NYU CENTER FOR DATA SCIENCE

UCL Université catholique de Louvain

CENTER FOR COSMOLOGY AND PARTICLE PHYSICS

CHAIRE GEORGES LEMAÎTRE 2017

RETHINKING PHYSICS IN THE AGE OF DEEP LEARNING

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

COLLABORATORS


Gilles Louppe
U. Liège


Kyunghyun Cho


Joan Bruna


Brenden Lake


Meghan Frate


Juan Pavez


Tilman Plehn


Johann Brehmer


Isaac Henrion


Lukas Heinrich


Heiko Müller


Tim Head


Michael Kagan


David Rousseau


Peter Sadowski


Daniel Whiteson


Pierre Baldi


Lezcano Casado


Atılım Güneş Baydin
University of Oxford

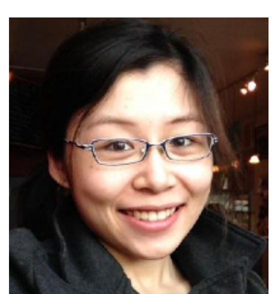

Prabhat
NERSC, Berkeley Lab


Wahid Bhimji
NERSC, Berkeley Lab

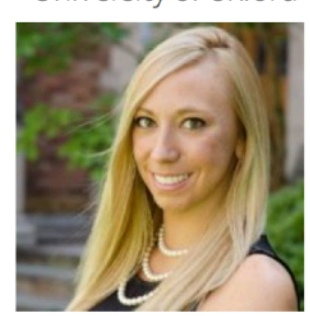

Frank Wood
University of Oxford

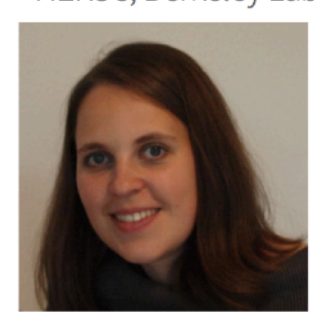

Phiala Shanahan

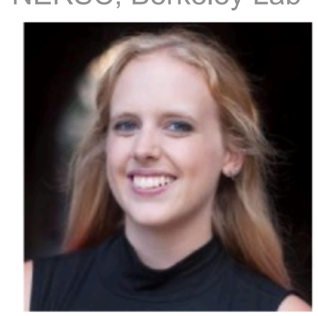

William Detmold

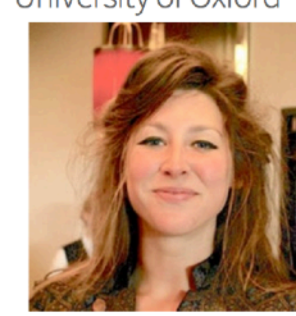

Karen Ng

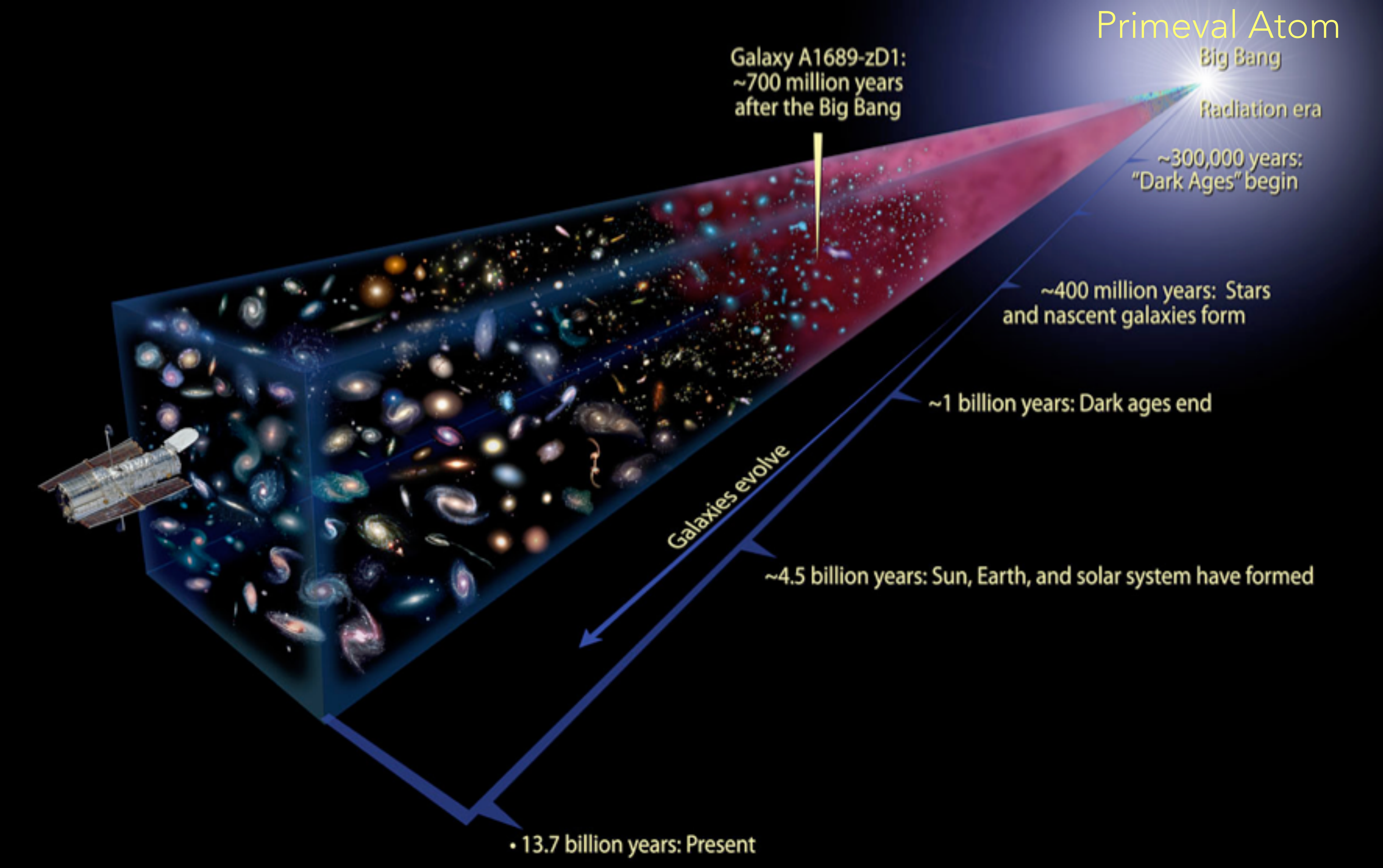

Tuan Anh Le

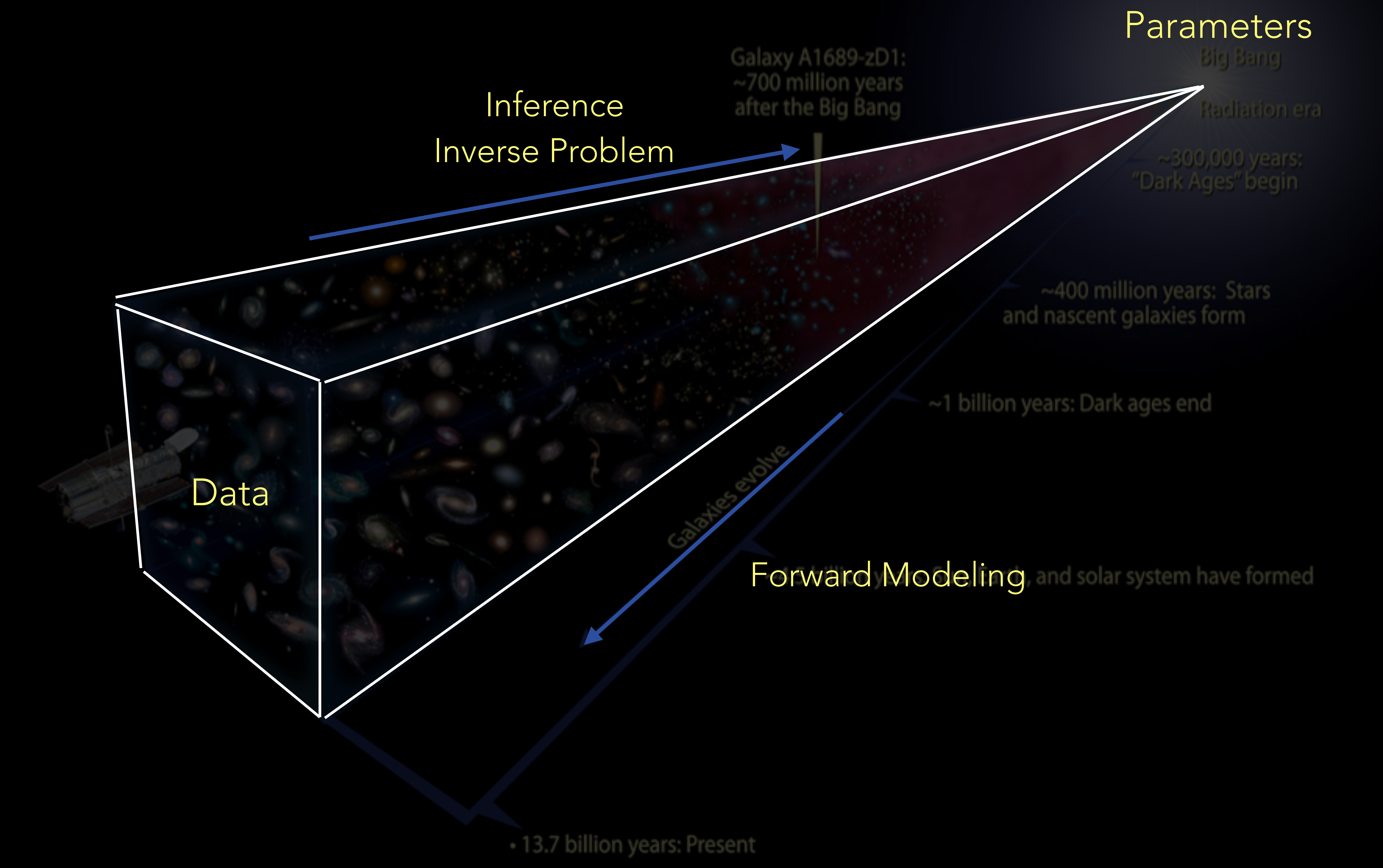

Michela Paganini
Yale University

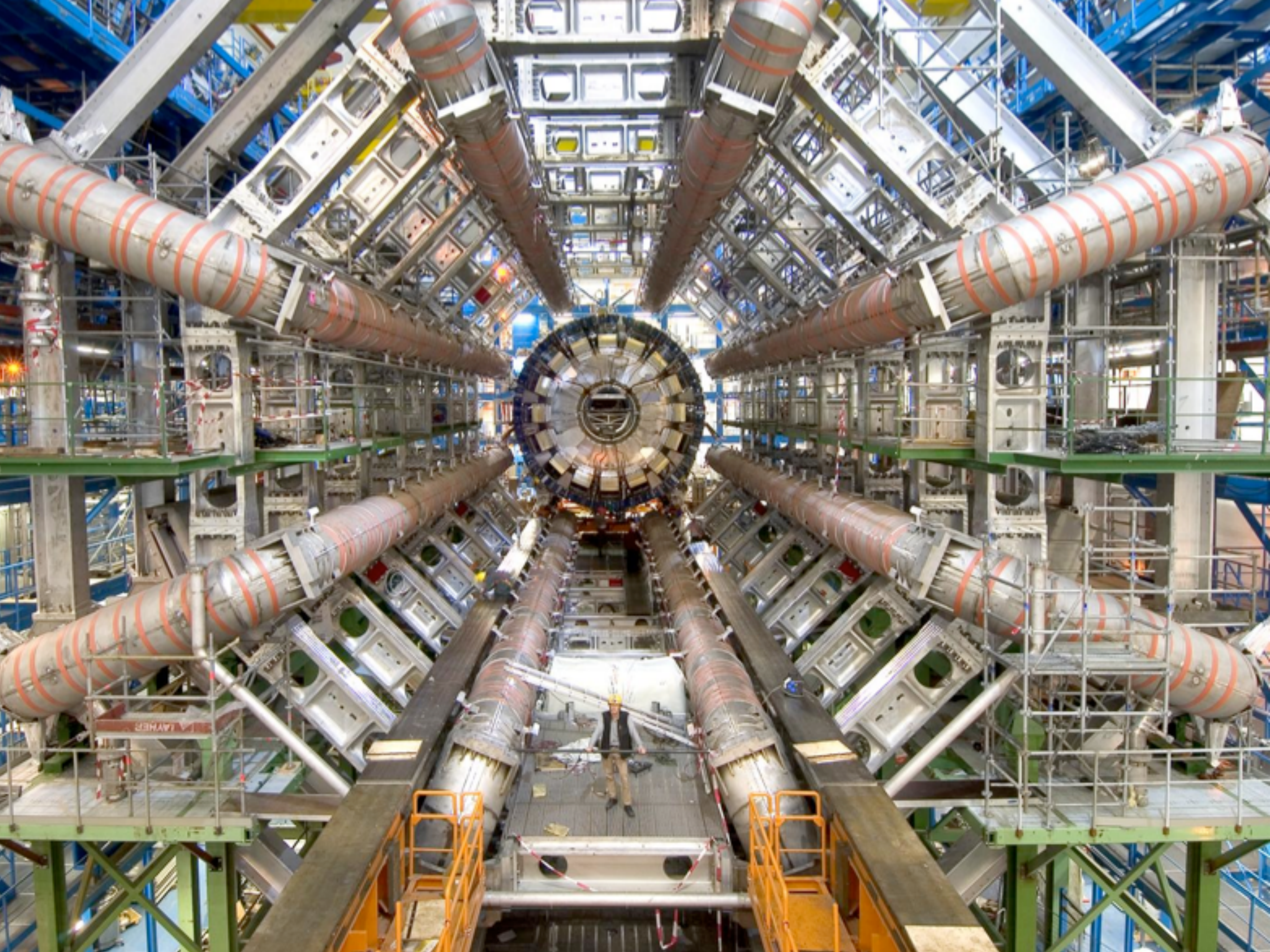

Daniela Huppenkothen
New York University


Savannah Thais
Yale University

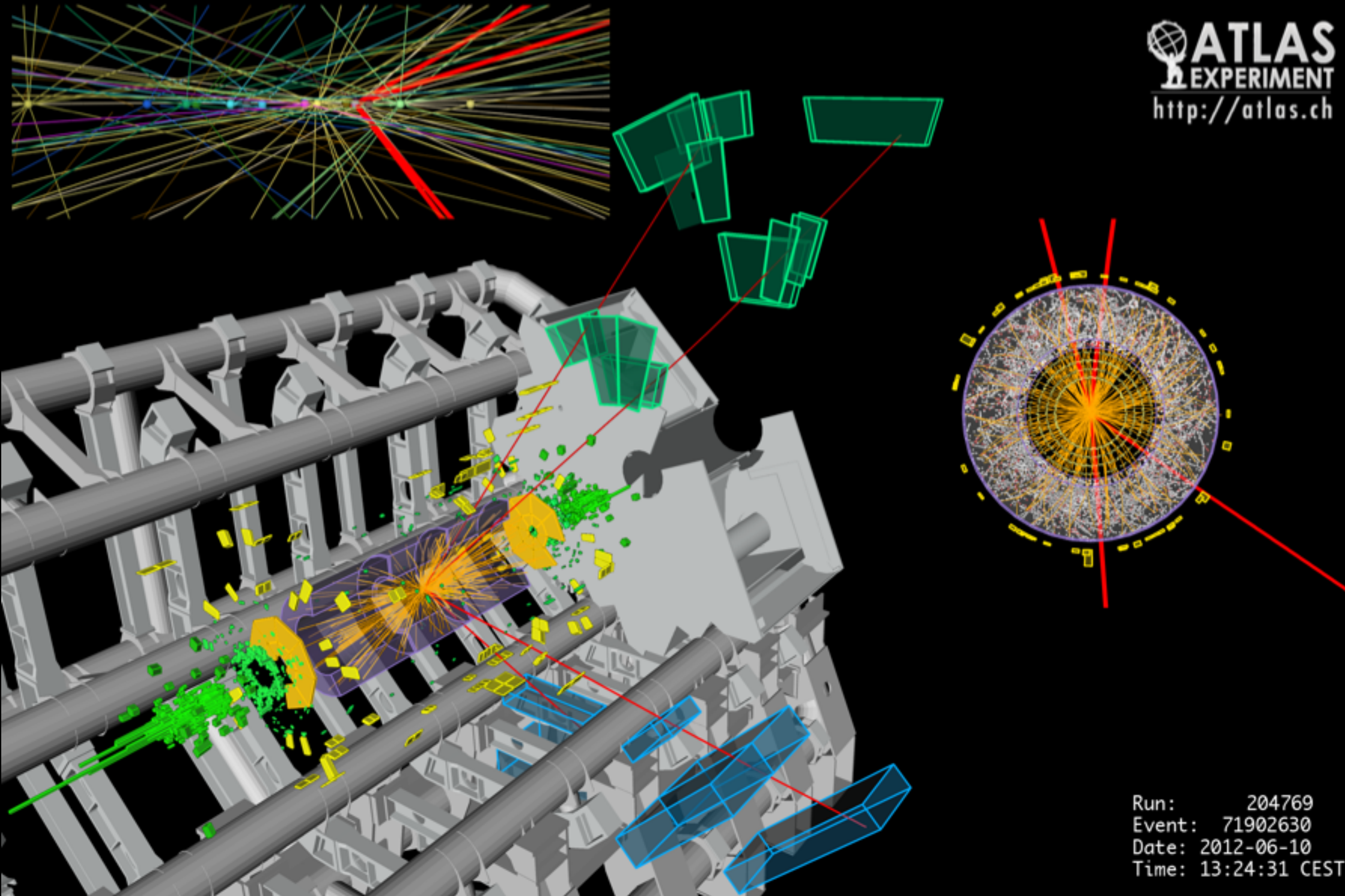

Ruth Angus
Columbia University


Primeval Atom
Big Bang
Radiation era
~300,000 years:
"Dark Ages" begin
Galaxy A1689-zD1:
~700 million years
after the Big Bang
~400 million years: Stars
and nascent galaxies form
~1 billion years: Dark ages end
Galaxies evolve
~4.5 billion years: Sun, Earth, and solar system have formed
• 13.7 billion years: Present

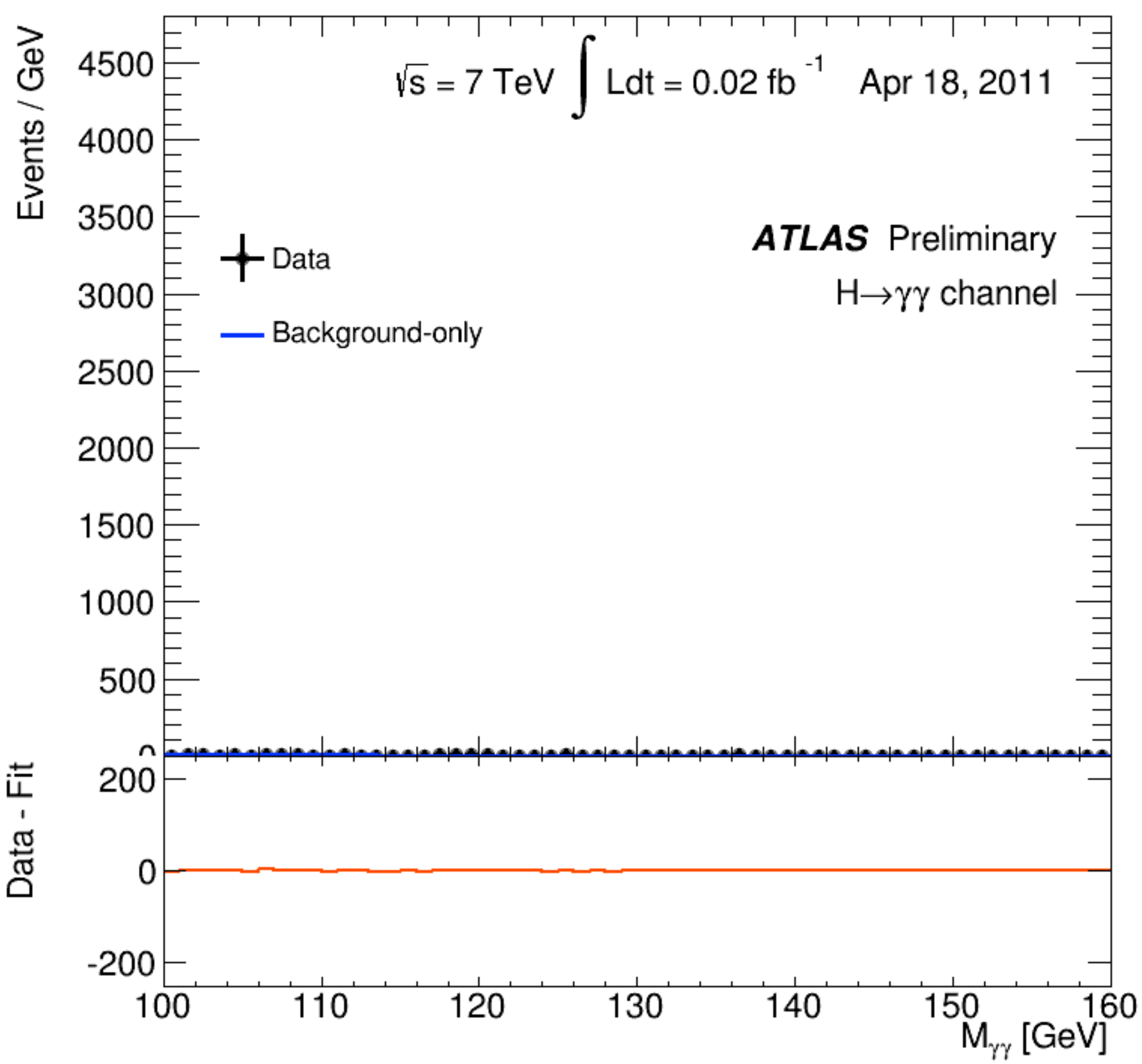

Parameters
Big Bang
Galaxy A1689-zD1:
~700 million years
after the Big Bang
Inference
Inverse Problem
Radiation era
~300,000 years:
"Dark Ages" begin
~400 million years: Stars
and nascent galaxies form
~1 billion years: Dark ages end
Data
Galaxies evolve
Forward Modeling
and solar system have formed
• 13.7 billion years: Present



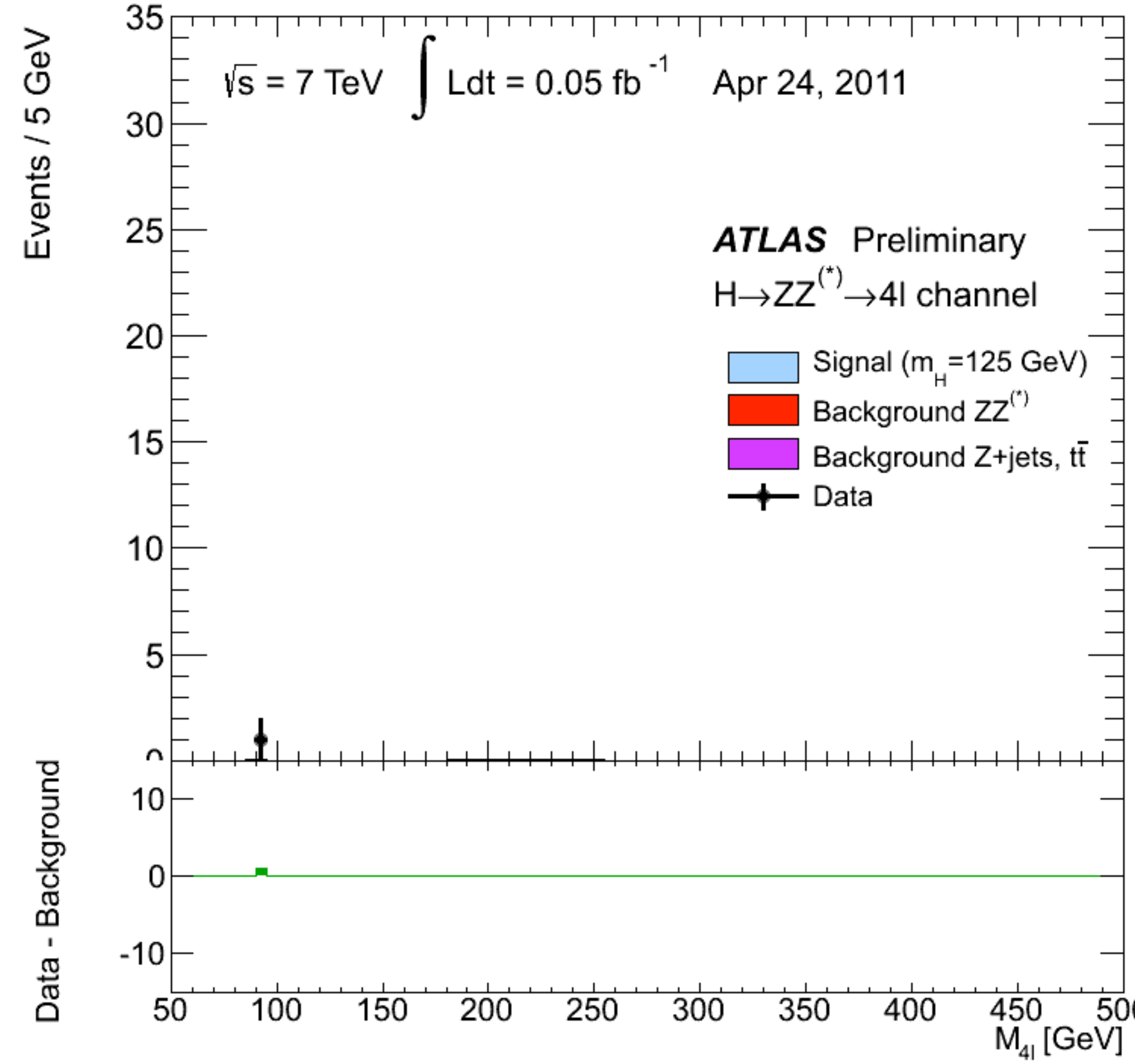
$$H \to ZZ \to 4l$$


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
Mγγ [GeV]



√s = 7 TeV ∫ Ldt = 0.05 fb -1 Apr 24, 2011
ATLAS Preliminary
H→ZZ(*)→4l channel
Signal (mH=125 GeV)
Background ZZ(*)
Background Z+jets, tt̄
Data
Events / 5 GeV
Data - Background
M4l [GeV]

The Nobel Prize in Physics 2013

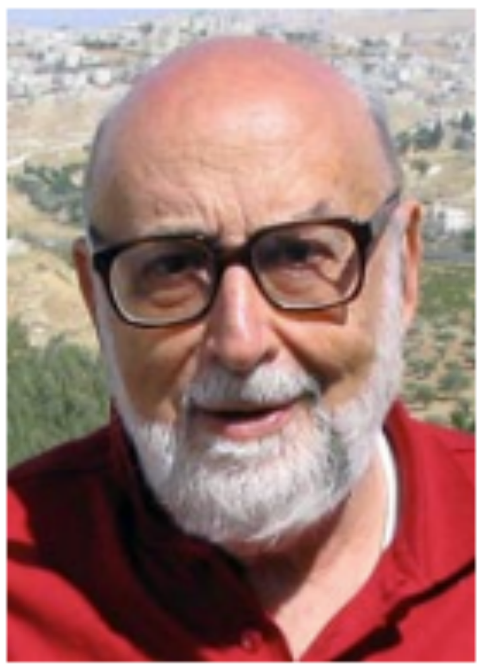


Photo: Pnicolet via Wikimedia Commons

François Englert

Photo: G-M Greuel via Wikimedia Commons

Peter W. Higgs

The Nobel Prize in Physics 2013 was awarded jointly to François Englert and Peter W. Higgs *"for the theoretical discovery of a mechanism that contributes to our understanding of the origin of mass of subatomic particles, and which recently was confirmed through the discovery of the predicted fundamental particle, by the ATLAS and CMS experiments at CERN's Large Hadron Collider"*

Revolution in AI

DEEP LEARNING TIMELINE

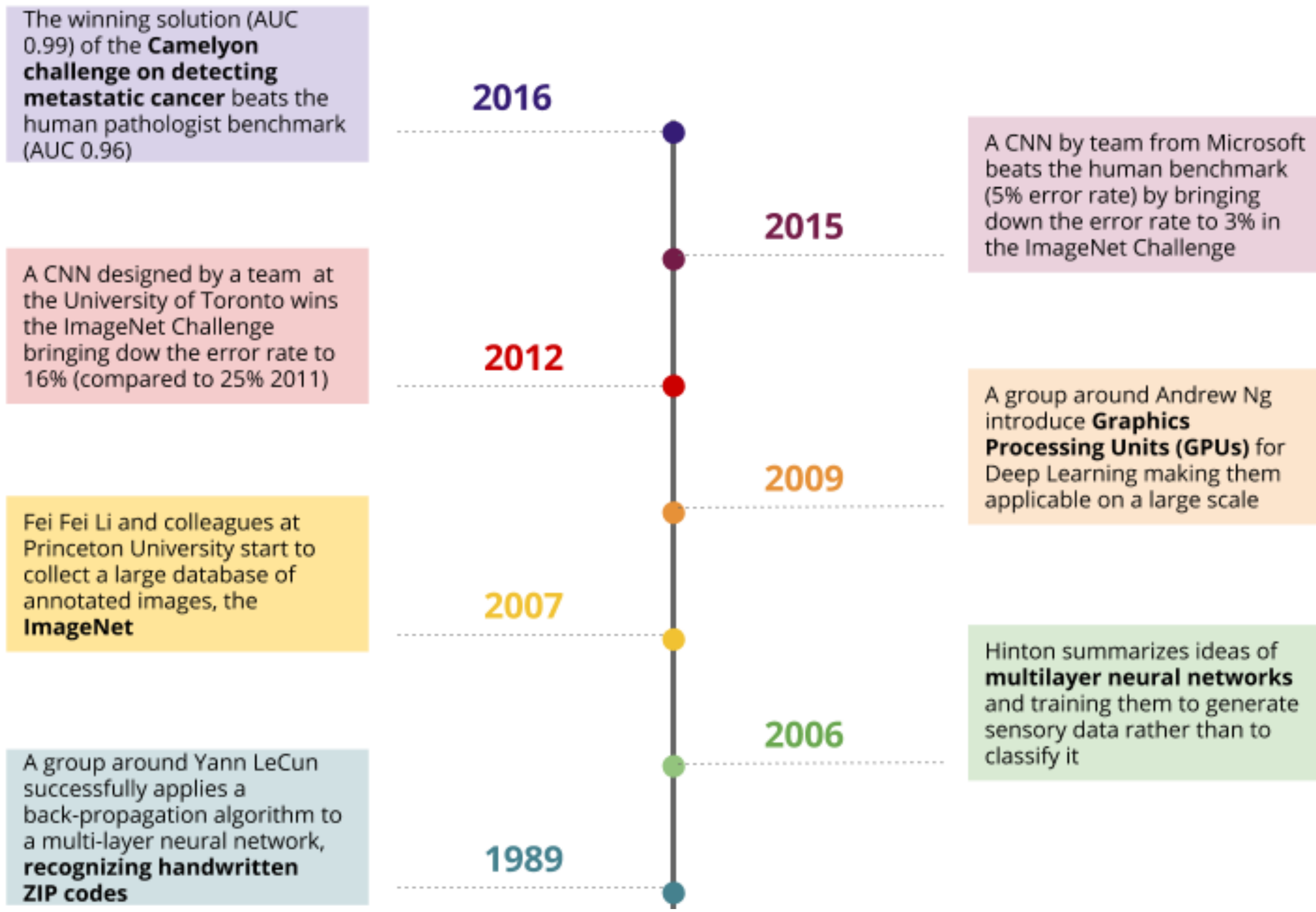
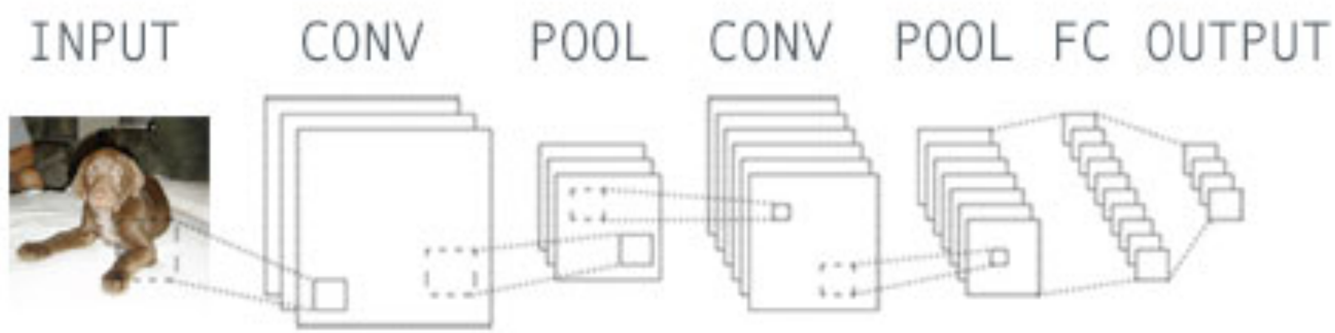
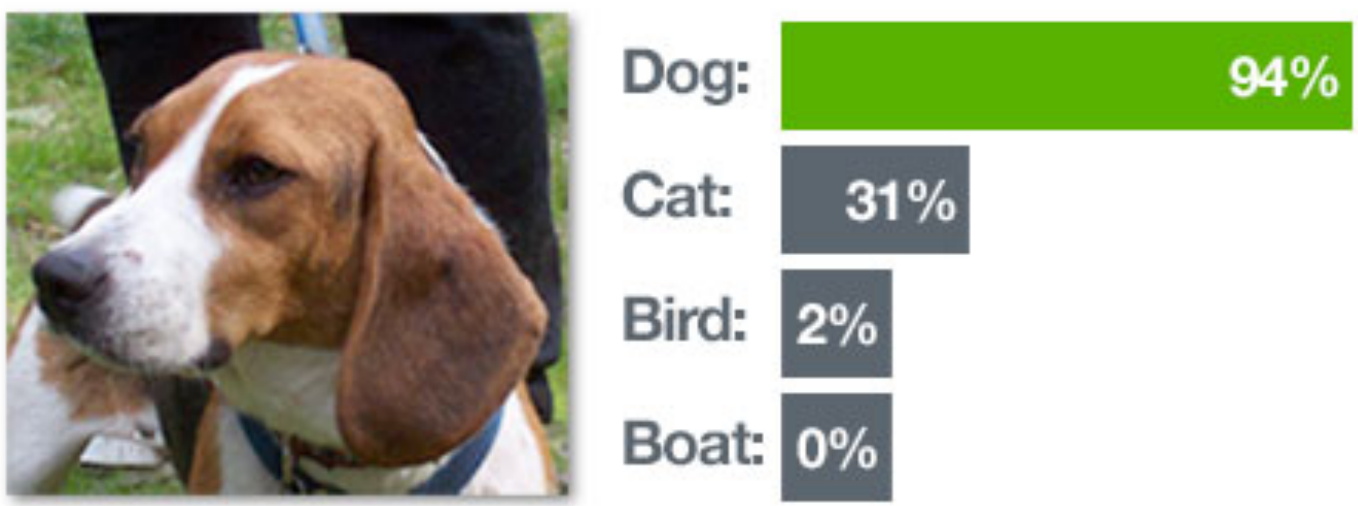
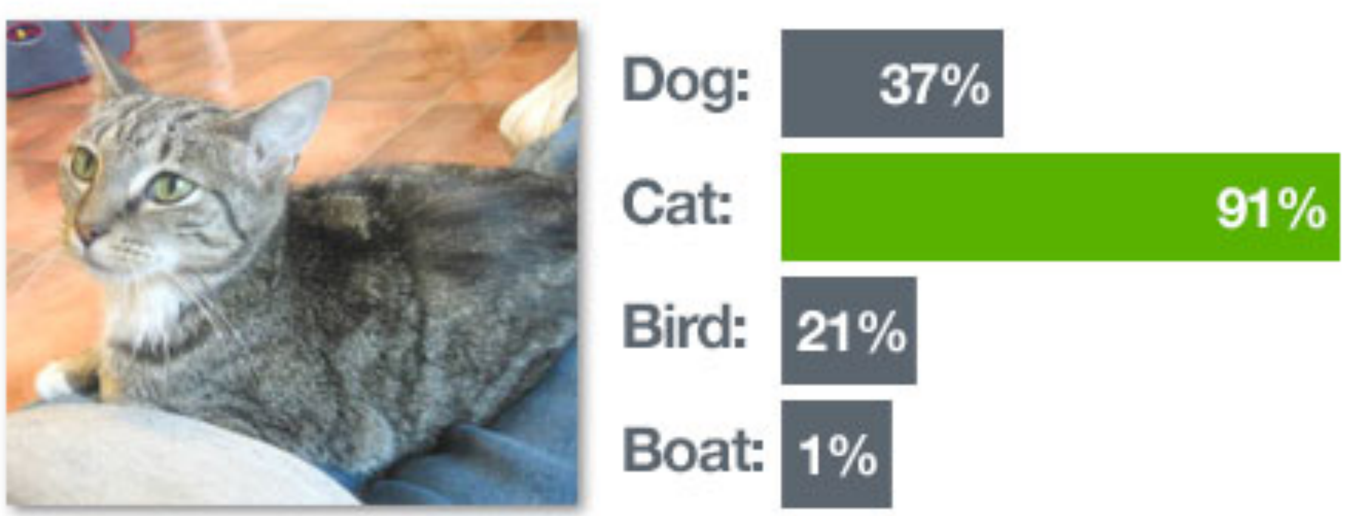
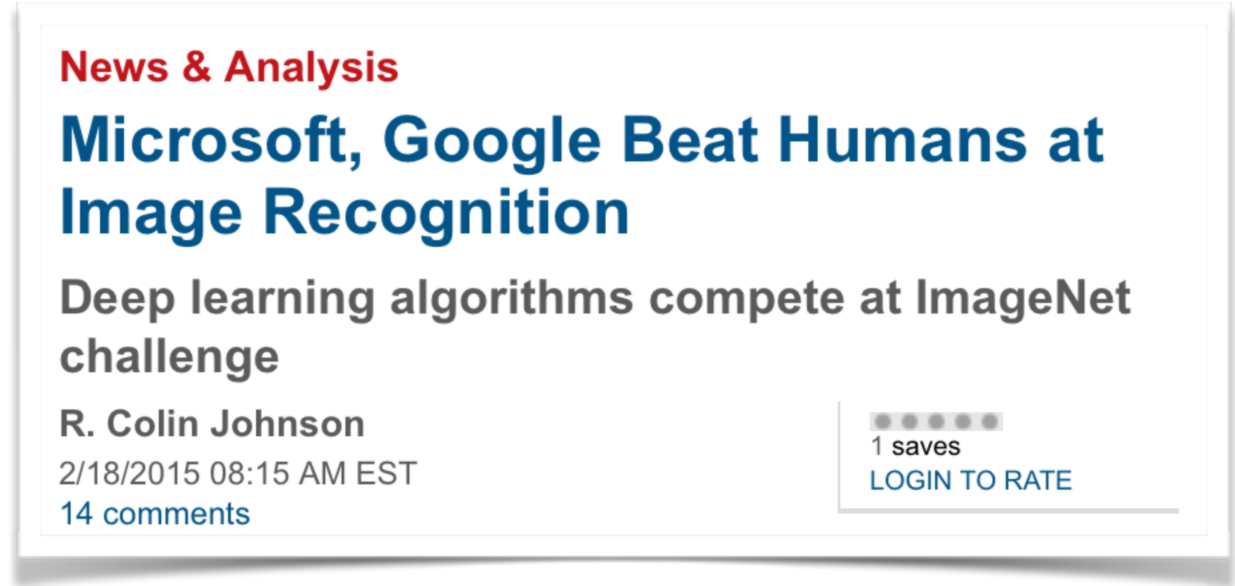


IMAGE CLASSIFICATION









News & Analysis

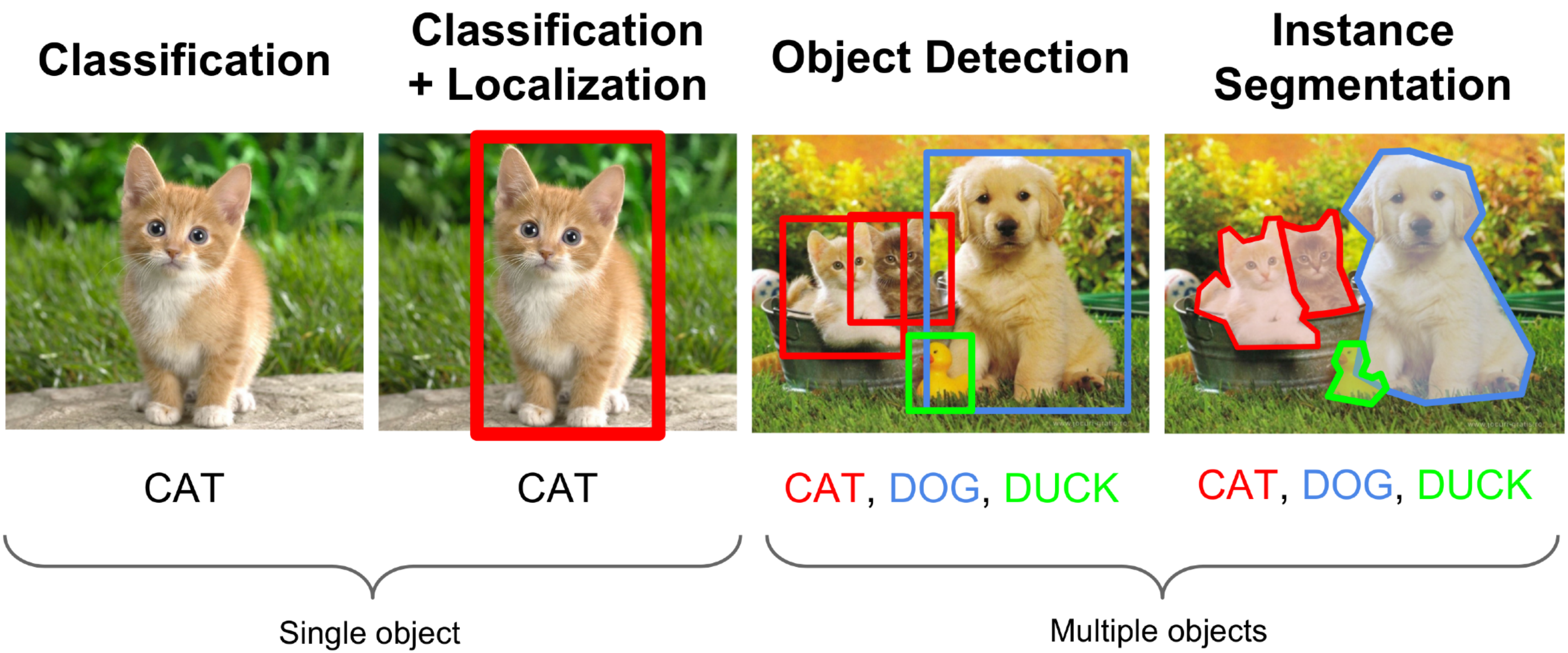
Microsoft, Google Beat Humans at Image Recognition

Deep learning algorithms compete at ImageNet challenge

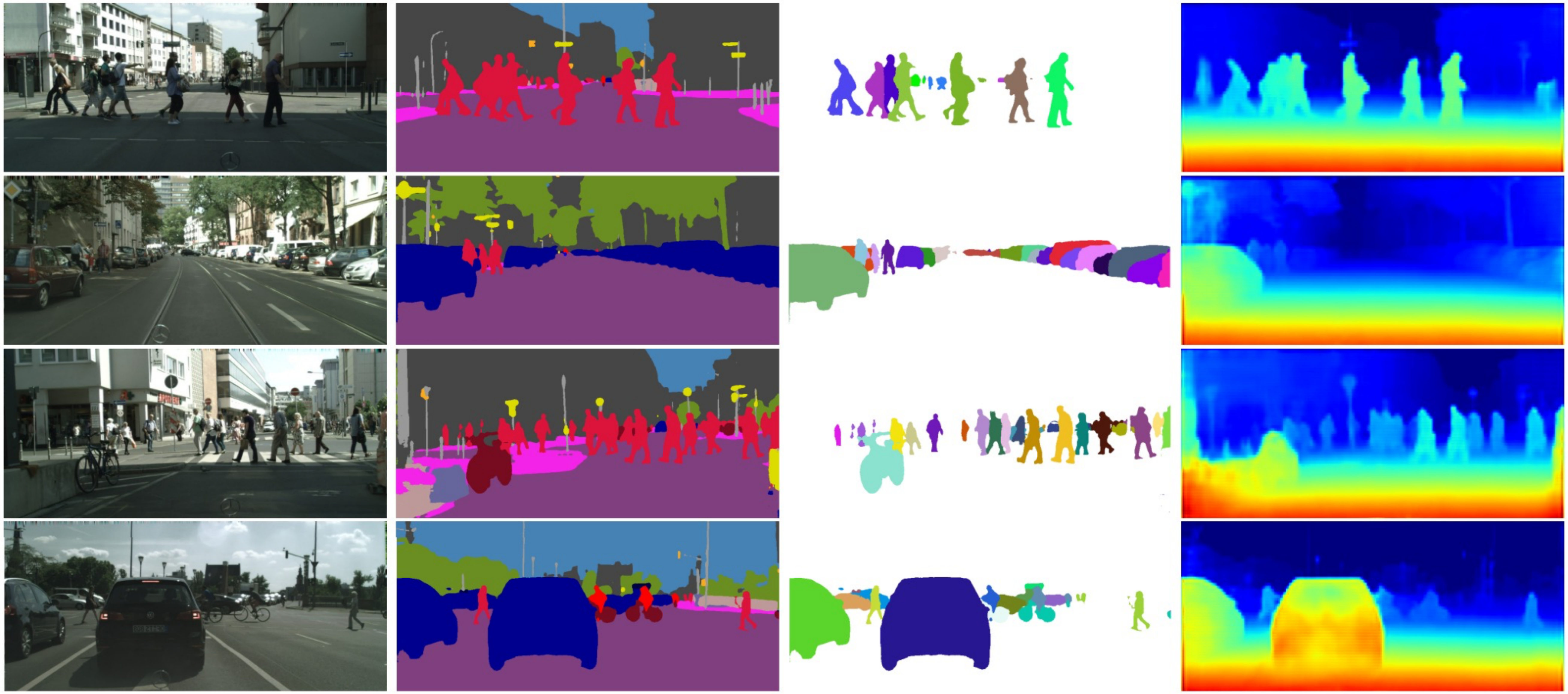
R. Colin Johnson
2/18/2015 08:15 AM EST
14 comments

1 saves
LOGIN TO RATE

CLASSIFICATION → SEGMENTATION



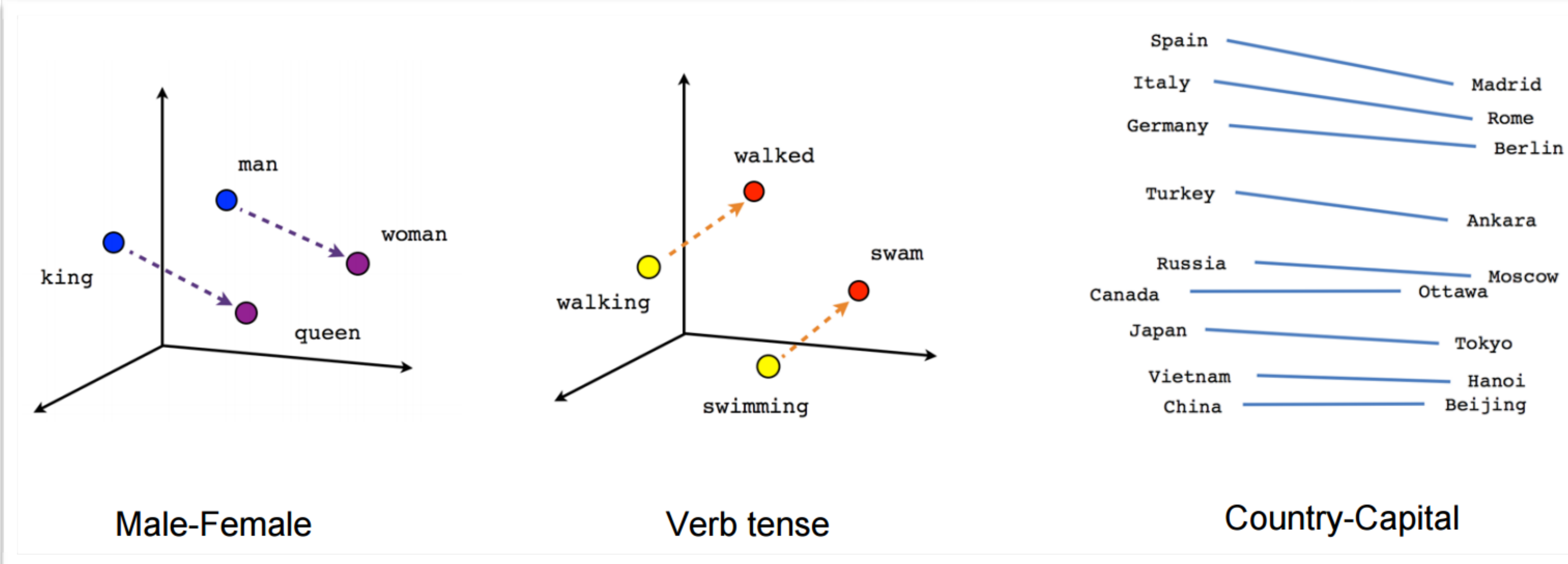
COMPUTER VISION

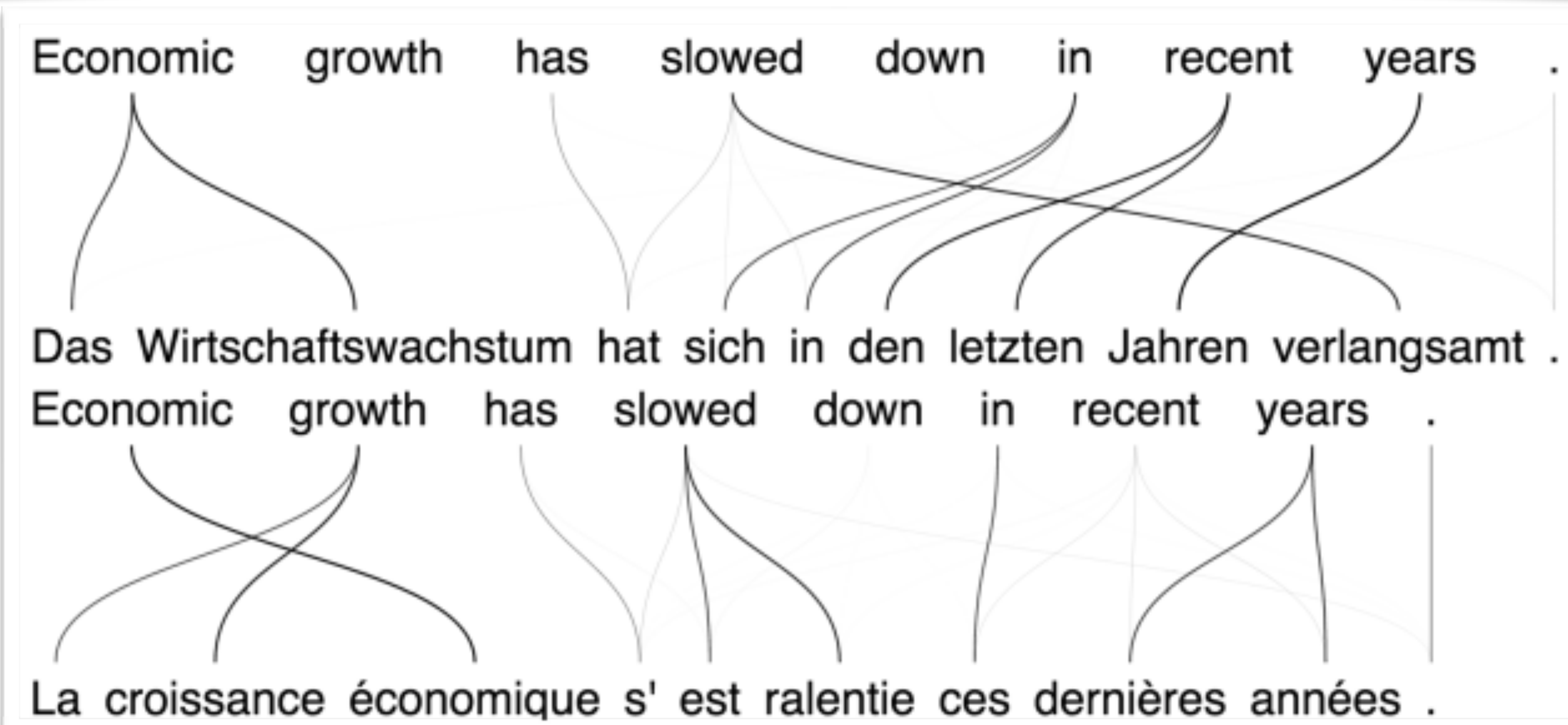


(a) Input image (b) Segmentation output (c) Instance output (d) Depth output

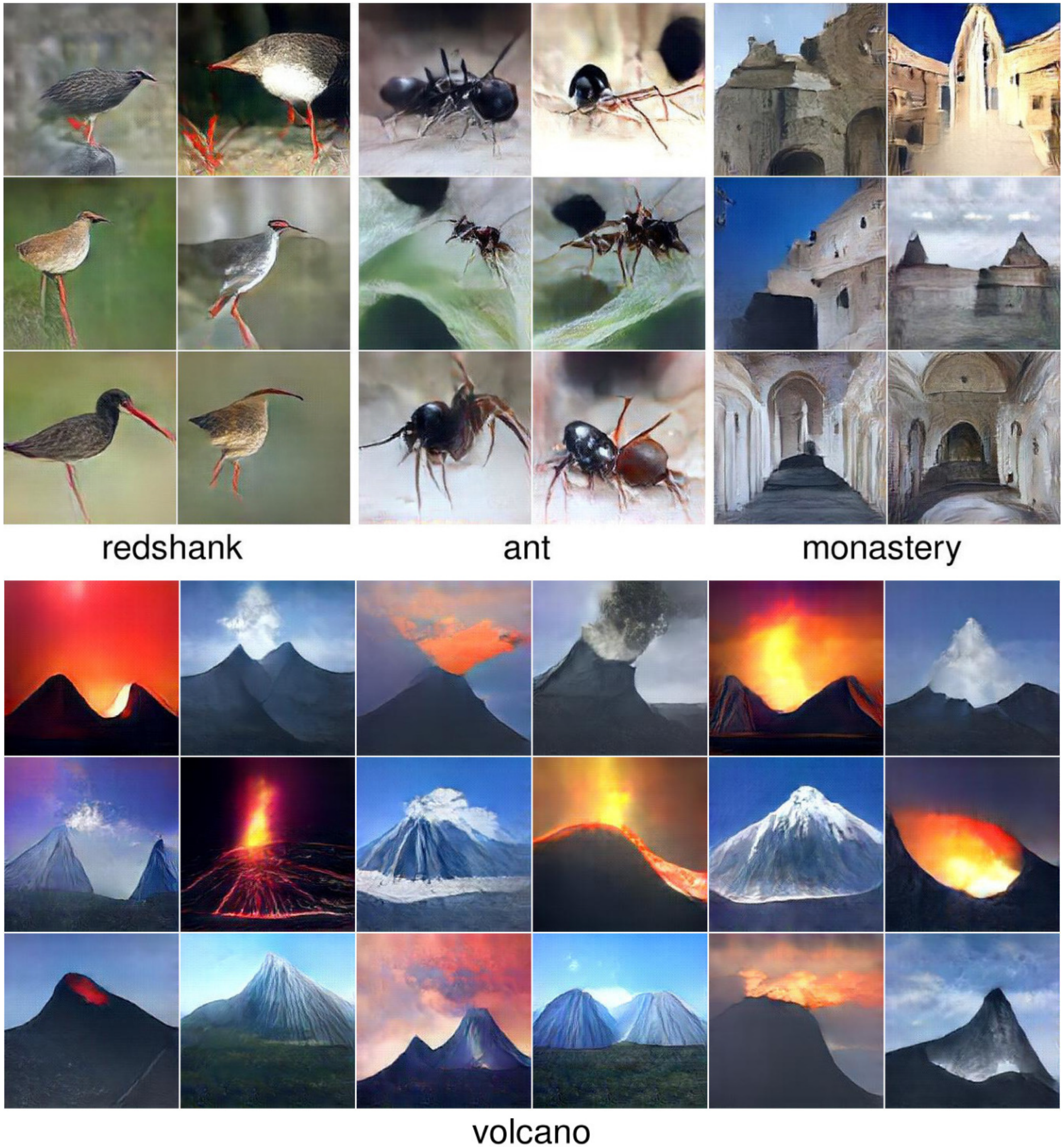
https://alexgkendall.com/computer_vision/bayesian_deep_learning_for_safe_ai/

WORD EMBEDDINGS & TRANSLATION

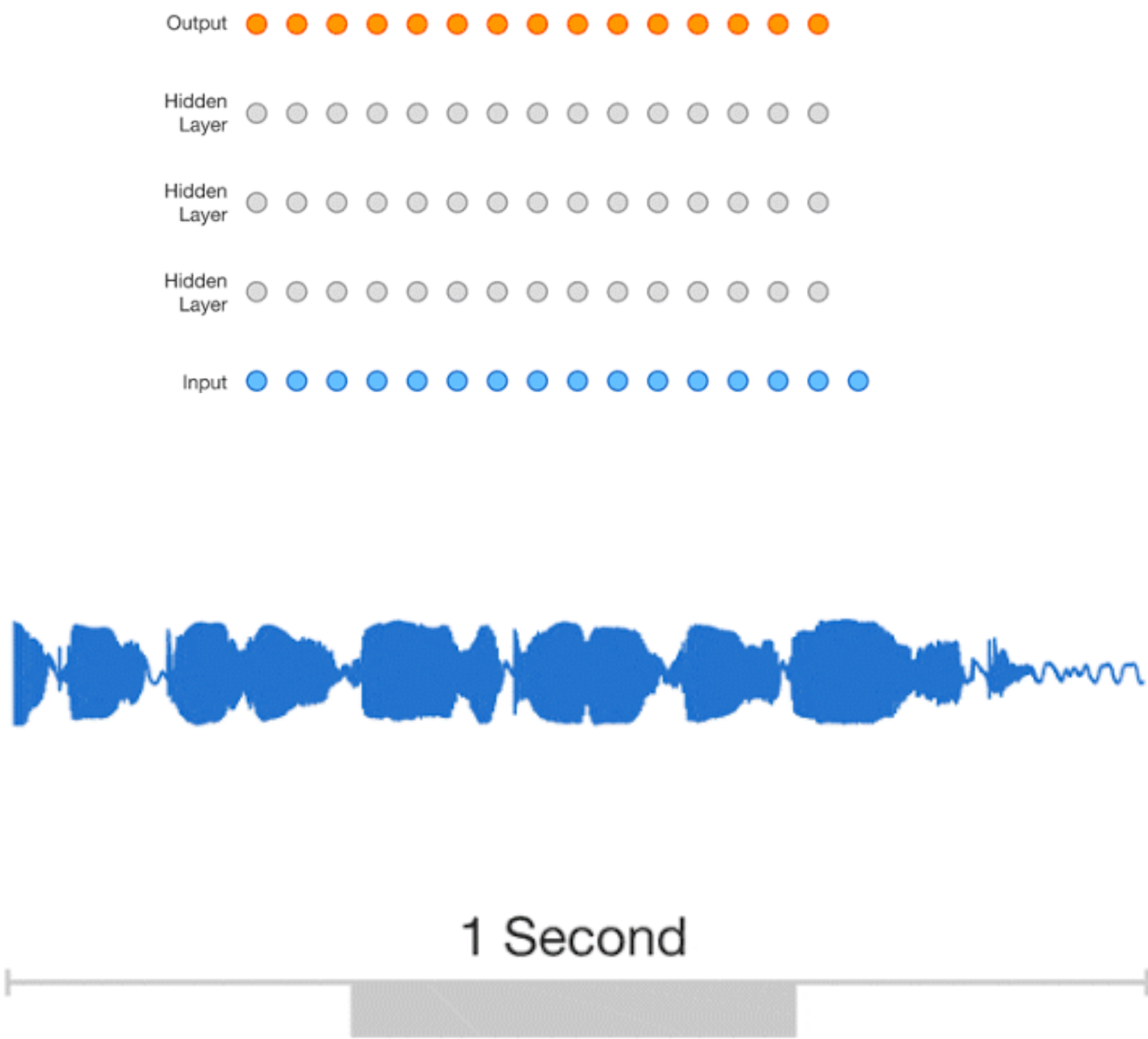




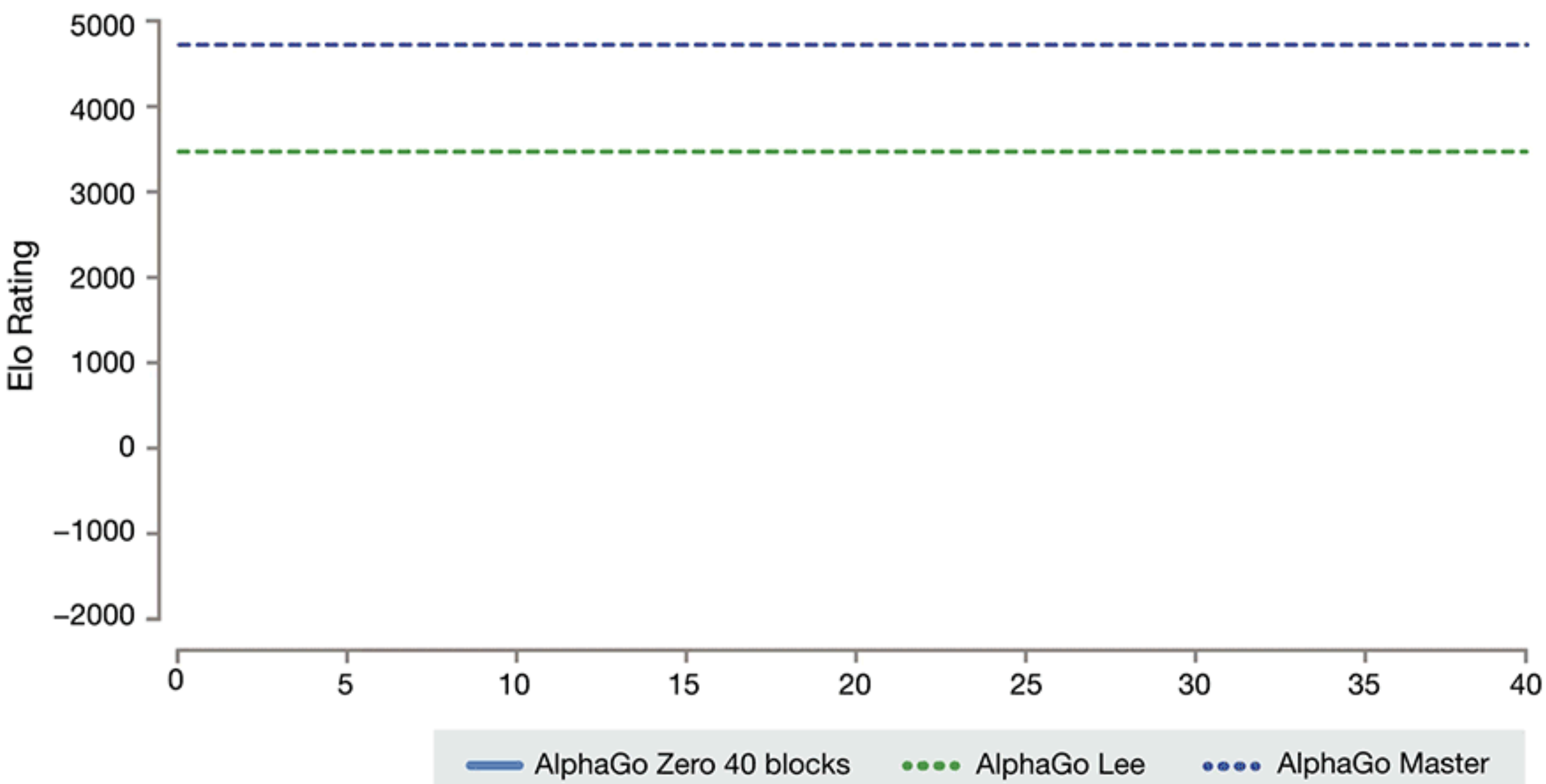
GENERATIVE MODEL FOR IMAGES



WAVENET: A GENERATIVE MODEL FOR RAW AUDIO



AlphaGo



5000
4000
3000
2000
1000
0
−1000
−2000
Elo Rating
0
5
10
15
20
25
30
35
40
AlphaGo Zero 40 blocks
AlphaGo Lee
AlphaGo Master

Why should physicists care?

THE PLAYERS

forward modeling
generation
simulation

PREDICTION

θ
parameters of interest

p(x, z | θ, ν)

z
latent variables
Monte Carlo truth

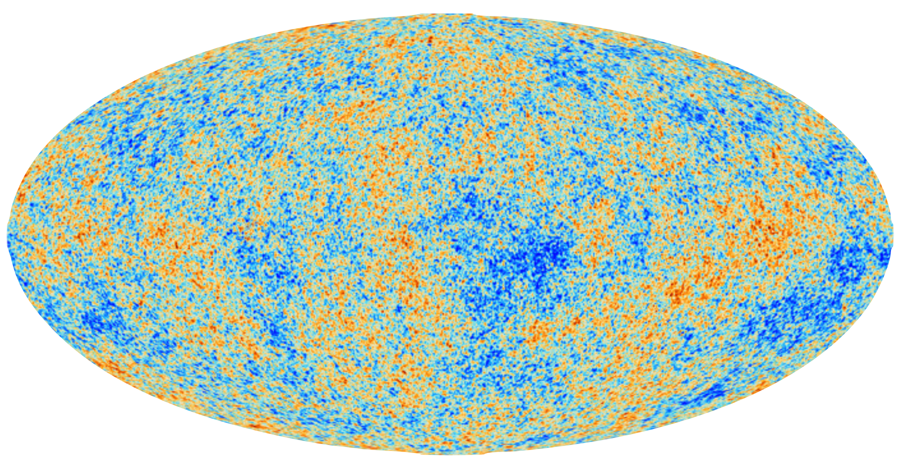
x
observed data
simulated data

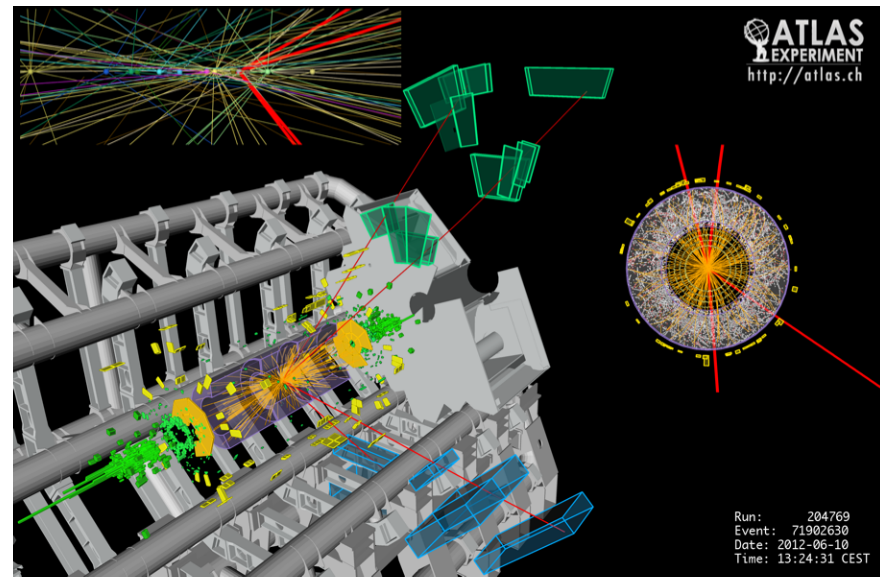
ν
nuisance parameters

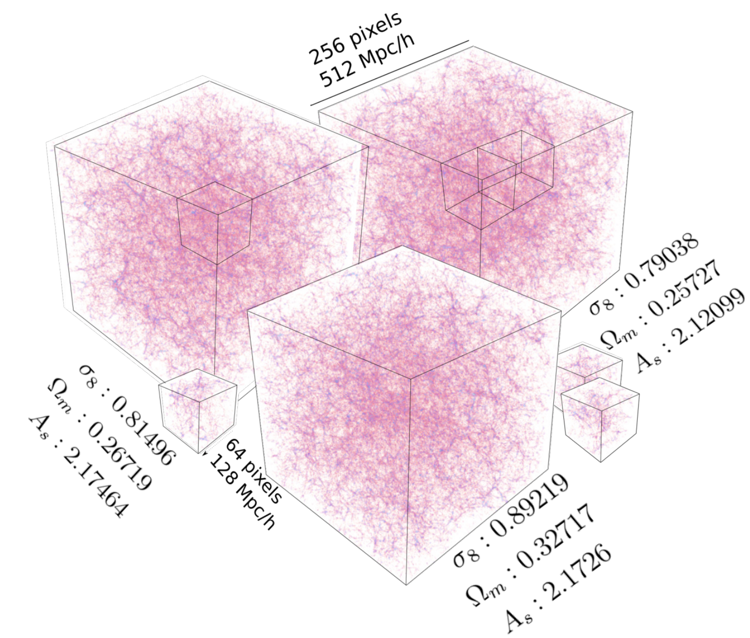
INFERENCE

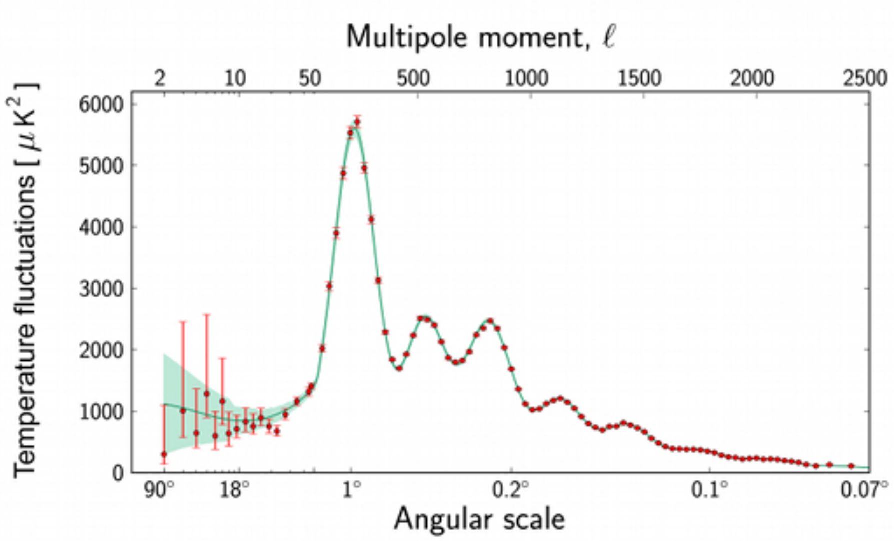
inverse problem
measurement
parameter estimation

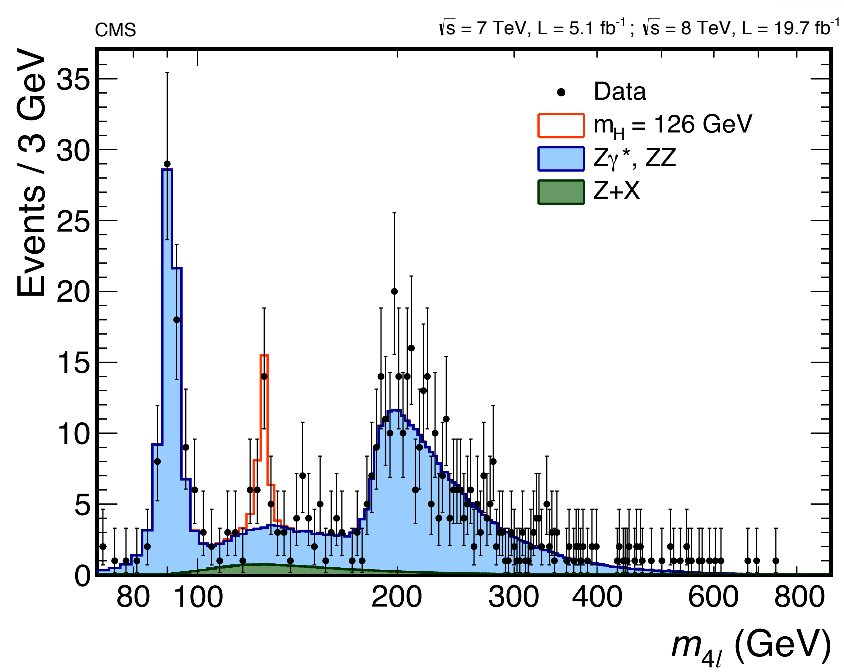
PREDICTION: THE FORWARD MODEL

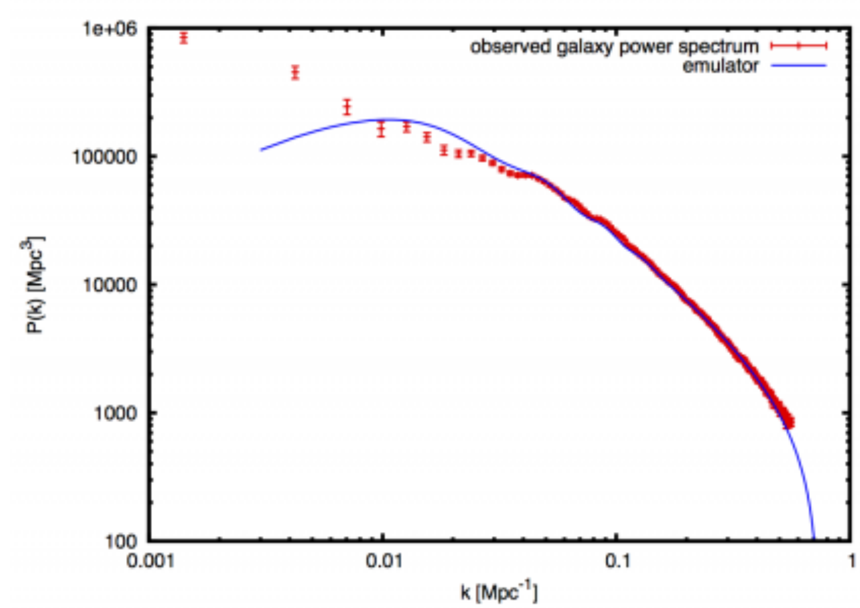












WHY WE SHOULD CARE

Many areas of science have simulations based on some well-motivated mechanistic model.

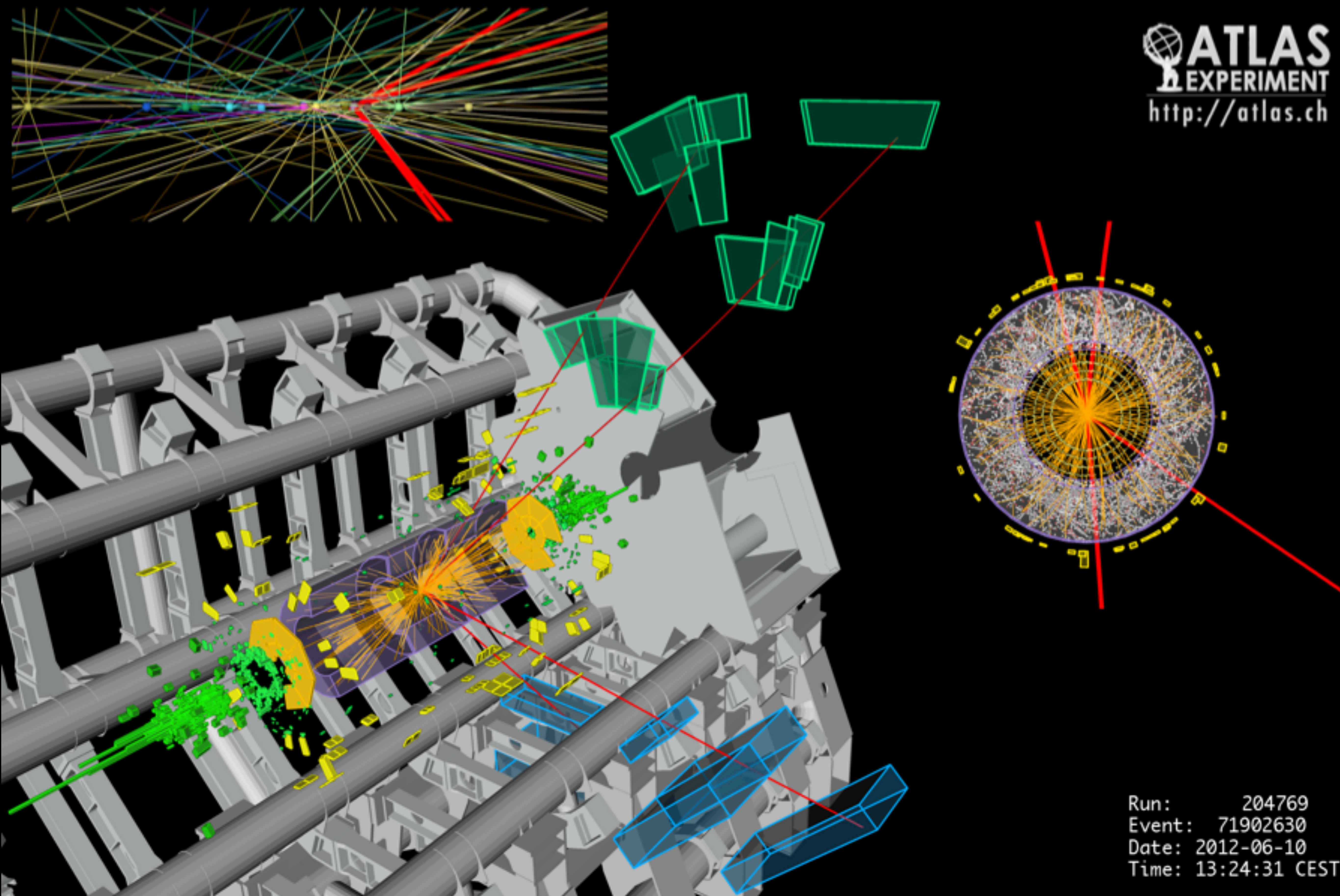
However, the aggregate effect of many interactions between these low-level components leads to an intractable inverse problem.

The developments in machine learning and AI go way beyond improved classifiers and have the potential to effectively bridge the microscopic - macroscopic divide & aid in the inverse problem.

- they can provide effective statistical models that bridge macroscopic phenomena that are tied back to the low-level microscopic (reductionist) model
- generative models and likelihood-free inference are two particularly exciting areas

An example

$$H \to ZZ \to 4l$$


ATLAS
EXPERIMENT
http://atlas.ch
Run: 204769
Event: 71902630
Date: 2012-06-10
Time: 13:24:31 CEST

A Physically Motivated Feature

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

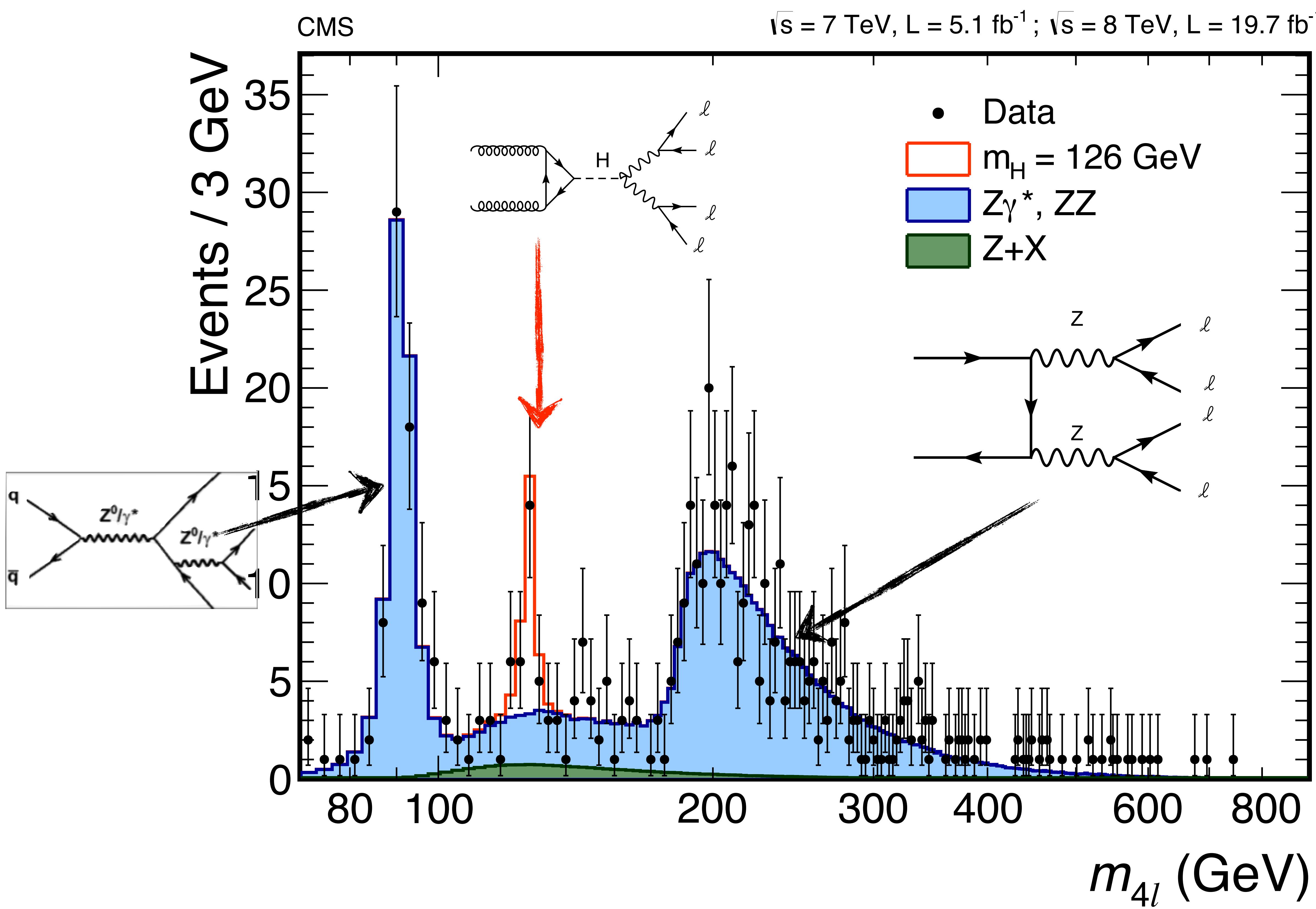
$$E_{\text{Higgs}} = E_{\text{before}} = E_{\text{after}} = \sum_i E_i$$

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

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



The Forward Model

$$\begin{aligned}\mathcal{L}_{SM} = & \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^a_{\mu\nu}G_a^{\mu\nu}}_{\text{kinetic energies and self-interactions of the gauge bosons}} \\ & + \underbrace{\bar{L}\gamma^\mu(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)L + \bar{R}\gamma^\mu(i\partial_\mu - \frac{1}{2}g'YB_\mu)R}_{\text{kinetic energies and electroweak interactions of fermions}} \\ & + \underbrace{\frac{1}{2}\left|(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)\,\phi\right|^2 - V(\phi)}_{W^\pm, Z, \gamma, \text{and Higgs masses and couplings}} \\ & + \underbrace{g''(\bar{q}\gamma^\mu T_a q)\,G^a_\mu}_{\text{interactions between quarks and gluons}} + \underbrace{(G_1\bar{L}\phi R + G_2\bar{L}\phi_c R + h.c.)}_{\text{fermion masses and couplings to Higgs}}\end{aligned}$$

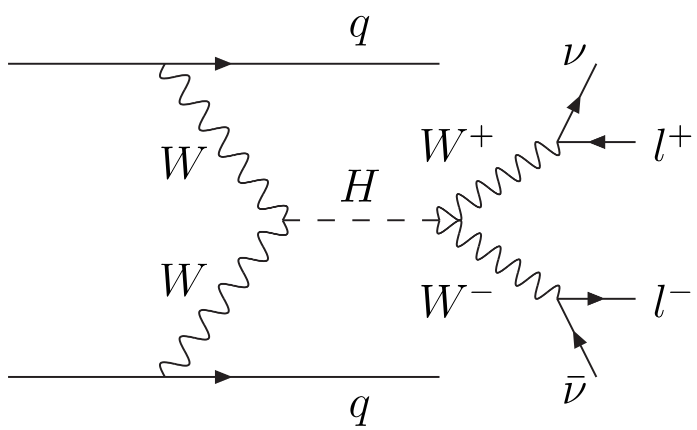
1) We begin with Quantum Field Theory

THE FORWARD MODEL

$$
\begin{aligned}
\mathcal{L}_{SM} = & \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^{a}_{\mu\nu}G^{\mu\nu}_{a}}_{\text{kinetic energies and self-interactions of the gauge bosons}} \\
& + \underbrace{\bar{L}\gamma^{\mu}(i\partial_{\mu} - \frac{1}{2}g\tau\cdot\mathbf{W}_{\mu} - \frac{1}{2}g'YB_{\mu})L + \bar{R}\gamma^{\mu}(i\partial_{\mu} - \frac{1}{2}g'YB_{\mu})R}_{\text{kinetic energies and electroweak interactions of fermions}} \\
& + \underbrace{\frac{1}{2}\left|(i\partial_{\mu} - \frac{1}{2}g\tau\cdot\mathbf{W}_{\mu} - \frac{1}{2}g'YB_{\mu})\,\phi\right|^{2} - V(\phi)}_{W^{\pm},Z,\gamma,\text{and Higgs masses and couplings}} \\
& + \underbrace{g''(\bar{q}\gamma^{\mu}T_{a}q)\,G^{a}_{\mu}}_{\text{interactions between quarks and gluons}} + \underbrace{(G_{1}\bar{L}\phi R + G_{2}\bar{L}\phi_{c}R + h.c.)}_{\text{fermion masses and couplings to Higgs}}
\end{aligned}
$$

1) We begin with Quantum Field Theory

2) Theory gives detailed prediction for high-energy collisions



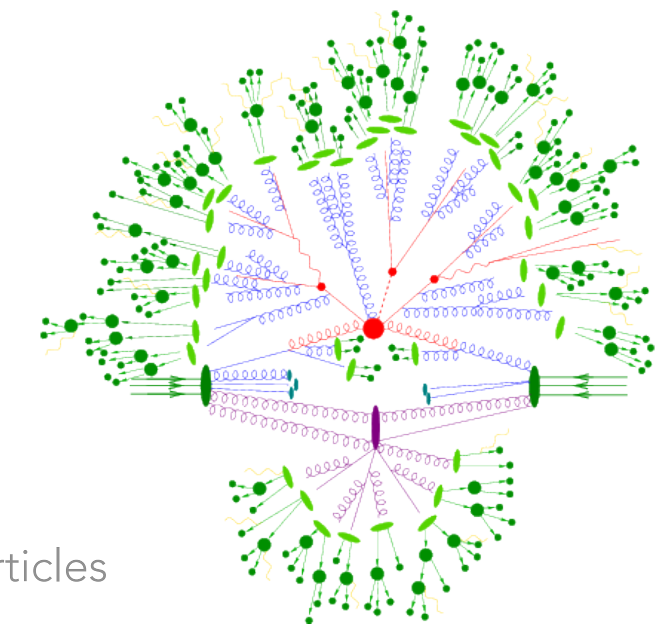
hierarchical: 2 → O(10) → O(100) particles

The Forward Model

$$
\begin{aligned}
\mathcal{L}_{SM} = \; & \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^{a}_{\mu\nu}G^{\mu\nu}_{a}}_{\text{kinetic energies and self-interactions of the gauge bosons}} \\
& + \underbrace{\bar{L}\gamma^{\mu}(i\partial_{\mu} - \frac{1}{2}g\tau\cdot\mathbf{W}_{\mu} - \frac{1}{2}g'YB_{\mu})L + \bar{R}\gamma^{\mu}(i\partial_{\mu} - \frac{1}{2}g'YB_{\mu})R}_{\text{kinetic energies and electroweak interactions of fermions}} \\
& + \underbrace{\frac{1}{2}\left|(i\partial_{\mu} - \frac{1}{2}g\tau\cdot\mathbf{W}_{\mu} - \frac{1}{2}g'YB_{\mu})\,\phi\right|^{2} - V(\phi)}_{W^{\pm},Z,\gamma,\text{and Higgs masses and couplings}} \\
& + \underbrace{g''(\bar{q}\gamma^{\mu}T_{a}q)\,G^{a}_{\mu}}_{\text{interactions between quarks and gluons}} + \underbrace{(G_{1}\bar{L}\phi R + G_{2}\bar{L}\phi_{c}R + h.c.)}_{\text{fermion masses and couplings to Higgs}}
\end{aligned}
$$

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

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

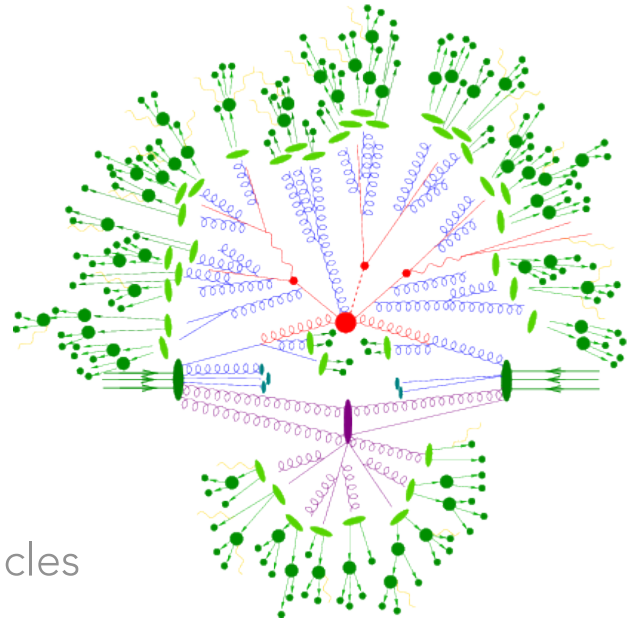
$$+ \underbrace{\bar{L}\gamma^\mu(i\partial_\mu - \frac{1}{2}g\tau\cdot\mathbf{W}_\mu - \frac{1}{2}g'YB_\mu)L + \bar{R}\gamma^\mu(i\partial_\mu - \frac{1}{2}g'YB_\mu)R}_{\text{kinetic energies and electroweak interactions of fermions}}$$

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

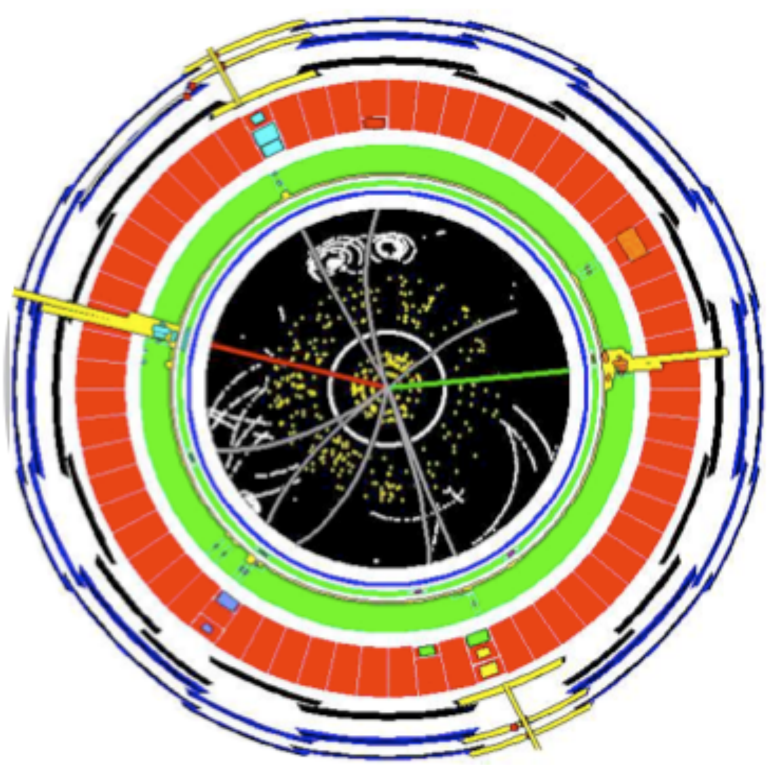
$$+ \underbrace{g''(\bar{q}\gamma^\mu T_a q)\,G^a_\mu}_{\text{interactions between quarks and gluons}} + \underbrace{(G_1\bar{L}\phi R + G_2\bar{L}\phi_c R + h.c.)}_{\text{fermion masses and couplings to Higgs}}$$

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.

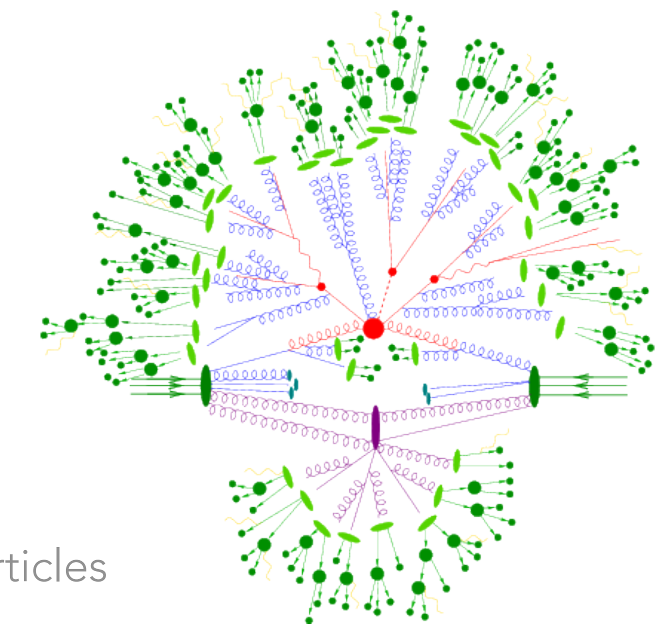
>100 million sensors

The Forward Model

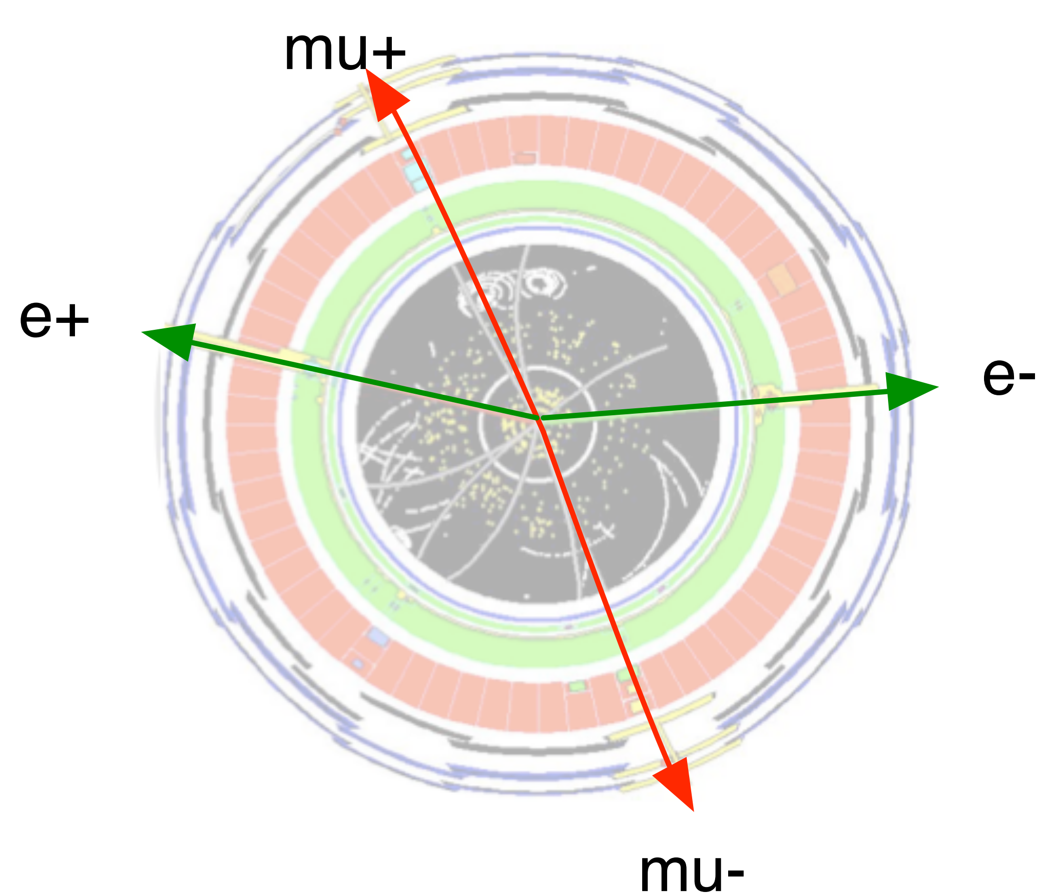
$$\mathcal{L}_{SM} = \underbrace{\frac{1}{4}\mathbf{W}_{\mu\nu}\cdot\mathbf{W}^{\mu\nu} - \frac{1}{4}B_{\mu\nu}B^{\mu\nu} - \frac{1}{4}G^{a}_{\mu\nu}G^{\mu\nu}_{a}}_{\text{kinetic energies and self-interactions of the gauge bosons}}$$
$$+ \underbrace{\bar{L}\gamma^{\mu}(i\partial_{\mu} - \frac{1}{2}g\tau\cdot\mathbf{W}_{\mu} - \frac{1}{2}g'YB_{\mu})L + \bar{R}\gamma^{\mu}(i\partial_{\mu} - \frac{1}{2}g'YB_{\mu})R}_{\text{kinetic energies and electroweak interactions of fermions}}$$
$$+ \underbrace{\frac{1}{2}\left|(i\partial_{\mu} - \frac{1}{2}g\tau\cdot\mathbf{W}_{\mu} - \frac{1}{2}g'YB_{\mu})\,\phi\right|^{2} - V(\phi)}_{W^{\pm},Z,\gamma,\text{and Higgs masses and couplings}}$$
$$+ \underbrace{g''(\bar{q}\gamma^{\mu}T_{a}q)\,G^{a}_{\mu}}_{\text{interactions between quarks and gluons}} + \underbrace{(G_{1}\bar{L}\phi R + G_{2}\bar{L}\phi_{c}R + h.c.)}_{\text{fermion masses and couplings to Higgs}}$$

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

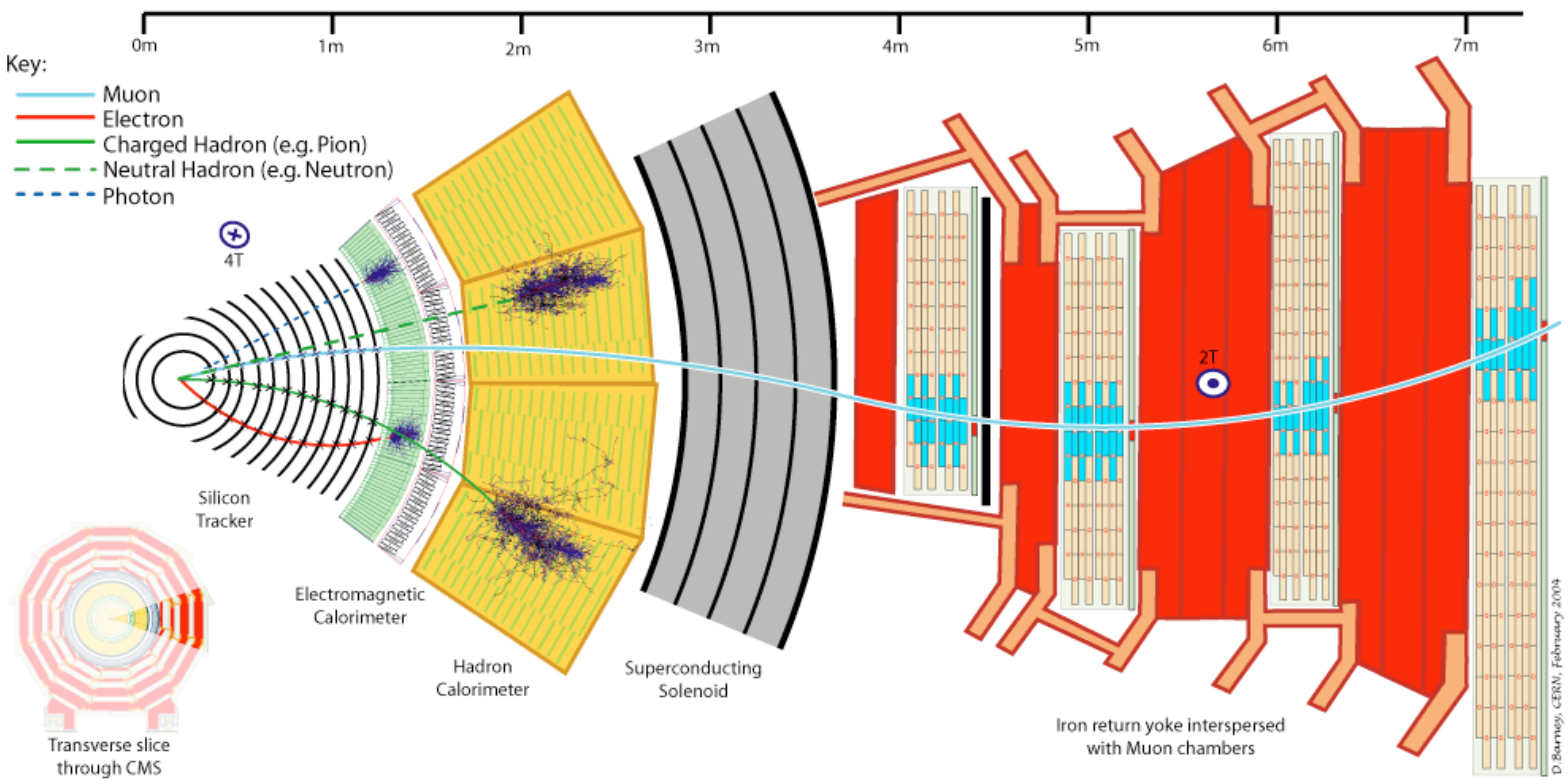
4) Finally, we run particle identification and feature extraction algorithms on the simulated data as if they were from real collisions.

~10-30 features describe interesting part

Conceptually: Prob(detector response | particles)

Implementation: Monte Carlo integration over micro-physics

Consequence: evaluation of the likelihood is intractable



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"

THE CRUX, AN INTRACTABLE INTEGRAL

Monte Carlo Sampling

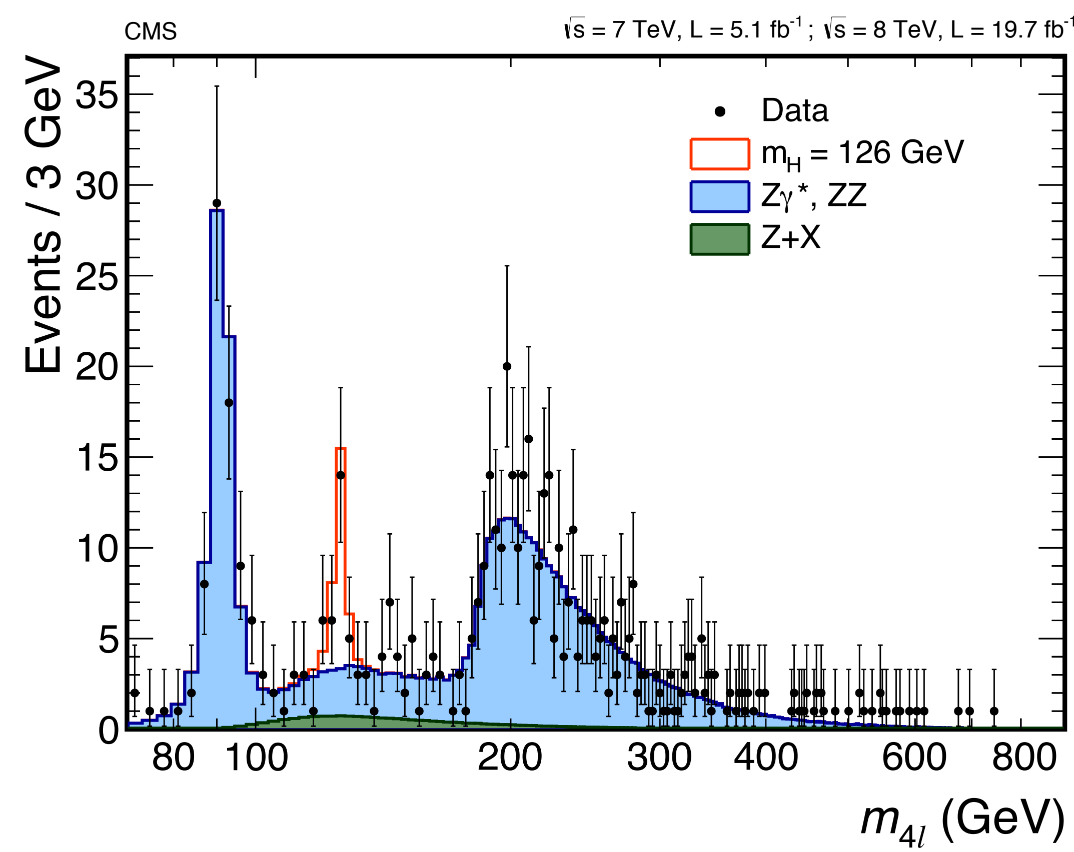
observed

what happened in simulation

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

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

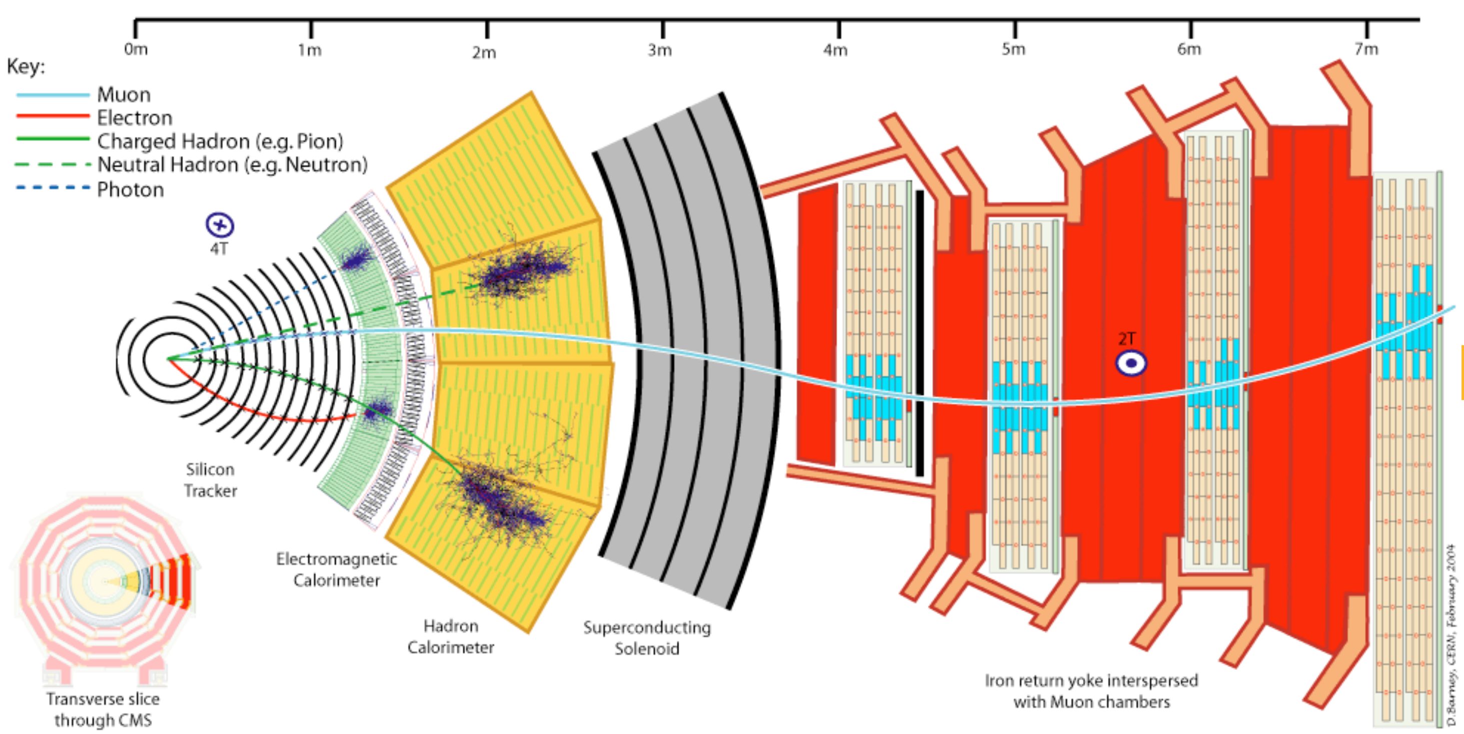
histogram approximation

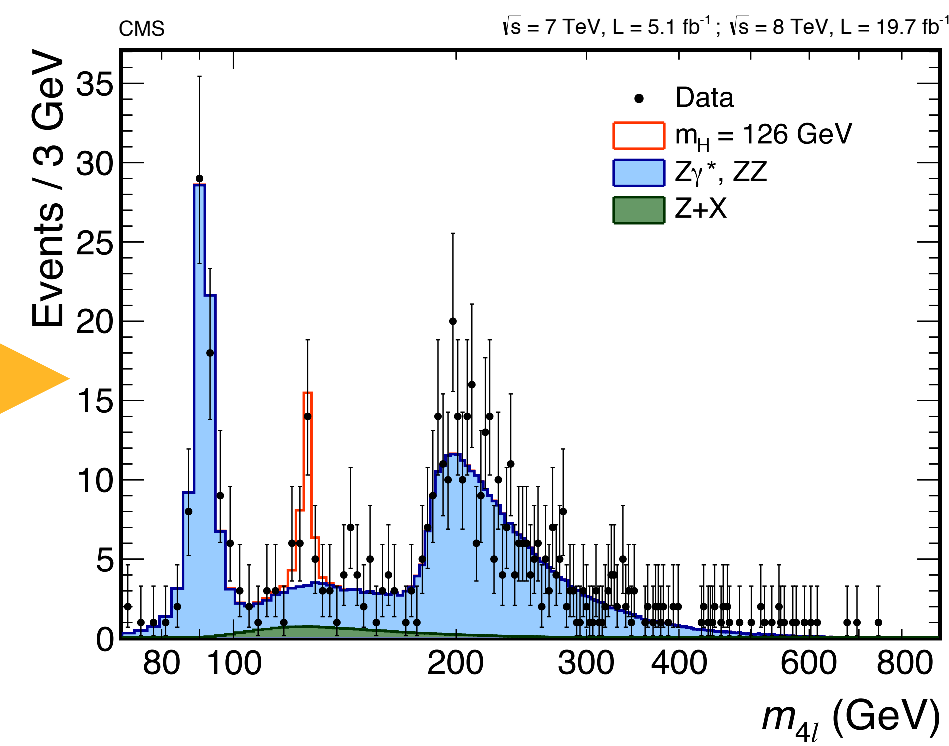


10^8 SENSORS → 1 REAL-VALUED QUANTITY

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|θ) **approximated** using histograms (univariate density estimation)

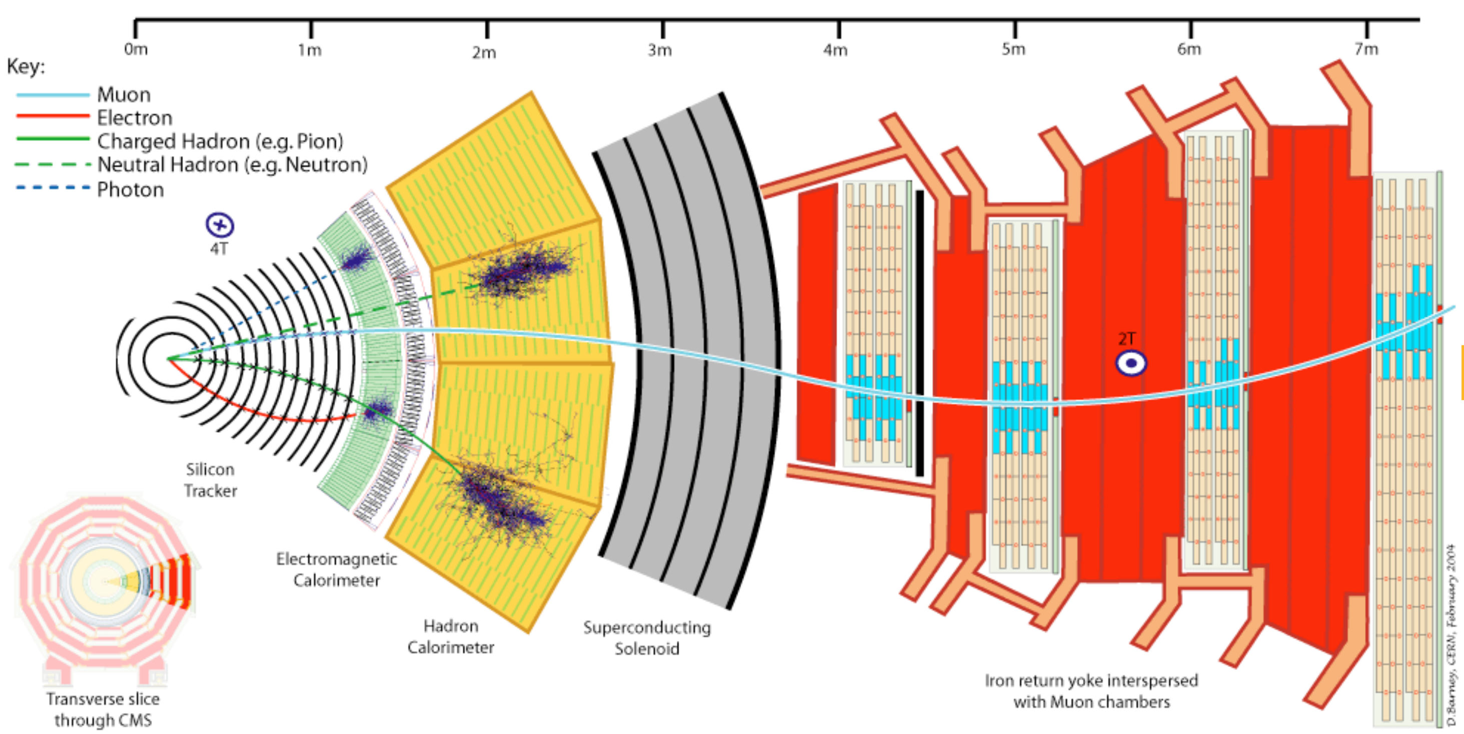


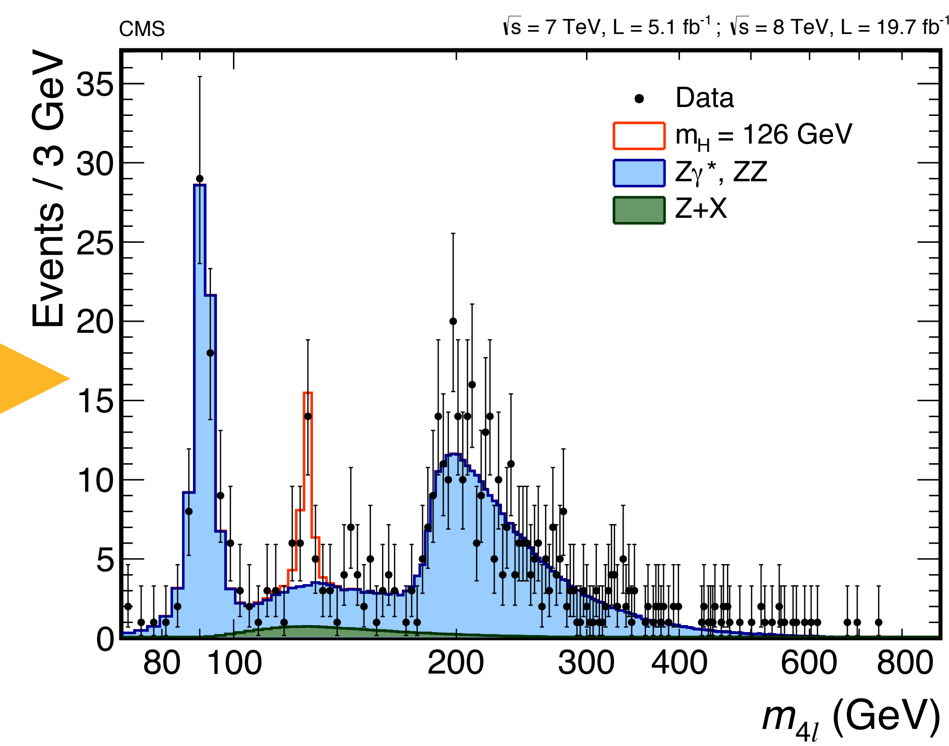


10^8 SENSORS → 1 REAL-VALUED QUANTITY

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|θ) **approximated** using histograms (univariate density estimation)



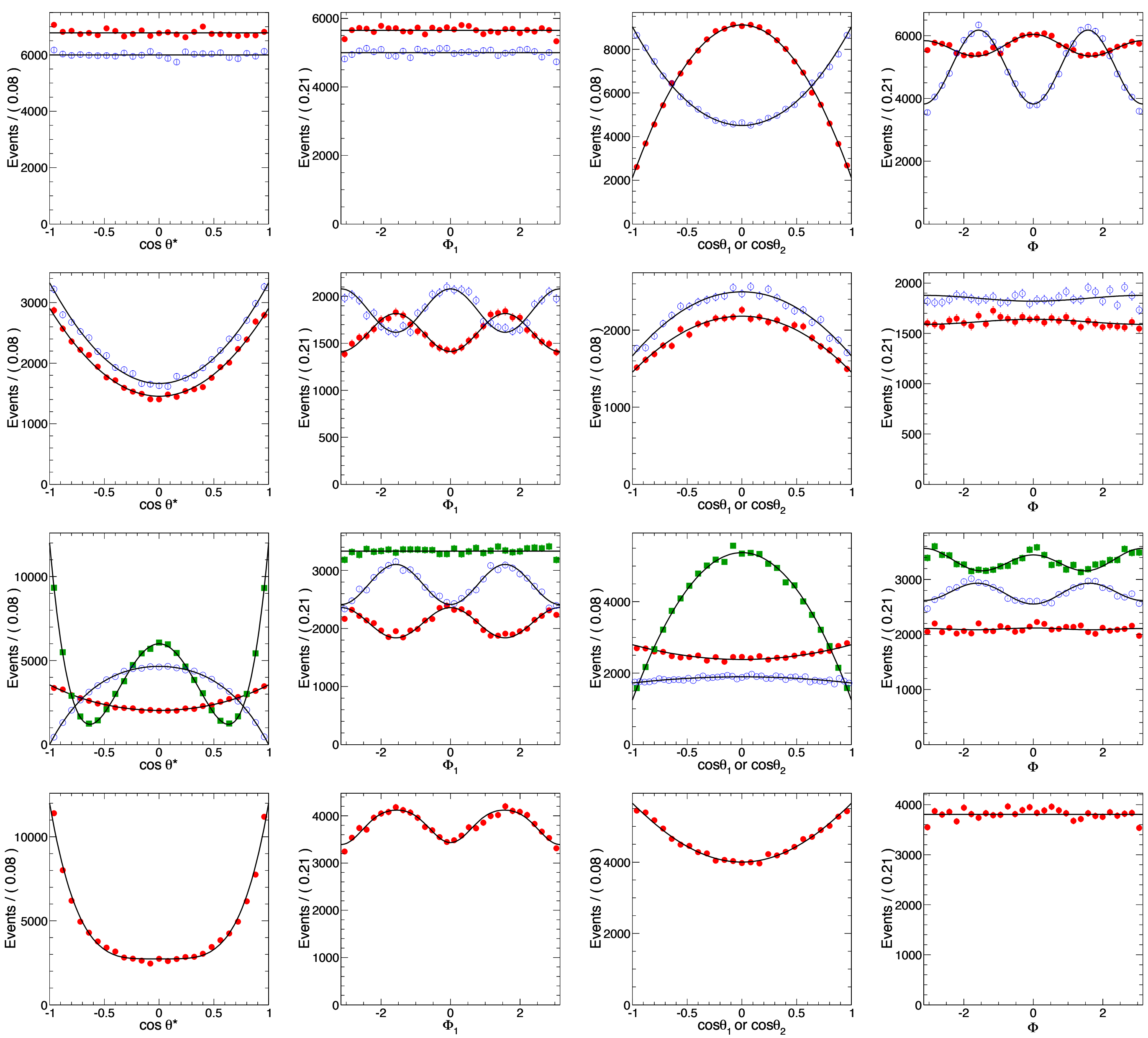


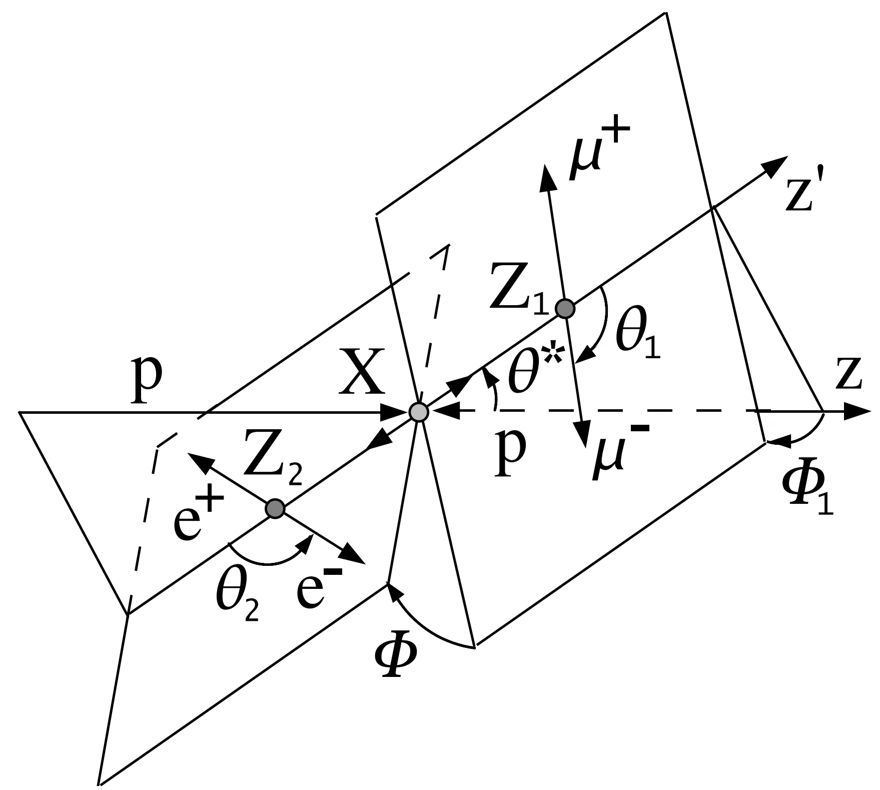
This doesn't scale if x is high dimensional!

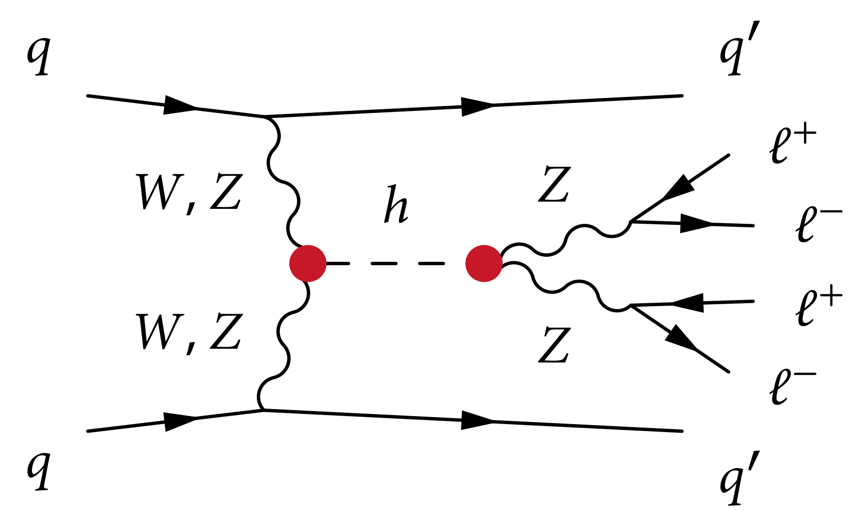
HIGH DIMENSIONAL EXAMPLE

For instance, when looking for deviations from the standard model Higgs, we would like to look at all sorts of kinematic correlations

- thus each observation **x** is high-dimensional

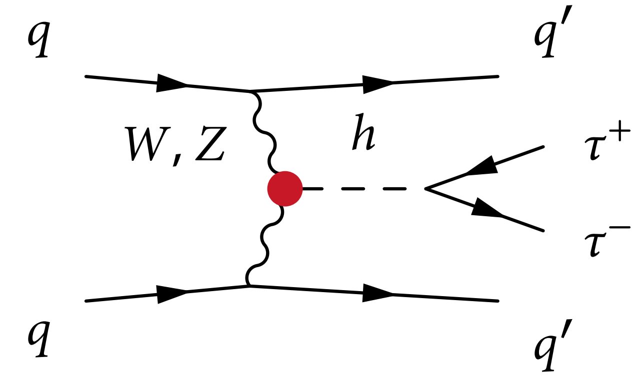






HIGGS EFT

"Better Higgs Measurements Through Information Geometry" [arXiv:1612.05261]



- Theory language: dimension-6 operators of SM EFT, $\mathcal{L} \supset \sum_i \frac{f_i}{\Lambda^2} \mathcal{O}_i$

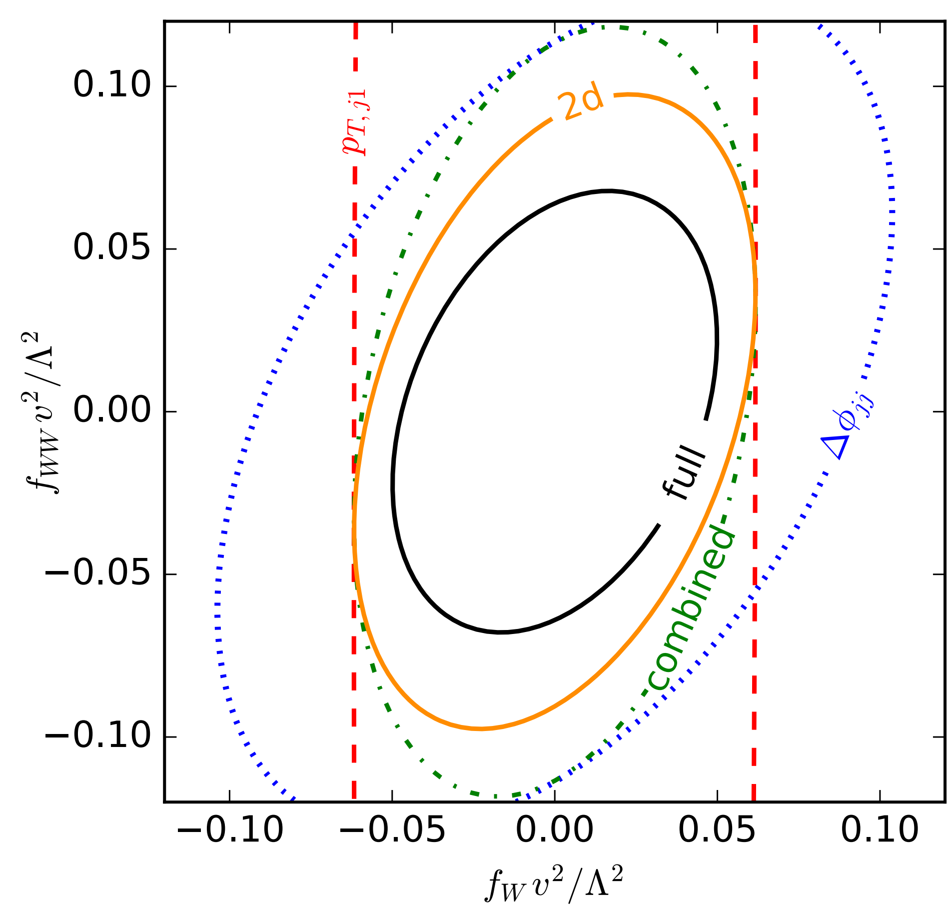
  [W. Buchmuller, D. Wyler 85; K. Hagiwara, S. Ishihara, S. R. Szalapski, D. Zeppenfeld 93; B. Grzadkowski, M. Iskrzynski, M. Misiak, J. Rosiek 1008.4884; …]

  - Total rate: $\mathcal{O}_{\phi,2} = \frac{1}{2} \partial^\mu (\phi^\dagger \phi) \partial_\mu (\phi^\dagger \phi)$
  - New kinematic structures:

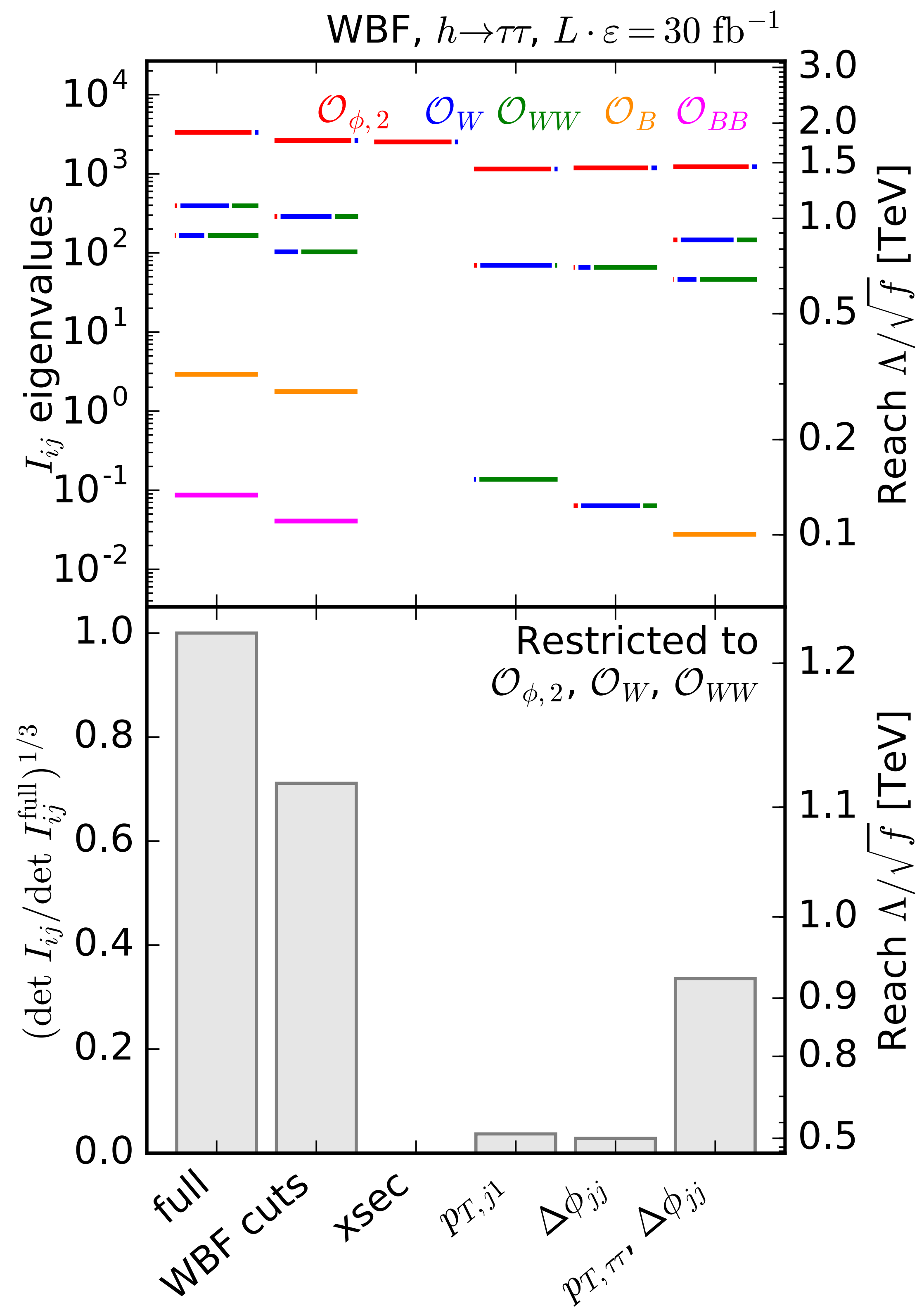
$$\mathcal{O}_B = \mathrm{i}\frac{g}{2} (D^\mu \phi^\dagger)(D^\nu \phi) B_{\mu\nu} \qquad \mathcal{O}_W = \mathrm{i}\frac{g}{2} (D^\mu \phi)^\dagger \sigma^k (D^\nu \phi) W^k_{\mu\nu}$$

$$\mathcal{O}_{BB} = -\frac{g'^2}{4} (\phi^\dagger \phi) B_{\mu\nu} B^{\mu\nu} \qquad \mathcal{O}_{WW} = -\frac{g^2}{4} (\phi^\dagger \phi) W^k_{\mu\nu} W^{\mu\nu\, k}$$

  - *CP* violation: $\mathcal{O}_{W\widetilde{W}} = -\frac{g^2}{4} (\phi^\dagger \phi) W^k_{\mu\nu} \widetilde{W}^{\mu\nu\, k}$
  - Others strongly constrained by EWPD or redundant



Equivalent to 3x more data!

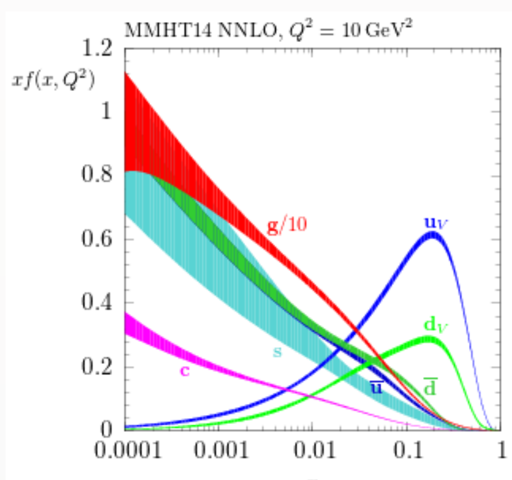


"MEM" approach uses a transfer function $W(x|z)$ to simplify parton shower and detector response and integrates other latent variables

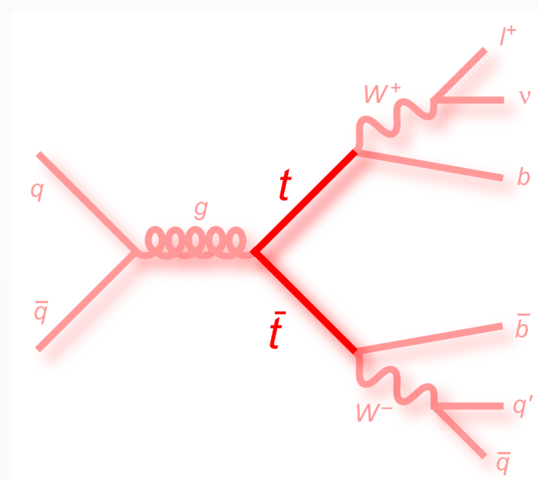
Introduction - The MEM

Probability (weight) of the experimental event x given the hypothesis α :

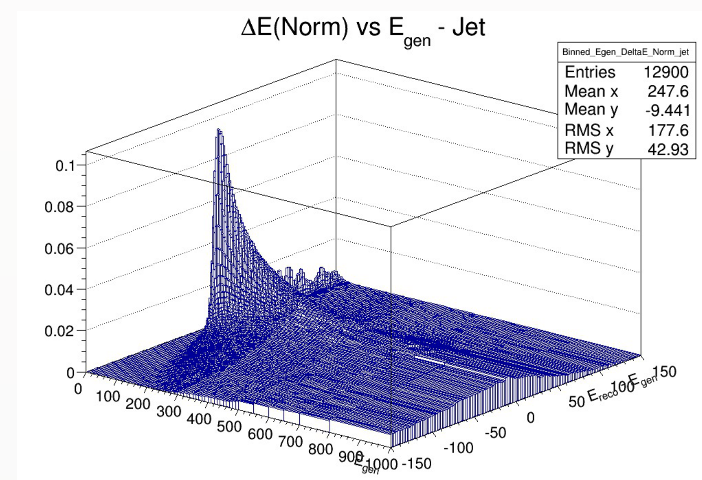
$$P(\mathrm{x}|\alpha) = \frac{1}{\sigma_\alpha} \int \mathrm{d}\Phi(z)\mathrm{d}x_1\mathrm{d}x_2 f(x_1)f(x_2)|M_\alpha(y, x_1, x_2)|^2 W(\mathrm{x}|z) \quad (1)$$



PDF



Matrix Element



Transfer Function

Efficiency and acceptance neglected in this sketch.

- Sébastien Brochet
- Brieuc François
- Alessia Saggio
- Miguel Vidal
- Sébastien Wertz

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.

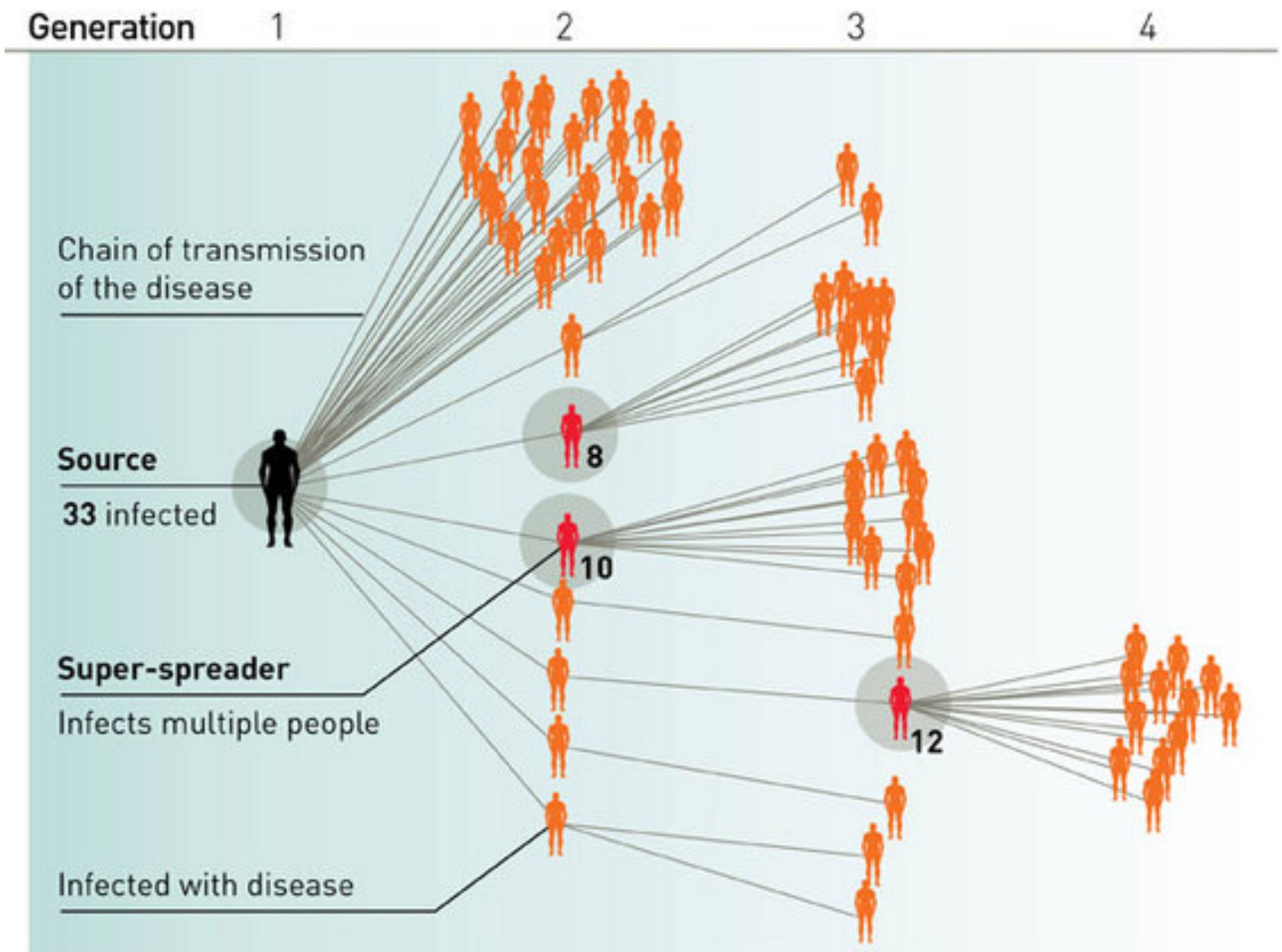
ABC in Montreal | ABC in Montreal (2014)

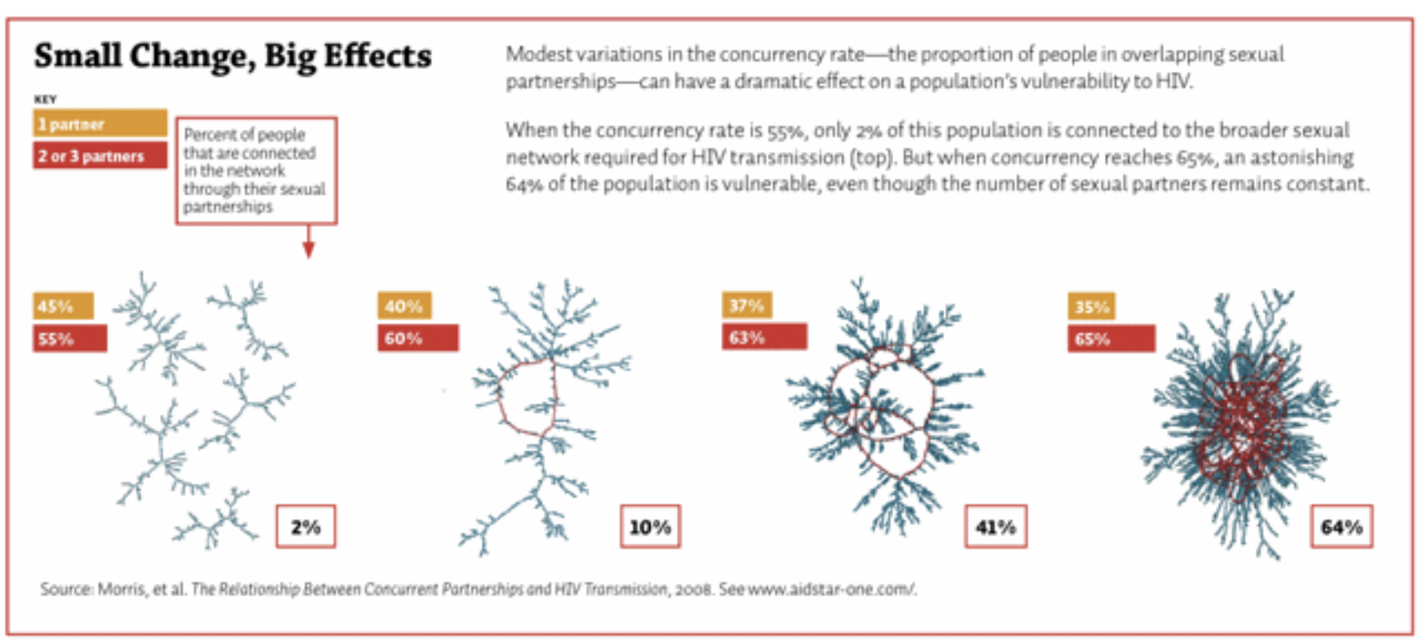
ABC in Montreal

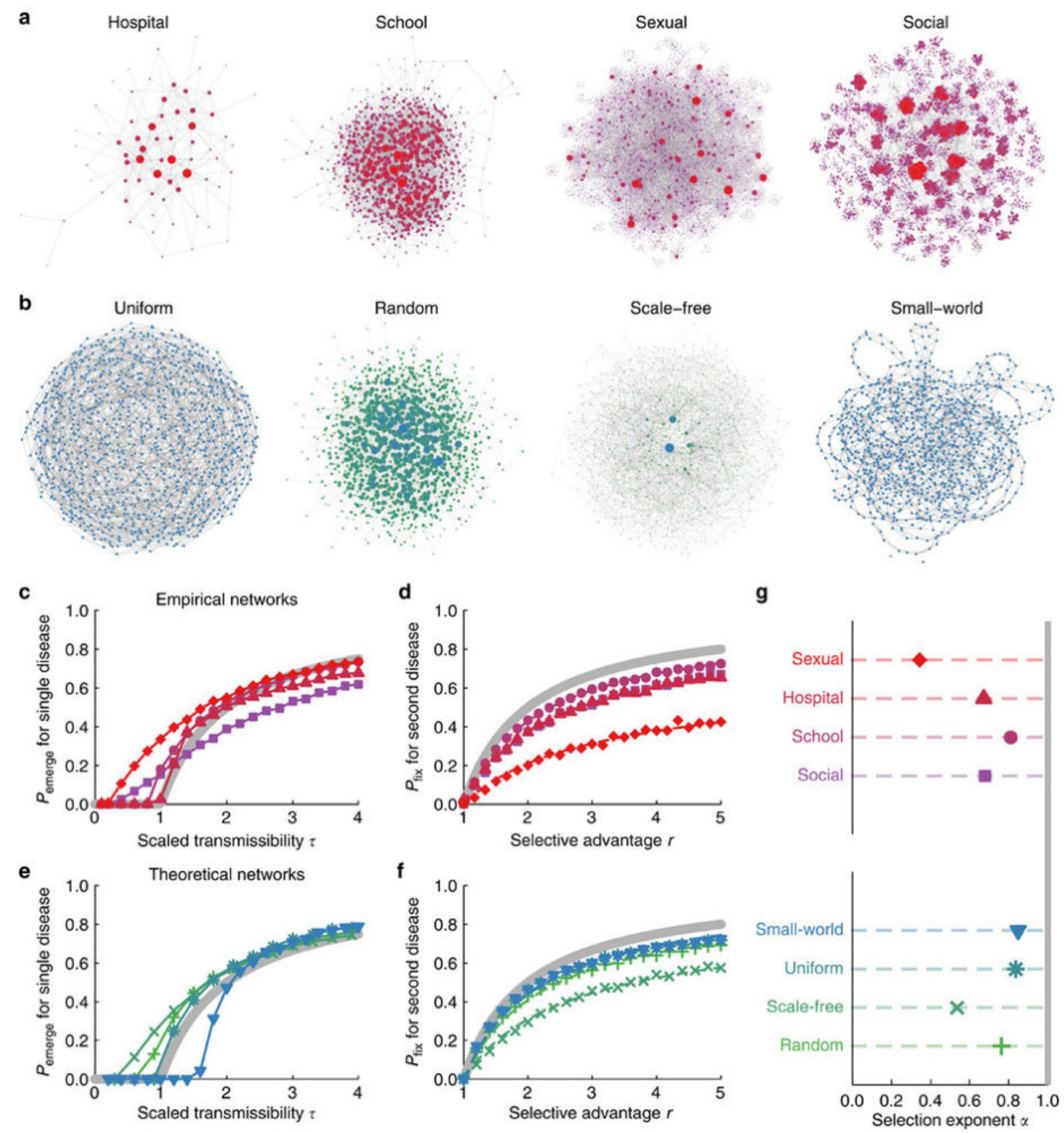
Approximate Bayesian computation (ABC) or likelihood-free (LF) methods have developed mostly beyond the radar of the machine learning community, but are important tools for a large and diverse segment of the scientific community. This is particularly true for systems and population biology, computational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

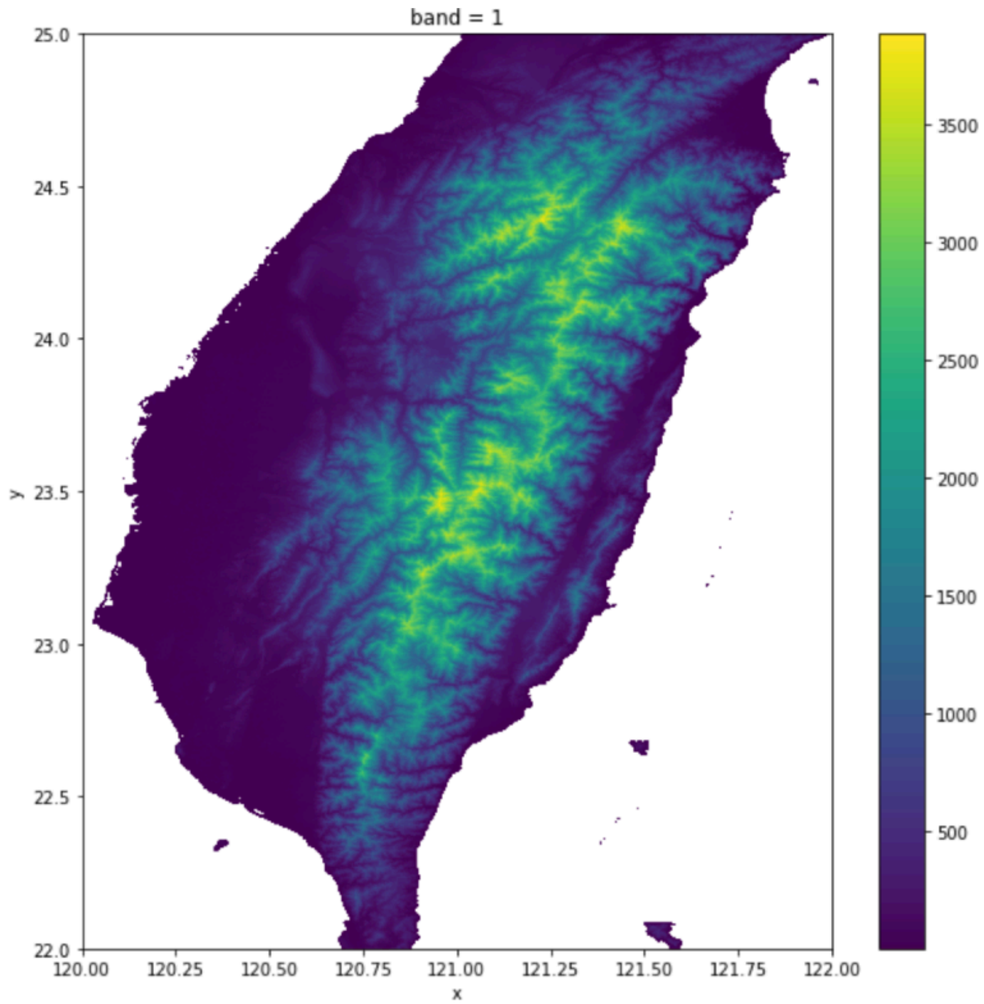
EPIDEMIOLOGY & POPULATION GENETICS

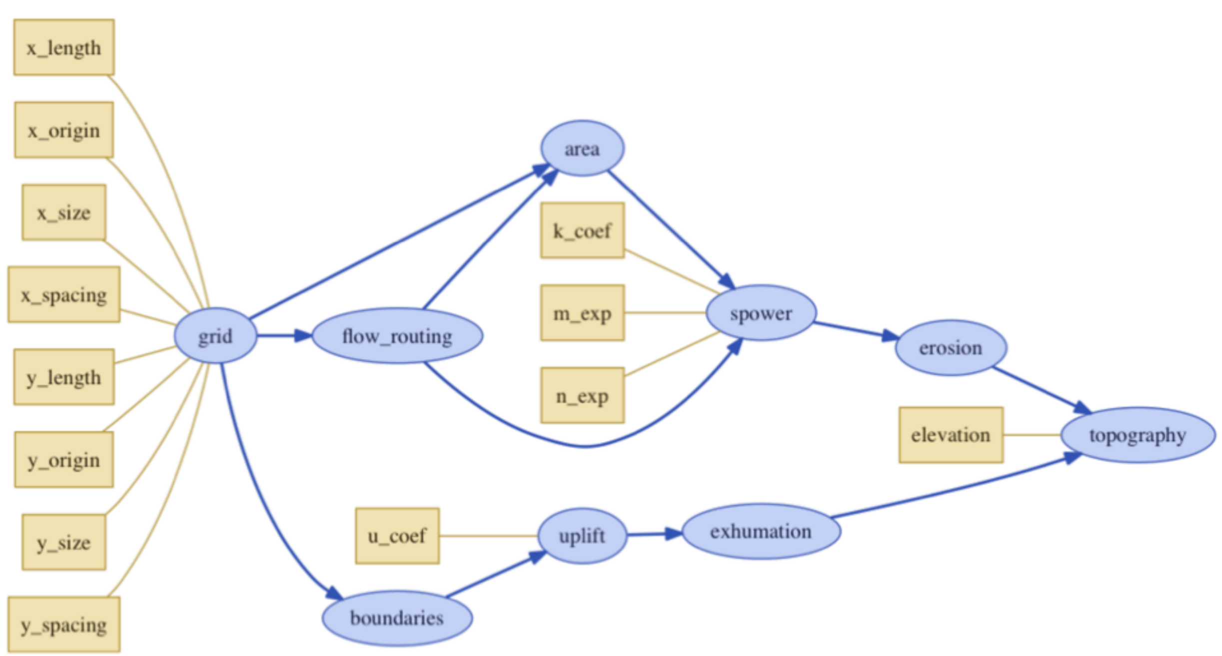




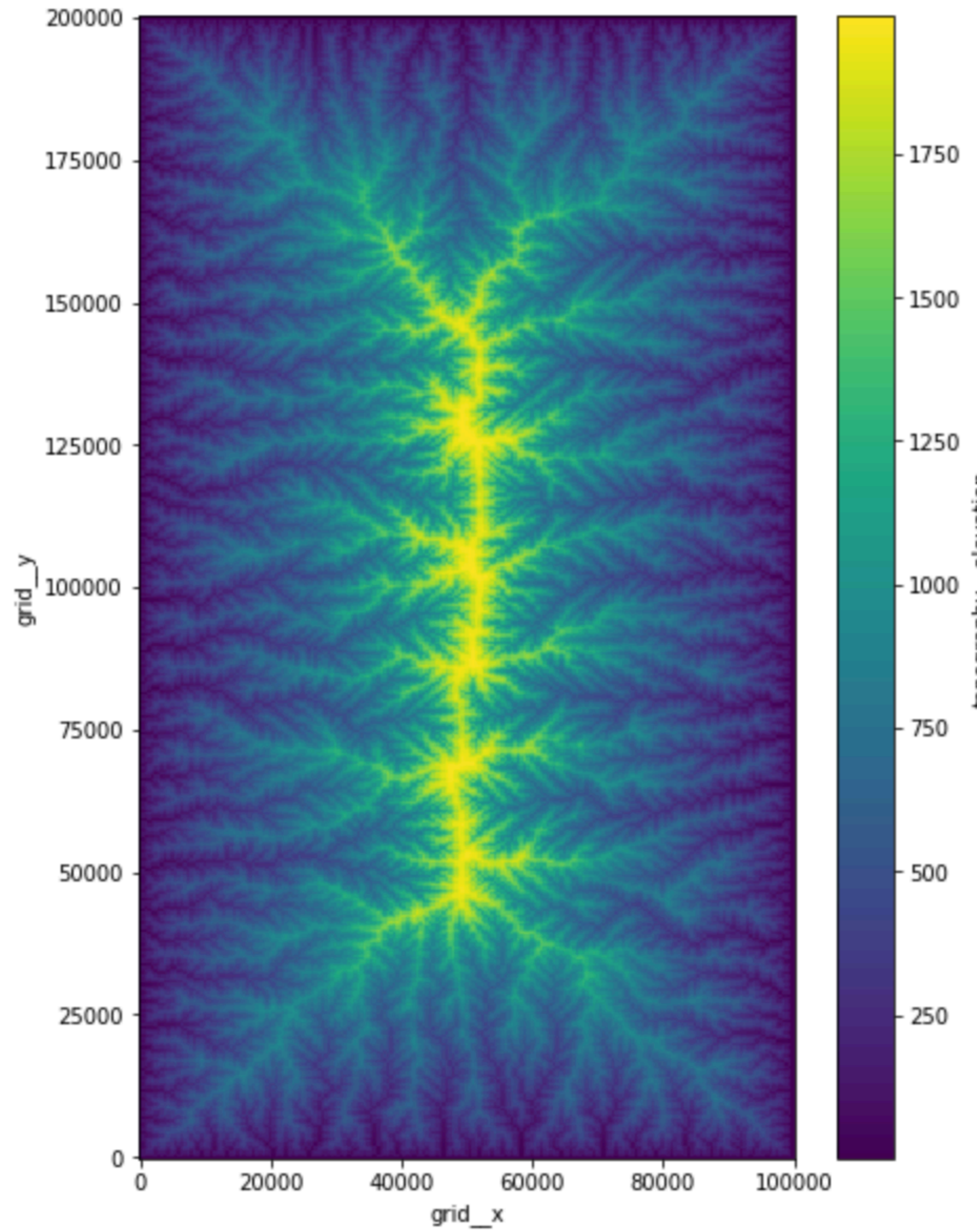


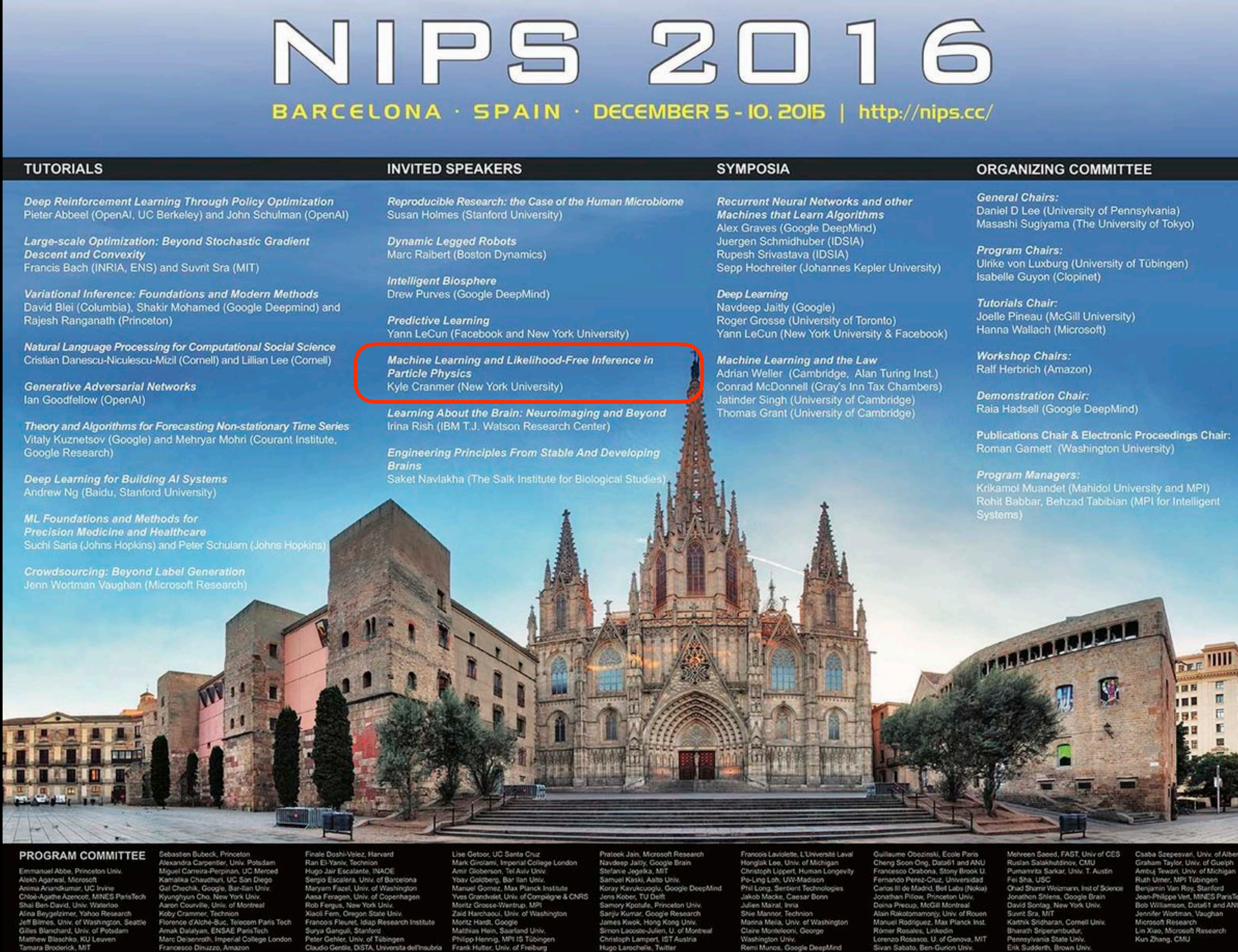
COMPUTATIONAL TOPOGRAPHY





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




NIPS 2016
BARCELONA · SPAIN · DECEMBER 5 - 10, 2016 | http://nips.cc/
TUTORIALS
Deep Reinforcement Learning Through Policy Optimization
Pieter Abbeel (OpenAI, UC Berkeley) and John Schulman (OpenAI)
Large-scale Optimization: Beyond Stochastic Gradient Descent and Convexity
Francis Bach (INRIA, ENS) and Suvrit Sra (MIT)
Variational Inference: Foundations and Modern Methods
David Blei (Columbia), Shakir Mohamed (Google Deepmind) and Rajesh Ranganath (Princeton)
Natural Language Processing for Computational Social Science
Cristian Danescu-Niculescu-Mizil (Cornell) and Lillian Lee (Cornell)
Generative Adversarial Networks
Ian Goodfellow (OpenAI)
Theory and Algorithms for Forecasting Non-stationary Time Series
Vitaly Kuznetsov (Google) and Mehryar Mohri (Courant Institute, Google Research)
Deep Learning for Building AI Systems
Andrew Ng (Baidu, Stanford University)
ML Foundations and Methods for Precision Medicine and Healthcare
Suchi Saria (Johns Hopkins) and Peter Schulam (Johns Hopkins)
Crowdsourcing: Beyond Label Generation
Jenn Wortman Vaughan (Microsoft Research)
INVITED SPEAKERS
Reproducible Research: the Case of the Human Microbiome
Susan Holmes (Stanford University)
Dynamic Legged Robots
Marc Raibert (Boston Dynamics)
Intelligent Biosphere
Drew Purves (Google DeepMind)
Predictive Learning
Yann LeCun (Facebook and New York University)
Machine Learning and Likelihood-Free Inference in Particle Physics
Kyle Cranmer (New York University)
Learning About the Brain: Neuroimaging and Beyond
Irina Rish (IBM T.J. Watson Research Center)
Engineering Principles From Stable And Developing Brains
Saket Navlakha (The Salk Institute for Biological Studies)
SYMPOSIA
Recurrent Neural Networks and other Machines that Learn Algorithms
Alex Graves (Google DeepMind)
Juergen Schmidhuber (IDSIA)
Rupesh Srivastava (IDSIA)
Sepp Hochreiter (Johannes Kepler University)
Deep Learning
Navdeep Jaitly (Google)
Roger Grosse (University of Toronto)
Yann LeCun (New York University & Facebook)
Machine Learning and the Law
Adrian Weller (Cambridge, Alan Turing Inst.)
Conrad McDonnell (Gray's Inn Tax Chambers)
Jatinder Singh (University of Cambridge)
Thomas Grant (University of Cambridge)
ORGANIZING COMMITTEE
General Chairs:
Daniel D Lee (University of Pennsylvania)
Masashi Sugiyama (The University of Tokyo)
Program Chairs:
Ulrike von Luxburg (University of Tübingen)
Isabelle Guyon (Clopinet)
Tutorials Chair:
Joelle Pineau (McGill University)
Hanna Wallach (Microsoft)
Workshop Chairs:
Ralf Herbrich (Amazon)
Demonstration Chair:
Raia Hadsell (Google DeepMind)
Publications Chair & Electronic Proceedings Chair:
Roman Garnett (Washington University)
Program Managers:
Krikamol Muandet (Mahidol University and MPI)
Rohit Babbar, Behzad Tabibian (MPI for Intelligent Systems)
PROGRAM COMMITTEE
Emmanuel Abbe, Princeton Univ.
Alekh Agarwal, Microsoft
Anima Anandkumar, UC Irvine
Chloé-Agathe Azencott, MINES ParisTech
Shai Ben-David, Univ. Waterloo
Alina Beygelzimer, Yahoo Research
Jeff Bilmes, Univ. of Washington, Seattle
Gilles Blanchard, Univ. of Potsdam
Matthew Blaschko, KU Leuven
Tamara Broderick, MIT
Sebastien Bubeck, Princeton
Alexandra Carpentier, Univ. Potsdam
Miguel Carreira-Perpinan, UC Merced
Kamalika Chaudhuri, UC San Diego
Gal Chechik, Google, Bar-Ilan Univ.
Kyunghyun Cho, New York Univ.
Aaron Courville, Univ. of Montreal
Koby Crammer, Technion
Florence d'Alché-Buc, Telecom Paris Tech
Arnak Dalalyan, ENSAE ParisTech
Marc Deisenroth, Imperial College London
Francesco Dinuzzo, Amazon
Finale Doshi-Velez, Harvard
Ran El-Yaniv, Technion
Hugo Jair Escalante, INAOE
Sergio Escalera, Univ. of Barcelona
Maryam Fazel, Univ. of Washington
Aasa Feragen, Univ. of Copenhagen
Rob Fergus, New York Univ.
Xiaoli Fern, Oregon State Univ.
Francois Fleuret, Idiap Research Institute
Surya Ganguli, Stanford
Peter Gehler, Univ. of Tübingen
Claudio Gentile, DiSTA, Universita dell'Insubria
Lise Getoor, UC Santa Cruz
Mark Girolami, Imperial College London
Amir Globerson, Tel Aviv Univ.
Yoav Goldberg, Bar Ilan Univ.
Manuel Gomez, Max Planck Institute
Yves Grandvalet, Univ. of Compiègne & CNRS
Moritz Grosse-Wentrup, MPI
Zaid Harchaoui, Univ. of Washington
Moritz Hardt, Google
Matthias Hein, Saarland Univ.
Philipp Hennig, MPI IS Tübingen
Frank Hutter, Univ. of Freiburg
Prateek Jain, Microsoft Research
Navdeep Jaitly, Google Brain
Stefanie Jegelka, MIT
Samuel Kaski, Aalto Univ.
Koray Kavukcuoglu, Google DeepMind
Jens Kober, TU Delft
Samory Kpotufe, Princeton Univ.
Sanjiv Kumar, Google Research
James Kwok, Hong Kong Univ.
Simon Lacoste-Julien, U. of Montreal
Christoph Lampert, IST Austria
Hugo Larochelle, Twitter
Francois Laviolette, L'Université Laval
Honglak Lee, Univ. of Michigan
Christoph Lippert, Human Longevity
Po-Ling Loh, UW-Madison
Phil Long, Sentient Technologies
Jakob Macke, Caesar Bonn
Julien Mairal, Inria
Shie Mannor, Technion
Marina Meila, Univ. of Washington
Claire Monteleoni, George Washington Univ.
Remi Munos, Google DeepMind
Guillaume Obozinski, Ecole Paris
Cheng Soon Ong, Data61 and ANU
Francesco Orabona, Stony Brook U.
Fernando Perez-Cruz, Universidad Carlos III de Madrid, Bell Labs (Nokia)
Jonathan Pillow, Princeton Univ.
Doina Precup, McGill Montreal
Alain Rakotomamonjy, Univ. of Rouen
Manuel Rodriguez, Max Planck Inst.
Rómer Rosales, Linkedin
Lorenzo Rosasco, U. of Genova, MIT
Sivan Sabato, Ben-Gurion Univ.
Mehreen Saeed, FAST, Univ of CES
Ruslan Salakhutdinov, CMU
Purnamrita Sarkar, Univ. T. Austin
Fei Sha, USC
Ohad Shamir Weizmann, Inst of Science
Jonathon Shlens, Google Brain
David Sontag, New York Univ.
Suvrit Sra, MIT
Karthik Sridharan, Cornell Univ.
Bharath Sriperumbudur, Pennsylvania State Univ.
Erik Sudderth, Brown Univ.
Csaba Szepesvari, Univ. of Alberta
Graham Taylor, Univ. of Guelph
Ambuj Tewari, Univ. of Michigan
Ruth Urner, MPI Tubingen
Benjamin Van Roy, Stanford
Jean-Philippe Vert, MINES ParisTech
Bob Williamson, Data61 and ANU
Jennifer Wortman, Vaughan Microsoft Research
Lin Xiao, Microsoft Research
Kun Zhang, CMU

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.

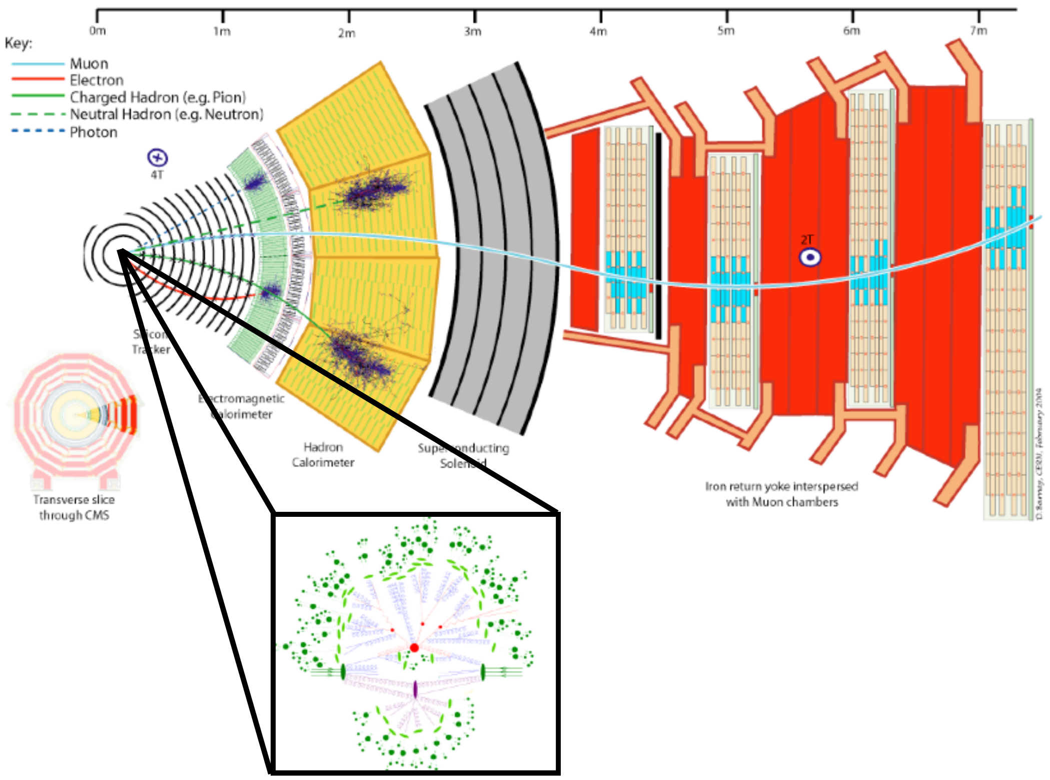
Of particular interest at this workshop is to unite fields that work on implicit models. For example:

- **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.

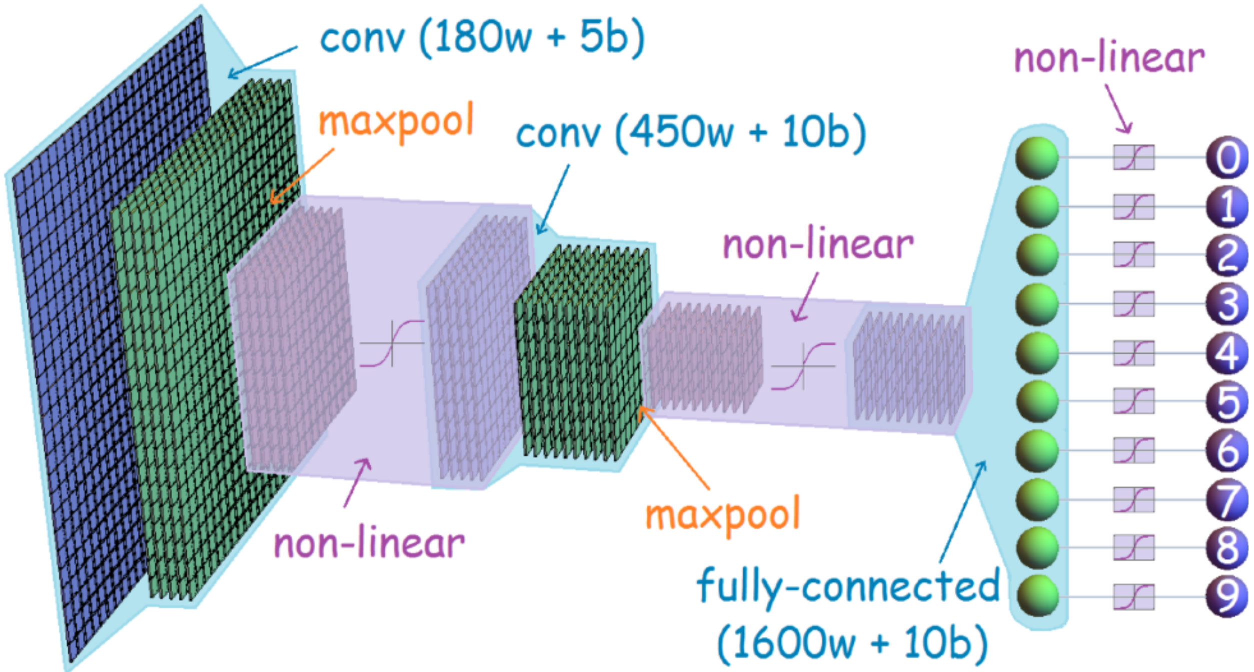
TWO APPROACHES

Use simulator
(much more efficiently)



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

Learn simulator
(with deep learning)

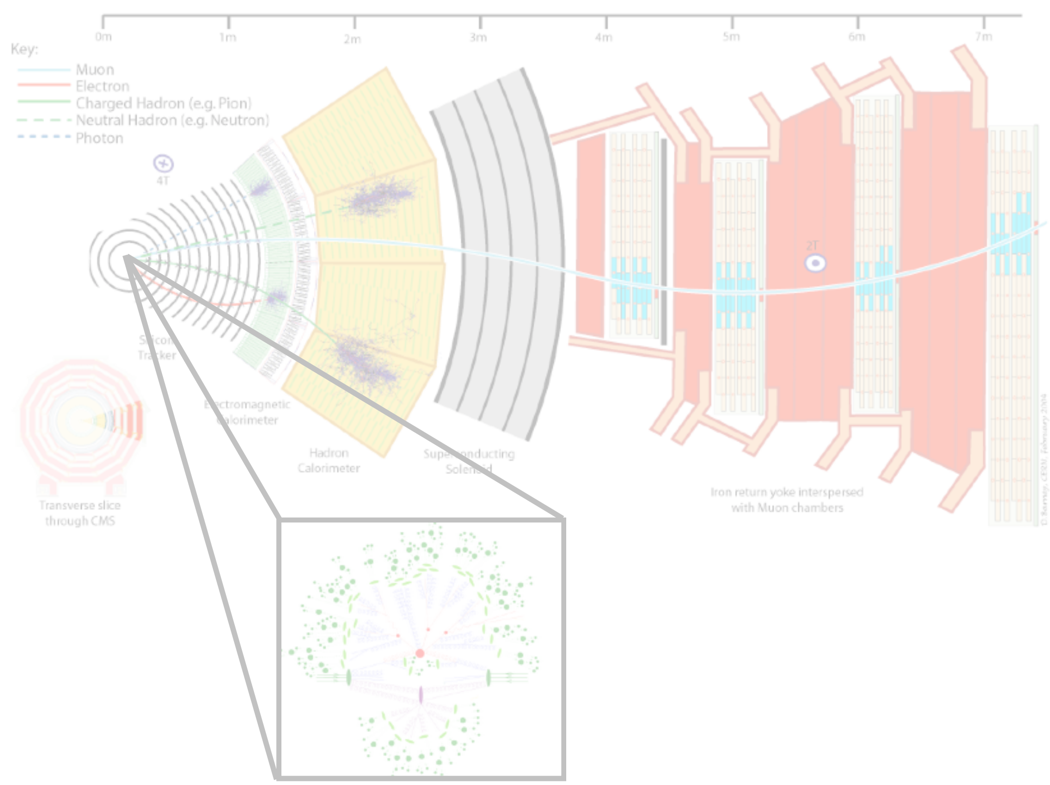


- Generative Adversarial Networks (GANs), Variational Auto-Encoders (VAE)
- Likelihood ratio from classifiers (CARL)
- Autogregressive models, Normalizing Flows

[image credit: A.P. Goucher]

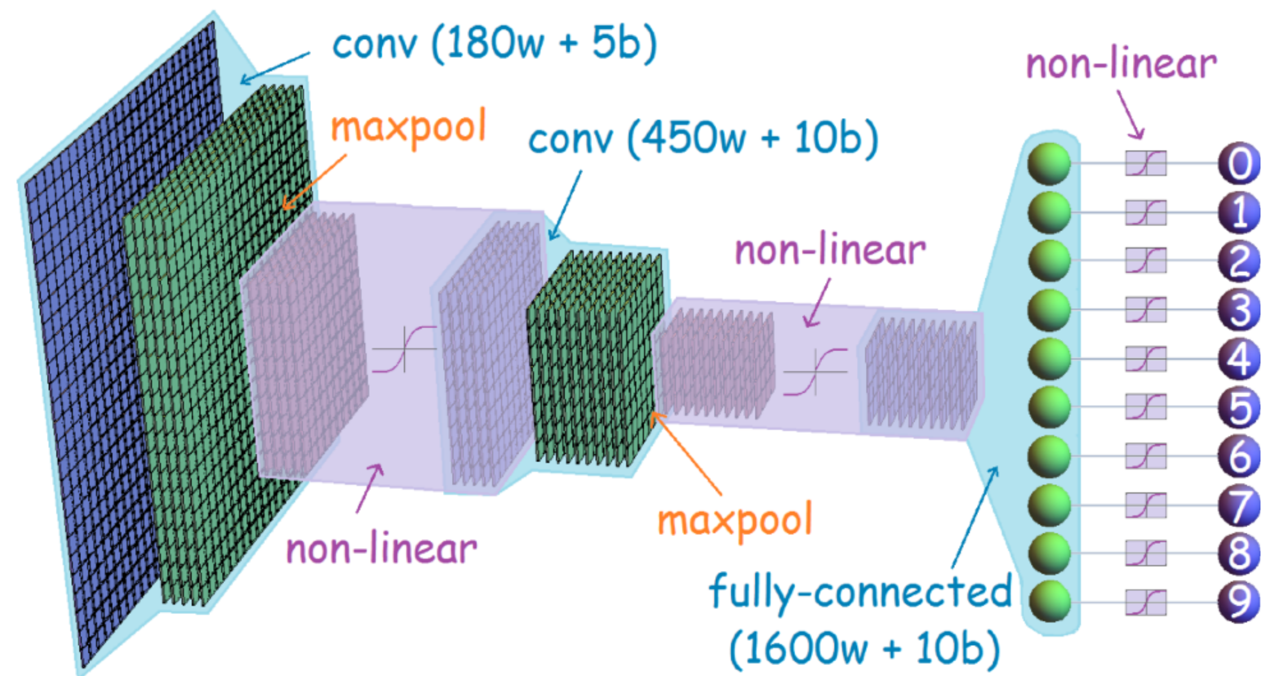
TWO APPROACHES

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

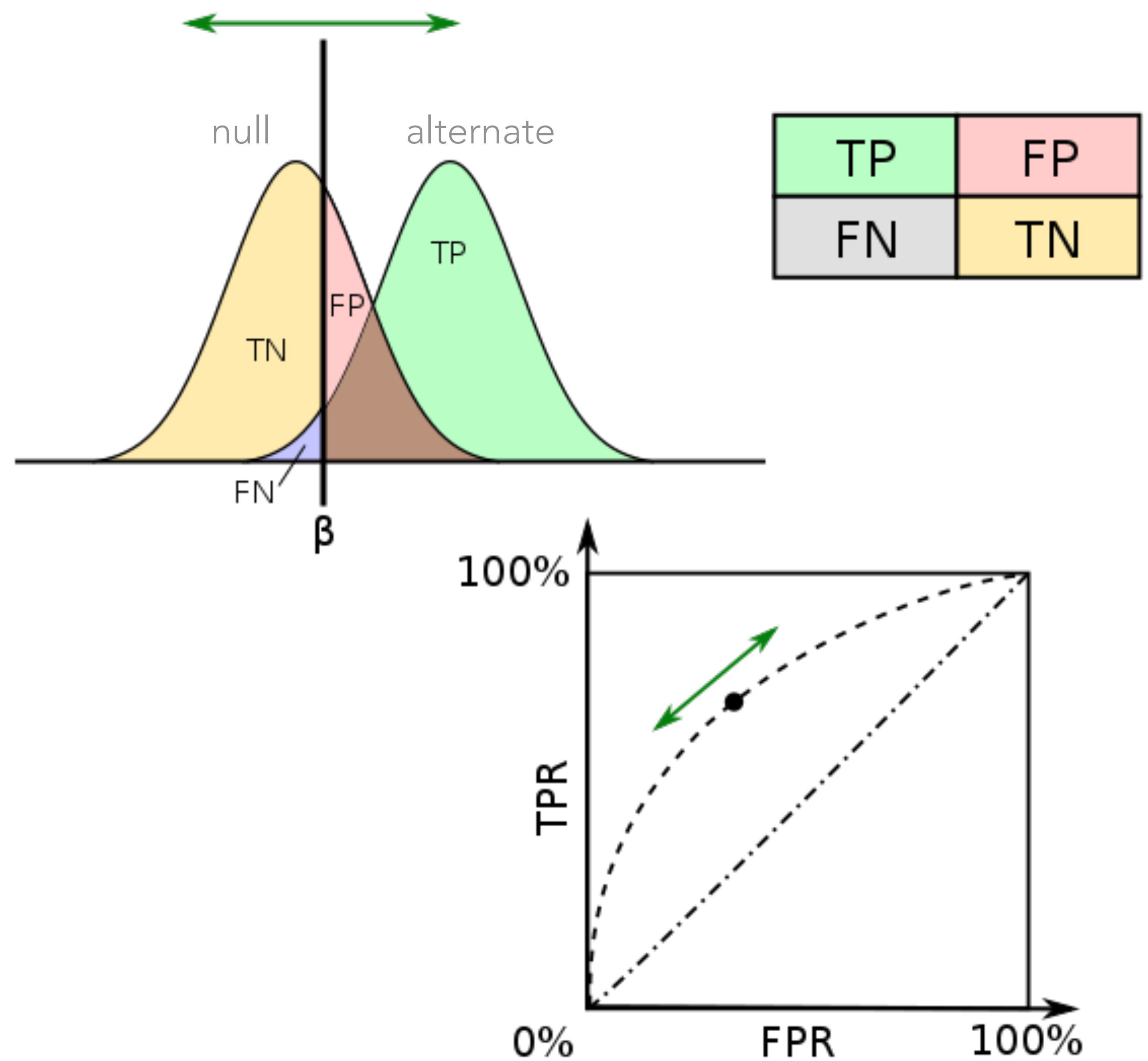
[image credit: A.P. Goucher]

Likelihood-Free Warm-up

Hypothesis Testing & Classification

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

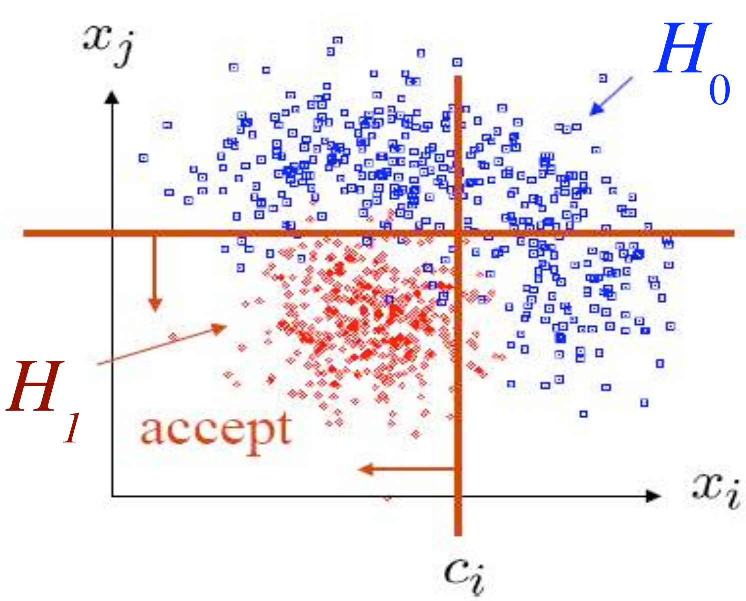
		Actual condition	
		Guilty	**Not guilty**
Decision	**Verdict of 'guilty'**	True Positive **power**	False Positive (i.e. guilt reported unfairly) **Type I error**
	Verdict of 'not guilty'	False Negative (i.e. guilt not detected) **Type II error**	True Negative

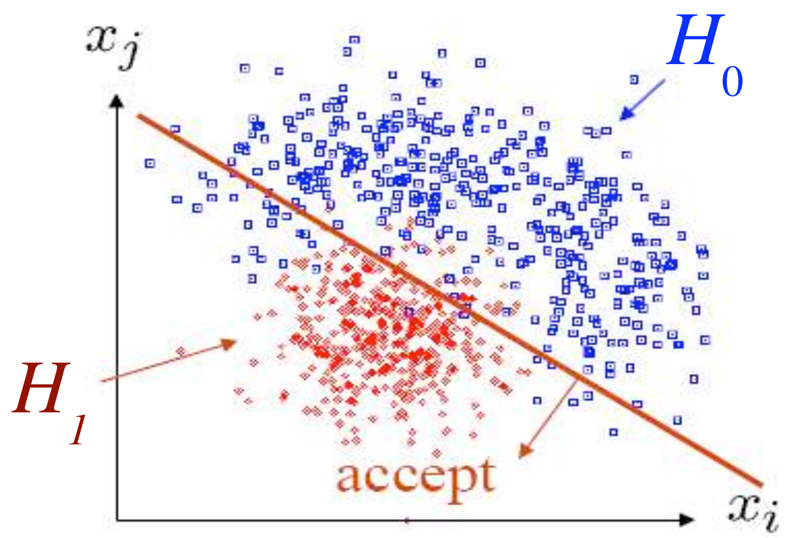


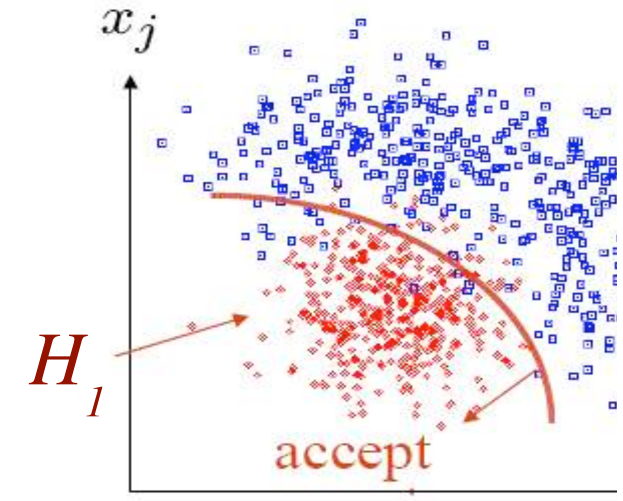
actually guilty ↔ new physics

verdict guilty ↔ claim discovery

If the data are high-dimensional, it's not obvious how to draw the boundary between accept/reject the null hypothesis

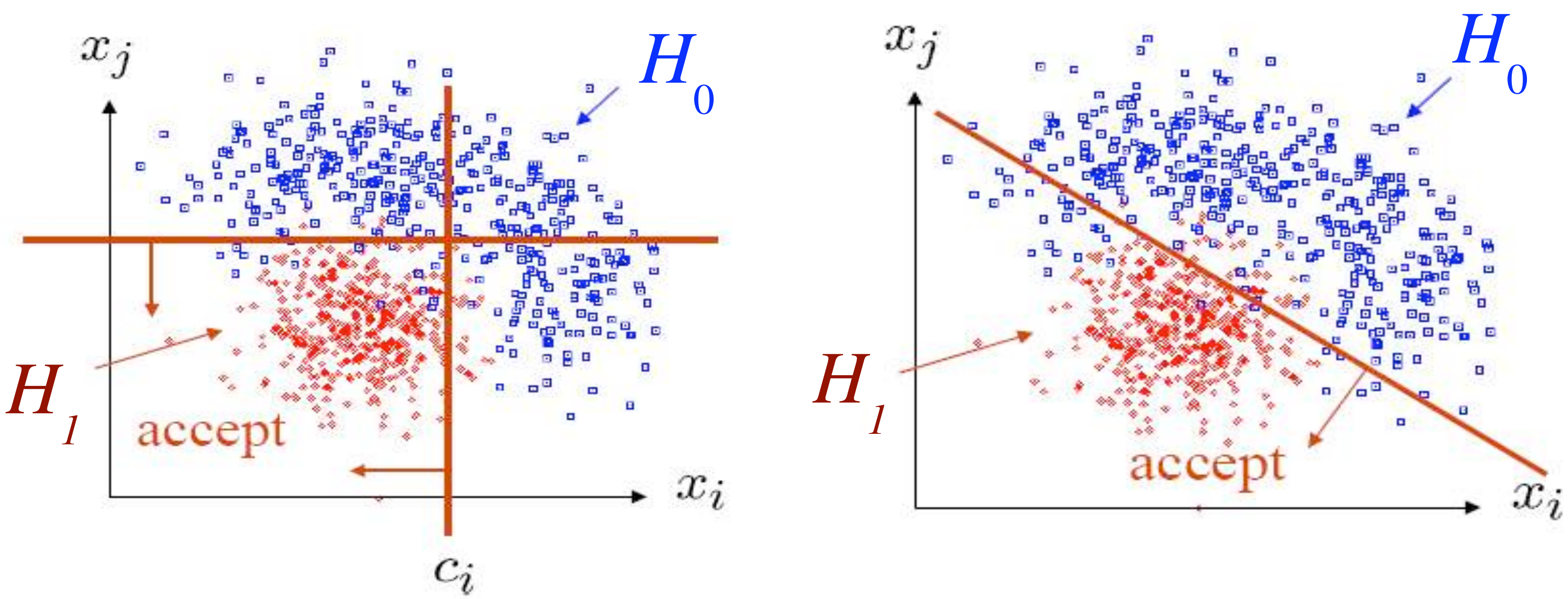


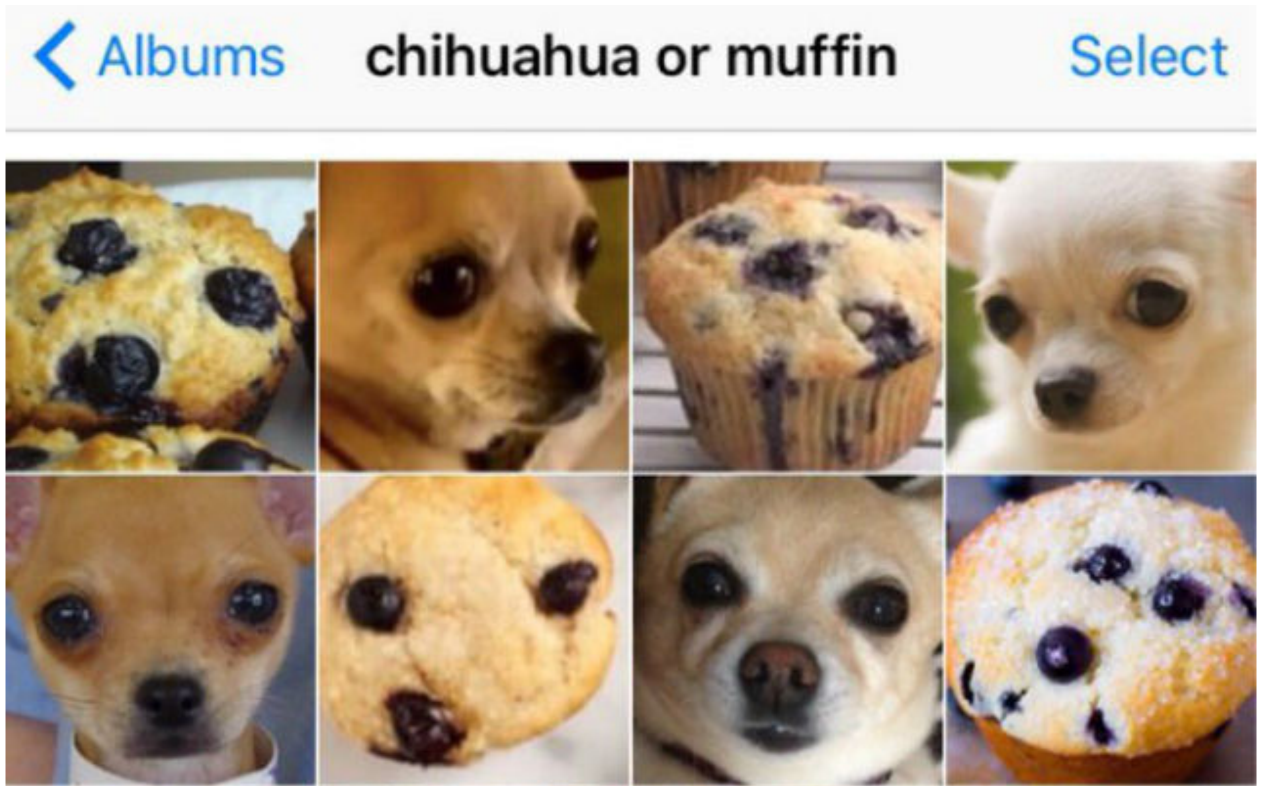




HYPOTHESIS TESTING

If the data are high-dimensional, it's not obvious how to draw the boundary between accept/reject the null hypothesis





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

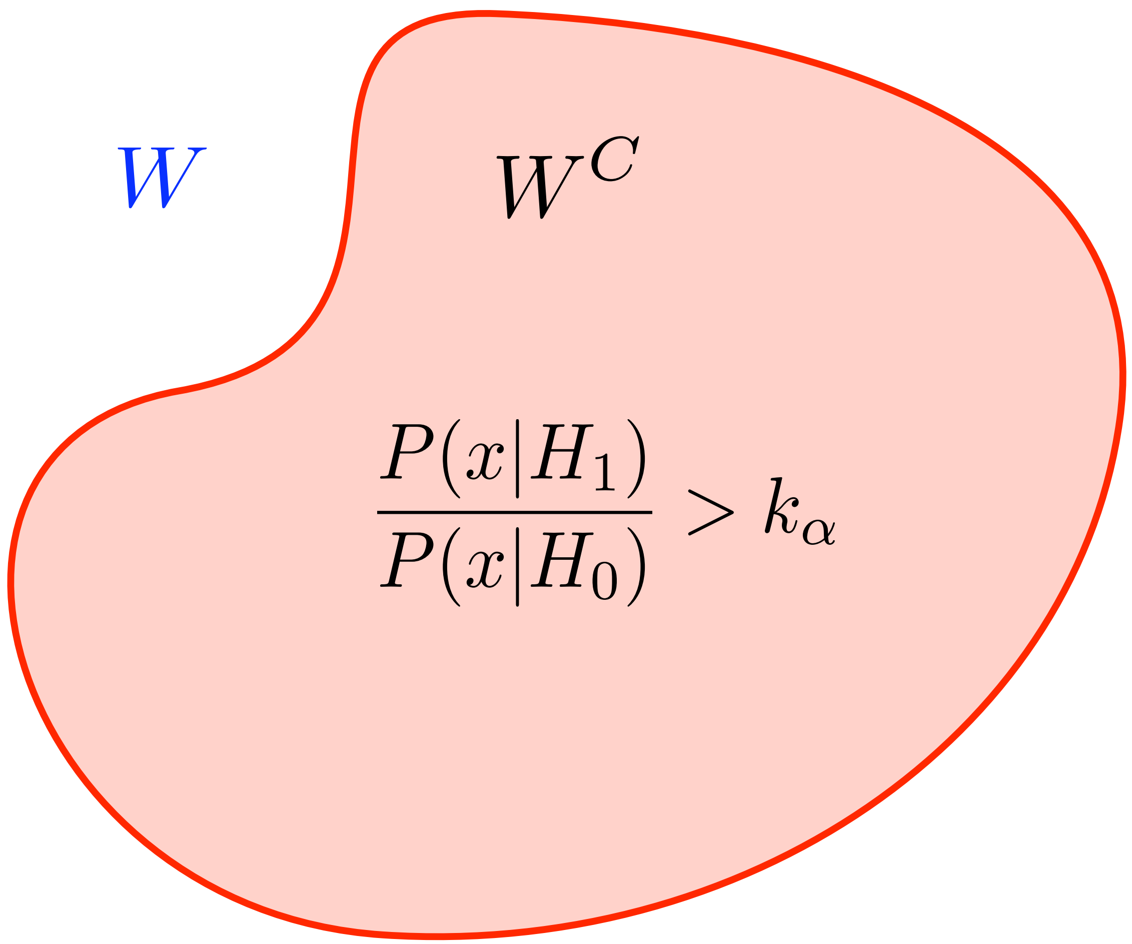
$$\alpha = P(x \notin W | H_0)$$

(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)

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

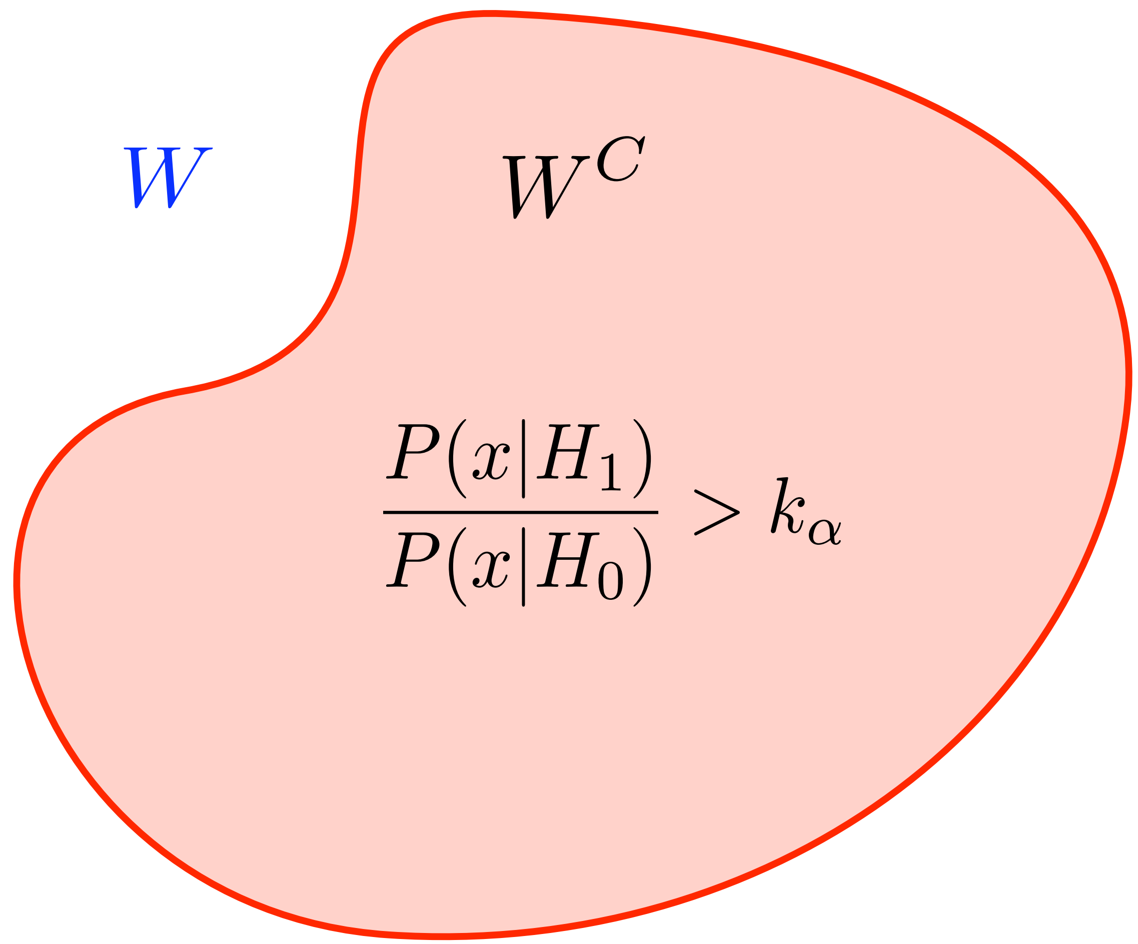
THE NEYMAN-PEARSON LEMMA



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

Any other region of the same size will have less power

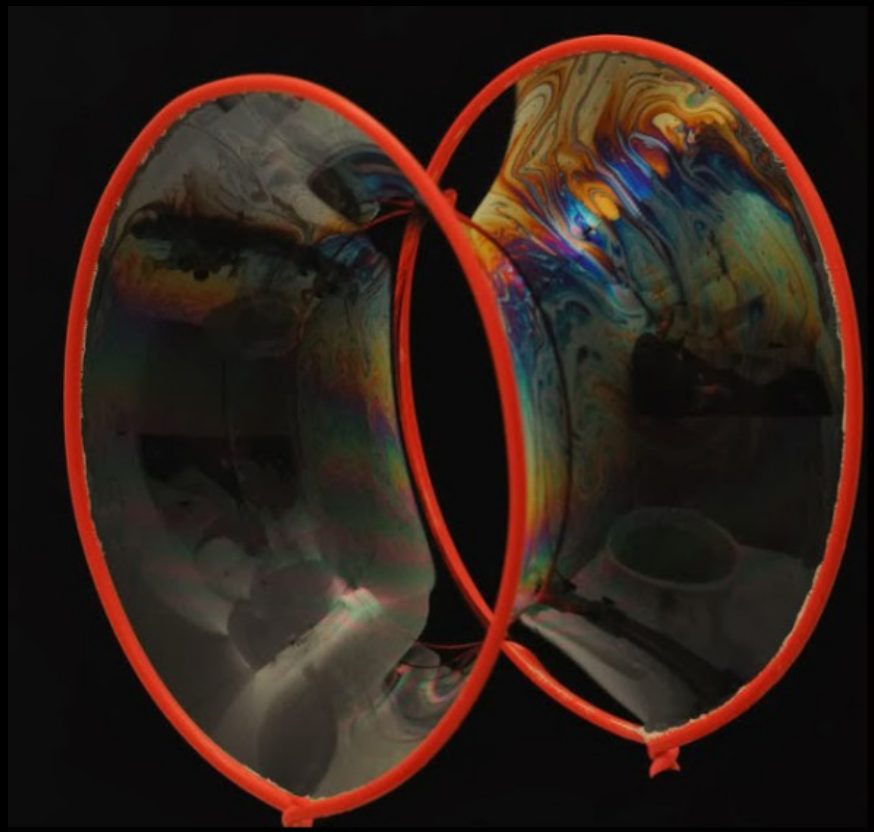
Problem with Neyman-Pearson

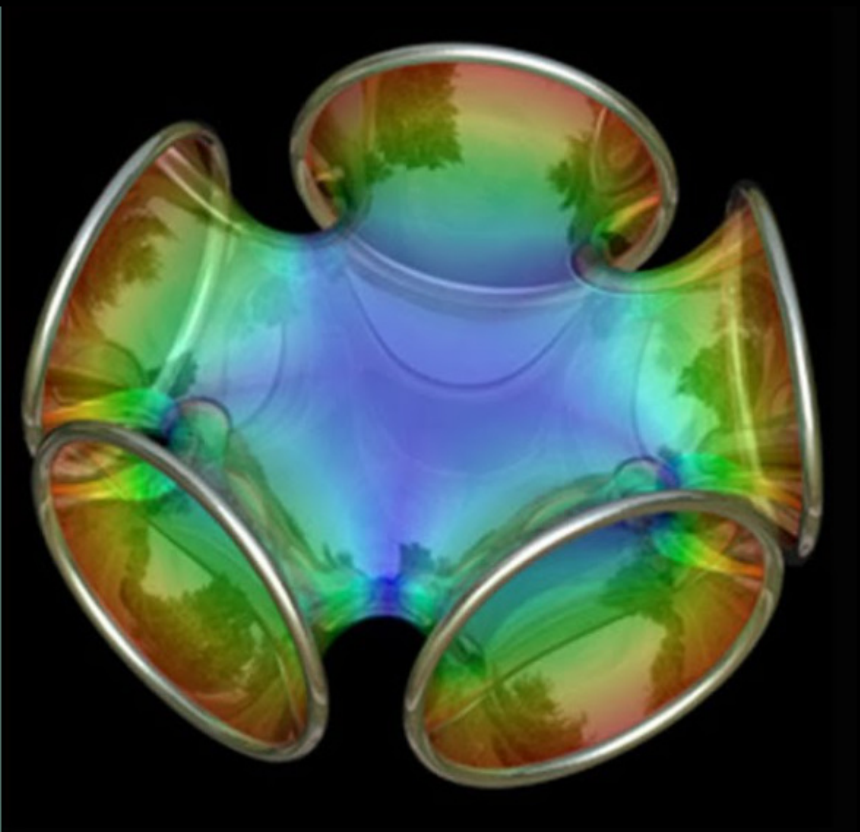


But, If I don't know $P(x|H_1)$ and $P(x|H_0)$

I can't evaluate this likelihood ratio!

Machine Learning = Applied Calculus of Variations





MACHINE LEARNING = APPLIED CALCULUS OF VARIATIONS

Kyle Cranmer added **3 new photos** — with **Sarah Demers Konezny** and **Paul Tipton**.
April 20, 2016 · New Haven, CT ·

Seminar at Yale today. Felt good to talk about new ideas... Equally confusing for theorists and experimentalists 😜

Machine Learning = Applied Calculus of Variations

2 Deriving BP using the Hamiltonian/Lagrangian formalism

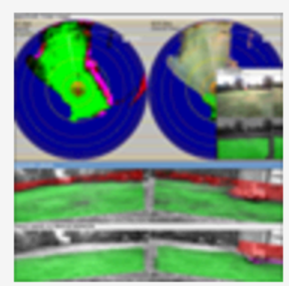
2.1 Notations

For the sake of clarity, we will introduce the formalism in a simple case. A more general formulation will be presented afterwards. It will be assumed that the network is composed of a number of layers connected in a feed-forward manner. Furthermore, we make the assumption that connections cannot skip layers. These assumptions can be easily relaxed [le Cun, 1987].

Yann LeCun Deep learning = calculus of variations

Backprop is like the Langrangian formulation of classical mechanics.

Y. LeCun: A theoretical framework for Back-Propagation, in Touretzky, D. and Hinton, G. and Sejnowski, T. (Eds), Proceedings of the 1988 Connectionist Models Summer School, 21-28, Morgan Kaufmann, CMU, Pittsburgh, Pa, 1988.
http://yann.lecun.com/exdb/publis/index.html#lecun-88



[bib2web] Yann LeCun's Publications

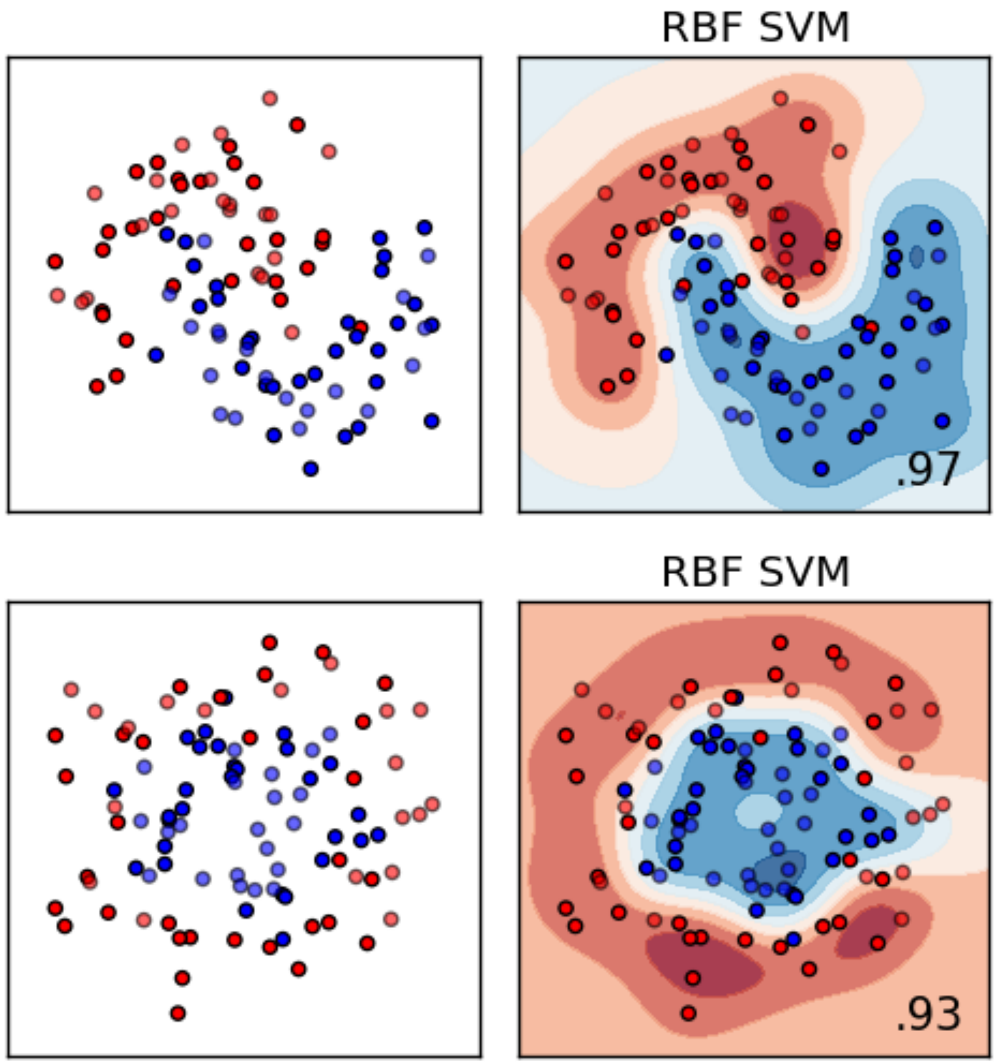
YANN.LECUN.COM

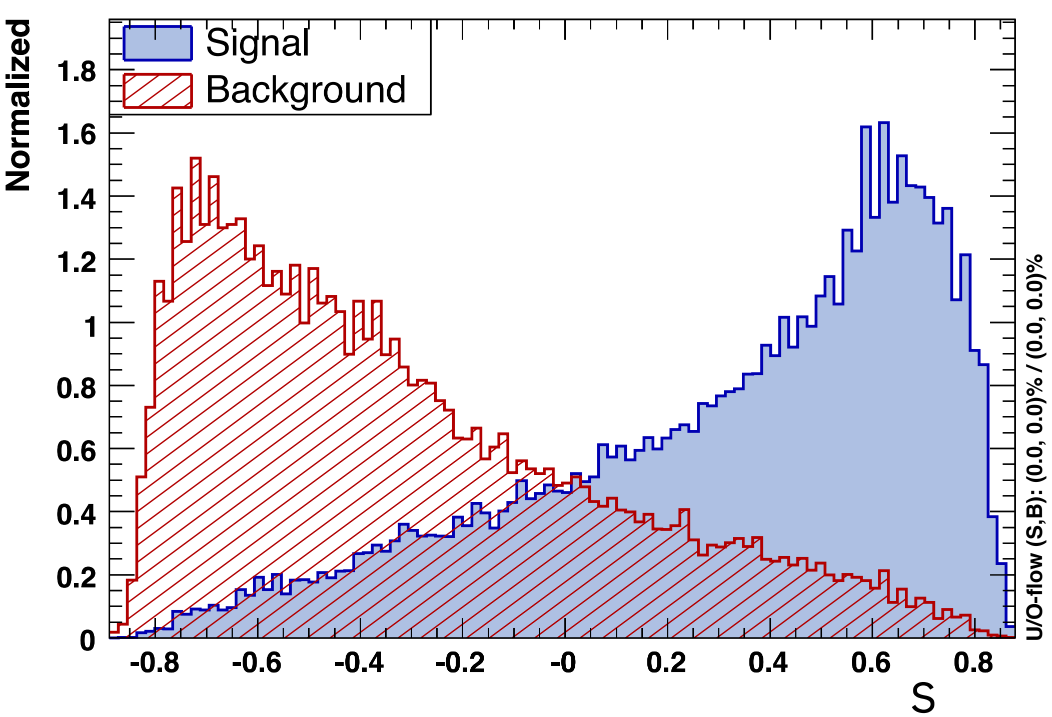
Like · Reply · Remove Preview · 2 · April 20, 2016 at 2:30am

Kyle Cranmer I guess this counts as an endorsement for this point of view 😃

Many physicists (particularly theoretical ones) are skeptical of machine learning because it usually is explained to them in some ad hoc way (neurons, etc). But minimizing a loss function(al) is much more palatable.

Like · Reply · 2 · April 20, 2016 at 2:39am · Edited

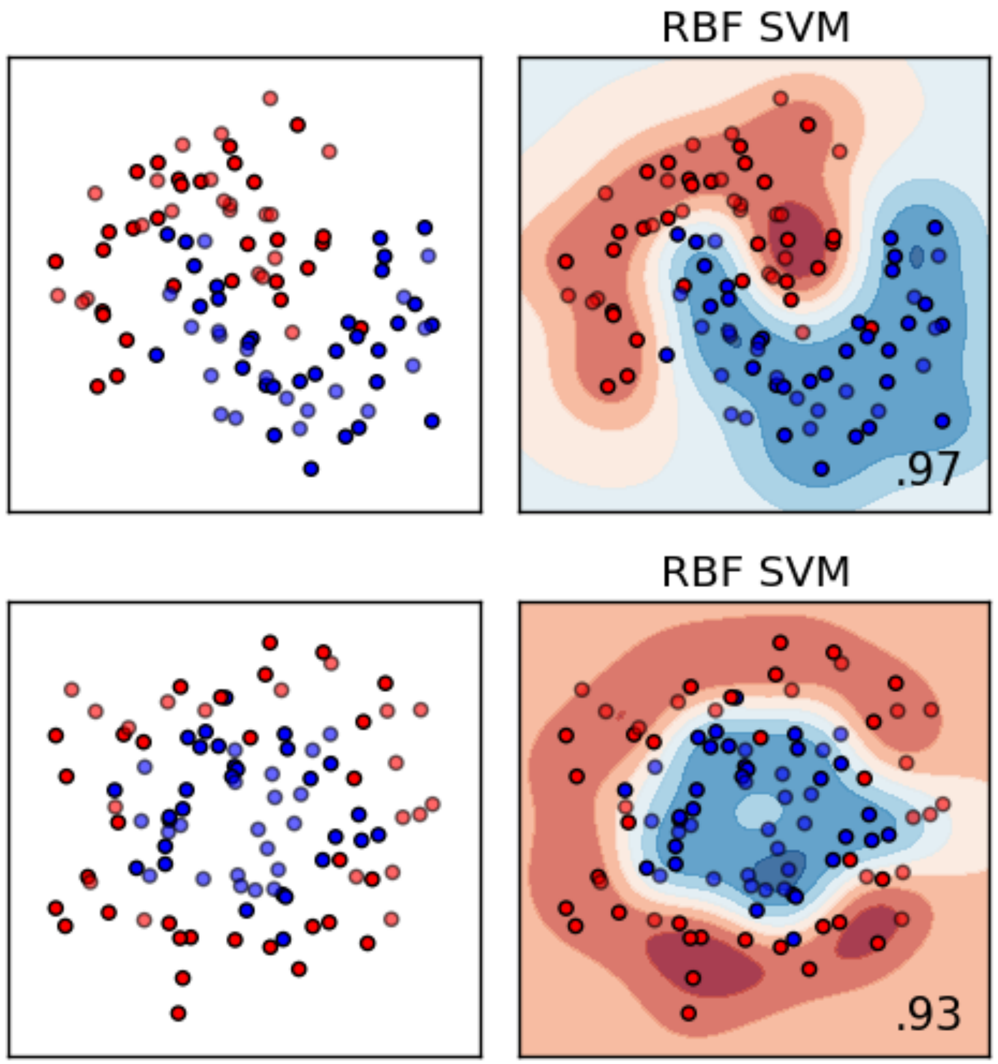


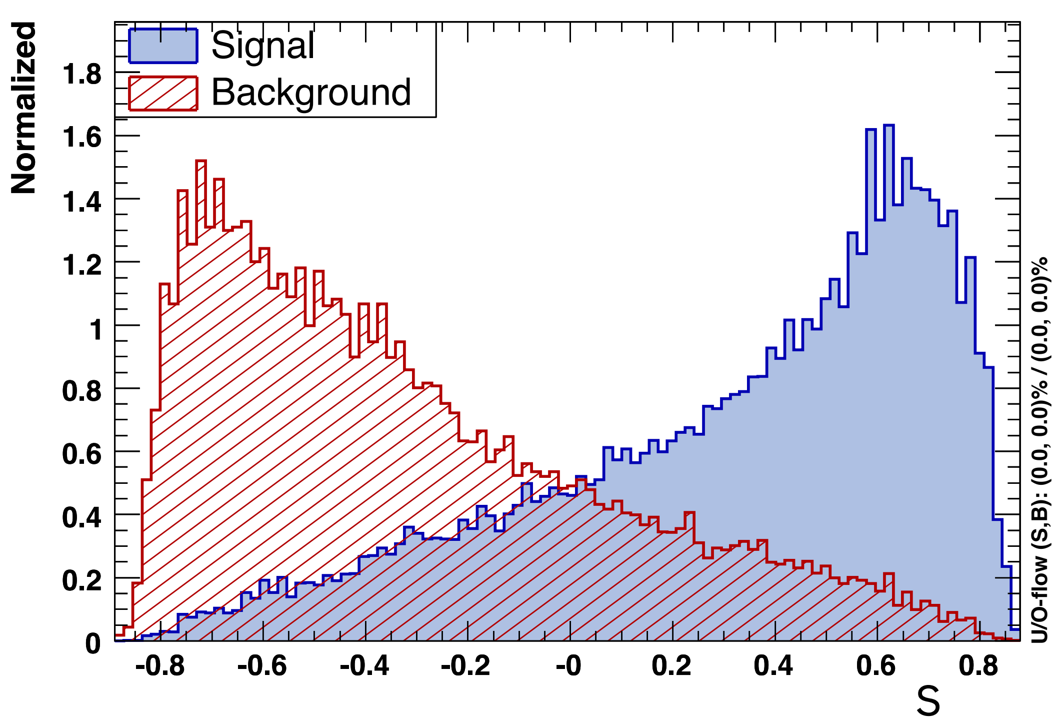


Common to use machine learning classifiers to separate signal (H_1) vs. background (H_0)

- want a function s: X→ Y that maps signal to y=1 and background to y=0
- **calculus of variations**: find function s(x) that minimizes ***loss***:

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





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

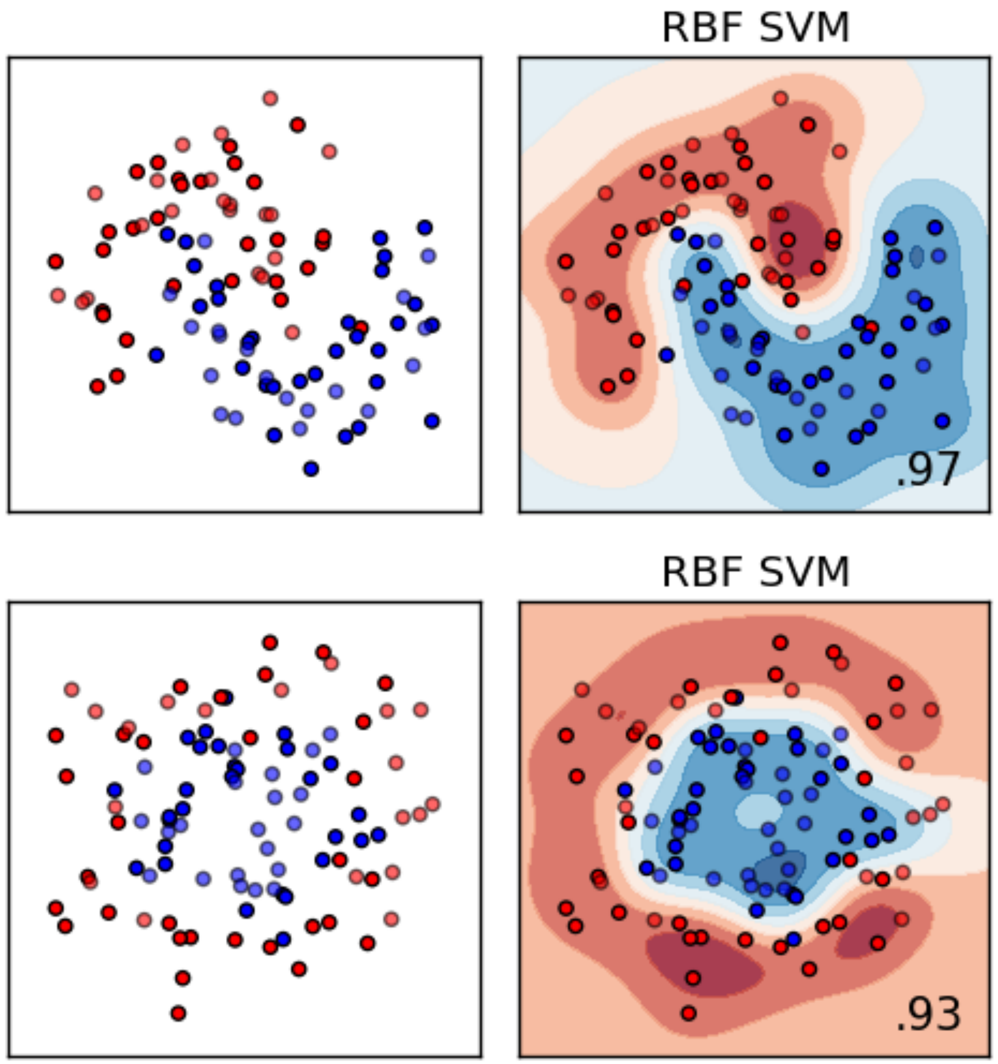
$$L[s] = \int p(x|H_0)\,(0 - s(x))^2\,dx + \int p(x|H_1)\,(1 - s(x))^2 dx$$

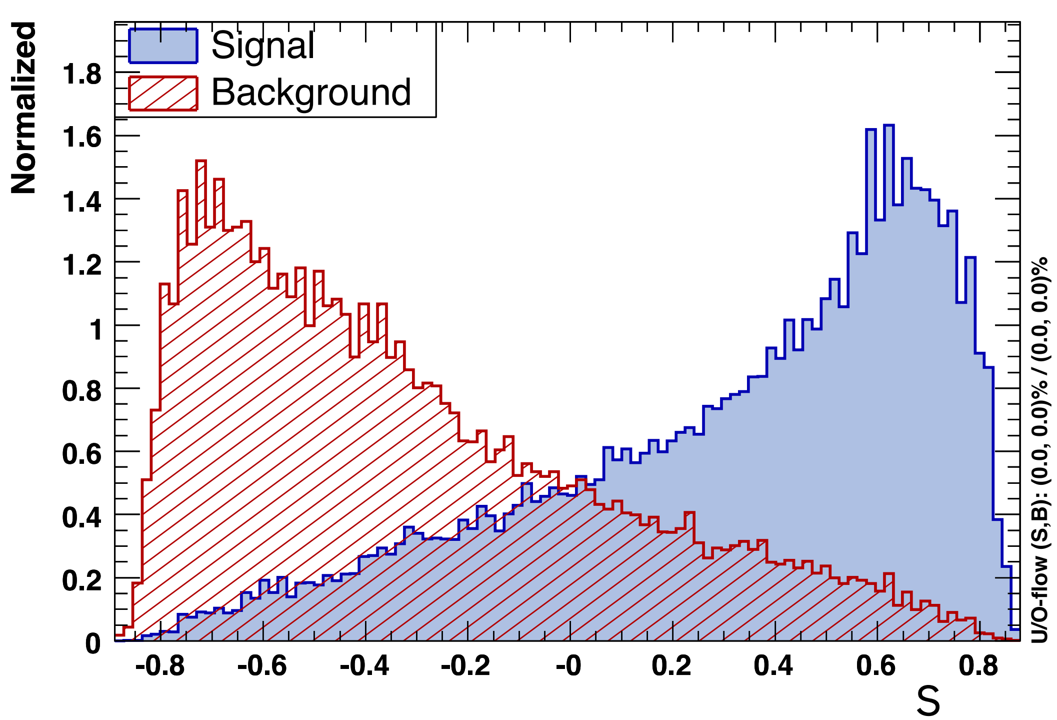
- i.e. approximate the optimal classifier

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

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

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





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

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

$$\approx \frac{1}{N}\sum_{i=1}^{N}(y_i - s(x_i))^2$$

- i.e. approximate the optimal classifier

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

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

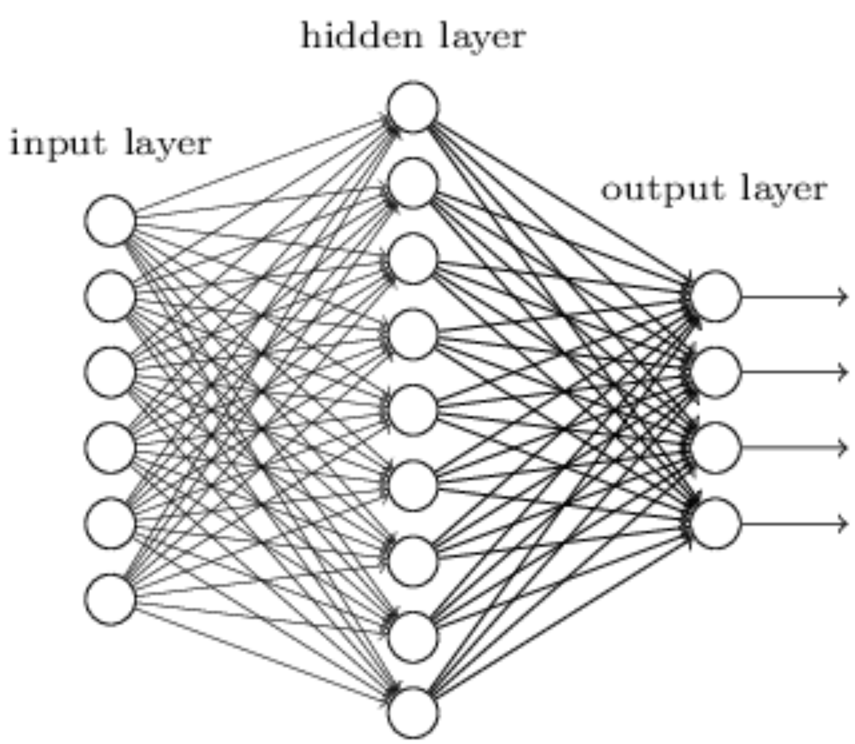
$$\frac{p(x|H_1)}{p(x|H_0)}$$

NN = A HIGHLY FLEXIBLE FAMILY OF FUNCTIONS

In calculus of variations, the optimization is over all functions: $\hat{s} = \underset{s}{\operatorname{argmin}}\, L[s]$

- In applied calculus of variations, we consider a highly flexible family of functions s_ϕ and optimize: i.e. $\hat{\phi} = \underset{\phi}{\operatorname{argmin}}\, L[s_\phi]$ and $\hat{s} \approx s_{\hat{\phi}}$
- Think of neural networks as a highly flexible family of functions
- Machine learning also includes non-convex optimization algorithms that are effective even with millions of parameters!

Shallow neural network



Deep neural network

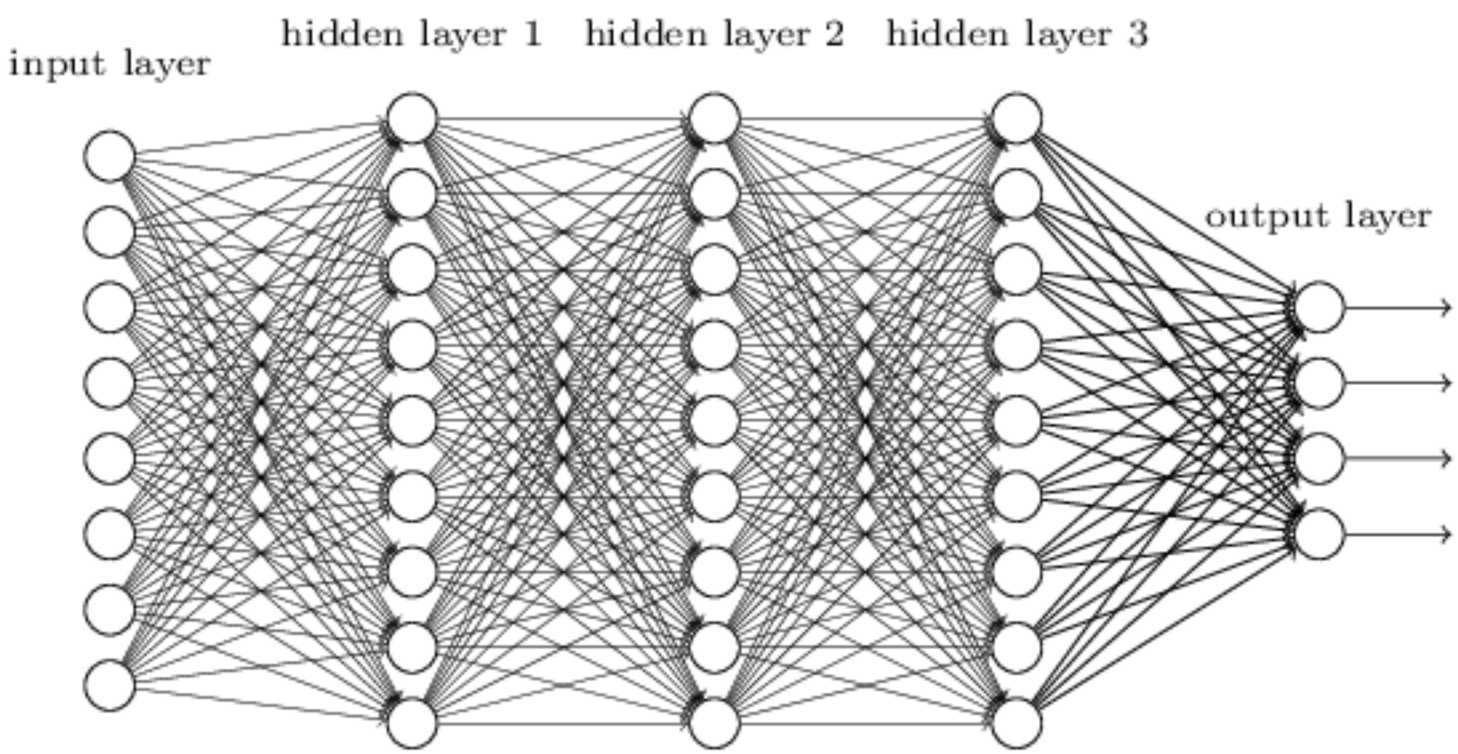
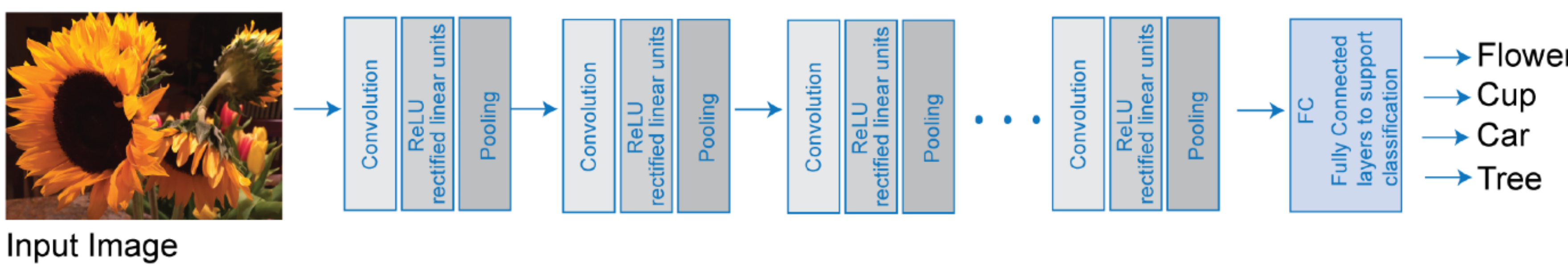


image credit: Michael Nielsen

Variational family should take advantage of domain knowledge

- the world is compositional ⇒ hierarchical architecture
- images are translationally invariant ⇒ shared weights



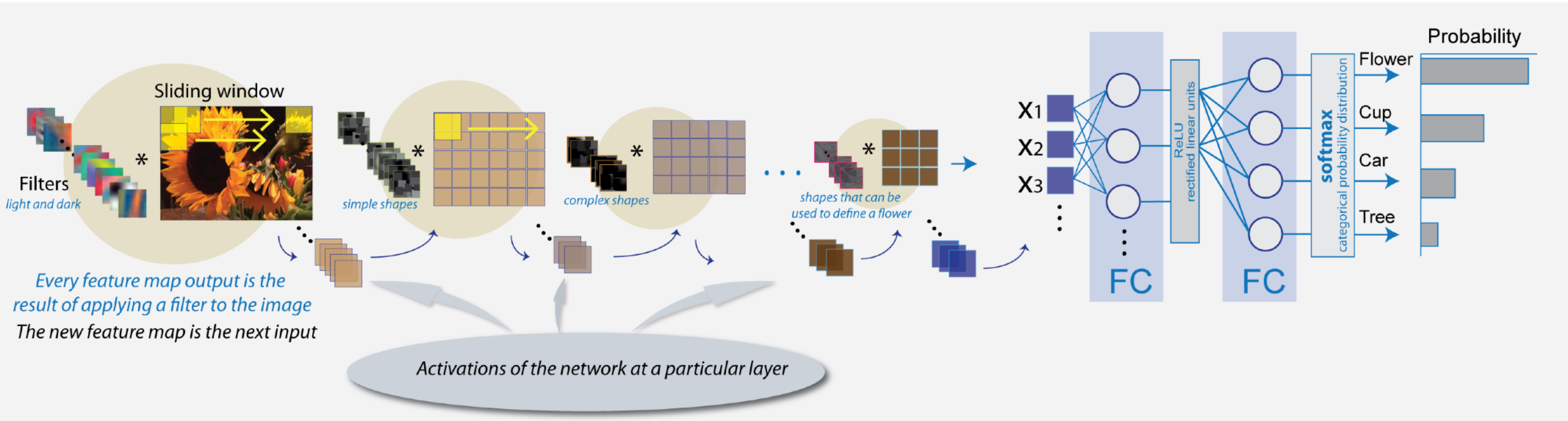


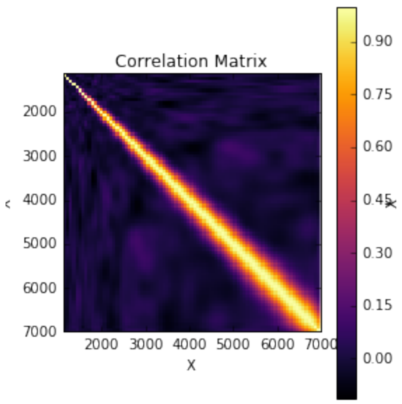
image credit: MathWorks

PHYSICS-AWARE MACHINE LEARNING

We can inject our knowledge of physics into the variational family

Physics-aware Gaussian Processes

arXiv:1709.05681

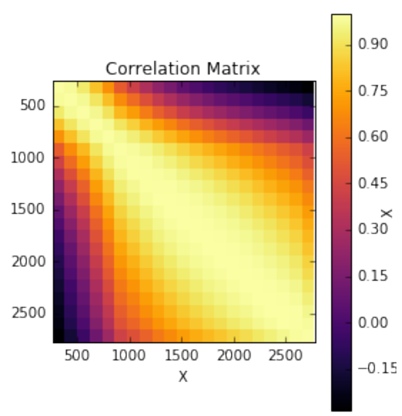


Final Kernel =

Poisson fluctuations

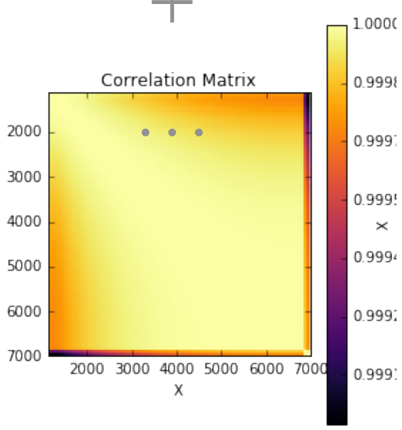
=

+ Mass Resolution



+ Parton Density Functions

+

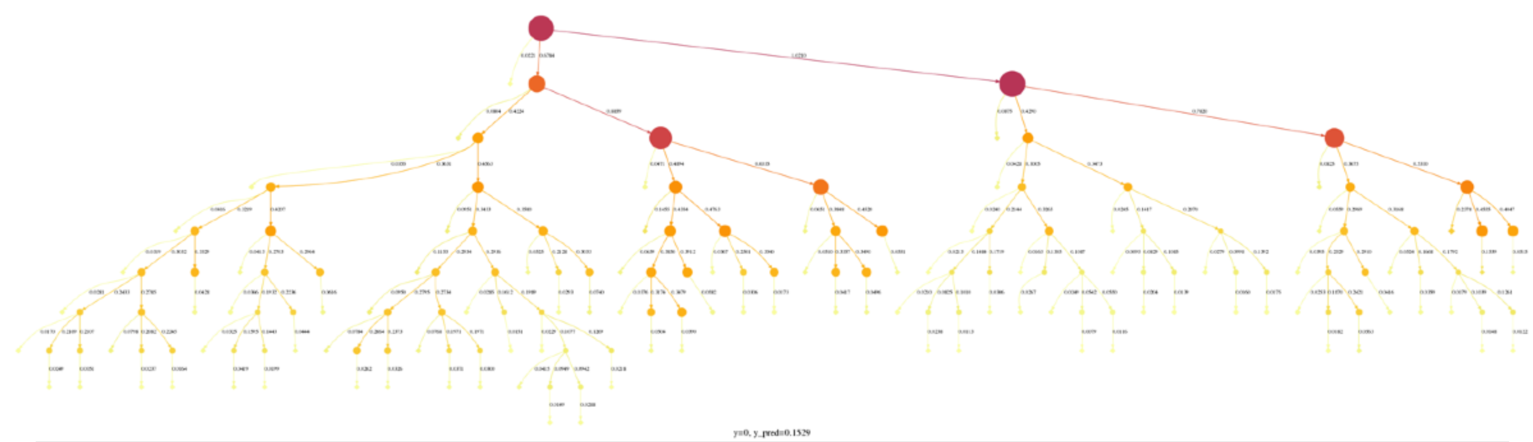


+ Jet Energy Scale

+

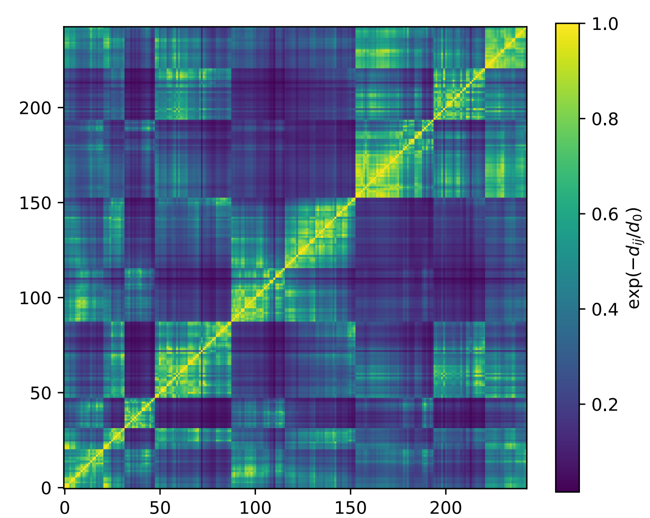
QCD-Aware recursive neural networks

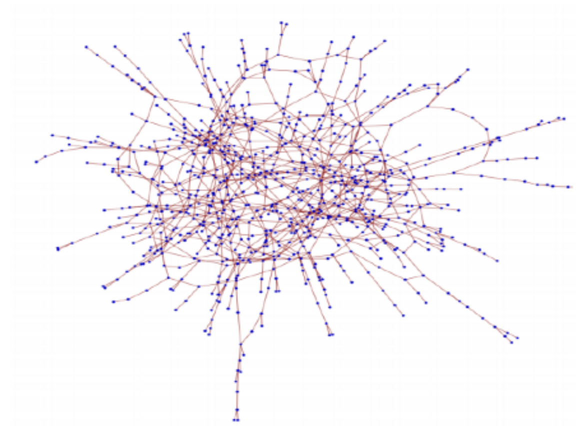
arXiv:1702.00748



QCD-Aware graph convolutional neural networks

NIPS2017 workshop





$$d^{\alpha}_{ii'} = \min(p_{ti}^{2\alpha}, p_{ti'}^{2\alpha})\frac{\Delta R^2_{ii'}}{R^2}$$

Likelihood-Free Inference

&

Inverse Problems

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

We showed a binary classifier approximates

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

Which is one-to-one with the likelihood ratio

$$\frac{p(x|H_1)}{p(x|H_0)} = 1 - \frac{1}{s(x)}$$

Can do the same thing for any two points θ_0 & θ_1 in parameter space $\boldsymbol{\Theta}$. I call this a **parametrized classifier**

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

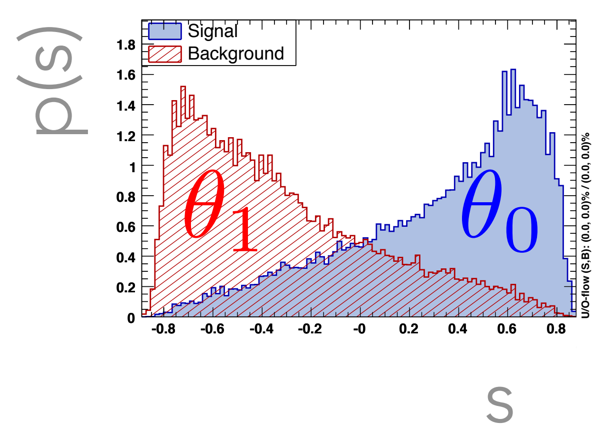
K.C., G. Louppe, J. Pavez: http://arxiv.org/abs/1506.02169

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

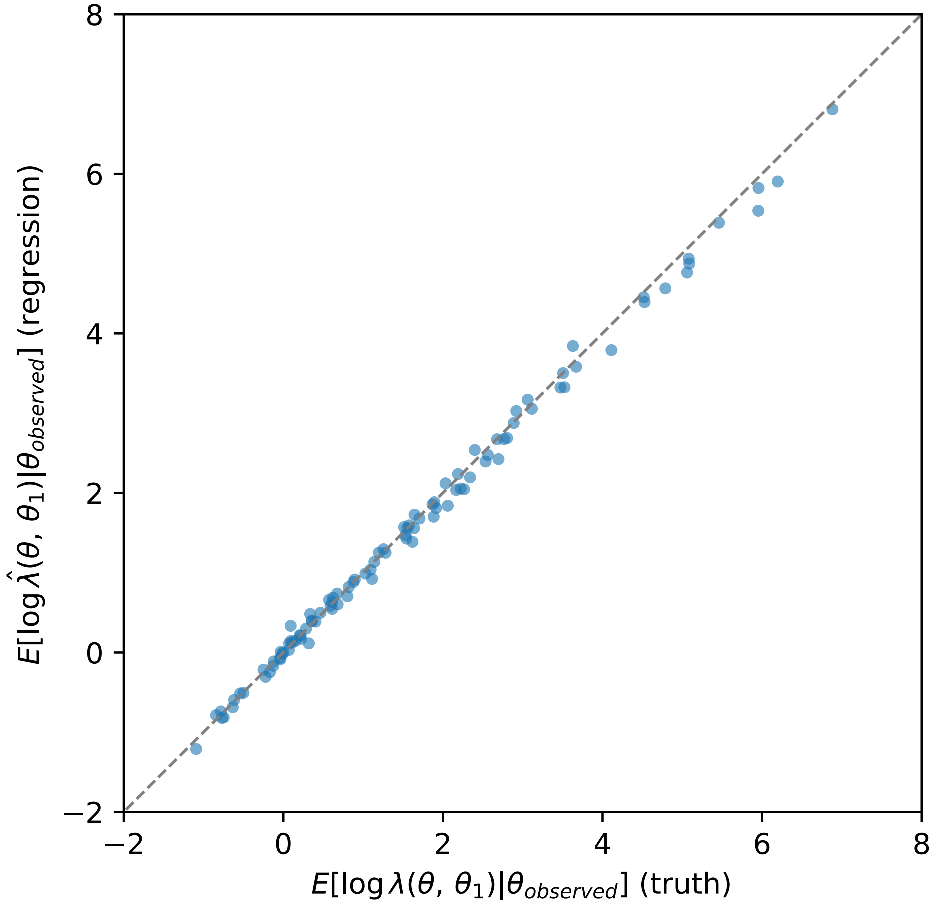
$$s(x;\theta_0;\theta_1) = \text{monotonic}[\ p(x|\theta_0)/p(x|\theta_1)\ ]$$

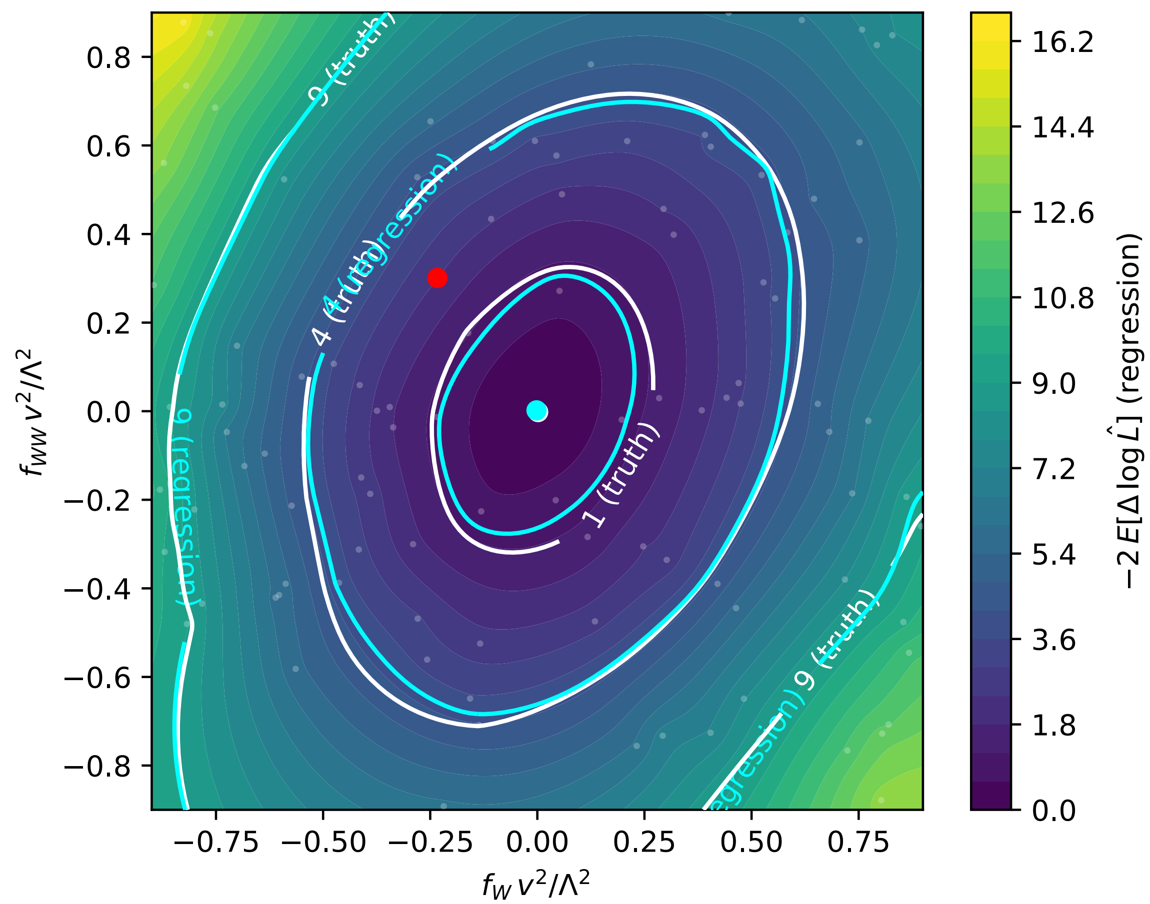
Estimating the density of $s(x;\theta_0,\theta_1)$ via the simulator calibrates the ratio.

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

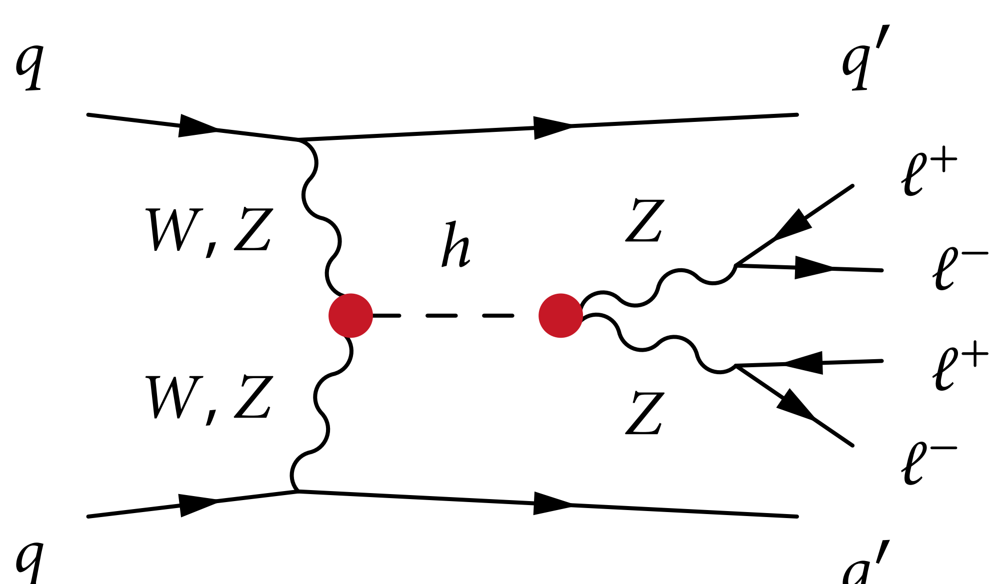
LEARNING THE HIGGS EFFECTIVE FIELD THEORY

Estimated likelihood





True likelihood



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] & CARL [arxiv:1506.02169]

STATISTICAL TASKS & LEARNING PARADIGMS

Statistical Tasks:

- Classification
- Regression
- Density Estimation
- Statistical Inference
- Decision Making

Learning Paradigms:

- Supervised Learning
- Weakly Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

Decision Making

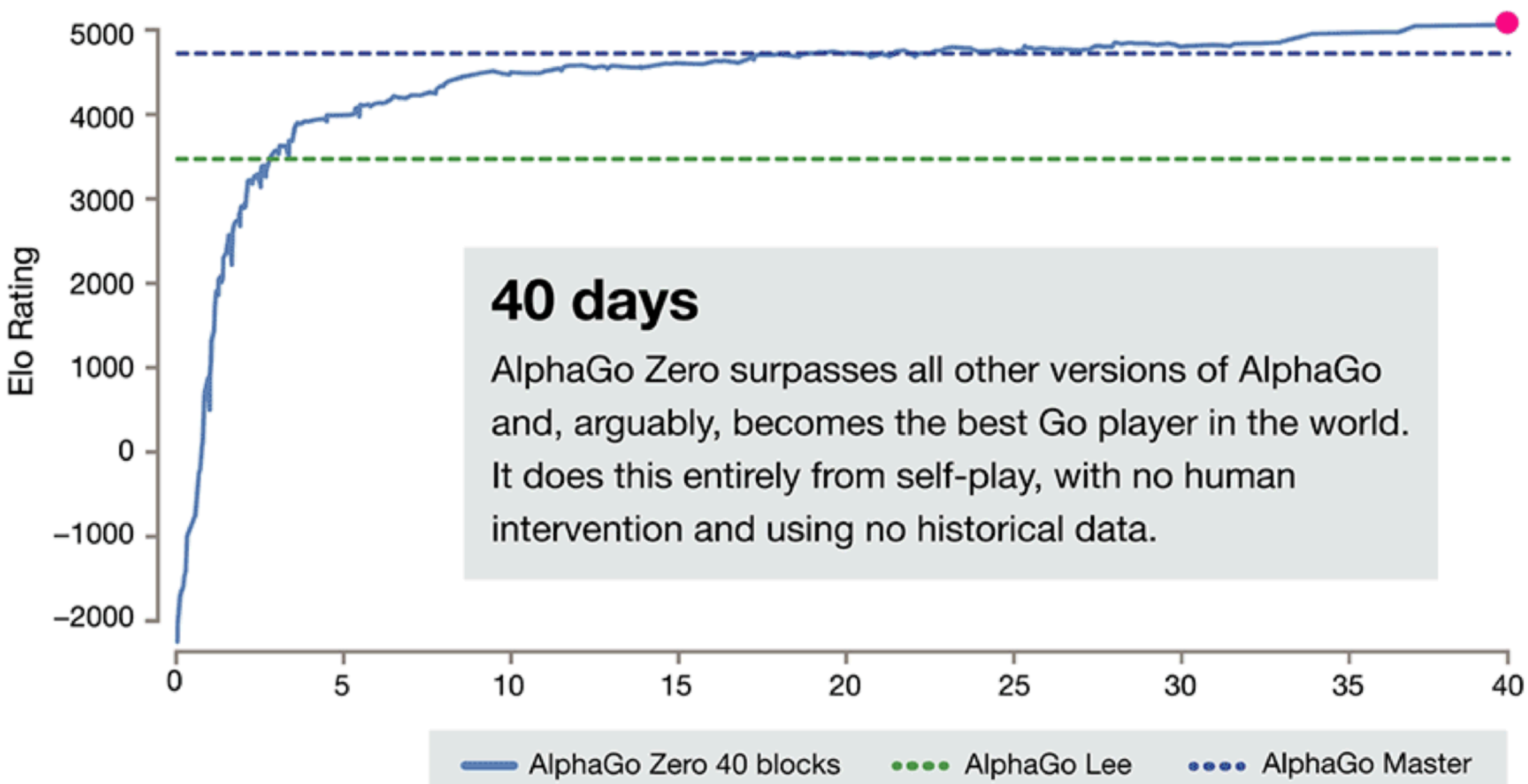
Reinforcement Learning

AlphaGo

Captured Stones

70 hours

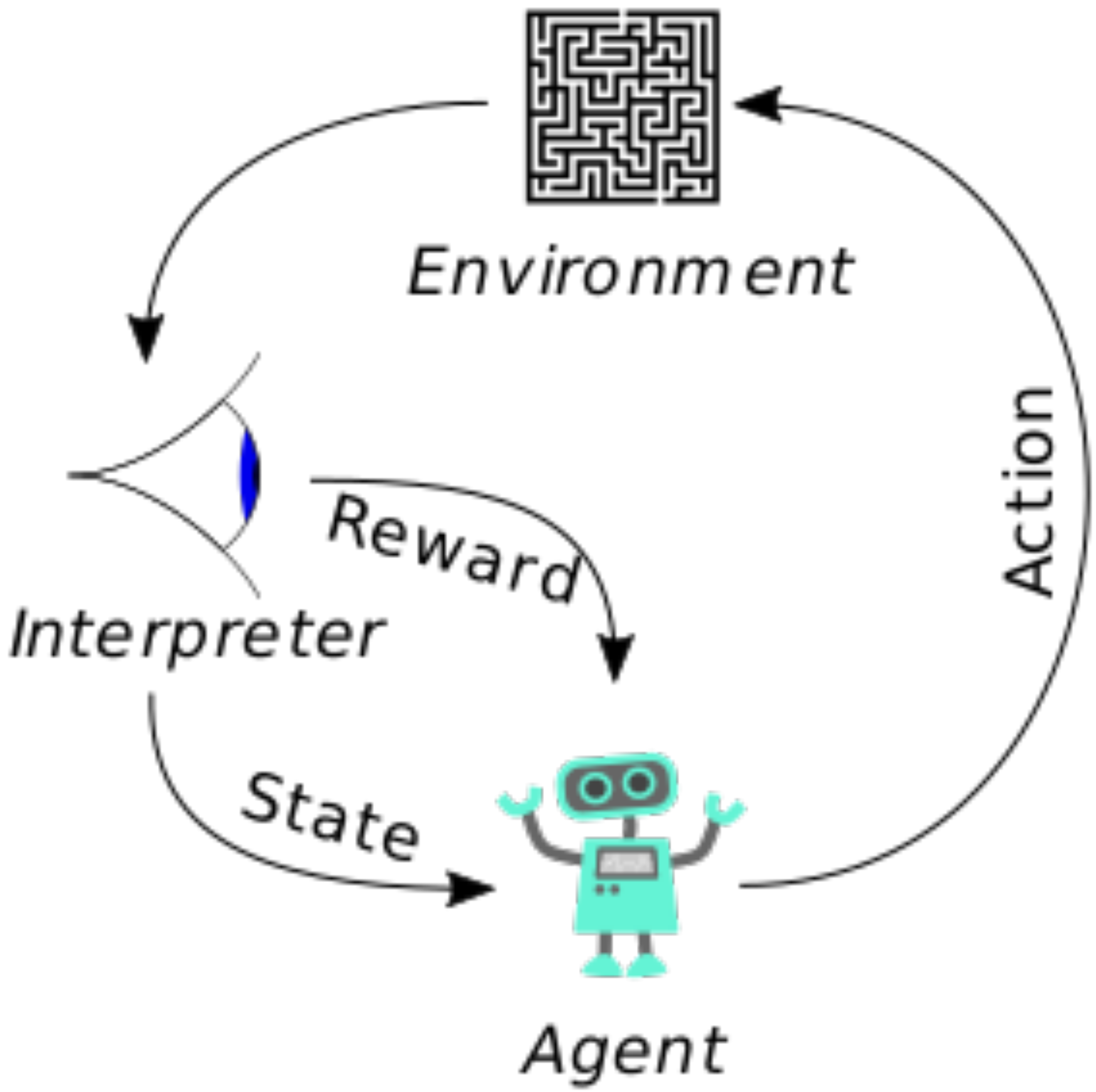
AlphaGo Zero plays at super-human level. The game is disciplined and involves multiple challenges across the board.



40 days

AlphaGo Zero surpasses all other versions of AlphaGo and, arguably, becomes the best Go player in the world. It does this entirely from self-play, with no human intervention and using no historical data.

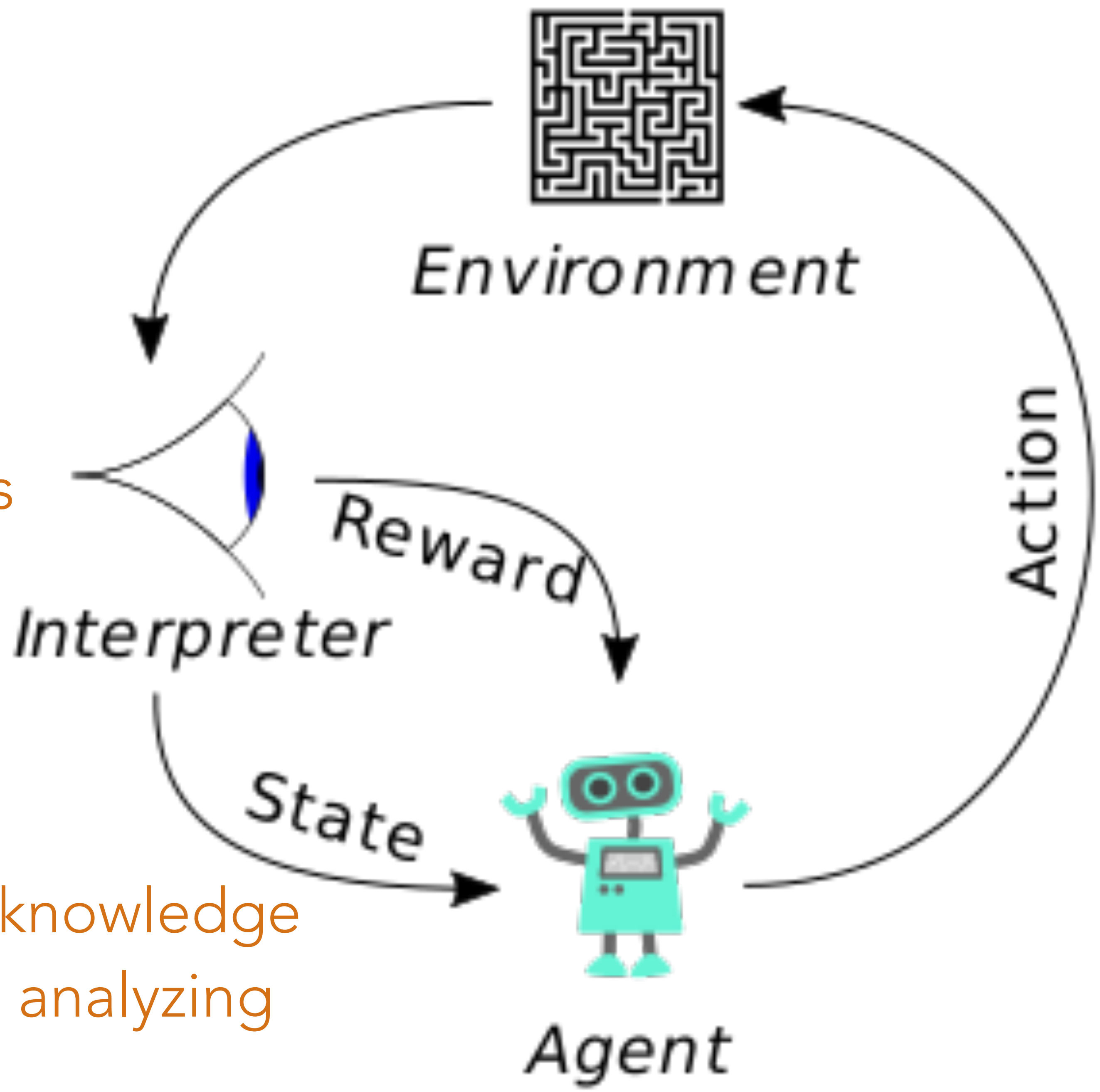
Scientist trying to decide what experiment to do next



Scientist trying to decide what experiment to do next

perform experiment,
gather data

statistical analysis



decide which
experiment to
perform

updated knowledge
based on analyzing
data

STATISTICAL DECISION THEORY IN 1 SLIDE

Θ - States of nature; X - possible observations; A - action to be taken

$p(x|\theta)$ - statistical model; $\pi(\theta)$ - prior

$\delta: X \to A$ - **decision rule** (take some action based on observation)

$L: \Theta \times A \to \mathbb{R}$ - **loss function**, real-valued function true parameter and action

$R(\theta,\delta) = E_{p(x|\theta)}[L(\theta, \delta)]$ - **risk**

- A decision δ^* rule **dominates** a decision rule δ if and only if $R(\theta,\delta^*) \leq R(\theta,\delta)$ for all θ, and the inequality is strict for some θ.

- A decision rule is **admissible** if and only if no other rule dominates it; otherwise it is inadmissible

$r(\pi, \delta) = E_{\pi(\theta)}[R(\theta,\delta)]$ - **Bayes risk** (expectation over θ w.r.t. prior and possible observations)

$\rho(\pi, \delta \mid x) = E_{\pi(\theta|x)}[L(\theta,\delta(x))]$ - **expected loss** (expectation over θ w.r.t. posterior $\pi(\theta|x)$)

- δ' is a (generalized) Bayes rule if it minimizes the expected loss

AN EXAMPLE

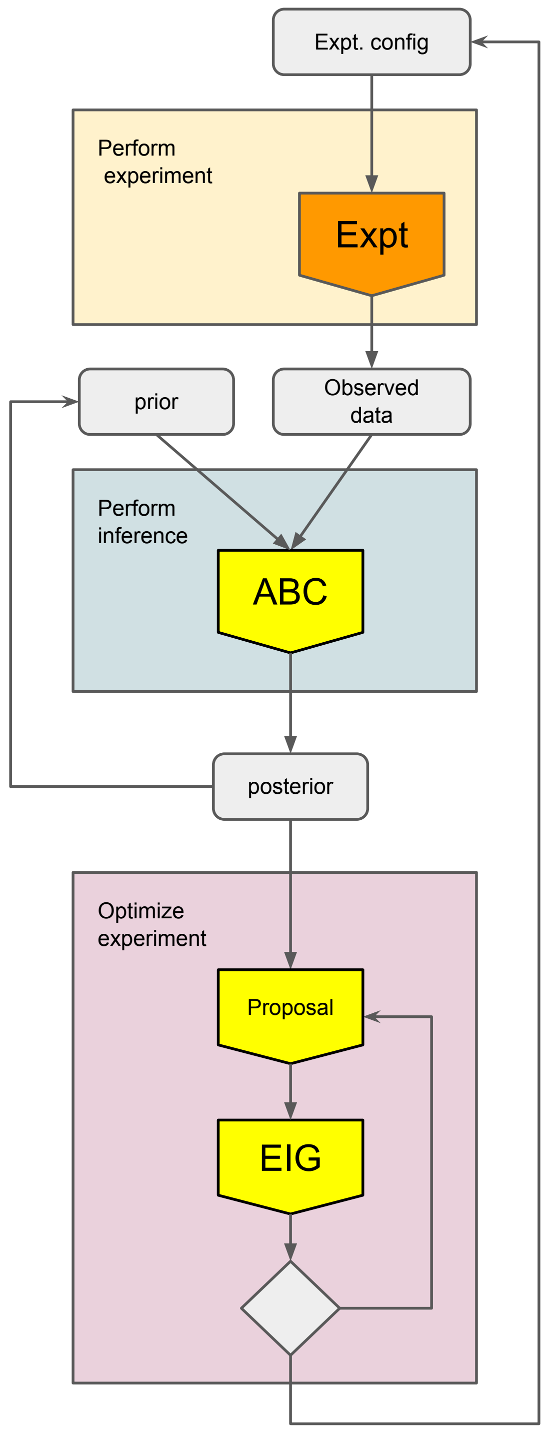
Say we want to measure the Weinberg angle

- experiments are $e^+e^- \rightarrow \mu^+\mu^-$ and we can adjust the beam energy and beam polarization
- data are 4-momenta of $\mu^+$and μ^- without knowing forward-backward asymmetry is interesting observable

Can we use likelihood-free inference to:

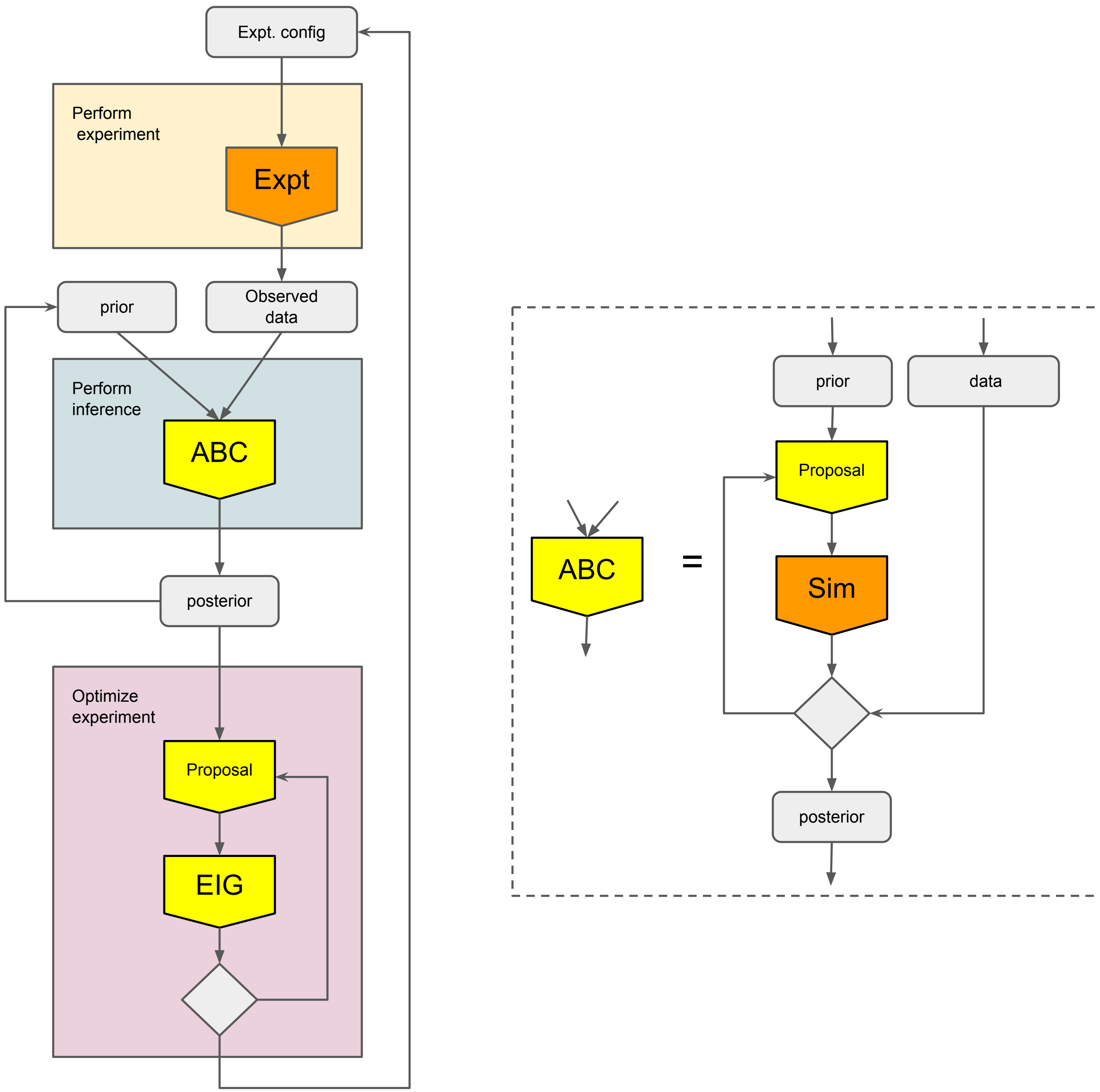
- estimate θ_W from p_{μ^+} & p_{μ^-} generated from simulator?
- decide which beam energy and polarization are optimal for this measurement?

ACTIVE SCIENCING



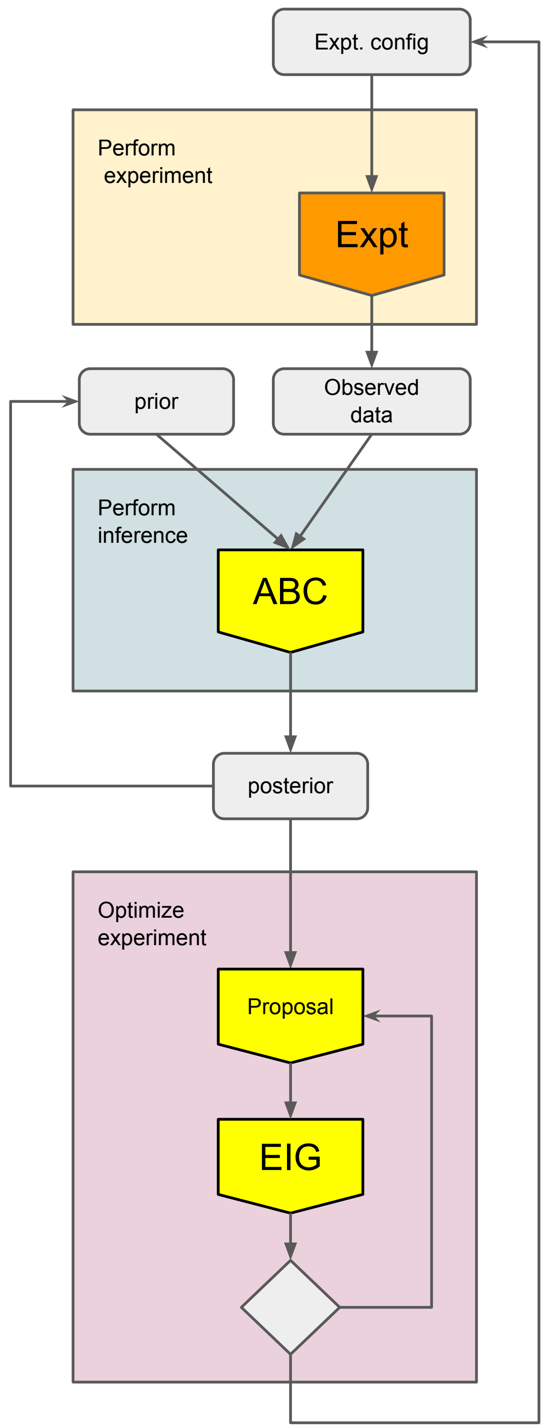
https://github.com/cranmer/active_sciencing

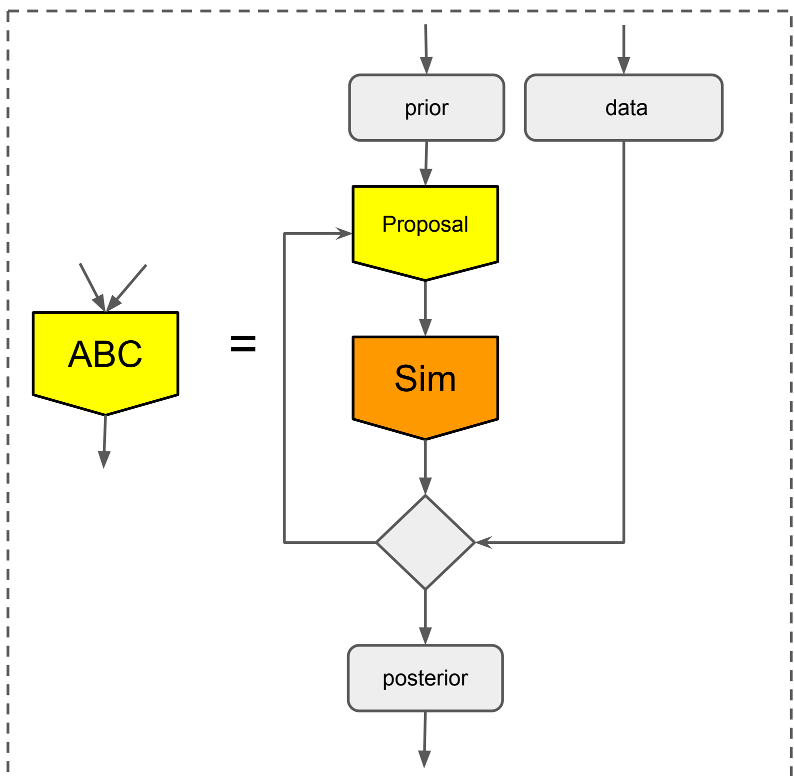
ACTIVE SCIENCING

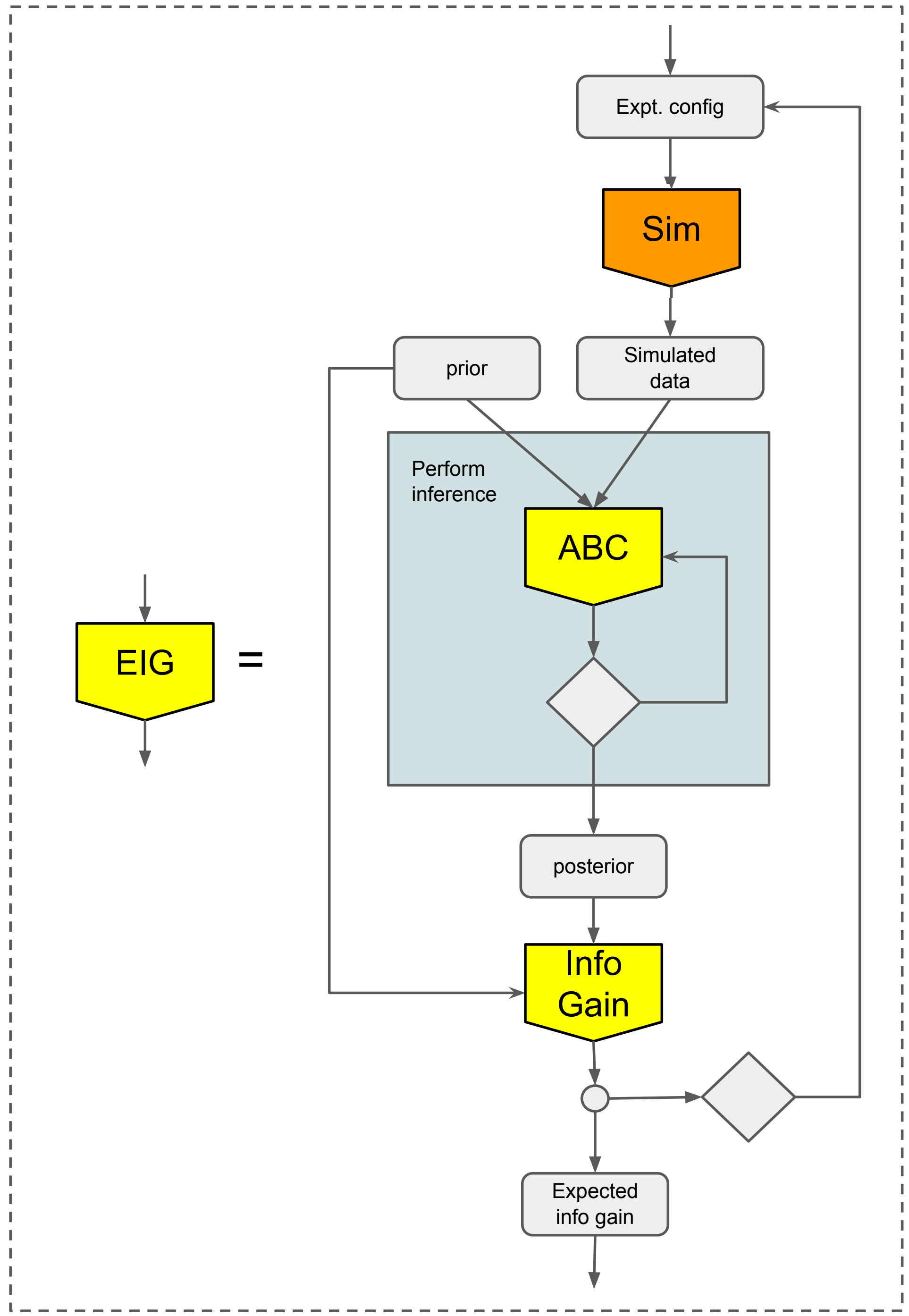


https://github.com/cranmer/active_sciencing

ACTIVE SCIENCING





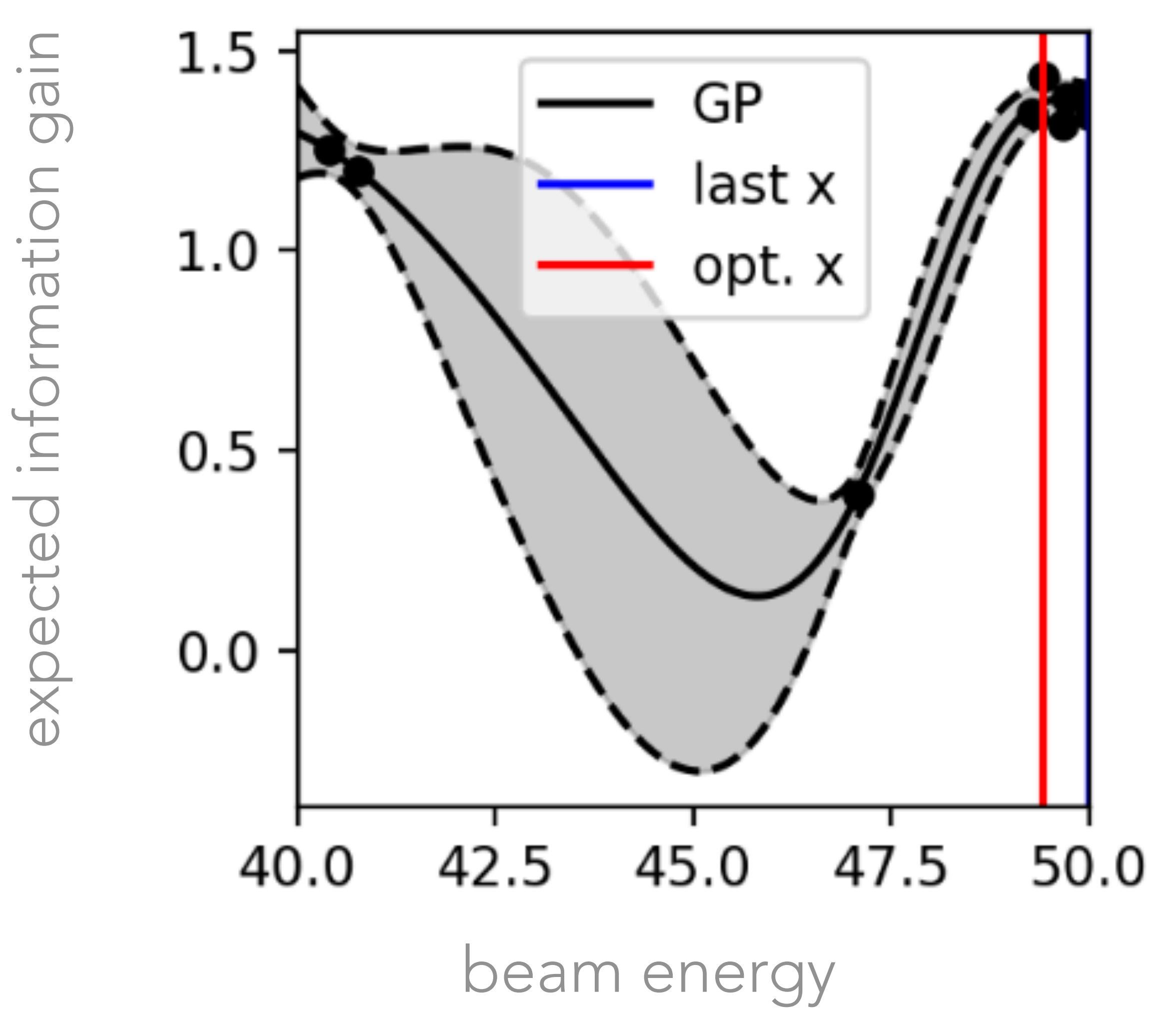


https://github.com/cranmer/active_sciencing

Input:

- workflow for performing "real" experiment that returns data
- workflow for running simulator given parameters of theory and experimental configuration

Automated system can measure the Weinberg angle and optimize beam energy (eg. just above or below $M_Z/2$) just from using simulator



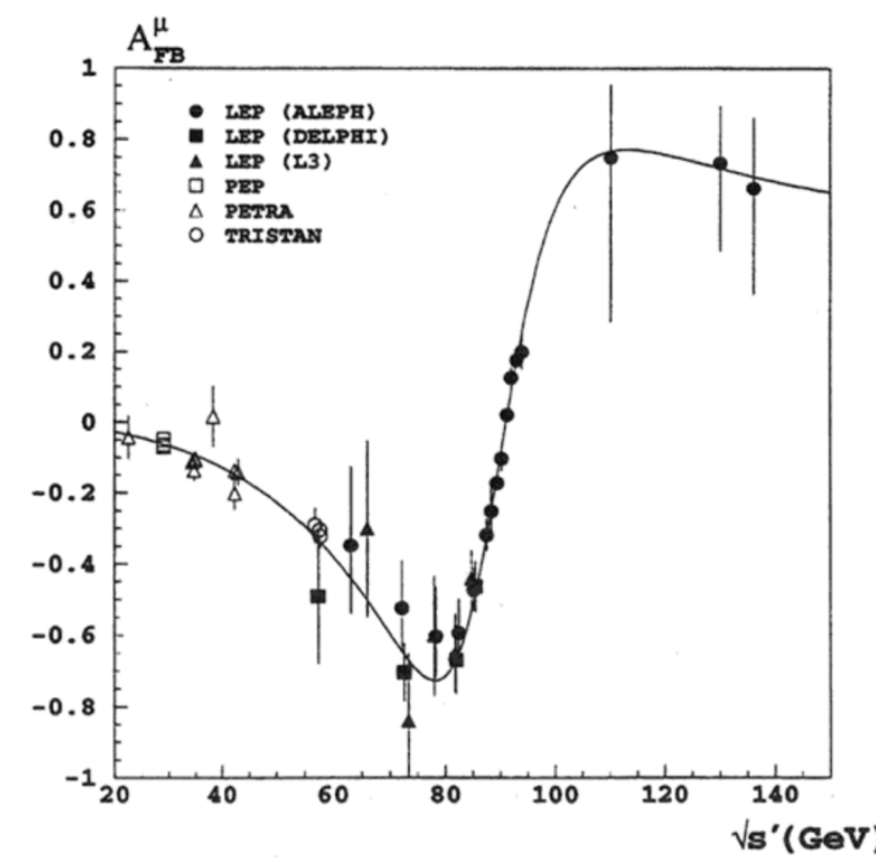


Figure 2: Measured forward-backward asymmetries of muon-pair production compared with the model independent fit results.

Generative Models:

“What I cannot create, I do not understand.”

—RICHARD FEYNMAN

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

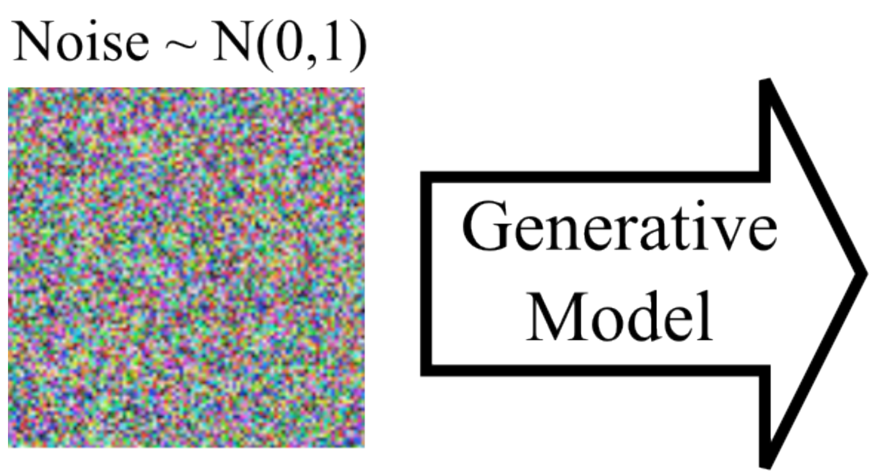
INFERENCE

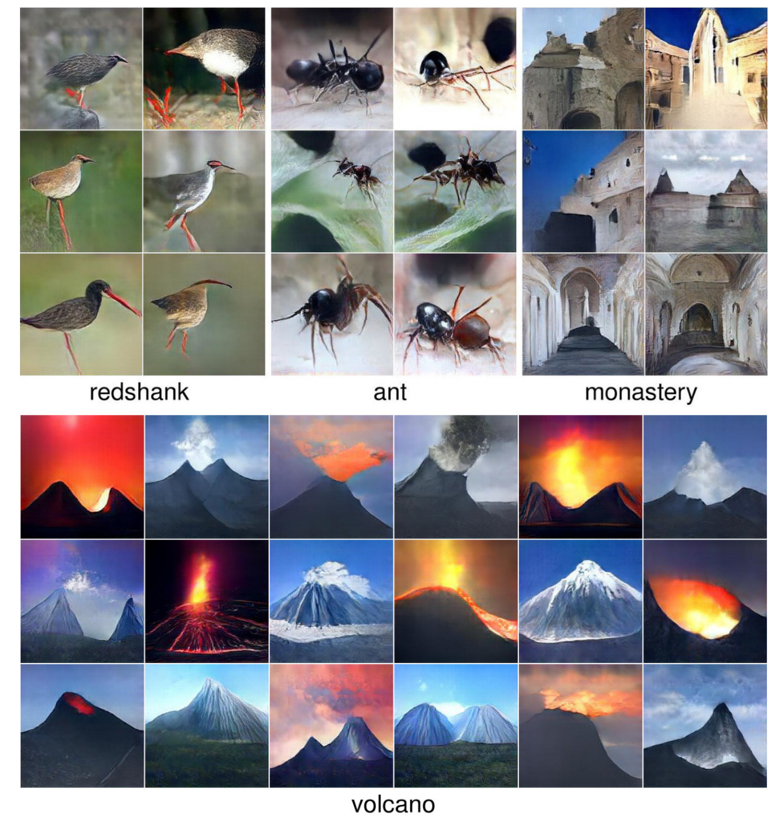
inverse problem
measurement
parameter estimation

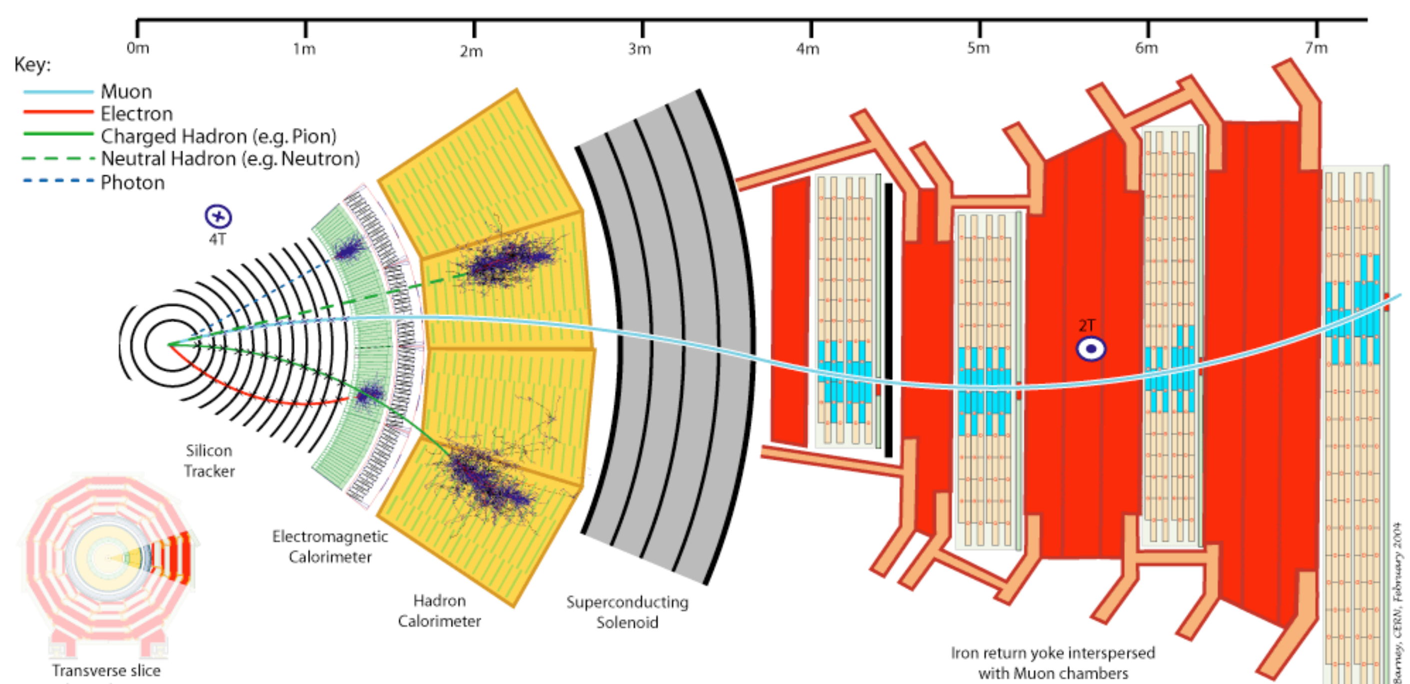
LEARNING THE GENERATIVE MODEL

Z

X



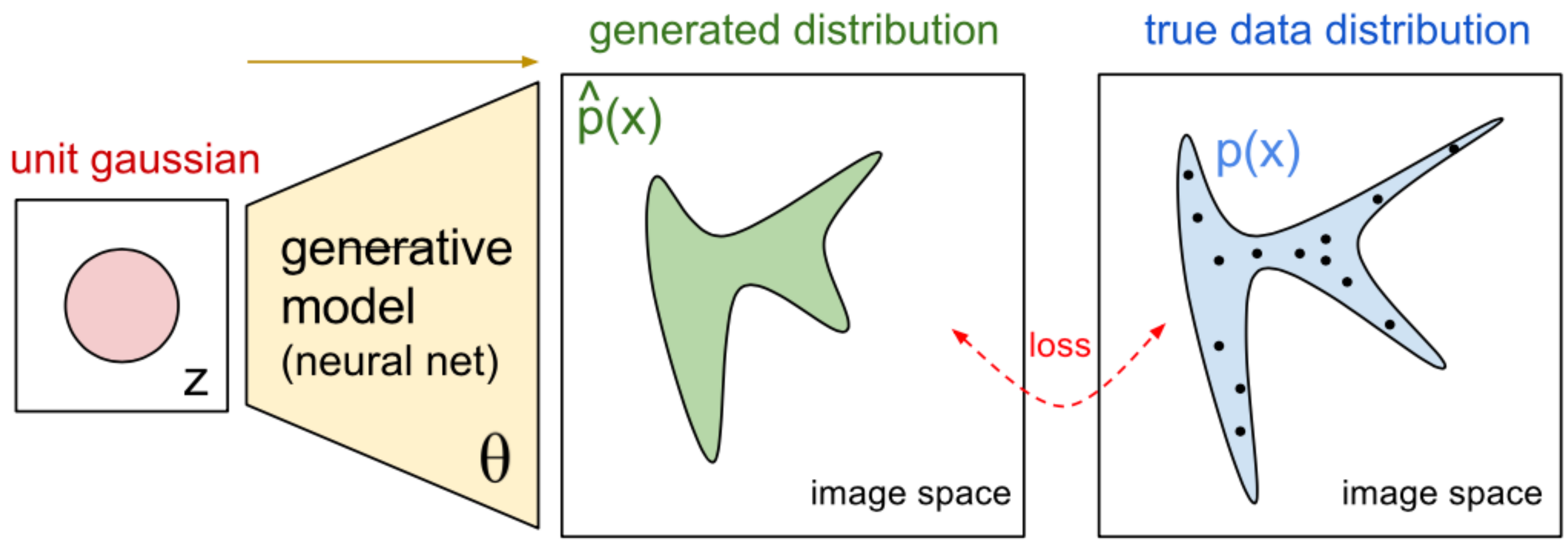




http://torch.ch/blog/2015/11/13/gan.html

GENERATIVE ADVERSARIAL NETWORKS

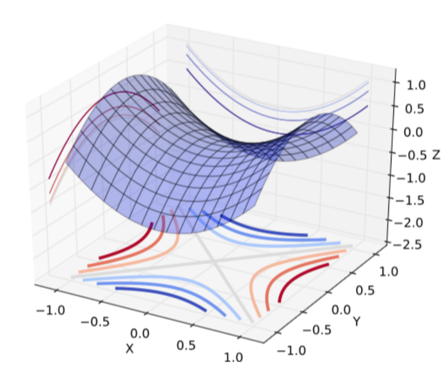
Goodfellow, et al arXiv:1406.2661
slide Gilles Louppe



Leo is G Tom is D

- 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.

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



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

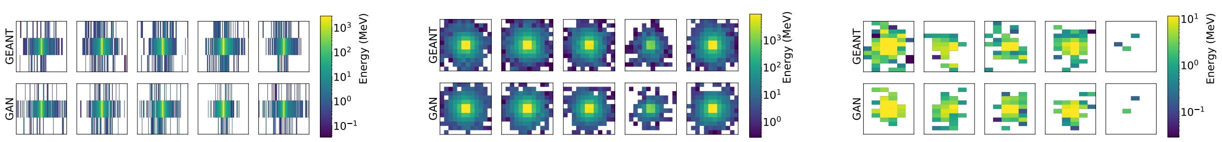


Figure 9: Five randomly selected e^+ showers per calorimeter layer from the training set (top) and the five nearest neighbors (by euclidean distance) from a set of CaloGAN candidates.

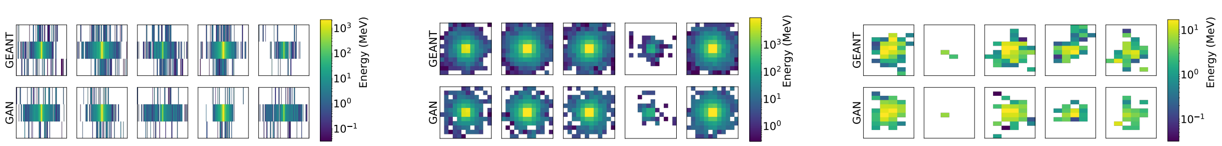


Figure 10: Five randomly selected γ showers per calorimeter layer from the training set (top) and the five nearest neighbors (by euclidean distance) from a set of CaloGAN candidates.

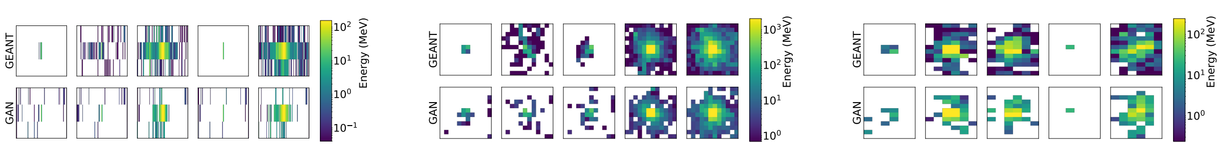
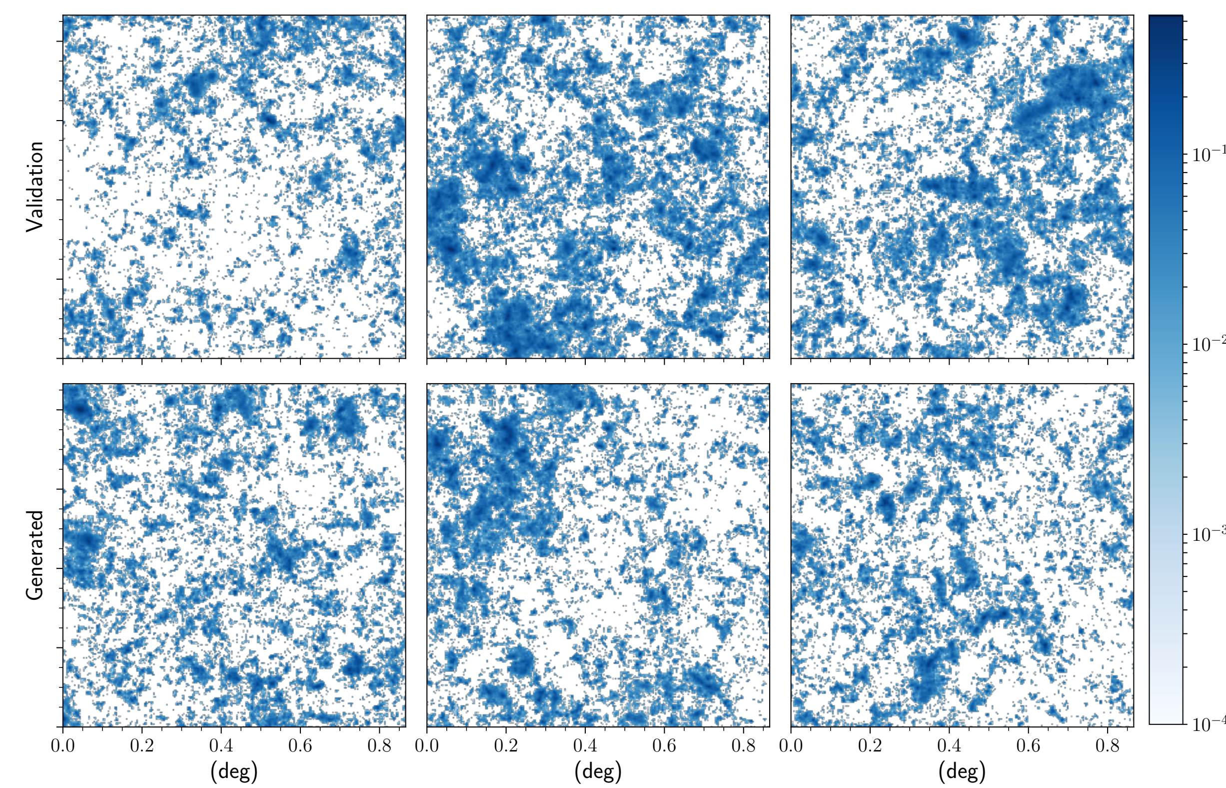


Figure 11: Five randomly selected π^+ showers per calorimeter layer from the training set (top) and the five nearest neighbors (by euclidean distance) from a set of CaloGAN candidates.

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



GENERATIVE MODELS FOR CALIBRATION

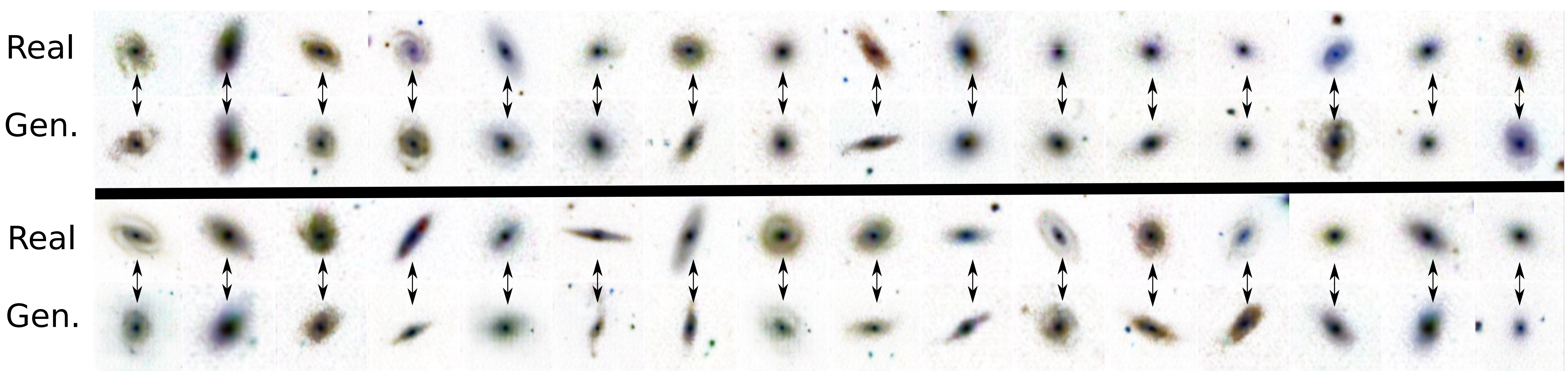
Use of generative models of galaxy images to help calibrate down-stream analysis in next-generation surveys.

Enabling Dark Energy Science with Deep Generative Models of Galaxy Images

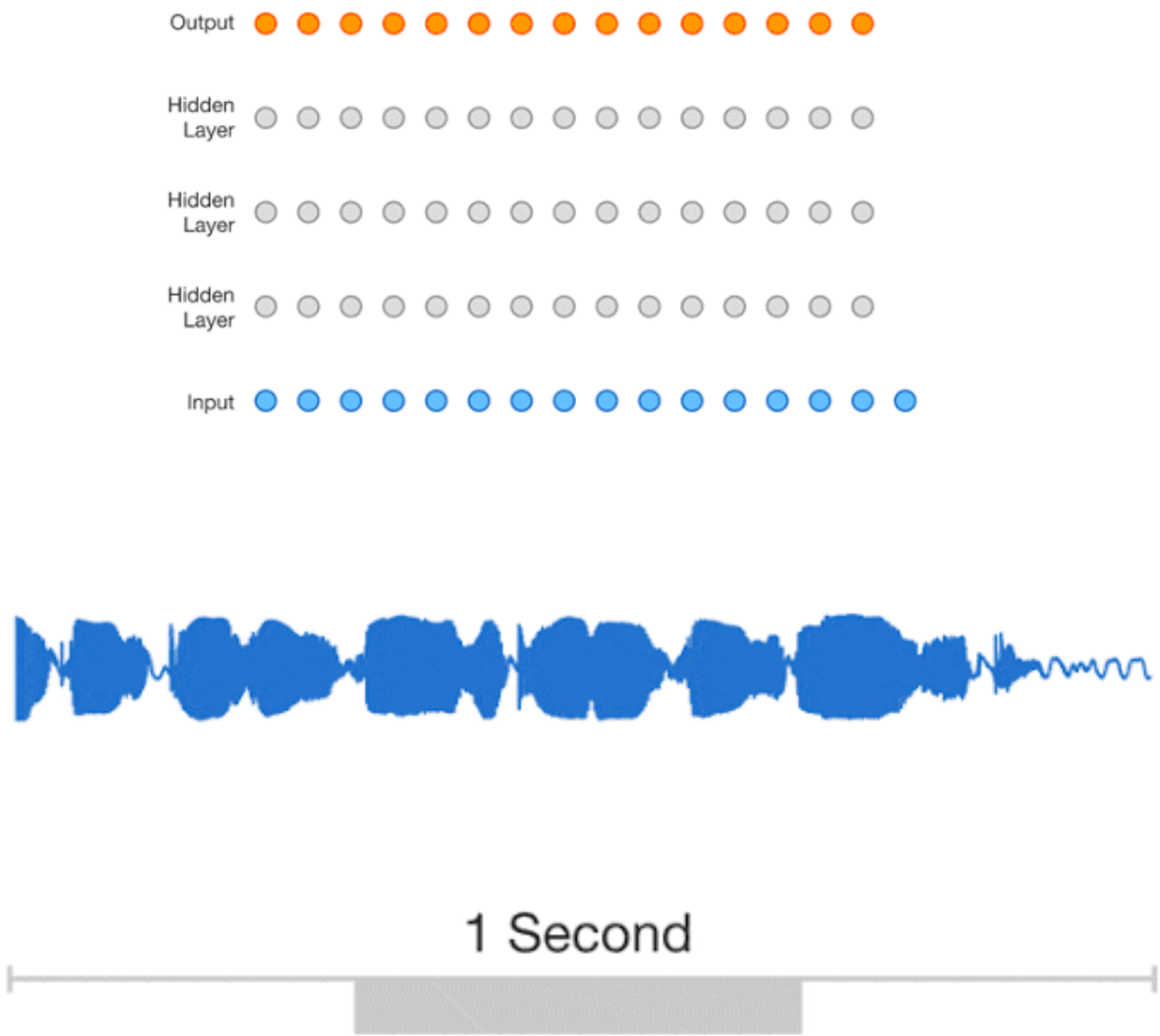
Siamak Ravanbakhsh[1], François Lanusse[2], Rachel Mandelbaum[2], Jeff Schneider[1], and Barnabás Póczos[1]

[1]School of Computer Science, Carnegie Mellon University
[2]McWilliams Center for Cosmology, Carnegie Mellon University

***Abstract*—Understanding the nature of dark energy, the mysterious force driving the accelerated expansion of the Universe, is a major challenge of modern cosmology. The next generation of cosmological surveys, specifically designed to address this issue, rely on accurate measurements of the apparent shapes of distant galaxies. However, shape measurement methods suffer from various unavoidable biases and therefore will rely on a precise calibration to meet the accuracy requirements of the science analysis. This calibration process remains an open challenge as it requires large sets of high quality galaxy images. To this end, we study the application of deep conditional generative models in generating realistic galaxy images. In particular we consider variations on conditional variational autoencoder and introduce a new adversarial objective for training of conditional generative networks. Our results suggest a reliable alternative to the acquisition of expensive high quality observations for generating the calibration data needed by the next generation of cosmological surveys.**



Some generative models can be inverted ⇒ likelihood-free inference!



CONCLUSIONS

The developments in machine learning and AI go way beyond improved classifiers and have the potential to transform how we do science

- many areas of science have simulations based on some well-motivated mechanistic model
- generative models and likelihood-free inference are two particularly exciting areas
- they can provide effective theories of macroscopic phenomena that are tied back to the low-level microscopic (reductionist) model

Scientific challenges also motivate machine learning research

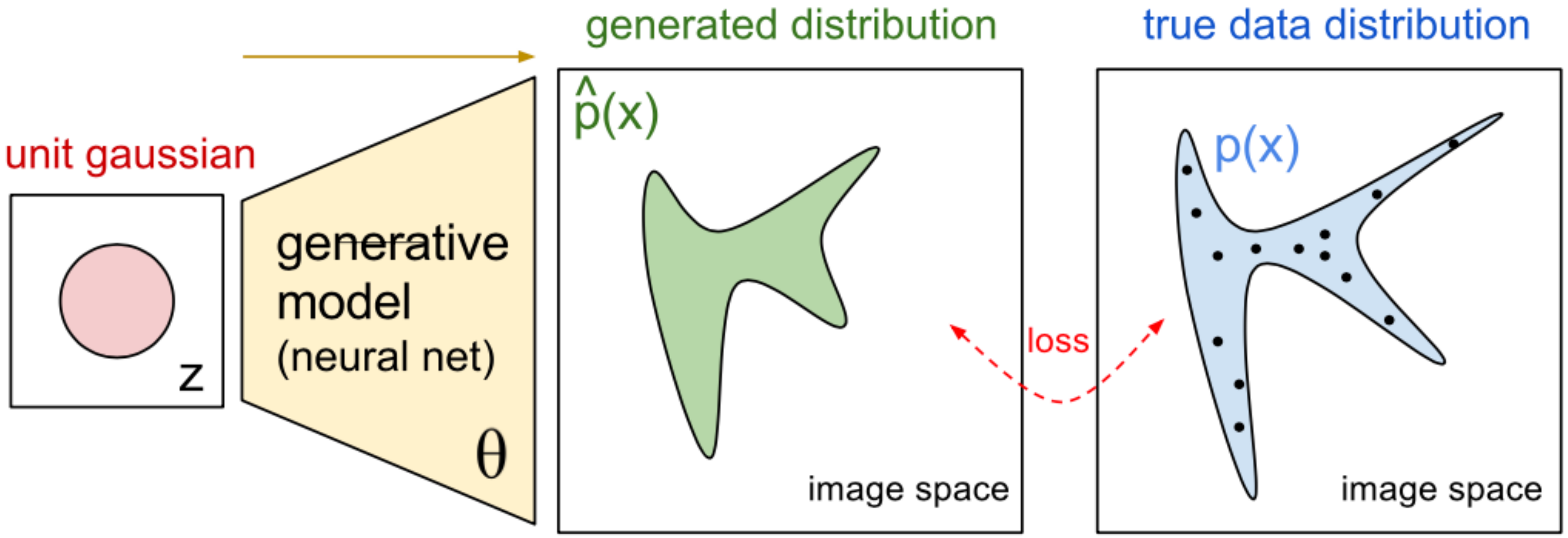
- incorporation of domain knowledge, robustness to systematic uncertainties, modularization & interpretability, non-differentiable simulators, …

Backup

Adversarial Training
(not just for GANs)

GENERATIVE ADVERSARIAL NETWORKS

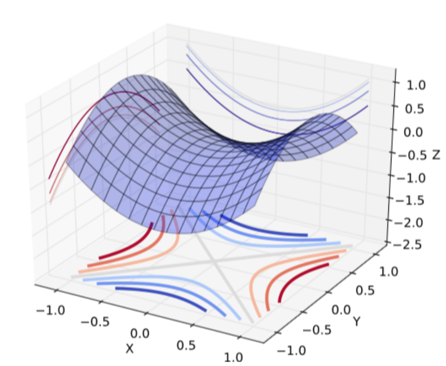
Goodfellow, et al arXiv:1406.2661
slide Gilles Louppe



Leo is G Tom is D

- 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.

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



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**!

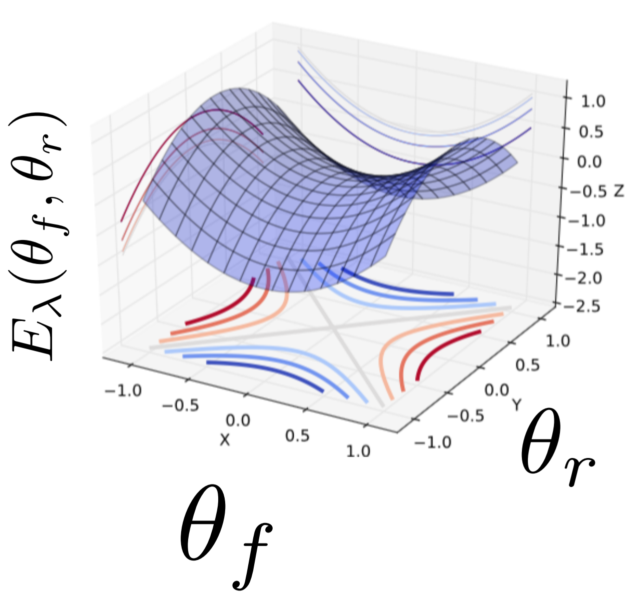
Allows us to efficiently fit / **tune simulation** with stochastic gradient techniques!

Typically classifier **f(x)** trained to minimize loss $\mathbf{L_f}$.

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

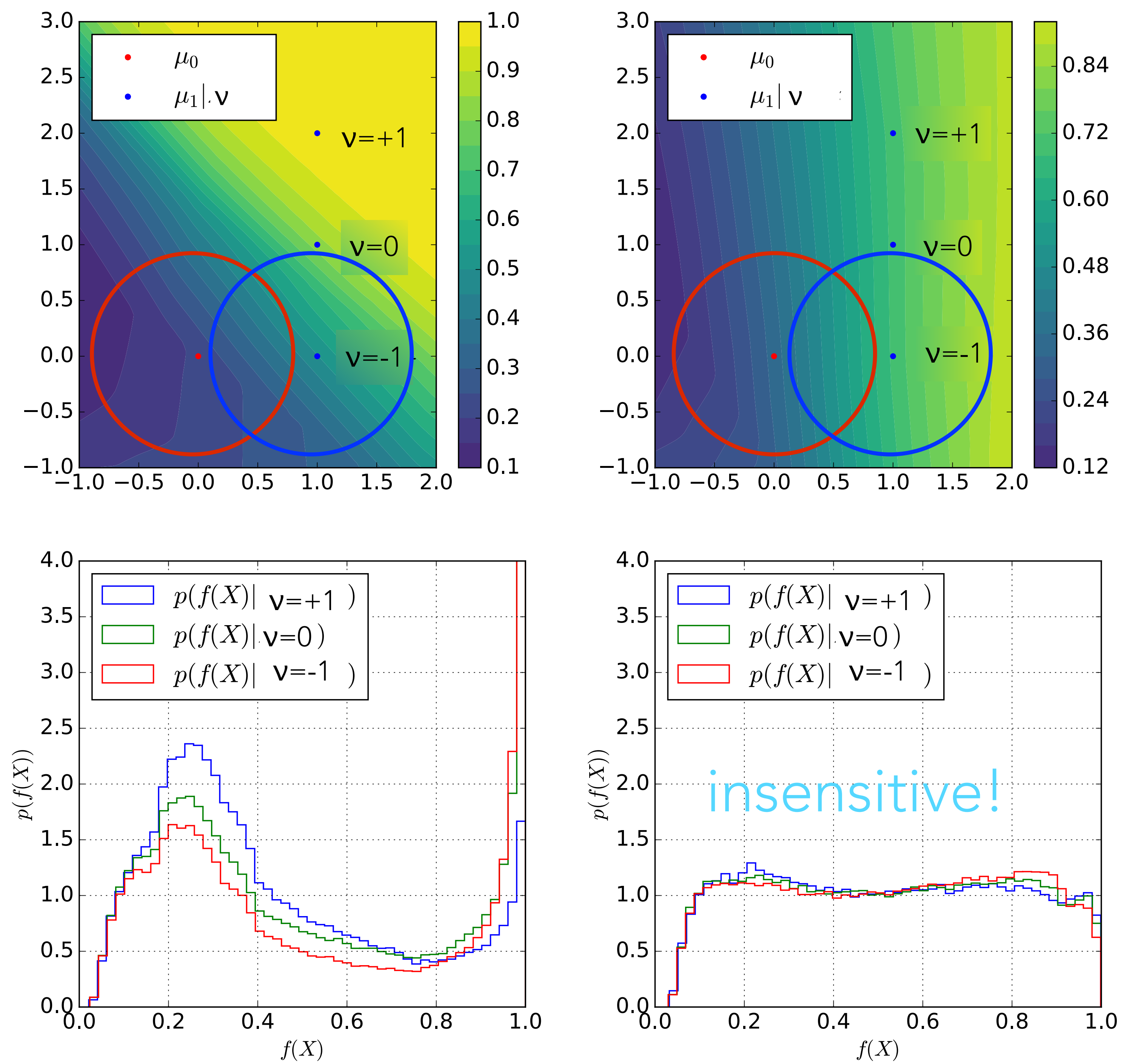
$$\hat{\theta}_f, \hat{\theta}_r = \arg \min_{\theta_f} \max_{\theta_r} E(\theta_f, \theta_r).$$

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



normal training

adversarial training



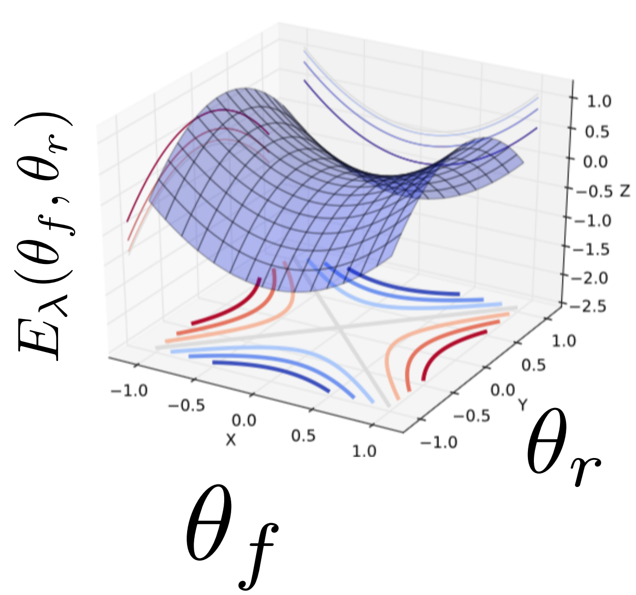
G. Louppe, M. Kagan, K. Cranmer, arXiv:1611.01046

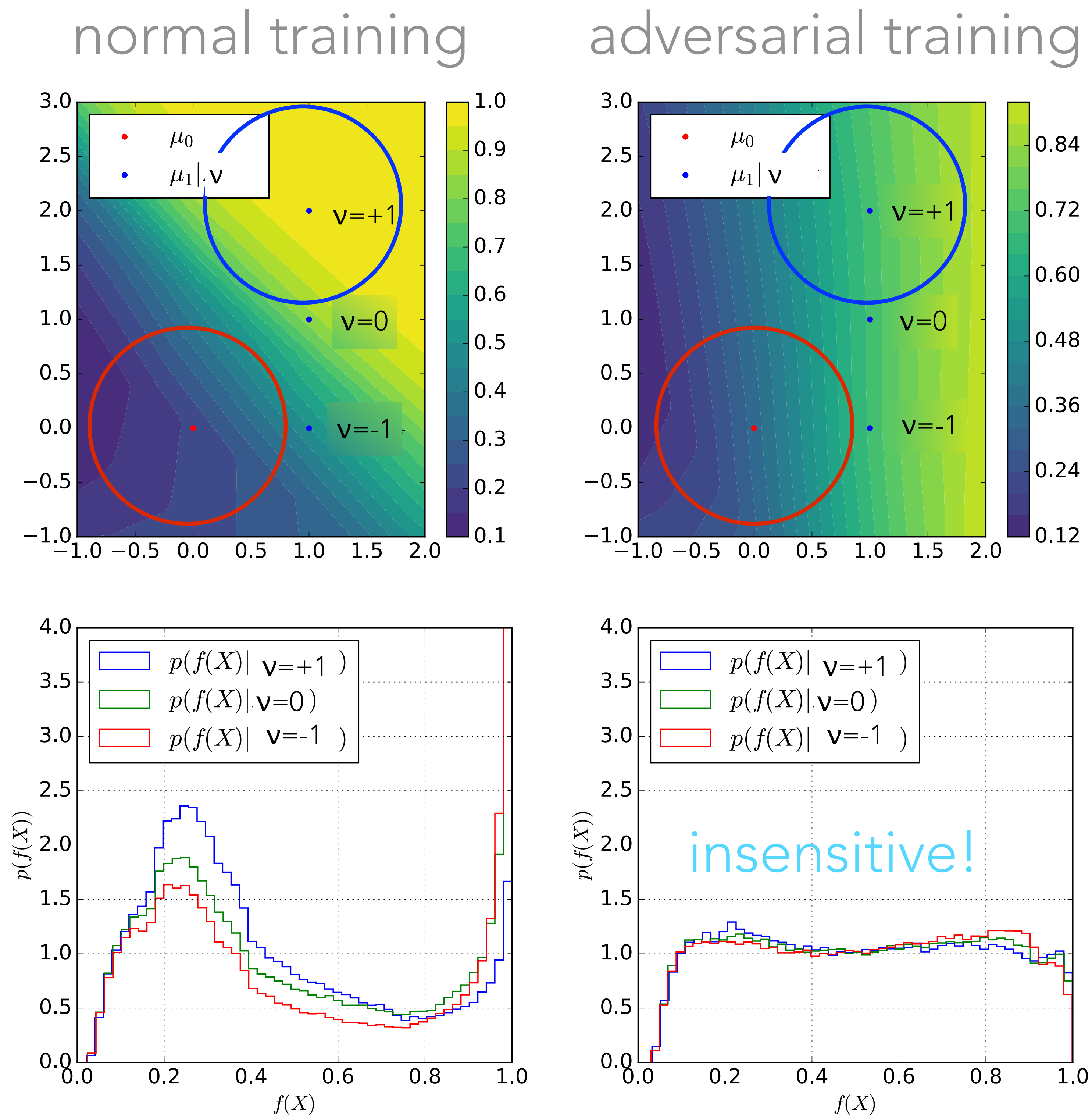
Typically classifier **f(x)** trained to minimize loss **L_f**.

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

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

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





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

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

Technique allows us to tune λ, the tradeoff between classification power and robustness to systematic uncertainty

An example:

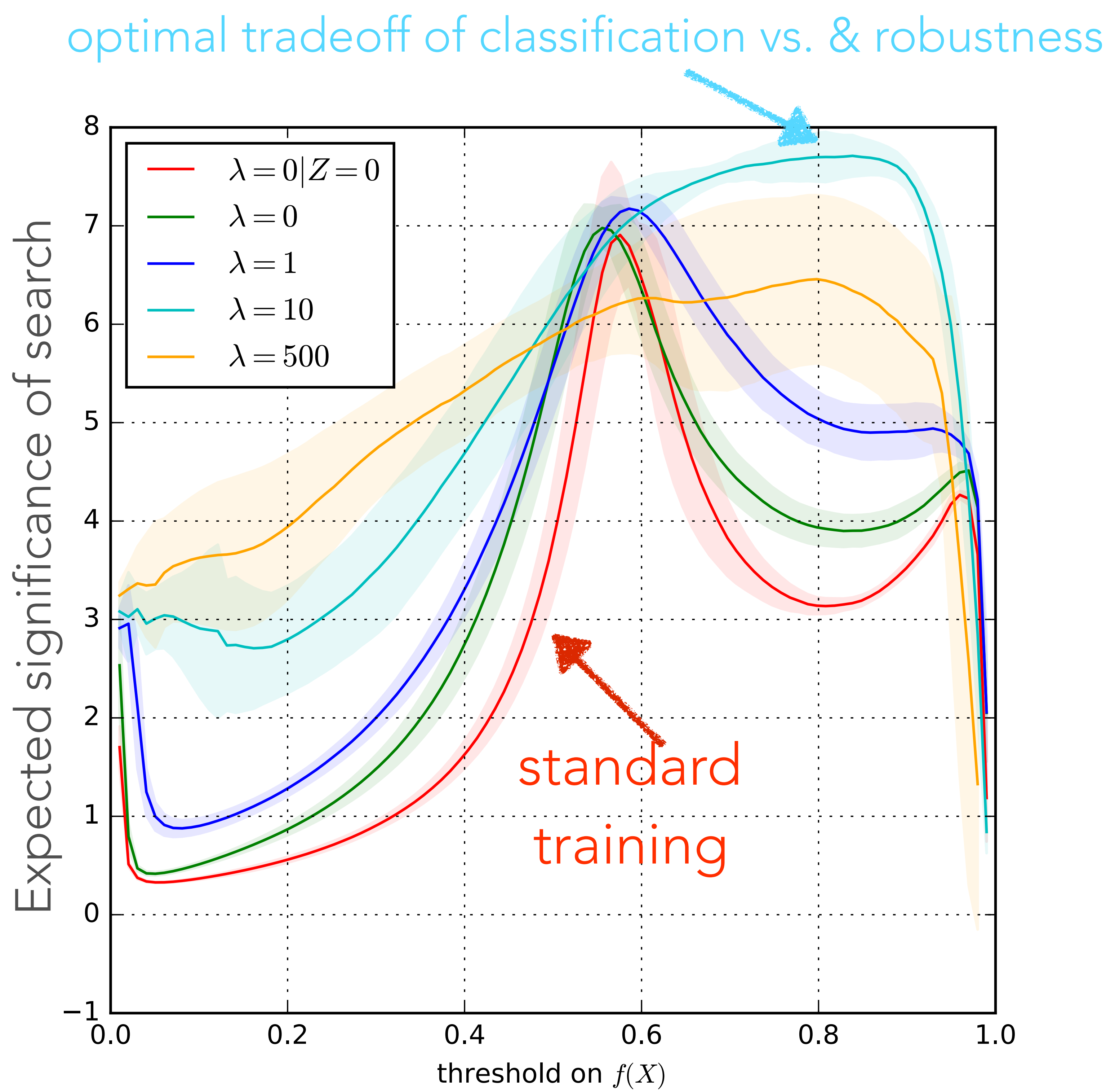
background: 1000 QCD jets

signal: 100 boosted W's

Train W vs. QCD classifier

Pileup as source of uncertainty

Simple cut-and-count analysis with background uncertainty.

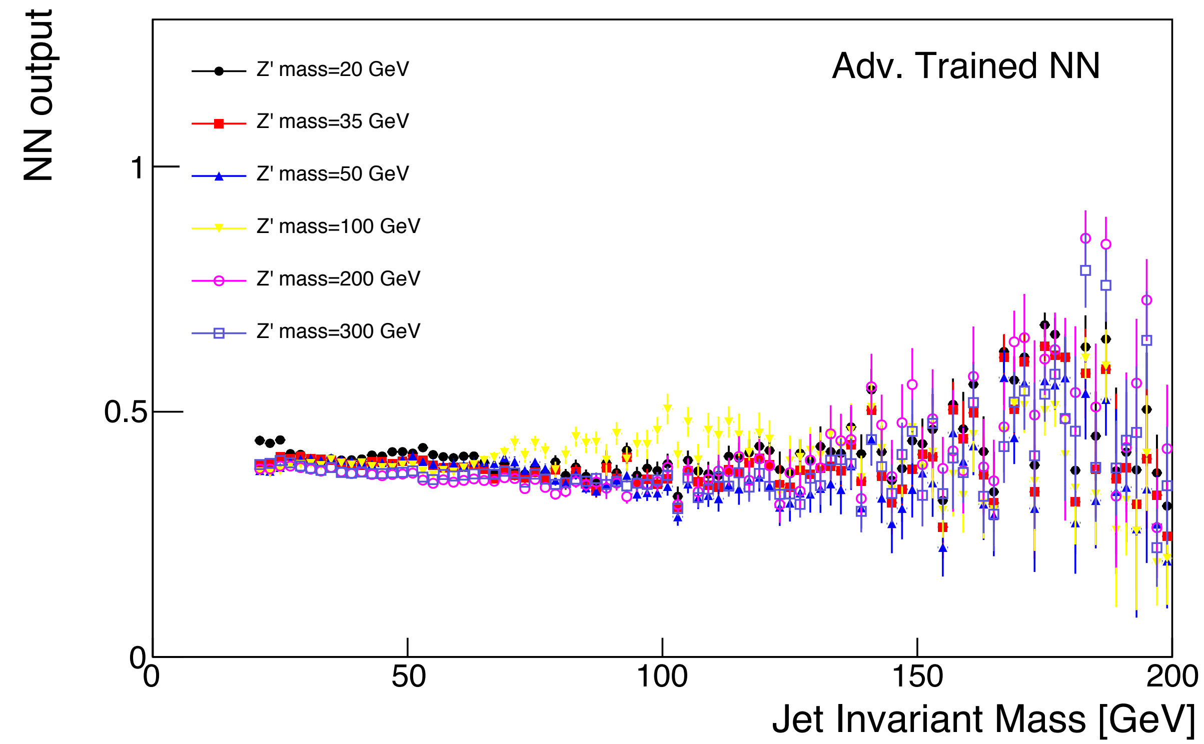


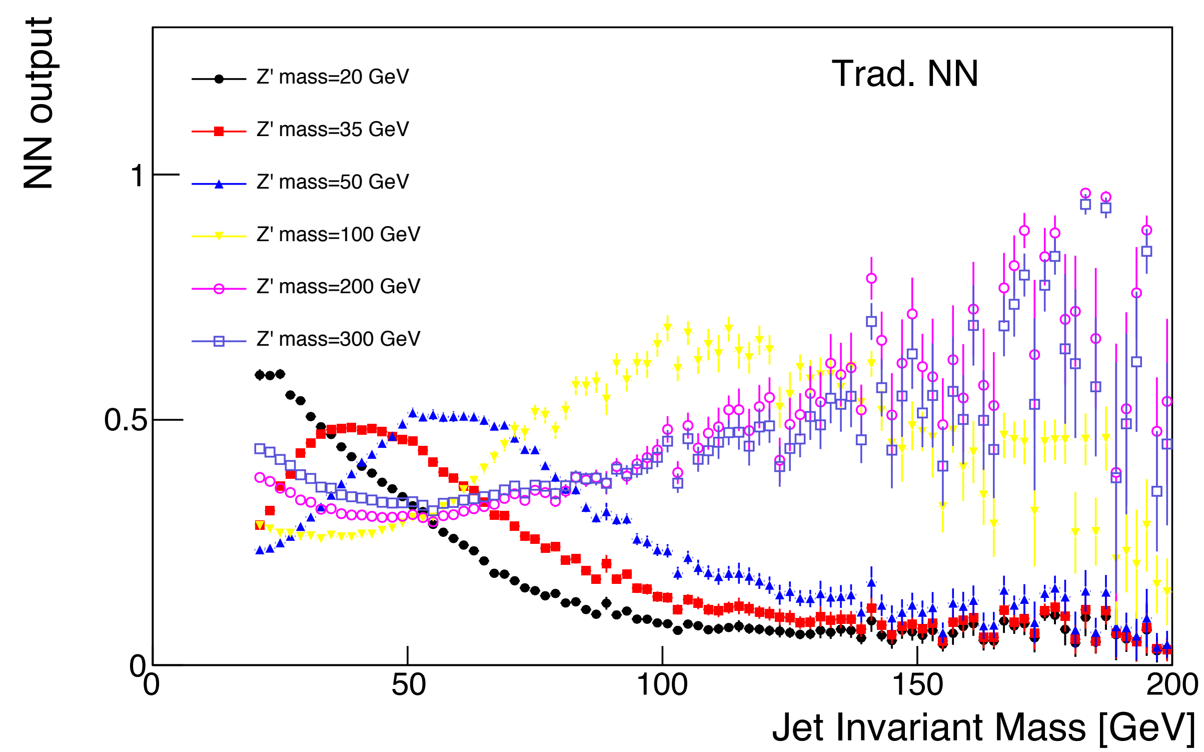
DECORRELATED TAGGERS

K.C, J. Pavez, and G. Louppe, arXiv:1506.02169
P. Baldi, K.C, T. Faucett, P. Sadowski, D. Whiteson arXiv:1601.07913
G. Louppe, M. Kagan, K.C, arXiv:1611.01046
Shimmin, et. al. arXiv:1703.03507

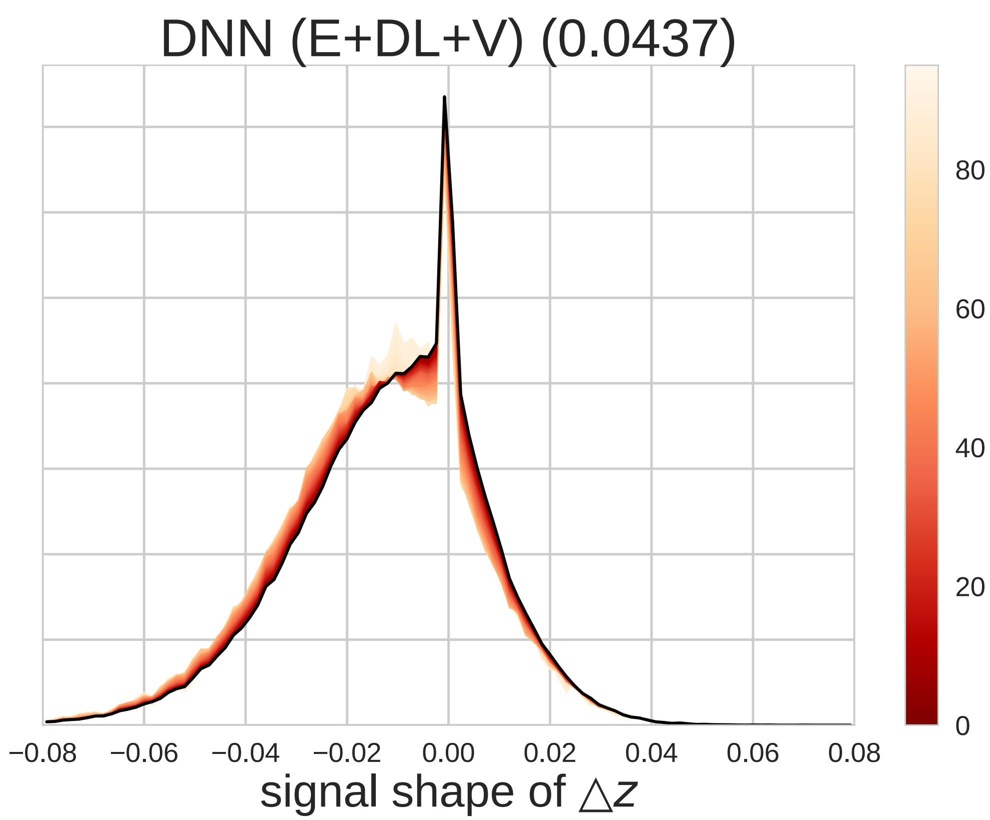
Adversarial approach of "Learning to Pivot" can also be used to train a classifier that is "decorrelated" to some other variable.

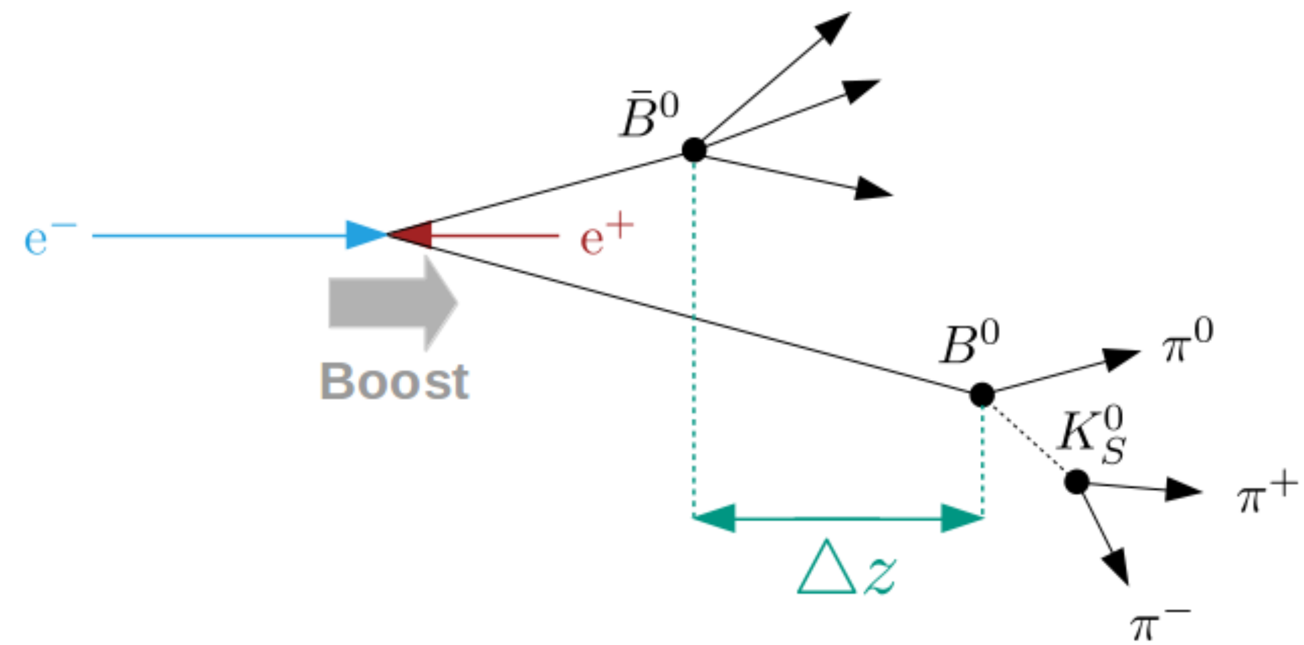
- want jet taggers that are decorrelated with jet invariant mass
- so that analysis can still search for a bump using jet invariant mass
- avoids sculpting background

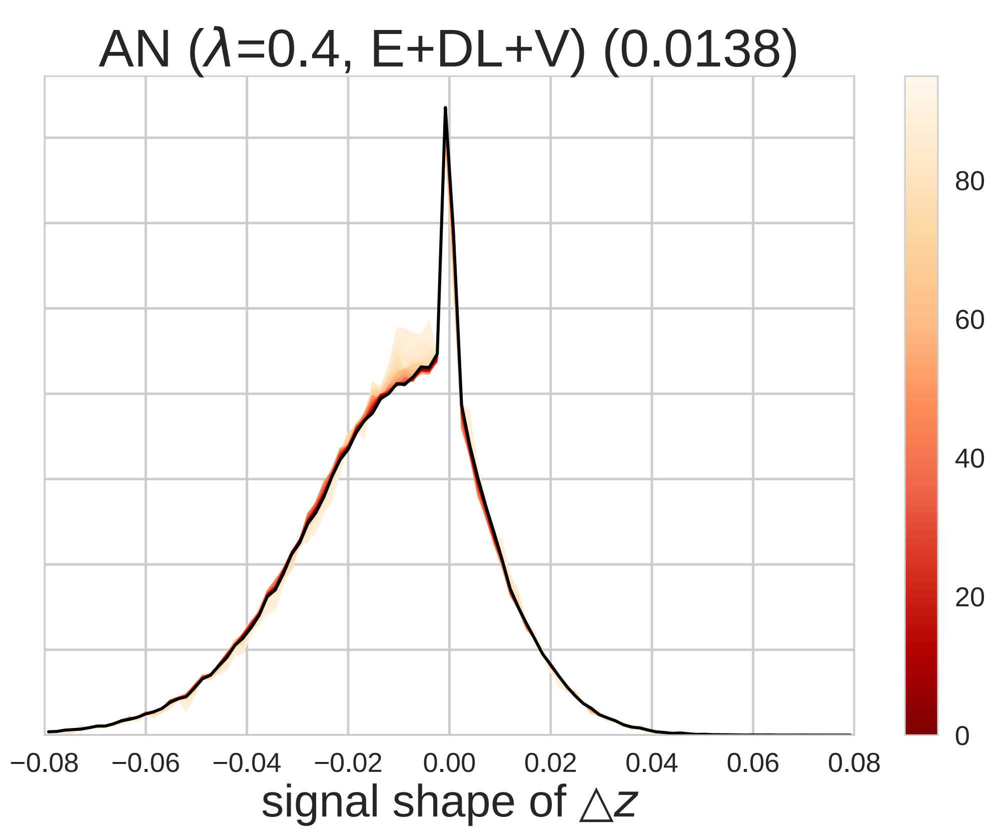


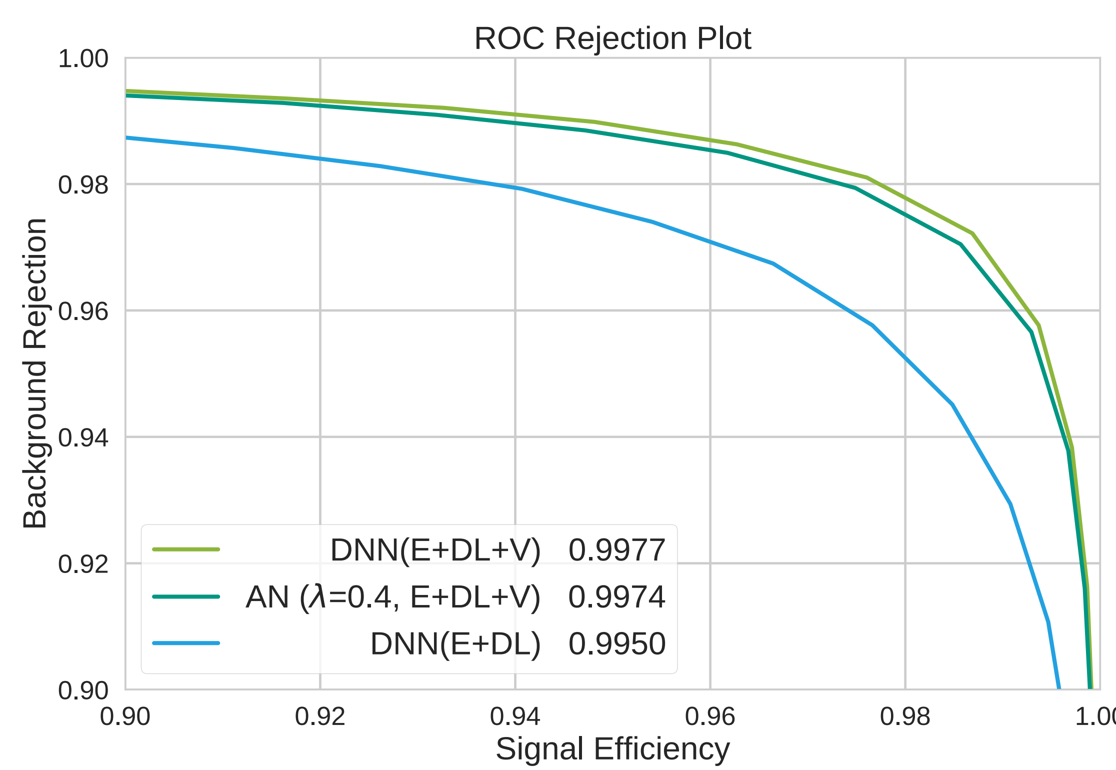


DECORRELATION IN BELLE II





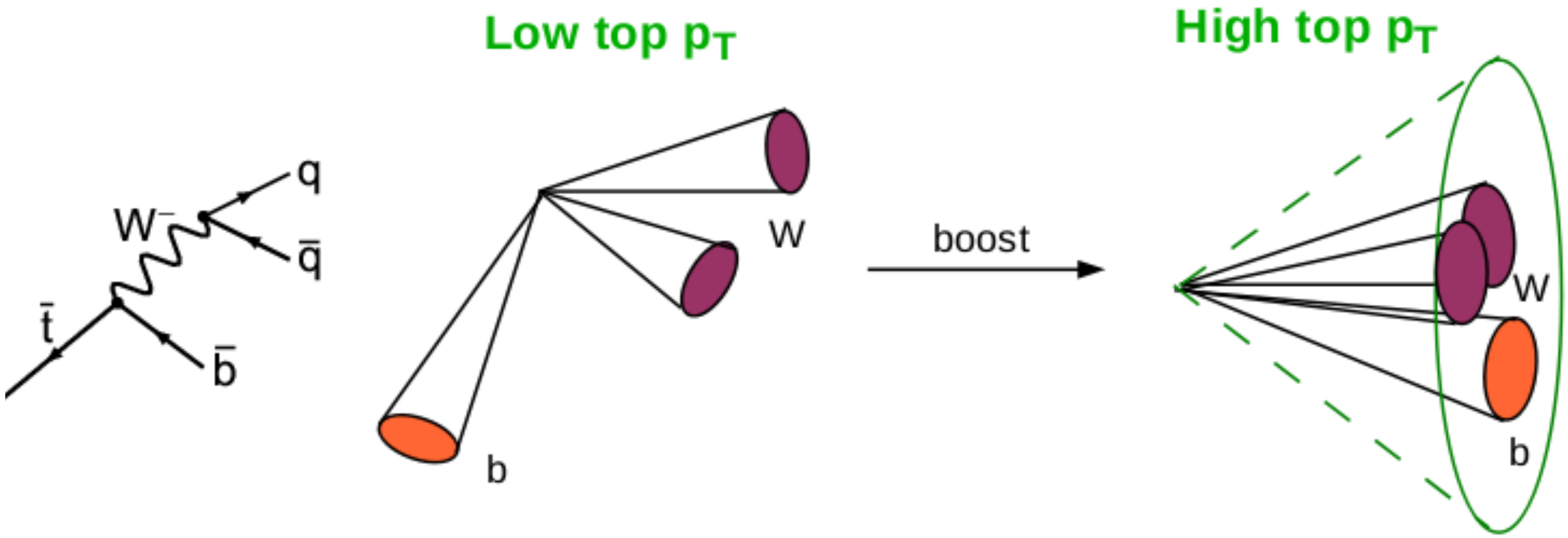




Physics-Aware Machine Learning

(choosing the variational family)

Many scenarios for physics Beyond the Standard Model include highly boosted W, Z, H bosons or top quarks



Identifying these rests on subtle substructure inside jets

- an enormous number of theoretical effort in developing observables and techniques to tag jets like this

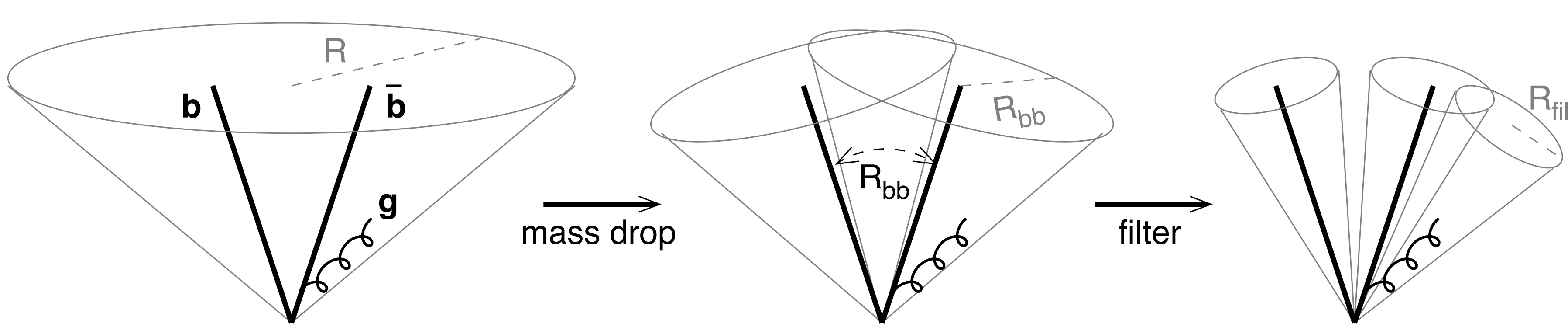
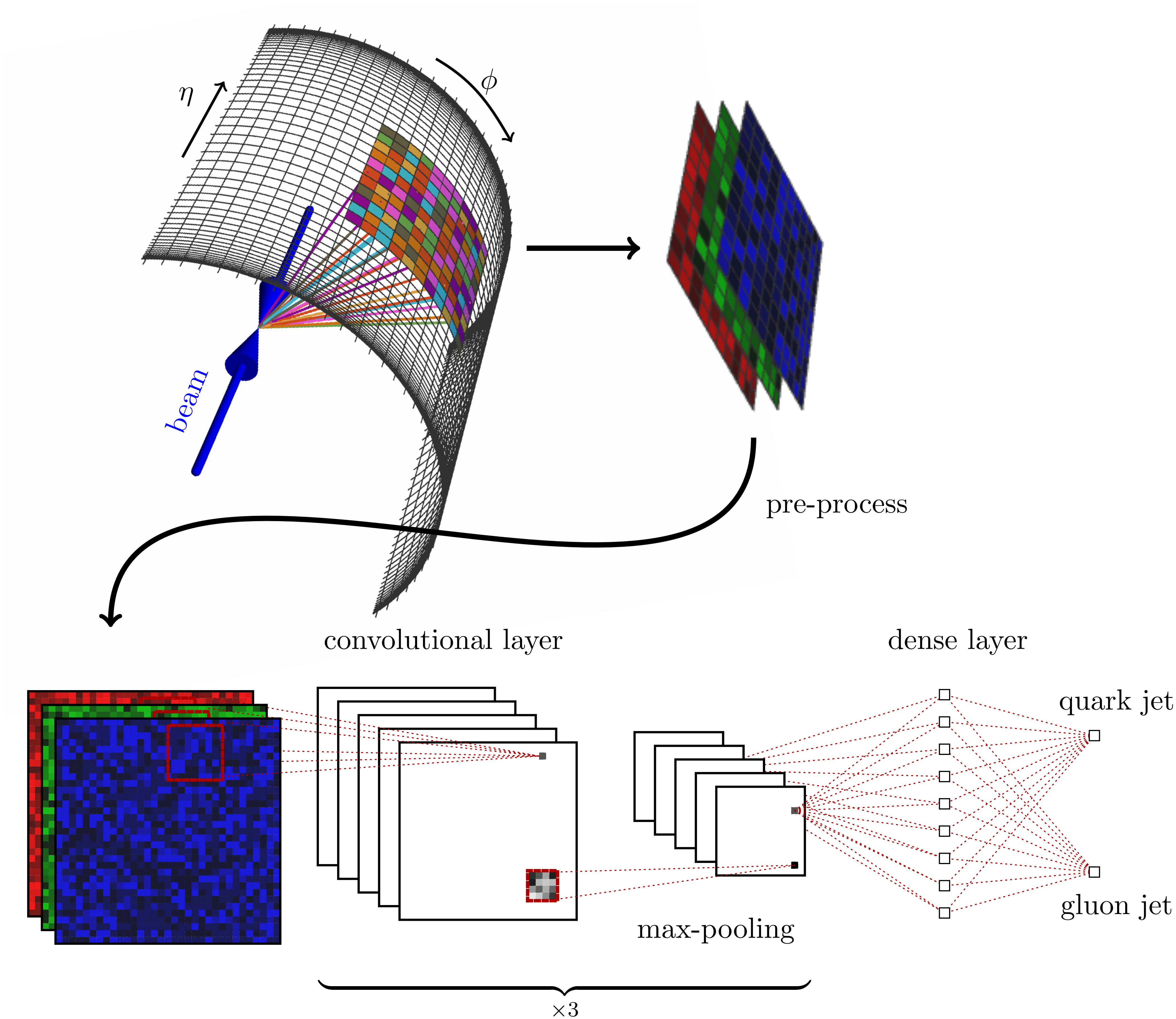


image: Komiske, Metodiev, Schwartz arxiv:1612.01551
Oliveira, et. al arXiv:1511.05190
Whiteson, et al arXiv:1603.09349
Barnard, et al arXiv:1609.00607



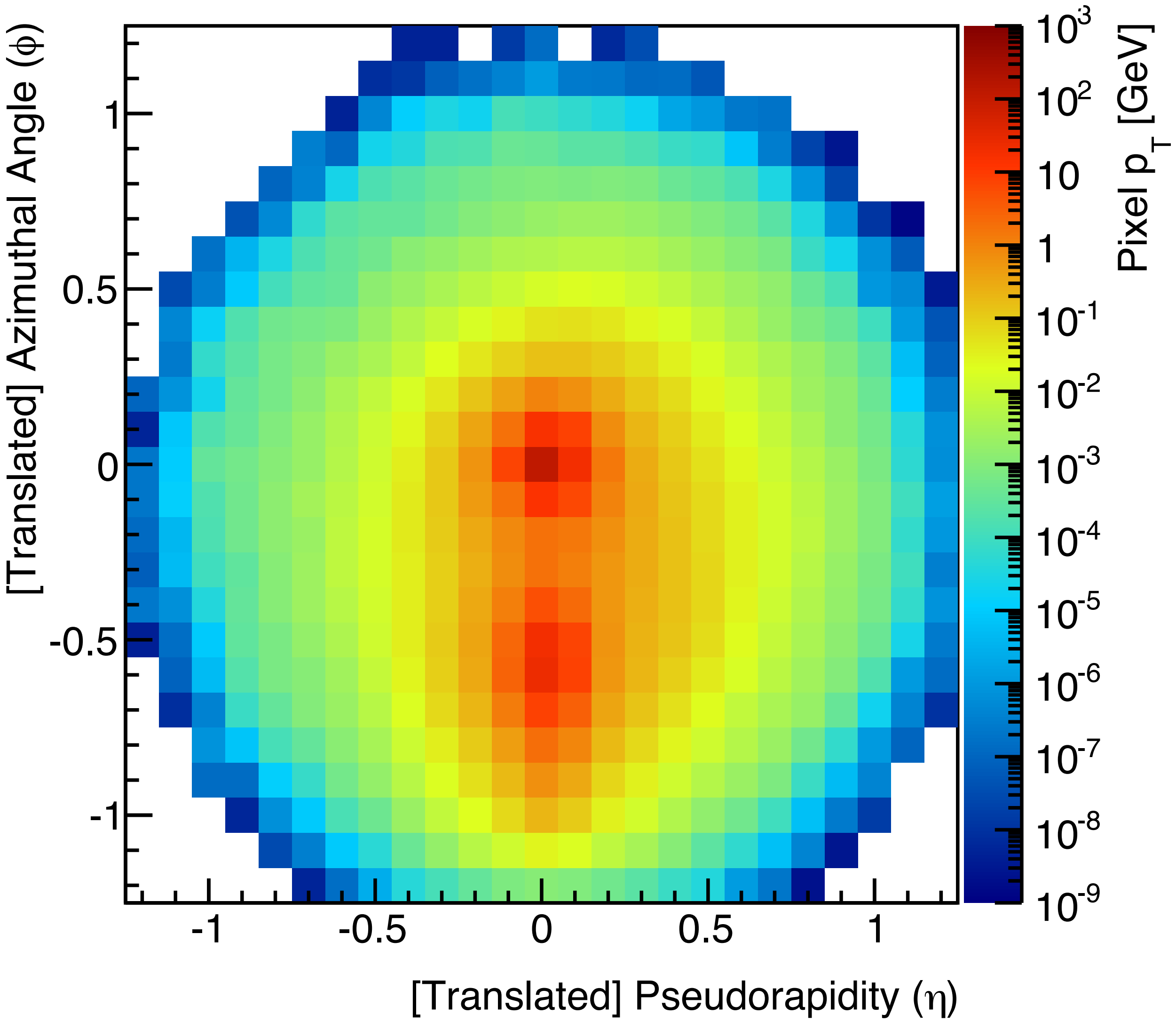
JET IMAGES

Oliveira, et. al arXiv:1511.05190
Whiteson, et al arXiv:1603.09349
Dawe, et al arXiv:1609.00607

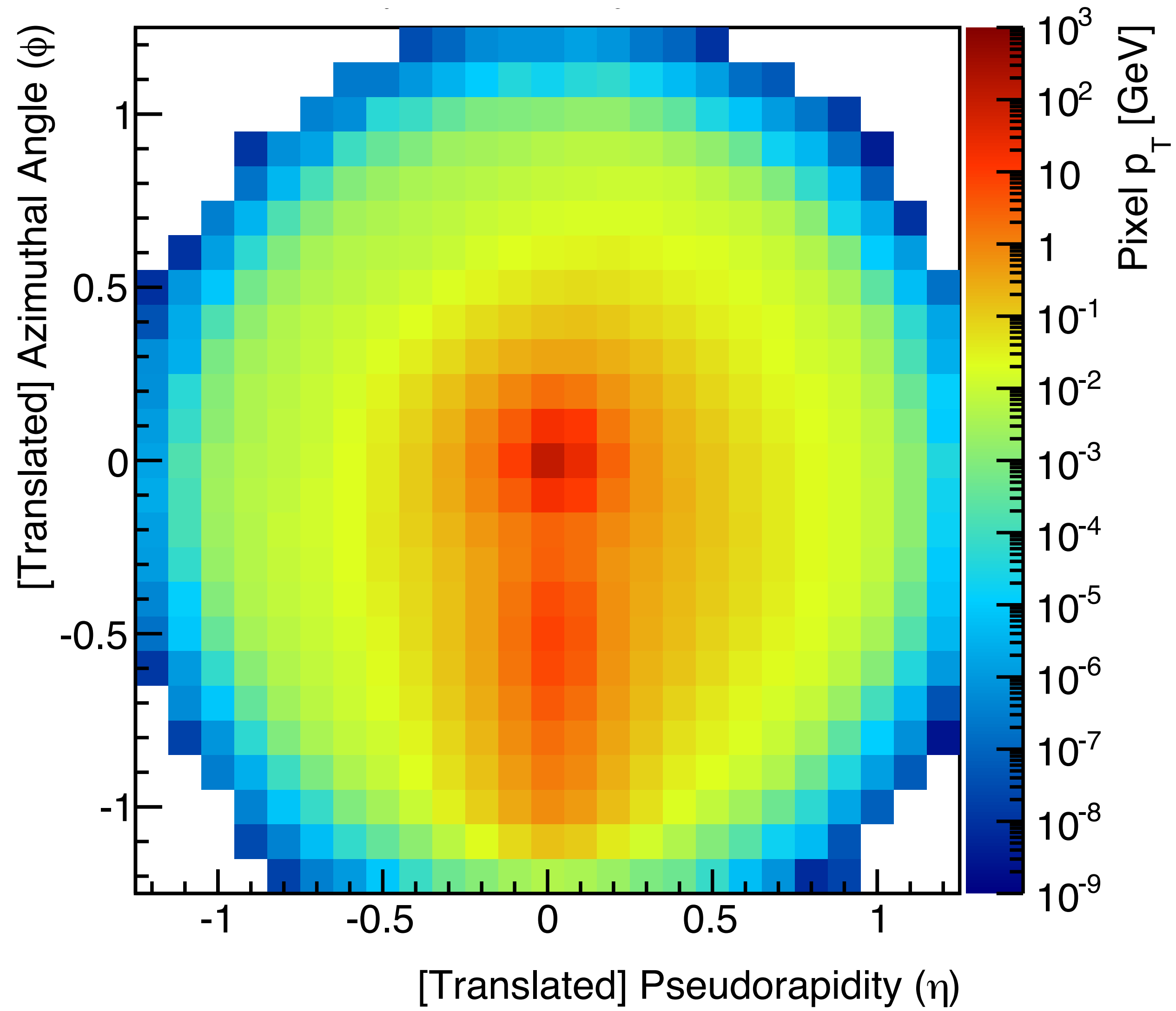
Apply deep learning algorithms to classify to "jet images"

- good results (based on fast simulation & idealized uniform calorimeter)
- preprocessed to mod out symmetries in the data
- discretization into images looses information

Average Boosted W Jet (y=1)



Average QCD Jet (y=0)



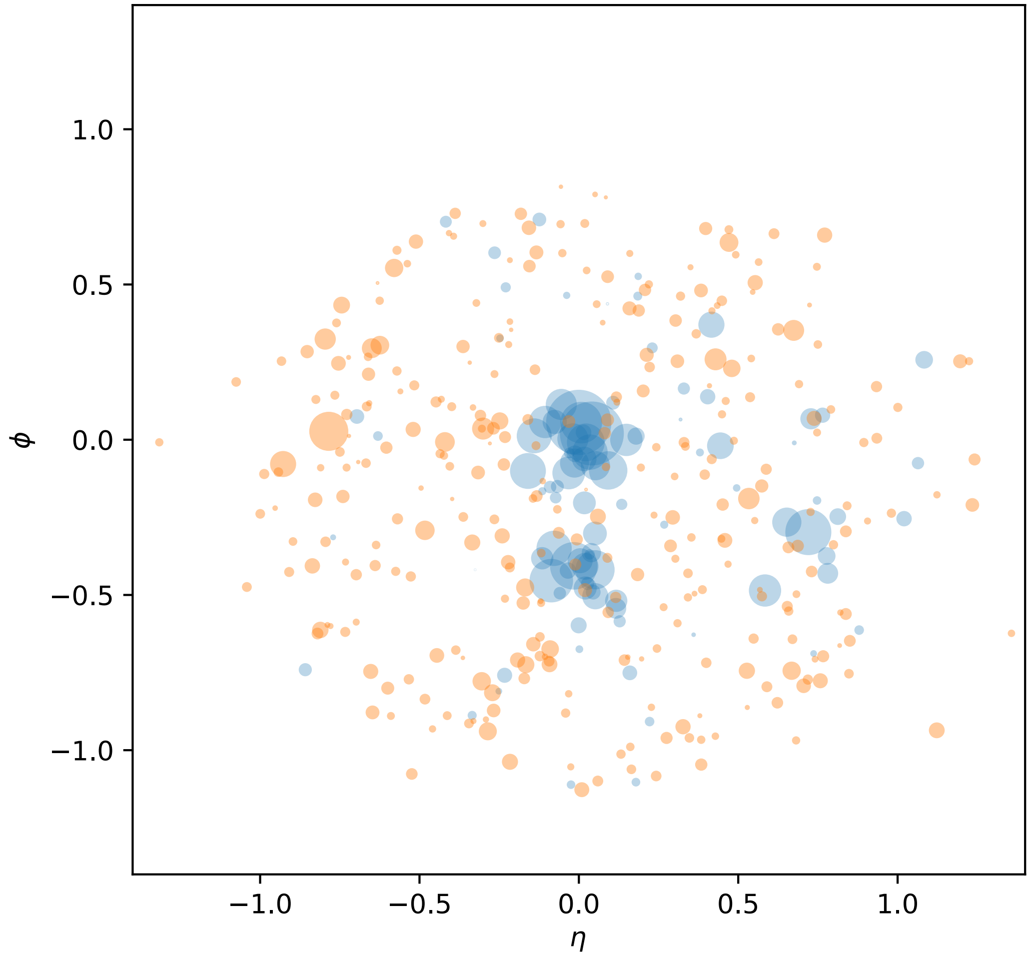
JETS AS A GRAPH

Using message passing neural networks over a fully connected graph on the particles

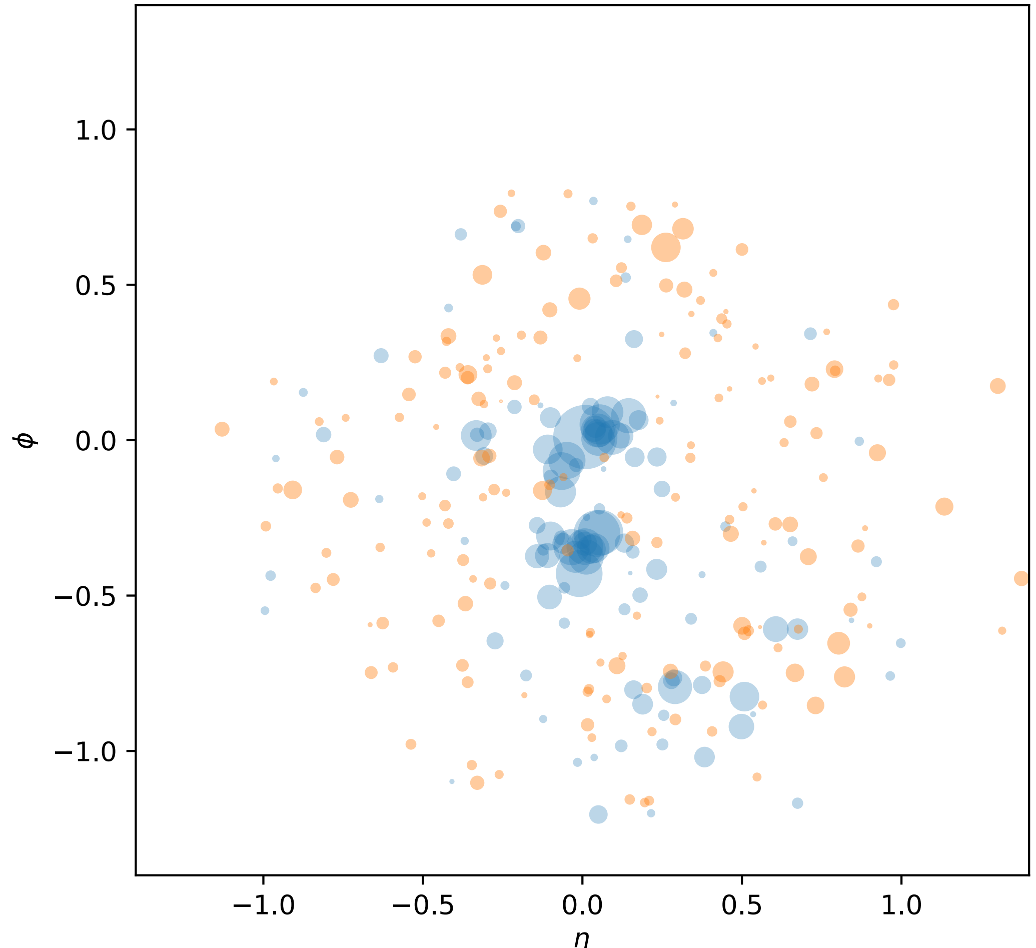
- Two approaches for adjacency matrix for edges
  - inject physics knowledge by using d_{ij} of jet algorithms
  - learn adjacency matrix and export new jet algorithm

Isaac Henrion

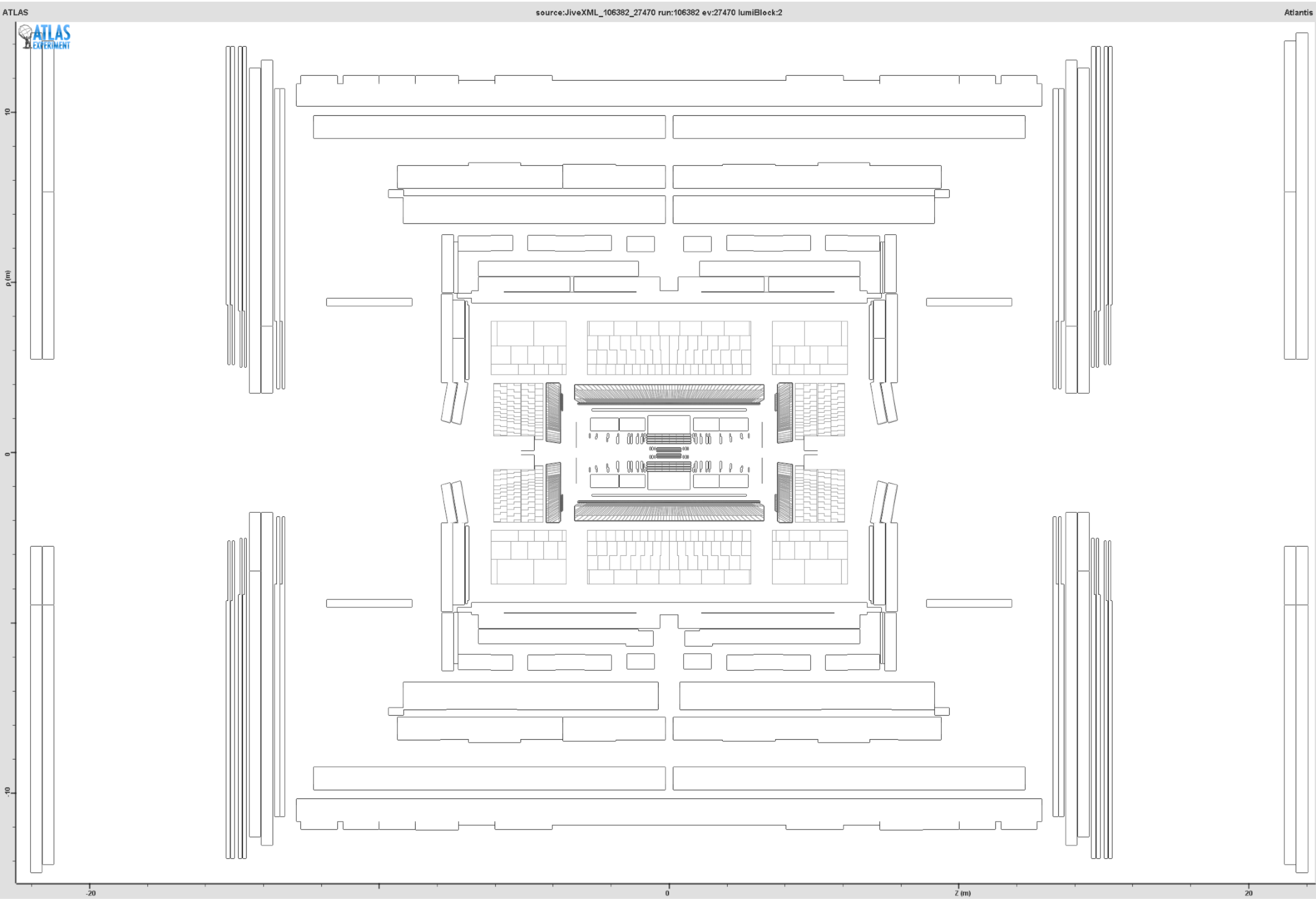
Example Boosted W Jet (y=1)



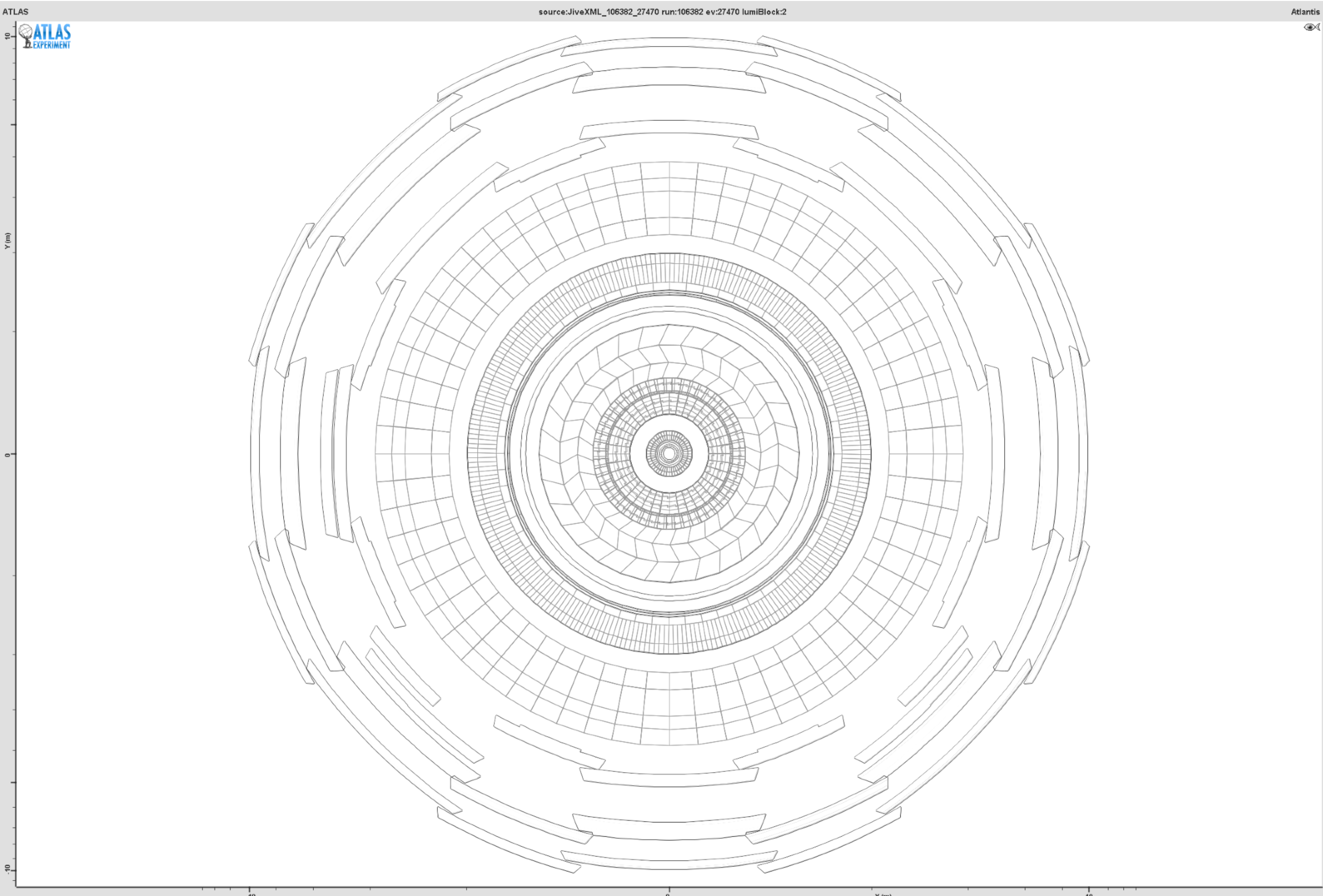
Example QCD Jet (y=0)



NON-UNIFORM GEOMETRY



NON-UNIFORM GEOMETRY



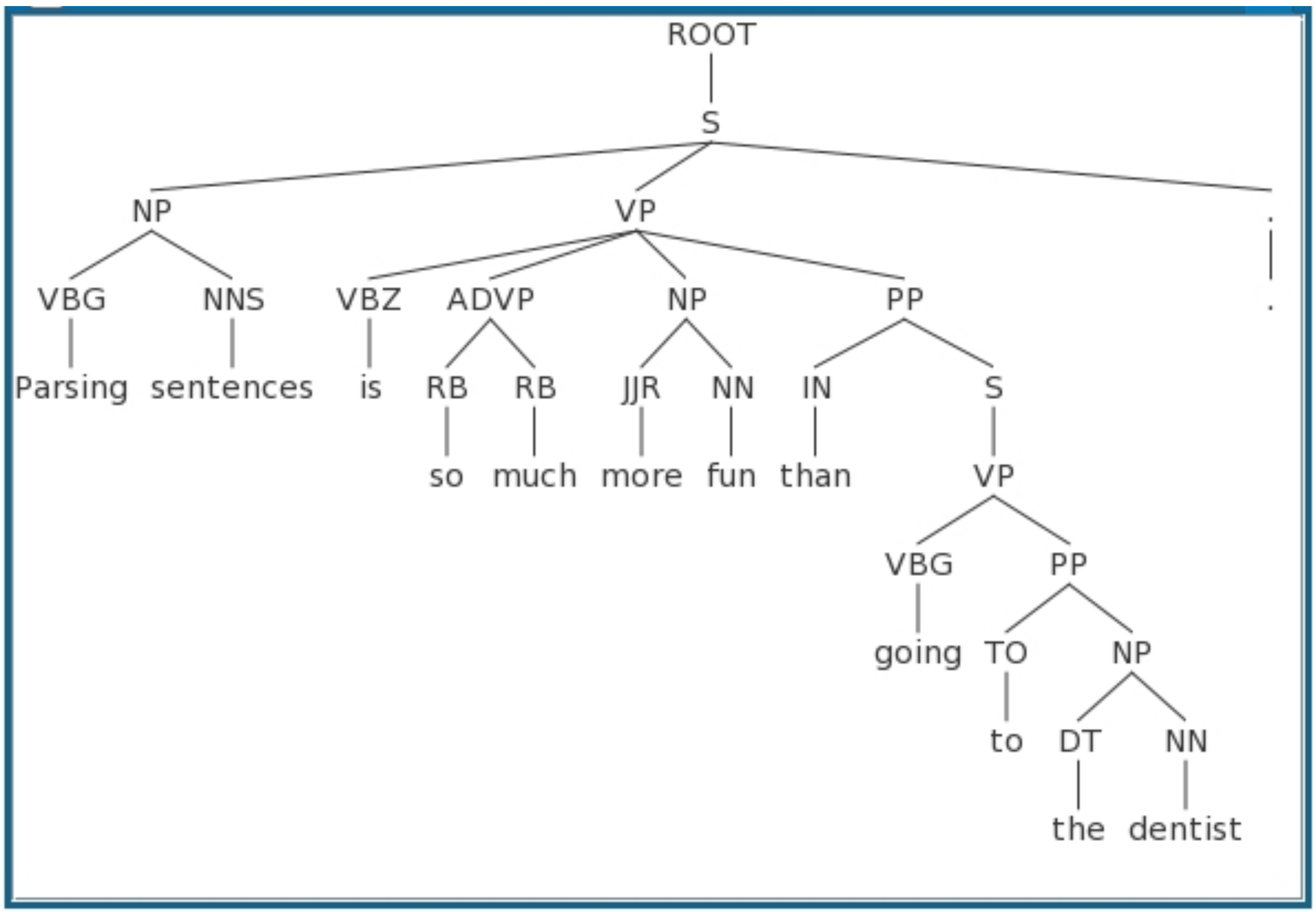
HOW CAN WE IMPROVE?

Image based approaches are doing well, but….

- would be nice to be able to work with a variable length input
  - avoid pre-processing into a regular-grid (eg. non-uniform calorimeters)
  - avoid representing empty pixels (sparse input)
- would be nice if classifier had nice theoretical properties
  - infrared & collinear safety, robustness to pileup, etc.
- would be nice to be more data efficient, most image-based networks use a LOT of training data.

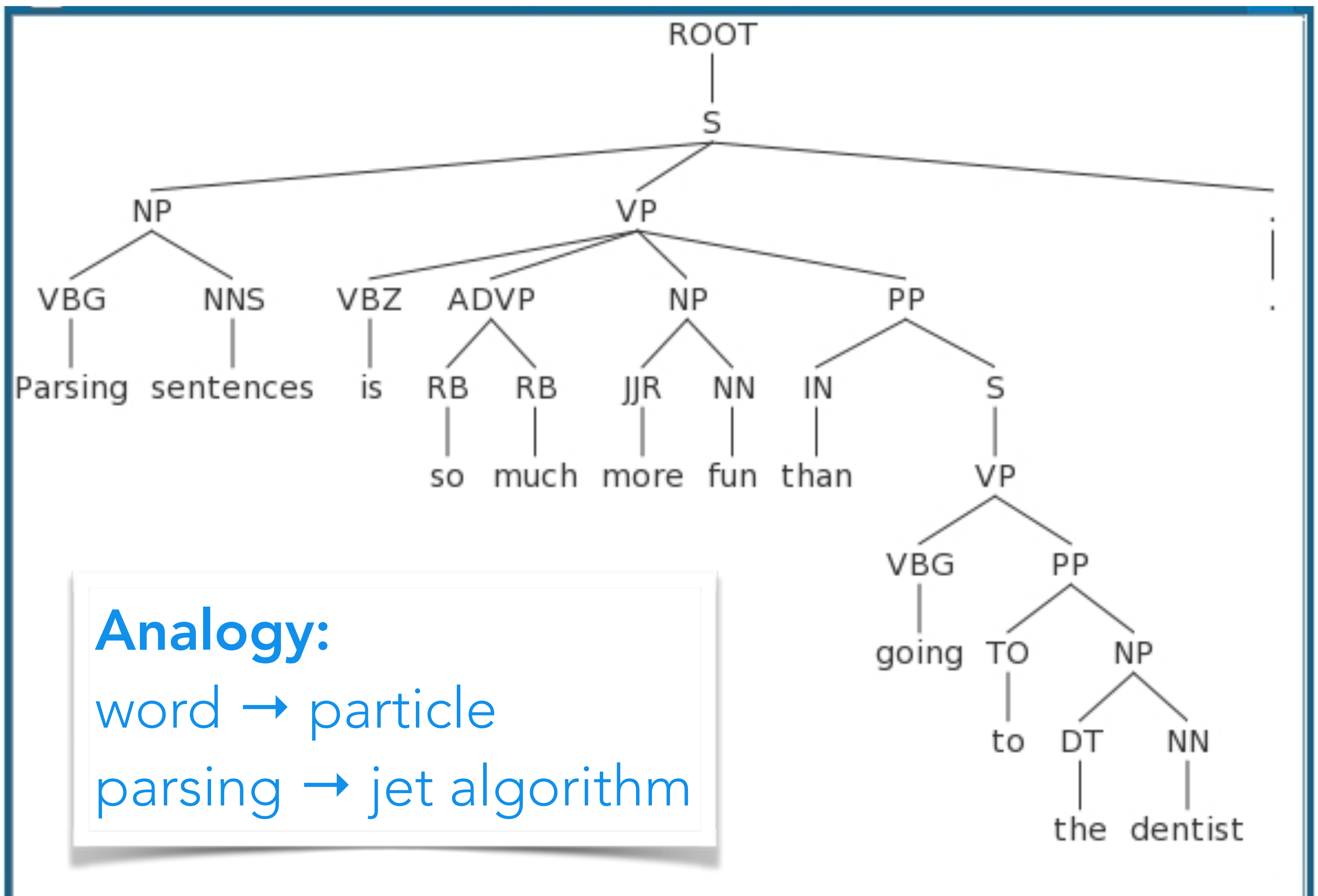
Recursive Neural Networks showing great performance for Natural Language Processing tasks

- neural network's topology given by parsing of sentence!



Recursive Neural Networks showing great performance for Natural Language Processing tasks

- neural network's topology given by parsing of sentence!

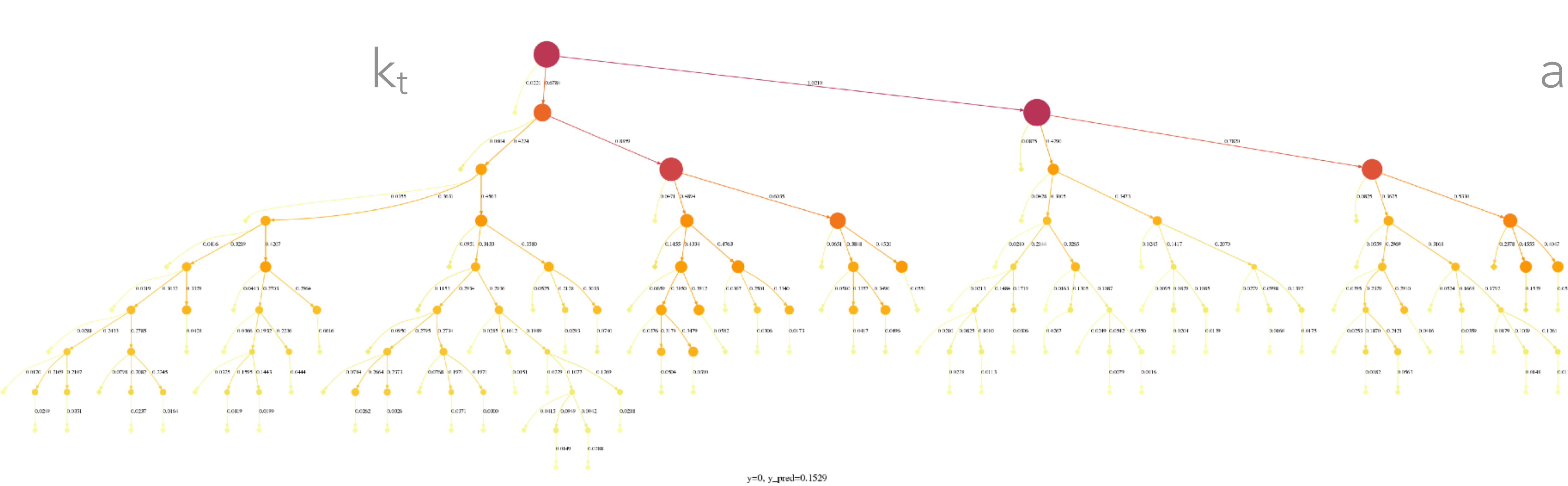


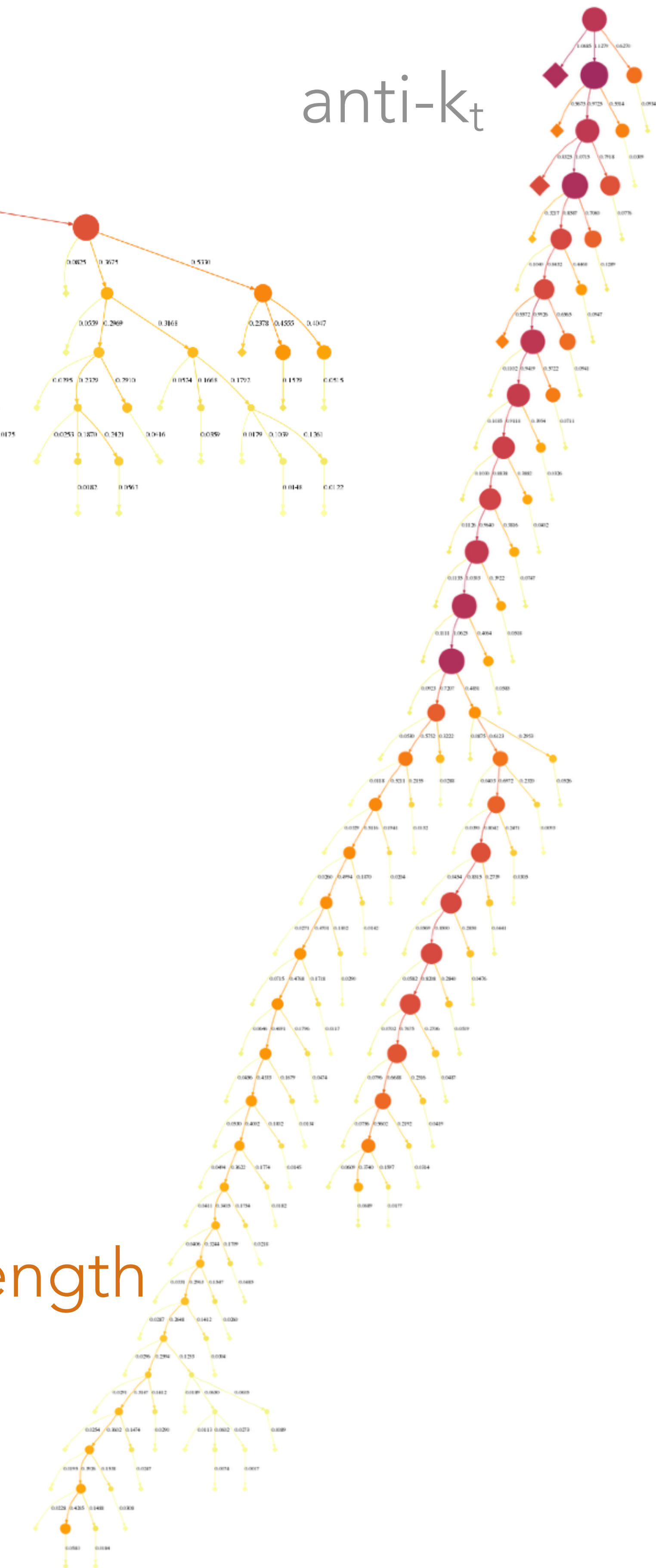
Analogy:

word → particle

parsing → jet algorithm

QCD-INSPIRED RECURSIVE NEURAL NETWORKS



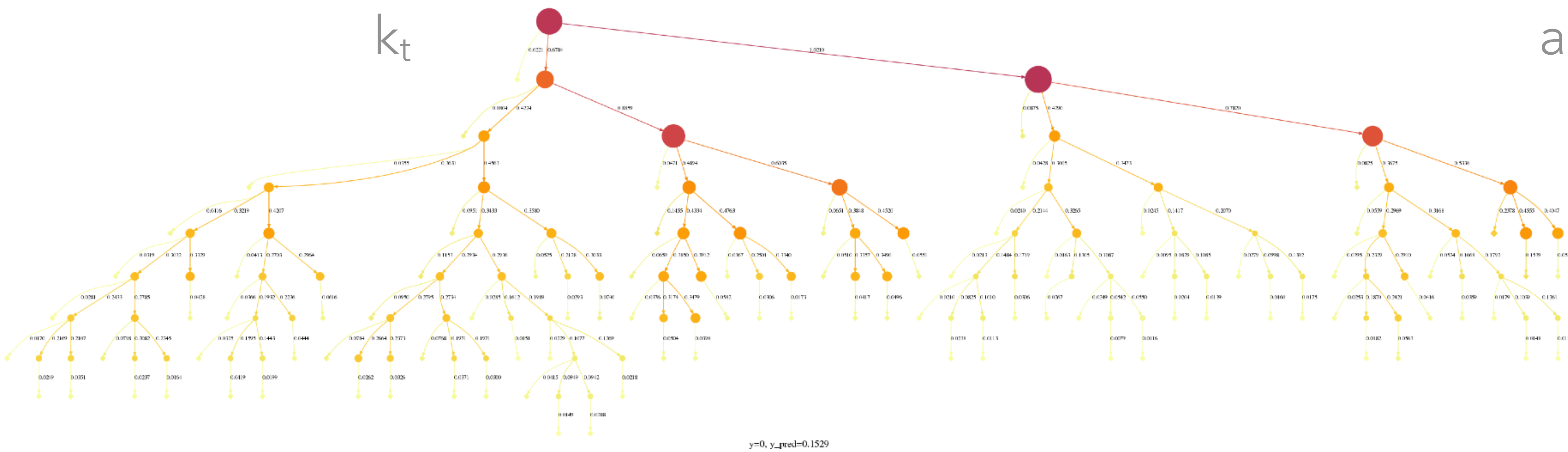


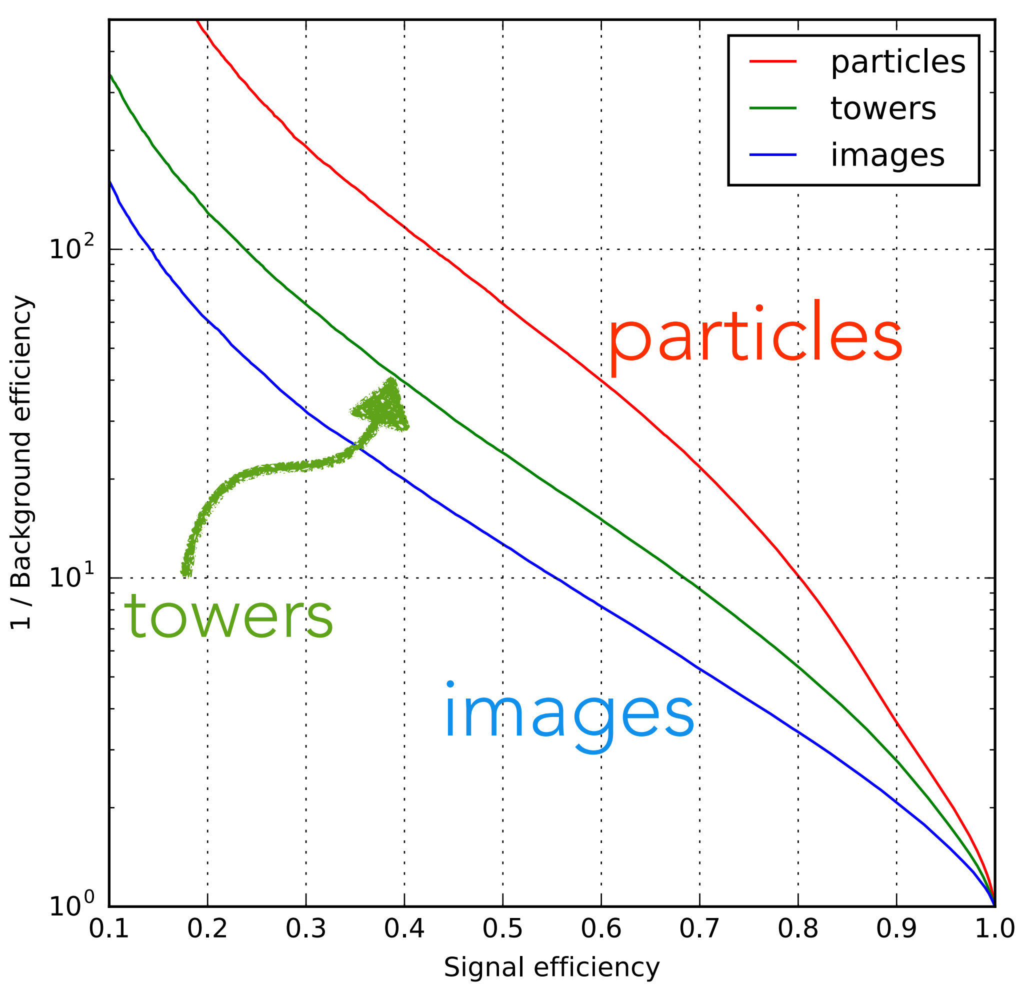
Work with Gilles Louppe, Kyunghyun Cho, Cyril Becot

- Use sequential recombination jet algorithms to provide network topology (**on a per-jet basis**)
- path towards ML models with good theoretical properties
- Top node of recursive network provides a fixed-length **embedding** of a jet that can be fed to a classifier

arXiv:1702.00748 & follow up work with Joan Bruna using graph conv nets

QCD-INSPIRED RECURSIVE NEURAL NETWORKS



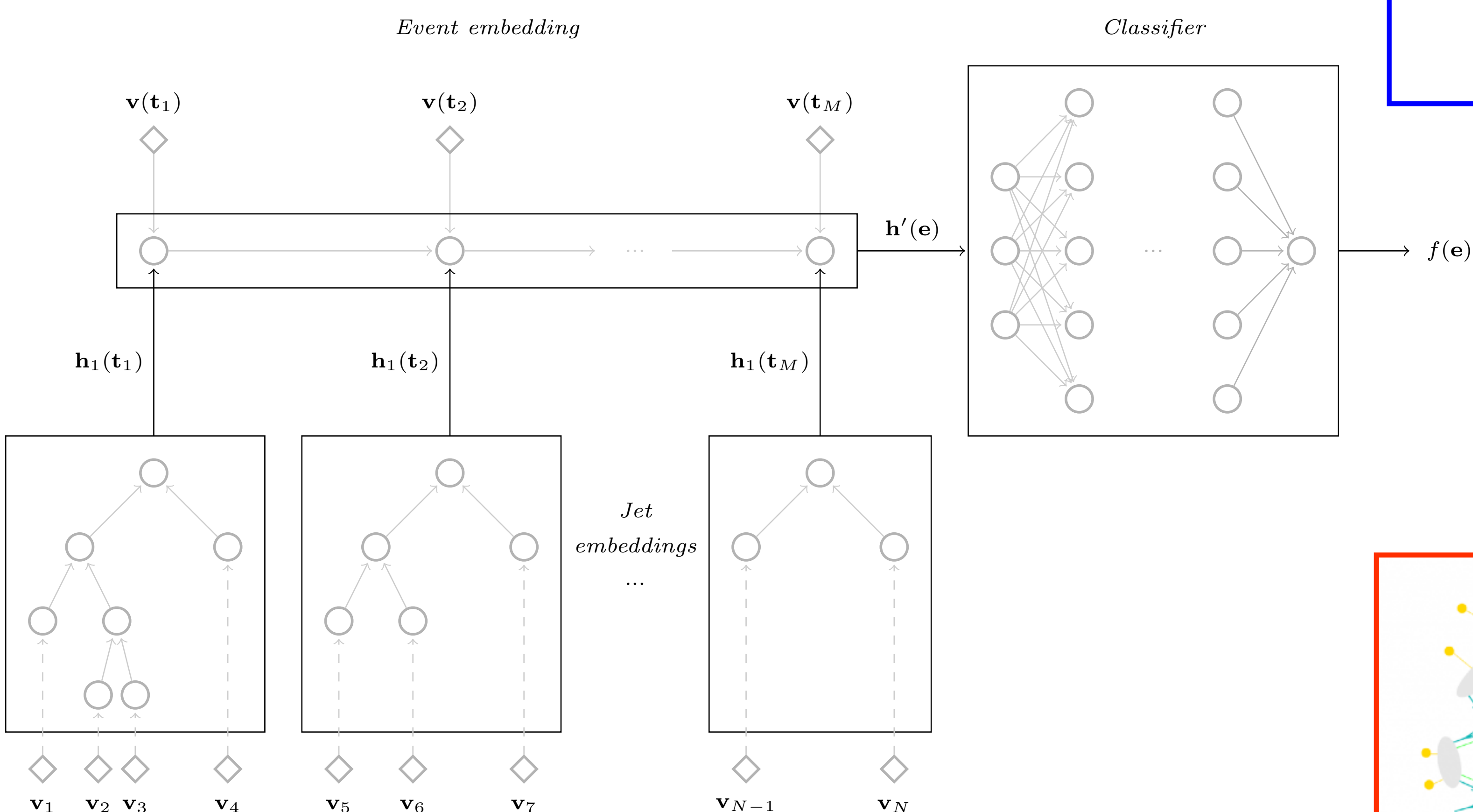


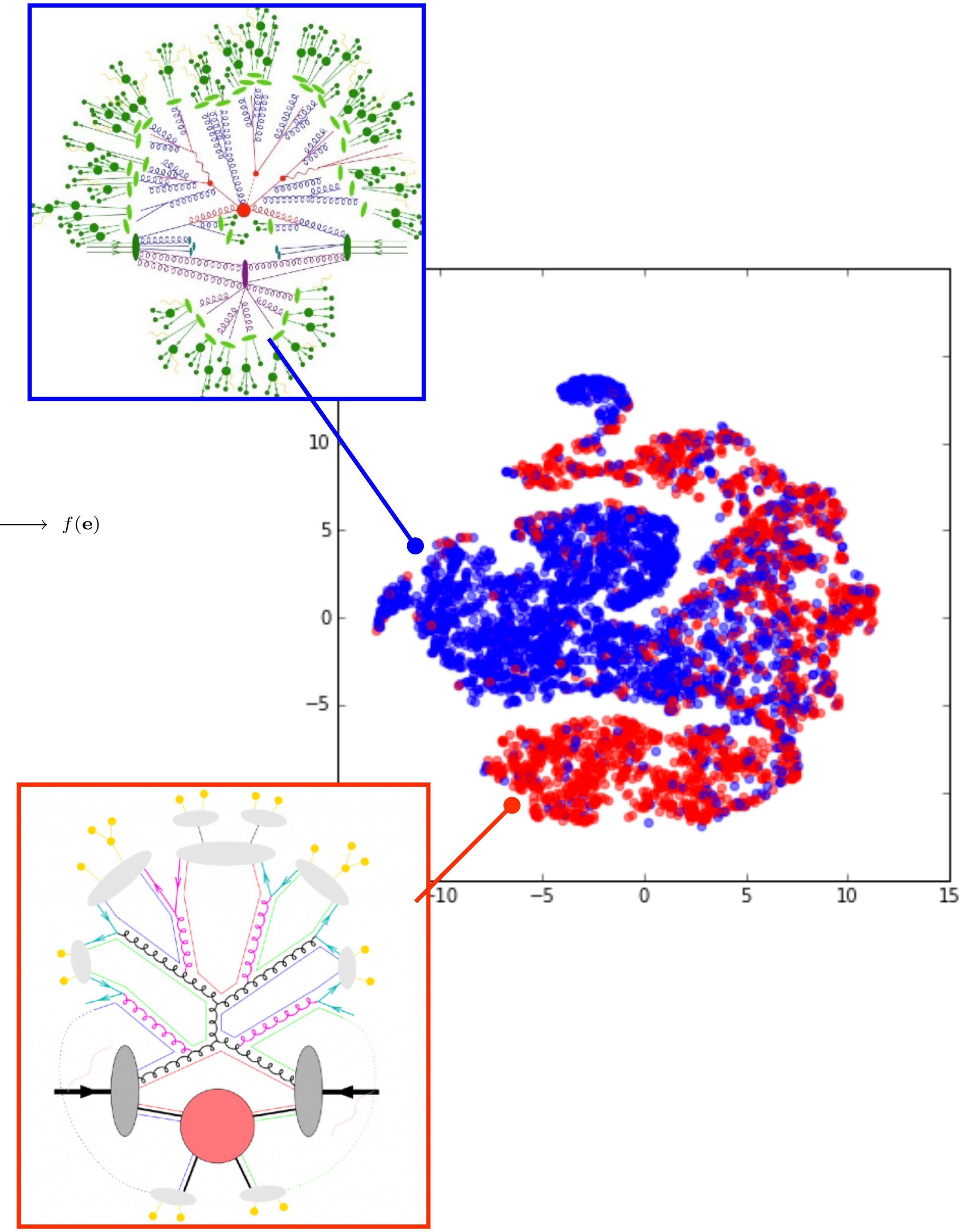
- W-jet tagging example using data from Dawe, et al arXiv:1609.00607
- down-sampling by projecting into images looses information
- RNN needs much less data to train!

HIERARCHICAL MODEL FOR THE ENTIRE EVENT

particle embedding → jet embedding → event embedding → classifier

It scales!





arXiv:1702.00748 & follow up work with Joan Bruna using graph conv nets

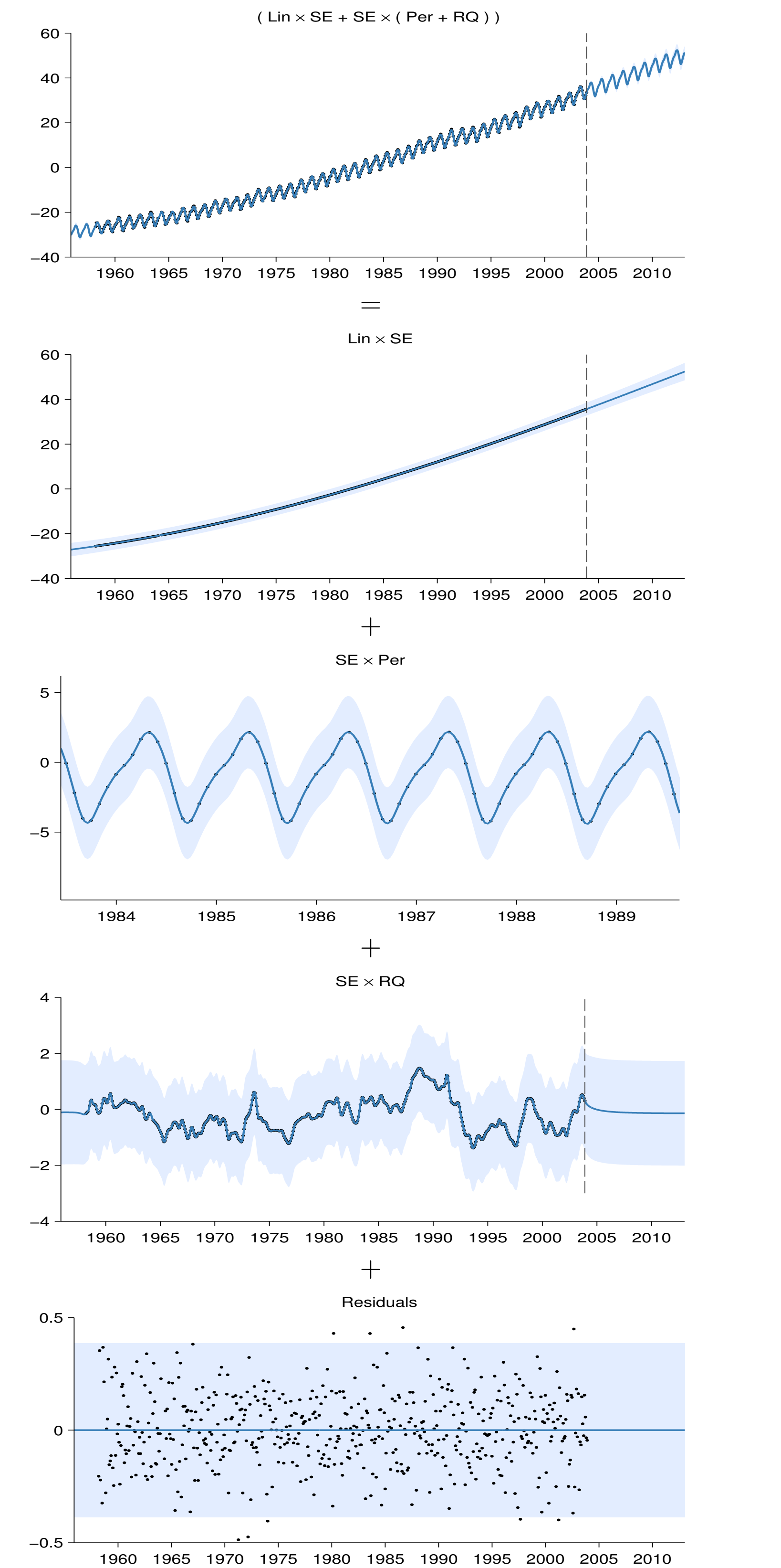
Physics Aware

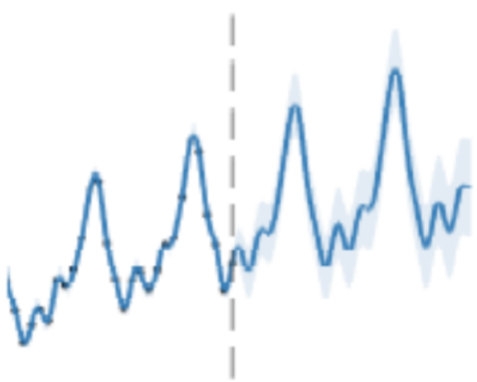
FUTURE DIRECTIONS

Vocabulary of kernels + grammar for composition

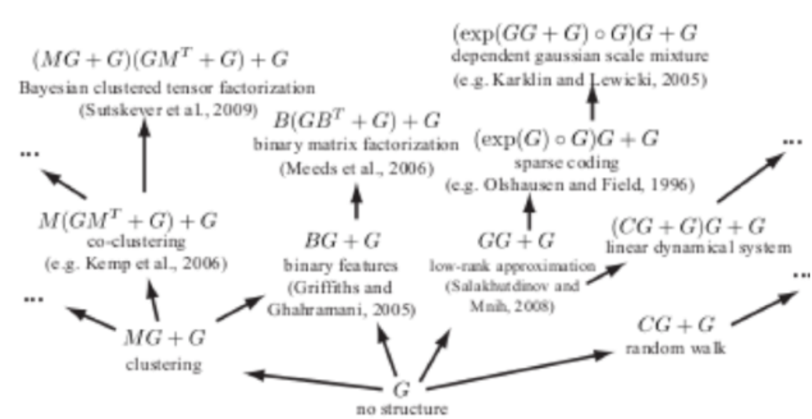
- physics goes into the construction of a "Kernel" that describes covariance of data

Mauna Loa atmospheric CO_2





Structure Discovery in Nonparametric Regression through Compositional Kernel Search
David Duvenaud, James Robert Lloyd, Roger Grosse, Joshua B. Tenenbaum, Zoubin Ghahramani
International Conference on Machine Learning, 2013
pdf | code | poster | bibtex

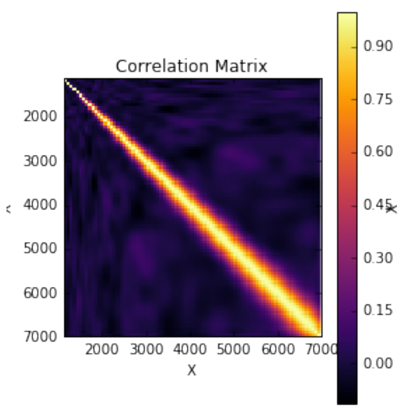


Exploiting compositionality to explore a large space of model structures
Roger Grosse, Ruslan Salakhutdinov, William T. Freeman, Joshua B. Tenenbaum
Conference on Uncertainty in Artificial Intelligence, 2012
pdf | code | bibtex

FUTURE DIRECTIONS

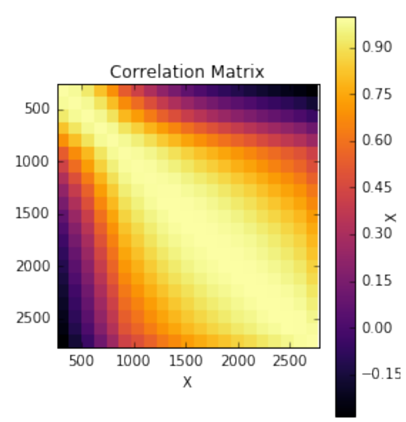
with Meghan Frate
& Daniel Whiteson

Instead of fitting the dijet spectrum with an ad hoc 3-5 parameter function, use GP with kernel motivated from physics



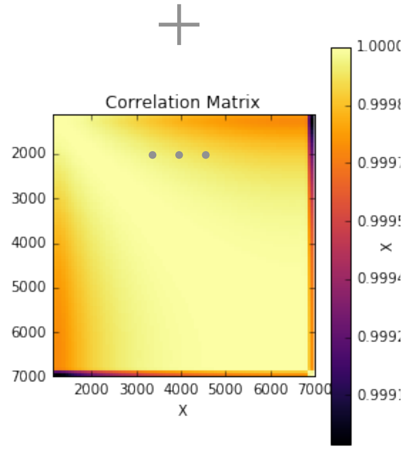
Final Kernel =

=



Poisson stats
+ Mass Resolution

+

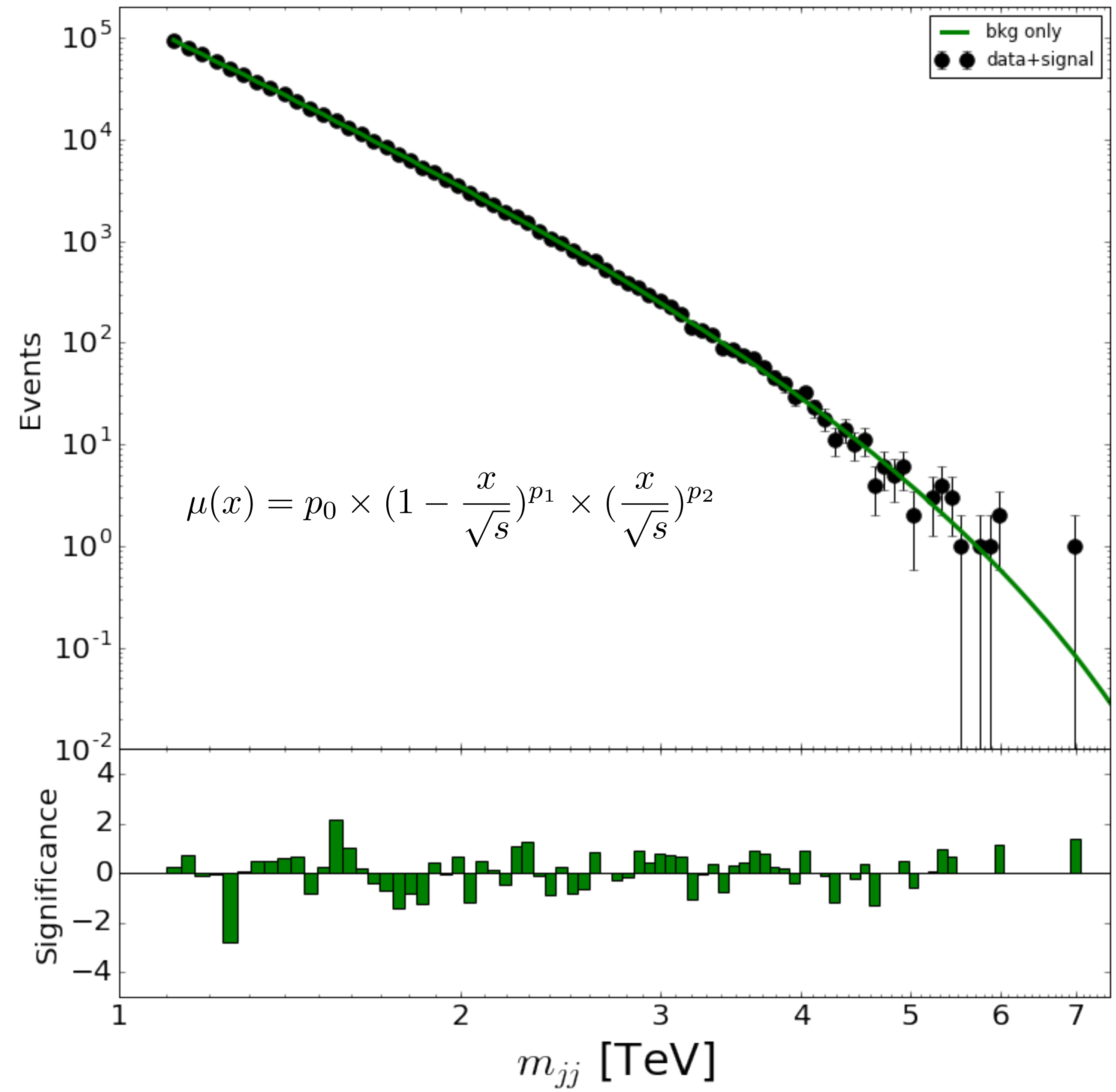


+ Parton Density Functions

+ Jet Energy Scale

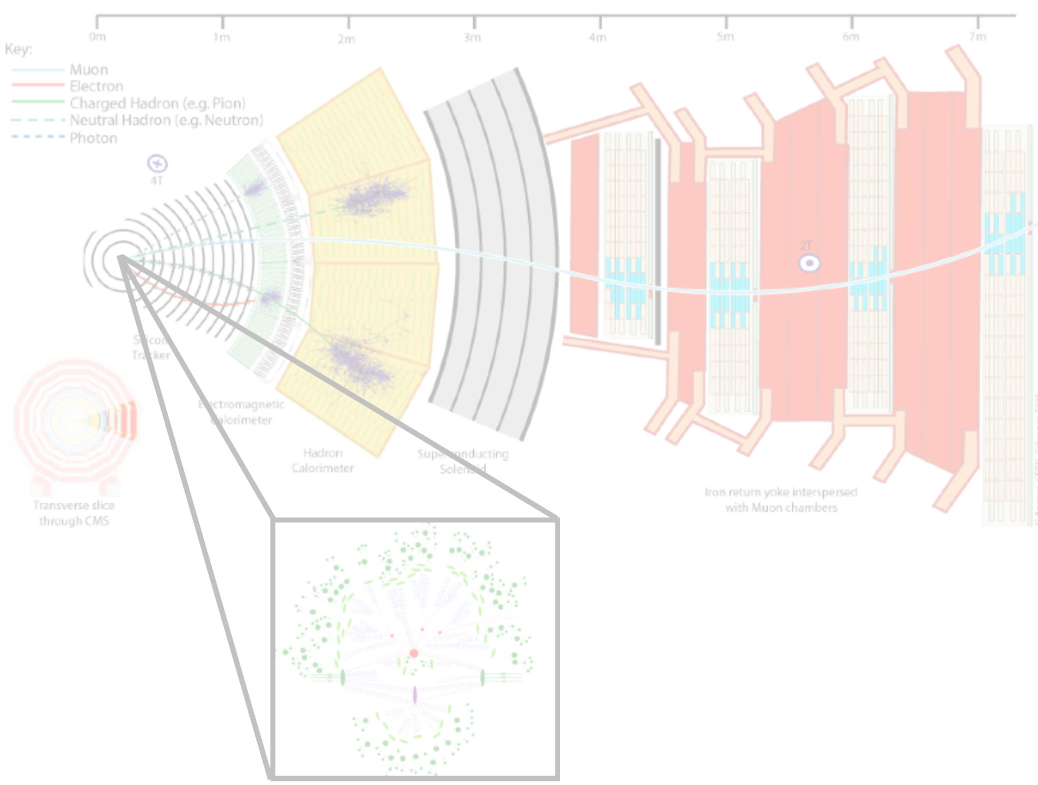
+

. . .



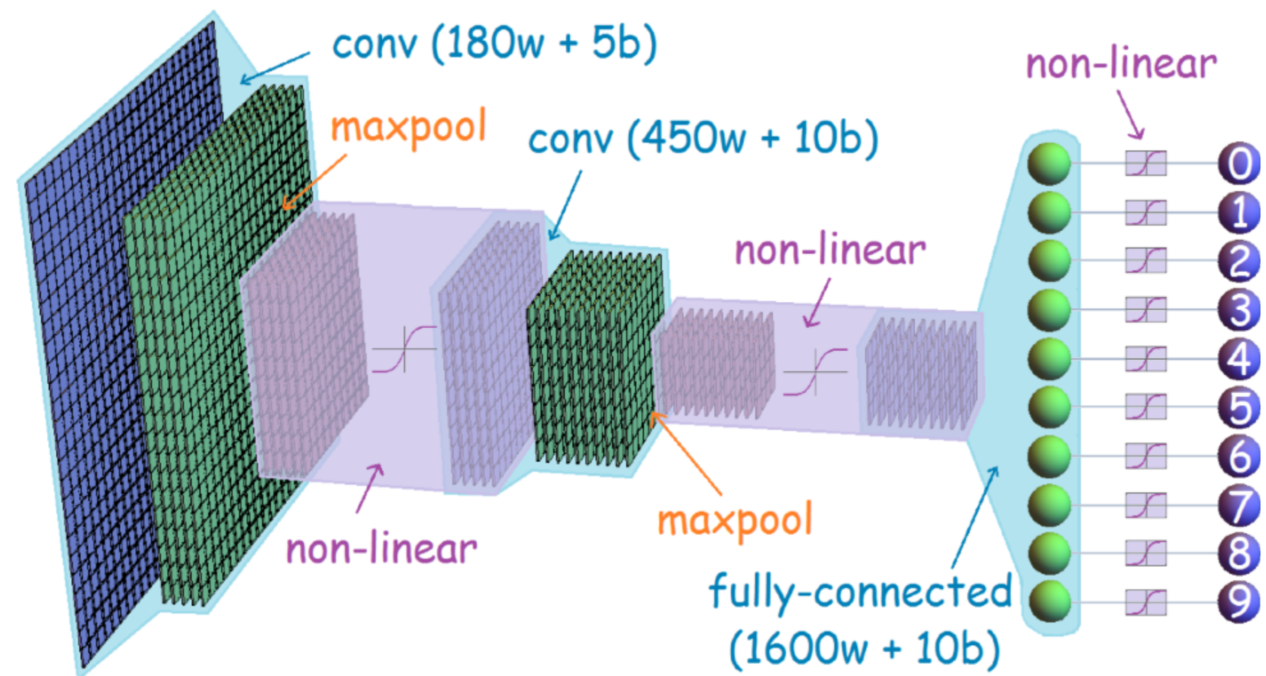
TWO APPROACHES

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

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

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

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

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

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

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

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

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

- 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 $r(x) = p(x)$

How do we create complicated probability densities $p(x)$ that are tractable

and

are normalized such that $\int p(x)\, dx = 1$?

K.C. & G. Louppe: http://beta.briefideas.org/ideas/5c2f74aedbf3618ca180382e393c7617

If I have a bijection: $f : X \to Z$

and an arbitrary tractable density on Z: $p(z)$

Then density on X follows from a simple change of variables

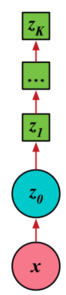
$$p(x) = p(f_\phi(x)) \left| \det \left(\frac{\partial f_\phi(x)}{\partial x_T} \right) \right|$$

Now construct neural networks f_ϕ that are bijections & optimize "cross entropy" loss

If it is a bijection, I can generate samples of x from inverse transformation $f^{-1}(z)$

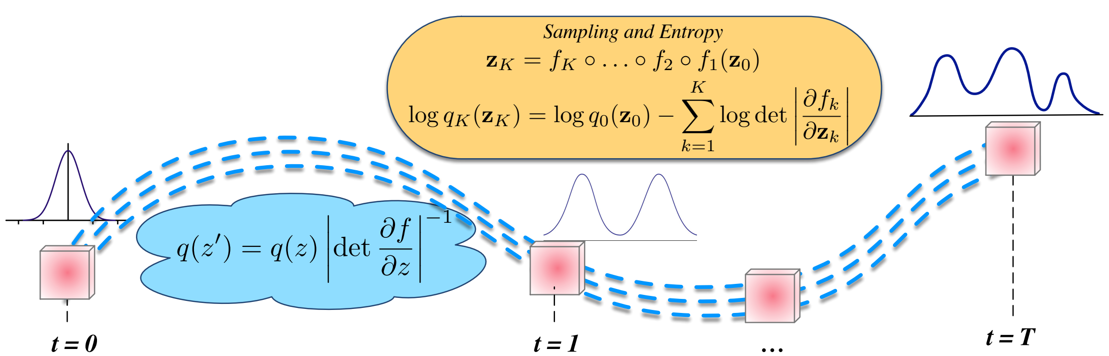
ENGINEERING BIJECTIONS

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.



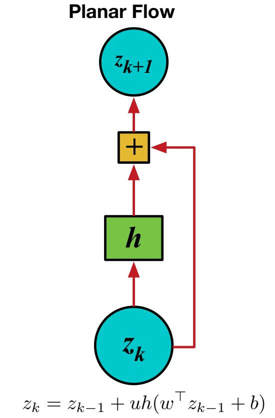
Distribution flows through a sequence of invertible transforms

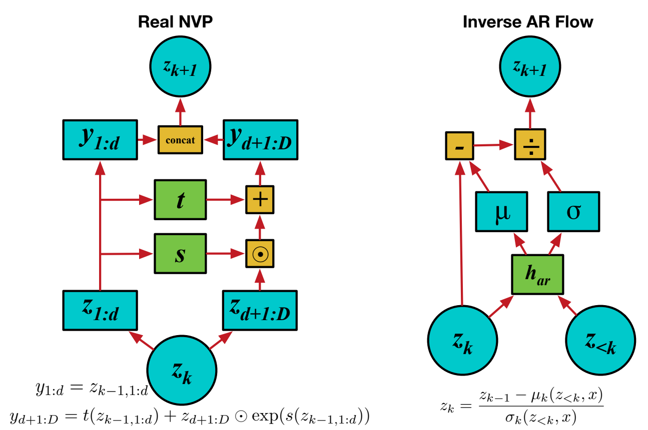
[Rezende and Mohamed, 2015]

Choice of Transformation Function

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

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

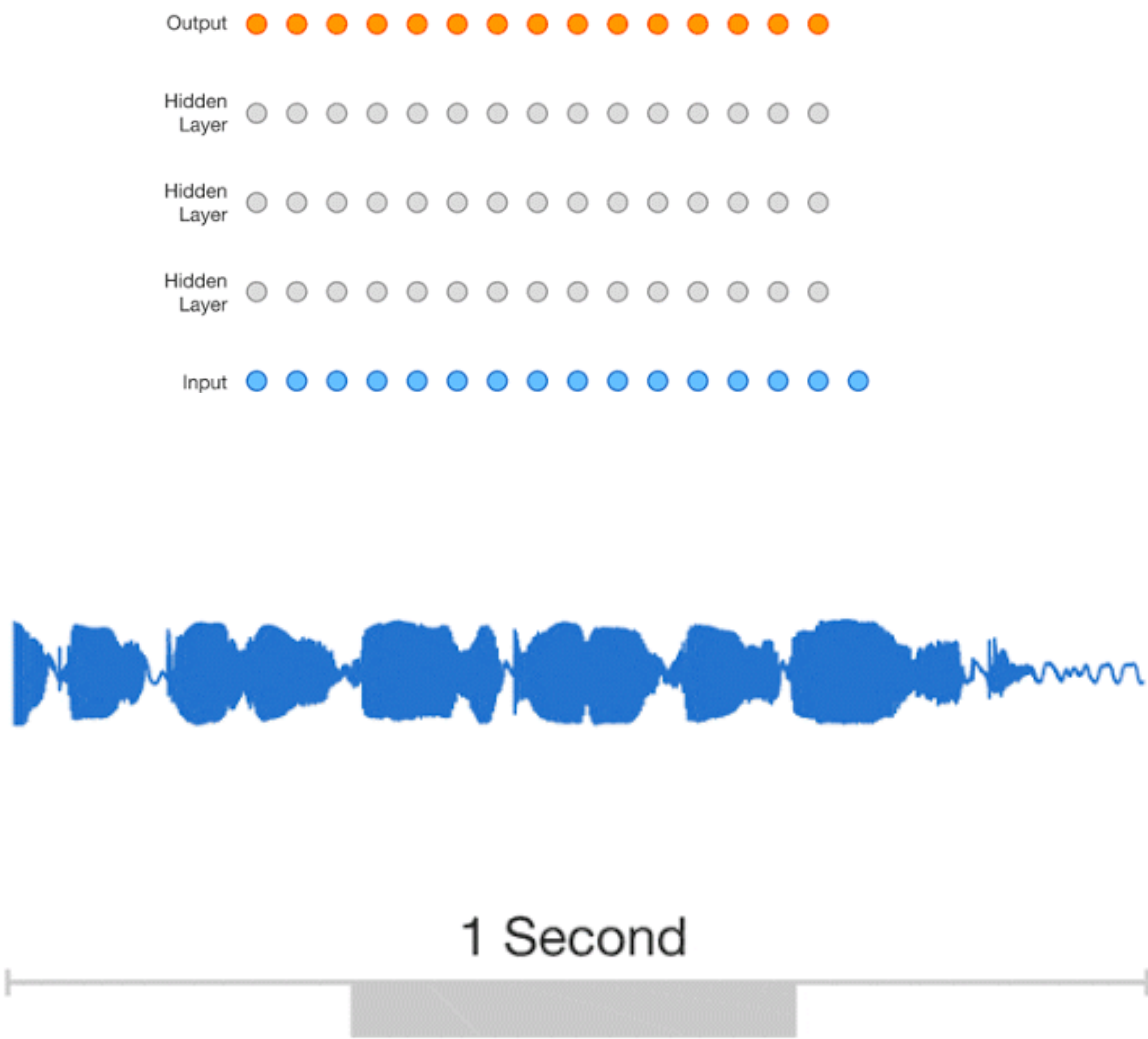




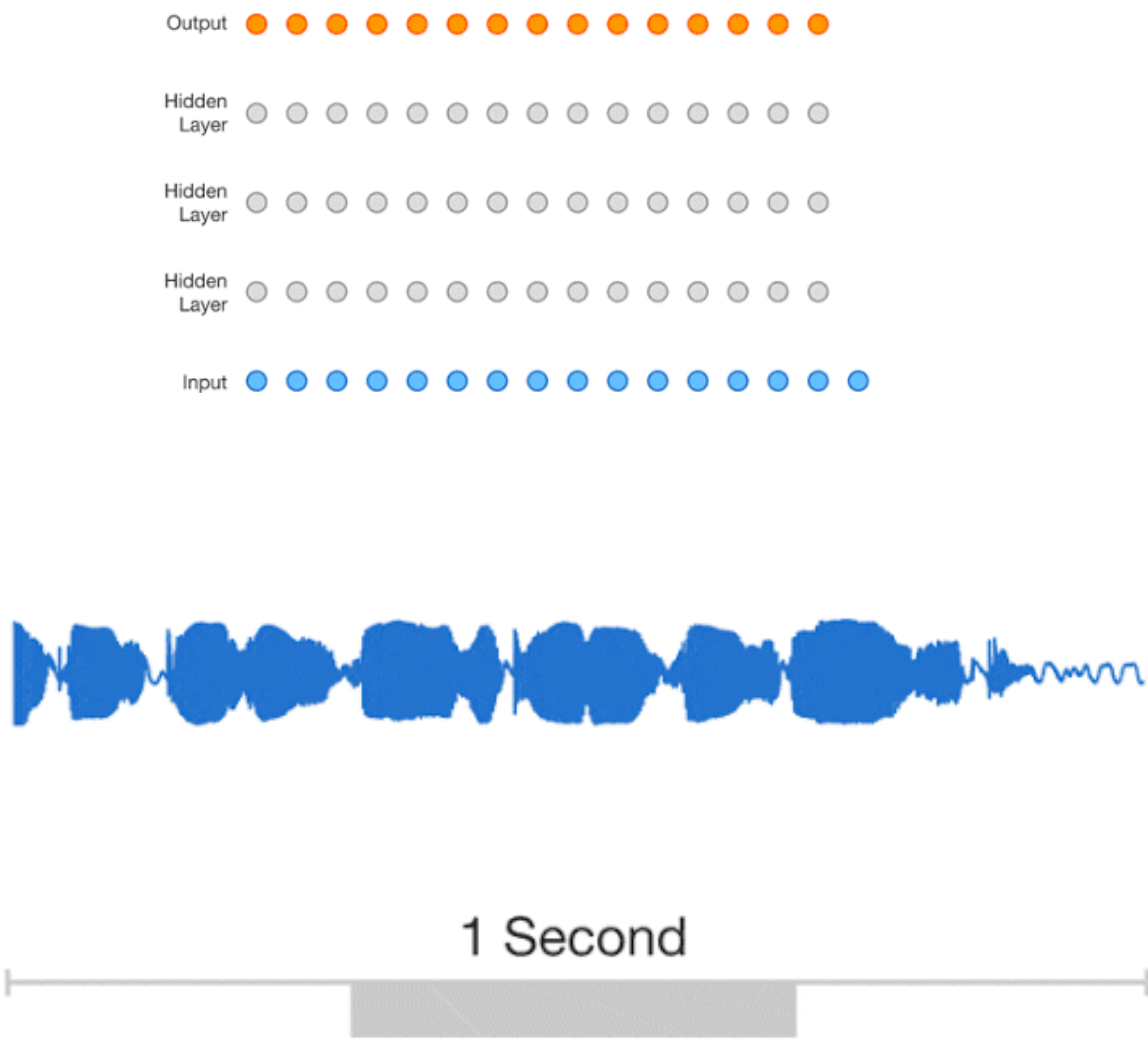
[Rezende and Mohamed, 2016; Dinh et al., 2016; Kingma et al., 2016]

Linear time computation of the determinant and its gradient.

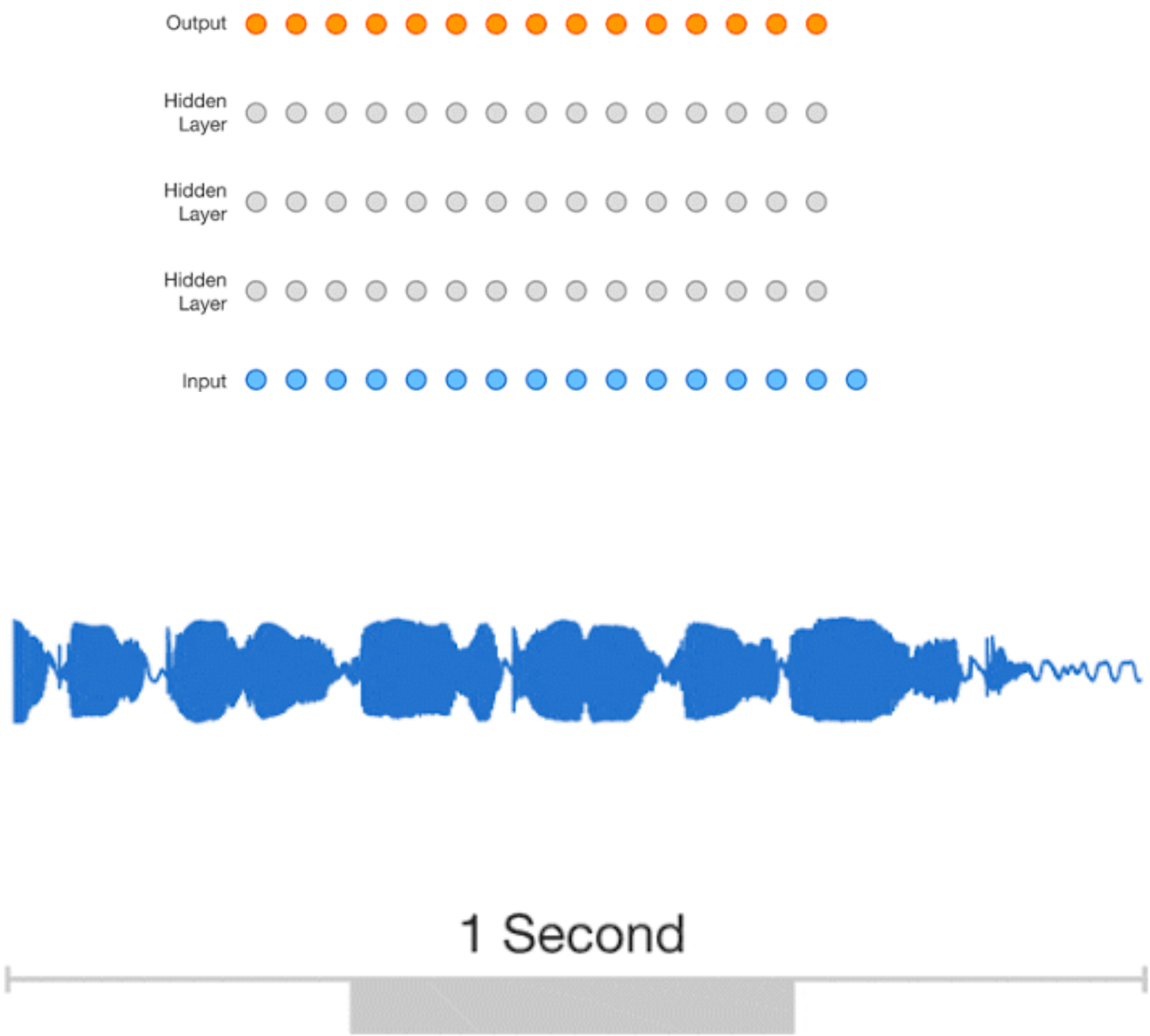
WAVENET: A GENERATIVE MODEL FOR RAW AUDIO



WAVENET: A GENERATIVE MODEL FOR RAW AUDIO

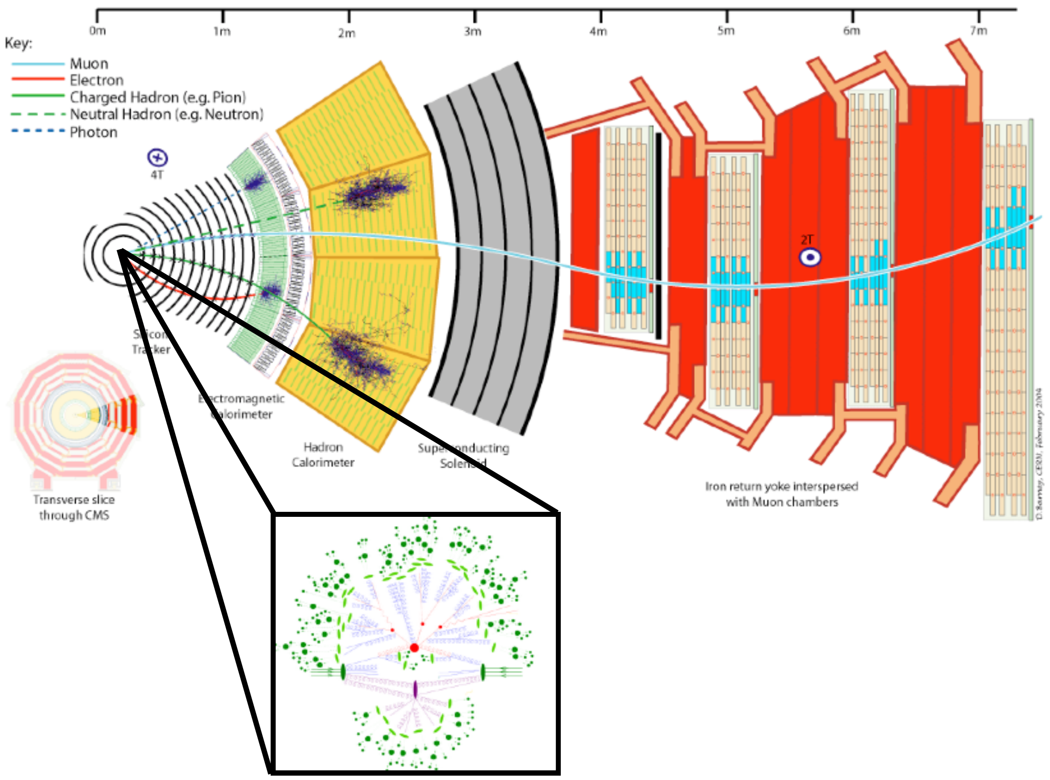


WAVENET: A GENERATIVE MODEL FOR RAW AUDIO



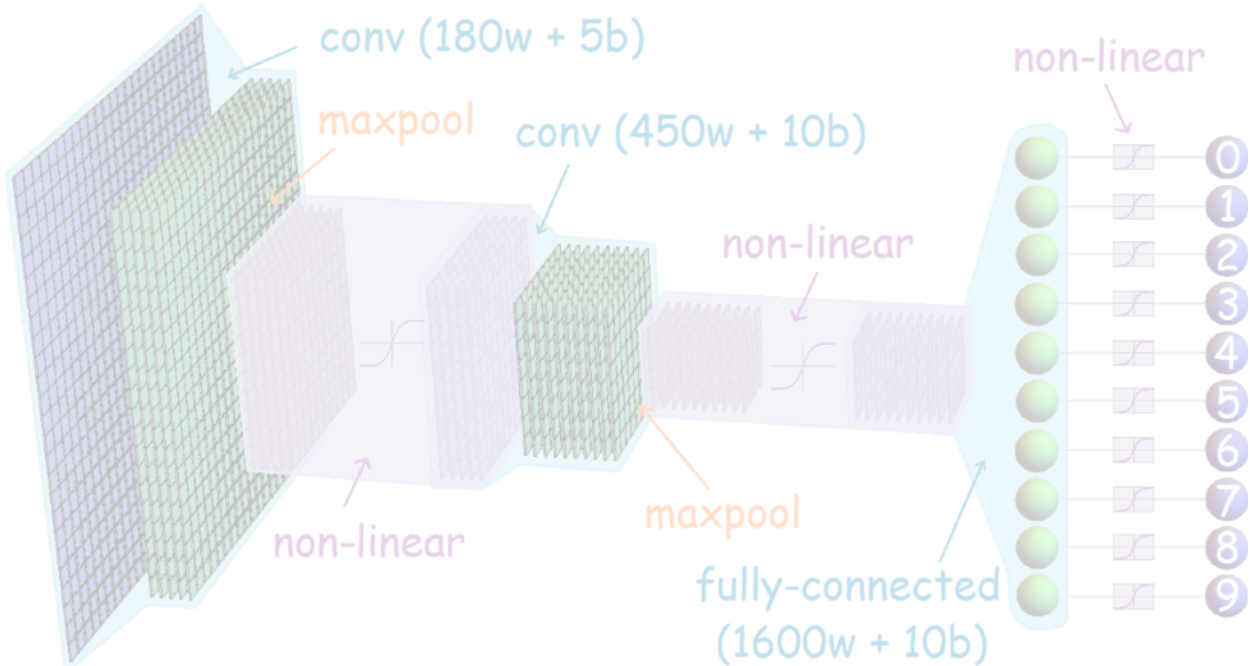
TWO APPROACHES

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

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 mini-max 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.

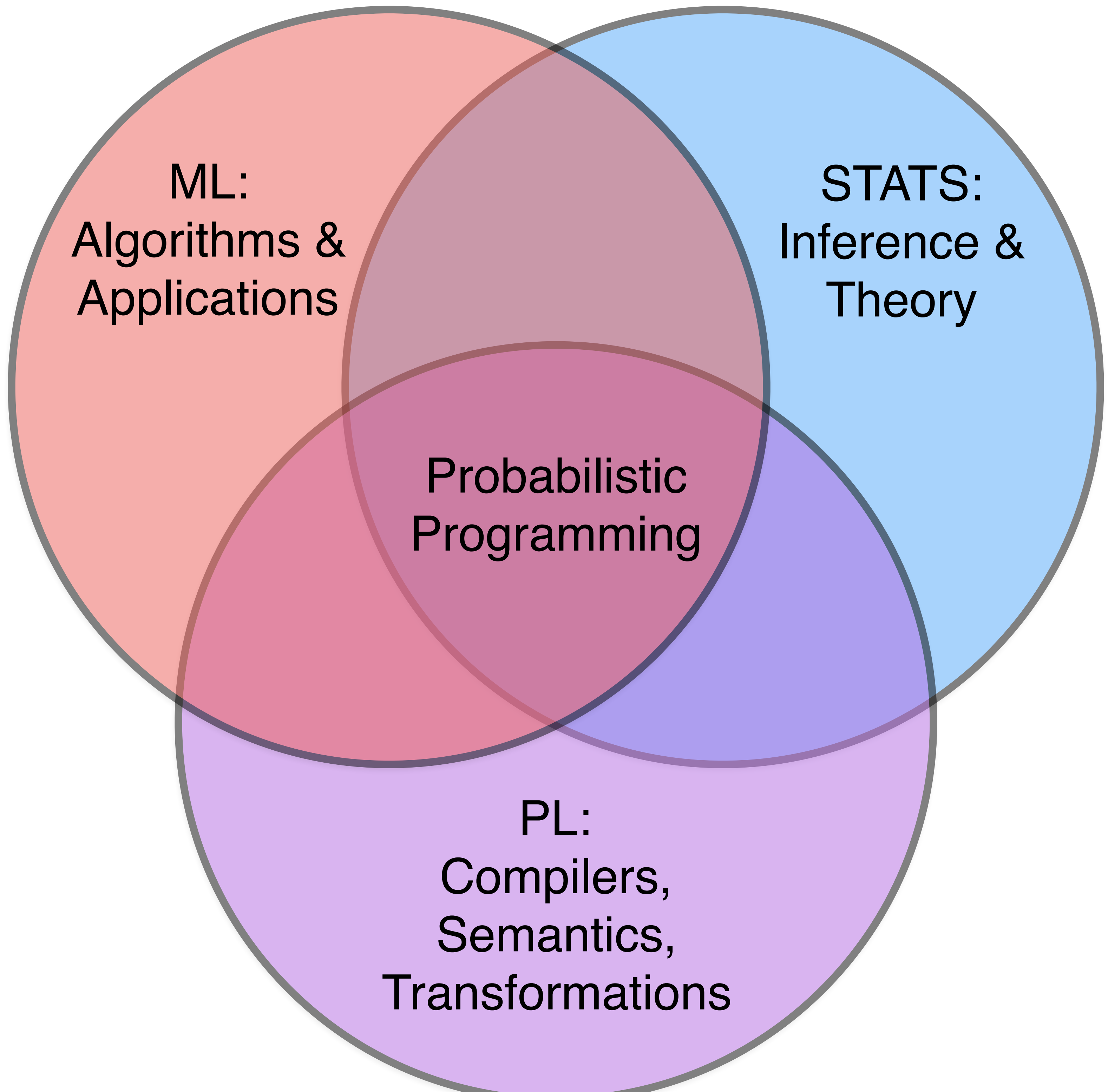
Difficulty: the simulator isn't differentiable, but there's a **trick**!

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

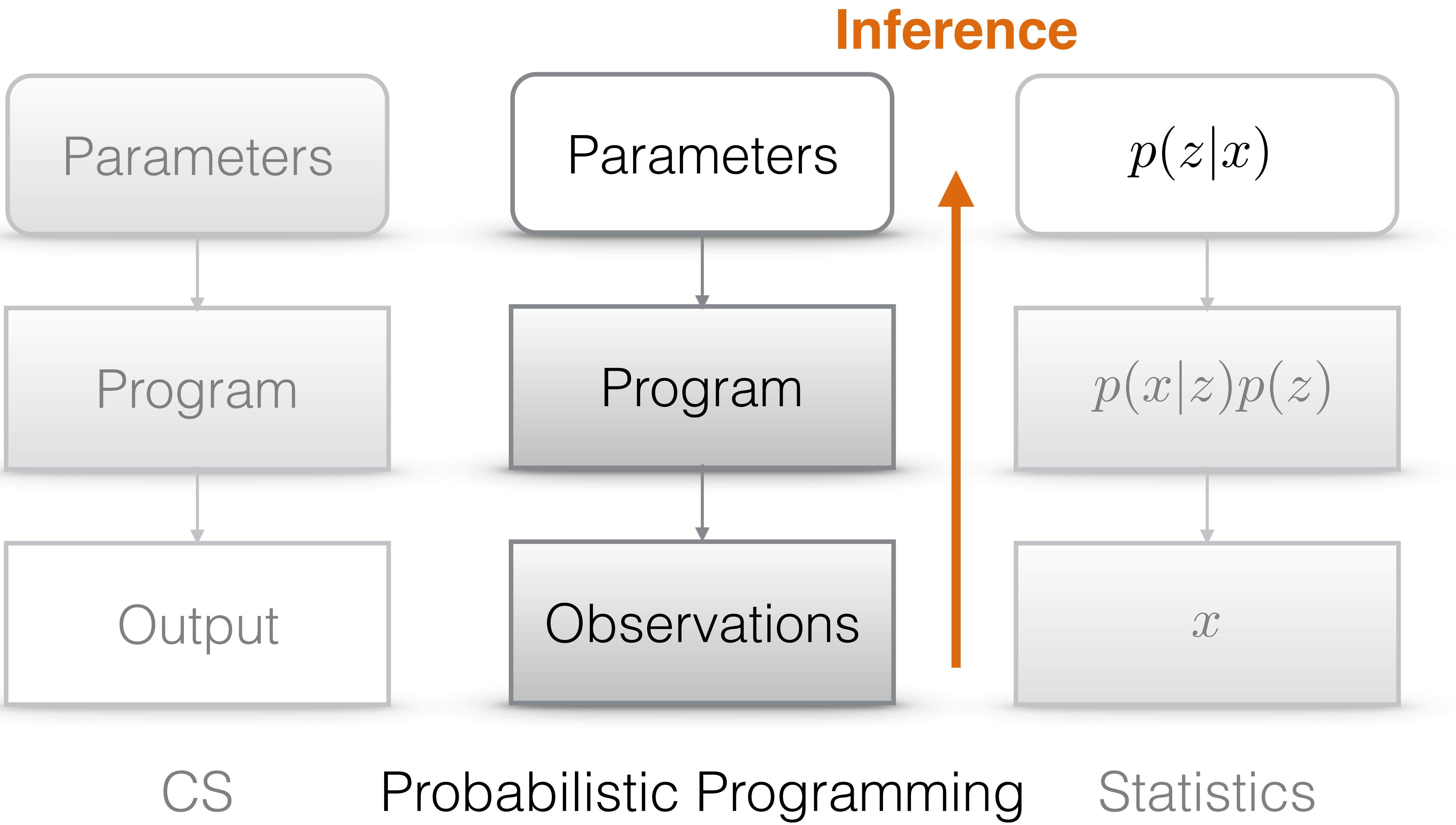
Probabilistic Programming:
Inverting the simulation

(very ambitious)

Probabilistic Programming



Intuition



[slides, Frank Wood]

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))}))
```

$\mathbf{x}$	$\mathbf{y}$
text	image

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

CAPTCHA breaking

Observation

Posterior Samples

Generative Model

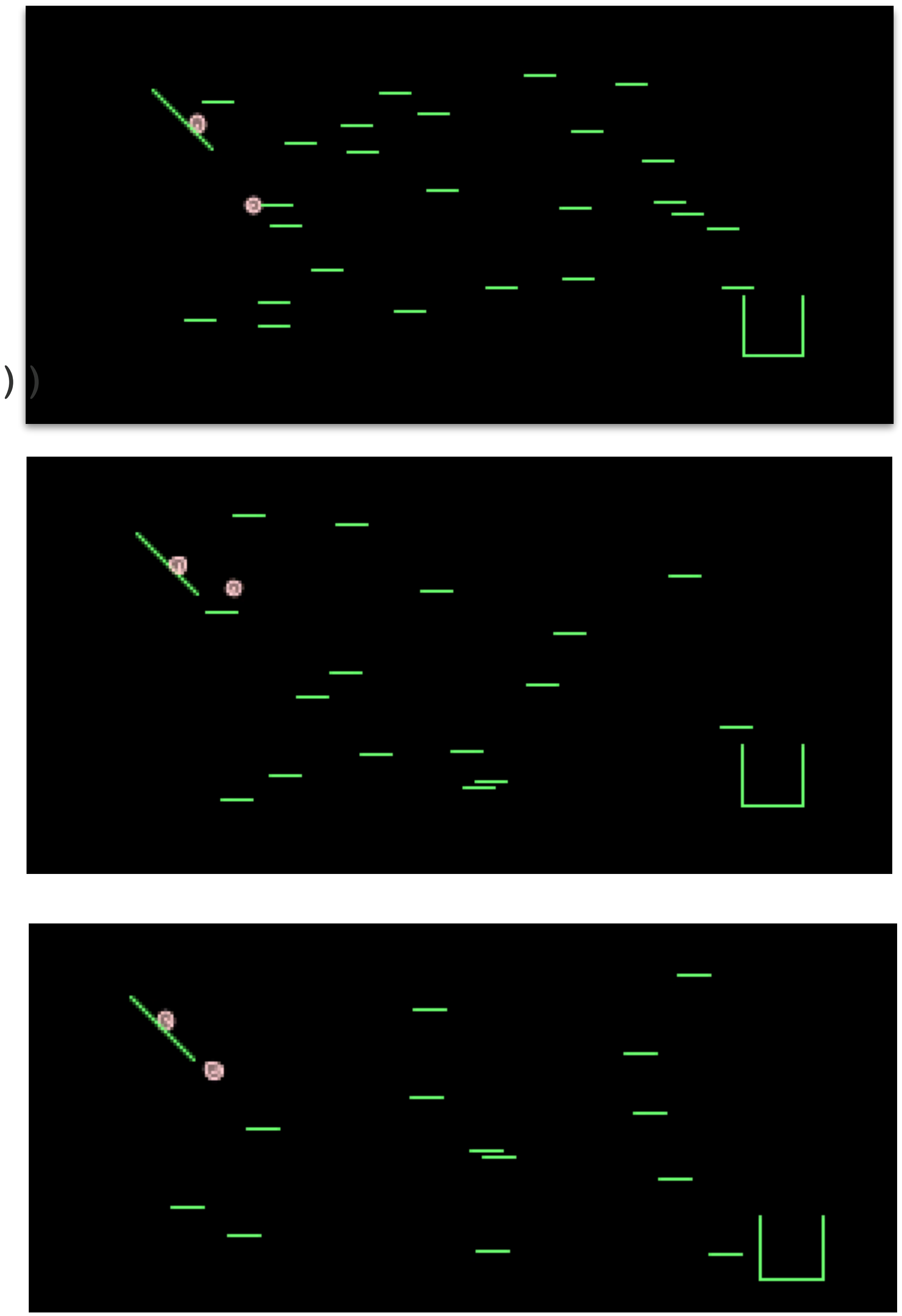
```
(defquery captcha
 [image num-chars tol]
 (let [[w h] (size image)
       ;; sample random characters
       num-chars (sample
                   (poisson num-chars))
       chars (repeatedly
               num-chars sample-char)]
  ;; compare rendering to true image
  (map (fn [y z]
         (observe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

$\mathbf{x}$	$\mathbf{y}$
text	image

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

ANALOGY: RANDOM BUMPERS ~ RANDOM CALORIMETER SHOWER

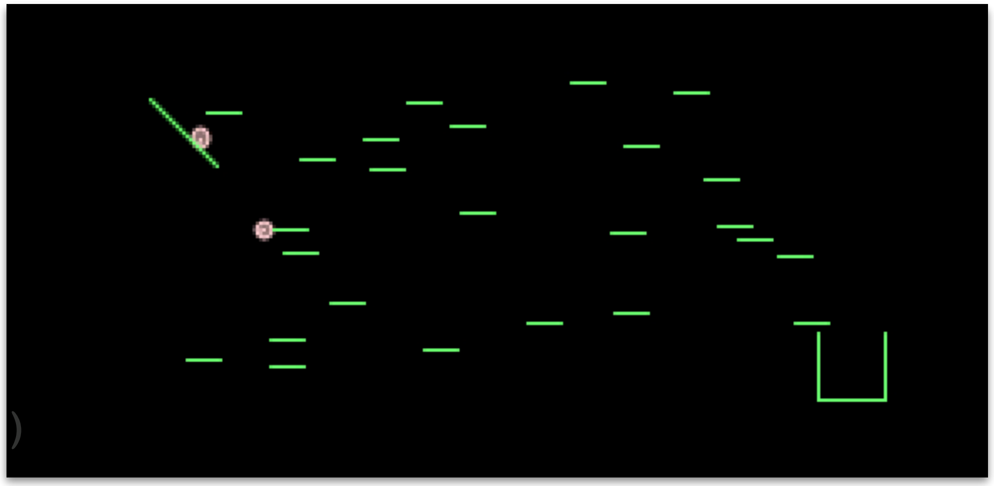
```
(defquery arrange-bumpers []
    (let [number-of-bumpers (sample (poisson 20))
          bumpydist (uniform-continuous 0 10)
          bumpxdist (uniform-continuous -5 14)
          bumper-positions (repeatedly
                             number-of-bumpers
                             #(vector (sample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

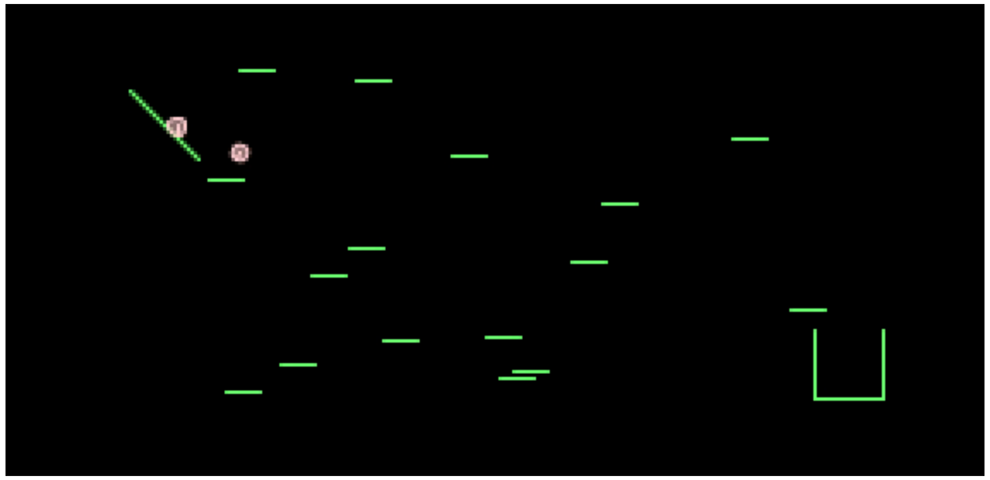


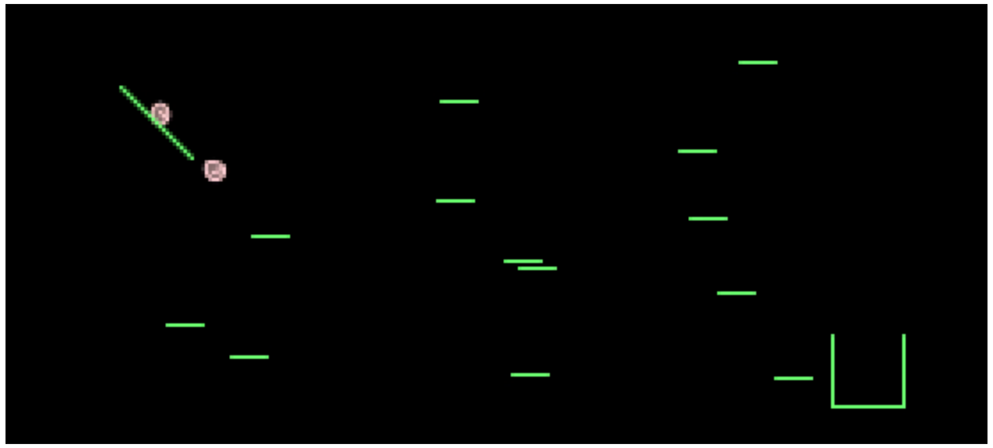
3 examples generated from simulator

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}))
```





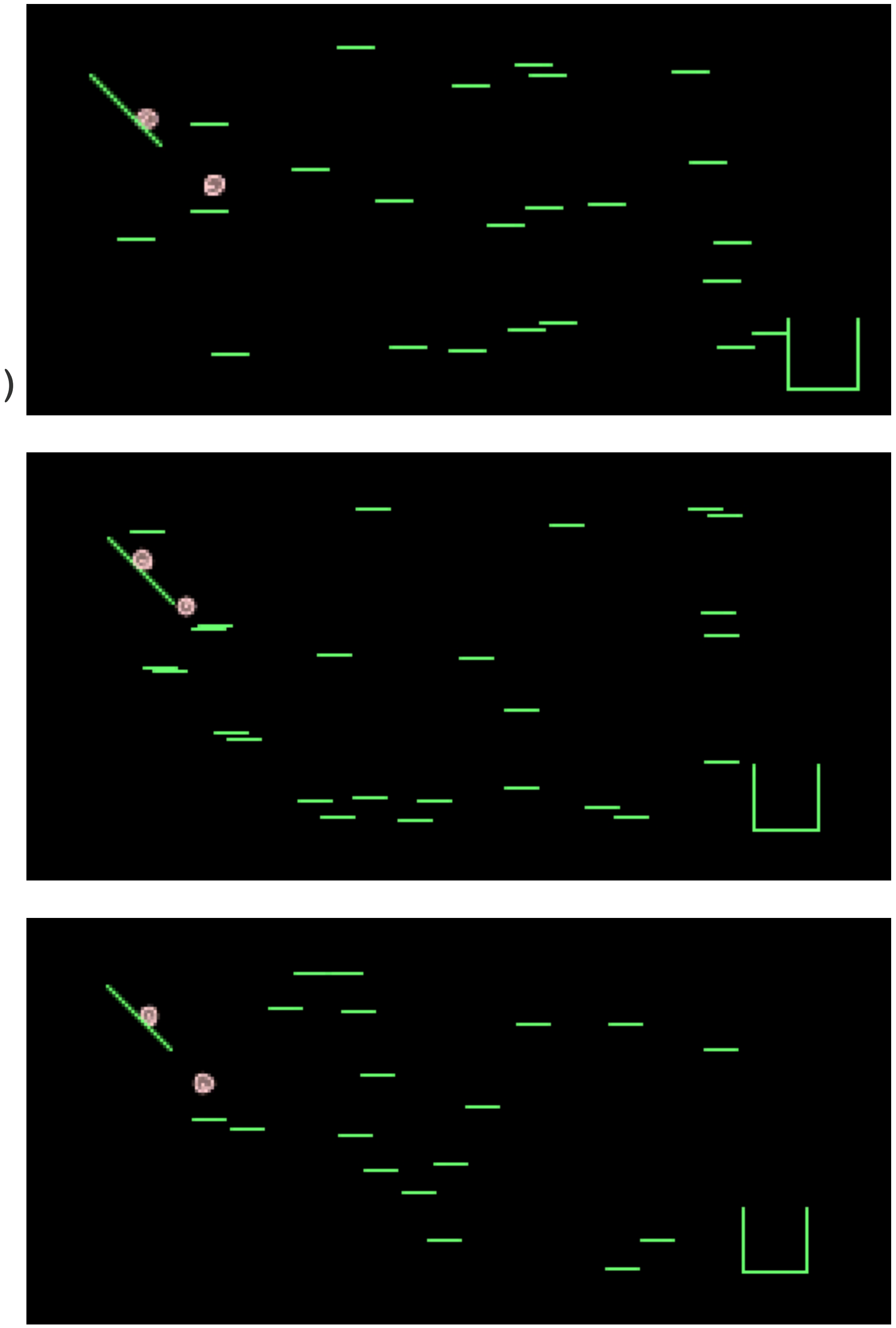


3 examples generated from simulator

[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)
```

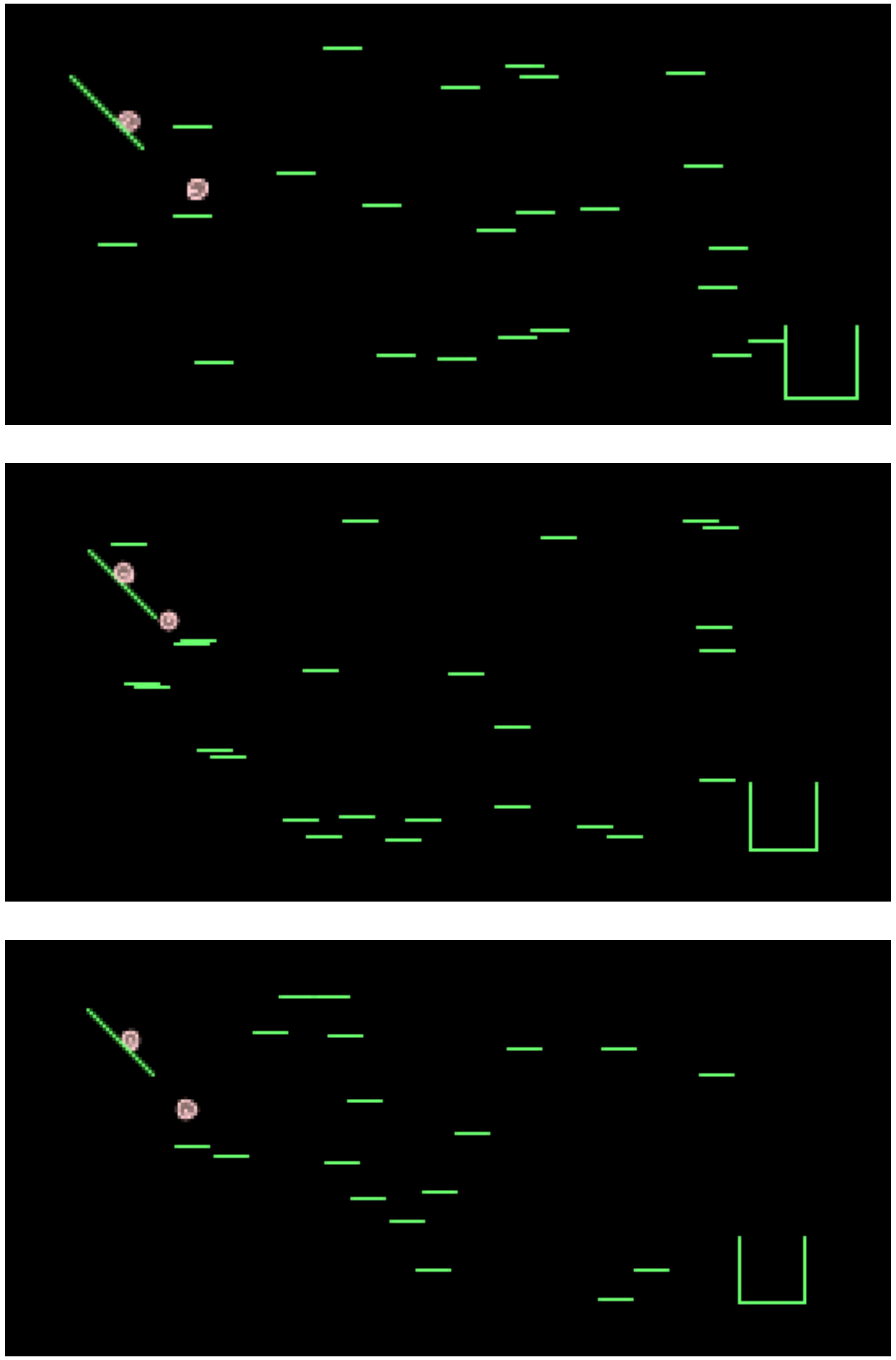


3 examples generated from simulator **conditioned** on ~20% of balls land in box (~ given observed energy deposits)

[slides, Frank Wood]

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)
```



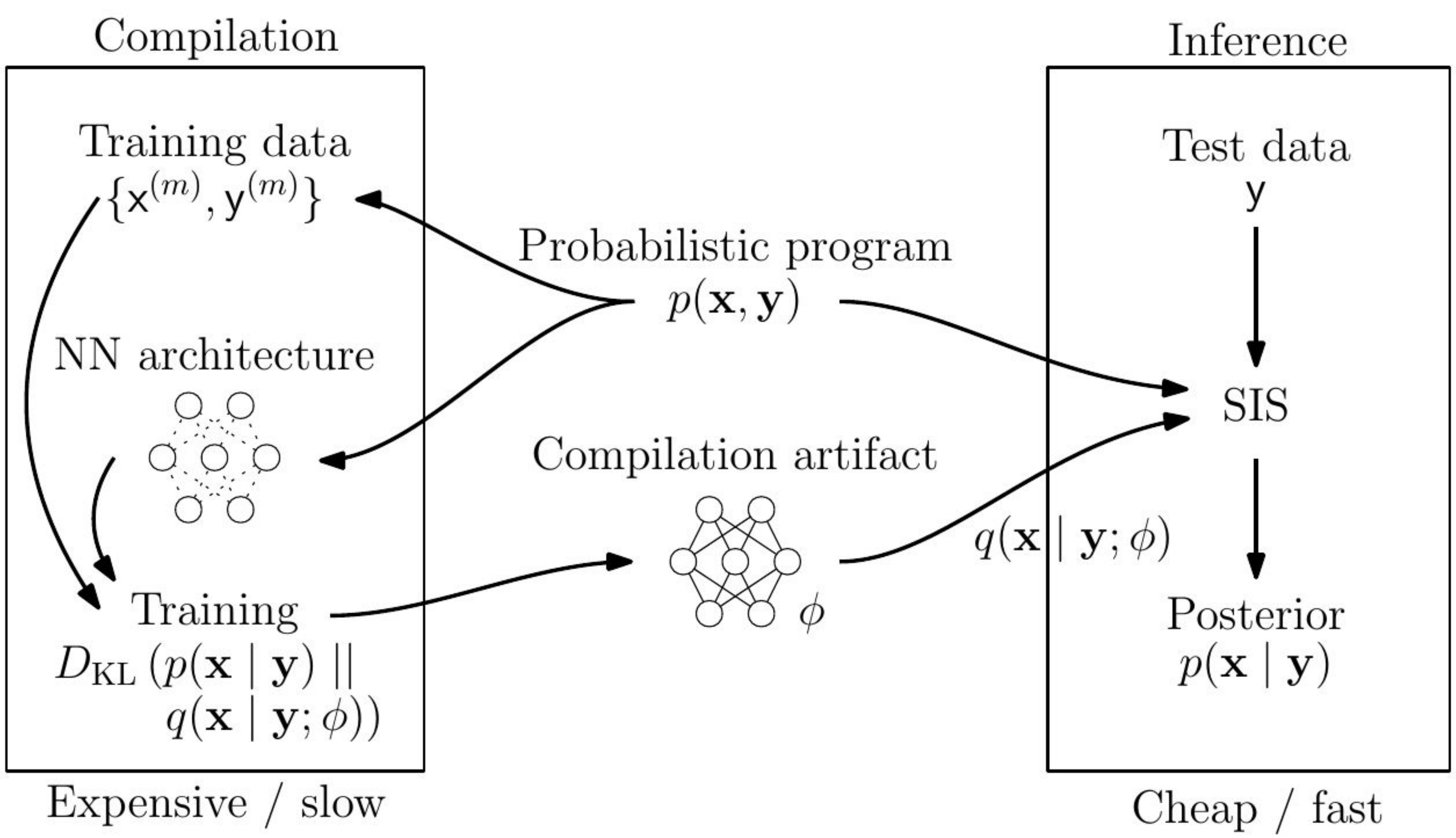
3 examples generated from simulator **conditioned** on ~20% of balls land in box (~ given observed energy deposits)

[slides, Frank Wood]

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)

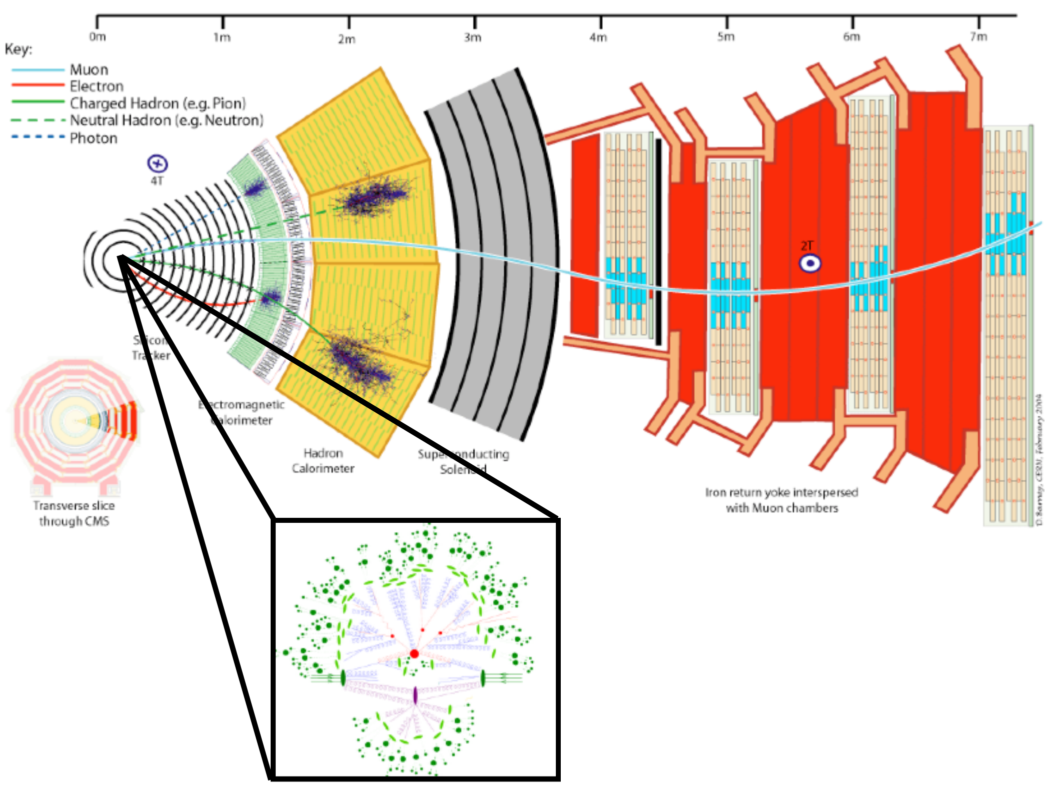
Output: a trained inference network, or "compilation artifact" (Torch, PyTorch)



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

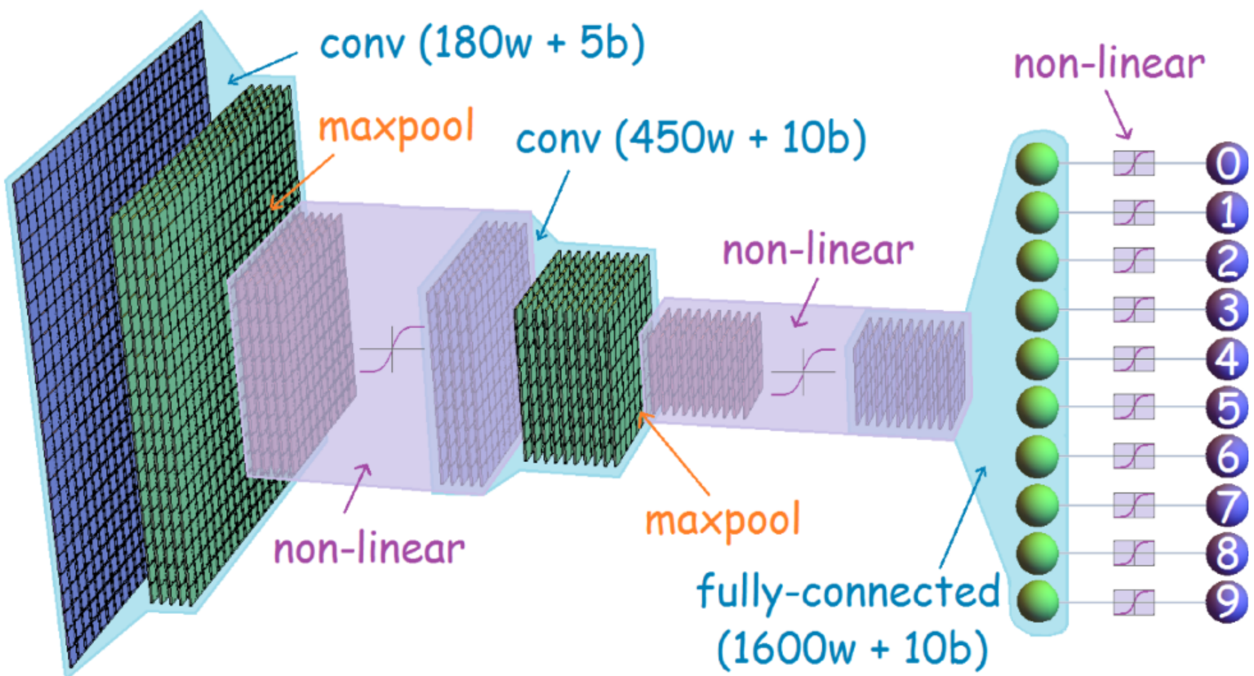
TWO APPROACHES

Use simulator
(much more efficiently)



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

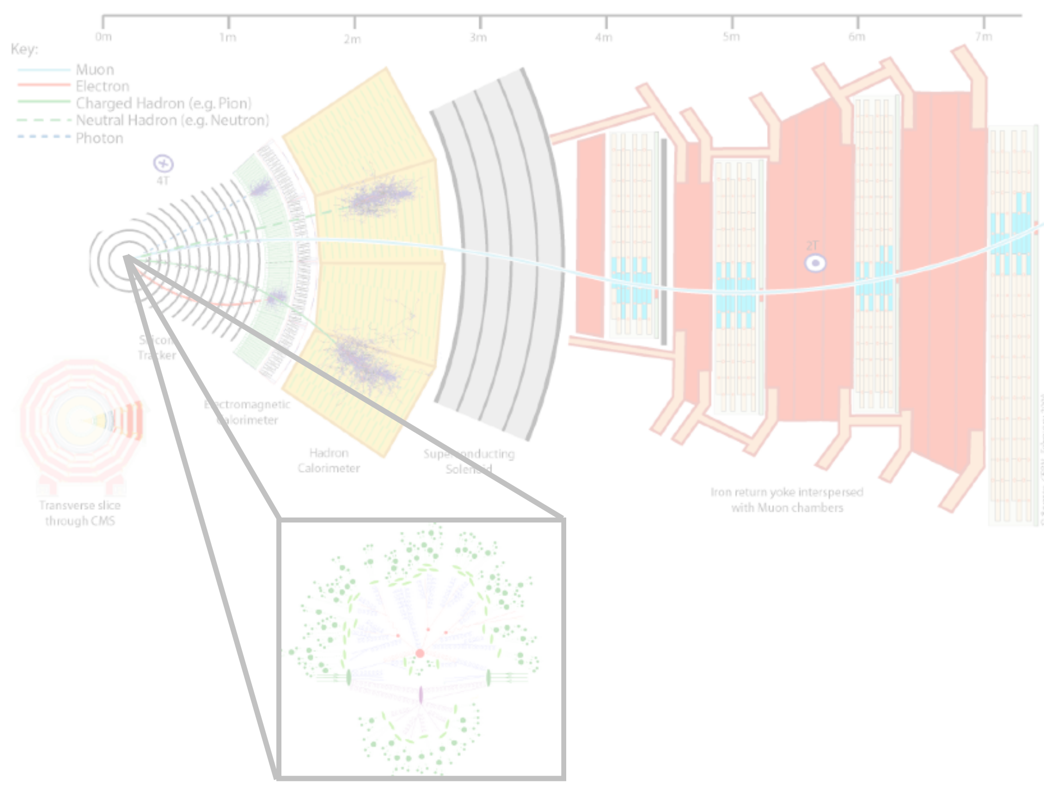
Learn simulator
(with deep learning)



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

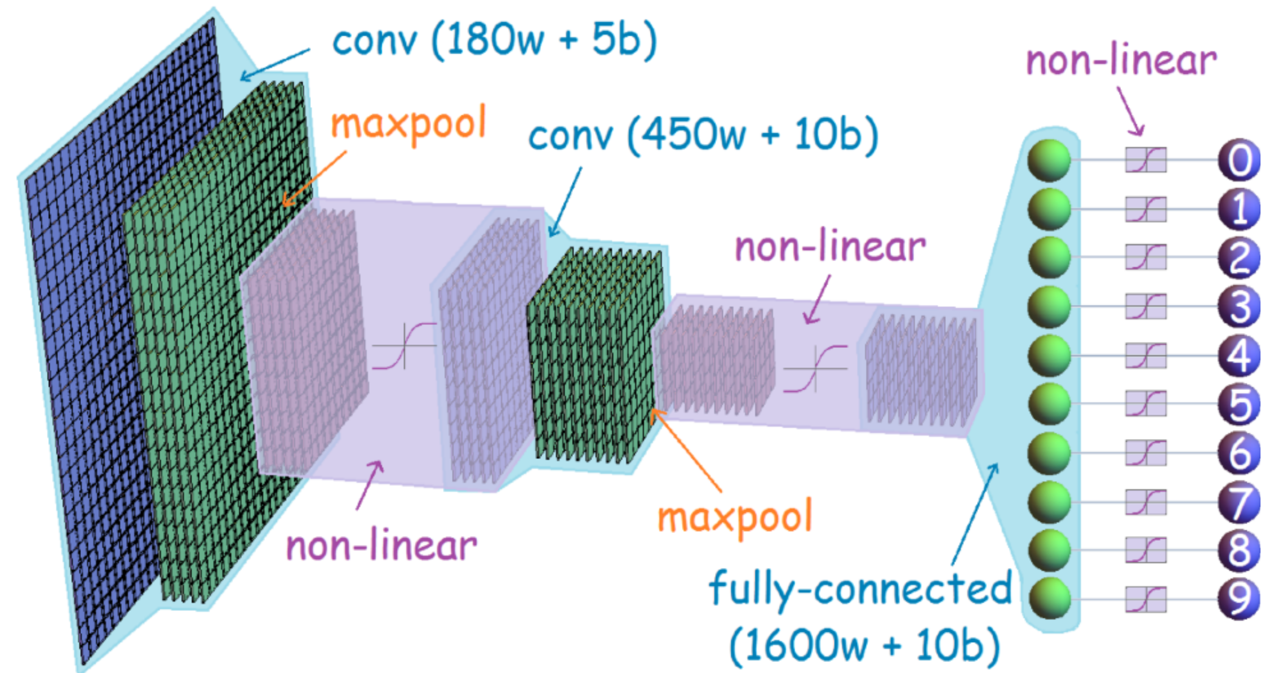
TWO APPROACHES

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

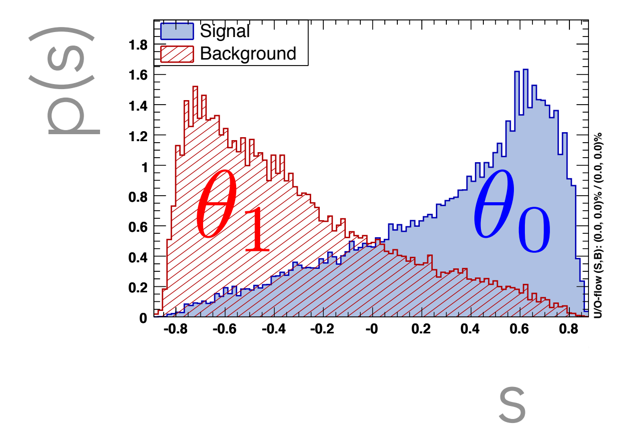
[image credit: A.P. Goucher]

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

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

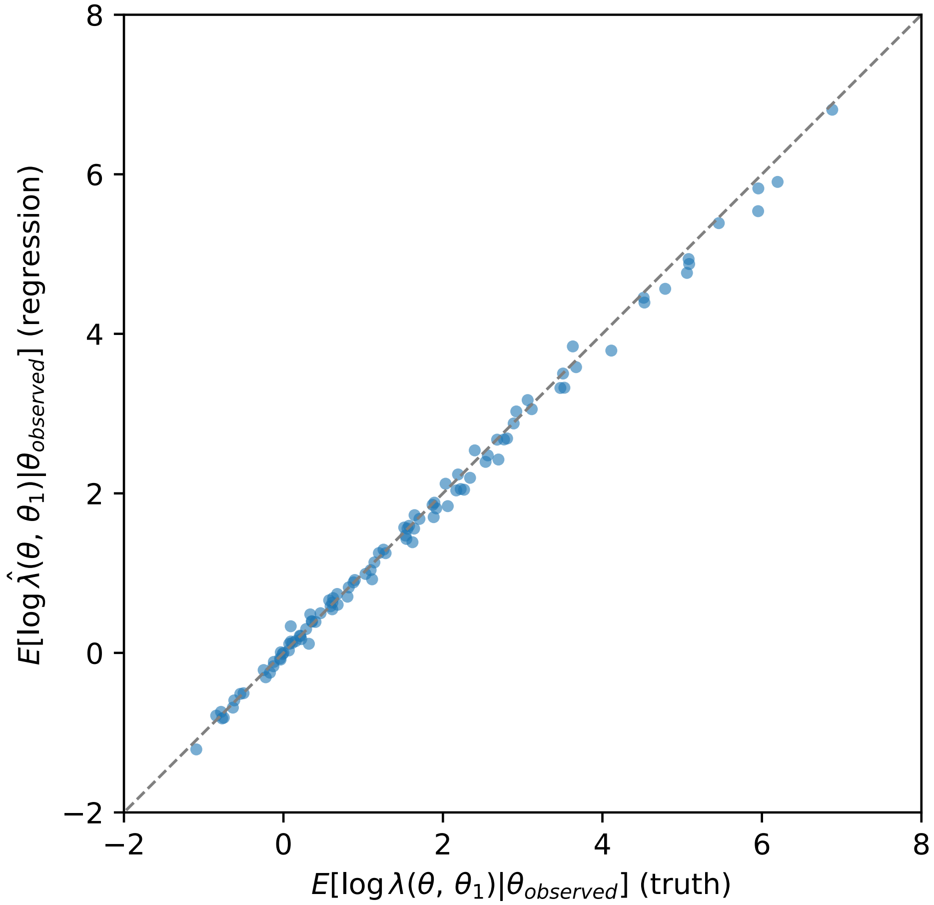
Can do the same thing for any two points θ_0 & θ_1 in parameter space. I call this a **parametrized classifier**

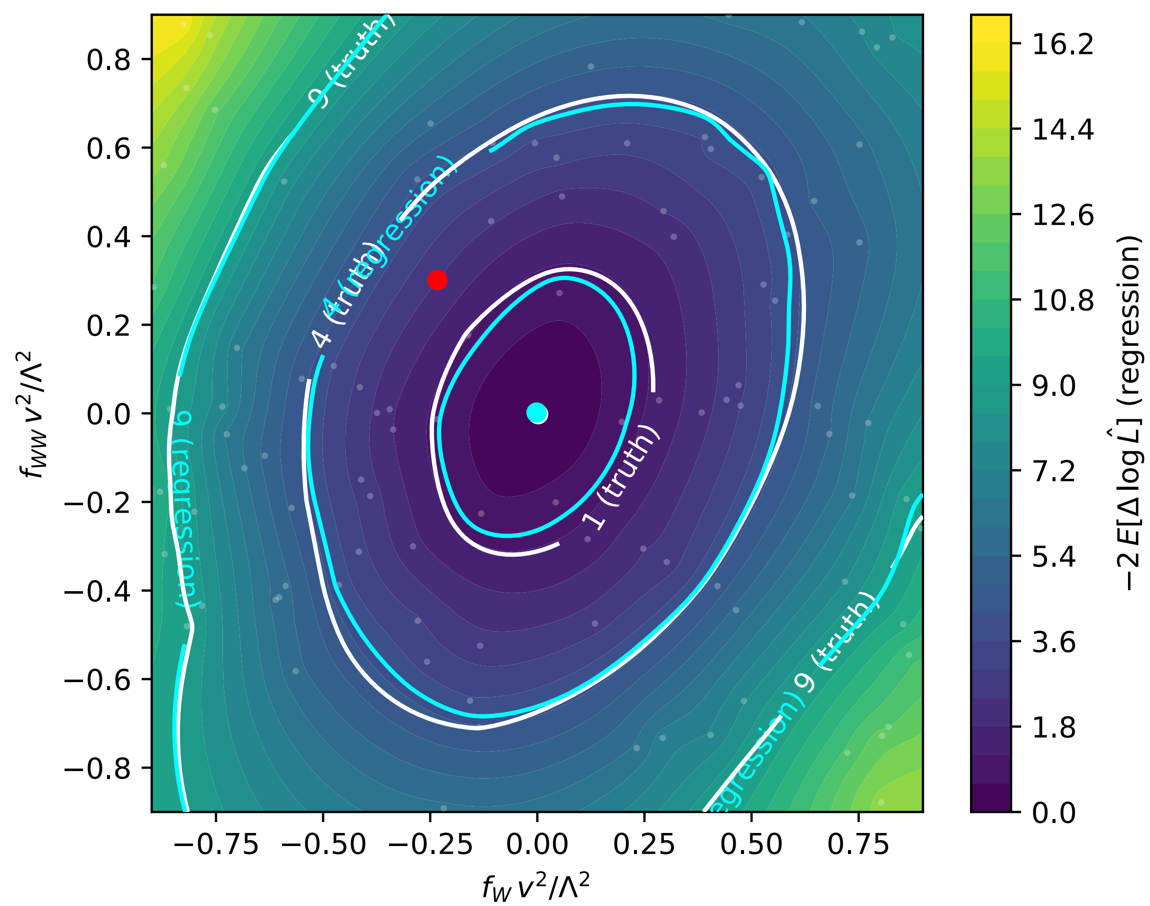
$$s(x;\theta_0,\theta_1) = \frac{p(x|\theta_1)}{p(x|\theta_0) + p(x|\theta_1)}$$

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

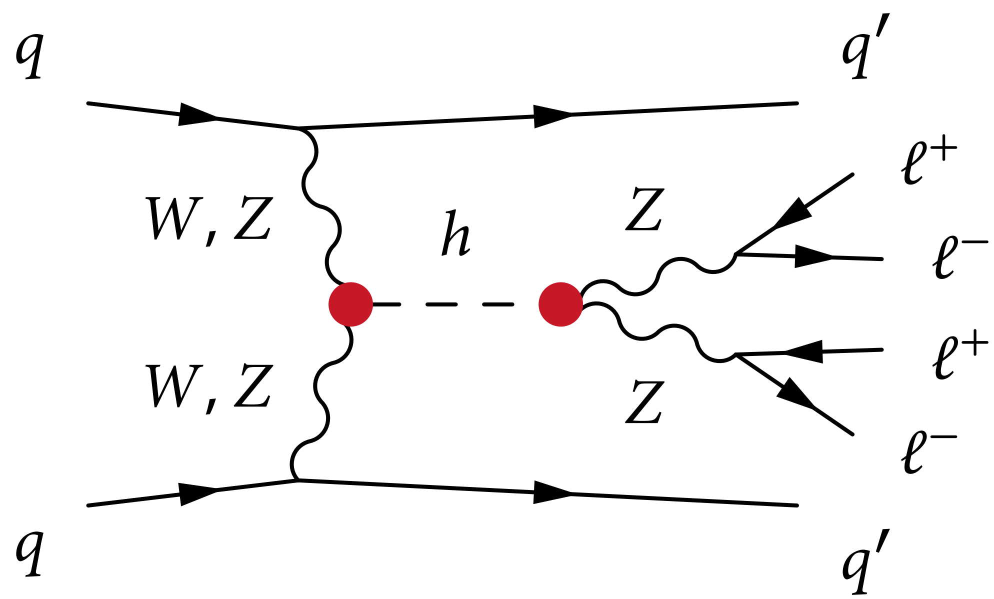
LEARNING A 16 DIM LIKELIHOOD

Estimated likelihood

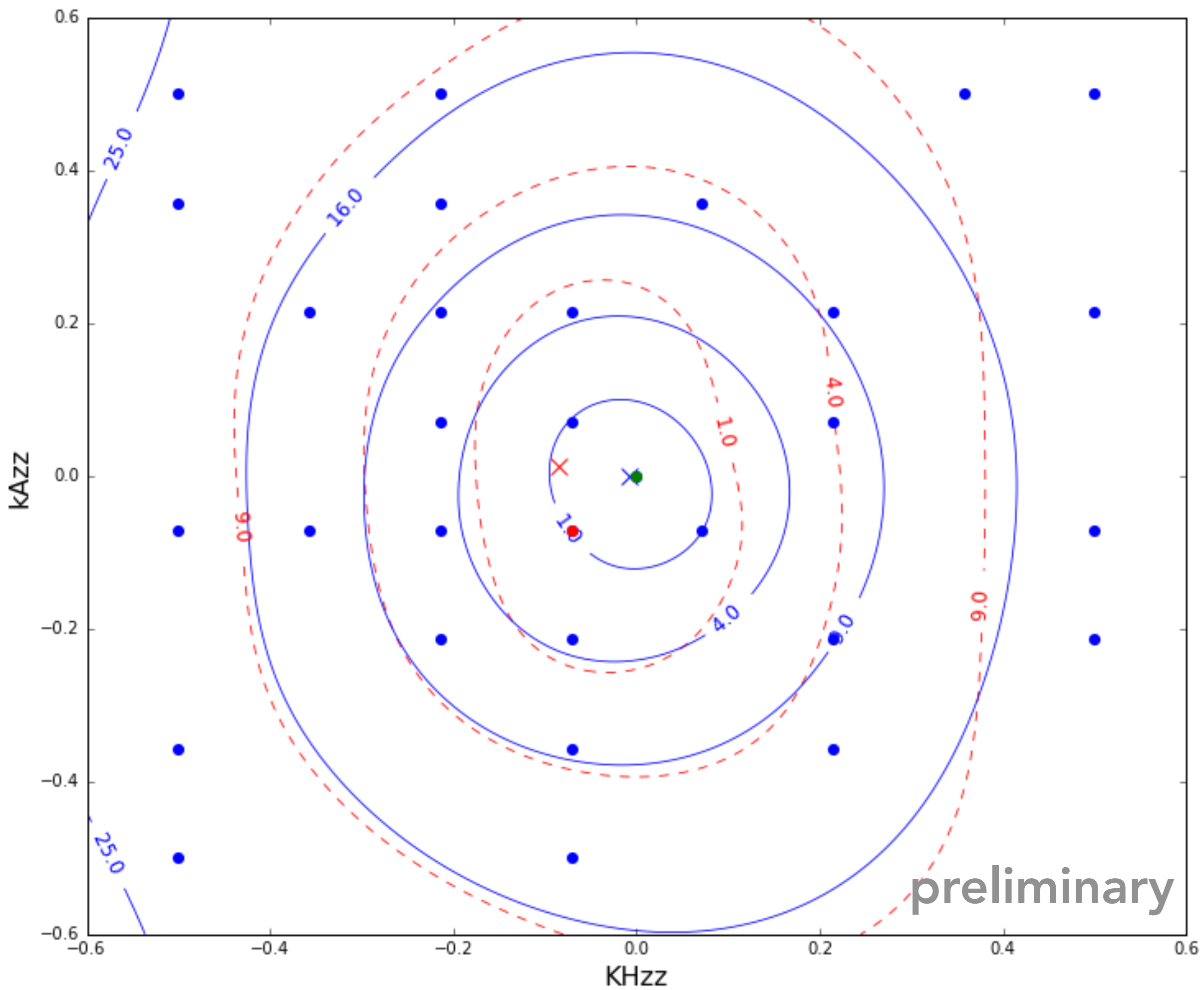


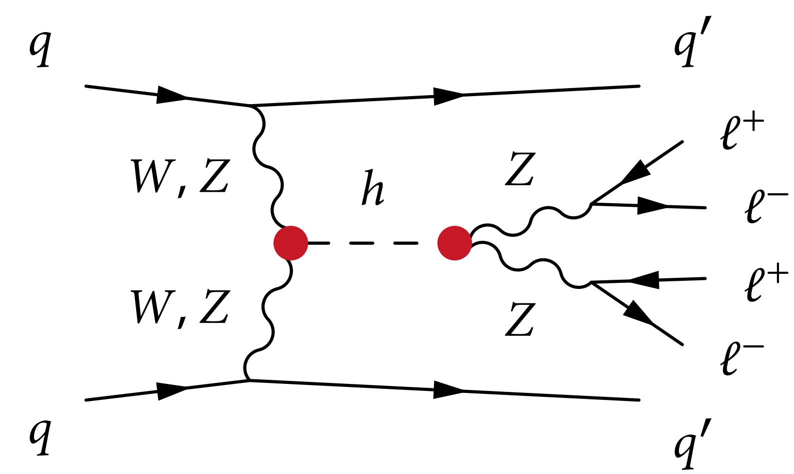


True likelihood



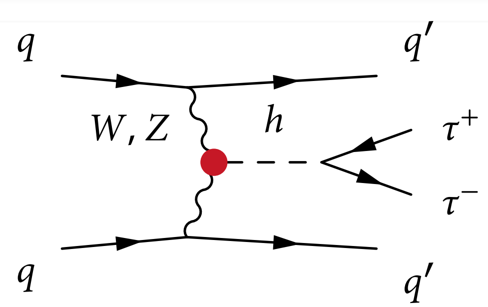
Preliminary work using fast detector simulation and CARL to approximate likelihoods using full kinematic information parametrized in 5-d coefficients of a Quantum Field Theory





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]

Now we can go beyond classification, and estimate parameters of theory and confidence intervals

Denote the maximum likelihood estimator

$$\hat{\theta} = \arg\max_{\theta} p(D|\theta) \tag{4.2}$$

The denominator in the likelihood ratio is just a constant

$$\hat{\theta} = \arg\max_{\theta} \sum \ln \frac{p(x_e|\theta)}{p(x_e|\theta_1)} = \arg\max_{\theta} \sum \ln \frac{p(s(x_e;\theta,\theta_1)|\theta)}{p(s(x_e;\theta,\theta_1)|\theta_1)} \,. \tag{4.4}$$

It is important that we include the denominator $p(s(x_e;\theta,\theta_1)|\theta_1)$ because this cancels Jacobian factors that vary with θ.

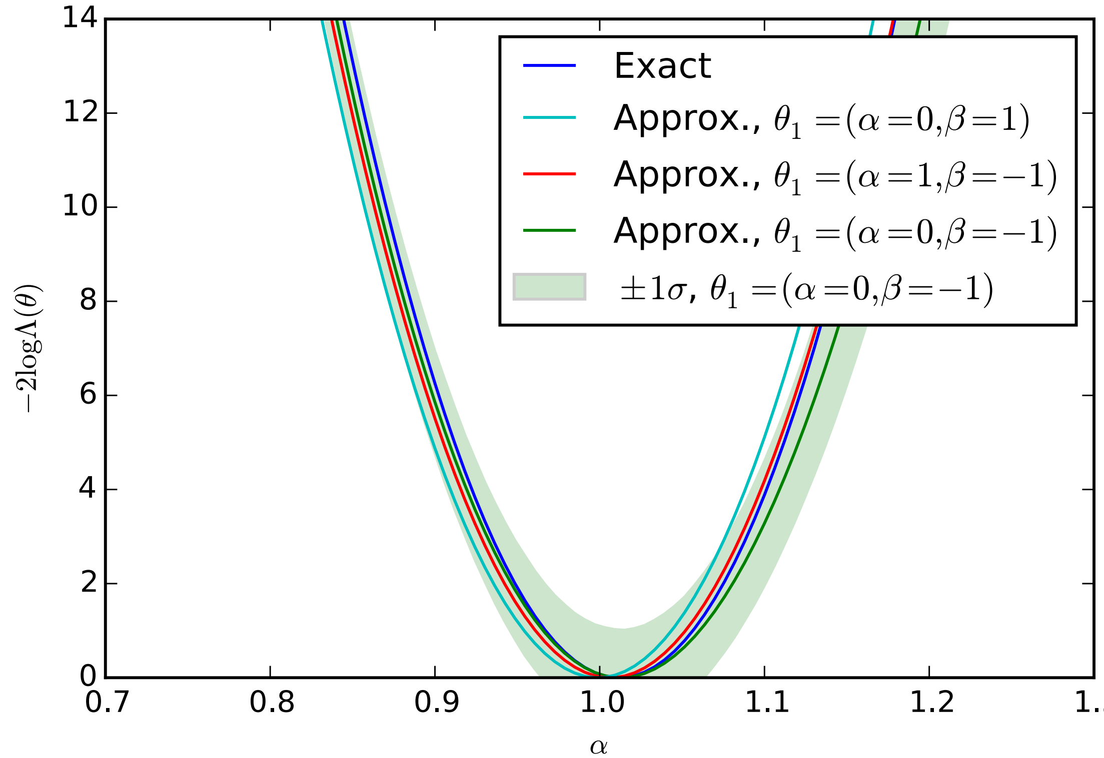
Provides a non-trivial diagnostic:

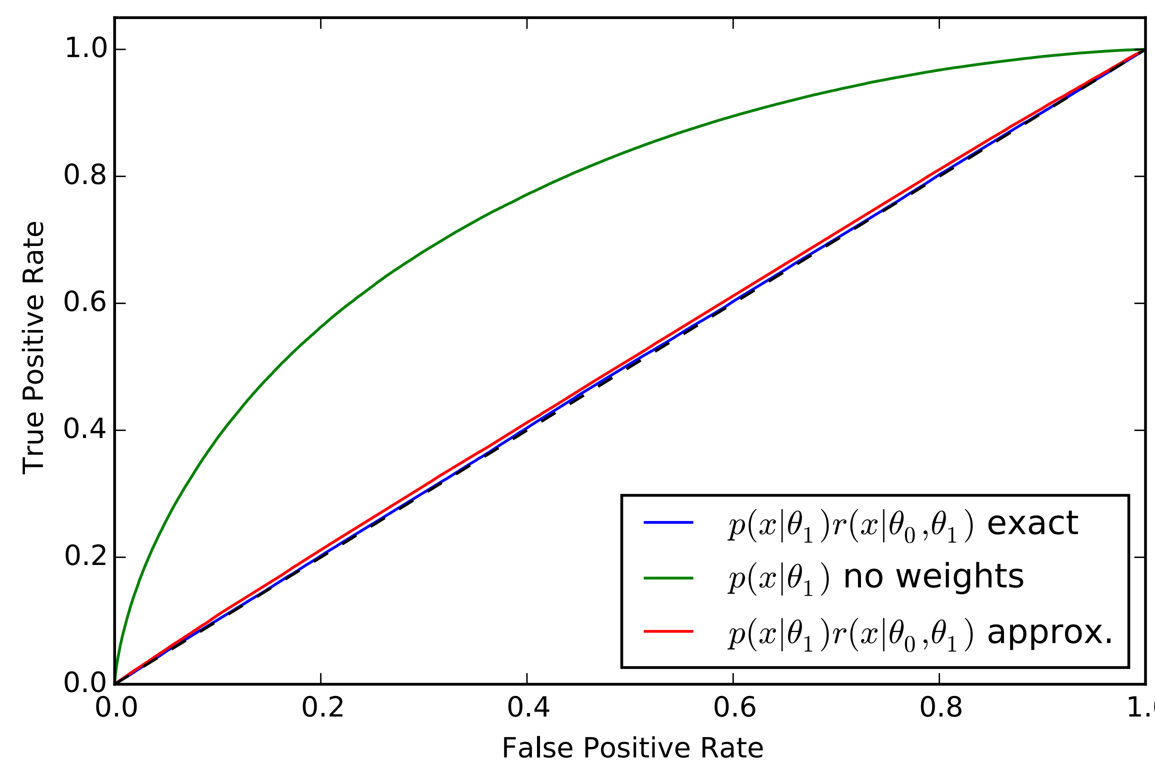
$$\frac{p_1(s^*)}{p_0(s^*)} = \frac{p_1(x)}{p_0(x)} \boxed{\frac{\int d\Omega_{s^*} p_0(x)/|\hat{n}\cdot\nabla s|}{\int d\Omega_{s^*} p_0(x)/|\hat{n}\cdot\nabla s|}} = \frac{p_1(x)}{p_0(x)}$$

DIAGNOSTICS

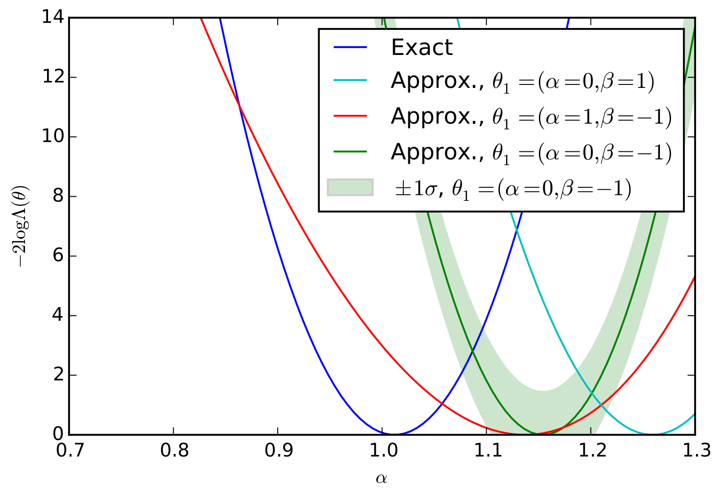
In practice $\hat{r}(\hat{s}(\mathbf{x}; \theta_0, \theta_1))$ will not be exact. Diagnostic procedures are needed to assess the quality of this approximation.

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 $\hat{r}(\hat{s}(\mathbf{x}; \theta_0, \theta_1))$.

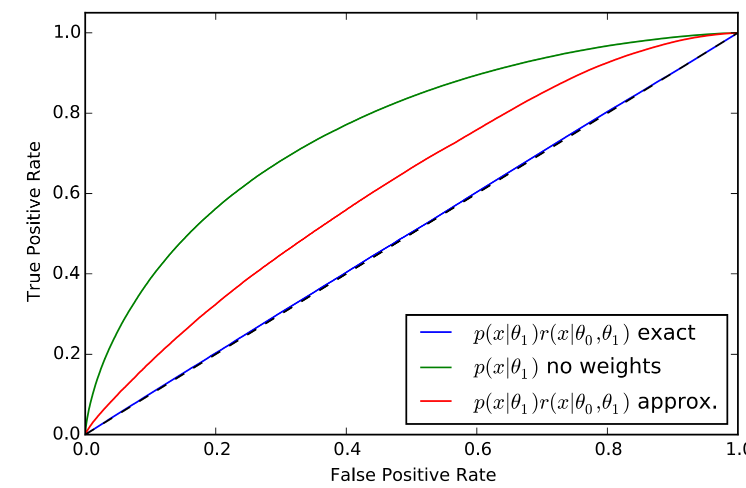




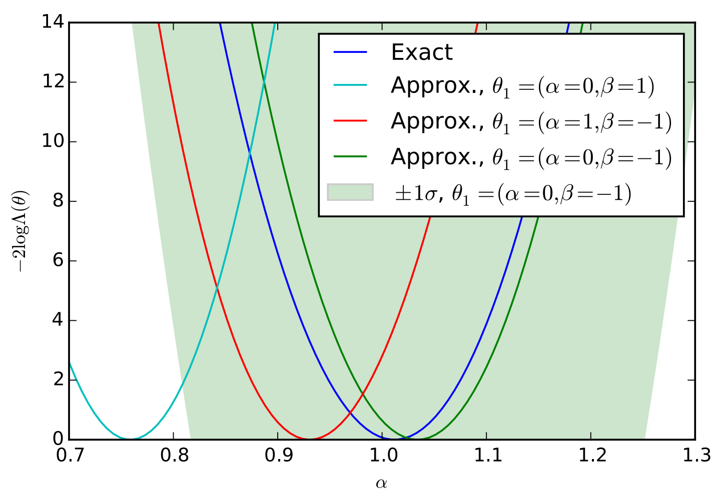
$$\frac{p_1(s^*)}{p_0(s^*)} = \frac{p_1(x)}{p_0(x)} \boxed{\frac{\int d\Omega_{s^*} p_0(x)/|\hat{n} \cdot \nabla s|}{\int d\Omega_{s^*} p_0(x)/|\hat{n} \cdot \nabla s|}} = \frac{p_1(x)}{p_0(x)} = r(x)$$



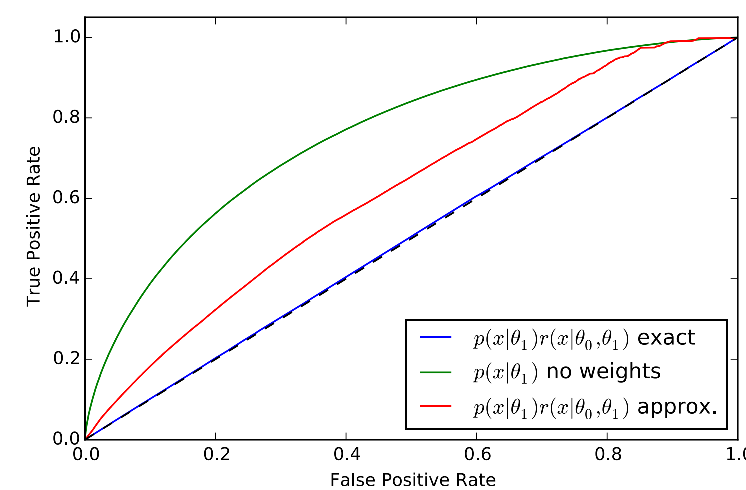
(a) Poorly trained, well calibrated.



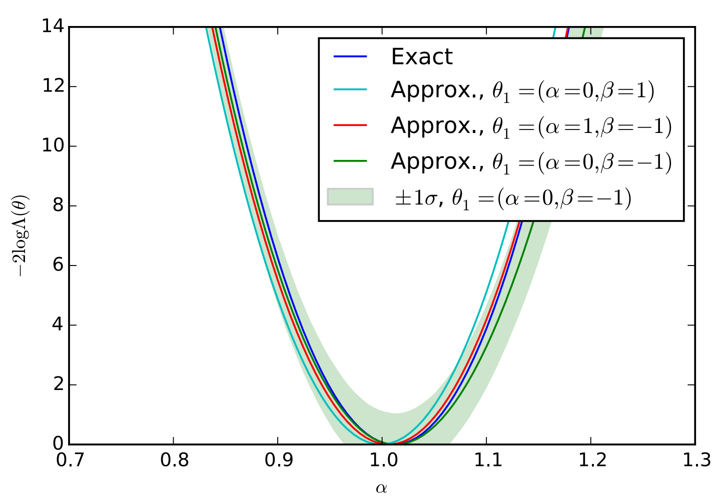
(b) Poorly trained, well calibrated.



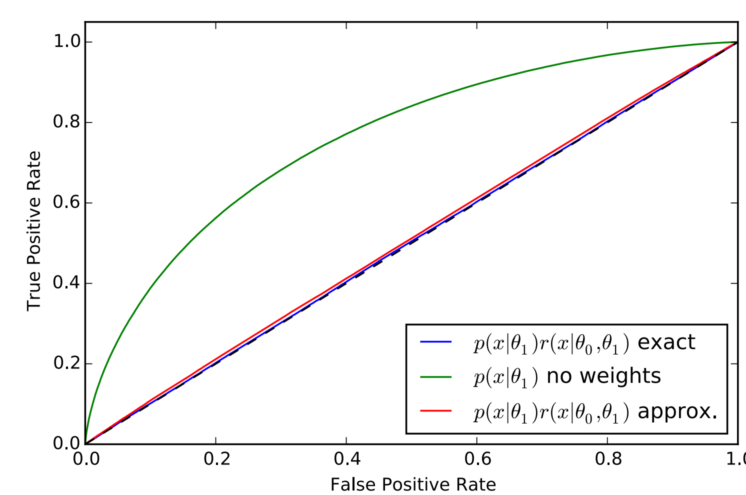
(c) Poorly calibrated, well trained.



(d) Poorly calibrated, well trained.



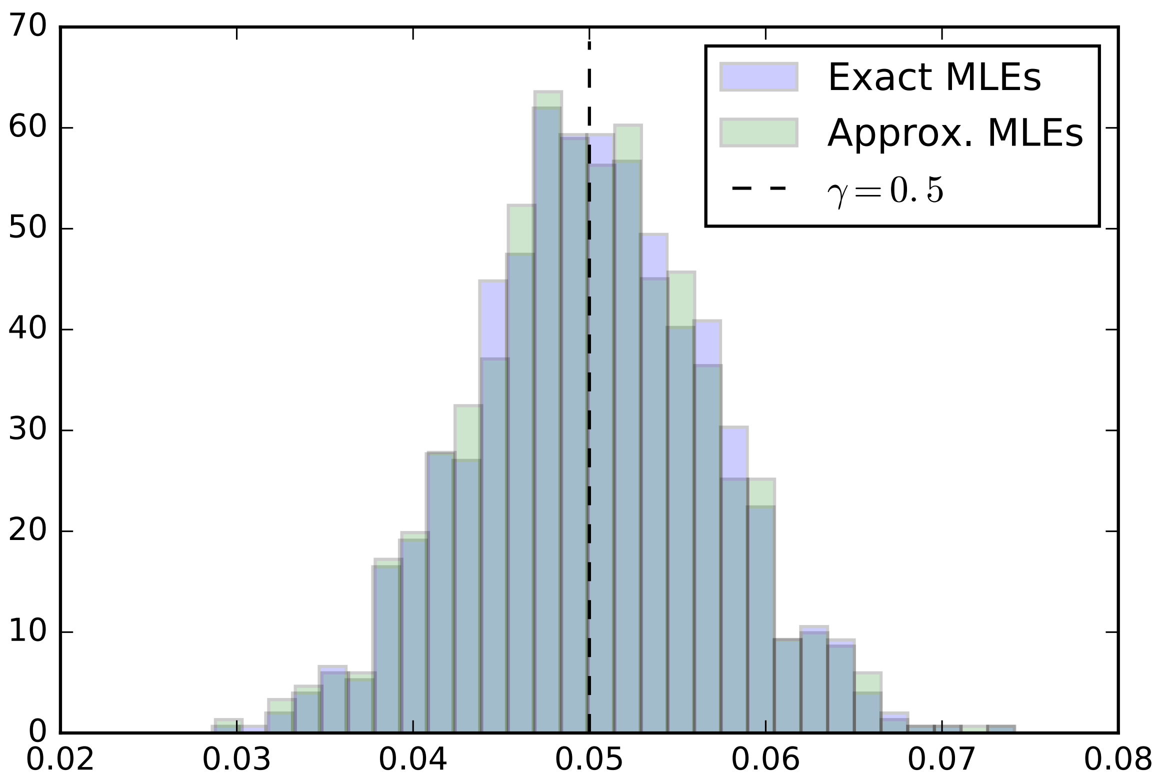
(e) Well trained, well calibrated.



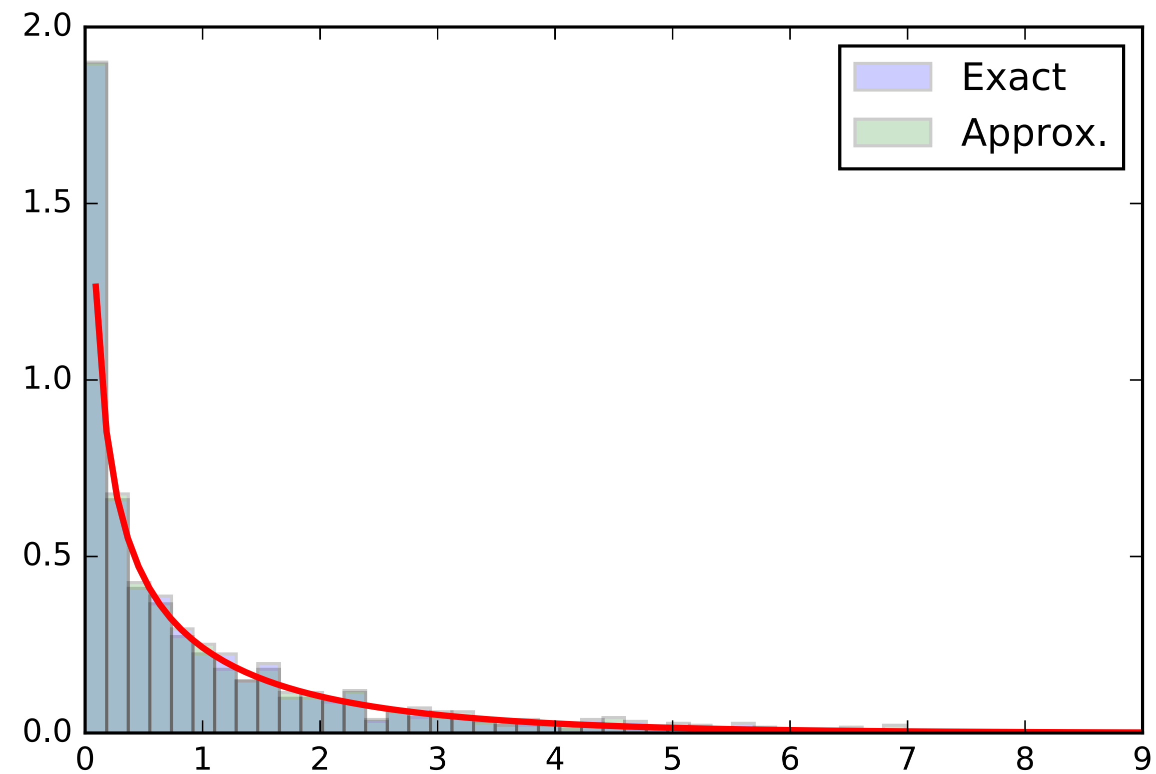
(f) Well trained, well calibrated.

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

WHAT IS THE OBJECTIVE?

ML: What is the problem you are trying to solve?

Physicist: [eventually describes problem and formalizes objective]

ML: Ok, well let's optimize this directly …

Physicist: but, I also want….

Used to criticize physicists for constantly changing problem statement, but traditional approach to physics problems has many advantages

- modular, reusable components (facilitates transfer learning, "ML2.0")
- interpretable & individually validated
- a form of structural regularization

STATISTICAL DECISION THEORY IN 1 SLIDE

Θ - States of nature; X - possible observations; A - action to be taken

$p(x|\theta)$ - statistical model; $\pi(\theta)$ - prior

$\delta: X \rightarrow A$ - **decision rule** (take some action based on observation)

$L: \Theta \times A \rightarrow \mathbb{R}$ - **loss function**, real-valued function true parameter and action

$R(\theta,\delta) = E_{p(x|\theta)}[L(\theta, \delta)]$ - **risk**

- A decision δ^* rule **dominates** a decision rule δ if and only if $R(\theta,\delta^*) \leq R(\theta,\delta)$ for all θ, and the inequality is strict for some θ.

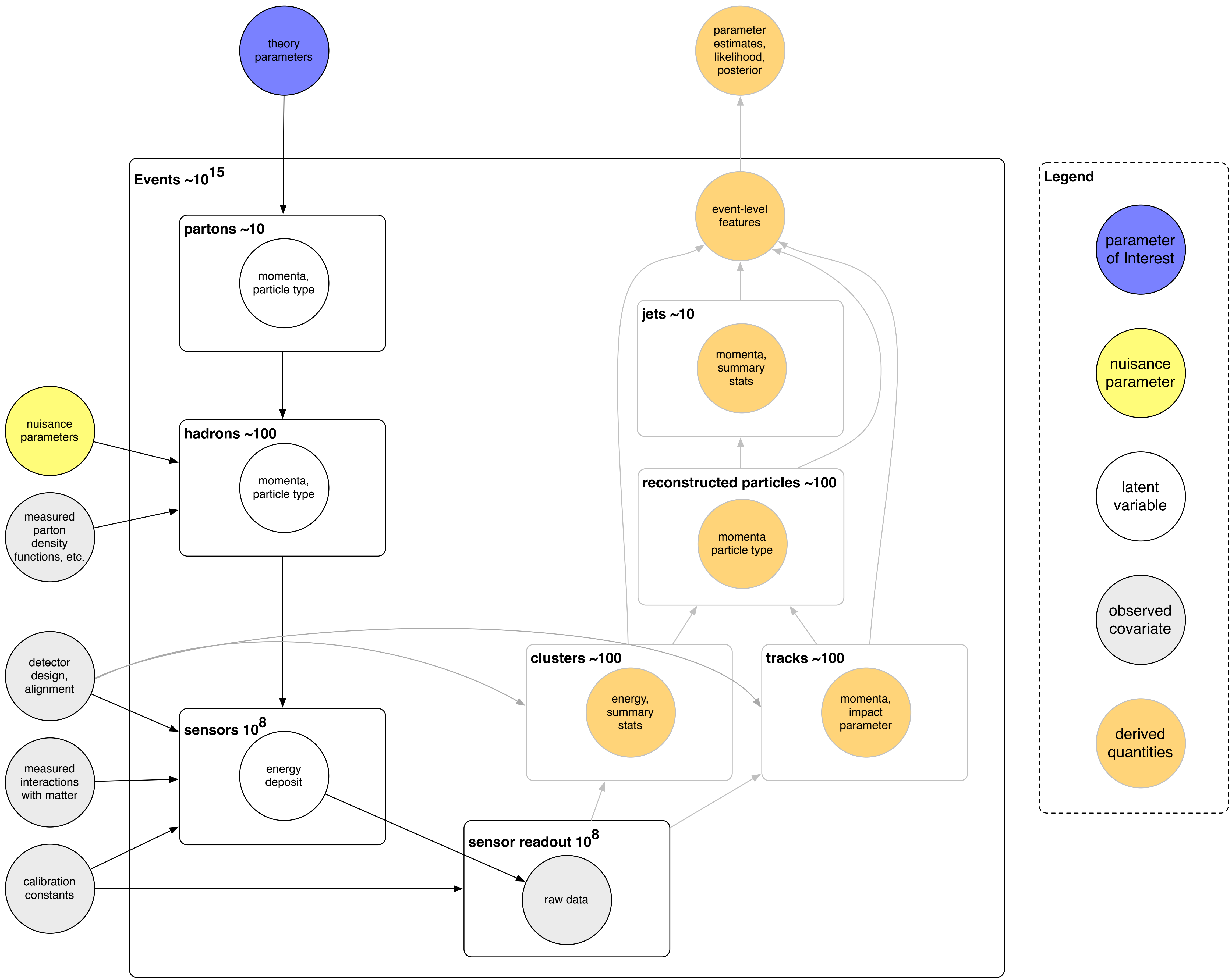
- A decision rule is **admissible** if and only if no other rule dominates it; otherwise it is inadmissible

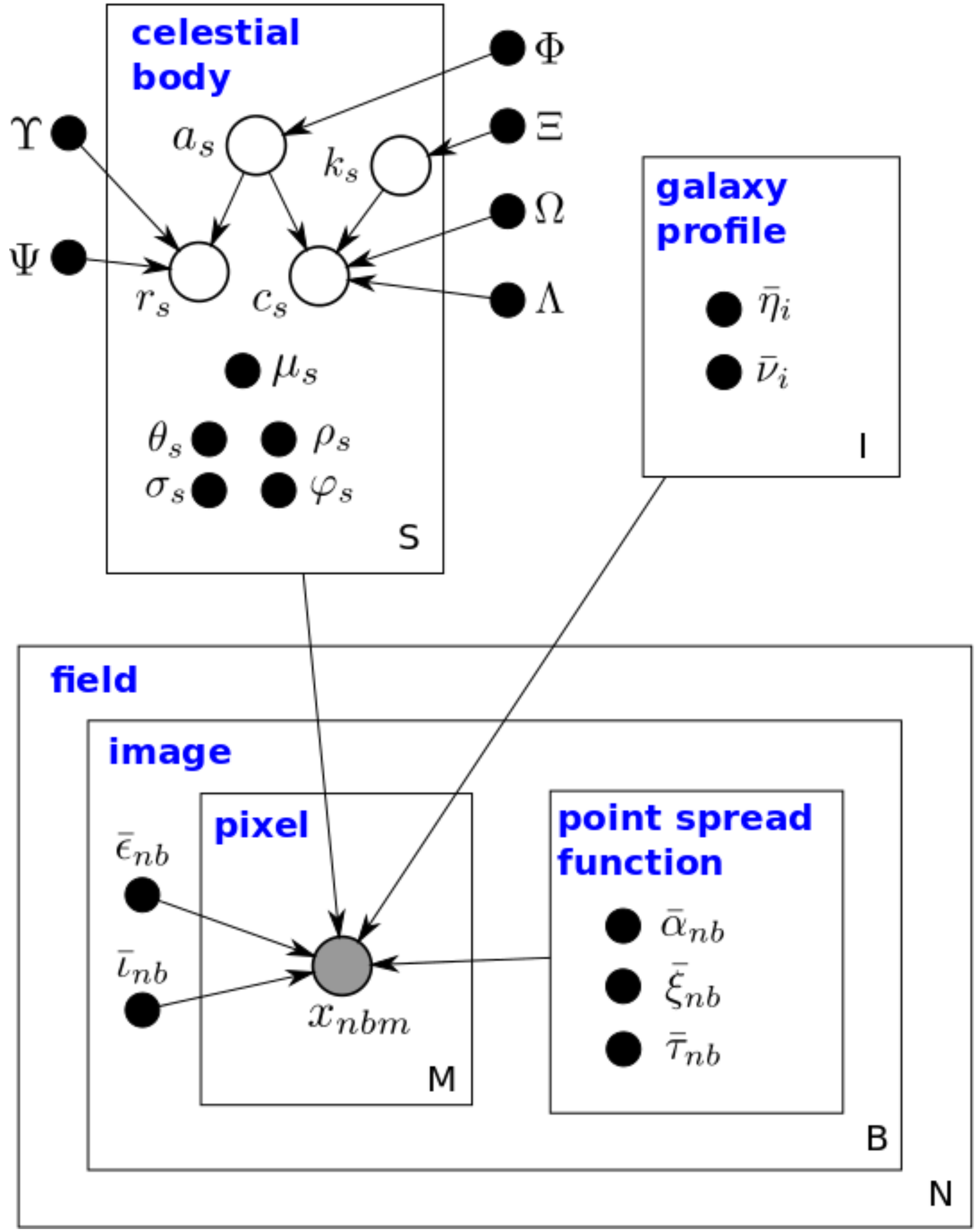
$r(\pi, \delta) = E_{\pi(\theta)}[R(\theta,\delta)]$ - **Bayes risk** (expectation over θ w.r.t. prior and possible observations)

$\rho(\pi, \delta \mid x) = E_{\pi(\theta|x)}[L(\theta,\delta(x))]$ - **expected loss** (expectation over θ w.r.t. posterior $\pi(\theta|x)$)

- δ' is a (generalized) Bayes rule if it minimizes the expected loss

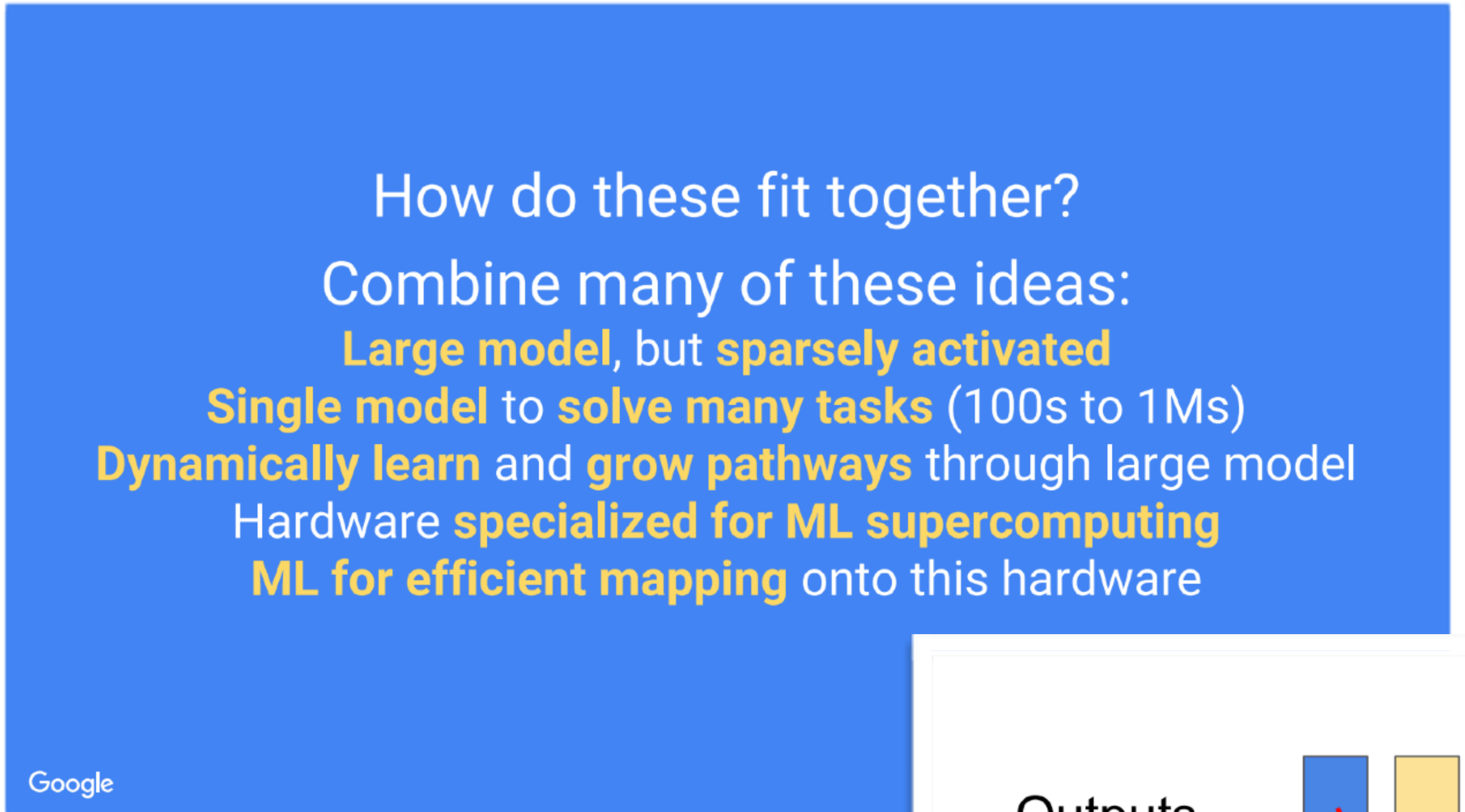
FULL SIMULATION + RECONSTRUCTION

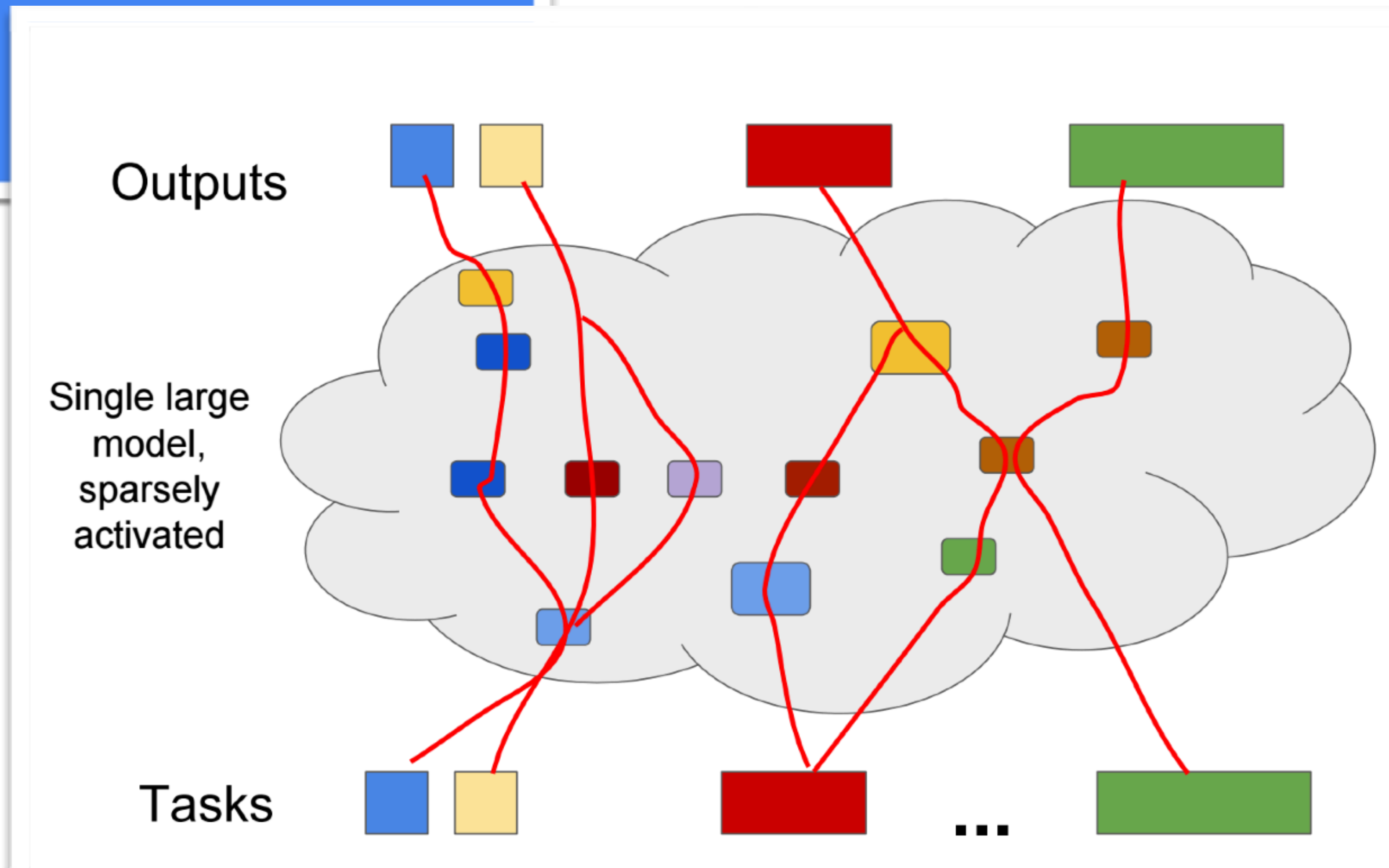




Celeste: Variational inference for a generative model of astronomical images

ML2.0?





Slides from Jeff Dean of Google Brain @ Jeju July 2017

https://drive.google.com/file/d/0B8z5oUpB2DysZWF1RTFuX1NEZUk/view