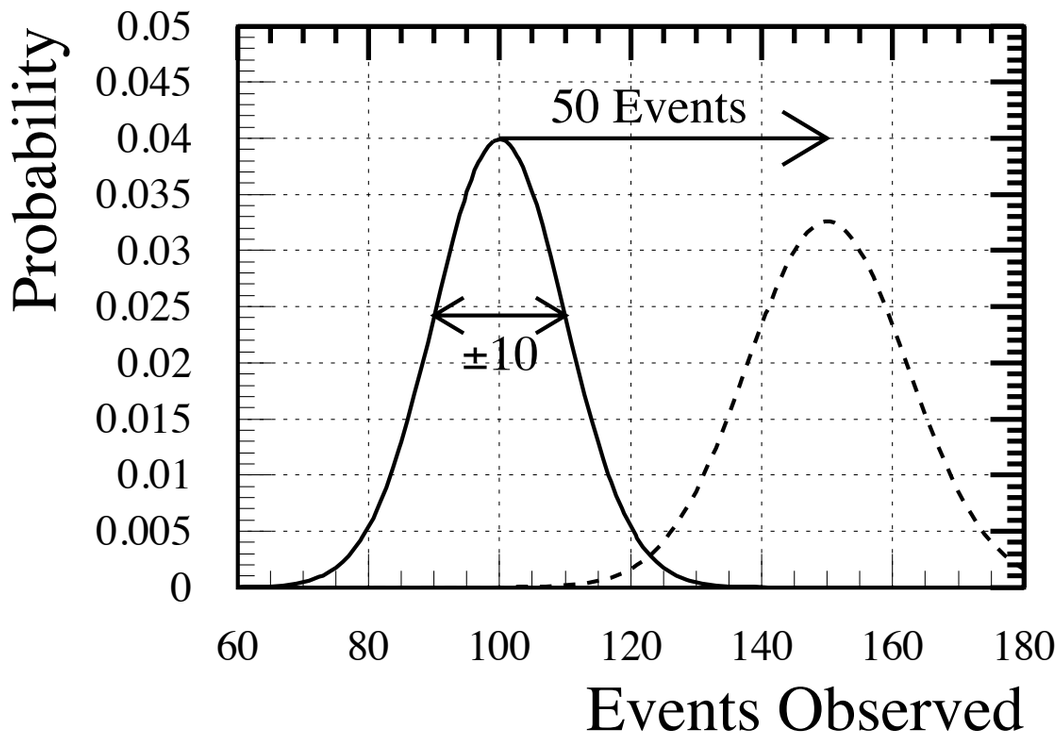
PREDICTIONS FROM SIMULATION



HYPOTHESIS TESTING

One of the most common uses of statistics in particle physics is Hypothesis Testing (e.g. for discovery of a new particle)

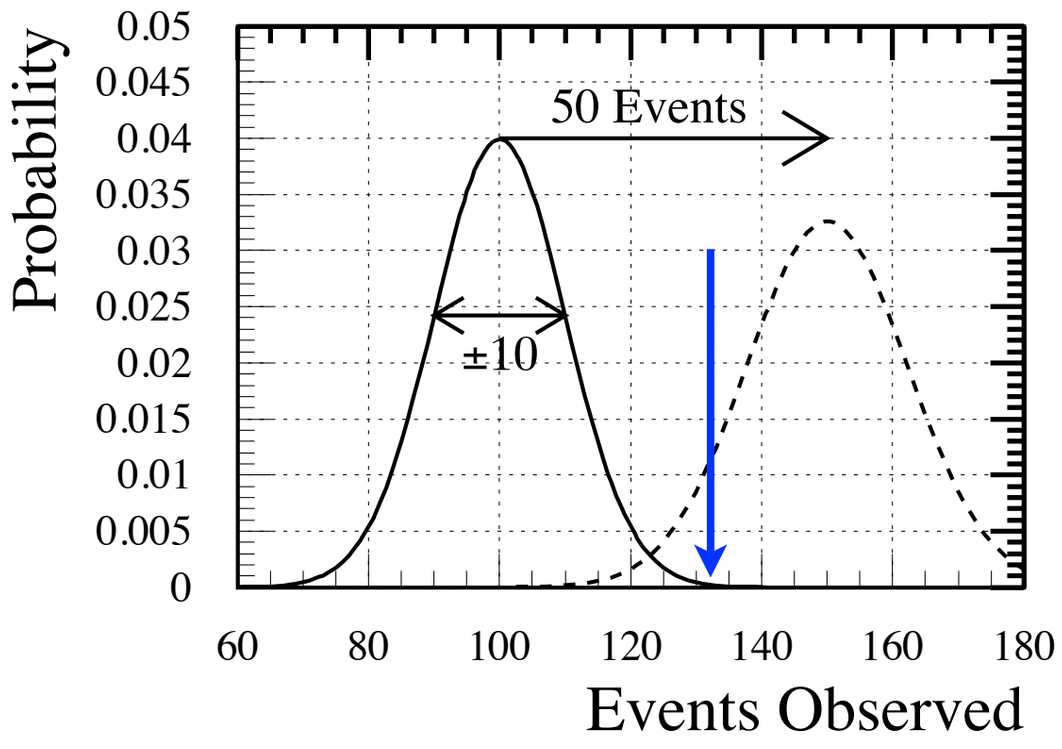
- assume one has pdf for data under two hypotheses:
  - Null-Hypothesis, H_0: eg. background-only
  - Alternate-Hypothesis H_1: eg. signal-plus-background
- one makes a measurement and then needs to decide whether to **reject** or **accept** H_0



HYPOTHESIS TESTING

One of the most common uses of statistics in particle physics is Hypothesis Testing (e.g. for discovery of a new particle)

- assume one has pdf for data under two hypotheses:
  - Null-Hypothesis, H_0: eg. background-only
  - Alternate-Hypothesis H_1: eg. signal-plus-background
- one makes a measurement and then needs to decide whether to **reject** or **accept** H_0



HYPOTHESIS TESTING

Classical hypothesis testing typically framed in terms of true/false positive/negative

- first let us define a few terms:
  - Rate of Type I error α
  - Rate of Type II β
  - Power = $1 - \beta$

		Actual condition	
		Guilty	**Not guilty**
Decision	**Verdict of 'guilty'**	True Positive	False Positive (i.e. guilt reported unfairly) **Type I error**
	Verdict of 'not guilty'	False Negative (i.e. guilt not detected) **Type II error**	True Negative

Treat the two hypotheses asymmetrically

- the Null is special.
  - Fix rate of Type I error, call it “the size of the test”

Now one can state “a well-defined goal”

- Maximize power for a fixed rate of Type I error

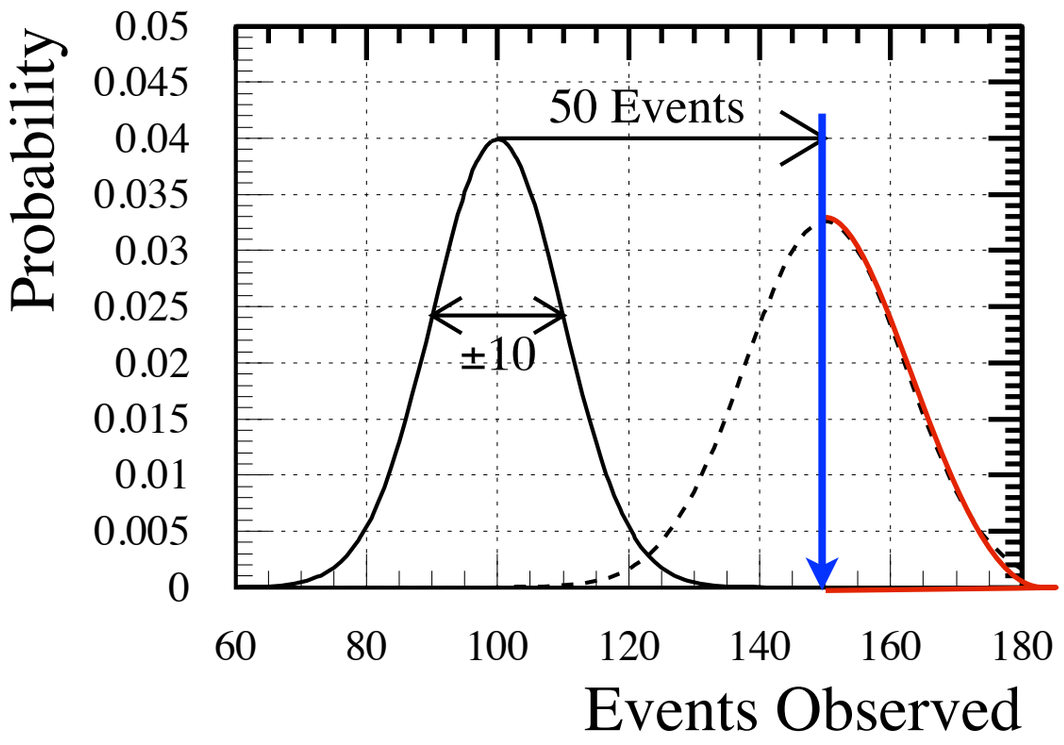
HYPOTHESIS TESTING

The idea of a “5σ” discovery criteria for particle physics is really a conventional way to specify the size of the test

- usually 5σ corresponds to $\alpha = 2.87 \cdot 10^{-7}$
  - eg. a very small chance we reject the standard model

In the simple case of number counting it is obvious what region is sensitive to the presence of a new signal

- but in higher dimensions it is not so easy



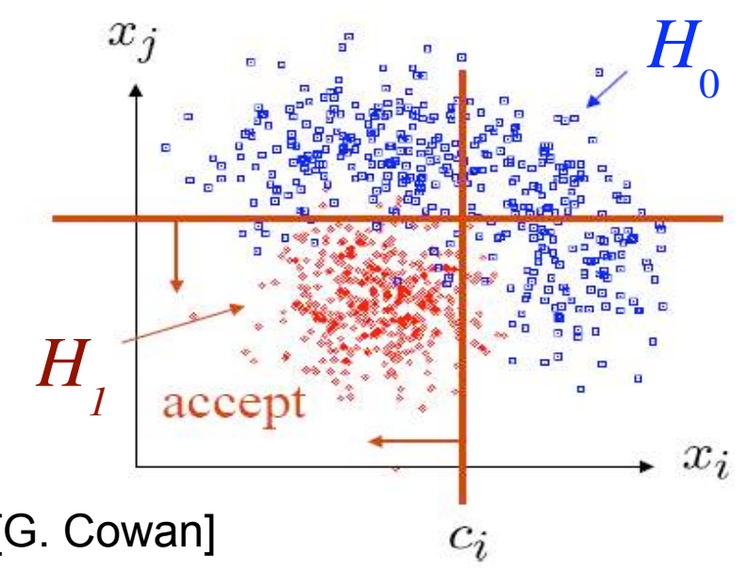
HYPOTHESIS TESTING

The idea of a “5σ” discovery criteria for particle physics is really a conventional way to specify the size of the test

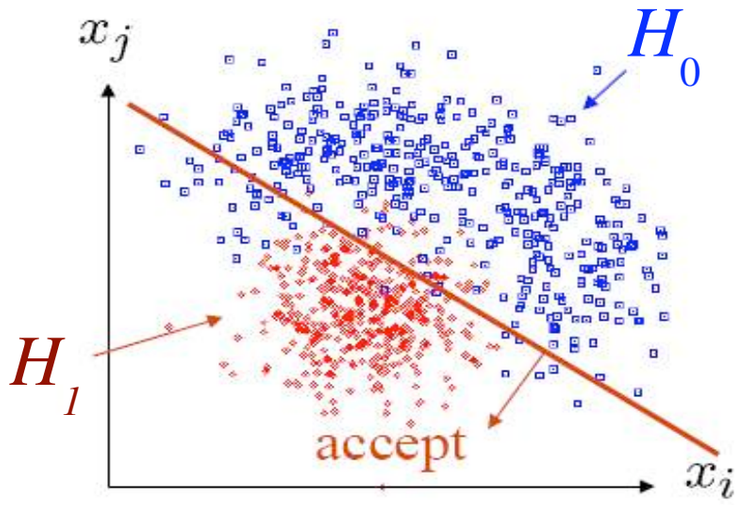
- usually 5σ corresponds to $\alpha = 2.87 \cdot 10^{-7}$
  - eg. a very small chance we reject the standard model

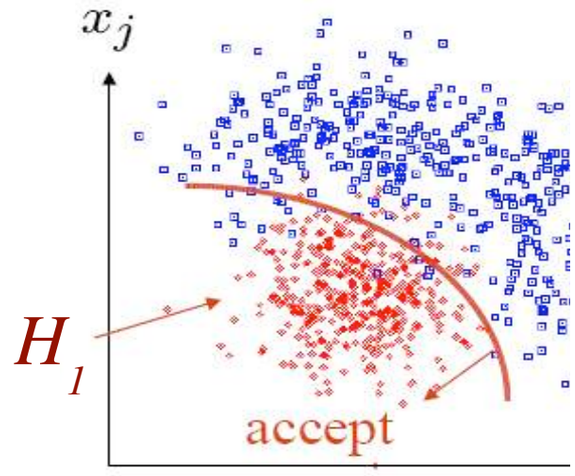
In the simple case of number counting it is obvious what region is sensitive to the presence of a new signal

- but in higher dimensions it is not so easy

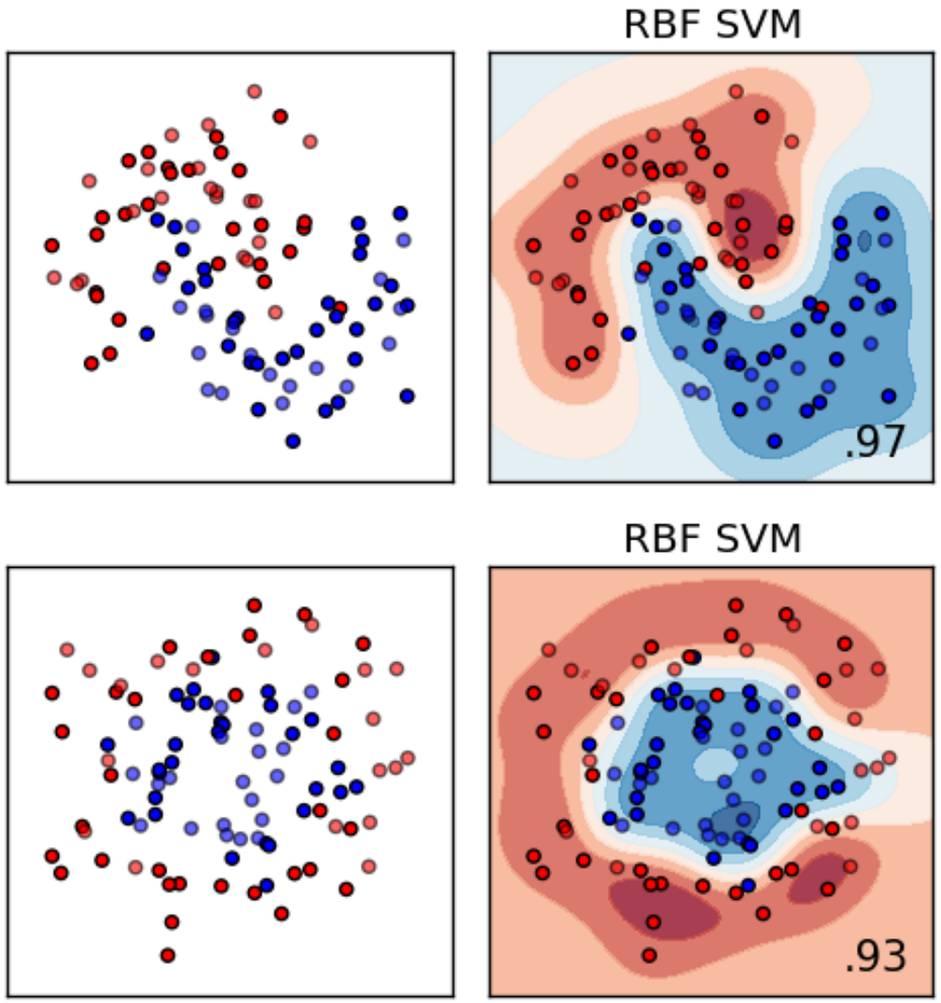


[G. Cowan]

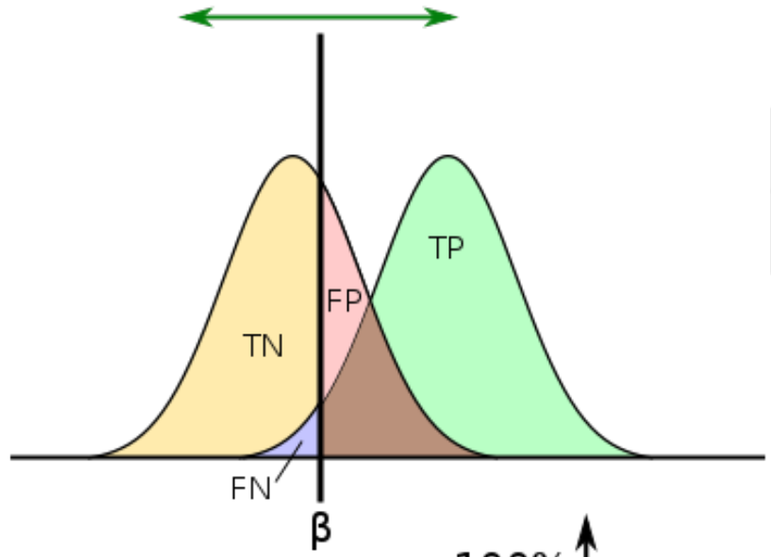




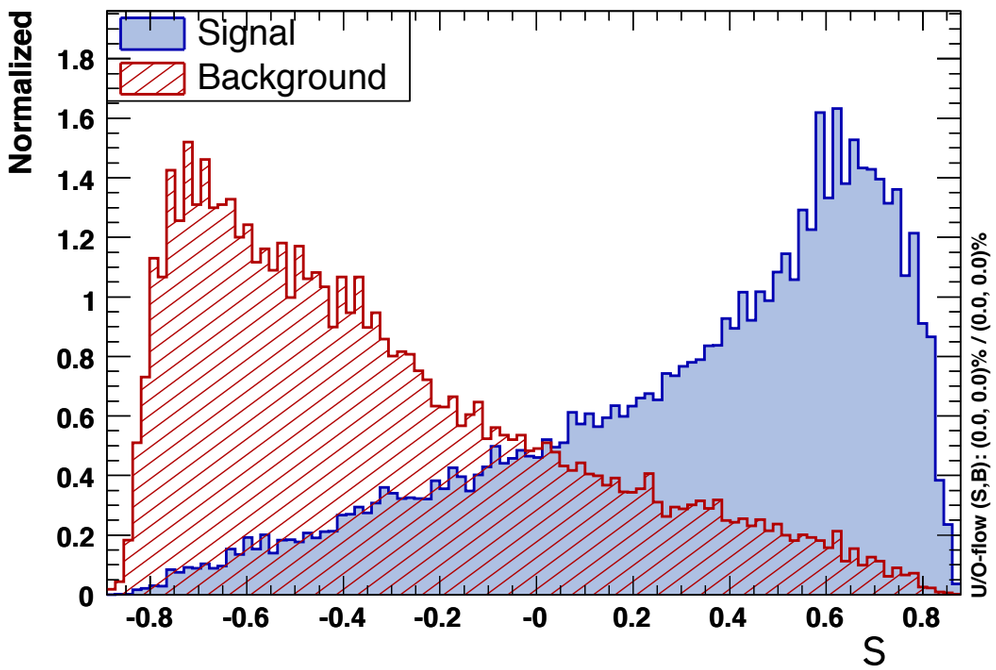
MACHINE LEARNING: CLASSIFIERS

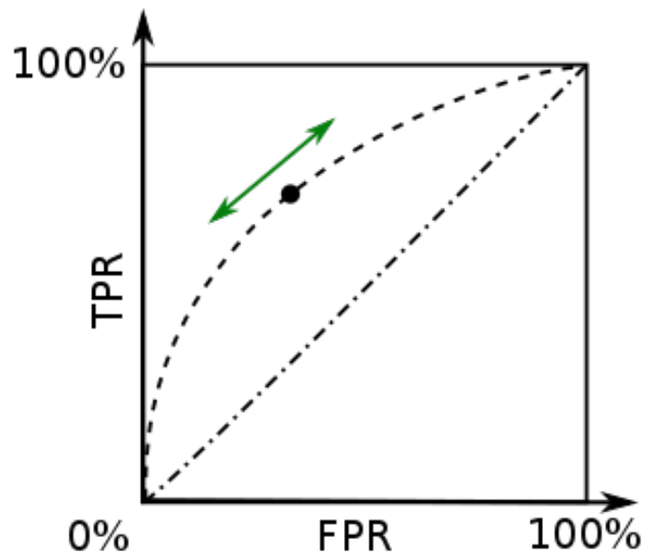


Hypothesis testing between two simple hypotheses H0 vs. H1 is equivalent to binary classification



TP	FP
FN	TN





The Neyman-Pearson Lemma

In 1928-1938 Neyman & Pearson developed a theory in which one must consider competing Hypotheses:

- the Null Hypothesis H_0 (background only)
- the Alternate Hypothesis H_1 (signal-plus-background)

Given some probability that we wrongly reject the Null Hypothesis

$$\alpha = P(x \notin W | H_0)$$

(Convention: if data falls in W then we accept H_0)

Find the region W such that we minimize the probability of wrongly accepting the H_0 (when H_1 is true)

$$\beta = P(x \in W | H_1)$$

The Neyman-Pearson Lemma

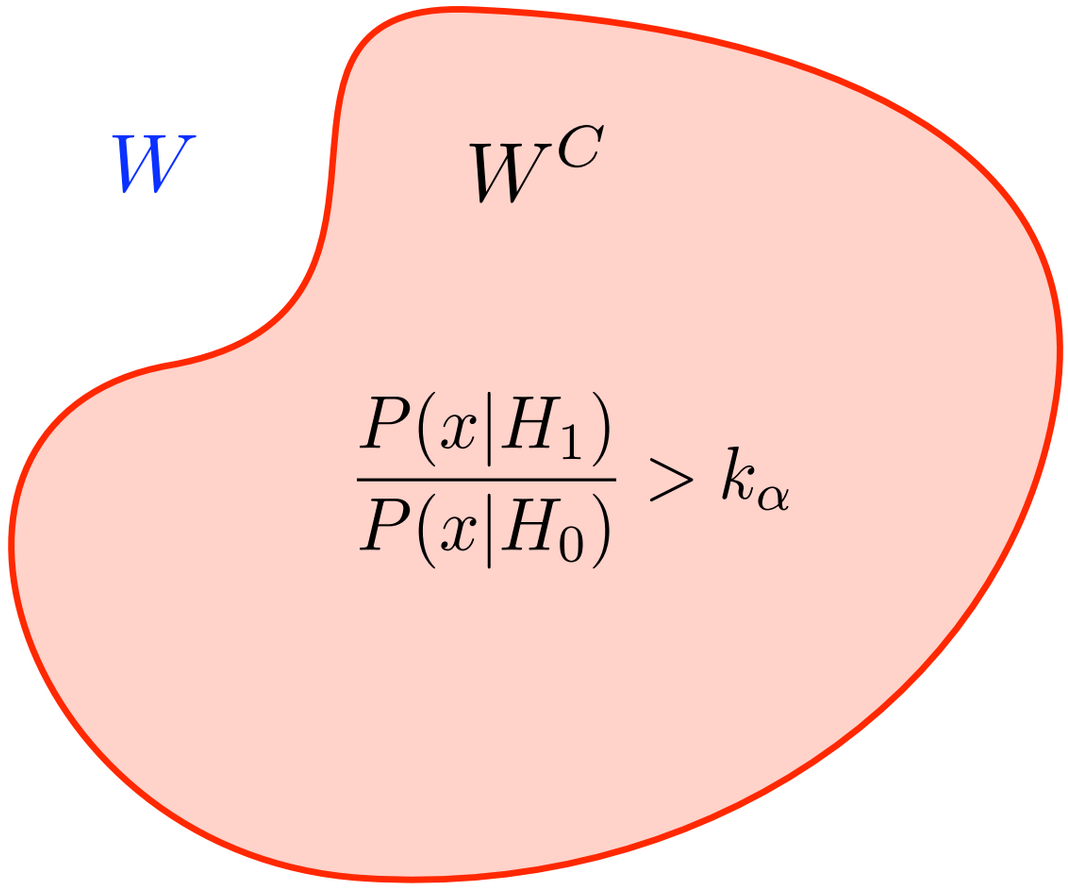
The region W that minimizes the probability of wrongly accepting H_0 is just a contour of the Likelihood Ratio

$$\frac{P(x|H_1)}{P(x|H_0)} > k_\alpha$$

Any other region of the same size will have less power

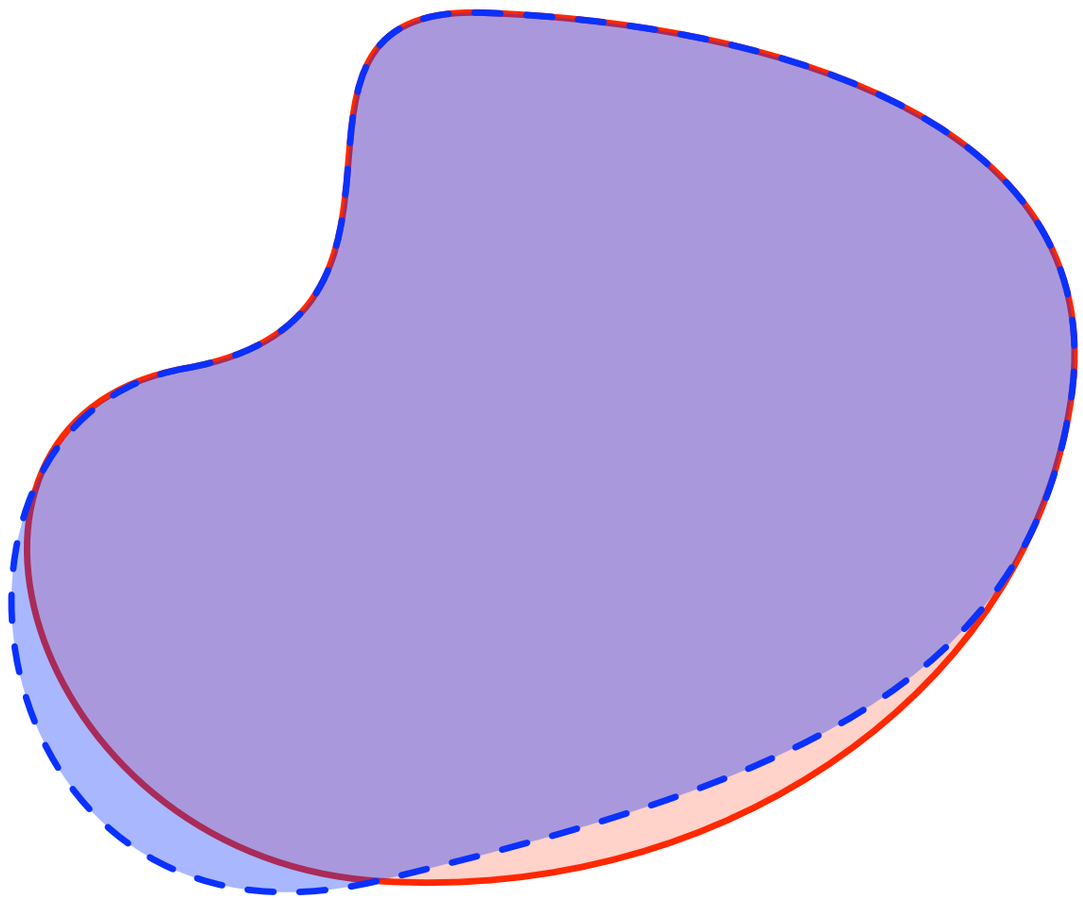
The likelihood ratio is an example of a **Test Statistic**, eg. a real-valued function that summarizes the data in a way relevant to the hypotheses that are being tested

A short proof of Neyman-Pearson



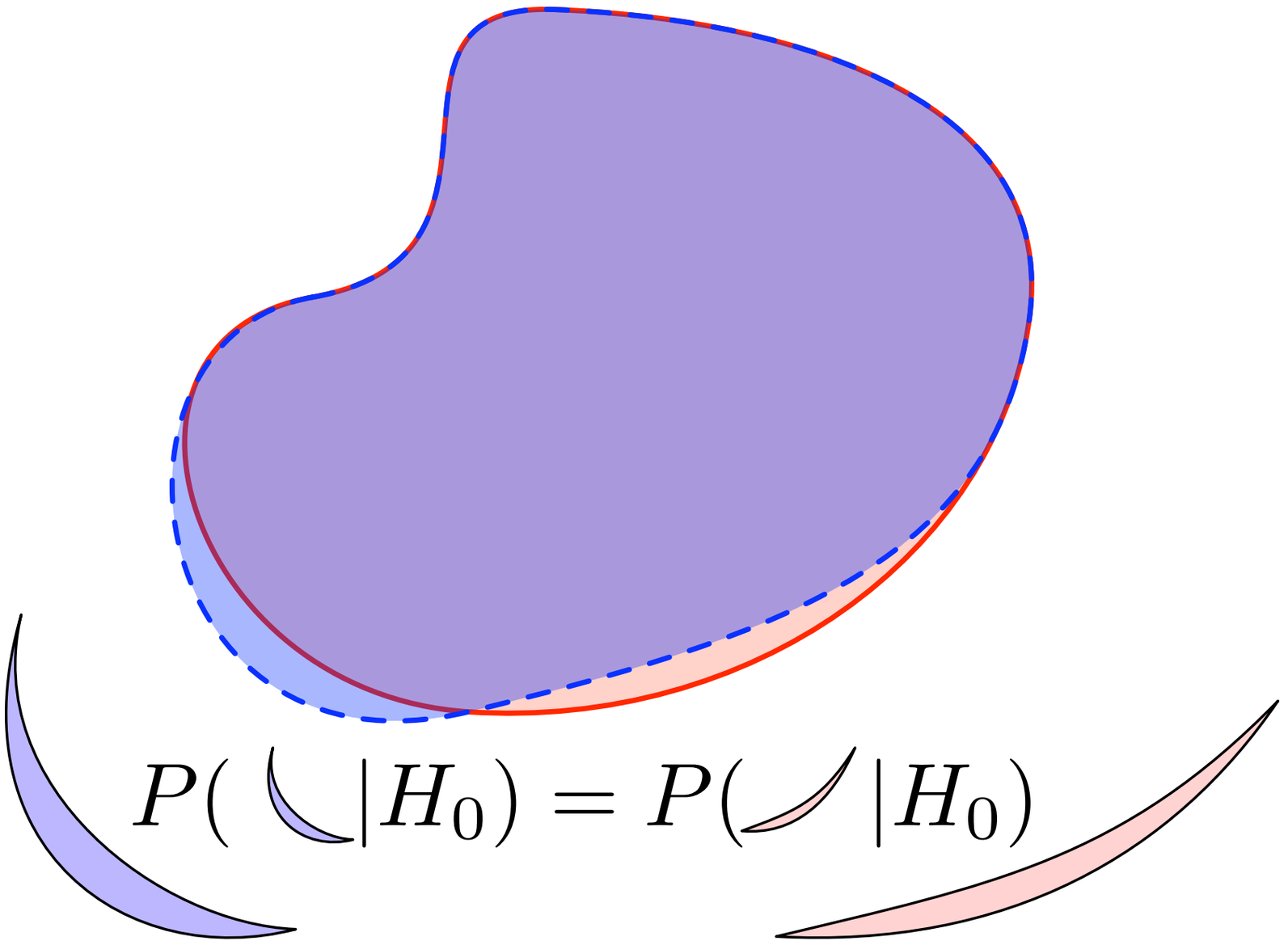
Consider the contour of the likelihood ratio that has size a given size (eg. probability under H_0 is 1-α)

A short proof of Neyman-Pearson



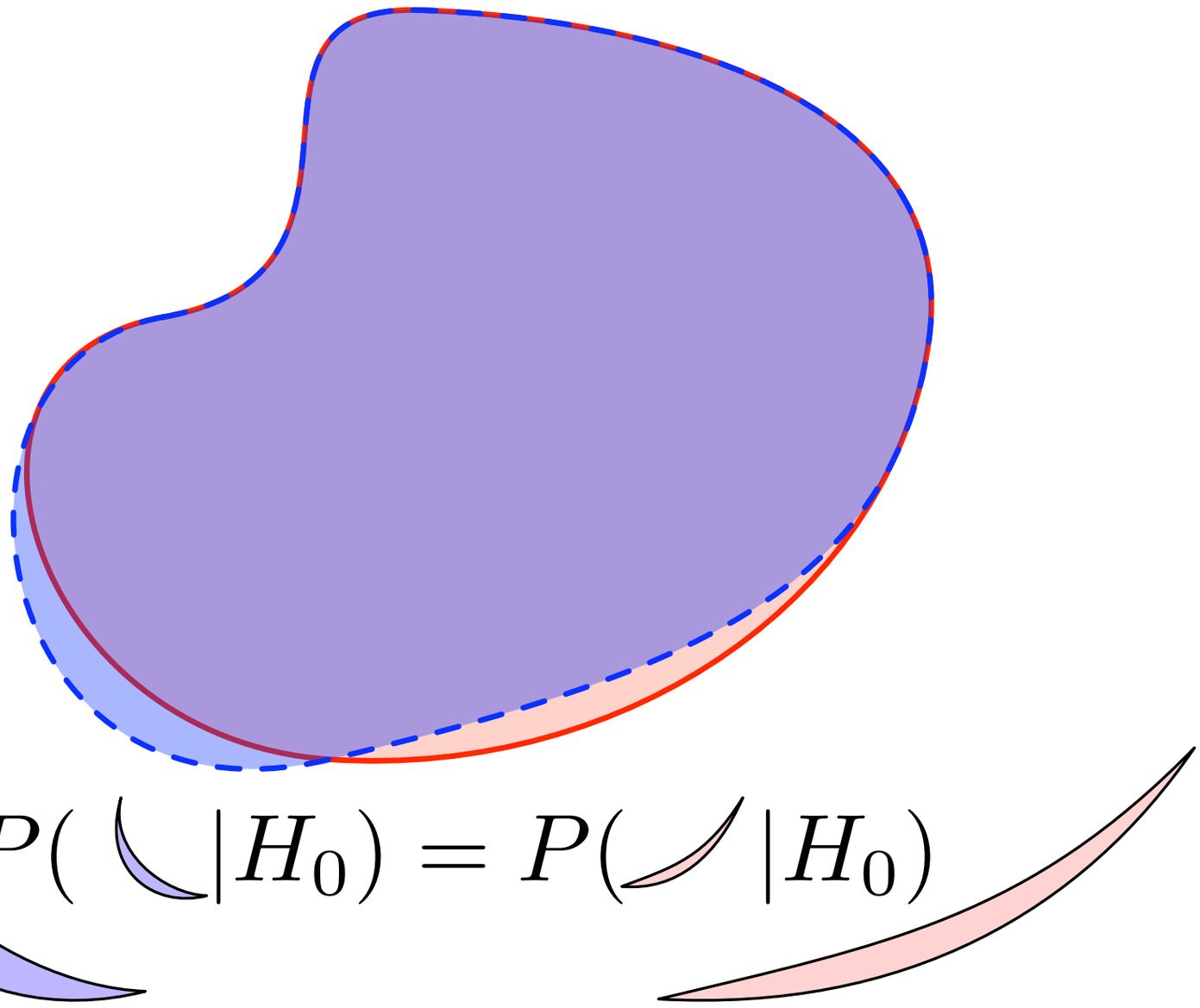
Now consider a variation on the contour that has the same size

A short proof of Neyman-Pearson



Now consider a variation on the contour that has the same size (eg. same probability under H_0)

A short proof of Neyman-Pearson



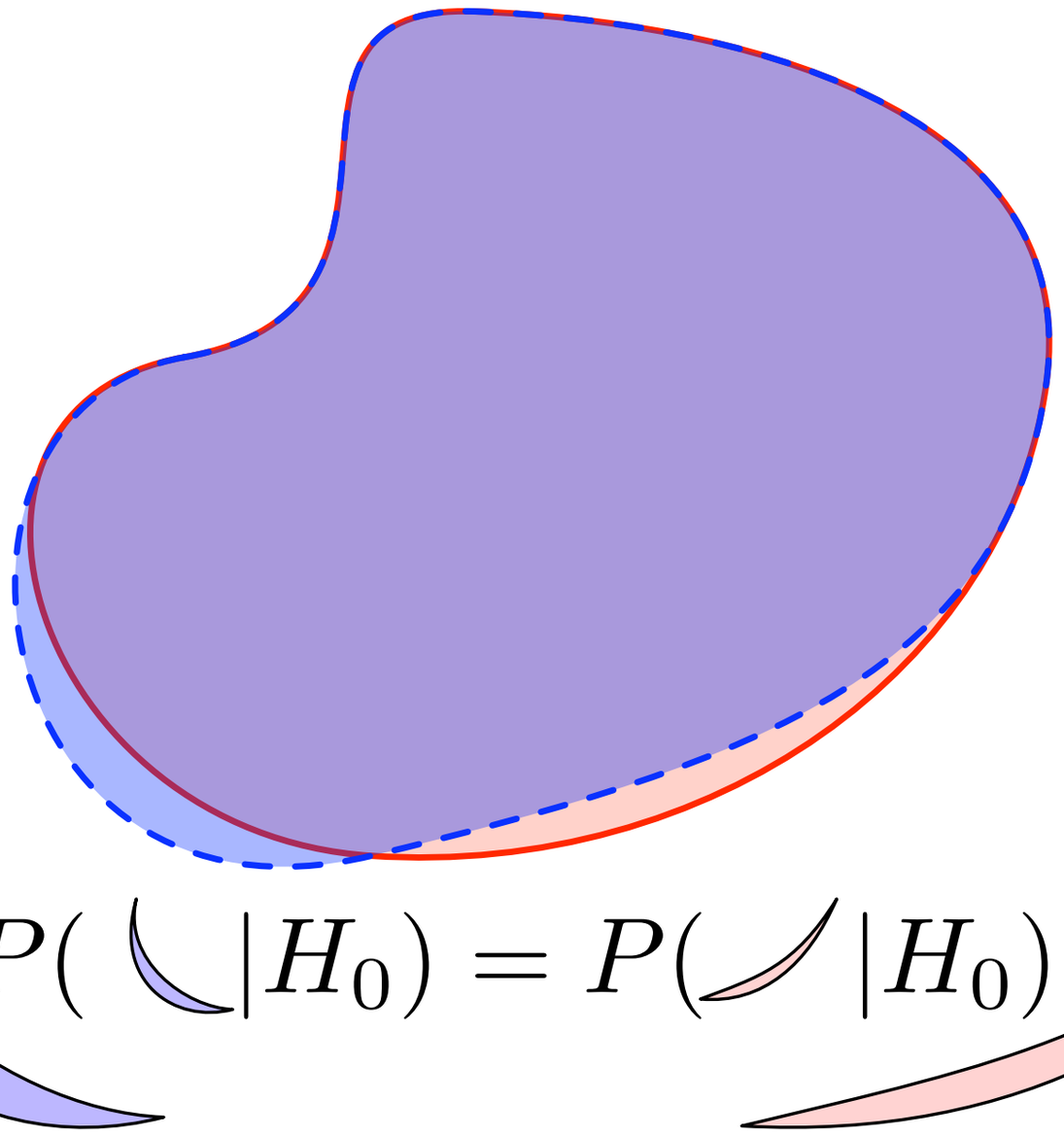
$$P(\ |H_0) = P(\ |H_0)$$

$$\frac{P(x|H_1)}{P(x|H_0)} < k_\alpha$$

$$P(\ |H_1) < P(\ |H_0)k_\alpha$$

Because the new area is outside the contour of the likelihood ratio, we have an inequality

A short proof of Neyman-Pearson



$$P(\ |H_0) = P(\ |H_0)$$

$$\frac{P(x|H_1)}{P(x|H_0)} < k_\alpha$$

$$\frac{P(x|H_1)}{P(x|H_0)} > k_\alpha$$

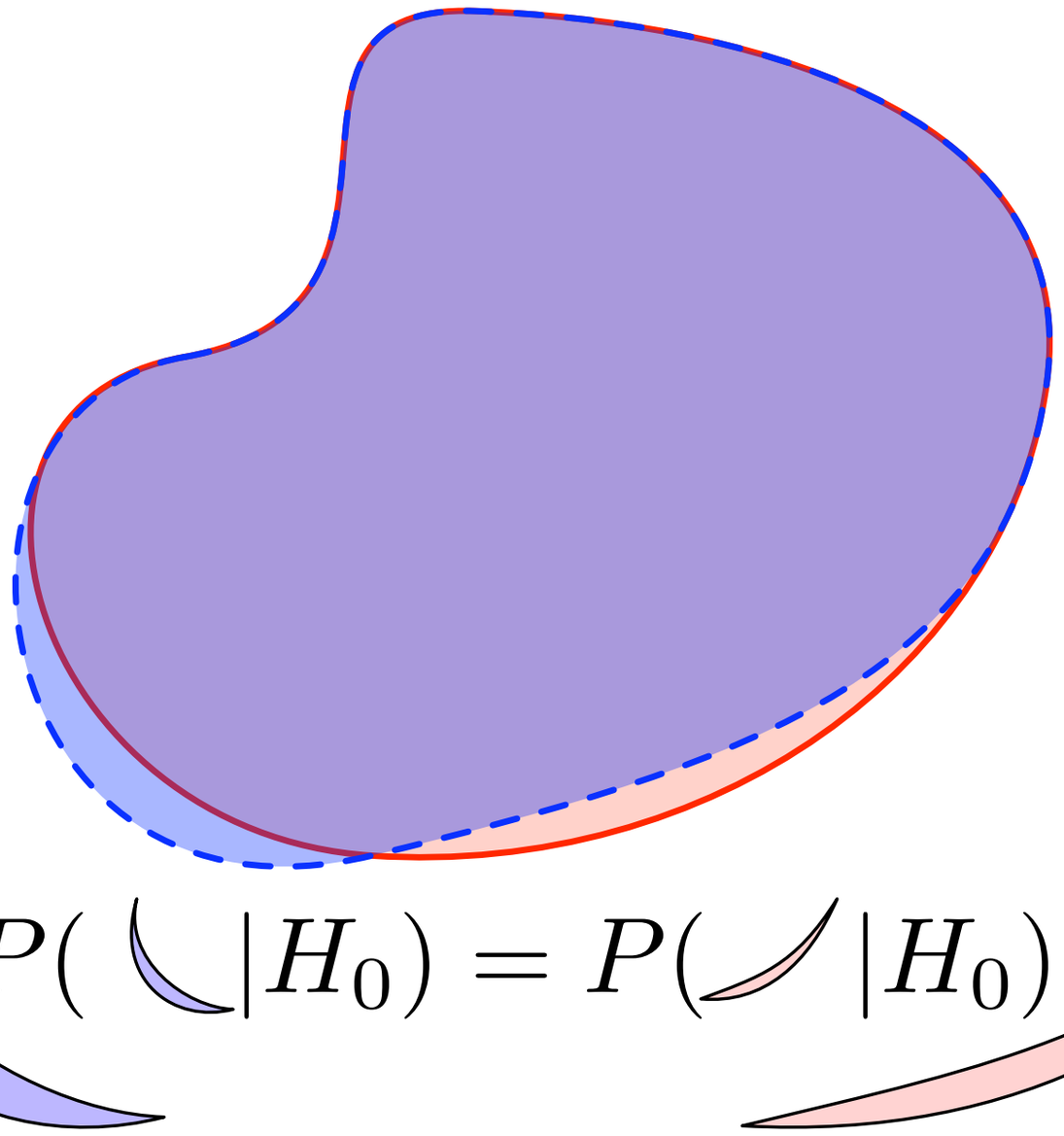
$$P(\ |H_1) < P(\ |H_0) k_\alpha$$

$$P(\ |H_1) > P(\ |H_0) k_\alpha$$

And for the region we lost, we also have an inequality

Together they give...

A short proof of Neyman-Pearson



$$P(\text{ }|H_0) = P(\text{ }|H_0)$$

$$\frac{P(x|H_1)}{P(x|H_0)} < k_\alpha$$

$$\frac{P(x|H_1)}{P(x|H_0)} > k_\alpha$$

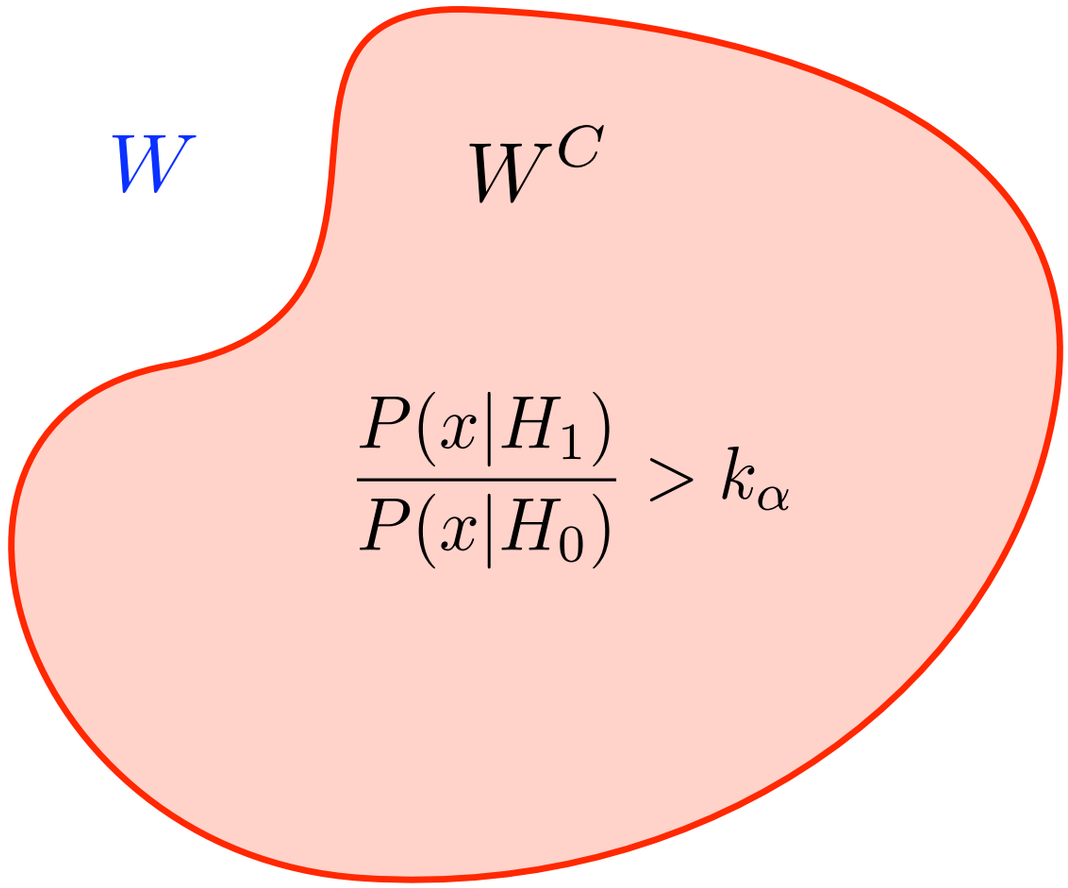
$$P(\text{ }|H_1) < P(\text{ }|H_0)k_\alpha$$

$$P(\text{ }|H_1) > P(\text{ }|H_0)k_\alpha$$

$$P(\text{ }|H_1) < P(\text{ }|H_1)$$

The new region region has less power.

Problem with Neyman-Pearson



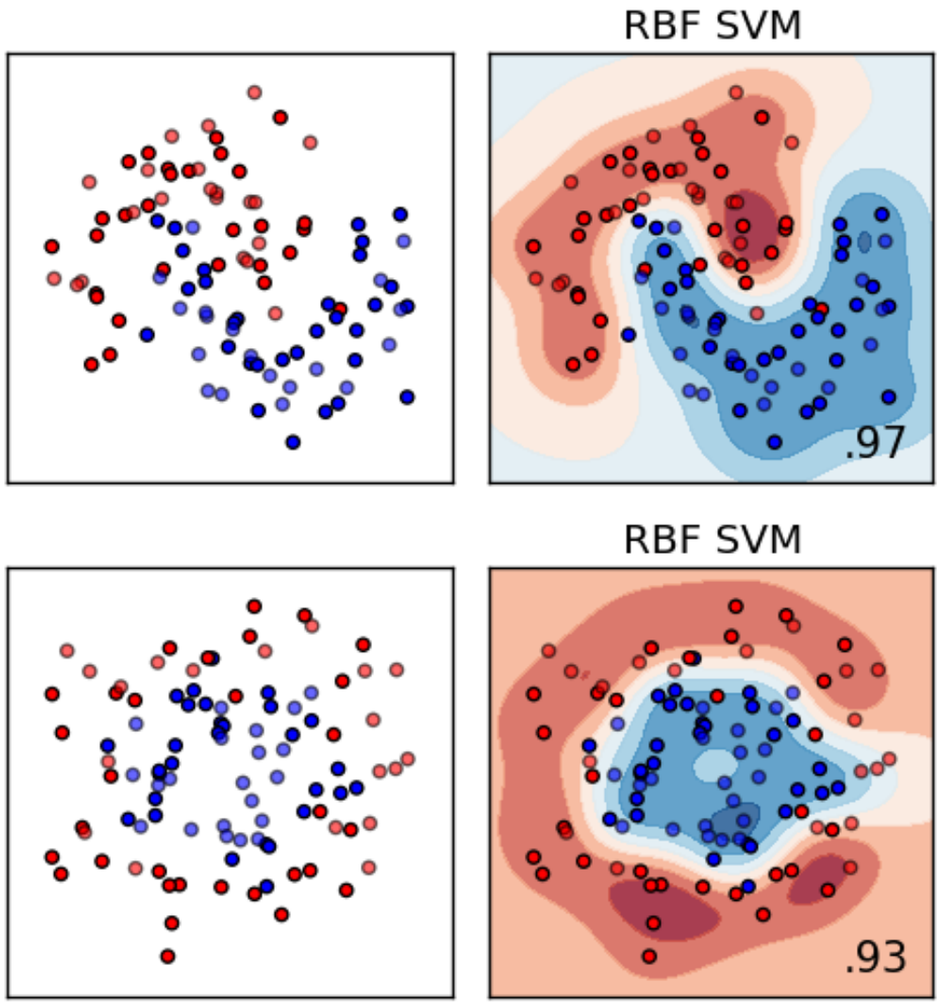
But, if I don’t know $P(x|H_1)$ and $P(x|H_0)$ I can’t evaluate this likelihood ratio!

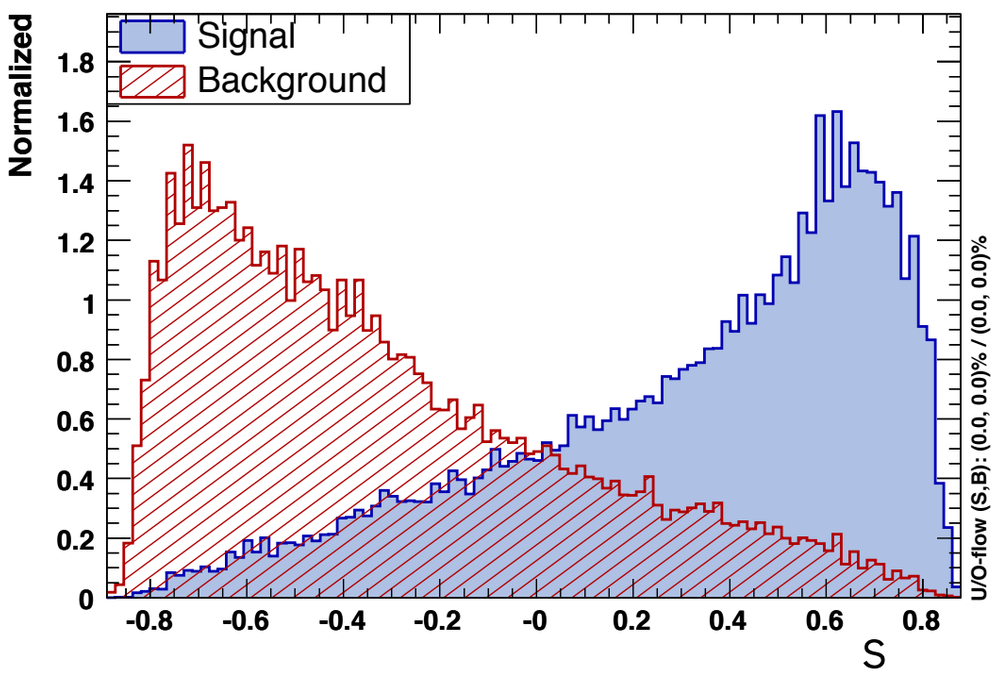
Note, the Neyman-Pearson lemma did not require a prior on H_0 or H_1.

How is this likelihood ratio related to the Bayes optimal classifier?

What do we do if we can't evaluate the likelihood, but we only have samples from H0 & H1?

MACHINE LEARNING: CLASSIFIERS



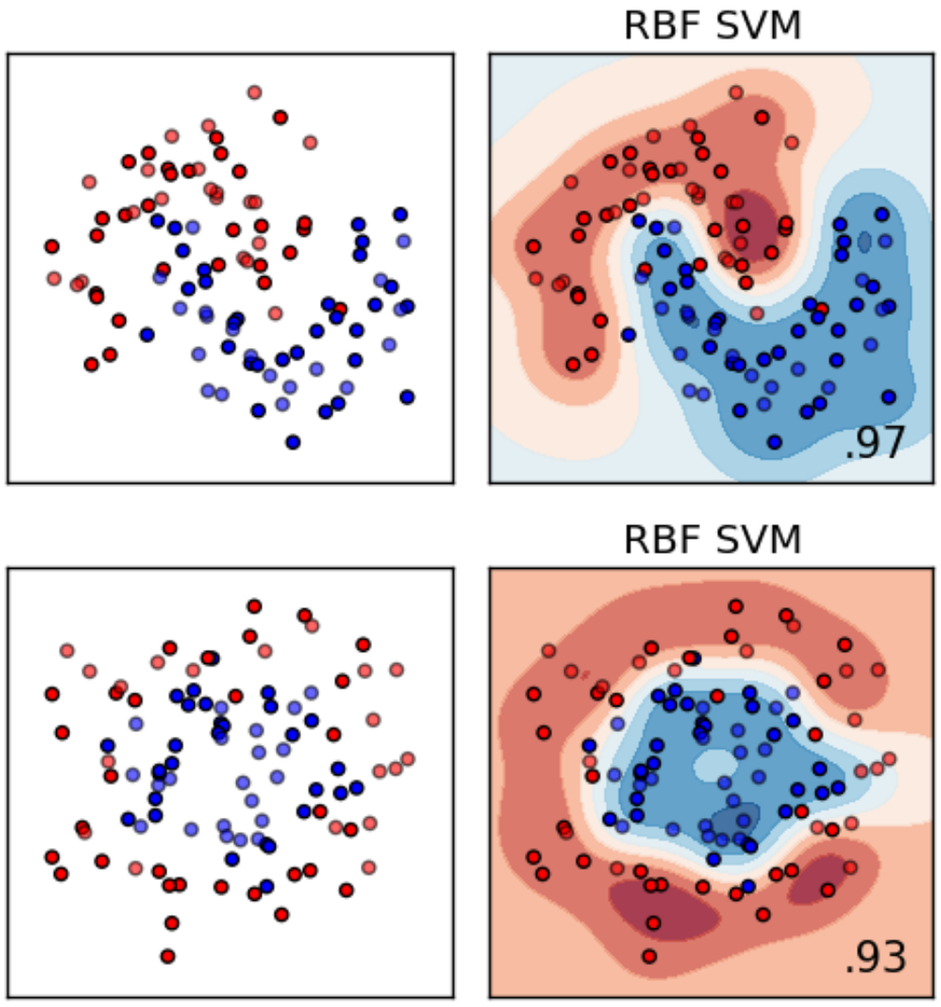


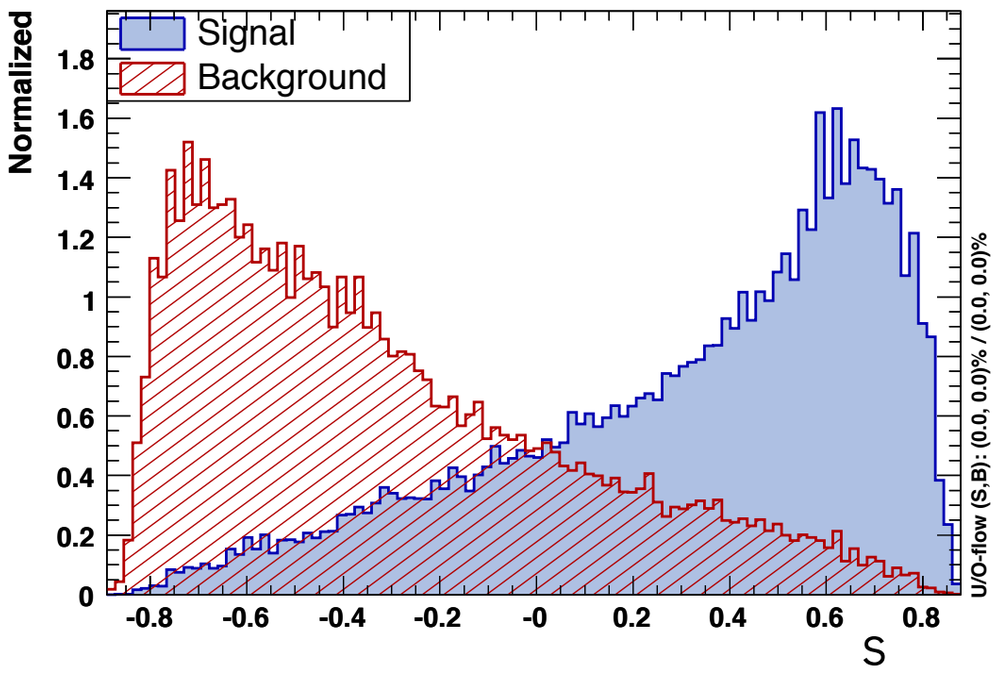
Common to use machine learning classifiers to separate signal (H_1) vs. background (H_0)

- want a function that maps signal to y=1 and background to y=0
- think of it as applied calculus of variations: find function s(x) that minimizes *loss*:

$$L[s] = \int p(x|H_0)\,(0 - s(x))^2\,dx + \int p(x|H_1)\,(1 - s(x))^2 dx \approx \sum_i (y_i - s(x_i))^2$$

MACHINE LEARNING: CLASSIFIERS





- applied calculus of variations: find function s(x) that minimizes *loss*:

$$L[s] = \int p(x|H_0)\,(0 - s(x))^2\,dx + \int p(x|H_1)\,(1 - s(x))^2 dx \approx \sum_i (y_i - s(x_i))^2$$

- the optimal classifier would learn the regression function

$$s(x) = \frac{p(x|H_1)}{p(x|H_0) + p(x|H_1)}$$

- which is 1-to-1 with the likelihood ratio

$$\frac{p(x|H_1)}{p(x|H_0)}$$

FROM SIMPLE TO COMPOSITE HYPOTHESES

Discussion of Hypothesis Testing was for two simple hypotheses $p(x|H0)$ & $p(x|H1)$

- **Simple hypothesis** means distribution is totally specified, no free parameters

In contrast, $p(x|\theta)$ is a **composite hypothesis**.

One might want to:

- provide a **point estimate** $\hat{\theta}(x)$
- provide intervals of θ compatible with the data
  - here, notions of Type I & Type II error depend on θ

Confidence Intervals and Credible Intervals

Confidence Interval

What is a "Confidence Interval?

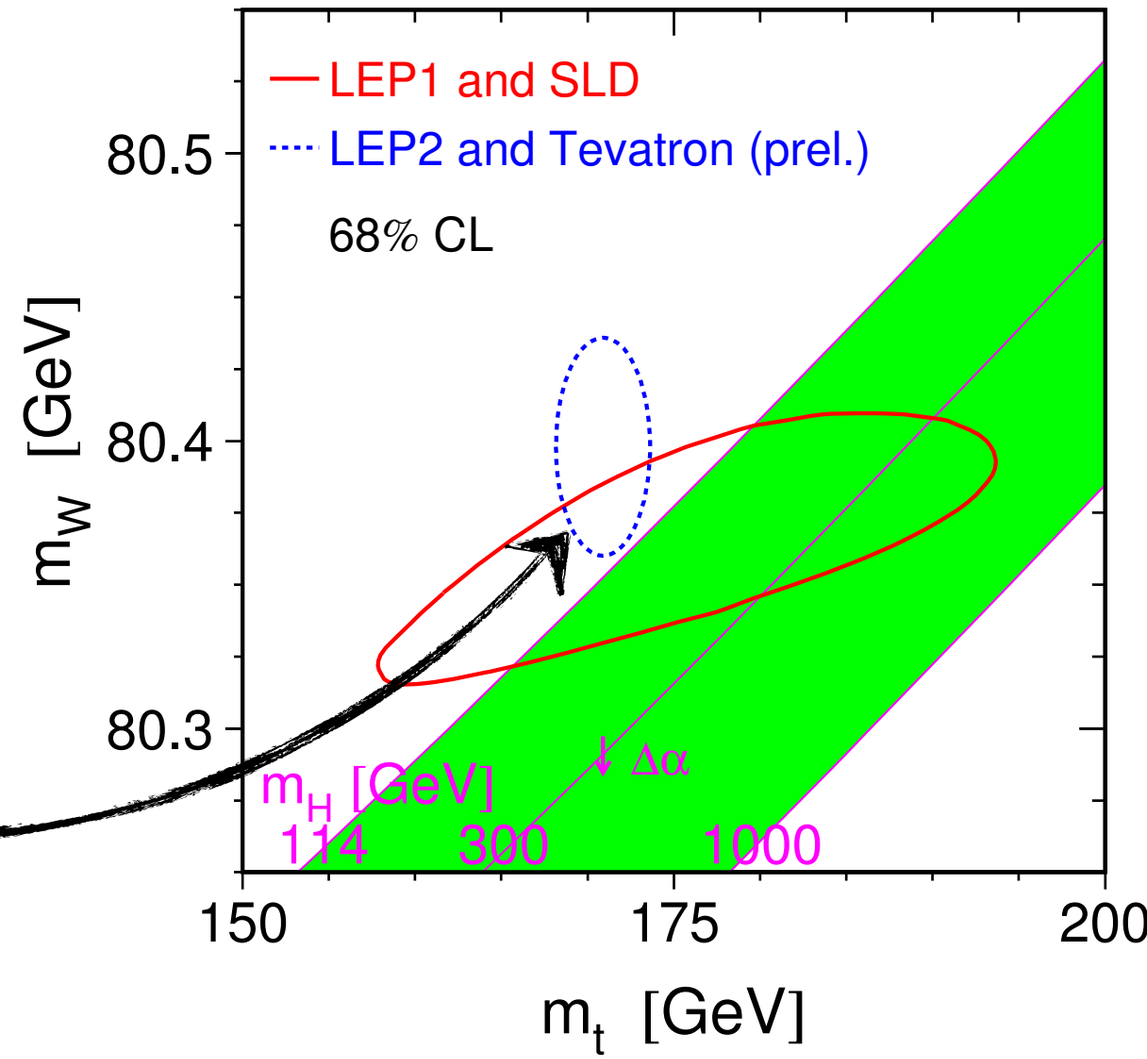
- goes beyond point estimate $\hat{\theta}$
- provides uncertainty quantification

Want to say there is a 68% chance that the true value of parameter $\theta=(m_W, m_t)$ is in this interval

- that's $P(\theta|x) = P(\text{theory}|\text{data})$!

Correct frequentist statement is that the interval **covers** the true value 68% of the time

- remember, the contour is a function of the data, which is random. So it moves around from experiment to experiment



- Bayesian "credible interval" does refer to probability parameter is in interval V. The procedure is very intuitive:

$$P(\theta \in V) = \int_V \pi(\theta|x) = \int_V d\theta \frac{f(x|\theta)\pi(\theta)}{\int d\theta f(x|\theta)\pi(\theta)}$$

Inverting Hypothesis Tests

There is a precise dictionary that explains how to move from from hypothesis testing to confidence intervals

- **Type I error:** probability interval does not cover true value of the parameters (eg. it is now a function of the parameters)
- **Power** is probability interval does not cover a false value of the parameters (eg. it is now a function of the parameters)
  - We don't know the true value, consider each point θ_0 as if it were true

What about null and alternate hypotheses?

- when testing a point θ_0 it is considered the null
- all other points considered "alternate"

So what about the Neyman-Pearson lemma & Likelihood ratio?

- as mentioned earlier, there are no guarantees like before
- a common generalization that has good power is:

$$\frac{f(x|H_0)}{f(x|H_1)} \quad \Rightarrow \quad \frac{f(x|\theta_0)}{f(x|\theta_{best}(x))}$$

GENERALIZING THE LIKELIHOOD RATIO WITH NUISANCE PARAMETERS

Initially, we started with 2 simple hypotheses, and showed the likelihood ratio was most powerful (Neyman-Pearson)

Initially, we started with 2 simple hypotheses, and showed the likelihood ratio was most powerful (Neyman-Pearson)

How do we generalized it to composite hypotheses.

$$\frac{f(x|H_0)}{f(x|H_1)} \quad \Rightarrow \quad \frac{f(x|\theta_0)}{f(x|\theta_{best}(x))}$$

GENERALIZING THE LIKELIHOOD RATIO WITH NUISANCE PARAMETERS

Initially, we started with 2 simple hypotheses, and showed the likelihood ratio was most powerful (Neyman-Pearson)

How do we generalized it to composite hypotheses.

How do we generalize it to include nuisance parameters?

Initially, we started with 2 simple hypotheses, and showed the likelihood ratio was most powerful (Neyman-Pearson)

How do we generalized it to composite hypotheses.

How do we generalize it to include nuisance parameters?

Variable	Meaning
θ_r	physics parameters
θ_s	nuisance parameters
θ_r, θ_s	unconditionally maximize $L(x\|\theta_r, \theta_s)$
$\hat{\hat{\theta}}_s$	conditionally maximize $L(x\|\theta_{r0}, \hat{\hat{\theta}}_s)$

From Kendall

Initially, we started with 2 simple hypotheses, and showed the likelihood ratio was most powerful (Neyman-Pearson)

How do we generalized it to composite hypotheses.

How do we generalize it to include nuisance parameters?

Variable	Meaning
θ_r	physics parameters
θ_s	nuisance parameters
θ_r, θ_s	unconditionally maximize $L(x\|\theta_r, \theta_s)$
$\hat{\hat{\theta}}_s$	conditionally maximize $L(x\|\theta_{r0}, \hat{\hat{\theta}}_s)$

$(H_0 : \theta_r = \theta_{r0})$
$(H_1 : \theta_r \neq \theta_{r0})$

From Kendall

Initially, we started with 2 simple hypotheses, and showed the likelihood ratio was most powerful (Neyman-Pearson)

How do we generalized it to composite hypotheses.

How do we generalize it to include nuisance parameters?

Variable	Meaning
θ_r	physics parameters
θ_s	nuisance parameters
θ_r, θ_s	unconditionally maximize $L(x\|\theta_r, \theta_s)$
$\hat{\hat{\theta}}_s$	conditionally maximize $L(x\|\theta_{r0}, \hat{\hat{\theta}}_s)$

$(H_0 : \theta_r = \theta_{r0})$
$(H_1 : \theta_r \neq \theta_{r0})$

Now consider the Likelihood Ratio

$$l = \frac{L(x|\theta_{r0}, \hat{\hat{\theta}}_s)}{L(x|\hat{\theta}_r, \hat{\theta}_s)} = \lambda(\theta_{r0})$$

Intuitively l is a reasonable test statistic for H_0: it is the maximum likelihood under H_0 as a fraction of its largest possible value, and large values of l signify that H_0 is reasonably acceptable.

From Kendall

PROPERTIES OF THE PROFILE LIKELIHOOD RATIO

After a close look at the "profile" likelihood ratio (aka generalized likelihood ratio)

$$l = \frac{L(x|\theta_{r0}, \hat{\hat{\theta}}_s)}{L(x|\hat{\theta}_r, \hat{\theta}_s)} = \lambda(\theta_{r0})$$

one can see that one does not need to specify nuisance parameter θ_s to evaluate the function

- though its distribution might depend indirectly

Wilks's theorem states that under certain conditions the distribution of $-2 \ln \lambda\ (\theta_r = \theta_{r0})$ given that the true value of θ_r *is* θ_{r0} converges to a chi-square distribution

- "asymptotic distribution" is known and it is independent of θ_s !

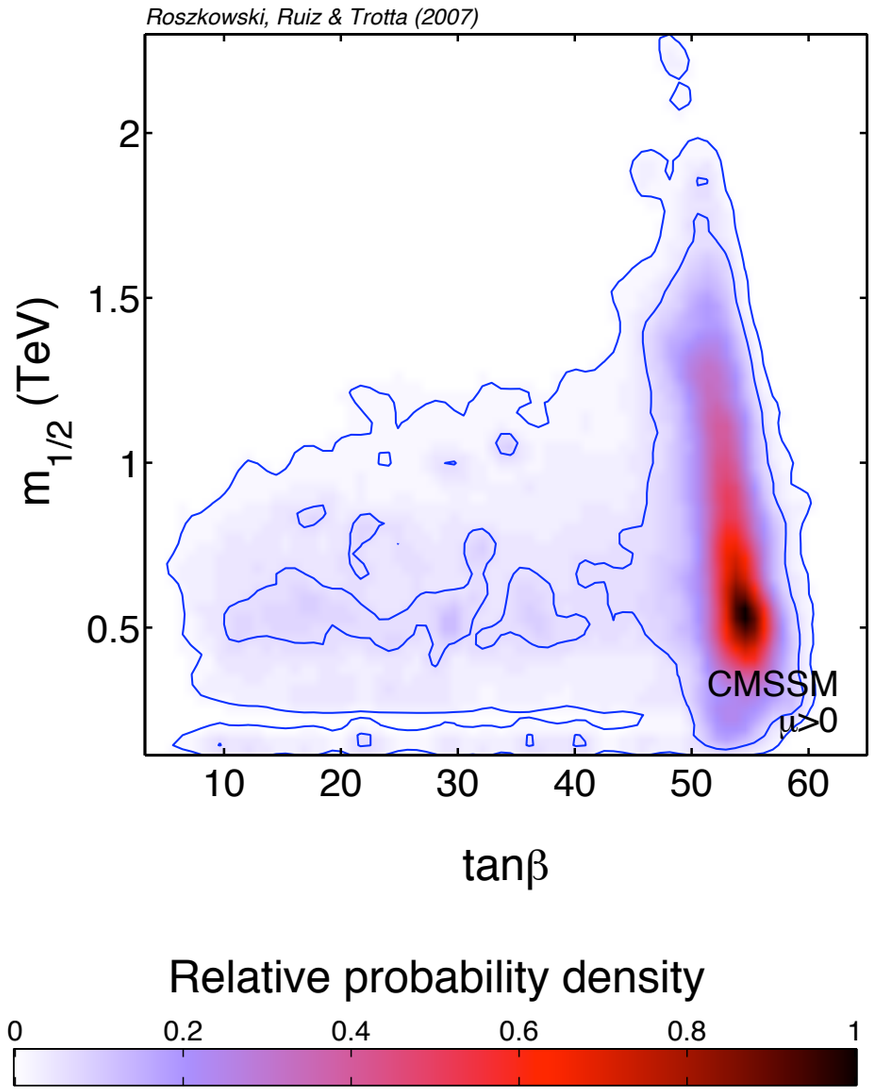
- some regularity conditions are needed

Thus, we can calculate the p-values for the θ_r hypothesis that are robust to nuisance parameter without having to sample x from $p(x|\theta_r, \theta_s)$

BAYESIAN CREDIBLE INTERVALS

Bayesian “credible interval” V does mean that there is a 95% that the probability parameter is in interval.

The procedure is very intuitive:



$$P(\theta \in V) = \int_V \pi(\theta|x) = \int_V d\theta \frac{f(x|\theta)\pi(\theta)}{\int d\theta f(x|\theta)\pi(\theta)}$$

Markov Chain Monte Carlo

Markov Chain Monte Carlo (MCMC) is a nice technique which will produce a sampling of a parameter space which is proportional to a posterior

- it works well in high dimensional problems
- Metropolis-Hastings Algorithm: generates a sequence of points $\{\vec{\alpha}^{(t)}\}$
  - Given the likelihood function $L(\vec{\alpha})$ & prior $P(\vec{\alpha})$, the posterior is proportional to $L(\vec{\alpha}) \cdot P(\vec{\alpha})$
  - propose a point $\vec{\alpha}'$ to be added to the chain according to a proposal density Q($\vec{\alpha}$'|$\vec{\alpha}$) that depends only on current point $\vec{\alpha}$
  - if posterior is higher at $\vec{\alpha}'$ than at $\vec{\alpha}$, then add new point to chain
  - else: add $\vec{\alpha}'$ to the chain with probability

$$\rho = \frac{L(\vec{\alpha}') \cdot P(\vec{\alpha}')}{L(\vec{\alpha}) \cdot P(\vec{\alpha})} \cdot \frac{Q(\vec{\alpha}|\vec{\alpha}')}{Q(\vec{\alpha}'|\vec{\alpha})}$$

  - (appending original point $\vec{\alpha}$ with complementary probability)

VARIATIONAL INFERENCE

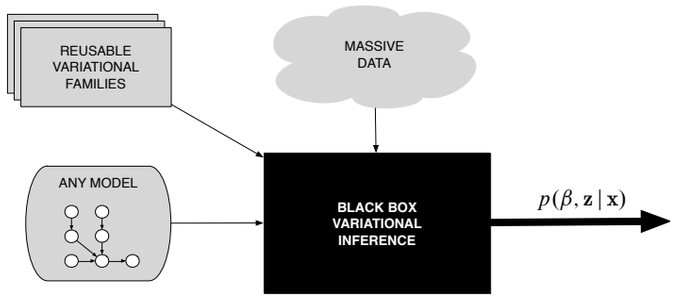
Variational Inference: Foundations and Modern Methods

David Blei, Rajesh Ranganath, Shakir Mohamed

NIPS 2016 Tutorial · December 5, 2016

COLUMBIA UNIVERSITY IN THE CITY OF NEW YORK
PRINCETON UNIVERSITY
DeepMind

Black Box Variational Inference (BBVI)



The requirements for inference

The noisy gradient:

$$\frac{1}{S}\sum_{s=1}^{S}\nabla_{\nu}\log q(\mathbf{z}_s;\boldsymbol{\nu})(\log p(\mathbf{x},\mathbf{z}_s)-\log q(\mathbf{z}_s;\boldsymbol{\nu})),$$

where $\mathbf{z}_s \sim q(\mathbf{z};\boldsymbol{\nu})$

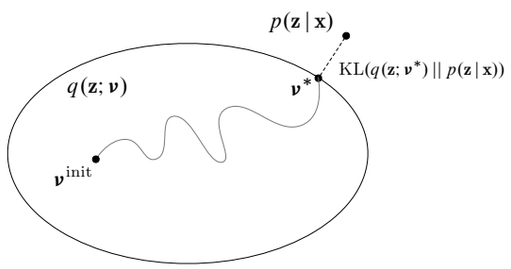
To compute the noisy gradient of the ELBO we need

- Sampling from $q(\mathbf{z})$
- Evaluating $\nabla_{\nu}\log q(\mathbf{z};\boldsymbol{\nu})$
- Evaluating $\log p(\mathbf{x},\mathbf{z})$ and $\log q(\mathbf{z})$

There is no model specific work: black box criteria are satisfied

need likelihood

Variational Inference: Foundations and Modern Methods



VI approximates difficult quantities from complex models.

With **stochastic optimization** we can

- scale up VI to massive data
- enable VI on a wide class of difficult models
- enable VI with elaborate and flexible families of approximations

TL;DR

The reason for those slides was just to reinforce that none of those procedures work if you don't know the likelihood function $p(x|\theta)$.

Parameter Estimation

Parameter Estimation

- Let us focus on (1) first. $\{\mathbf{X}^1, \ldots, \mathbf{X}^L\} \sim p^* \text{ iid.}$
- Suppose $p^* = p_{\theta^*} \text{ for some } \theta^*.$
- Two main approaches for parameter estimation:
  - Maximum Likelihood Estimation:

$$E(\theta) = \log p(\{\mathbf{X}^1, \ldots, \mathbf{X}^L\} \mid \theta) = \sum_{l<L} \log p(\mathbf{X}^l \mid \theta)$$

$$\hat{\theta}_{MLE} = \arg\max_{\theta} E(\theta)$$

  - Under appropriate assumptions, $\hat{\theta}_{MLE}$ is
    - consistent (as sample size grows, $\hat{\theta}_{MLE} \to \theta^* \text{ (in probability)}$
    - asymptotically efficient (no other consistent estimator has lower asymptotic mean-squared error).
  - However, in general this estimation is computationally intractable.

Cramér-Rao Bound

The minimum variance bound on an estimator is given by the Cramér-Rao inequality:

- simple univariate case:

$$\mathrm{Var}[\hat{\theta}|\theta] = E[(\hat{\theta} - E[\theta|\theta])^2]\,|\theta]$$

- For an unbiased estimator the Cramér-Rao bound states

$$\mathrm{Var}[\hat{\theta}|\theta] \geq \frac{1}{I(\theta)}$$

- where $I(\theta)$ is the Fisher information

$$(\mathcal{I}\,(\theta))_{i,j} = \mathrm{E}\left[\frac{\partial}{\partial\theta_i}\ln f(X;\theta)\frac{\partial}{\partial\theta_j}\ln f(X;\theta)\middle|\theta\right].$$

- General form for multiple parameters:

$$\mathrm{cov}[\hat{\theta}|\theta]_{ij} \geq I^{-1}_{ij}(\theta)$$

Maximum Likelihood Estimators *asymptotically* reach this bound

If we could approximate the conditional density p(x|θ), then we could

- approximate the maximum likelihood estimate

- construct frequentist confidence intervals or Bayesian credible intervals

Loss functions for Density Estimation

Task-driven inference

- Depending on the task, we might want to perform different kinds of estimation.
  1. Density Estimation: we are interested in the joint distribution, which can be subsequently used to perform any inference query.
  2. Prediction: we are only interested in a specific set of conditional distribution, e.g classification, or output prediction.
  3. Structural discovery: We are interested in the graph itself (not so much the parameters), e.g. determining dependencies between genes.

- (1) is typically harder than (2). (3) is typically harder than (2) and (1).

CROSS ENTROPY

What function r(x) minimizes the cross-entropy?

$$L[r] = -\int \underbrace{p(x)\log r(x)}_{F(x,r)}\, dx$$

- Subject to $\int r(x)dx = 1$

CROSS ENTROPY

What function r(x) minimizes the cross-entropy?

$$L[r] = -\int \underbrace{p(x)\log r(x)}_{F(x,r)}\, dx$$

- Subject to $\int r(x)dx = 1$

Euler-Lagrange Equation w/ Lagrange-multiplier

$$L[r,\lambda] = F(x,r) + \lambda r(x)$$

$$\underbrace{\frac{d}{dx}\left(\frac{\delta L}{\delta r'}\right)}_{=0} - \frac{\delta L}{\delta r} = 0 \qquad \frac{\delta L}{\delta r} = 0 = \frac{-p(x)}{r(x)} + \lambda$$

$$r(x) = p(x)/\lambda$$

imposing the constraint gives $\lambda = 1$ thus $r(x) = p(x)$

SQUARED LOSS

What function r(x) minimizes the squared loss?

$$L[r] = -\int \underbrace{p(x)(p(x) - r(x))^2}_{F(x,r)} dx$$

- Subject to $\int r(x)dx = 1$

SQUARED LOSS

What function r(x) minimizes the squared loss?

$$L[r] = -\int \underbrace{p(x)(p(x) - r(x))^2}_{F(x,r)} dx$$

- Subject to $\int r(x)dx = 1$

Euler-Lagrange Equation w/ Lagrange-multiplier

$$L[r, \lambda] = F(x, r) + \lambda r(x)$$

$$\underbrace{\frac{d}{dx}\left(\frac{\delta L}{\delta r'}\right)}_{=0} - \frac{\delta L}{\delta r} = 0 \qquad \frac{\delta L}{\delta r} = 0 = \lambda - 2p(x)(p(x) - r(x))$$

$$r(x) = p - \frac{\lambda}{2p}$$

imposing the constraint gives $\lambda = 0$ thus $r(x) = p(x)$

APPROXIMATING FROM DATA

If we have samples from an unknown p(x): $\{x_i\}_{i=1}^N \sim p(x)$

We can effectively approximate the true cross-entropy loss:

$$L[r] = -\int \underbrace{p(x)\log r(x)}_{F(x,r)}\, dx \approx \frac{1}{N}\sum_{i=1}^{N} \log r(x_i)$$

and approximate p(x) even though we can't evaluate it.

In contrast, we can't use the squared loss if since can't evaluate p(x):

$$L[r] = -\int \underbrace{p(x)(p(x) - r(x))^2}_{F(x,r)}\, dx \approx \frac{1}{N}\sum_{i=1}^{N} \log(p(x_i) - r(x_i))^2$$

FLOWS & AUTOREGRESSIVE MODELS

http://beta.briefideas.org/ideas/5c2f74aedbf3618ca180382e393c7617

Recent work in density estimation uses a bijection $f : X \to Z$ (e.g. an invertible flow or autoregressive model) and a tractable density $p(z)$ (e.g. [1] [2] [3] [4]).

$$p(x) = p(f_\phi(x)) \left| \det \left(\frac{\partial f_\phi(x)}{\partial x_T} \right) \right| ,$$

where ϕ are the internal network parameters for the bijection f_ϕ. Learning proceeds via gradient ascent $\nabla_\phi \sum_i \log p(x_i)$ with data x_i (i.e. maximum likelihood wrt. the internal parameters ϕ). Since f is invertible, then this model can also be used as a generative model for X.

This can be generalized to the conditional density $p(x|\theta)$ by utilizing a family of bijections $f_\theta : X \to Z$ parametrized by θ (e.g. [5] [6]).

$$p(x|\theta) = p(f_{\phi;\theta}(x)) \left| \det \left(\frac{\partial f_{\phi;\theta}(x)}{\partial x_T} \right) \right|$$

Here θ and x are input to the network (and its inverse) and ϕ are internal network parameters. Again, learning proceeds via gradient ascent $\nabla_\phi \sum_i \log p(x_i|\theta_i)$ with data x_i, θ_i.

We observe that not only can this model be used as a conditional generative model $p(x|\theta)$, but it can also be used to perform asymptotically exact, amortized likelihood-free inference on θ.

This is particularly interesting when θ is identified with the parameters of an intractable, non-differentiable computer simulation or the conditions of some real world data collection process.

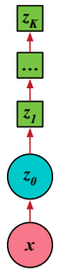
CHANGE OF VARIABLES

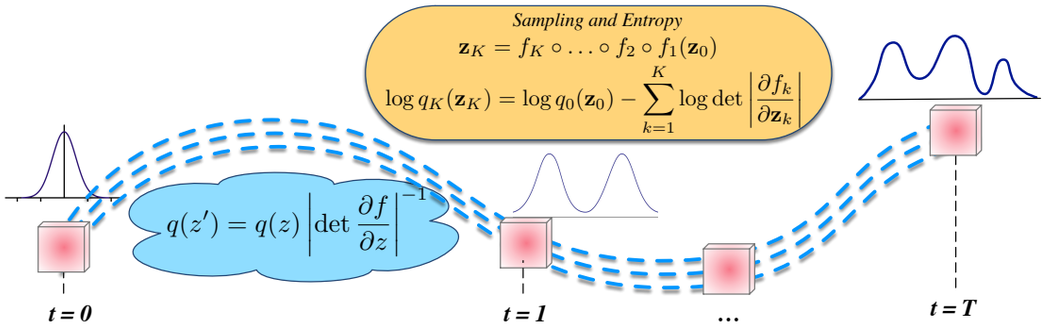
Flows to be covered in lecture 10

Approximations using Change-of-variables

Exploit the rule for change of variables for random variables:

- Begin with an initial distribution $q_0(\mathbf{z}_0|\mathbf{x})$.
- Apply a sequence of K invertible functions f_k.





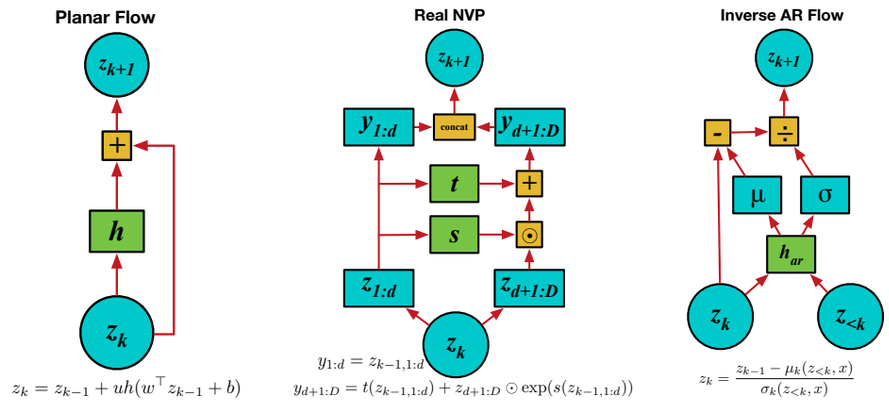
Distribution flows through a sequence of invertible transforms

[Rezende and Mohamed, 2015]

Choice of Transformation Function

$$\mathscr{L} = \mathbb{E}_{q_0(\mathbf{z}_0)}[\log p(\mathbf{x}, \mathbf{z}_K)] - \mathbb{E}_{q_0(\mathbf{z}_0)}[\log q_0(\mathbf{z}_0)] - \mathbb{E}_{q_0(\mathbf{z}_0)}\left[\sum_{k=1}^{K} \log \det \left| \frac{\partial f_k}{\partial \mathbf{z}_k} \right|\right]$$

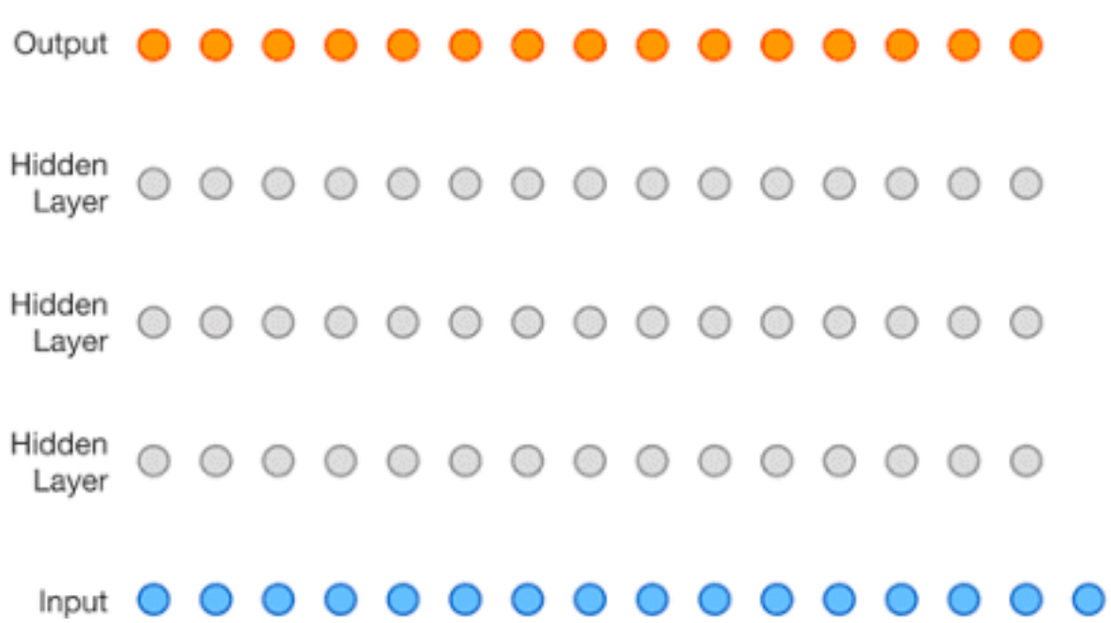
- Begin with a fully-factorised Gaussian and improve by change of variables.
- Triangular Jacobians allow for computational efficiency.

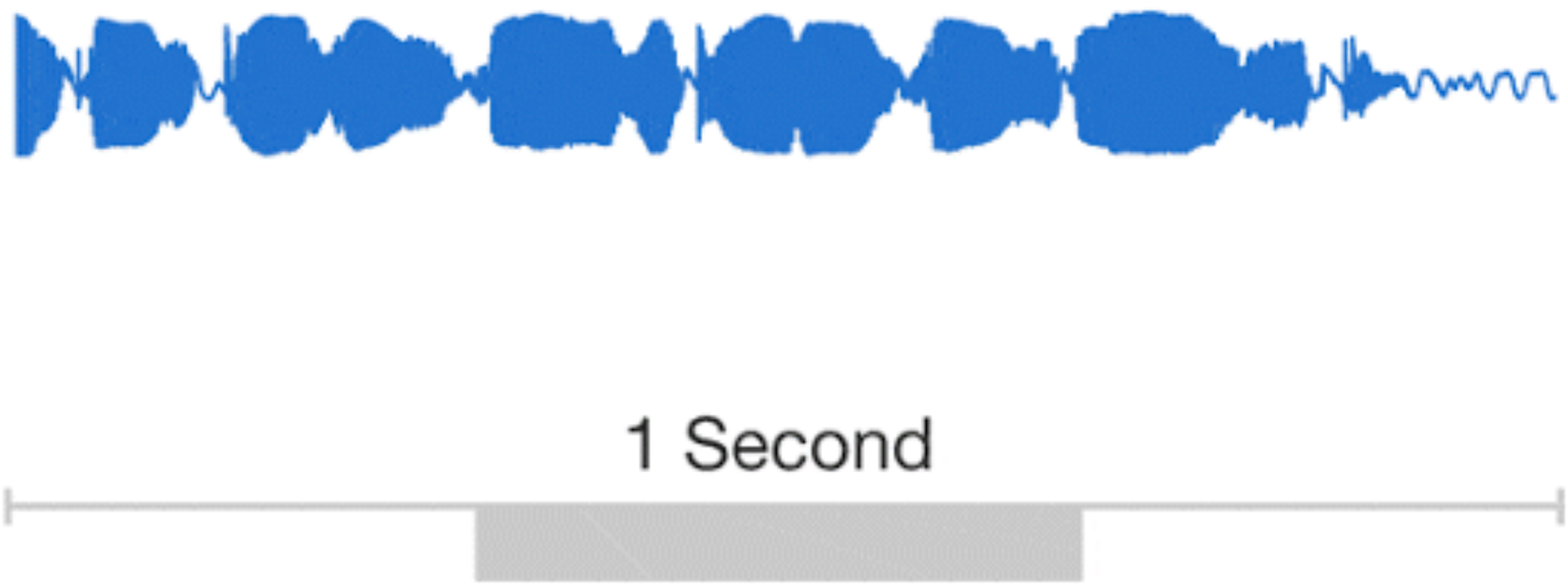


[Rezende and Mohamed, 2016; Dinh et al., 2016; Kingma et al., 2016]

Linear time computation of the determinant and its gradient.

WAVENET: A GENERATIVE MODEL FOR RAW AUDIO





Approaches to inference that aren’t based on approximating the conditional density p(x|θ)

THE PLAYERS

forward modeling
generation
simulation

PREDICTION

θ
parameters of interest

$p(x, z \mid \theta, \nu)$

z
latent variables
Monte Carlo truth

x
observed data
simulated data

ν
nuisance parameters

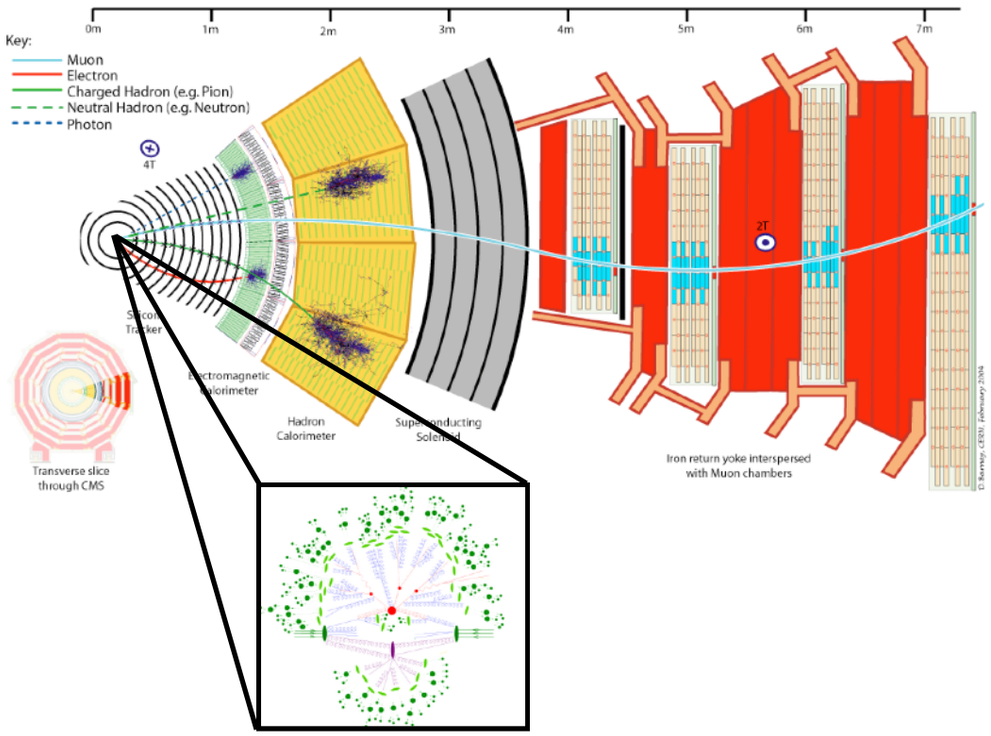
INFERENCE

inverse problem
measurement
parameter estimation

TWO APPROACHES

Use simulator

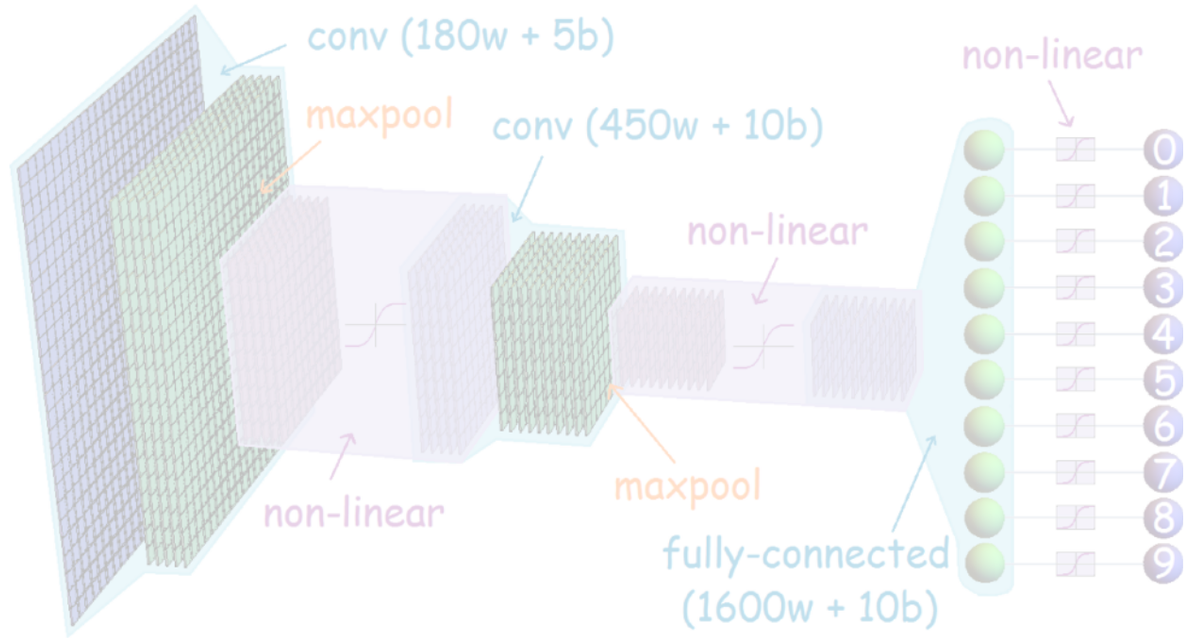
(much more efficiently)



- Approximate Bayesian Computation (ABC)
- Probabilistic Programming
- Adversarial Variational Optimization (AVO)

Learn simulator

(with deep learning)



- Generative Adversarial Networks (GANs), Variational Auto-Encoders (VAE)
- Likelihood ratio from classifiers (CARL)
- Autogregressive models, Normalizing Flows

[image credit: A.P. Goucher]

‘Likelihood-Free’ Inference ← exact Bayesian Computation

Rejection Algorithm

- Draw θ from prior $\pi(\cdot)$
- Accept θ with probability $\pi(D \mid \theta)$

Accepted θ are independent draws from the posterior distribution, $\pi(\theta \mid D)$.

If the likelihood, $\pi(D|\theta)$, is unknown:

‘Mechanical’ Rejection Algorithm

- Draw θ from $\pi(\cdot)$
- Simulate $X \sim f(\theta)$ from the computer model
- Accept θ if $D = X$, i.e., if computer output equals observation

The acceptance rate is $\int \mathbb{P}(D|\theta)\pi(\theta)\mathrm{d}\theta = \mathbb{P}(D)$.

*From Richard Wilkinson’s talk at Data science @LHC

Rejection ABC

If $\mathbb{P}(D)$ is small (or D continuous), we will rarely accept any θ. Instead, there is an approximate version:

Uniform Rejection Algorithm

- Draw θ from $\pi(\theta)$
- Simulate $X \sim f(\theta)$
- Accept θ if $\rho(D, X) \leq \epsilon$

ϵ reflects the tension between computability and accuracy.

- As $\epsilon \to \infty$, we get observations from the prior, $\pi(\theta)$.
- If $\epsilon = 0$, we generate observations from $\pi(\theta \mid D)$.

For reasons that will become clear later, we call this *uniform-ABC*.

*From Richard Wilkinson's talk at Data science @LHC

NEW! AVO

G. Louppe & K.C. arXiv:1707.07113

Adversarial Variational Optimization of Non-Differentiable Simulators

Gilles Louppe[1] and **Kyle Cranmer**[1]
[1]*New York University*

Complex computer simulators are increasingly used across fields of science as generative models tying parameters of an underlying theory to experimental observations. Inference in this setup is often difficult, as simulators rarely admit a tractable density or likelihood function. We introduce Adversarial Variational Optimization (AVO), a likelihood-free inference algorithm for fitting a non-differentiable generative model incorporating ideas from empirical Bayes and variational inference. We adapt the training procedure of generative adversarial networks by replacing the differentiable generative network with a domain-specific simulator. We solve the resulting non-differentiable minimax problem by minimizing variational upper bounds of the two adversarial objectives. Effectively, the procedure results in learning a proposal distribution over simulator parameters, such that the corresponding marginal distribution of the generated data matches the observations. We present results of the method with simulators producing both discrete and continuous data.

Leo is *G* Tom is *D*

Similar to GAN setup, but instead of using a neural network as the generator, use the actual simulation (eg. Pythia, GEANT)

Continue to use a neural network discriminator / critic.

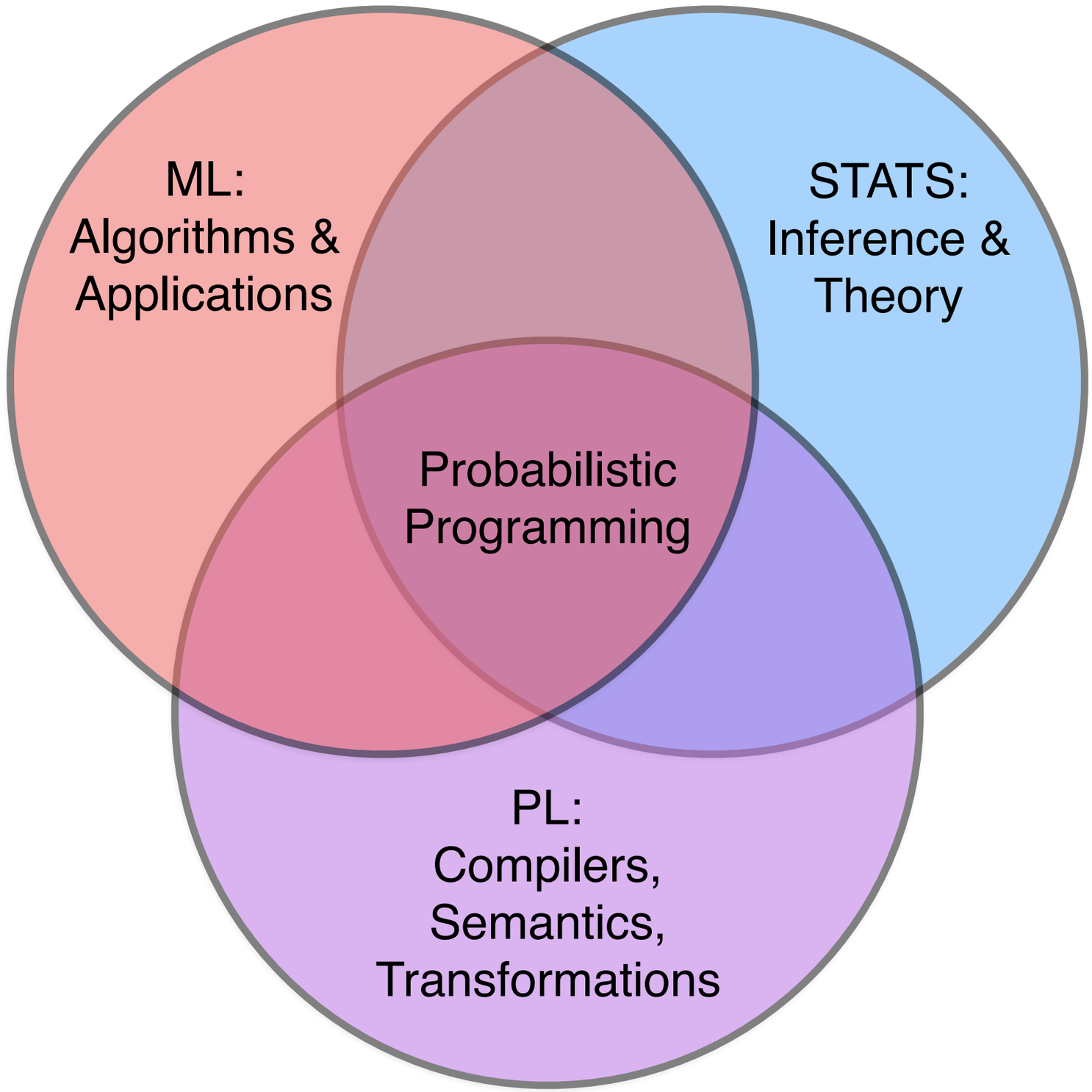
Difficulty: the simulator isn't differentiable, but there's a **trick**!

Allows us to efficiently fit / **tune simulation** with stochastic gradient techniques!

Probabilistic Programming:
Inverting the simulation

(very ambitious)

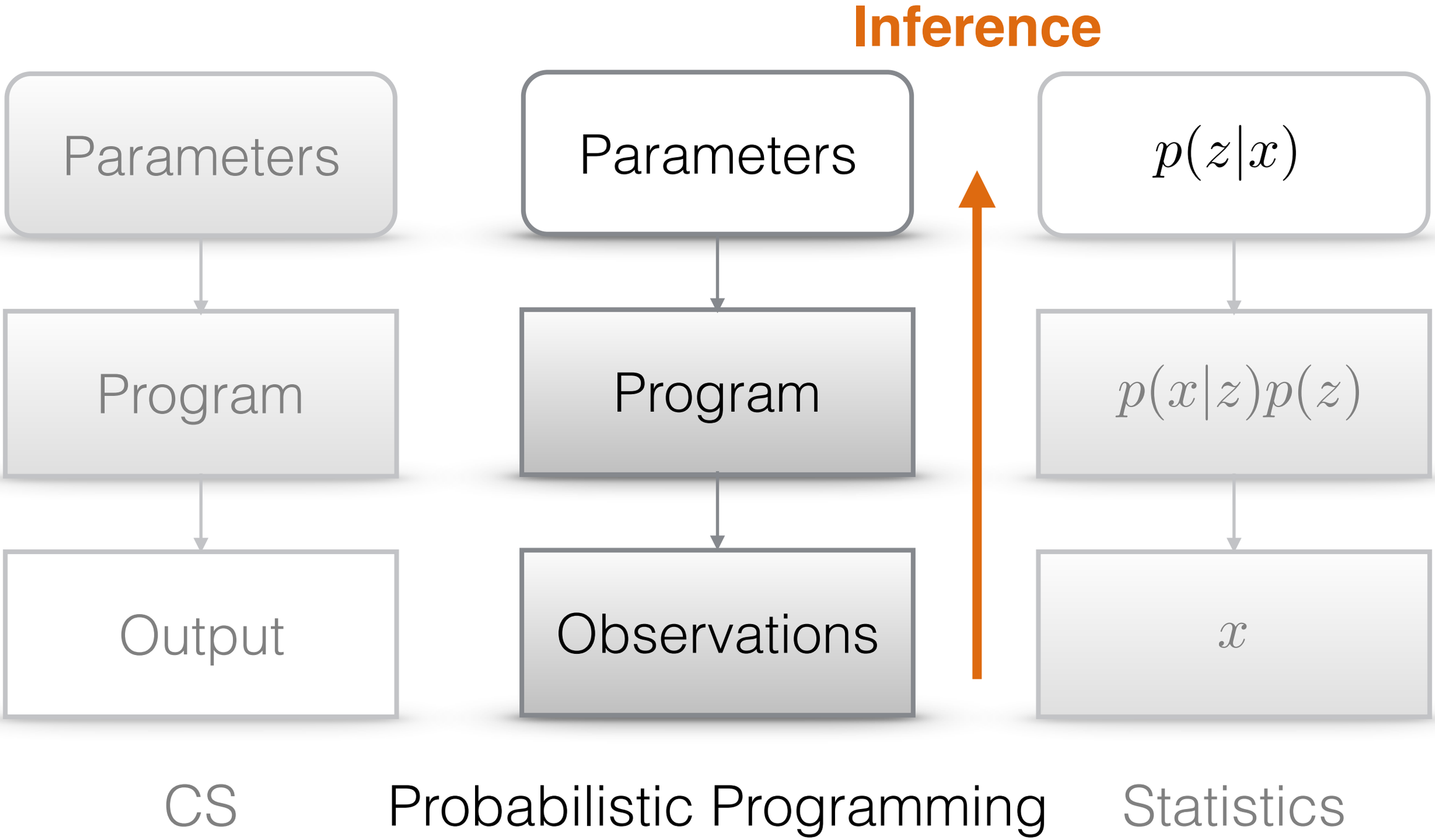
Probabilistic Programming



Intuition

[slides, Frank Wood]

Intuition



[slides, Frank Wood]

CAPTCHA breaking

Observation

Posterior Samples

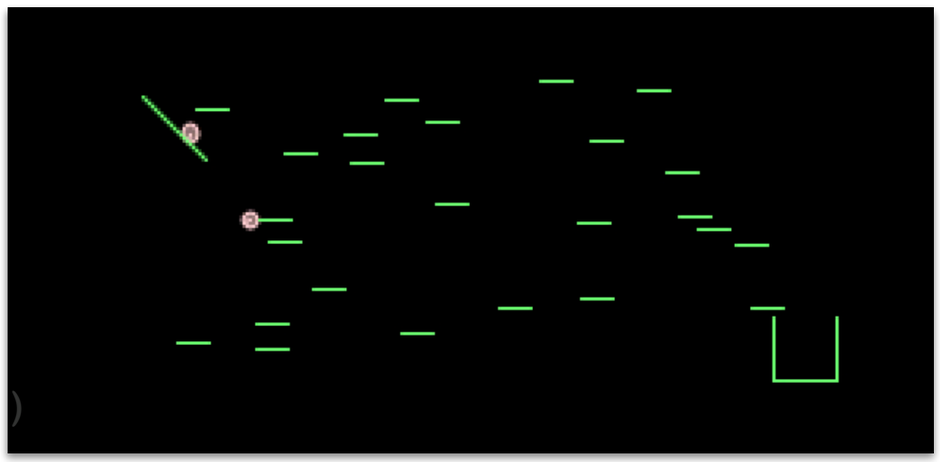
Generative Model

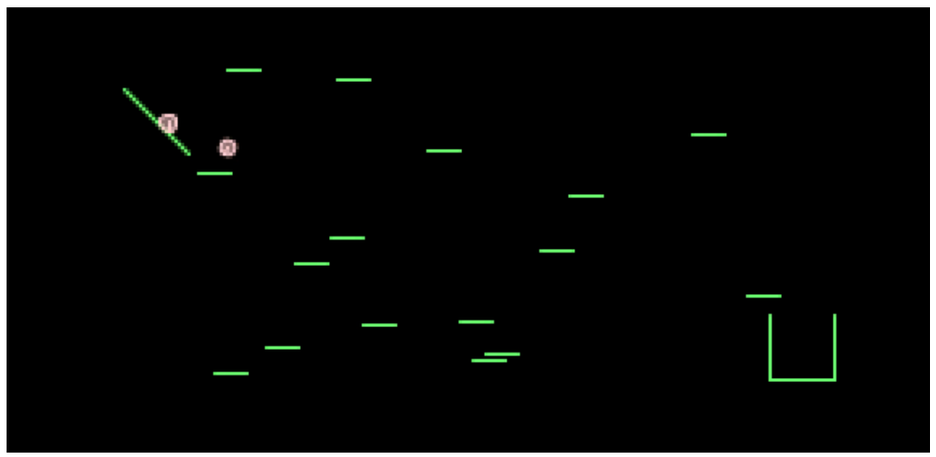
```
(defquery captcha
 [image num-chars tol]
 (let [[w h] (size image)
       ;; sample random characters
       num-chars (sample
                  (poisson num-chars))
       chars (repeatedly
              num-chars sample-char)]
  ;; compare rendering to true image
  (map (fn [y z]
         (observe (normal z tol) y))
       (reduce-dim image)
       (reduce-dim (render chars w h)))
 ;; predict captcha text
 {:text
  (map :symbol (sort-by :x chars))}))
```

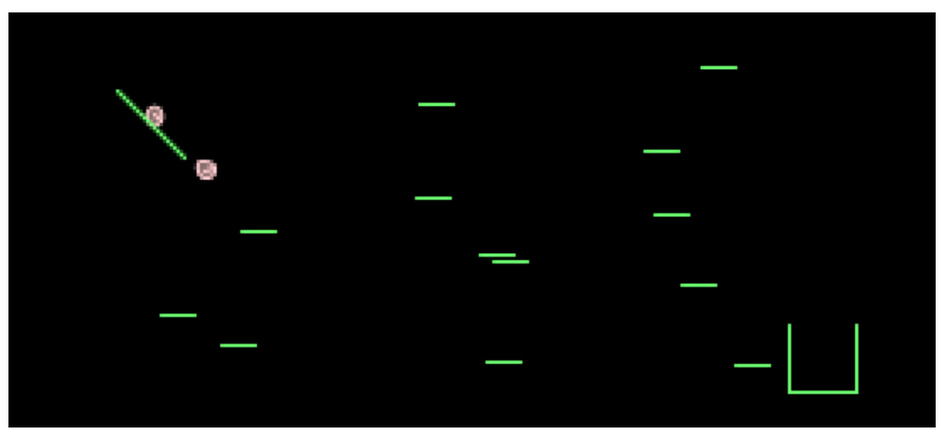
Mansinghka,, Kulkarni, Perov, and Tenenbaum
“Approximate Bayesian image interpretation using generative probabilistic graphics programs." NIPS (2013).

ANALOGY: RANDOM BUMPERS ~ RANDOM CALORIMETER SHOWER

```
(defquery arrange-bumpers []
    (let [number-of-bumpers (sample (poisson 20))
          bumpydist (uniform-continuous 0 10)
          bumpxdist (uniform-continuous -5 14)
          bumper-positions (repeatedly
                            number-of-bumpers
                            #(vector (sample bumpxdist)
                                     (sample bumpydist)))

          ;; code to simulate the world
          world (create-world bumper-positions)
          end-world (simulate-world world)
          balls (:balls end-world)

          ;; how many balls entered the box?
          num-balls-in-box (balls-in-box end-world)]

      {:balls balls
       :num-balls-in-box num-balls-in-box
       :bumper-positions bumper-positions}))
```





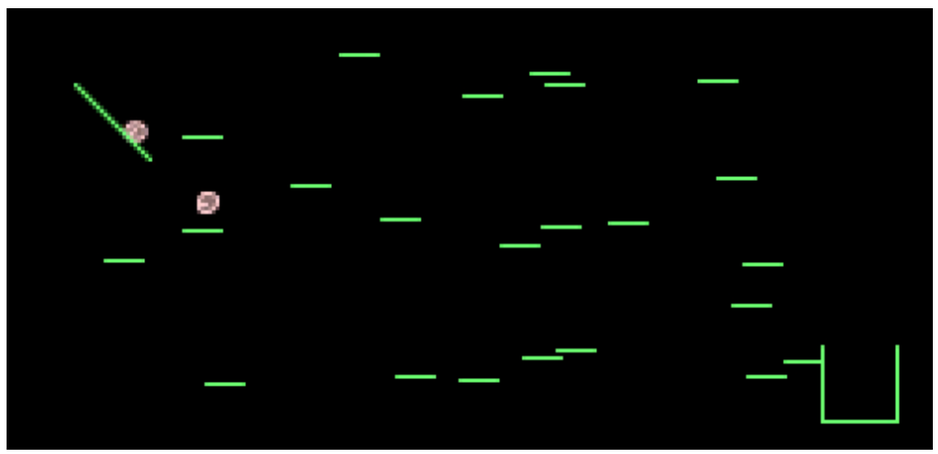


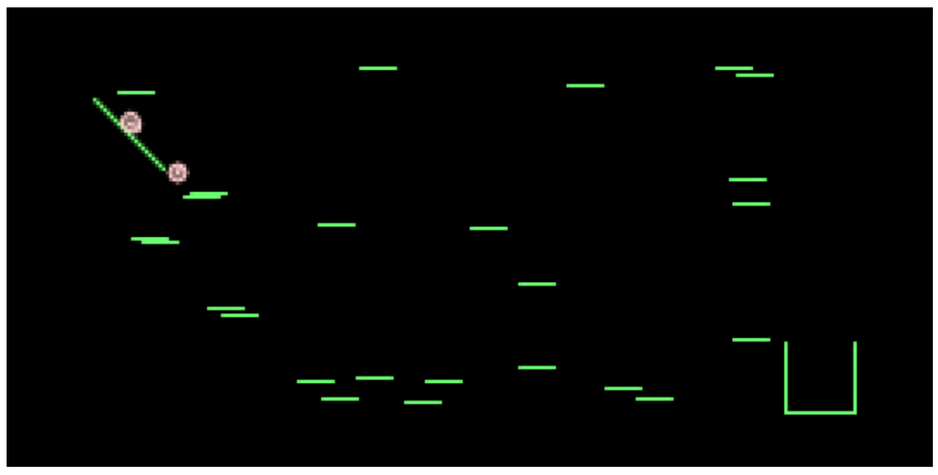
3 examples generated from simulator

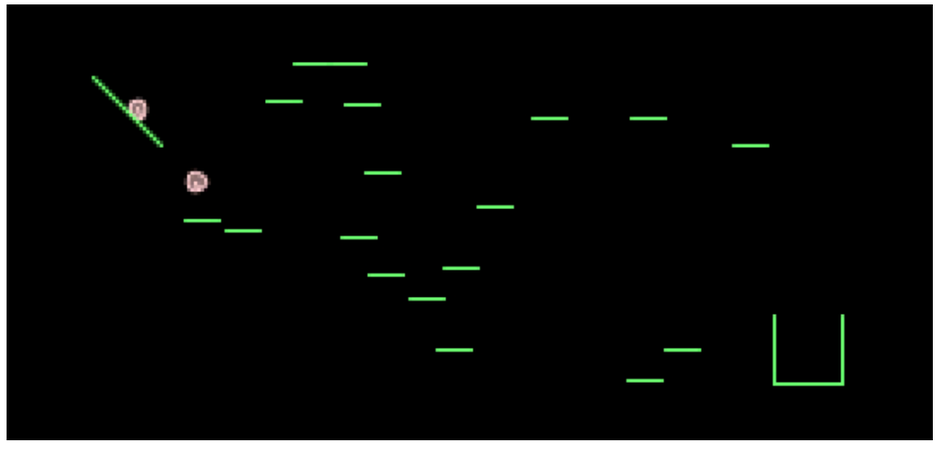
[slides, Frank Wood]

UNDERSTANDING THE TAILS OF DISTRIBUTIONS

```
(defquery arrange-bumpers []
    (let [number-of-bumpers (sample (poisson 20))
          bumpydist (uniform-continuous 0 10)
          bumpxdist (uniform-continuous -5 14)
          bumper-positions (repeatedly
                            number-of-bumpers
                            #(vector (sample bumpxdist)
                                     (sample bumpydist)))

          ;; code to simulate the world
          world (create-world bumper-positions)
          end-world (simulate-world world)
          balls (:balls end-world)

          ;; how many balls entered the box?
          num-balls-in-box (balls-in-box end-world)

          obs-dist (normal 4 0.1)]

    (observe obs-dist num-balls-in-box)
```







3 examples generated from simulator
conditioned on ~20% of balls land in box
(~ given observed energy deposits)

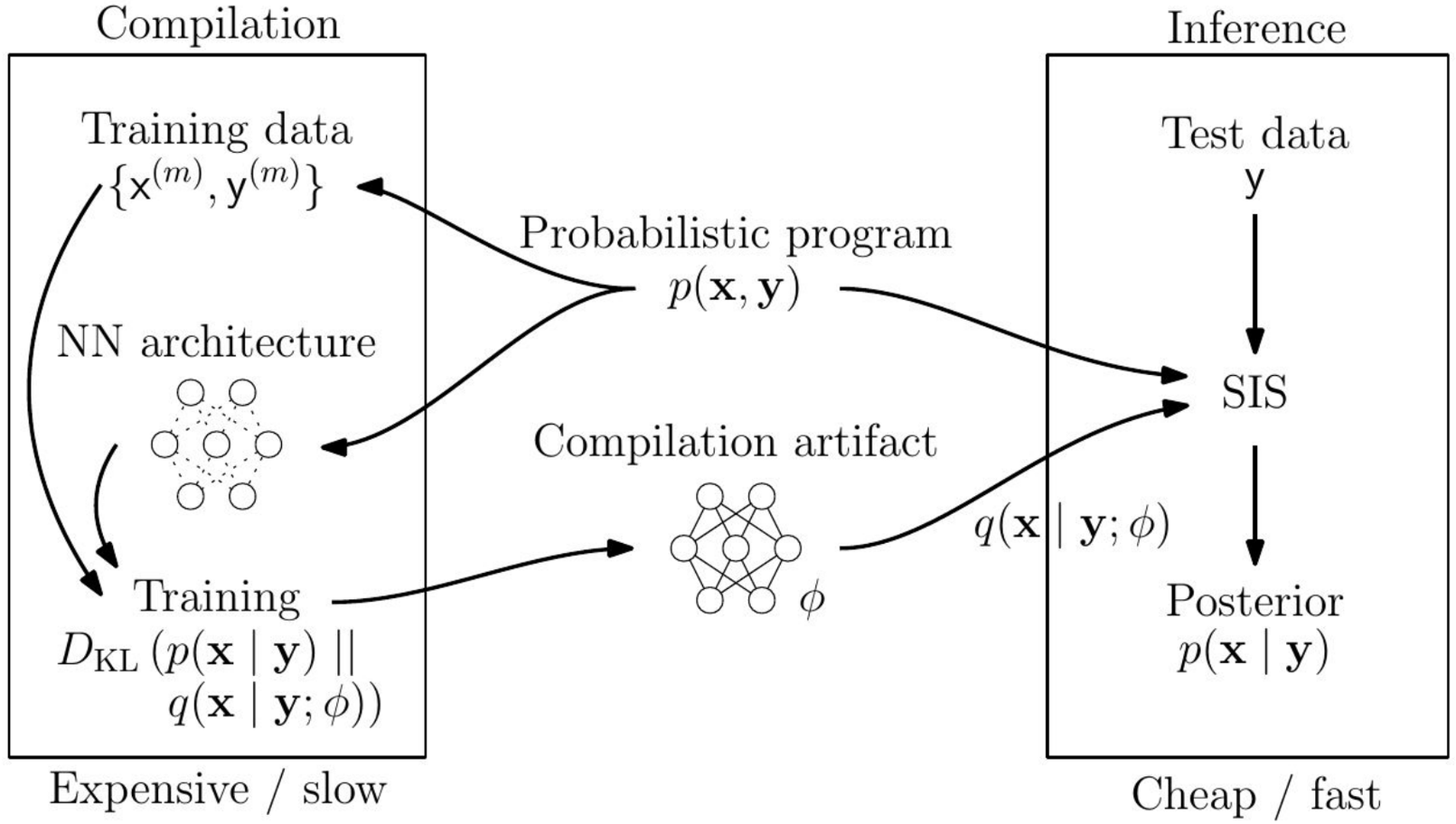
[slides, Frank Wood]

HOW DOES IT WORK?

In short: hijack the random number generators and use NN's to perform a *very* smart type of importance sampling

Input: an inference problem denoted in a universal PPL (Anglican, CPProb)

Output: a trained inference network, or "compilation artifact" (Torch, PyTorch)

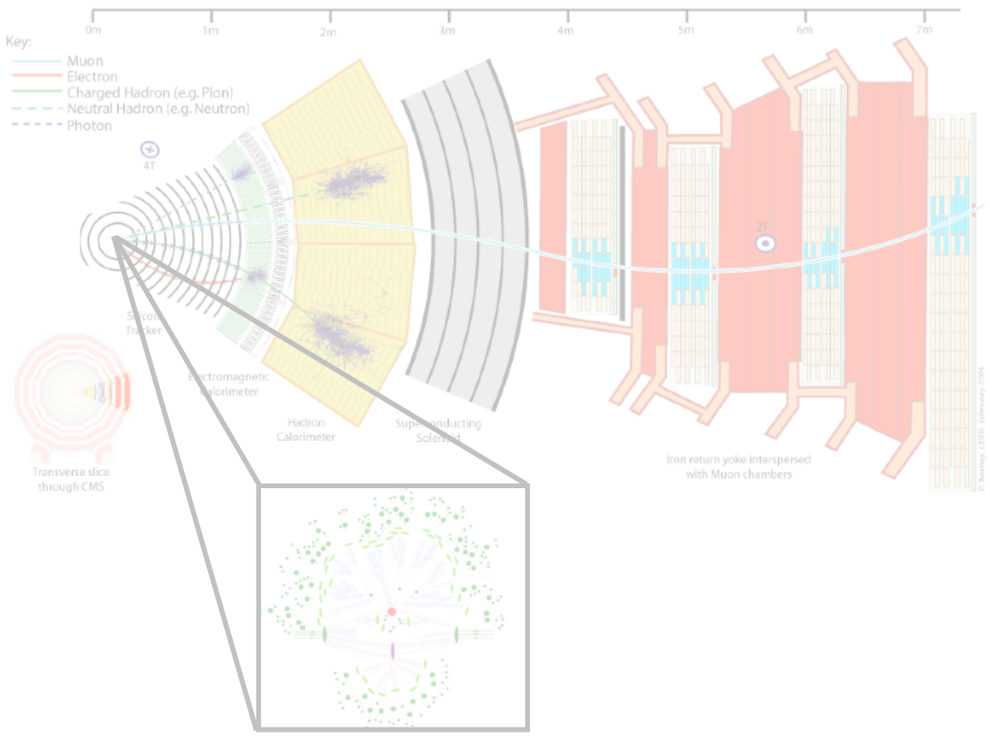


Le, Baydin and Wood. Inference Compilation and Universal Probabilistic Programming. AISTATS 2017. *arXiv:1610.09900*

TWO APPROACHES

Use simulator

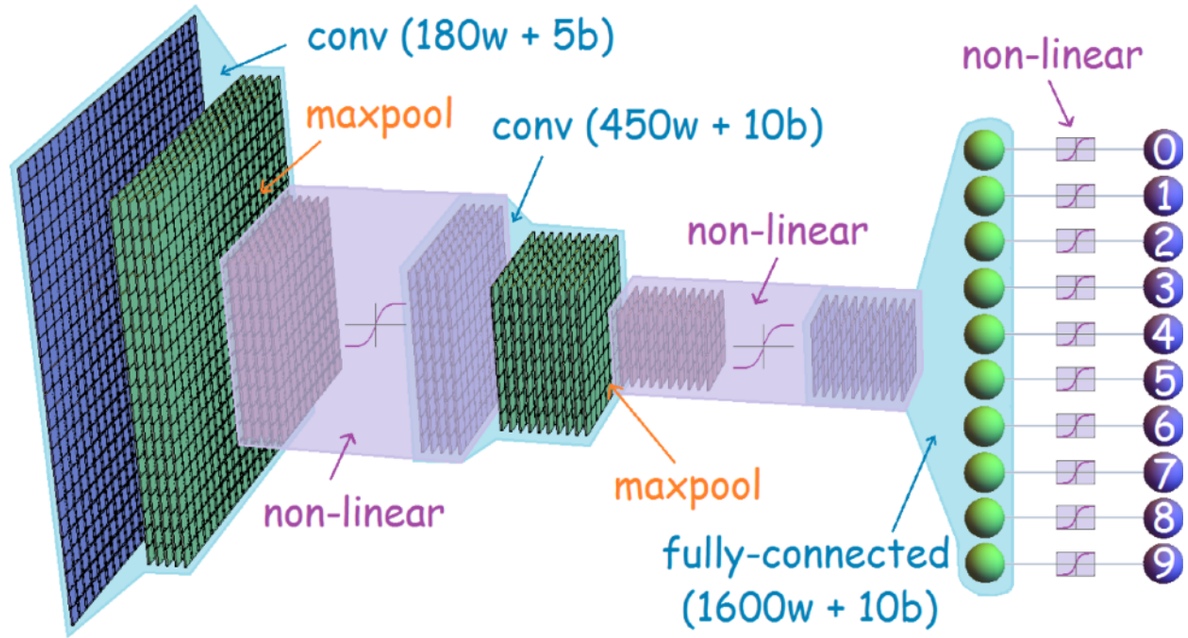
(much more efficiently)



- Approximate Bayesian Computation (ABC)
- Probabilistic Programming
- Adversarial Variational Optimization (AVO)

Learn simulator

(with deep learning)



- Generative Adversarial Networks (GANs), Variational Auto-Encoders (VAE)
- Likelihood ratio from classifiers (CARL)
- Autogregressive models, Normalizing Flows

[image credit: A.P. Goucher]

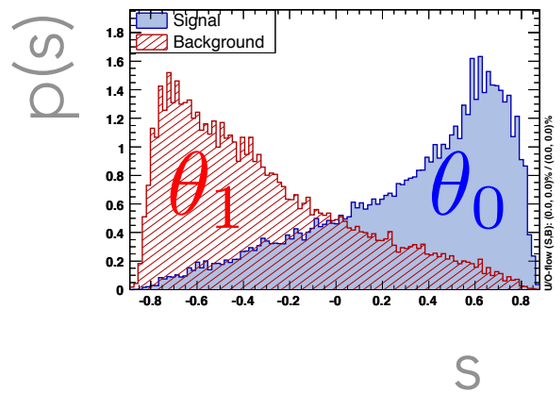
CARL

The intractable likelihood ratio based on high-dimensional features x is:

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

We can show that an **equivalent test** can be made from 1-D projection

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



if the scalar map s: X → $\mathbb{R}$ has the same level sets as the likelihood ratio

$$s(x;\theta_0;\theta_1) = \text{monotonic}[\ p(x|\theta_0)/p(x|\theta_1)\ ]$$

Estimating the density of $s(x;\theta_0,\theta_1)$ via the simulator calibrates the ratio.

K.C., G. Louppe, J. Pavez: http://arxiv.org/abs/1506.02169

Binary classifier on balanced y=0 and y=1 labels learns

$$s(x) = \frac{p(x|y=1)}{p(x|y=0) + p(x|y=1)}$$

Which is one-to-one with the likelihood ratio

$$\frac{p(x|y=0)}{p(x|y=1)} = 1 - \frac{1}{s(x)}$$

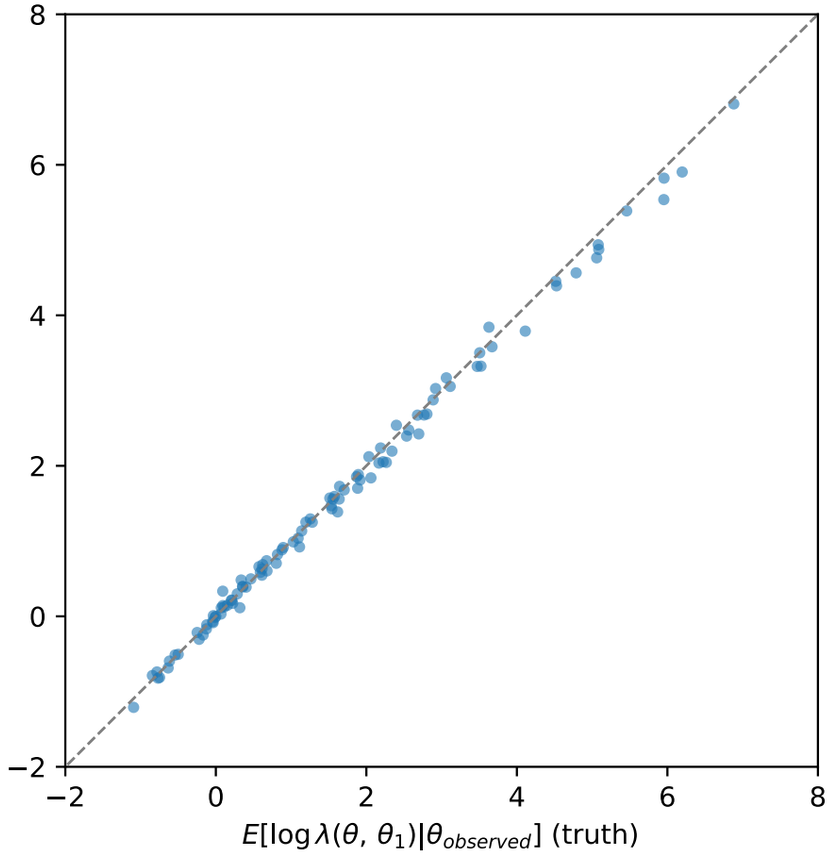
Can do the same thing for any two points θ_0 & θ_1 in parameter space. I call this a **parametrized classifier**

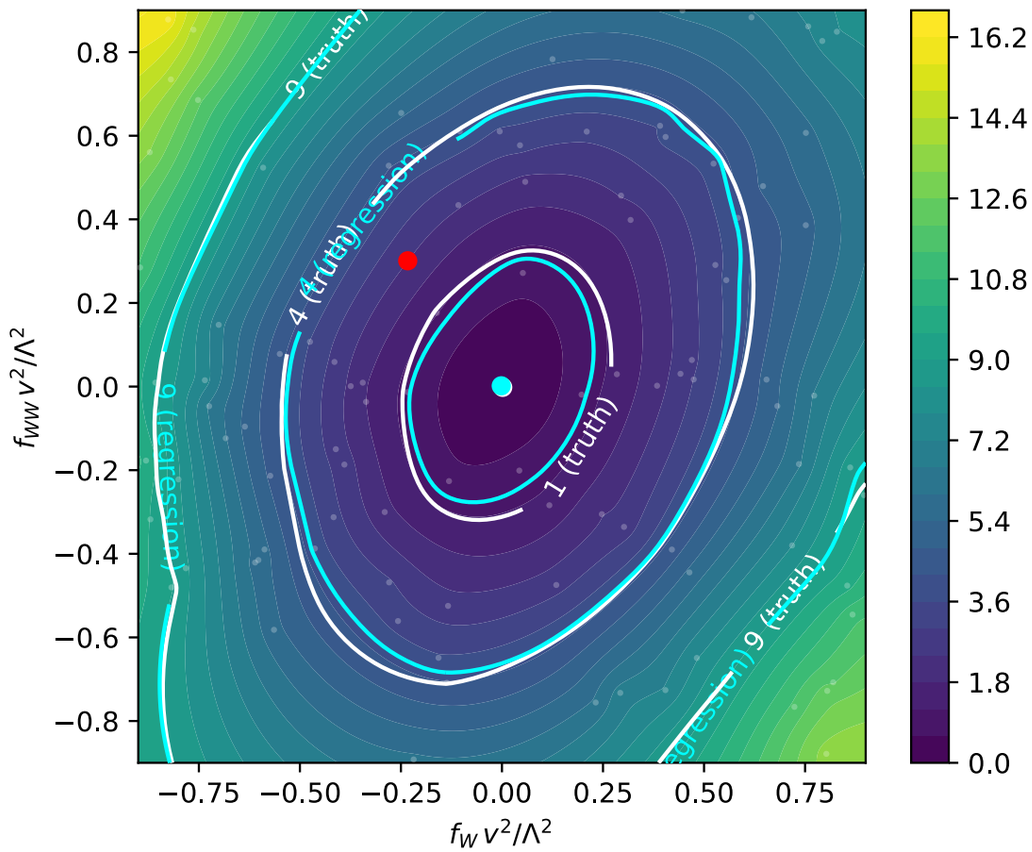
$$s(x;\theta_0,\theta_1) = \frac{p(x|\theta_1)}{p(x|\theta_0) + p(x|\theta_1)}$$

K.C., G. Louppe, J. Pavez: http://arxiv.org/abs/1506.02169

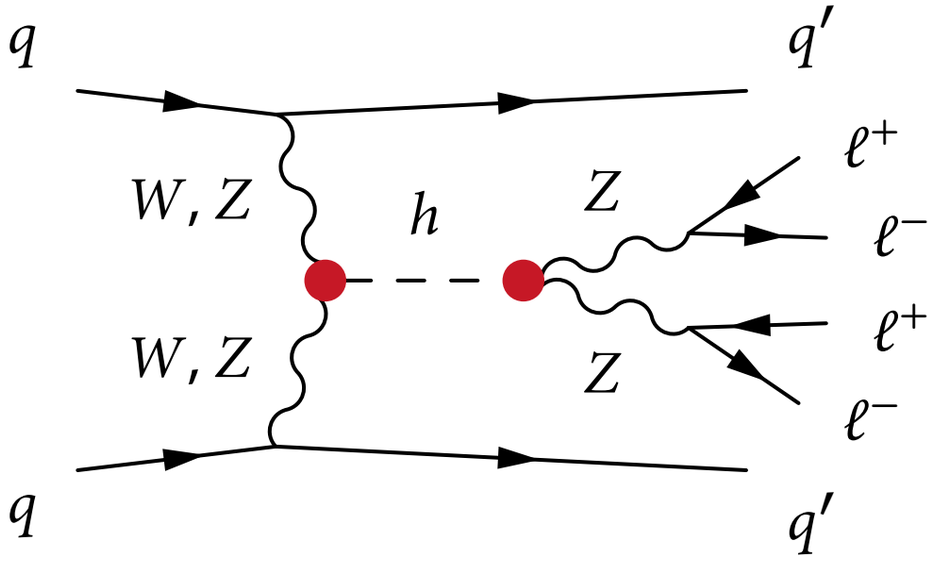
LEARNING A 16 DIM LIKELIHOOD

Estimated likelihood



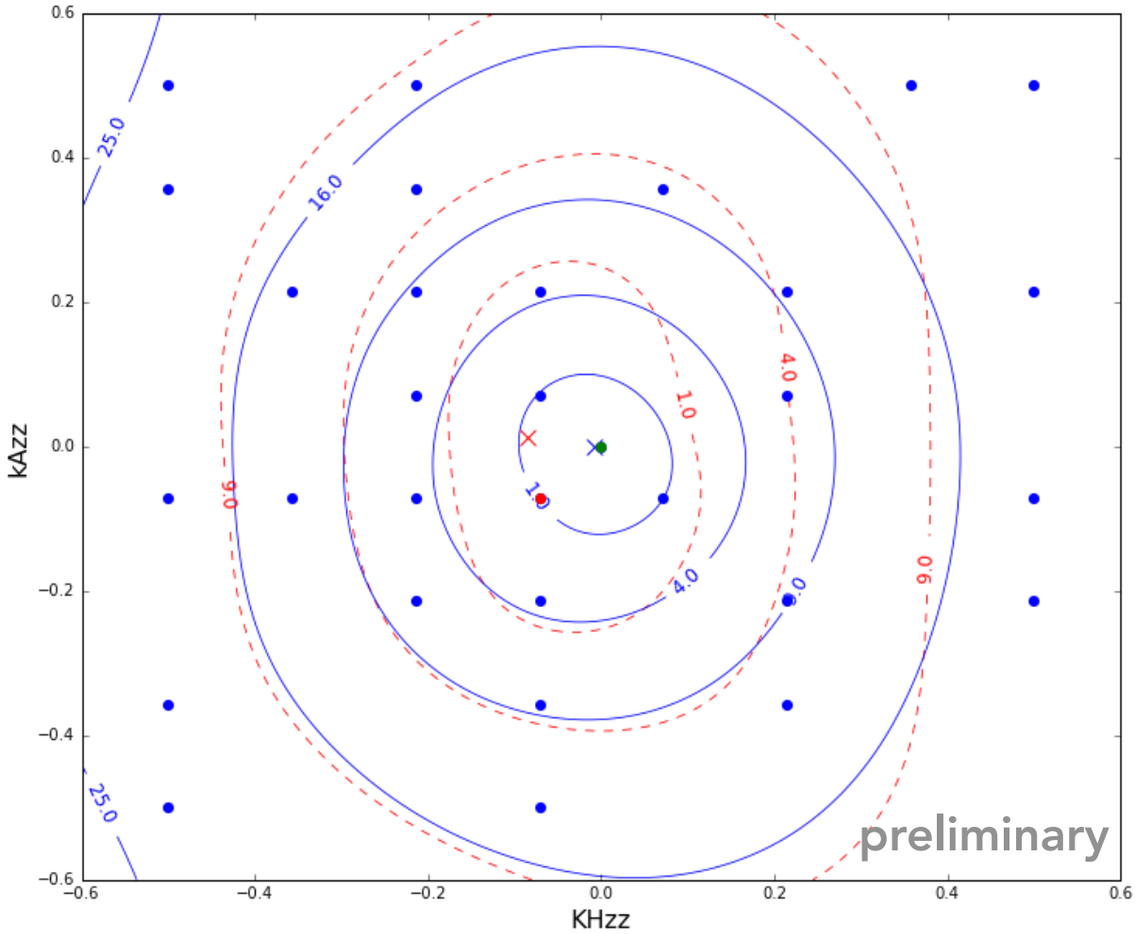


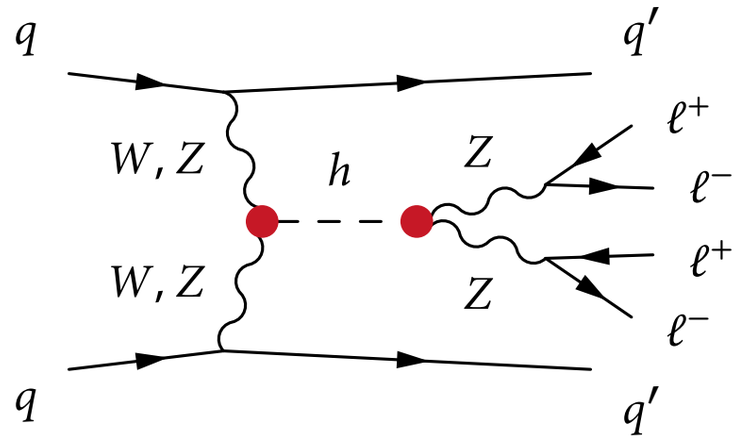
True likelihood



APPLICATION TO THE HIGGS

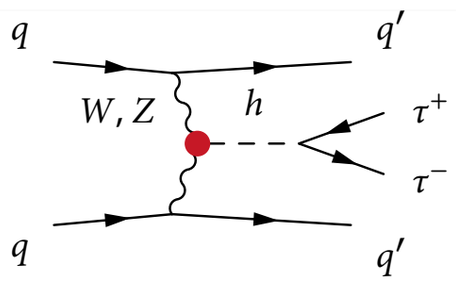
Preliminary work using fast detector simulation and CARL to approximate likelihoods using full kinematic information parametrized in 5-d coefficients of a Quantum Field Theory





16 observables
(using the CARL)

2 observables
(histogram templates)



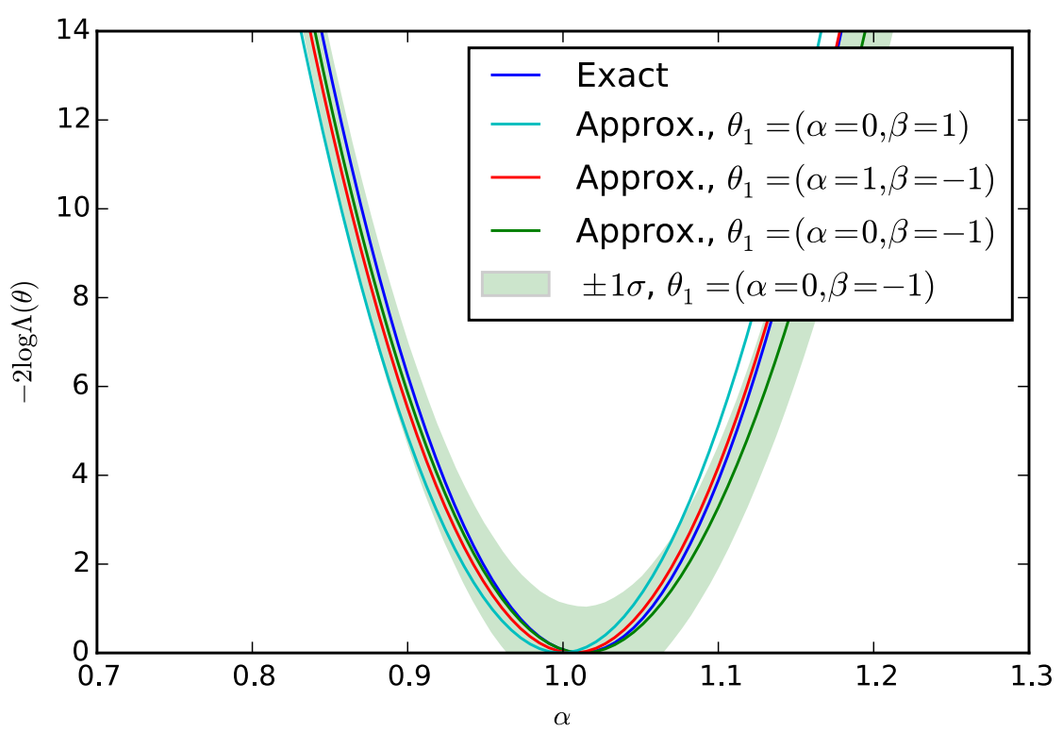
Equivalent to 3x more data.
(idealized, no systematic uncertainty)

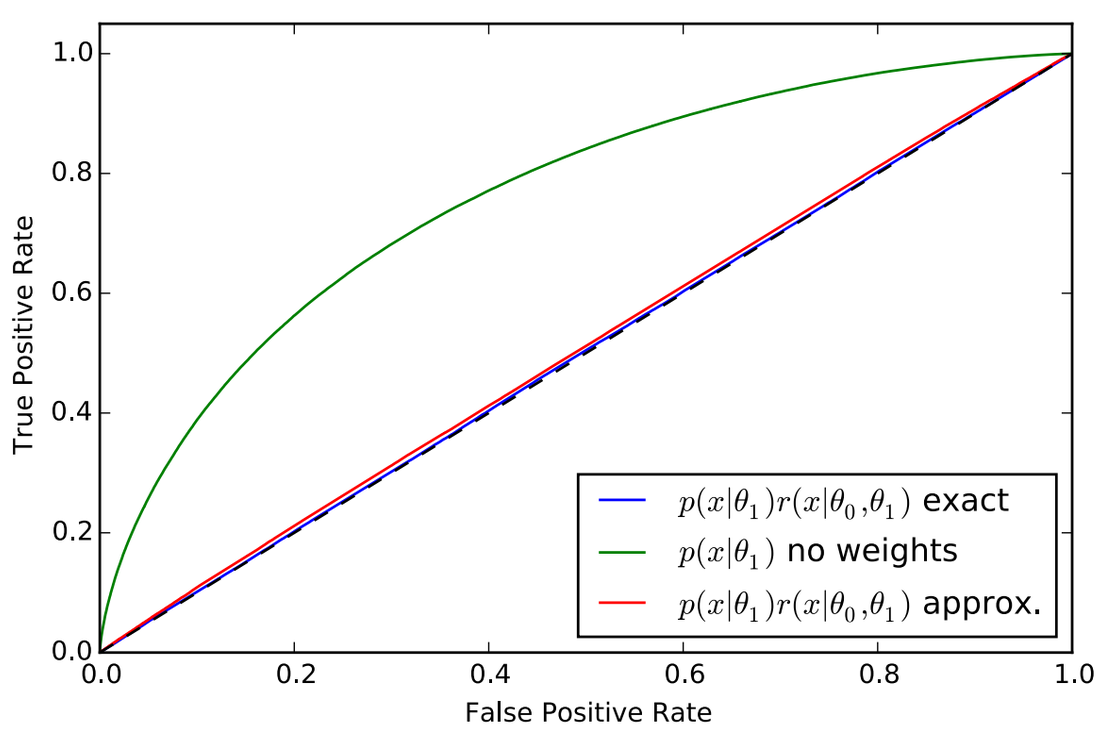
work with Juan Pavez, Gilles Louppe, Cyril Becot, and Lukas Heinrich; Johann Brehmer, Felix Kling, and Tilman Plehn
"Better Higgs Measurements Through Information Geometry" [arXiv:1612.05261]

DIAGNOSTICS

In practice $\hat{r}(\hat{s}(\mathbf{x};\theta_0,\theta_1))$ will not be exact. Diagnostic procedures are needed to assess the quality of this approximation.

1. For inference, the value of the MLE $\hat{\theta}$ should be independent of the value of θ_1 used in the denominator of the ratio.
2. Train a classifier to distinguish between unweighted samples from $p(\mathbf{x}|\theta_0)$ and samples from $p(\mathbf{x}|\theta_1)$ weighted by $\hat{r}(\hat{s}(\mathbf{x};\theta_0,\theta_1))$.



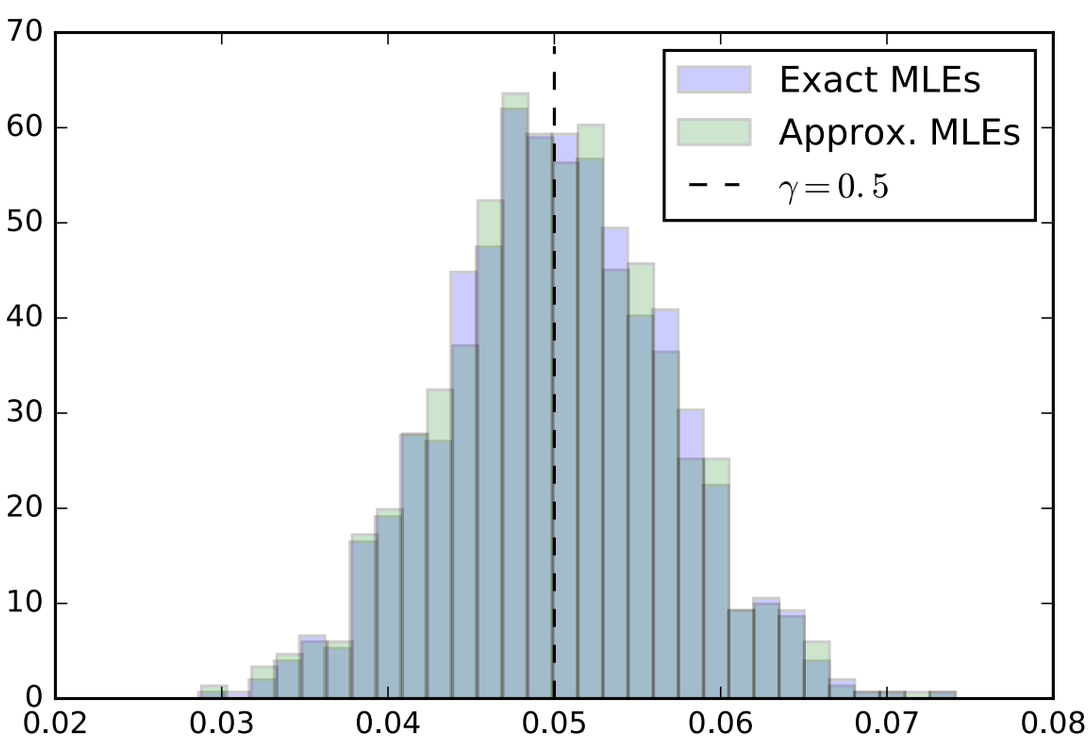


$$\frac{p_1(s^*)}{p_0(s^*)} = \frac{p_1(x)}{p_0(x)} \frac{\int d\Omega_{s^*} p_0(x)/|\hat{n}\cdot\nabla s|}{\int d\Omega_{s^*} p_0(x)/|\hat{n}\cdot\nabla s|} = \frac{p_1(x)}{p_0(x)} = r(x)$$

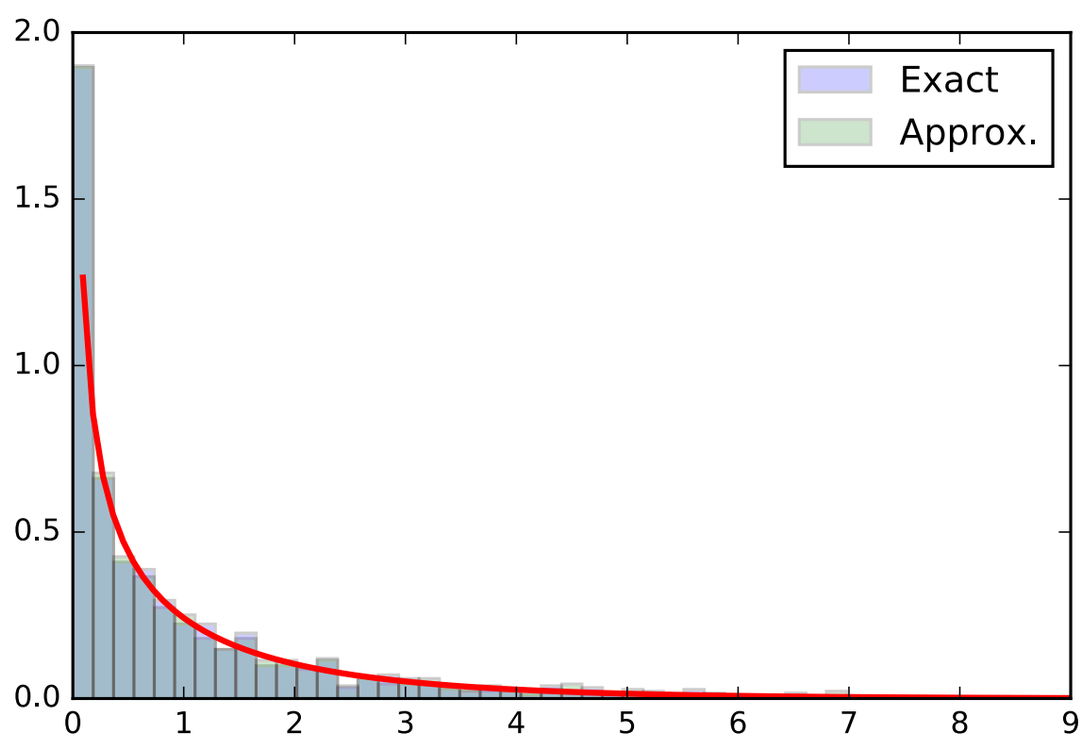
AMORTIZED LIKELIHOOD-FREE INFERENCE

Once we've learned the function s(x; θ) to approximate the likelihood, we can apply it to any data x.

- unlike MCMC, we pay biggest computational costs up front
- Here we repeat inference thousands of times & check asymptotic statistical theory



(a) Exact vs. approximated MLEs.



(b) $p(-2\log\Lambda(\gamma = 0.05)\,|\,\gamma = 0.05)$

K.C., G. Louppe, J. Pavez: http://arxiv.org/abs/1506.02169