LHC

Neural networks

Examples

Amplitudes

Generative AI

MadNIS

Transformation

Machine Learning with Precision and Error Bars

Tilman Plehn

Universität Heidelberg

Brookhaven National Lab, March 2025

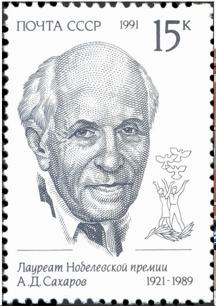
LHC: precision & uncertainties

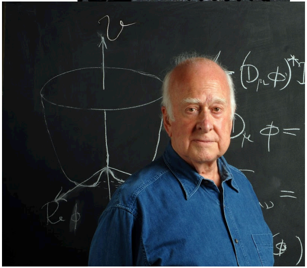
Classic motivation

- dark matter?
- matter vs antimatter?
- origin of Higgs boson?

Strengths

- fundamental questions
- huge, complex data set
- first-principle, precision simulations





LHC: precision & uncertainties

Classic motivation

- dark matter?
- matter vs antimatter?
- origin of Higgs boson?

Strengths

- fundamental questions
- huge, complex data set
- first-principle, precision simulations

First-principle simulations

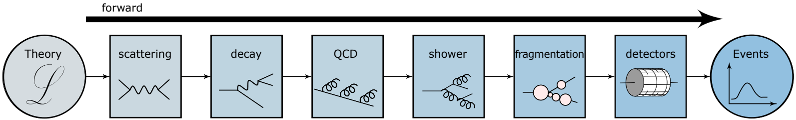
- start with Lagrangian
- calculate scattering using QFT
- simulate collisions
- simulate detectors

→ LHC events in virtual worlds

Searches and measurements

- compare simulations and data
- infer underlying theory [SM or BSM]
- publish data to re-interpret

→ Understand LHC data systematically



Brief ML-intro

Similar to fit

- approximate $f_\theta(x) \approx f(x)$
- no function, but very many θ
- data representation θ

Applications

- regression $x \to f_\theta(x)$
- classification $x \to p_\theta(x) \in [0, 1]$
- generation $r \sim \mathcal{N} \to p_\theta(r)$
- conditional generation $r \sim \mathcal{N} \to p_\theta(r|x)$

LHC

- training on simulations
- x always interpretable phase space
- symmetries, locality, etc known

$\rightarrow$ Benefitting from complexity?!

Training and uncertainties

Learned scalar field $f_\theta(x) \approx f(x)$

- maximize parameter probability given (f_j, σ_j)

$$\theta = \text{argmax } p(\theta|x) = \text{argmax } \frac{p(x|\theta)\; p(\theta)}{p(x)}$$

$\to$ Gaussian likelihood loss

$$p(x|\theta) \propto \prod_j \exp\left(- \frac{|f_j - f_\theta(x_j)|^2}{2\sigma_j^2} \right)$$

$$\Rightarrow \qquad \mathcal{L} \equiv -\log p(x|\theta) = \sum_j \frac{|f_j - f_\theta(x_j)|^2}{2\sigma_j^2}$$

Training and uncertainties

Learned scalar field $f_\theta(x) \approx f(x)$

- maximize parameter probability given (f_j, σ_j)

$$\theta = \text{argmax } p(\theta|x) = \text{argmax } \frac{p(x|\theta)\; p(\theta)}{p(x)}$$

→ Gaussian likelihood loss

$$p(x|\theta) \propto \prod_j \exp\left(- \frac{|f_j - f_\theta(x_j)|^2}{2\sigma_j^2} \right)$$

$$\Rightarrow \qquad \mathcal{L} \equiv - \log p(x|\theta) = \sum_j \frac{|f_j - f_\theta(x_j)|^2}{2\sigma_j^2}$$

Unknown uncertainties

- loss including normalization

$$\mathcal{L} = \frac{|f(x) - f_\theta(x)|^2}{2\sigma_\theta(x)^2} + \log \sigma_\theta(x) + \cdots$$

- if needed replace with Gaussian mixture model

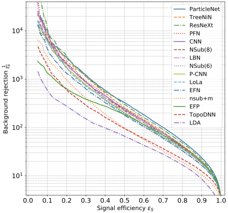
→ Learning function and (systematic) uncertainty

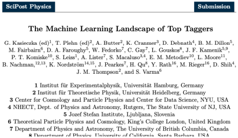
ML in experiment

Top tagging [classification, 2016-today]

- ‘hello world’ of LHC-ML
- end of QCD-taggers
- ever-improving [Huilin Qu]

→ Driving NN-architectures

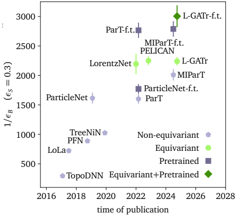



SciPost Physics — Submission

The Machine Learning Landscape of Top Taggers

G. Kasieczka (ed)[1], T. Plehn (ed)[2], A. Butter[2], K. Cranmer[3], D. Debnath[4], B. M. Dillon[5], M. Fairbairn[6], D. A. Faroughy[5], W. Fedorko[7], C. Gay[7], L. Gouskos[8], J. F. Kamenik[5,9], P. T. Komiske[10], S. Leiss[1], A. Lister[7], S. Macaluso[3,4], E. M. Metodiev[10], L. Moore[11], B. Nachman,[12,13], K. Nordström[14,15], J. Pearkes[7], H. Qu[8], Y. Rath[16], M. Rieger[16], D. Shih[4], J. M. Thompson[2], and S. Varma[6]

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 Jozef Stefan Institute, Ljubljana, Slovenia
6 Theoretical Particle Physics and Cosmology, King’s College London, United Kingdom
7 Department of Physics and Astronomy, The University of British Columbia, Canada
8 Department of Physics, University of California, Santa Barbara, USA

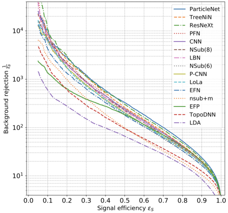


ML in experiment

Top tagging [classification, 2016-today]

- 'hello world' of LHC-ML
- end of QCD-taggers
- ever-improving [Huilin Qu]

→ Driving NN-architectures



SciPost Physics | Submission

The Machine Learning Landscape of Top Taggers

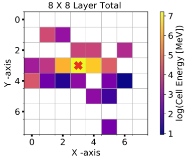
G. Kasieczka (ed)[1], T. Plehn (ed)[2], A. Butter[2], K. Cranmer[3], D. Debnath[4], B. M. Dillon[5], M. Fairbairn[6], D. A. Faroughy[5], W. Fedorko[7], C. Gay[7], L. Gouskos[8], J. F. Kamenik[5,9], P. T. Komiske[10], S. Leiss[1], A. Lister[7], S. Macaluso[3,4], E. M. Metodiev[10], L. Moore[11], B. Nachman,[12,13], K. Nordström[14,15], J. Pearkes[7], H. Qu[8], Y. Rath[16], M. Rieger[16], D. Shih[4], J. M. Thompson[2], and S. Varma[6]

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 Jozef Stefan Institute, Ljubljana, Slovenia
6 Theoretical Particle Physics and Cosmology, King's College London, United Kingdom
7 Department of Physics and Astronomy, The University of British Columbia, Canada
8 Department of Physics, University of California, Santa Barbara, USA
9 Faculty of Mathematics and Physics, University of Ljubljana, Ljubljana, Slovenia
10 Center for Theoretical Physics, MIT, Cambridge, USA
11 CP3, Universitéxx Catholique de Louvain, Louvain-la-Neuve, Belgium
12 Physics Division, Lawrence Berkeley National Laboratory, Berkeley, USA
13 Simons Inst. for the Theory of Computing, University of California, Berkeley, USA
14 National Institute for Subatomic Physics (NIKHEF), Amsterdam, Netherlands
15 LPTHE, CNRS & Sorbonne Université, Paris, France
16 III. Physics Institute A, RWTH Aachen University, Germany

Particle flow [2020-today]

- basis of jet analyses
- combining detectors with different resolution

→ Optimality the key



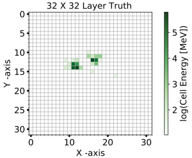
Towards a Computer Vision Particle Flow ⋆

Francesco Armando Di Bello[a,3], Sanmay Ganguly[b,1], Eilam Gross[1], Marumi Kado[3,4], Michael Pitt[2], Lorenzo Santi [3], Jonathan Shlomi[1]

[1]Weizmann Institute of Science, Rehovot 76100, Israel
[2]CERN, CH 1211, Geneva 23, Switzerland
[3]Università di Roma Sapienza, Piazza Aldo Moro, 2, 00185 Roma, Italy e INFN, Italy
[4]Université Paris-Saclay, CNRS/IN2P3, IJCLab, 91405, Orsay, France

Progress towards an improved particle flow algorithm at CMS with machine learning

Farouk Mokhtar[1], Joosep Pata[2], Javier Duarte[1], Eric Wulff[3], Maurizio Pierini[3] and Jean-Roch Vlimant[4]
(on behalf of the CMS Collaboration)
[1]University of California San Diego, La Jolla, CA 92093, USA
[2]NICPB, Rävala pst 10, 10143 Tallinn, Estonia
[3]European Organization for Nuclear Research (CERN), CH 1211, Geneva 23, Switzerland
[4]California Institute of Technology, Pasadena, CA 91125, USA
E-mail: fmokhtar@ucsd.edu, joosep.pata@cern.ch, jduarte@ucsd.edu

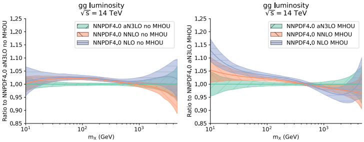


ML in phenomenology

Parton densities [NNPDF, 2002-today]

- LHC-ML classic
- pdfs with uncertainties and without bias

→ Driving precision



The Path to N^{3}LO Parton Distributions

The NNPDF Collaboration:
Richard D. Ball[1], Andrea Barontini[2], Alessandro Candido[2,3], Stefano Carrazza[2], Juan Cruz-Martinez[3], Luigi Del Debbio[1], Stefano Forte[2], Tommaso Giani[4,5], Felix Hekhorn[2,6,7], Zahari Kassabov[8], Niccolò Laurenti,[2] Giacomo Magni[4,5], Emanuele R. Nocera[9], Tanjona R. Rabemananjara[4,5], Juan Rojo[4,5], Christopher Schwan[10], Roy Stegeman[1], and Maria Ubiali[8]

[1] *The Higgs Centre for Theoretical Physics, University of Edinburgh, JCMB, KB, Mayfield Rd, Edinburgh EH9 3JZ, Scotland*
[2] *Tif Lab, Dipartimento di Fisica, Università di Milano and INFN, Sezione di Milano, Via Celoria 16, I-20133 Milano, Italy*
[3] *CERN, Theoretical Physics Department, CH-1211 Geneva 23, Switzerland*
[4] *Department of Physics and Astronomy, Vrije Universiteit, NL-1081 HV Amsterdam*
[5] *Nikhef Theory Group, Science Park 105, 1098 XG Amsterdam, The Netherlands*
[6] *University of Jyvaskyla, Department of Physics, P.O. Box 35, FI-40014 University of Jyvaskyla, Finland*
[7] *Helsinki Institute of Physics, P.O. Box 64, FI-00014 University of Helsinki, Finland*
[8] *DAMTP, University of Cambridge, Wilberforce Road, Cambridge, CB3 0WA, United Kingdom*
[9] *Dipartimento di Fisica, Università degli Studi di Torino and INFN, Sezione di Torino, Via Pietro Giuria 1, I-10125 Torino, Italy*
[10] *Universität Würzburg, Institut für Theoretische Physik und Astrophysik, 97074 Würzburg, Germany*

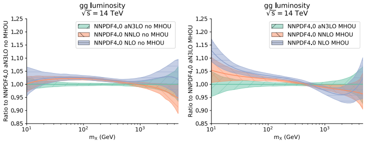
This paper is dedicated to the memory of Stefano Catani, Grand Master of QCD, great scientist and human being

ML in phenomenology

Parton densities [NNPDF, 2002-today]

- LHC-ML classic
- pdfs with uncertainties and without bias

→ Driving precision



The Path to N^{3}LO Parton Distributions

The NNPDF Collaboration:
Richard D. Ball[1], Andrea Barontini[2], Alessandro Candido[2,3], Stefano Carrazza[2], Juan Cruz-Martinez[3], Luigi Del Debbio[1], Stefano Forte[2], Tommaso Giani[4,5], Felix Hekhorn[2,6,7], Zahari Kassabov[8], Niccolò Laurenti,[2] Giacomo Magni[4,5], Emanuele R. Nocera[9], Tanjona R. Rabemananjara[4,5], Juan Rojo[4,5], Christopher Schwan[10], Roy Stegeman[1], and Maria Ubiali[8]

[1] *The Higgs Centre for Theoretical Physics, University of Edinburgh, JCMB, KB, Mayfield Rd, Edinburgh EH9 3JZ, Scotland*
[2] *Tif Lab, Dipartimento di Fisica, Università di Milano and INFN, Sezione di Milano, Via Celoria 16, I-20133 Milano, Italy*
[3] *CERN, Theoretical Physics Department, CH-1211 Geneva 23, Switzerland*
[4] *Department of Physics and Astronomy, Vrije Universiteit, NL-1081 HV Amsterdam*
[5] *Nikhef Theory Group, Science Park 105, 1098 XG Amsterdam, The Netherlands*
[6] *University of Jyvaskyla, Department of Physics, P.O. Box 35, FI-40014 University of Jyvaskyla, Finland*
[7] *Helsinki Institute of Physics, P.O. Box 64, FI-00014 University of Helsinki, Finland*
[8] *DAMTP, University of Cambridge, Wilberforce Road, Cambridge, CB3 0WA, United Kingdom*
[9] *Dipartimento di Fisica, Università degli Studi di Torino and INFN, Sezione di Torino, Via Pietro Giuria 1, I-10125 Torino, Italy*
[10] *Universität Würzburg, Institut für Theoretische Physik und Astrophysik, 97074 Würzburg, Germany*

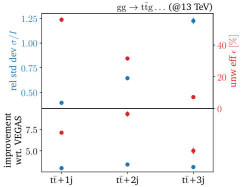
This paper is dedicated to the memory of Stefano Catani, Grand Master of QCD, great scientist and human being

Ultra-fast simulations [Sherpa, MadNIS, MLHad]

- event generation modular
- better ML-modules

→ MadNIS → MadGraph7

Triple-W	$u\bar{d} \to W^+W^+W^-$		
VBS	$uc \to W^+W^+\, ds$		
W+jets	$gg \to W^+ d\bar{u}$	$gg \to W^+ d\bar{u}g$	$gg \to W^+ d\bar{u}gg$
$t\bar{t}$+jets	$gg \to t\bar{t} + g$	$gg \to t\bar{t} + gg$	$gg \to t\bar{t} + ggg\,,$



SciPost Physics Submission

The MadNIS Reloaded

Theo Heimel[1], Nathan Huetsch[1], Fabio Maltoni[2,3], Olivier Mattelaer[2], Tilman Plehn[1], and Ramon Winterhalder[2]

1 Institut für Theoretische Physik, Universität Heidelberg, Germany
2 CP3, Université catholique de Louvain, Louvain-la-Neuve, Belgium
3 Dipartimento di Fisica e Astronomia, Universitá di Bologna, Italy

December 17, 2024

Abstract

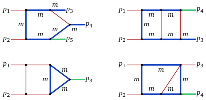
In pursuit of precise and fast theory predictions for the LHC, we present an implementation of the MadNIS method in the MadGraph event generator. A series of improvements in MadNIS further enhance its efficiency and speed. We validate this implementation for realistic partonic processes and find significant gains from using modern machine learning in event generators.

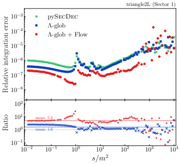
ML in theory

Optimizing integration paths [invertible networks]

- compute Feynman integrals
- learn optimal integration pat

→ To be implemented...





SciPost SciPost Phys. 12, 129 (2022)

Targeting multi-loop integrals with neural networks

Ramon Winterhalder[1,2,3], Vitaly Magerya[4], Emilio Villa[4], Stephen P. Jones[5], Matthias Kerner[4,6], Anja Butter[1,2], Gudrun Heinrich[2,4] and Tilman Plehn[1,2]

1 Institut für Theoretische Physik, Universität Heidelberg, Germany
2 HEiKA - Heidelberg Karlsruhe Strategic Partnership, Heidelberg University, Karlsruhe Institute of Technology (KIT), Germany
3 Centre for Cosmology, Particle Physics and Phenomenology (CP3), Université catholique de Louvain, Belgium
4 Institut für Theoretische Physik, Karlsruher Institut für Technologie, Germany
5 Institute for Particle Physics Phenomenology, Durham University, UK
6 Institut für Astroteilchenphysik, Karlsruher Institut für Technologie, Germany

Abstract

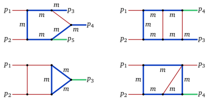
Numerical evaluations of Feynman integrals often proceed via a deformation of the integration contour into the complex plane. While valid contours are easy to construct, the numerical precision for a multi-loop integral can depend critically on the chosen contour. We present methods to optimize this contour using a combination of optimized, global complex shifts and a normalizing flow. They can lead to a significant gain in precision.

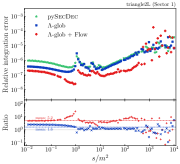
ML in theory

Optimizing integration paths [invertible networks]

- compute Feynman integrals
- learn optimal integration pat

→ To be implemented...





SciPost Phys. 12, 129 (2022)

Targeting multi-loop integrals with neural networks

Ramon Winterhalder[1,2,3], Vitaly Magerya[4], Emilio Villa[4], Stephen P. Jones[5], Matthias Kerner[4,6], Anja Butter[1,2], Gudrun Heinrich[2,4] and Tilman Plehn[1,2]

1 Institut für Theoretische Physik, Universität Heidelberg, Germany
2 HEiKA - Heidelberg Karlsruhe Strategic Partnership, Heidelberg University, Karlsruhe Institute of Technology (KIT), Germany
3 Centre for Cosmology, Particle Physics and Phenomenology (CP3), Université catholique de Louvain, Belgium
4 Institut für Theoretische Physik, Karlsruher Institut für Technologie, Germany
5 Institute for Particle Physics Phenomenology, Durham University, UK
6 Institut für Astroteilchenphysik, Karlsruher Institut für Technologie, Germany

Abstract

Numerical evaluations of Feynman integrals often proceed via a deformation of the integration contour into the complex plane. While valid contours are easy to construct, the numerical precision for a multi-loop integral can depend critically on the chosen contour. We present methods to optimize this contour using a combination of optimized, global complex shifts and a normalizing flow. They can lead to a significant gain in precision.

String landscape [reinforcement learning]

- searching for viable vacua
- high dimensions, unknown global structure

→ Islands of Standard Model?

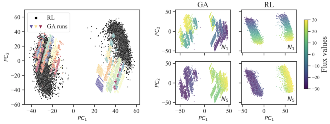


Figure 1: *Left:* Cluster structure in dimensionally reduced flux samples for RL and 25 GA runs (PCA on all samples of GA and RL). The colors indicate individual GA runs. *Right:* Dependence on flux (input) values (N_3 and N_5 respectively) in relation to principal components for a PCA fit of the individual output of GA and RL.

Probing the Structure of String Theory Vacua with Genetic Algorithms and Reinforcement Learning

Alex Cole
University of Amsterdam
a.e.cole@uva.nl

Sven Krippendorf
Arnold Sommerfeld Center for Theoretical Physics
LMU Munich
sven.krippendorf@physik.uni-muenchen.de

Andreas Schachner
Centre for Mathematical Sciences
University of Cambridge
as2673@cam.ac.uk

Gary Shiu
University of Wisconsin-Madison
shiu@physics.wisc.edu

Abstract

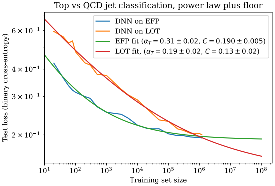
Identifying string theory vacua with desired physical properties at low energies requires searching through high-dimensional solution spaces – collectively referred to as the string landscape. We highlight that this search problem is amenable to reinforcement learning and genetic algorithms. In the context of flux vacua, we are able to reveal novel features (suggesting previously unidentified symmetries) in the string theory solutions required for properties such as the string coupling. In order to identify these features robustly, we combine results from both search methods, which we argue is imperative for reducing sampling bias.

Theory for ML

Scaling laws for classification networks [statistical learning]

- networks are complex systems
- training as statistical process

→ Now solving problems



SCALING LAWS IN JET CLASSIFICATION

Joshua Batson*
Independent Researcher
Oakland, CA 94607
joshua.batson@gmail.com

Yonatan Kahn
Center for Artificial Intelligence Innovation and
Department of Physics
University of Illinois Urbana-Champaign
Urbana, IL 61801
yfkahn@illinois.edu

ABSTRACT

We demonstrate the emergence of scaling laws in the benchmark top versus QCD jet classification problem in collider physics. Six distinct physically-motivated classifiers exhibit power-law scaling of the binary cross-entropy test loss as a function of training set size, with distinct power law indices. This result highlights the importance of comparing classifiers as a function of dataset size rather than for a fixed training set, as the optimal classifier may change considerably as the dataset is scaled up. We speculate on the interpretation of our results in terms of previous models of scaling laws observed in natural language and image datasets.

Collective variables of neural networks: empirical time evolution and scaling laws

Samuel Tovey
Institute for Computational Physics
University of Stuttgart
Stuttgart, Germany, 70569
stovey@icp.uni-stuttgart.de

Sven Krippendorf
Cavendish Laboratory and DAMTP
University of Cambridge
Cambridge, United Kingdom, CB3 0WA
slk38@cam.ac.uk

Michael Spannowsky
Institute for Particle Physics Phenomenology
Department of Physics
Durham University
Durham, DH1 3LE, U.K

Konstantin Nikolaou
Institute for Computational Physics
University of Stuttgart
Stuttgart, Germany, 70569

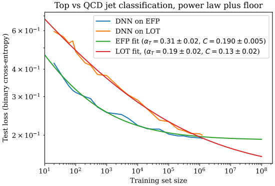
Christian Holm
Institute for Computational Physics
University of Stuttgart
Stuttgart, Germany, 70569

Theory for ML

Scaling laws for classification networks [statistical learning]

- networks are complex systems
- training as statistical process

→ Now solving problems



SCALING LAWS IN JET CLASSIFICATION

Joshua Batson*
Independent Researcher
Oakland, CA 94607
joshua.batson@gmail.com

Yonatan Kahn
Center for Artificial Intelligence Innovation and
Department of Physics
University of Illinois Urbana-Champaign
Urbana, IL 61801
yfkahn@illinois.edu

ABSTRACT

We demonstrate the emergence of scaling laws in the benchmark top versus QCD jet classification problem in collider physics. Six distinct physically-motivated classifiers exhibit power-law scaling of the binary cross-entropy test loss as a function of training set size, with distinct power law indices. This result highlights the importance of comparing classifiers as a function of dataset size rather than for a fixed training set, as the optimal classifier may change considerably as the dataset is scaled up. We speculate on the interpretation of our results in terms of previous models of scaling laws observed in natural language and image datasets.

Collective variables of neural networks: empirical time evolution and scaling laws

Samuel Tovey
Institute for Computational Physics
University of Stuttgart
Stuttgart, Germany, 70569
stovey@icp.uni-stuttgart.de

Sven Krippendorf
Cavendish Laboratory and DAMTP
University of Cambridge
Cambridge, United Kingdom, CB3 0WA
slk38@cam.ac.uk

Michael Spannowsky
Institute for Particle Physics Phenomenology
Department of Physics
Durham University
Durham, DH1 3LE, U.K

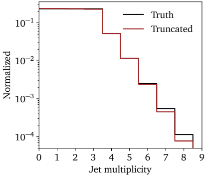
Konstantin Nikolaou
Institute for Computational Physics
University of Stuttgart
Stuttgart, Germany, 70569

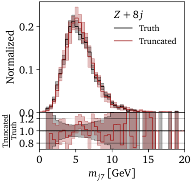
Christian Holm
Institute for Computational Physics
University of Stuttgart
Stuttgart, Germany, 70569

Extrapolating transformers

- train on QCD jet radiation
- learn to generate universal patterns

→ Extrapolation at work





SciPost Physics — Submission

Extrapolating Jet Radiation with Autoregressive Transformers

Anja Butter[1,2], François Charton[3], Javier Mariño Villadamigo[1], Ayodele Ore[1], Tilman Plehn[1,4], and Jonas Spinner[1]

1 Institut für Theoretische Physik, Universität Heidelberg, Germany
2 LPNHE, Sorbonne Université, Université Paris Cité, CNRS/IN2P3, Paris, France
3 Meta FAIR, CERMICS - Ecole des Ponts
4 Interdisciplinary Center for Scientific Computing (IWR), Universität Heidelberg, Germany

December 17, 2024

Abstract

Generative networks are an exciting tool for fast LHC event generation. Usually, they are used to generate configurations with a fixed number of particles. Autoregressive transformers allow us to generate events with variable numbers of particles, very much in line with the physics of QCD jet radiation. We show how they can learn a factorized likelihood for jet radiation and extrapolate in terms of the number of generated jets. For this extrapolation, bootstrapping training data and training with modifications of the likelihood loss can be used.

Statistics and systematics

Statistical approach [Bahl, Elmer, Favaro, Haußmann, TP, Winterhalder]

- expectation value with internal representation θ

$$\langle A \rangle = \int dA\; A\; p(A|x) = \int dA\; A \int d\theta\; p(A|\theta)\; p(\theta|A_\text{train})$$

- training a generalization

$$\int d\theta\; p(A|\theta)\; p(\theta|A_\text{train}) \approx \int d\theta\; p(A|\theta)\; q(\theta)$$

Statistics and systematics

Statistical approach [Bahl, Elmer, Favaro, Haußmann, TP, Winterhalder]

- expectation value with internal representation θ

$$\langle A \rangle = \int dA\; A\; p(A|x) = \int dA\; A \int d\theta\; p(A|\theta)\; p(\theta|A_\text{train})$$

- training a generalization

$$\int d\theta\; p(A|\theta)\; p(\theta|A_\text{train}) \approx \int d\theta\; p(A|\theta)\; q(\theta)$$

- similarity from minimal KL-divergence

$$\begin{aligned} D_\text{KL}[q(\theta), p(\theta|A_\text{train})] &\equiv \int d\theta\; q(\theta)\; \log \frac{q(\theta)}{p(\theta|A_\text{train})} \\ &= \int d\theta\; q(\theta)\; \log \frac{q(\theta) p(A_\text{train})}{p(A_\text{train}|\theta) p(\theta)} \\ &= - \int d\theta\; q(\theta)\; \log p(A_\text{train}|\theta) + \int d\theta\; q(\theta) \log \frac{q(\theta)}{p(\theta)} + \cdots \end{aligned}$$

- regularized likelihood loss

$$\mathcal{L} = - \int d\theta\; q(\theta)\; \log p(A_\text{train}|\theta) + D_\text{KL}[q(\theta), p(\theta)]$$

Statistics and systematics

Statistical approach [Bahl, Elmer, Favaro, Haußmann, TP, Winterhalder]

- expectation value with internal representation θ

$$\langle A \rangle = \int dA \; A \; p(A|x) = \int dA \; A \int d\theta \; p(A|\theta) \; p(\theta|A_\text{train})$$

- training a generalization

$$\int d\theta \; p(A|\theta) \; p(\theta|A_\text{train}) \approx \int d\theta \; p(A|\theta) \; q(\theta)$$

- similarity from minimal KL-divergence

$$\begin{aligned} D_\text{KL}[q(\theta), p(\theta|A_\text{train})] &\equiv \int d\theta \; q(\theta) \; \log \frac{q(\theta)}{p(\theta|A_\text{train})} \\ &= \int d\theta \; q(\theta) \; \log \frac{q(\theta) p(A_\text{train})}{p(A_\text{train}|\theta) p(\theta)} \\ &= - \int d\theta \; q(\theta) \; \log p(A_\text{train}|\theta) + \int d\theta \; q(\theta) \log \frac{q(\theta)}{p(\theta)} + \cdots \end{aligned}$$

- regularized likelihood loss

$$\mathcal{L} = - \int d\theta \; q(\theta) \; \log p(A_\text{train}|\theta) + D_\text{KL}[q(\theta), p(\theta)]$$

$\to$ Variance [Bayesian networks]

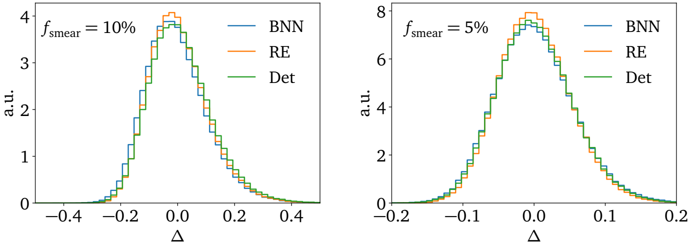
$$\sigma^2 = \int dA \int d\theta \; (A - \langle A \rangle)^2 \; p(A|\theta) \; q(\theta) \equiv \sigma^2_\text{syst} + \sigma^2_\text{stat}$$

Amplitudes with calibrated uncertainties

Loop amplitude $gg \to \gamma\gamma g(g)$ over phase space [Badger, Butter, Luchmann, Pitz, TP]

- systematics: artificial noise
- statistics plateau
- accuracy over phase space

$$\Delta(x) = \frac{A_{\text{NN}}(x) - A_{\text{true}}(x)}{A_{\text{true}}(x)}$$



Amplitudes with calibrated uncertainties

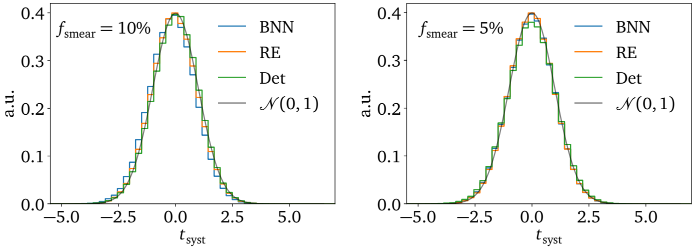
Loop amplitude $gg \to \gamma\gamma g(g)$ over phase space [Badger, Butter, Luchmann, Pitz, TP]

- systematics: artificial noise
- statistics plateau
- accuracy over phase space

$$\Delta(x) = \frac{A_{\text{NN}}(x) - A_{\text{true}}(x)}{A_{\text{true}}(x)}$$

- pull over phase space

$$t_{\text{syst}}(x) = \frac{A_{\text{NN}}(x) - A_{\text{true}}(x)}{\sigma_{\text{syst}}(x)}$$



Amplitudes with calibrated uncertainties

Loop amplitude $gg \to \gamma\gamma g(g)$ over phase space [Badger, Butter, Luchmann, Pitz, TP]

- systematics: artificial noise
- statistics plateau
- accuracy over phase space

$$\Delta(x) = \frac{A_\text{NN}(x) - A_\text{true}(x)}{A_\text{true}(x)}$$

- pull over phase space

$$t_\text{syst}(x) = \frac{A_\text{NN}(x) - A_\text{true}(x)}{\sigma_\text{syst}(x)}$$

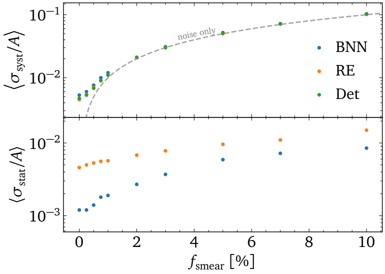
Towards zero noise

- scaling

$$\sigma^2_\text{syst} - \sigma^2_\text{syst,0} \approx \sigma^2_\text{train}$$

- plateau $\langle \sigma_\text{syst}/A \rangle \sim 0.4\%$

→ Limiting factor??



Amplitudes with calibrated uncertainties

Loop amplitude $gg \to \gamma\gamma g(g)$ over phase space [Badger, Butter, Luchmann, Pitz, TP]

- systematics: artificial noise
- statistics plateau
- accuracy over phase space

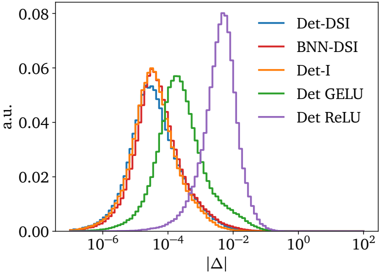
$$\Delta(x) = \frac{A_\text{NN}(x) - A_\text{true}(x)}{A_\text{true}(x)}$$

- pull over phase space

$$t_\text{syst}(x) = \frac{A_\text{NN}(x) - A_\text{true}(x)}{\sigma_\text{syst}(x)}$$

Data pre-processing

- amplitude from invariants
- learn Minkowski metric
- Deep-sets-invariant network
  L-GATr transformer



Amplitudes with calibrated uncertainties

Loop amplitude $gg \to \gamma\gamma g(g)$ over phase space [Badger, Butter, Luchmann, Pitz, TP]

- systematics: artificial noise
- statistics plateau
- accuracy over phase space

$$\Delta(x) = \frac{A_\text{NN}(x) - A_\text{true}(x)}{A_\text{true}(x)}$$

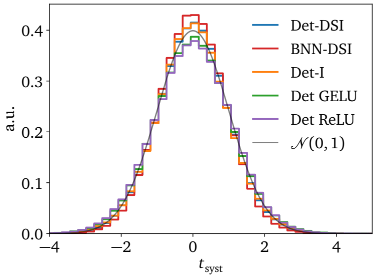
- pull over phase space

$$t_\text{syst}(x) = \frac{A_\text{NN}(x) - A_\text{true}(x)}{\sigma_\text{syst}(x)}$$

Data pre-processing

- amplitude from invariants
- learn Minkowski metric
- Deep-sets-invariant network
  L-GATr transformer

$\to$ Calibrated systematics



ATLAS calibration

Energy calibration with uncertainties [ATLAS + Heimel, TP, Vogel]

- interpretable calorimeter phase space x
- learned calibration function

$$\mathcal{R}_{\text{NN}}(x) \pm \Delta\mathcal{R}_{\text{NN}}(x) \approx \frac{E^{\text{obs}}(x)}{E^{\text{dep}}(x)}$$

- trained on simulations, statistics neglibigle
- systematics: noise in data
  network expressivity
  data representation ...

ATLAS calibration

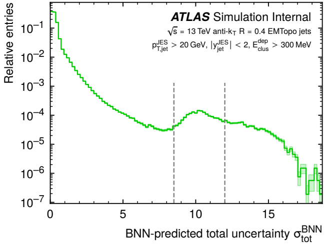
Energy calibration with uncertainties [ATLAS + Heimel, TP, Vogel]

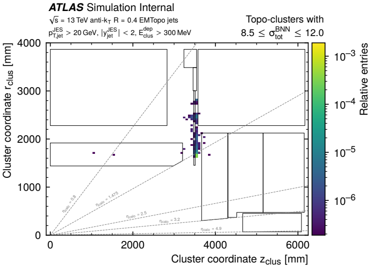
- interpretable calorimeter phase space x
- learned calibration function

$$\mathcal{R}_{\text{NN}}(x) \pm \Delta\mathcal{R}_{\text{NN}}(x) \approx \frac{E^{\text{obs}}(x)}{E^{\text{dep}}(x)}$$

- trained on simulations, statistics neglibigle
- systematics: noise in data
  network expressivity
  data representation ...

→ Understand (simulated) detector

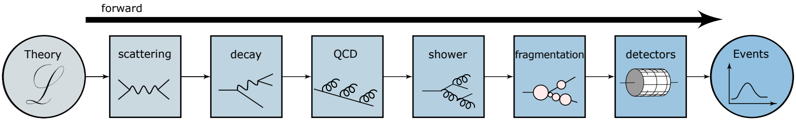




Generative AI

Simulations, MadNIS, calorimeters,...

- learn phase space density
  fast sampling Gaussian → phase space
  Bayesian generative network → uncertainties
- Variational Autoencoder
  → low-dimensional physics
- GAN [Butter, TP, Winterhalder]
  → generator trained by classifier
- Normalizing Flow [Bellagente, Haußmann, Luchmann, TP]
  → bijective mapping
- Diffusion [Butter, Hütsch, Palacios, TP, Sorrenson, Spinner]
  → ODE solving
- JetGPT, ViT [Favaro, Ore, Palacios, TP]
  → non-local structures
- L-GATr for LHC [Brehmer, Breso, de Haan, TP, Qu, Spinner, Thaler]
  → Lorentz-covariant data representation



Controlling generative AI

Compare generated with training data [Das, Favaro, Heimel, Krause, TP, Shi]

- generation: unsupervised density
- classify training vs generated events $D(x)$
  learned density ratio [Neyman-Pearson]

$$w(x_i) = \frac{D(x_i)}{1 - D(x_i)} = \frac{p_{\text{data}}(x_i)}{p_{\text{model}}(x_i)}$$

$\rightarrow$ Test ratio over phase space

Controlling generative AI

Compare generated with training data [Das, Favaro, Heimel, Krause, TP, Shi]

- generation: unsupervised density
- classify training vs generated events $D(x)$
  learned density ratio [Neyman-Pearson]

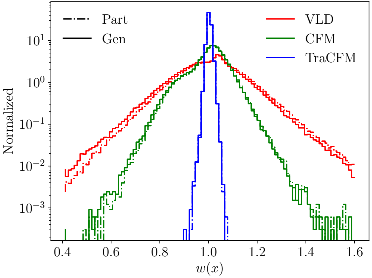
$$w(x_i) = \frac{D(x_i)}{1 - D(x_i)} = \frac{p_{\text{data}}(x_i)}{p_{\text{model}}(x_i)}$$

→ Test ratio over phase space

Testing NN-generators [Heidelberg-Berkeley-Irvine]

- accuracy from width of weight distribution
- tails indicating failure mode

→ Systematic performance test

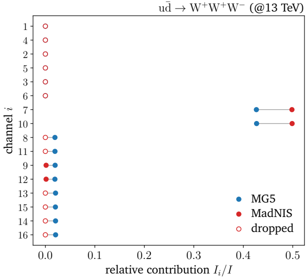


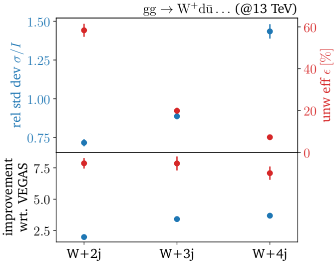
Neural importance sampling

ML-channel weights & ML-Vegas [Heimel, Hütsch, Maltoni, Mattelaer, TP, Winterhalder]

- simple goal 1: learn channel weights [regression]
- simple goal 2: learn Vegas mapping [invertible generatation]
- technically: online + buffered training
- minimize integration variance

→ Beat MadGraph and its team...





Transforming LHC physics

Number of searches

- optimal inference: signal and background simulations
- CPU-limitation for many signals?

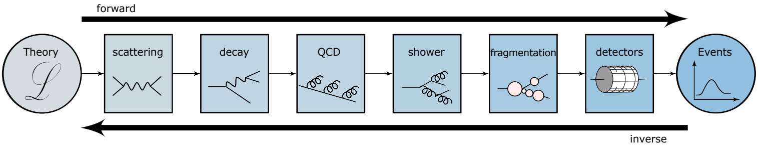
Optimal analyses

- update theory predictions
- include predictions not in event generators

Public LHC data

- common lore:
  LHC data too complicated for amateurs
- in truth:
  hard scattering and decay simulations public
  BSM physics not in hadronization and detector

→ Unfold to suitable level



ML-Unfolding

View as generative inference [Köthe etal, Macke etal]

- four phase space distributions

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

- learn conditional probabilities from $(x_\text{part}, x_\text{reco})$ [forward-inverse symmetric]

→ Unbinned and high-dimensional unfolding

ML-Unfolding

View as generative inference [Köthe etal, Macke etal]

- four phase space distributions

$p_\text{sim}(x_\text{part})$ ← unfolding inference → $p_\text{unfold}(x_\text{part})$

$p(x_\text{reco}|x_\text{part})$ ↓ ↑ $p(x_\text{part}|x_\text{reco})$

$p_\text{sim}(x_\text{reco})$ ← forward inference → $p_\text{data}(x_\text{reco})$

- learn conditional probabilities from $(x_\text{part}, x_\text{reco})$ [forward-inverse symmetric]

$\to$ Unbinned and high-dimensional unfolding

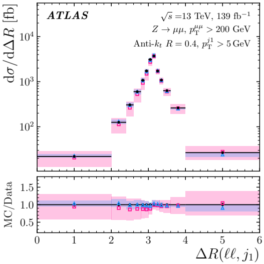
OmniFold

- learn $p_\text{sim}(x_\text{reco}) \leftrightarrow p_\text{data}(x_\text{reco})$
- reweight $p_\text{sim}(x_\text{part}) \to p_\text{unfold}(x_\text{part})$

$p_\text{sim}(x_\text{part})$ — classifier weights → $p_\text{unfold}(x_\text{part})$

pull/push weights ↕

$p_\text{sim}(x_\text{reco})$ ← classifier weights → $p_\text{data}(x_\text{reco})$



$\to$ Z+jets in 24D [ATLAS]

Unfolding top decays

Top mass as high school project [Favaro, Palacios, TP + CMS]

- first measure m_t in unfolded data
  then unfold full kinematics
- simulation m_s vs data m_d [too bad to reweight]

$$p_\text{sim}(x_\text{part}|m_s) \qquad\qquad p_\text{unfold}(x_\text{part}|m_s, m_d)$$

$$p(x_\text{reco}|x_\text{part}) \downarrow \qquad\qquad \uparrow p_\text{model}(x_\text{part}|x_\text{reco}, m_s)$$

$$p_\text{sim}(x_\text{reco}|m_s) \xleftrightarrow{\text{correspondence}} p_\text{data}(x_\text{reco}|m_d)$$

$\rightarrow$ train on m_s-range
include batch-wise $M_{jjj} \in x_\text{reco}$

Unfolding top decays

Top mass as high school project [Favaro, Palacios, TP + CMS]

- first measure m_t in unfolded data
  then unfold full kinematics
- simulation m_s vs data m_d [too bad to reweight]

$$p_{\text{sim}}(x_{\text{part}}|m_s) \xrightarrow{\quad} \quad p_{\text{unfold}}(x_{\text{part}}|m_s, m_d)$$

$$p(x_{\text{reco}}|x_{\text{part}}) \downarrow \qquad \uparrow p_{\text{model}}(x_{\text{part}}|x_{\text{reco}}, m_s)$$

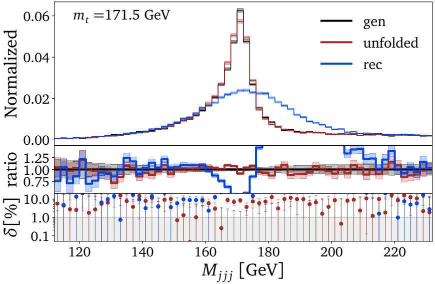
$$p_{\text{sim}}(x_{\text{reco}}|m_s) \xleftrightarrow{\text{correspondence}} p_{\text{data}}(x_{\text{reco}}|m_d)$$

→ train on m_s-range
include batch-wise $M_{jjj} \in x_{\text{reco}}$

Preliminary unfolding results [TraCFM]

- 4D for m_t
  uncluding m_W-calibration
- 12D published data

→ CMS data next



Unfolding top decays

Top mass as high school project [Favaro, Palacios, TP + CMS]

- first measure m_t in unfolded data
  then unfold full kinematics
- simulation m_s vs data m_d [too bad to reweight]

$$p_\text{sim}(x_\text{part}|m_s) \xrightarrow{\text{unfold}} p_\text{unfold}(x_\text{part}|m_s, m_d)$$

$$p_\text{sim}(x_\text{part}|m_s) \xrightarrow{p(x_\text{reco}|x_\text{part})} p_\text{sim}(x_\text{reco}|m_s)$$

$$p_\text{sim}(x_\text{reco}|m_s) \xleftrightarrow{\text{correspondence}} p_\text{data}(x_\text{reco}|m_d)$$

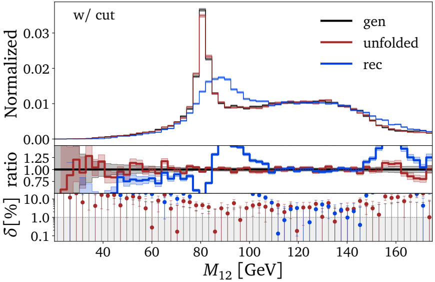
$$p_\text{data}(x_\text{reco}|m_d) \xrightarrow{p_\text{model}(x_\text{part}|x_\text{reco}, m_s)} p_\text{unfold}(x_\text{part}|m_s, m_d)$$

→ train on m_s-range
include batch-wise $M_{jjj} \in x_\text{reco}$

Preliminary unfolding results [TraCFM]

- 4D for m_t
  uncluding m_W-calibration
- 12D published data

→ CMS data next



Unfolding top decays

Top mass as high school project [Favaro, Palacios, TP + CMS]

- first measure m_t in unfolded data
  then unfold full kinematics
- simulation m_s vs data m_d [too bad to reweight]

$$p_\text{sim}(x_\text{part}|m_s) \xrightarrow{} \quad p_\text{unfold}(x_\text{part}|m_s, m_d)$$

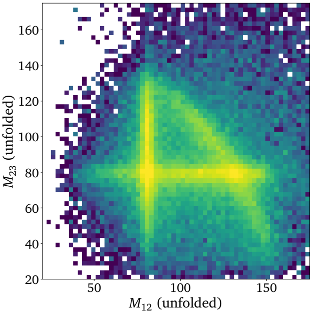
$$p_\text{sim}(x_\text{part}|m_s) \;\overset{p(x_\text{reco}|x_\text{part})}{\longrightarrow}\; p_\text{sim}(x_\text{reco}|m_s) \;\overset{\text{correspondence}}{\longleftrightarrow}\; p_\text{data}(x_\text{reco}|m_d) \;\overset{p_\text{model}(x_\text{part}|x_\text{reco}, m_s)}{\longrightarrow}\; p_\text{unfold}(x_\text{part}|m_s, m_d)$$

→ train on m_s-range
include batch-wise $M_{jjj} \in x_\text{reco}$

Preliminary unfolding results [TraCFM]

- 4D for m_t
  uncluding m_W-calibration
- 12D published data

→ CMS data next



ML for LHC Theory

ML is particle physics method development

1 just another numerical tool for a numerical field

2 completely transformative new language

- driven by (money from) data science and medical research
- particle physics should be leading scientific AI
- 10000 Einsteins...

...improving established tools
...developing new tools for established tasks
...transforming through new ideas

$\rightarrow$ Complexity becoming our friend

Modern Machine Learning for LHC Physicists

Tilman Plehn[a*], Anja Butter[a,b], Barry Dillon[a],
Theo Heimel[a], Claudius Krause[c], and Ramon Winterhalder[d]

[a] Institut für Theoretische Physik, Universität Heidelberg, Germany
[b] LPNHE, Sorbonne Université, Université Paris Cité, CNRS/IN2P3, Paris, France
[c] HEPHY, Austrian Academy of Sciences. Vienna, Austria
[d] CP3, Université catholique de Louvain, Louvain-la-Neuve, Belgium

March 19, 2024

Abstract

Modern machine learning is transforming particle physics fast, bullying its way into our numerical tool box. For young researchers it is crucial to stay on top of this development, which means applying cutting-edge methods and tools to the full range of LHC physics problems. These lecture notes lead students with basic knowledge of particle physics and significant enthusiasm for machine learning to relevant applications. They start with an LHC-specific motivation and a non-standard introduction to neural networks and then cover classification, unsupervised classification, generative networks, and inverse problems. Two themes defining much of the discussion are well-defined loss functions and uncertainty-aware networks. As part of the applications, the notes include some aspects of theoretical LHC physics. All examples are chosen from particle physics publications of the last few years.[1]