Neural Networks with Calibrated Learned Uncertainties

Tilman Plehn

Universität Heidelberg

Imperial College, February 2025

Shortest ML-intro ever

Fit-like approximation [2211.01421]

- approximate $f_\theta(x) \approx f(x)$
- no parametrization, but many θ
- new representation/latent space θ

Construction and contol

- define loss function
- minimize loss to find best θ
- compare $x \to f_\theta(x)$ for training vs test data

LHC applications

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

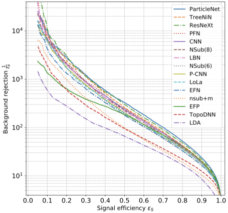
$\to$ Transforming numerical science

ML in experiment

Top tagging [classification, 2016-today]

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

→ Driving NN-architectures

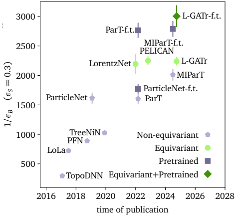


SciPost Physics Submission

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



Uncertainties

Tilman Plehn

ML for LHC

Uncertainties

Bayesian NNs

Ensembles

Amplitudes

Generation

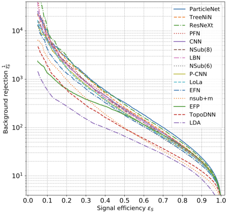
Amplification

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

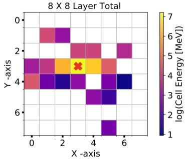
G. Kasieczka (ed)[1], T. Plehn (ed)[2], A. Butter[2], K. Cranmer[3], D. Debnath[4], B. M. Dillon[5], M. Fairbairn[6], D. A. Faroughy[5], W. Fedorko[7], C. Gay[7], L. Gouskos[8], J. F. Kamenik[5,9], P. T. Komiske[10], S. Leiss[1], A. Lister[7], S. Macaluso[3,4], E. M. Metodiev[10], L. Moore[11], B. Nachman,[12,13], 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]

- mother of jet analyses
- combining detectors with different resolution
- optimality the key

→ Modern jet analysis basics



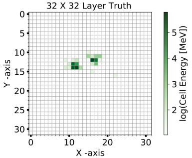
Towards a Computer Vision Particle Flow ⋆

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

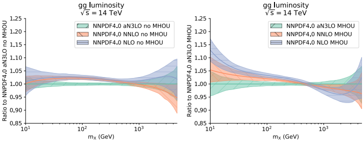


ML in phenomenology

Parton densities [NNPDF, 2002-today]

- pdfs without functional bias and full uncertainties
- precision and calibrated uncertainties

→ Drivers of ML-theory



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*

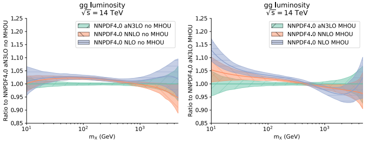
This paper is dedicated to the memory of Stefano Catani, Grand Master of QCD, great scientist and human being

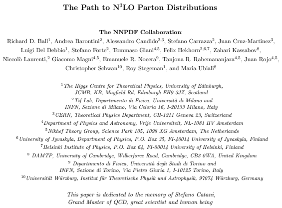
ML in phenomenology

Parton densities [NNPDF, 2002-today]

- pdfs without functional bias and full uncertainties
- precision and calibrated uncertainties

→ Drivers of ML-theory





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

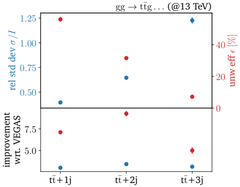
This paper is dedicated to the memory of Stefano Catani, Grand Master of QCD, great scientist and human being

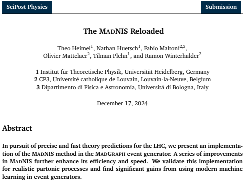
Ultra-fast event generators [Sherpa, MadNIS, MLHad]

- event generation modular
- improve and replace by ML-modules

→ Beat state of the art

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

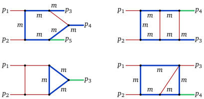
In pursuit of precise and fast theory predictions for the LHC, we present an implementation of the MadNIS method in the MadGraph event generator. A series of improvements in MadNIS further enhance its efficiency and speed. We validate this implementation for realistic partonic processes and find significant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

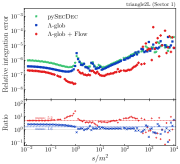
ML in theory

Optimizing integration paths [invertible networks]

- find optimal integration paths
- learn variable transformation

$\rightarrow$ Theory-integrator





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

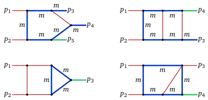
Numerical evaluations of Feynman integrals often proceed via a deformation of the integration contour into the complex plane. While valid contours are easy to construct, the numerical precision for a multi-loop integral can depend critically on the chosen contour. We present methods to optimize this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

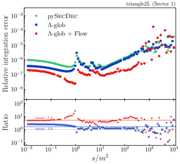
ML in theory

Optimizing integration paths [invertible networks]

- find optimal integration paths
- learn variable transformatio

→ Theory-integrator






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.

Navigating string landscape [reinforcement learning]

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

→ Model space sampling

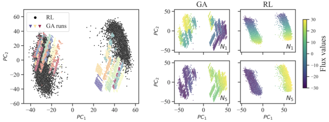
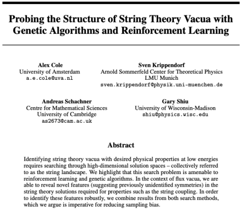


Figure 1: *Left:* Cluster structure in dimensionally reduced flux samples for RL and 25 GA runs (PCA on all samples of GA and RL). The colors indicate individual GA runs. *Right:* Dependence on flux (input) values (N_3 and N_5 respectively) in relation to principal components for a PCA fit of the individual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

Learned uncertainties

Network training as a fit

- learn scalar field $f_\theta(x) \approx f(x)$
- statistics: maximize parameter probability given (f_j, σ_j)

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

$\to$ maximize tractable likelihood instead

$$p(x|\theta) = \prod_j \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left(- \frac{|f_j - f_\theta(x_j)|^2}{2\sigma_j^2} \right)$$

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

Learned uncertainties

Network training as a fit

- learn scalar field $\quad f_\theta(x) \approx f(x)$
- statistics: maximize parameter probability given (f_j, σ_j)

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

→ maximize tractable likelihood instead

$$p(x|\theta) = \prod_j \frac{1}{\sqrt{2\pi}\sigma_j} \exp\left(- \frac{|f_j - f_\theta(x_j)|^2}{2\sigma_j^2} \right)$$

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

Learned local uncertainty

- Gaussian log-likelihood with normalization

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

- if needed replace $\sigma_\theta(x)$ by mixture model

→ learn $f_\theta(x)$ and $\sigma_\theta(x)$ together

Bayesian networks

Learned function statistically

- amplitude over phase phase

$$\langle A \rangle = \int dA\, A\, p(A)$$

- internal representation θ of training data T [think Gaussian with mean and width]

$$p(A) = \int d\theta\; p(A|\theta)\; p(\theta|T)$$

$\rightarrow$ θ-distribution defining Bayesian NN

Bayesian networks

Learned function statistically

- amplitude over phase phase

$$\langle A \rangle = \int dA \; A \; p(A)$$

- internal representation θ of training data T [think Gaussian with mean and width]

$$p(A) = \int d\theta \; p(A|\theta) \; p(\theta|T)$$

$\to$ θ-distribution defining Bayesian NN

Variational approximation

- definition of training

$$p(A) = \int d\theta \; p(A|\theta) \; p(\theta|T) \approx \int d\theta \; p(A|\theta) \; q(\theta)$$

- similarity through minimal KL-divergence [Bayes' theorem to remove unknown posterior]

$$\begin{aligned} D_{\mathrm{KL}}[q(\theta), p(\theta|T)] &= \int d\theta \; q(\theta) \; \log \frac{q(\theta)}{p(\theta|T)} \\ &= \int d\theta \; q(\theta) \; \log \frac{q(\theta)p(T)}{p(T|\theta)p(\theta)} \\ &\approx D_{\mathrm{KL}}[q(\theta), p(\theta)] - \int d\theta \; q(\theta) \; \log p(T|\theta) \equiv \mathcal{L} \end{aligned}$$

$\to$ Two-term loss: likelihood + prior

Statistics vs systematics

Statistical network evaluation

- expectation value from network $q(\theta)$

$$\langle A \rangle = \int dA d\theta \; A \; p(A|\theta) \; q(\theta)$$
$$\equiv \int d\theta \; q(\theta) \overline{A}(\theta) \qquad \text{with} \qquad \overline{A}(\theta) = \int dA \; A \; p(A|\theta)$$

- corresponding variance

$$\sigma_\text{tot}^2 = \int dA d\theta \; \left(A - \langle A \rangle\right)^2 \; p(A|\theta) \; q(\theta)$$
$$= \int d\theta \; q(\theta) \left[\overline{A^2}(\theta) - \overline{A}(\theta)^2 + \left(\overline{A}(\theta) - \langle A \rangle \right)^2 \right] \equiv \sigma_\text{syst}^2 + \sigma_\text{stat}^2$$

Two uncertainties

- statistical — vanishing for perfect training: $q(\theta) \to \delta(\theta - \theta_0)$

$$\sigma_\text{stat}^2 = \int d\theta \; q(\theta) \left[\overline{A}(\theta) - \langle A \rangle \right]^2$$

- systematic — vanishing for perfect data: $p(A|\theta) \to \delta(A - A_0)$

$$\sigma_\text{syst}^2 = \int d\theta \; q(\theta) \left[\overline{A^2}(\theta) - \overline{A}(\theta)^2 \right]$$

$\to$ Systematics dominant for LHC

Repulsive ensembles

Posterior from network ensemble

- OED vs continuity equation

$$\frac{d\theta}{dt} = v(\theta, t) \qquad \Leftrightarrow \qquad \frac{\partial \rho(\theta, t)}{\partial t} = -\nabla_\theta \left[v(\theta, t) \rho(\theta, t) \right]$$

- Fokker-Planck equation with stationary $\rho(\theta, t) = \pi(\theta)$

$$\frac{d\theta}{dt} = -\nabla_\theta \log \frac{\rho(\theta, t)}{\pi(\theta)}$$

- ODE describing training progress

$$\begin{aligned} \theta^{t+1} - \theta^t &\propto -\nabla_{\theta^t} \left[\log \rho(\theta^t) - \log \pi(\theta^t) \right] \\ &= -\nabla_{\theta^t} \left[\log \sum_j k(\theta^t, \theta_j^t) - \log p(\theta | x_{\text{train}}^t) \right] \equiv -\nabla_{\theta^t} \mathcal{L}_{\text{RE}} \end{aligned}$$

$\rightarrow$ Joint ensemble training

Repulsive ensembles

Posterior from network ensemble

- OED vs continuity equation

$$\frac{d\theta}{dt} = v(\theta, t) \qquad \Leftrightarrow \qquad \frac{\partial \rho(\theta, t)}{\partial t} = -\nabla_\theta \left[v(\theta, t) \rho(\theta, t) \right]$$

- Fokker-Planck equation with stationary $\rho(\theta, t) = \pi(\theta)$

$$\frac{d\theta}{dt} = -\nabla_\theta \log \frac{\rho(\theta, t)}{\pi(\theta)}$$

- ODE describing training progress

$$\begin{aligned} \theta^{t+1} - \theta^t \propto -\nabla_{\theta^t} &\left[\log \rho(\theta^t) - \log \pi(\theta^t) \right] \\ = -\nabla_{\theta^t} &\left[\log \sum_j k(\theta^t, \theta_j^t) - \log p(\theta | x_{\text{train}}^t) \right] \equiv -\nabla_{\theta^t} \mathcal{L}_{\text{RE}} \end{aligned}$$

$\to$ Joint ensemble training

Repulsive ensembles

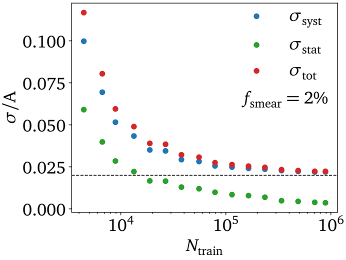
- train network ensemble
- apply repulsive force
  kernel in function space

$\to$ Alternative for statistical uncertainty

Network amplitudes

Loop amplitude $gg \to \gamma\gamma g(g)$ [Bahl, Elmer, Favaro, Haussmann, TP, Winterhalder]

- regression of exact scalar over phase space [Aylett-Bullock, Badger, Moodie]
- example systematics: artificial noise
- assume $\sigma_\text{stat} \ll \sigma_\text{syst}$

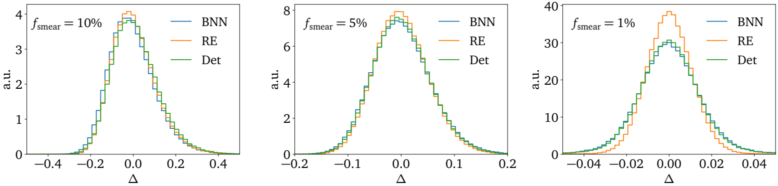


Network amplitudes

Loop amplitude $gg \to \gamma\gamma g(g)$ [Bahl, Elmer, Favaro, Haussmann, TP, Winterhalder]

- regression of exact scalar over phase space [Aylett-Bullock, Badger, Moodie]
- example systematics: artificial noise
- assume $\sigma_{\text{stat}} \ll \sigma_{\text{syst}}$
- accuracy over phase space

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



Network amplitudes

Loop amplitude $gg \to \gamma\gamma g(g)$ [Bahl, Elmer, Favaro, Haussmann, TP, Winterhalder]

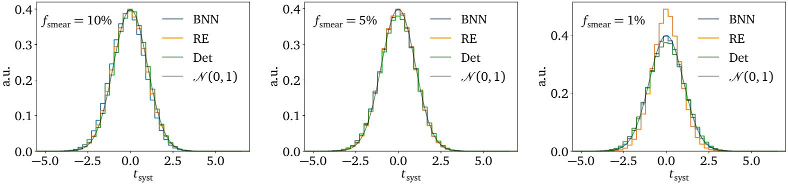
- regression of exact scalar over phase space [Aylett-Bullock, Badger, Moodie]
- example systematics: artificial noise
- assume $\sigma_{\text{stat}} \ll \sigma_{\text{syst}}$
- accuracy over phase space

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

- pull over phase space

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

$\to$ calibrated leading systematics



Network amplitudes

Loop amplitude $gg \to \gamma\gamma g(g)$ [Bahl, Elmer, Favaro, Haussmann, TP, Winterhalder]

- regression of exact scalar over phase space [Aylett-Bullock, Badger, Moodie]
- example systematics: artificial noise
- assume $\sigma_\text{stat} \ll \sigma_\text{syst}$
- accuracy over phase space

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

- pull over phase space

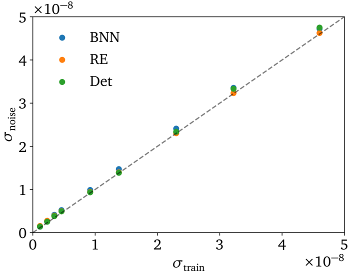
$$t(x) = \frac{A_\text{NN}(x) - A_\text{true}(x)}{\sigma(x)}$$

$\to$ calibrated leading systematics

Towards zero noise

- extrapolate to zero noise

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



Network amplitudes

Loop amplitude $gg \to \gamma\gamma g(g)$ [Bahl, Elmer, Favaro, Haussmann, TP, Winterhalder]

- regression of exact scalar over phase space [Aylett-Bullock, Badger, Moodie]
- example systematics: artificial noise
- assume $\sigma_\text{stat} \ll \sigma_\text{syst}$
- accuracy over phase space

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

- pull over phase space

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

$\to$ calibrated leading systematics

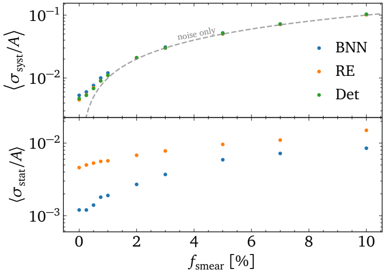
Towards zero noise

- extrapolate to zero noise

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

- systematics plateau $\langle \sigma/A \rangle \sim 0.4\%$

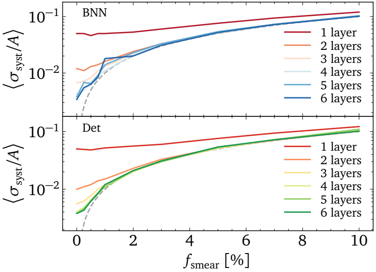
$\to$ Limiting factor??



Improved accuracy

Network expressivity

- large range of amplitude values
- resolution of (collinear) peaks
- network breaks for large amplitudes
- 3 hidden layers needed
- activation function
  machine precision...



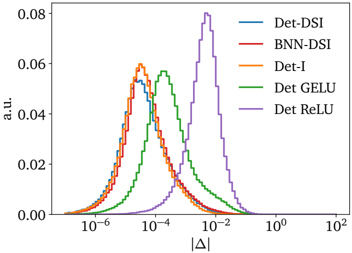
Improved accuracy

Network expressivity

- large range of amplitude values
- resolution of (collinear) peaks
- network breaks for large amplitudes
- 3 hidden layers needed
- activation function
  machine precision...

Data pre-processing

- amplitude from invariants
- learn Minkowski metric?
- Deep-sets-invariant network [Heinrich etal]
  L-GATr transformer



Improved accuracy

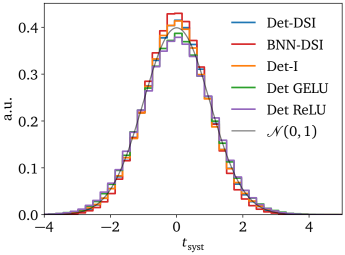
Network expressivity

- large range of amplitude values
- resolution of (collinear) peaks
- network breaks for large amplitudes
- 3 hidden layers needed
- activation function
  machine precision...

Data pre-processing

- amplitude from invariants
- learn Minkowski metric?
- Deep-sets-invariant network [Heinrich etal]
  L-GATr transformer
- uncertainty scaling with accuracy
  pull unit Gaussian

→ Calibrated leading systematics



Improved accuracy

Network expressivity

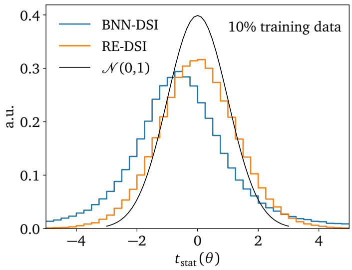
- large range of amplitude values
- resolution of (collinear) peaks
- network breaks for large amplitudes
- 3 hidden layers needed
- activation function
  machine precision...

Statistical uncertainties

- well-defined for $\sigma_\text{stat} \ll \sigma_\text{syst}$
  calibration of $\sigma_\text{tot} \sim \sigma_\text{syst}$
- systematics per network
  statistics sampled
- calibration from pull

$$t_\text{stat}(x,\theta) = \frac{\overline{A}(x,\theta) - \langle A \rangle(x)}{\sigma_\text{stat}(x)} \approx \frac{\overline{A}(x,\theta) - A_\text{true}(x)}{\sigma_\text{stat}(x)}$$

$\to$ Work to do...



ATLAS calibration

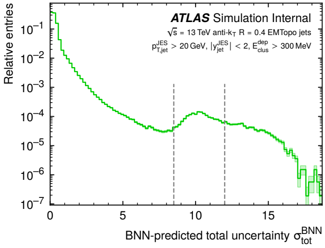
Energy calibration with uncertainties [ATLAS + Heidelberg, 2412.04370]

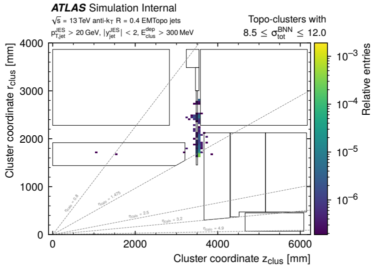
- interpretable calorimeter phase space x
- learned calibration function

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

- systematics: noisy training data ...

$\to$ Understand (simulated) detector





Generative AI with uncertainties

Bayesian generative networks [Bellagente, Haussmann, Luchmann, TP]

- network weight distributions for density
- sampling phase space
  events with error bars on weights
- learned density & uncertainty
  reflecting network learning?

→ Generative networks like fitted densities

Generative AI with uncertainties

Bayesian generative networks [Bellagente, Haussmann, Luchmann, TP]

- network weight distributions for density
- sampling phase space
  events with error bars on weights
- learned density & uncertainty
  reflecting network learning?

→ Generative networks like fitted densities

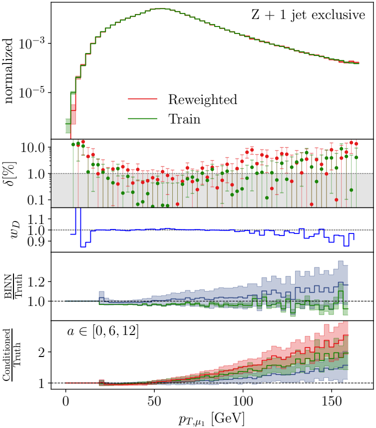
Z+jets events [Heimel, Vent...]

- per-cent accuracy on density
- statistical uncertainty from BNN
- systematics in training data

$$w = 1 + a \left(\frac{p_{T,j_1} - 15\ \text{GeV}}{100\ \text{GeV}} \right)^2$$

sampling a conditionally

→ Precision and uncertainty control



Controlling generative AI

Compare generated with training data

- regression accuracy $\Delta = (A_\text{data} - A_\theta)/A_\text{data}$
- harder for 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

- regression accuracy $\Delta = (A_\text{data} - A_\theta)/A_\text{data}$
- harder for generation, unsupervised density
  classify training vs generated events $D(x)$
  learned density ratio [Neyman-Pearson]

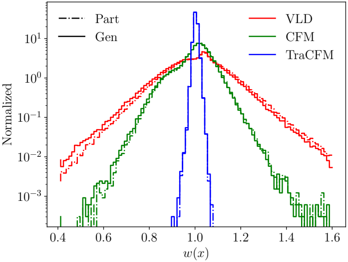
$$w(x_i) = \frac{D(x_i)}{1 - D(x_i)} = \frac{p_\text{data}(x_i)}{p_\text{model}(x_i)}$$

→ Test ratio over phase space

Progress in NN-unfolding

- generative network example
- compare different architectures
- accuracy from width of weight distribution
- tails indicating failure mode

→ Systematic performance test

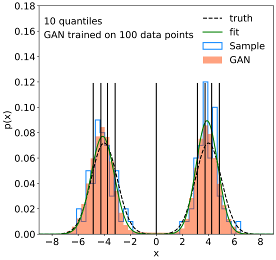


Amplification (for Louis)

Improving training data [Butter, Diefenbacher, Kasieczka, Nachman , TP]

– true function known
  compare GAN vs sampling vs fit

– χ^2-sum of quantiles

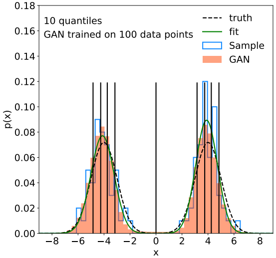


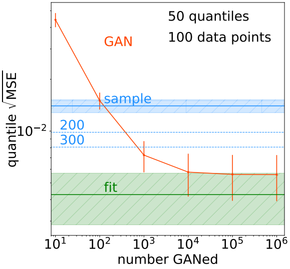
Amplification (for Louis)

Improving training data [Butter, Diefenbacher, Kasieczka, Nachman , TP]

- true function known
  compare GAN vs sampling vs fit
- χ^2-sum of quantiles
- training and fit with 100 data points
- fit like 500-1000 sampled points
  GAN like 500 sampled points [amplifictation factor 5]
  requiring 10,000 GANned events

⇒ Generative networks like fits





Modern ML for LHC

Developing ML for the best science

- just another numerical tool for a numerical field
- transformative new common language
- driven by money from data science and medical research
- be 10000 Einsteins,

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

→ It's the future, let's not miss it

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]

Generative Uncertainties

Unsupervised Bayesian networks

- data: event sample [points in 2D space]

  learn phase space density
  standard distribution in latent space [Gaussian]
  sample from latent space
- Bayesian version

  allow weight distributions
  learn uncertainty map
- 2D wedge ramp

$$p(x) = ax + b = ax + \frac{1 - \frac{a}{2}(x_{\text{max}}^2 - x_{\text{min}}^2)}{x_{\text{max}} - x_{\text{min}}}$$

$$(\Delta p)^2 = \left(x - \frac{1}{2}\right)^2 (\Delta a)^2 + \left(1 + \frac{a}{2}\right)^2 (\Delta x_{\text{max}})^2 + \left(1 - \frac{a}{2}\right)^2 (\Delta x_{\text{min}})^2$$

explaining minimum in $\sigma(x)$

$\rightarrow$ INNs, diffusion just (non-parameterized) fits