# Machine Learning for the LHC

Tilman Plehn

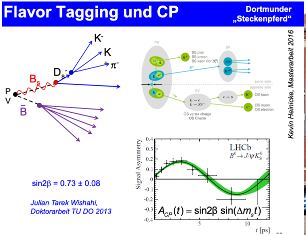
Universität Heidelberg

München 9/2022

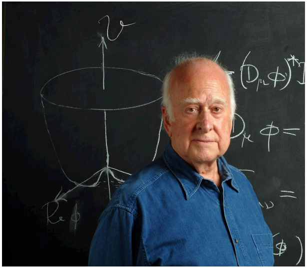
# Modern LHC physics

## Classic motivation

- dark matter
- baryogenesis
- Higgs VEV







# Modern LHC physics

## Classic motivation

- dark matter
- baryogenesis
- Higgs VEV

## LHC physics

- fundamental questions
- huge data set
- complete uncertainty control
- first-principle precision simulations

# Modern LHC physics

## Classic motivation

- dark matter
- baryogenesis
- Higgs VEV

## LHC physics

- fundamental questions
- huge data set
- complete uncertainty control
- first-principle precision simulations

## Traditional methods

- discover in rates
- unveil little black holes
- find supersymmetry
- travel extra dimensions
- measure couplings

# Modern LHC physics

## Classic motivation

- dark matter
- baryogenesis
- Higgs VEV

## LHC physics

- fundamental questions
- huge data set
- complete uncertainty control
- first-principle precision simulations

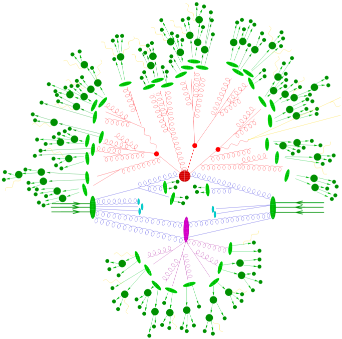
## Traditional methods

- discover in rates
- unveil little black holes
- find supersymmetry
- travel extra dimensions
- measure couplings

## First-principle simulations

- start with Lagrangian
- calculate scattering using QFT
- simulate events
- simulate detectors

$\rightarrow$ LHC events in virtual worlds



# Modern LHC physics

## Classic motivation

- dark matter
- baryogenesis
- Higgs VEV

## LHC physics

- fundamental questions
- huge data set
- complete uncertainty control
- first-principle precision simulations

## Traditional methods

- discover in rates
- unveil little black holes
- find supersymmetry
- travel extra dimensions
- measure couplings

## First-principle simulations

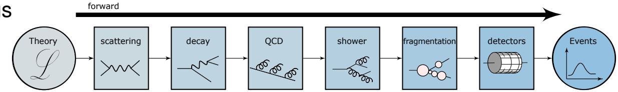
- start with Lagrangian
- calculate scattering using QFT
- simulate events
- simulate detectors

$\rightarrow$ LHC events in virtual worlds

## New physics searches

- compare simulations and data
- analyze data systematically
- understand LHC dataset [SM or BSM]
- publish useable results

$\rightarrow$ With a little help from data science...



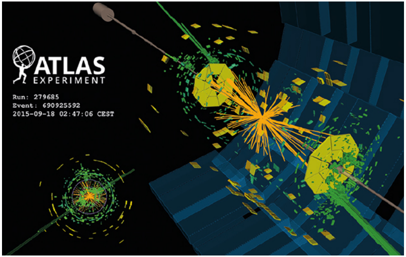
# LHC data

## Data from ATLAS & CMS

- protons on protons at $E \approx 13000 \times m_p \rightarrow$ relativistic kinematics
- crossing every 25 ns, 40 MHz, 1.6 MB per event $\rightarrow$ 1 PB/s
- frequency vs size

$$\frac{10\ \text{m}}{3 \times 10^8\ \text{m/s}} \approx 3 \times 10^{-8}\ \text{s} = 30\ \text{ns}$$

$\rightarrow$ Big and fast data



# LHC data

## Data from ATLAS & CMS

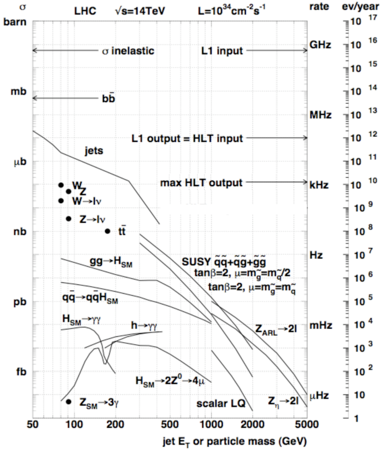
- protons on protons at $E \approx 13000 \times m_p \to$ relativistic kinematics
- crossing every 25 ns, 40 MHz, 1.6 MB per event $\to$ 1 PB/s
- frequency vs size

$$\frac{10\ \text{m}}{3 \times 10^8\ \text{m/s}} \approx 3 \times 10^{-8}\ \text{s} = 30\ \text{ns}$$

$\to$ Big and fast data

## Triggering

- $10^{-6}$ suppression physics-loss-less
- L1 hardware 40 MHz $\to$ 100 kHz
- L2/HL software $\to$ 3 kHz
- L3 software $\to$ 200 Hz, 320 MB/s



# LHC data

## Data from ATLAS & CMS

- protons on protons at $E \approx 13000 \times m_p \to$ relativistic kinematics
- crossing every 25 ns, 40 MHz, 1.6 MB per event $\to$ 1 PB/s
- frequency vs size

$$\frac{10\text{ m}}{3 \times 10^8\text{ m/s}} \approx 3 \times 10^{-8}\text{ s} = 30\text{ ns}$$

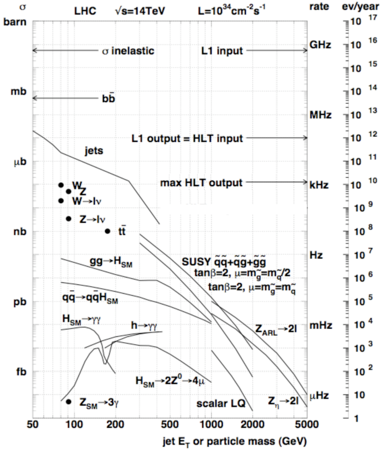
$\to$ Big and fast data

## Triggering

- $10^{-6}$ suppression physics-loss-less
- L1 hardware 40 MHz $\to$ 100 kHz
- L2/HL software $\to$ 3 kHz
- L3 software $\to$ 200 Hz, 320 MB/s

## Strategies

- classic trigger cuts
- probabilistic prescale trigger
- downsized data scouting



# LHC data

## Data from ATLAS & CMS

- protons on protons at $E \approx 13000 \times m_p \to$ relativistic kinematics
- crossing every 25 ns, 40 MHz, 1.6 MB per event $\to$ 1 PB/s
- frequency vs size

$$\frac{10\text{ m}}{3 \times 10^8\text{ m/s}} \approx 3 \times 10^{-8}\text{ s} = 30\text{ ns}$$

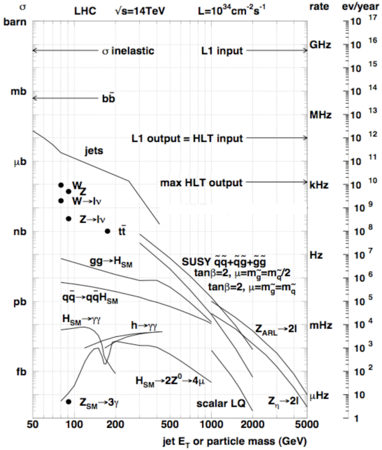
$\to$ Big and fast data

## Triggering

- $10^{-6}$ suppression physics-loss-less
- L1 hardware 40 MHz $\to$ 100 kHz
- L2/HL software $\to$ 3 kHz
- L3 software $\to$ 200 Hz, 320 MB/s

## ML-questions

- identification of interesting events?
- identification unexpected events?
- data compression for analyses?

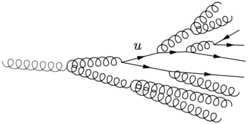


# Jets

## Partons as QCD jets

- most interactions $q\bar{q}, gg \to q\bar{q}, gg$

$$\sigma_{pp\to jj} \times \mathcal{L} \approx 10^8 \text{fb} \times \frac{80}{\text{fb}} \approx 10^{10} \text{ events}$$




- quarks/gluon visible as jets
  splittings described by QCD
  hadronization and hadron decays in jets
- jets as decay products

  67% $W \to jj$ 70% $Z \to jj$ 60% $H \to jj$ 67% $t \to jjj$ 60% $\tau \to j$ ...
- new physics in ‘dark jets’
- typical process $pp \to t\bar{t}H + \text{jets} \to bjj\ \bar{b}jj\ b\bar{b} + \text{jets}$

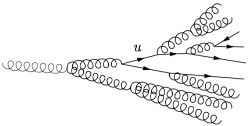
$\to$ Everywhere in LHC physics

# Jets

## Partons as QCD jets

- most interactions $q\bar{q}, gg \to q\bar{q}, gg$

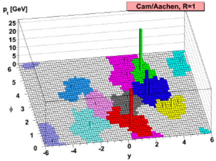
$$\sigma_{pp\to jj} \times \mathcal{L} \approx 10^8 \text{fb} \times \frac{80}{\text{fb}} \approx 10^{10} \text{ events}$$

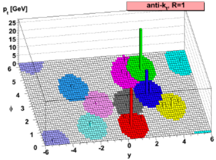



- quarks/gluon visible as jets
  splittings described by QCD
  hadronization and hadron decays in jets
- jets as decay products
  67% $W \to jj$ 70% $Z \to jj$ 60% $H \to jj$ 67% $t \to jjj$ 60% $\tau \to j$ ...
- new physics in ‘dark jets’
- typical process $pp \to t\bar{t}H + \text{jets} \to bjj\ \bar{b}jj\ b\bar{b} + \text{jets}$

$\to$ Everywhere in LHC physics

## Dealing with jets



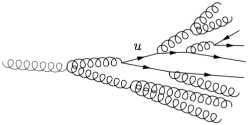

- 50-200 constituents per jet
  40 pile-up events on top
- calorimeter + tracking = particle-flow
- jet algorithms returning parton 4-momentum
- sub-jet physics new for LHC

# Jets

## Partons as QCD jets

- most interactions $q\bar{q}, gg \to q\bar{q}, gg$

$$\sigma_{pp\to jj} \times \mathcal{L} \approx 10^8 \text{fb} \times \frac{80}{\text{fb}} \approx 10^{10} \text{ events}$$




- quarks/gluon visible as jets
  splittings described by QCD
  hadronization and hadron decays in jets
- jets as decay products

  67% $W \to jj$ 70% $Z \to jj$ 60% $H \to jj$ 67% $t \to jjj$ 60% $\tau \to j$ ...
- new physics in ‘dark jets’
- typical process $pp \to t\bar{t}H + \text{jets} \to bjj\ \bar{b}jj\ b\bar{b} + \text{jets}$

$\to$ Everywhere in LHC physics

## ML-questions

- fast particle/parton identification?
- data denoising against jet radiation and pileup?
- combination of calorimeter and tracking resolution?
- combination of low-level and high-level observables?

# ML-tagging: nothing is ever new

## LHC visionaries

- 1991: NN-based quark-gluon tagger

### USING NEURAL NETWORKS TO IDENTIFY JETS

Leif LÖNNBLAD*, Carsten PETERSON** and Thorsteinn RÖGNVALDSSON***

*Department of Theoretical Physics, University of Lund, Sölvegatan 14A, S-22362 Lund, Sweden*

Received 29 June 1990

A neural network method for identifying the ancestor of a hadron jet is presented. The idea is to find an efficient mapping between certain observed hadronic kinematical variables and the quark-gluon identity. This is done with a neuronic expansion in terms of a network of sigmoidal functions using a gradient descent procedure, where the errors are back-propagated through the network. With this method we are able to separate gluon from quark jets originating from Monte Carlo generated $e^+e^-$ events with $\sim 85\%$ approach. The result is independent of the MC model used. This approach for isolating the gluon jet is then used to study the so-called string effect.

In addition, heavy quarks (b and c) in $e^+e^-$ reactions can be identified on the 50% level by just observing the hadrons. In particular we are able to separate b-quarks with an efficiency and purity, which is comparable with what is expected from vertex detectors. We also speculate on how the neural network method can be used to disentangle different hadronization schemes by compressing the dimensionality of the state space of hadrons.

# ML-tagging: nothing is ever new

## LHC visionaries

- 1991: NN-based quark-gluon tagger
- 1994: jet-algorithm $W$/top-tagger

**USING NEURAL NETWORKS TO IDENTIFY JETS**

Leif LÖNNBLAD*, Carsten PETERSON** and Thorsteinn RÖGNVALDSSON***

*Department of Theoretical Physics, University of Lund, Sölvegatan 14A, S-22362 Lund, Sweden*

Received 29 June 1990

A neural network method for identifying the ancestor of a hadron jet is presented. The idea is to find an efficient mapping between certain observed hadronic kinematical variables and the quark-gluon identity. This is done with a neuronic expansion in terms of a network of sigmoidal functions using a gradient descen
network. With this method we ar
Carlo generated $e^+e^-$ events v
model used. This approach for i
effect.

In addition, heavy quarks (b a
just observing the hadrons. In pa
purity, which is comparable with
how the neural network method
compressing the dimensionality c

**Searches for new particles using cone and cluster jet algorithms: a comparative study**

Michael H. Seymour

Department of Theoretical Physics, University of Lund, Sölvegatan 14A, S-22362 Lund, Sweden

Received 18 June 1993; in revised form 16 September 1993

**Abstract.** We discuss the reconstruction of the hadronic decays of heavy particles using jet algorithms. The ability to reconstruct the mass of the decaying particle is compared between a traditional cone-type algorithm and a recently proposed cluster-type algorithm. The specific examples considered are the semileptonic decays of a heavy Higgs boson at $\sqrt{s}=16$ TeV, and of top quark–antiquark pairs at $\sqrt{s}=1.8$ TeV. We find that the cluster algorithm offers considerable advantages in the former case, and a slight advantage in the latter. We briefly discuss the effects of calorimeter energy resolution, and show that a typical resolution dilutes these advantages, but does not remove them entirely.

except that the invariant mass of a pair is replaced by the transverse momentum of the softer particle relative to the other.

More recently, this algorithm was extended to collisions with incoming hadrons [5], and a longitudinally-invariant $k_\perp$-clustering algorithm for hadron-hadron collisions was proposed [6]. This algorithm has been compared with the more commonly used cone algorithm from the viewpoints of a parton-shower Monte Carlo program [6, 7], and a fixed-order matrix-element calculation [8], and advantages of the cluster algorithm were reported in both cases. This paper is concerned with a comparison between the algorithms for the task of reconstructing the hadronic decays of heavy particles, which was also studied in a preliminary way in [9].

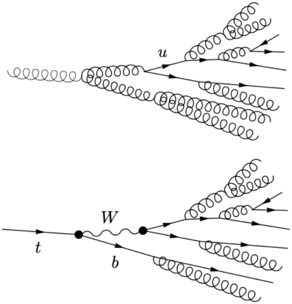
The only as-yet unobserved particles of the minimal Standard Model are the top quark and Higgs boson. The search for, and study of, these particles are among the most important goals of current and planned hadron-

# QCD jet representation

## Jet constituents

- historically

  only hard parton 4-momentum interesting
  parton content from 'tagging'
  QCD tests from theory observables



# QCD jet representation

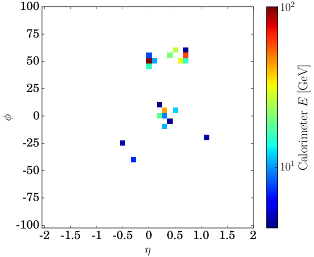
## Jet constituents

- historically

  only hard parton 4-momentum interesting
  parton content from 'tagging'
  QCD tests from theory observables

- ML-excitement phase [since 2015]

  data-driven jet analyses
  include as much data as possible
  avoid intermediate high-level variables
  calorimeter output as image



# QCD jet representation

## Jet constituents

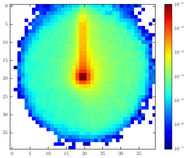
- historically

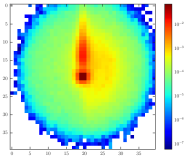
  only hard parton 4-momentum interesting
  parton content from 'tagging'
  QCD tests from theory observables

- ML-excitement phase [since 2015]

  data-driven jet analyses
  include as much data as possible
  avoid intermediate high-level variables
  calorimeter output as image

$\rightarrow$ Deep learning = modern networks on low-level observables





# QCD jet representation

## Jet constituents

- historically

  only hard parton 4-momentum interesting
  parton content from 'tagging'
  QCD tests from theory observables

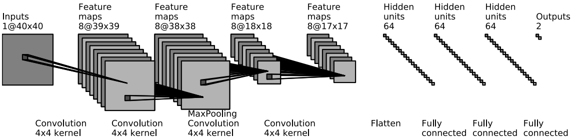
- ML-excitement phase [since 2015]

  data-driven jet analyses
  include as much data as possible
  avoid intermediate high-level variables
  calorimeter output as image

$\rightarrow$ Deep learning = modern networks on low-level observables

## Convolutional network

- image recognition standard ML task
- top tagging on 2D jet images
- 40 $\times$ 40 bins with calorimeter resolution



# Meet the professionals

## A brief history of achievement

- 2014/15: first jet image papers
- 2017: first (working) ML top tagger
- ML4Jets 2017: What architecture works best?
- ML4Jets 2018: Lots of architectures work

→ Jet classification understood and done

SciPost Physics Submission

### The Machine Learning Landscape of Top Taggers

G. Kasieczka (ed)[1], T. Plehn (ed)[2], A. Butter[2], K. Cranmer[3], D. Debnath[4], M. Fairbairn[5], W. Fedorko[6], C. Gay[6], L. Gouskos[7], P. T. Komiske[8], S. Leiss[1], A. Lister[6], S. Macaluso[3,4], E. M. Metodiev[8], L. Moore[9], B. Nachman,[10,11], K. Nordström[12,13], J. Pearkes[6], H. Qu[7], Y. Rath[14], M. Rieger[14], D. Shih[4], J. M. Thompson[2], and S. Varma[5]

**1** Institut für Experimentalphysik, Universität Hamburg, Germany
**2** Institut für Theoretische Physik, Universität Heidelberg, Germany
**3** Center for Cosmology and Particle Physics and Center for Data Science, NYU, USA
**4** NHECT, Dept. of Physics and Astronomy, Rutgers, The State University of NJ, USA
**5** Theoretical Particle Physics and Cosmology, King's College London, United Kingdom
**6** Department of Physics and Astronomy, The University of British Columbia, Canada
**7** Department of Physics, University of California, Santa Barbara, USA
**8** Center for Theoretical Physics, MIT, Cambridge, USA
**9** CP3, Université Catholique de Louvain, Louvain-la-Neuve, Belgium
**10** Physics Division, Lawrence Berkeley National Laboratory, Berkeley, USA
**11** Simons Inst. for the Theory of Computing, University of California, Berkeley, USA
**12** National Institute for Subatomic Physics (NIKHEF), Amsterdam, Netherlands
**13** LPTHE, CNRS & Sorbonne Université, Paris, France
**14** III. Physics Institute A, RWTH Aachen University, Germany

gregor.kasieczka@uni-hamburg.de
plehn@uni-heidelberg.de

April 12, 2019

#### Abstract

**Based on the established task of identifying boosted, hadronically decaying top quarks, we compare a wide range of modern machine learning approaches. We find that they are extremely powerful and great fun.**

**Content**

# Meet the professionals

## A brief history of achievement

- 2014/15: first jet image papers
- 2017: first (working) ML top tagger
- ML4Jets 2017: What architecture works best?
- ML4Jets 2018: Lots of architectures work

→ Jet classification understood and done

SciPost Physics | Submission

### The Machine Learning Landscape of Top Taggers

G. Kasieczka (ed)[1], T. Plehn (ed)[2], A. Butter[2], K. Cranmer[3], D. Debnath[4], M. Fairbairn[5], W. Fedorko[6], C. Gay[6], L. Gouskos[7], P. T. Komiske[8], S. Leiss[1], A. Lister[6], S. Macaluso[3,4], E. M. Metodiev[8], L. Moore[9], B. Nachman,[10,11], K. Nordström[12,13], J. Pearkes[6], H. Qu[7], Y. Rath[14], M. Rieger[14], D. Shih[4], J. M. Thompson[2], and S. Varma[5]

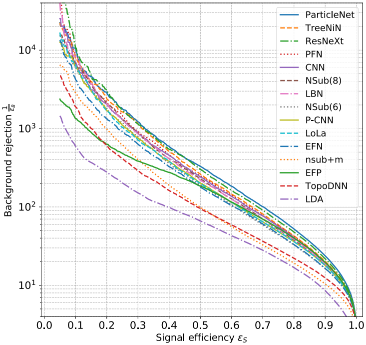
**1** Institut für Experimentalphysik, Universität Hamburg, Germany
**2** Institut für Theoretische Physik, Universität Heidelberg, Germany
**3** Center for Cosmology and Particle Physics and Center for Data Science, NYU, USA
**4** NHECT, Dept. of Physics and Astronomy, Rutgers, The State University of NJ, USA
**5** Theoretical Particle Physics and Cosmology, King's College London, United Kingdom
**6** Department of Physics and Astronomy, The University of British Columbia, Canada
**7** Department of Physics, University of California, Santa Barbara, USA
**8** Center for Theoretical Physics, MIT, Cambridge, USA
**9** CP3, Université Catholique de Louvain, Louvain-la-Neuve, Belgium
**10** Physics Division, Lawrence Berkeley National Laboratory, Berkeley, USA
**11** Simons Inst. for the Theory of Computing, University of California, Berkeley, USA
**12** National Institute for Subatomic Physics (NIKHEF), Amsterdam, Netherlands
**13** LPTHE, CNRS & Sorbonne Université, Paris, France
**14** III. Physics Institute A, RWTH Aachen University, Germany

gregor.kasieczka@uni-hamburg.de
plehn@uni-heidelberg.de

April 12, 2019

**Abstract**

Based on the established task of identifying boosted, hadronically decaying top quarks, we compare a wide range of modern machine learning approaches. We find that they are extremely powerful and great fun.



# Meet the professionals

## A brief history of achievement

- 2014/15: first jet image papers
- 2017: first (working) ML top tagger
- ML4Jets 2017: What architecture works best?
- ML4Jets 2018: Lots of architectures work

→ Jet classification understood and done

## Path to LHC reality

- application in analyses?
- beyond top and QCD jets?
- uncertainties?
- resilience in experimental reality?
- beyond fully supervised learning?
- from jets to events?
- analyses only ML will allow us to do?

etc

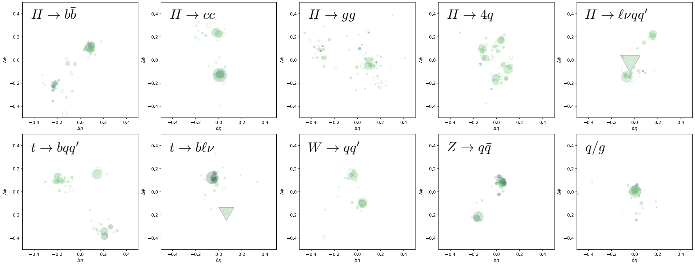
# State of the art [Huilin Qu, CMS]

## The Need for a Large Dataset

*H. Qu, C. Li, S. Qian, arXiv:2202.03772, https://github.com/jet-universe/particle_transformer/*

- **JetClass**: a new large-scale public jet dataset
  - 100M jets for training: ~ two orders of magnitude larger than existing public datasets
  - 10 classes: several unexplored scenarios, e.g., H->WW*->4q, H->WW*->ℓvqq, etc.
  - comprehensive information per particle: kinematics, particle ID, track displacement

*Simulated w/ MadGraph + Pythia + Delphes*



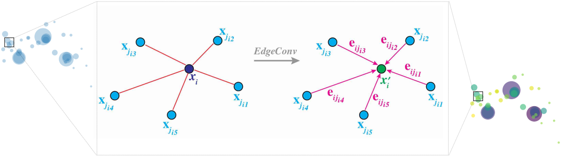
# State of the art [Huilin Qu, CMS]

## PARTICLENET

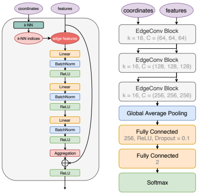
*H. Qu and L. Gouskos*
*Phys.Rev.D 101 (2020) 5, 056019*

- ParticleNet: jet tagging via particle clouds
  - treating a jet as an **unordered set of particles**, distributed in the $\eta - \phi$ space
  - **graph neural network architecture**, adapted from Dynamic Graph CNN [arXiv:1801.07829]
    - treating a point cloud as a graph: each point is a vertex
      - for each point, a local patch is defined by finding its k-nearest neighbors
    - designing a permutation-invariant "convolution" function
      - define "edge feature" for each center-neighbor pair: $e_{ij} = h_\Theta(x_i, x_j)$
      - aggregate the edge features in a symmetric way: $x_i' = \mathrm{mean}_j\, e_{ij}$

Jet Tagging in the Era of Deep Learning - June 9, 2022 - Huilin Qu (CERN)




*ParticleNet architecture*




*cf.* P.T. Komiske, E. M. Metodiev and J. Thaler, *JHEP 01 (2019) 121*;
V. Mikuni and F. Canelli, *Eur. Phys. J. Plus 135, 463 (2020)*; *Mach.Learn.Sci.Tech. 2 (2021) 3, 035027*.

# State of the art [Huilin Qu, CMS]

## PARTICLE TRANSFORMER

H. Qu, C. Li, S. Qian, arXiv:2202.03772, https://github.com/jet-universe/particle_transformer/

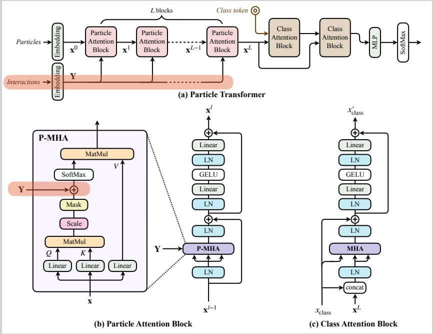
- **Attention mechanism and Transformers**: the new state-of-the-art architecture in ML
  - Large Language Models: BERT, GPT-3, ...
  - Computer Vision: ViT, Swin-T, ...
  - AlphaFold2 for protein structure prediction
- **Particle Transformer (ParT)**
  - Transformer-based architecture for jet tagging
  - injecting physics-inspired pairwise features to "bias" the dot-product self-attention

$$\text{P-MHA}(Q,K,V) = \text{SoftMax}(QK^T/\sqrt{d_k} + \mathbf{Y})V,$$

"Interaction" features

$$\begin{aligned}\Delta &= \sqrt{(y_a-y_b)^2+(\phi_a-\phi_b)^2},\\ k_\mathrm{T} &= \min(p_{\mathrm{T},a},p_{\mathrm{T},b})\Delta,\\ z &= \min(p_{\mathrm{T},a},p_{\mathrm{T},b})/(p_{\mathrm{T},a}+p_{\mathrm{T},b}),\\ m^2 &= (E_a+E_b)^2-\|\mathbf{p}_a+\mathbf{p}_b\|^2,\end{aligned}$$

*and more...*

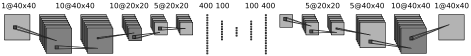





- we have a problem to solve
- progress never stops

→ LHC is all about performance

# Autoencoders

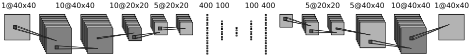



## Unsupervised classification

- train on background only
  extract unknown signal from reconstruction error
- reconstruct QCD jets $\rightarrow$ top jets hard to describe
- reconstruct top jets $\rightarrow$ QCD jets just simple top-like jet

$\rightarrow$ Symmetric performance $S \leftrightarrow B$?

# Autoencoders



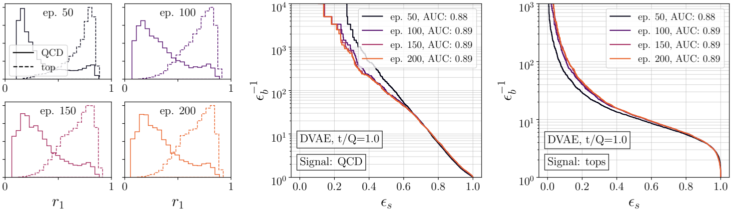

## Unsupervised classification

- train on background only
  extract unknown signal from reconstruction error
- reconstruct QCD jets $\to$ top jets hard to describe
- reconstruct top jets $\to$ QCD jets just simple top-like jet

$\to$ Symmetric performance $S \leftrightarrow B$?

## Moving to latent space

- anomaly score from latent space?
- VAE $\to$ does not work
  GMVAE $\to$ does not work
  Dirichlet VAE $\to$ works okay
  density estimation $\to$ does not work



# Normalized autoencoder

## Energy-based models

- goal    penalize features away from background
- train on normalized probability
- Boltzmann-distribution with $x \to E_\theta$ =MSE

$$p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z_\theta} \qquad \text{with} \qquad Z_\theta = \int_x dx e^{-E_\theta(x)}$$

$$L = -\Big\langle \log p_\theta(x) \Big\rangle_{p_\text{data}} = \Big\langle E_\theta(x) + \log Z_\theta \Big\rangle_{p_\text{data}}$$

$\to$ Small MSE for data, large MSE for model

# Normalized autoencoder

## Energy-based models

- goal    penalize features away from background
- train on normalized probability
- Boltzmann-distribution with $x \to E_\theta$ =MSE

$$p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z_\theta} \qquad \text{with} \qquad Z_\theta = \int_x dx e^{-E_\theta(x)}$$

$$L = -\Big\langle \log p_\theta(x) \Big\rangle_{p_\text{data}} = \Big\langle E_\theta(x) + \log Z_\theta \Big\rangle_{p_\text{data}}$$

- gradient of loss with normalization term

$$\begin{aligned}
-\nabla_\theta \log p_\theta(x) &= \nabla_\theta E_\theta(x) + \nabla_\theta \log Z_\theta \\
&= \nabla_\theta E_\theta(x) + \frac{1}{Z_\theta} \nabla_\theta \int_x dx e^{-E_\theta(x)} \\
&= \nabla_\theta E_\theta(x) - \int_x dx \frac{e^{-E_\theta(x)}}{Z_\theta} \nabla_\theta E_\theta(x) \\
&= \nabla_\theta E_\theta(x) - \Big\langle \nabla_\theta E_\theta(x) \Big\rangle_{p_\theta}
\end{aligned}$$

- background metric for expectation value

$$\Big\langle -\nabla_\theta \log p_\theta(x) \Big\rangle_{p_\text{data}} = \Big\langle \nabla_\theta E_\theta(x) \Big\rangle_{p_\text{data}} - \Big\langle \nabla_\theta E_\theta(x) \Big\rangle_{p_\theta}$$

$\to$ Small MSE for data, large MSE for model

# Normalized autoencoder

## Energy-based models

- goal    penalize features away from background
- train on normalized probability
- Boltzmann-distribution with $x \to E_\theta$ =MSE

$$p_\theta(x) = \frac{e^{-E_\theta(x)}}{Z_\theta} \qquad \text{with} \qquad Z_\theta = \int_x dx e^{-E_\theta(x)}$$

$$L = -\big\langle \log p_\theta(x) \big\rangle_{p_\text{data}} = \big\langle E_\theta(x) + \log Z_\theta \big\rangle_{p_\text{data}}$$

$\to$ Small MSE for data, large MSE for model

## Energy-based autoencoder

- still need to compute $Z_\theta$
  integration over phase space $x$
- (Langevin) Markov Chain

$$x_{t+1} = x_t + \lambda_x \nabla_x \log p_\theta(x) + \sigma_x \epsilon_t \qquad \text{with} \qquad \epsilon_t \sim \mathcal{N}_{0,1}$$

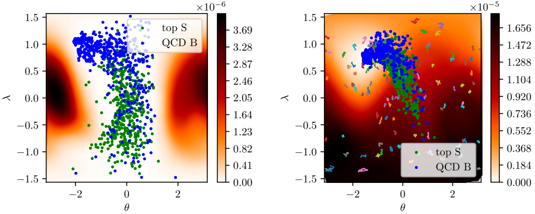
- problem    $x$-space high-dimensional and hard to model
  autoencoder    sample in and around latent space [physics manifold]
- MC abuse    100s of chains with 30 steps

$\to$ Autoencoder the perfect EBM

# NAE performance

## Top vs QCD autoencoding

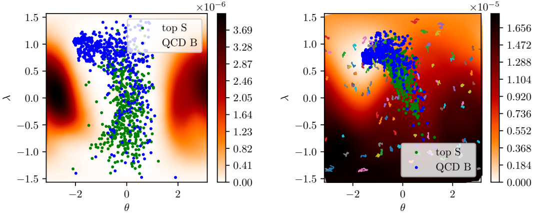
- regular autoencoder pre-training vs normalized training



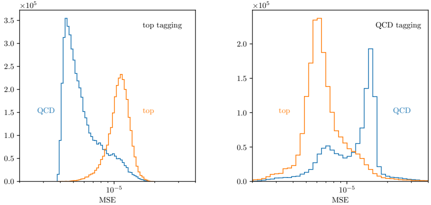
# NAE performance

## Top vs QCD autoencoding

- regular autoencoder pre-training vs normalized training




- MSE distributions for background and (unknown) signal




→ Still simple autoencoder with better training

# ML-Parton densities

## Dirty LHC secret

- proton-proton collisions from parton-parton predictions $[x = E_\text{parton}/E_\text{proton}]$

$$\sigma_\text{tot} = \int_0^1 dx_1 \int_0^1 dx_2 \sum_{\text{partons } ij} f_i(x_1)\, f_j(x_2)\, \hat{\sigma}_{ij}(x_1 x_2 E^2)$$

- DGLAP equation, including factorization scale $\mu$

$$\frac{df_i(x,\mu)}{d\log\mu^2} = \sum_{\text{partons} j} \int_x^1 \frac{dz}{z} \frac{\alpha_s}{2\pi} P_{i\leftarrow j}(z)\, f_j\left(\frac{x}{z},\mu\right) = \frac{\alpha_s}{2\pi} \sum_j \left(P_{i\leftarrow j} \otimes f_j\right)(x,\mu)$$

- historic parametrization

$$f_i(x,\mu_0) = a_0 x^{a_1} (1-x)^{a_2} e^{a_3 x + a_4 x^2}$$

$\to$ WTF... $\to$ lattice gauge theory?

# ML-Parton densities

## Dirty LHC secret

- proton-proton collisions from parton-parton predictions $[x = E_{\text{parton}} / E_{\text{proton}}]$

$$\sigma_{\text{tot}} = \int_0^1 dx_1 \int_0^1 dx_2 \sum_{\text{partons } ij} f_i(x_1)\, f_j(x_2)\, \hat{\sigma}_{ij}(x_1 x_2 E^2)$$

- DGLAP equation, including factorization scale $\mu$

$$\frac{df_i(x,\mu)}{d\log\mu^2} = \sum_{\text{partons} j} \int_x^1 \frac{dz}{z} \frac{\alpha_s}{2\pi} P_{i\leftarrow j}(z)\, f_j\left(\frac{x}{z},\mu\right) = \frac{\alpha_s}{2\pi} \sum_j \left(P_{i\leftarrow j} \otimes f_j\right)(x,\mu)$$

- historic parametrization

$$f_i(x,\mu_0) = a_0 x^{a_1} (1-x)^{a_2} e^{a_3 x + a_4 x^2}$$

$\to$ WTF... $\to$ lattice gauge theory?

## Non-parametric network fit

- parametrizations not useful
- bias problematic

$\to$ NNPDF

hep-ph/0204232
GeF/TH/3-02
RM3-TH/02-01

**Neural Network Parametrization of Deep–Inelastic Structure Functions**

Stefano Forte[a], Lluís Garrido[b], José I. Latorre[b] and Andrea Piccione[c]

[a]INFN, Sezione di Roma Tre
Via della Vasca Navale 84, I-00146 Rome, Italy

[b]Departament d'Estructura i Constituents de la Matèria, Universitat de Barcelona,
Diagonal 647, E-08028 Barcelona, Spain

[c]INFN sezione di Genova and Dipartimento di Fisica, Università di Genova,
via Dodecaneso 33, I-16146 Genova, Italy

**Abstract**

We construct a parametrization of deep-inelastic structure functions which retains information on experimental errors and correlations, and which does not introduce any theoretical bias while interpolating between existing data points. We generate a Monte Carlo sample of pseudo-data configurations and we train an ensemble of neural networks on them. This effectively provides us with a probability measure in the space of structure functions, within the whole kinematic region where data are available. This measure can then be used to determine the value of the structure function, its error, point–to–point correlations and generally the value and uncertainty of any function of the structure function itself. We apply this technique to the determination of the structure function $F_2$ of the proton and deuteron, and a precision determination of the isotriplet combination $F_2$[p-d]. We discuss in detail these results, check their stability and accuracy, and make them available in various formats for applications.

# ML-Parton densities

## Dirty LHC secret

- proton-proton collisions from parton-parton predictions $[x = E_{\text{parton}}/E_{\text{proton}}]$

$$\sigma_{\text{tot}} = \int_0^1 dx_1 \int_0^1 dx_2 \sum_{\text{partons } ij} f_i(x_1)\, f_j(x_2)\, \hat{\sigma}_{ij}(x_1 x_2 E^2)$$

- DGLAP equation, including factorization scale $\mu$

$$\frac{df_i(x,\mu)}{d\log\mu^2} = \sum_{\text{partons} j} \int_x^1 \frac{dz}{z} \frac{\alpha_s}{2\pi} P_{i\leftarrow j}(z)\, f_j\left(\frac{x}{z},\mu\right) = \frac{\alpha_s}{2\pi} \sum_j \left(P_{i\leftarrow j} \otimes f_j\right)(x,\mu)$$

- historic parametrization

$$f_i(x,\mu_0) = a_0 x^{a_1} (1-x)^{a_2} e^{a_3 x + a_4 x^2}$$

$\rightarrow$ WTF... $\rightarrow$ lattice gauge theory?

## Non-parametric network fit

- parametrizations not useful
- bias problematic

$\rightarrow$ NNPDF

### 6 Summary

We have presented a determination of the probability density in the space of structure functions for the structure function $F_2$ for proton, deuteron and nonsinglet, as determined from experimental data of the NMC and BCDMS collaborations. Our results, for each of the three structure functions, take the form of a set of 1000 neural nets, each of which gives a determination of $F_2$ for given $x$ and $Q^2$. The distribution of these functions is a Monte Carlo sampling of the probability density. This Monte Carlo sampling has been obtained by first, producing a sampling of the space of data points based on the available experimental information through a set of Monte Carlo replicas of the original data, and then, training each neural net to one of these replicas.

In practice, all functions are given by a FORTRAN routine which reproduces a feed–forward neural network (described in Section 3) entirely determined by a set of 47 real parameters. Each function is then specified by the set of values for these parameters. Our results are available at the web page `http://sophia.ecm.ub.es/f2neural/`. The full set of FORTRAN routines and parameters can be downloaded from this page. On–line plotting and computation facilities for

hep-ph/0204232
GeF/TH/3-02
RM3-TH/02-01

### Neural Network Parametrization of Deep–Inelastic Structure Functions

Stefano Forte[a], Lluís Garrido[b], José I. Latorre[b] and Andrea Piccione[c]

[a]INFN, Sezione di Roma Tre
Via della Vasca Navale 84, I-00146 Rome, Italy

[b]Departament d'Estructura i Constituents de la Matèria, Universitat de Barcelona,
Diagonal 647, E-08028 Barcelona, Spain

[c]INFN sezione di Genova and Dipartimento di Fisica, Università di Genova,
via Dodecaneso 33, I-16146 Genova, Italy

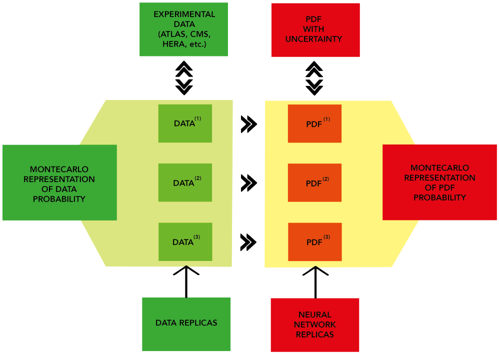
**Abstract**

We construct a parametrization of deep–inelastic structure functions which retains information on experimental errors and correlations, and which does not introduce any theoretical bias while interpolating between existing data points. We generate a Monte Carlo sample of pseudo–data configurations and we train an ensemble of neural networks on them. This effectively provides us with a probability measure in the space of structure functions, within the whole kinematic region where data are available. This measure can then be used to determine the value of the structure function, its error, point–to–point correlations and generally the value and uncertainty of any function of the structure function itself. We apply this technique to the determination of the structure function $F_2$ of the proton and deuteron, and a precision determination of the isotriplet combination $F_2$[p-d]. We discuss in detail these results, check their stability and accuracy, and make them available in various formats for applications.

# State of the art [Stefano Forte, NNPDF]

## THE FUNCTIONAL MONTE CARLO

REPLICA SAMPLE OF FUNCTIONS ⇔ PROBABILITY DENSITY IN FUNCTION SPACE
KNOWLEDGE OF LIKELIHHOD SHAPE (FUNCTIONAL FORM) NOT NECESSARY

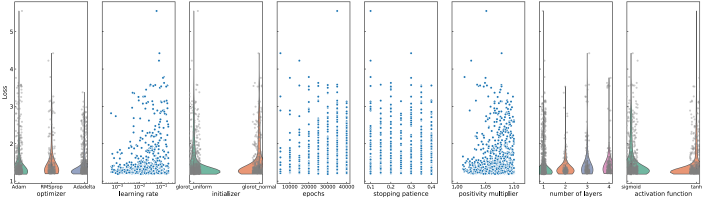



FINAL PDF SET: $f_i^{(a)}(x,\mu)$;
i =up, antiup, down, antidown, strange, antistrange, charm, gluon; $j = 1, 2, \dots N_{\text{rep}}$

# State of the art [Stefano Forte, NNPDF]

## FITTING THE METHODOLOGY

## HYPEROPTIMIZATION SCANS




## HYPEROPT PARAMETERS

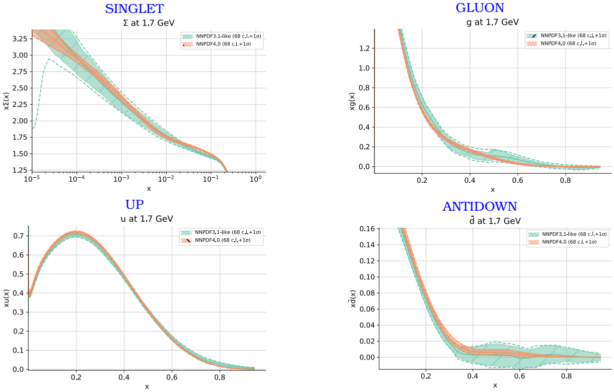
| NEURAL NETWORK | FIT OPTIONS |
|---|---|
| NUMBER OF LAYERS (*) | OPTIMIZER (*) |
| SIZE OF EACH LAYER | INITIAL LEARNING RATE (*) |
| DROPOUT | MAXIMUM NUMBER OF EPOCHS (*) |
| ACTIVATION FUNCTIONS (*) | STOPPING PATIENCE (*) |
| INITIALIZATION FUNCTIONS (*) | POSITIVITY MULTIPLIER (*) |

- SCAN PARAMETER SPACE
- OPTIMIZE FIGURE OF MERIT: VALIDATION $\chi^2$
- BAYESIAN UPDATING

# State of the art [Stefano Forte, NNPDF]

## NNPDF4.0 vs. NNPDF3.1

- FULL BACKWARD COMPATIBILITY
- SUBSTANTIAL REDUCTION IN UNCERTAINTY



# Simulation

## Event generation

- start from Lagrangian

$$\mathcal{L} = \sum_q \overline{\psi}_q \left( i\gamma^\mu \partial_\mu - m - gG_\mu \right) \psi_q - \frac{1}{4} G_{\mu\nu} G^{\mu\nu} + \dots - \mu^2 |\phi|^2 - \lambda |\phi|^4$$

- simulation factorized by energy
- Monte Carlo generation, LO or NLO in QCD
- production process particle decays
  QCD jet radiation
  QCD showering
  fragmentation/hadronization

$\rightarrow$ Theory task
Pythia, Madgraph, Sherpa, Herwig

arXiv:2203.07460v1 [hep-ph] 14 Mar 2022

**Machine Learning and LHC Event Generation**

Anja Butter[1,2], Tilman Plehn[1], Steffen Schumann[3] (Editors), Simon Badger[4], Sascha Caron[5, 6] Kyle Cranmer[7,8], Francesco Armando Di Bello[9], Etienne Dreyer[10], Stefano Forte[11], Sanmay Ganguly[12], Dorival Gonçalves[13], Eilam Gross[10], Theo Heimel[1], Gudrun Heinrich[14], Lukas Heinrich[15], Alexander Held[16], Stefan Höche[17], Jessica N. Howard[18], Philip Ilten[19], Joshua Isaacson[17], Timo Janßen[3], Stephen Jones[20], Marumi Kado[9,21], Michael Kagan[22], Gregor Kasieczka[23], Felix Kling[24], Sabine Kraml[25], Claudius Krause[26], Frank Krauss[20], Kevin Kröninger[27], Rahool Kumar Barman[13], Michel Luchmann[1], Vitaly Magerya[14], Daniel Maitre[20], Bogdan Malaescu[2], Fabio Maltoni[28,29], Till Martini[30], Olivier Mattelaer[28], Benjamin Nachman[31,32], Sebastian Pitz[1], Juan Rojo[33,34], Matthew Schwartz[35], David Shih[25], Frank Siegert[36], Roy Stegeman[11], Bob Stienen[5], Jesse Thaler[37], Rob Verheyen[38], Daniel Whiteson[18], Ramon Winterhalder[28], and Jure Zupan[19]

**Abstract**

**First-principle simulations are at the heart of the high-energy physics research program. They link the vast data output of multi-purpose detectors with fundamental theory predictions and interpretation. This review illustrates a wide range of applications of modern machine learning to event generation and simulation-based inference, including conceptional developments driven by the specific requirements of particle physics. New ideas and tools developed at the interface of particle physics and machine learning will improve the speed and precision of forward simulations, handle the complexity of collision data, and enhance inference as an inverse simulation problem.**

Submitted to the Proceedings of the US Community Study on the Future of Particle Physics (Snowmass)

# Simulation

## Event generation

- start from Lagrangian

$$\mathcal{L} = \sum_q \overline{\psi}_q \left( i\gamma^\mu \partial_\mu - m - gG_\mu \right) \psi_q - \frac{1}{4} G_{\mu\nu} G^{\mu\nu} + \dots - \mu^2 |\phi|^2 - \lambda |\phi|^4$$

- simulation factorized by energy
- Monte Carlo generation, LO or NLO in QCD
- production process particle decays
  QCD jet radiation
  QCD showering
  fragmentation/hadronization

$\to$ Theory task
  Pythia, Madgraph, Sherpa, Herwig

## ML-questions

- fast and precise surrogates?
- full phase space coverage?
- full feature mapping?
- variable-dimensional and high-dimensional phase spaces?
- improved data- and theory-driven models?

**Contents**

# Likelihood-based inference

## Unlabeled likelihood ratio [CWoLa]

- Neyman-Pearson lemma: LR optimal discriminator
- likelihood ratio for event samples

$$\mathrm{LR}(x) = \frac{p(x|H_{S+B})}{p(x|H_B)} = \frac{\mathrm{Pois}(n|s+b)\;\Pi_{j=1}^n f_{S+B}(x_j)}{\mathrm{Pois}(n|b)\;\Pi_{j=1}^n f_B(x_j)} = e^{-s}\left(\frac{s+b}{b}\right)^n \frac{\Pi_j f_{S+B}(x_j)}{\Pi_j f_B(x_j)}$$

- additive log-likelihood ratio

$$\mathrm{LLR}(x) = -s + \sum_j \log\left(1 + \frac{s f_S(x_j)}{b f_B(x_j)}\right)$$

- LLR from simulation and/or classifier

# Likelihood-based inference

## Unlabeled likelihood ratio [CWoLa]

- Neyman-Pearson lemma: LR optimal discriminator
- problem no signal and background samples to train on
  instead samples $p_j$ with signal fractions $f_j$ and background fractions $1 - f_j$
- phase space densities

$$\begin{pmatrix} p_1(x) \\ p_2(x) \end{pmatrix} = \begin{pmatrix} f_1 & 1 - f_1 \\ f_2 & 1 - f_2 \end{pmatrix} \begin{pmatrix} p_S(x) \\ p_B(x) \end{pmatrix}$$

$$\Leftrightarrow \qquad \begin{pmatrix} p_S(x) \\ p_B(x) \end{pmatrix} = \frac{1}{f_1 - f_2} \begin{pmatrix} 1 - f_2 & f_1 - 1 \\ -f_2 & f_1 \end{pmatrix} \begin{pmatrix} p_1(x) \\ p_2(x) \end{pmatrix}$$

- goal: train classifier to extract

$$\frac{p_S(x)}{p_B(x)} = \frac{(1 - f_2)p_1(x) + (f_1 - 1)p_2(x)}{-f_2 p_1(x) + f_1 p_2(x)}$$

# Likelihood-based inference

## Unlabeled likelihood ratio [CWoLa]

- Neyman-Pearson lemma: LR optimal discriminator
- problem no signal and background samples to train on
  instead samples $p_j$ with signal fractions $f_j$ and background fractions $1 - f_j$
- phase space densities

$$\begin{pmatrix} p_1(x) \\ p_2(x) \end{pmatrix} = \begin{pmatrix} f_1 & 1 - f_1 \\ f_2 & 1 - f_2 \end{pmatrix} \begin{pmatrix} p_S(x) \\ p_B(x) \end{pmatrix}$$

$$\Leftrightarrow \quad \begin{pmatrix} p_S(x) \\ p_B(x) \end{pmatrix} = \frac{1}{f_1 - f_2} \begin{pmatrix} 1 - f_2 & f_1 - 1 \\ -f_2 & f_1 \end{pmatrix} \begin{pmatrix} p_1(x) \\ p_2(x) \end{pmatrix}$$

- goal: train classifier to extract

$$\frac{p_S(x)}{p_B(x)} = \frac{(1 - f_2) p_1(x) + (f_1 - 1) p_2(x)}{-f_2 p_1(x) + f_1 p_2(x)}$$

- trick: train classifier for

$$\frac{p_1(x)}{p_2(x)} = \frac{f_1 p_S(x) + (1 - f_1) p_B(x)}{f_2 p_S(x) + (1 - f_2) p_B(x)} = \frac{f_1 \dfrac{p_S(x)}{p_B(x)} + 1 - f_1}{f_2 \dfrac{p_S(x)}{p_B(x)} + 1 - f_2}$$

$$\frac{d}{d(p_S/p_B)} \frac{p_1(x)}{p_2(x)} = \frac{f_1 \left[ f_2 \dfrac{p_S(x)}{p_B(x)} + 1 - f_2 \right] - f_2 \left[ f_1 \dfrac{p_S(x)}{p_B(x)} + 1 - f_1 \right]}{\left[ f_2 \dfrac{p_S(x)}{p_B(x)} + 1 - f_2 \right]^2} = \frac{f_1 - f_2}{\left[ f_2 \dfrac{p_S(x)}{p_B(x)} + 1 - f_2 \right]^2}$$

$\rightarrow$ Apply mixed instead of pure classifier

# Likelihood-based inference

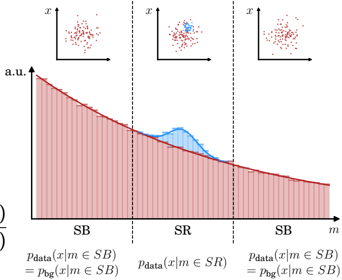
## Impoved bump hunts [CWoLa, Anode, Cathode]

- bump hunt in $m$
  orthogonal information in $x$

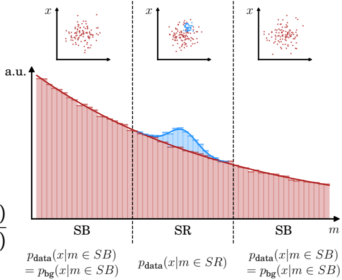
1. CWola on SB and SR samples

$$\frac{x \sim p_\text{data}(x|m \in SR)}{x \sim p_\text{data}(x|m \in SB)} \xrightarrow{\text{class}} \frac{p_{S+B}(x)}{p_B(x)} \to \frac{p_S(x)}{p_B(x)}$$

- but problem with correlations in $m$ and $x$



# Likelihood-based inference




## Impoved bump hunts [CWoLa, Anode, Cathode]

- bump hunt in $m$
  orthogonal information in $x$

1. CWola on SB and SR samples

$$\frac{x \sim p_{\text{data}}(x|m \in SR)}{x \sim p_{\text{data}}(x|m \in SB)} \xrightarrow{\text{class}} \frac{p_{S+B}(x)}{p_B(x)} \to \frac{p_S(x)}{p_B(x)}$$

- but problem with correlations in $m$ and $x$

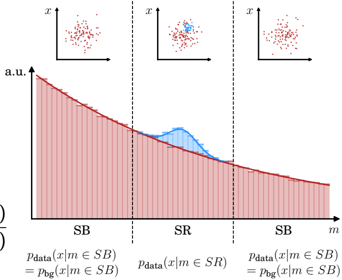
2. density estimation through normalizing flow

$$p_{\text{model}}(x|m \in SB) \xrightarrow{\text{interpol}} p_{\text{model}}(x|m \in SR)$$

- computable LR in signal regions

$$\text{LR}(x) = \frac{p_{\text{data}}(x|m \in SR)}{p_{\text{model}}(x|m \in SR)} \sim \frac{p_{S+B}(x)}{p_B(x)}$$

# Likelihood-based inference




## Impoved bump hunts [CWoLa, Anode, Cathode]

- bump hunt in $m$
  orthogonal information in $x$

1. CWola on SB and SR samples

$$\frac{x \sim p_\text{data}(x|m \in SR)}{x \sim p_\text{data}(x|m \in SB)} \stackrel{\text{class}}{\longrightarrow} \frac{p_{S+B}(x)}{p_B(x)} \to \frac{p_S(x)}{p_B(x)}$$

- but problem with correlations in $m$ and $x$

2. density estimation through normalizing flow

$$p_\text{model}(x|m \in SB) \stackrel{\text{interpol}}{\longrightarrow} p_\text{model}(x|m \in SR)$$

- computable LR in signal regions

$$\text{LR}(x) = \frac{p_\text{data}(x|m \in SR)}{p_\text{model}(x|m \in SR)} \sim \frac{p_{S+B}(x)}{p_B(x)}$$

3. background generation using normalizing flow

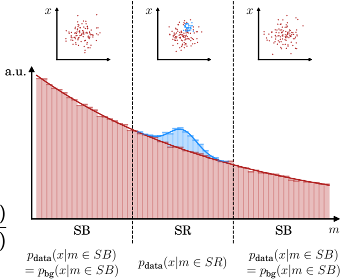
$$p_\text{model}(x|m \in SB) \stackrel{\text{sample}}{\longrightarrow} x \sim p_\text{model}(x|m \in SR)$$

- classifier on event samples

$$\frac{x \sim p_\text{model}(x|m \in SR)}{x \sim p_\text{model}(x|m \in SB)} \stackrel{\text{class}}{\longrightarrow} \frac{p_{S+B}(x)}{p_B(x)}$$

$\to$ Guess which works best?

# Likelihood-based inference




## Impoved bump hunts [CWoLa, Anode, Cathode]

- bump hunt in $m$
  orthogonal information in $x$

1. CWola on SB and SR samples

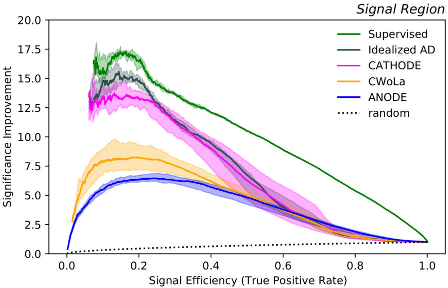
$$\frac{x \sim p_{\text{data}}(x|m \in SR)}{x \sim p_{\text{data}}(x|m \in SB)} \xrightarrow{\text{class}} \frac{p_{S+B}(x)}{p_B(x)} \to \frac{p_S(x)}{p_B(x)}$$

2. density estimation through normalizing flow

$$p_{\text{model}}(x|m \in SB) \xrightarrow{\text{interpol}} p_{\text{model}}(x|m \in SR)$$

3. background generation using normalizing flow

$$p_{\text{model}}(x|m \in SB) \xrightarrow{\text{sample}} x \sim p_{\text{model}}(x|m \in SR)$$



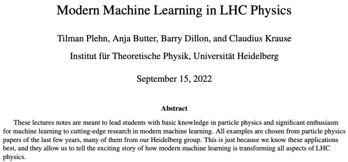
# ML-LHC introduction

## Summary

- particle physics has questions
- LHC is big and fast data
- data needs regression and classification
- knowledge comes through theory and simulation
- stochastic data and uncertainty craziness
- check out Heidelberg lecture notes

## Outlook

1. introduction (done)
2. normalizing flows, tutorial [Theo]
3. uncertainties and Bayesian networks [TP]
4. generative inversion and inference [Theo]



Modern Machine Learning in LHC Physics

Tilman Plehn, Anja Butter, Barry Dillon, and Claudius Krause

Institut für Theoretische Physik, Universität Heidelberg

September 15, 2022

**Abstract**

These lectures notes are meant to lead students with basic knowledge in particle physics and significant enthusiasm for machine learning to cutting-edge research in modern machine learning. All examples are chosen from particle physics papers of the last few years, many of them from our Heidelberg group. This is just because we know these applications best, and they allow us to tell the exciting story of how modern machine learning is transforming all aspects of LHC physics.