Machine Learning per la Fisica Applicata e la Fisica delle Alte Energie

Lezione 19: PDF determination as Machine Learning

Emanuele R. Nocera

12 dicembre 2022

1. A PDF Primer

Factorisation of physical observables

1. Sufficiently inclusive scattering processes allow for a factorised description

short-distance part		long-distance part
hard interaction of partons	$\xleftrightarrow[\text{scheme \& scale } \mu]{\text{factorization}}$	nucleon structure
process-dependent kernels		universal parton distributions

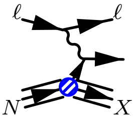
2. Physical observables are written as a convolution of coefficient functions and PDFs

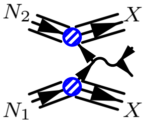
$$F_I(y,\mu^2) = \sum_{f=q,\bar{q},g} C_{If}(y,\alpha_s(\mu^2)) \otimes f(y,\mu^2)$$

$$\sigma(\mu^2,\tau,\mathbf{k}) = \sum_{ij} \int_\tau^1 \frac{dz}{z} \mathcal{L}_{ij}(z,\mu^2)\hat{\sigma}_{ij}\left(\frac{\tau}{z},\alpha_s(\mu^2),\mathbf{k}\right)$$

$$\mathcal{L}_{ij}(z,\mu^2) = (f_i^{h_1} \otimes f_j^{h_2})(z,\mu^2)$$

$$f \otimes g = \int_x^1 \frac{dy}{y} f\left(\frac{x}{y}\right) g(y)$$





3. The C_{If} and $\hat{\sigma}_{ij}$ allow for a perturbative expansion

$$C_{If}(y,\alpha_s) = \sum_{k=0} a_s^k C_{If}^{(k)}(y) \qquad \hat{\sigma}_{ij}(y,\alpha_s) = \sum_{k=0} a_s^k \hat{\sigma}_{ij}^{(k)}(y) \qquad a_s = \alpha_s/(4\pi)$$

4. After factorisation, all quantities (including PDFs) depend on μ

Evolution of PDFs: DGLAP equations

1. A set of $(2n_f+1)$ integro-differential equations, n_f is the number of active flavors

$$\frac{\partial}{\partial \ln \mu^2} f_i(x,\mu^2) = \sum_j^{n_f} \int_x^1 \frac{dz}{z} P_{ji}\left(z, \alpha_s(\mu^2)\right) f_j\left(\frac{x}{z}, \mu^2\right)$$

2. Often written in a convenient basis of PDFs

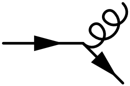
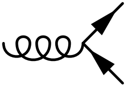
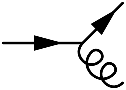
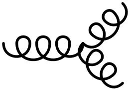
$$q_{\mathrm{NS};\pm} = (q_i \pm \bar{q}_i) - (q_j \pm \bar{q}_j) \qquad q_{\mathrm{NS;v}} = \sum_i^{n_f} (q_i - \bar{q}_j) \qquad \Sigma = \sum_i^{n_f} (q_i + \bar{q}_j)$$

$$\frac{\partial}{\partial \ln \mu^2} q_{\mathrm{NS};\pm,v}(x,\mu^2) = P^{\pm,\mathrm{v}}(x,\mu_F^2) \otimes q_{\mathrm{NS};\pm,v}(x,\mu^2)$$

$$\frac{\partial}{\partial \ln \mu^2} \begin{pmatrix} \Sigma(x,\mu^2) \\ g(x,\mu^2) \end{pmatrix} = \begin{pmatrix} P^{qq} & P^{gq} \\ P^{qg} & P^{gg} \end{pmatrix} \otimes \begin{pmatrix} \Sigma(x,\mu^2) \\ g(x,\mu^2) \end{pmatrix}$$

3. With perturbative computable splitting functions

$$P_{ji}(z,\alpha_s) = \sum_{k=0} a_s^{k+1} P_{ji}^{(k)}(z), \qquad a_s = \alpha_s/(4\pi)$$

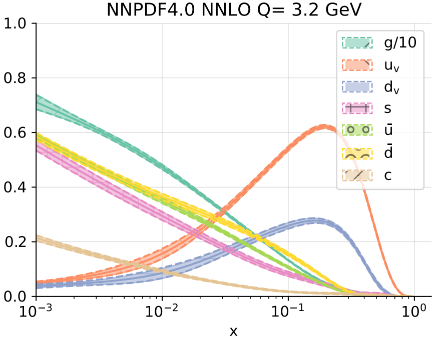
$P_{qq}^{(0)}$  $P_{gq}^{(0)}$  $P_{qg}^{(0)}$  $P_{gg}^{(0)}$ 

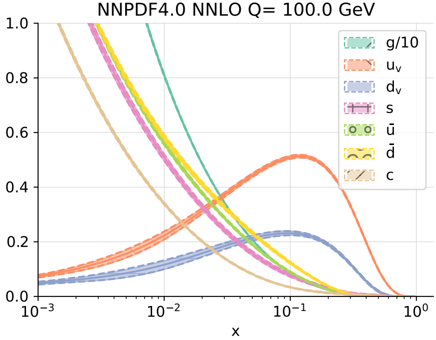
A modern PDF determination: NNPDF4.0

PDFs express the likelihood of a quark or gluon (partons) to enter a collision

That is, PDFs are momentum fraction distributions for each parton

PDFs are a set of probability distributions of probability distributions





[Plot from the 2022 PDG Review of Particle Physics]

Making predictions with PDFs

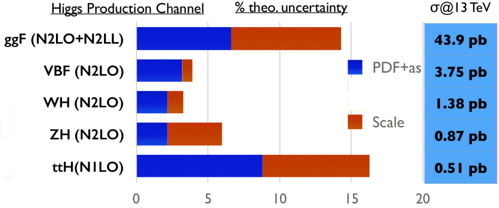
PDF uncertainty is often the dominant source of uncertainty in LHC cross sections

Higgs boson characterisation

Determination of SM parameters, such as the mass of the W boson

Searches for beyond SM physics at large invariant mass of the final state

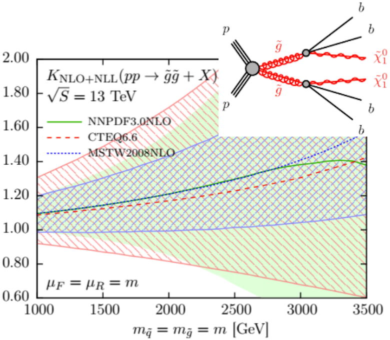
Precision



Channel	$m_{W^+} - m_{W^-}$ [MeV]	Stat. Unc.	Muon Unc.	Elec. Unc.	Recoil Unc.	Bckg. Unc.	QCD Unc.	EW Unc.	PDF Unc.	Total Unc.
$W \to e\nu$	-29.7	17.5	0.0	4.9	0.9	5.4	0.5	0.0	24.1	30.7
$W \to \mu\nu$	-28.6	16.3	11.7	0.0	1.1	5.0	0.4	0.0	26.0	33.2
Combined	-29.2	12.8	3.3	4.1	1.0	4.5	0.4	0.0	23.9	28.0

[Plot from the CERN Yellow Report 2016]

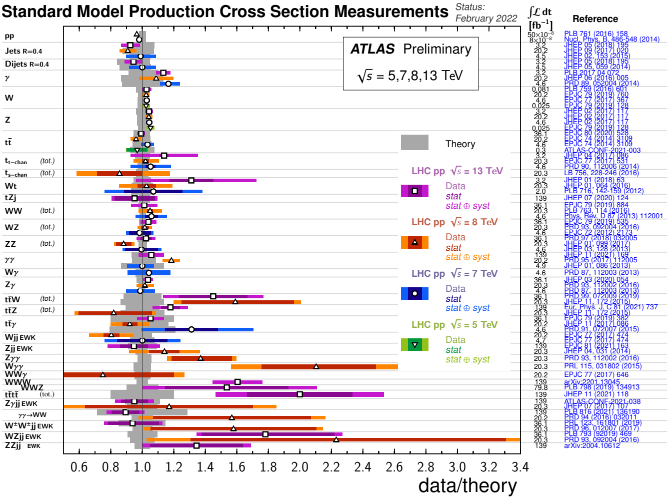
Discovery



[EPJC 76 (2016) 53]

Making predictions with PDFs

PDF uncertainty is often the dominant source of uncertainty in LHC cross sections



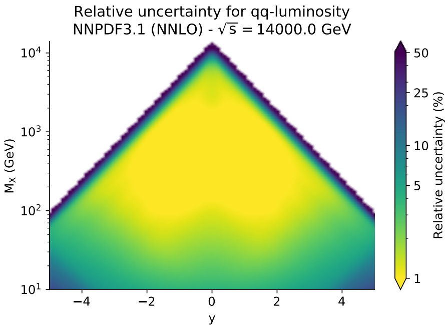
[Plot from ATLAS Collaboration web page]

So, how large are PDF uncertainties?

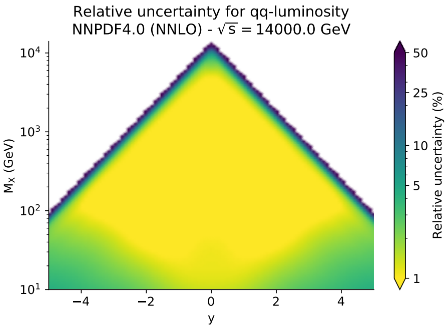
$$\mathcal{L}_{ij}(M_X, y, \sqrt{s}) = \frac{1}{s} f_i\left(\frac{M_X e^{y}}{\sqrt{s}}, M_X\right) f_j\left(\frac{M_X e^{-y}}{\sqrt{s}}, M_X\right)$$

SINGLET

NNPDF3.1 (NNLO)



NNPDF4.0 (NNLO)



Steady progress towards 1% relative uncertainties on $\mathcal{L}_{ij}$ in a broad kinematic range

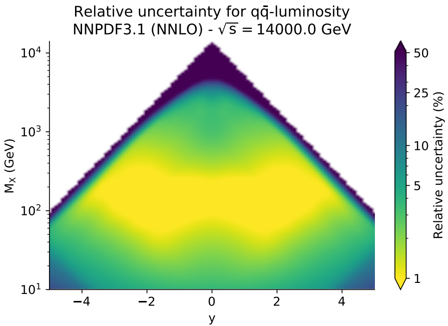
How are we getting there?

So, how large are PDF uncertainties?

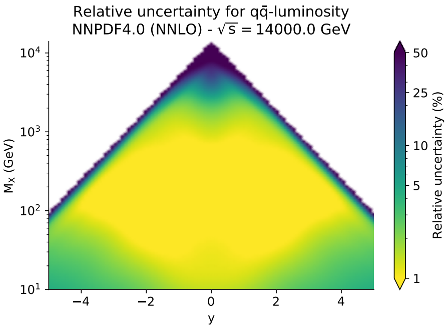
$$\mathcal{L}_{ij}(M_X, y, \sqrt{s}) = \frac{1}{s} f_i\left(\frac{M_X e^y}{\sqrt{s}}, M_X\right) f_j\left(\frac{M_X e^{-y}}{\sqrt{s}}, M_X\right)$$

SINGLET

NNPDF3.1 (NNLO)



NNPDF4.0 (NNLO)



Steady progress towards 1% relative uncertainties on $\mathcal{L}_{ij}$ in a broad kinematic range

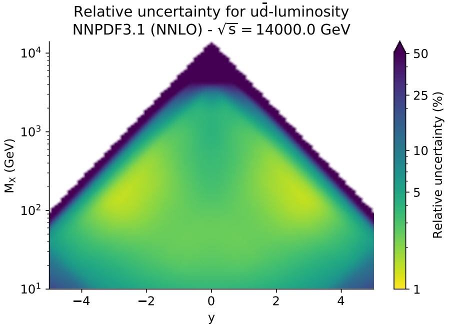
How are we getting there?

So, how large are PDF uncertainties?

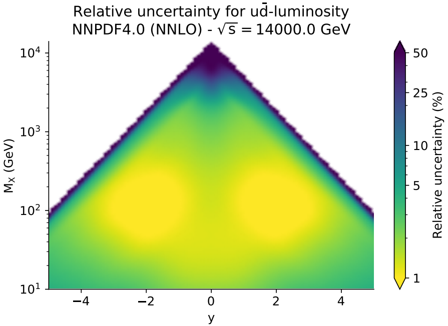
$$\mathcal{L}_{ij}(M_X, y, \sqrt{s}) = \frac{1}{s} f_i\left(\frac{M_X e^{y}}{\sqrt{s}}, M_X\right) f_j\left(\frac{M_X e^{-y}}{\sqrt{s}}, M_X\right)$$

<u>FLAVOURS</u>

NNPDF3.1 (NNLO)



NNPDF4.0 (NNLO)



Steady progress towards 1% relative uncertainties on $\mathcal{L}_{ij}$ in a broad kinematic range

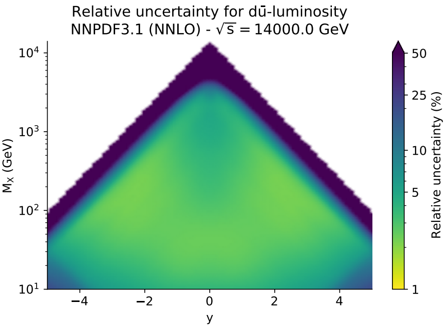
How are we getting there?

So, how large are PDF uncertainties?

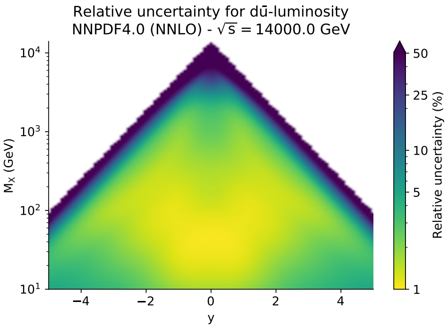
$$\mathcal{L}_{ij}(M_X, y, \sqrt{s}) = \frac{1}{s} f_i\left(\frac{M_X e^{y}}{\sqrt{s}}, M_X\right) f_j\left(\frac{M_X e^{-y}}{\sqrt{s}}, M_X\right)$$

FLAVOURS

NNPDF3.1 (NNLO)



NNPDF4.0 (NNLO)



Steady progress towards 1% relative uncertainties on $\mathcal{L}_{ij}$ in a broad kinematic range

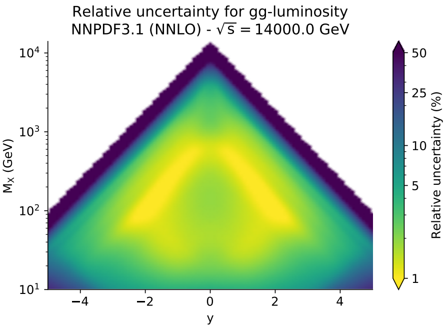
How are we getting there?

So, how large are PDF uncertainties?

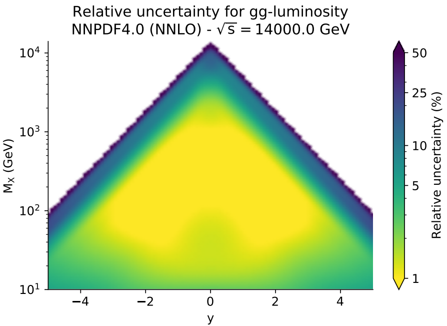
$$\mathcal{L}_{ij}(M_X, y, \sqrt{s}) = \frac{1}{s} f_i\left(\frac{M_X e^{y}}{\sqrt{s}}, M_X\right) f_j\left(\frac{M_X e^{-y}}{\sqrt{s}}, M_X\right)$$

GLUON

NNPDF3.1 (NNLO)



NNPDF4.0 (NNLO)



Steady progress towards 1% relative uncertainties on $\mathcal{L}_{ij}$ in a broad kinematic range

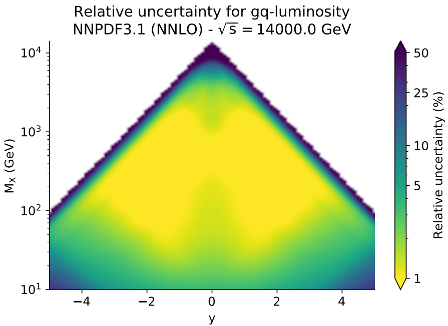
How are we getting there?

So, how large are PDF uncertainties?

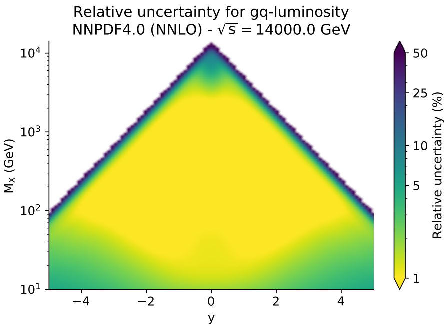
$$\mathcal{L}_{ij}(M_X, y, \sqrt{s}) = \frac{1}{s} f_i\left(\frac{M_X e^{y}}{\sqrt{s}}, M_X\right) f_j\left(\frac{M_X e^{-y}}{\sqrt{s}}, M_X\right)$$

GLUON

NNPDF3.1 (NNLO)



NNPDF4.0 (NNLO)

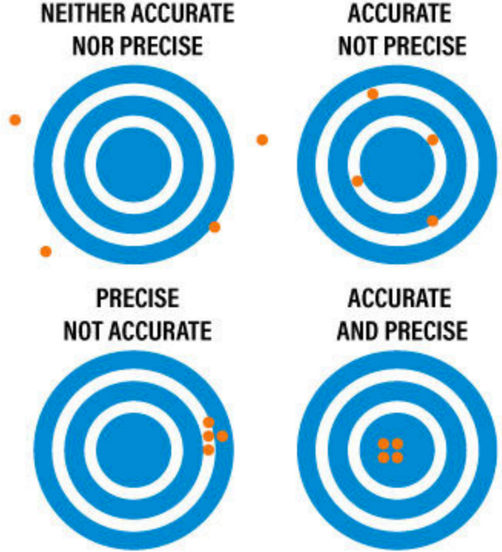


Steady progress towards 1% relative uncertainties on $\mathcal{L}_{ij}$ in a broad kinematic range

How are we getting there?

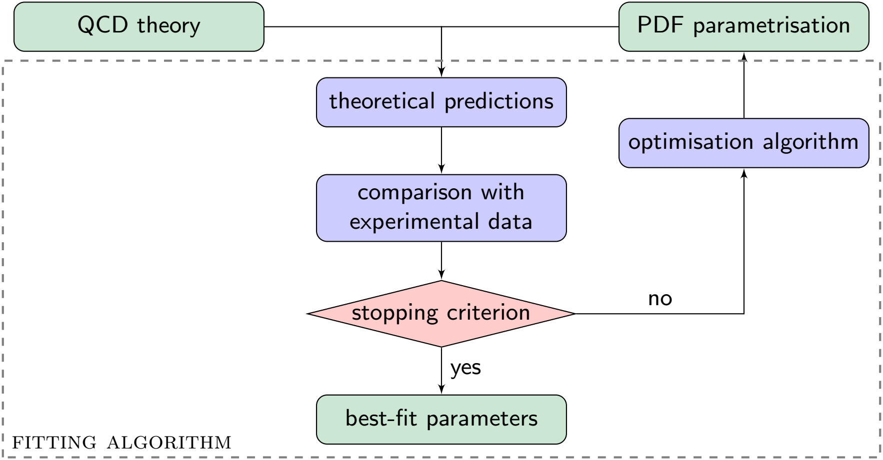
2. The problem

A tale of accuracy and precision



The path towards 1% PDF uncertainties goes through machine learning

A global PDF determination: the problem



Assume a reasonable PDF parametrisation

Obtain theoretical predictions for various processes and compare predictions to data

Determine the best-fit parameters via optimisation of a proper figure of merit

A global PDF determination: the ingredients

THEORY
or
the theoretical details of the QCD analysis

partonic cross sections

DGLAP evolution

DATA
or
the set of observables included in the analysis

experimental uncertainties

uncorr./corr.   add./mult.

METHODOLOGY
or
a prescription to determine PDFs

parametrisation

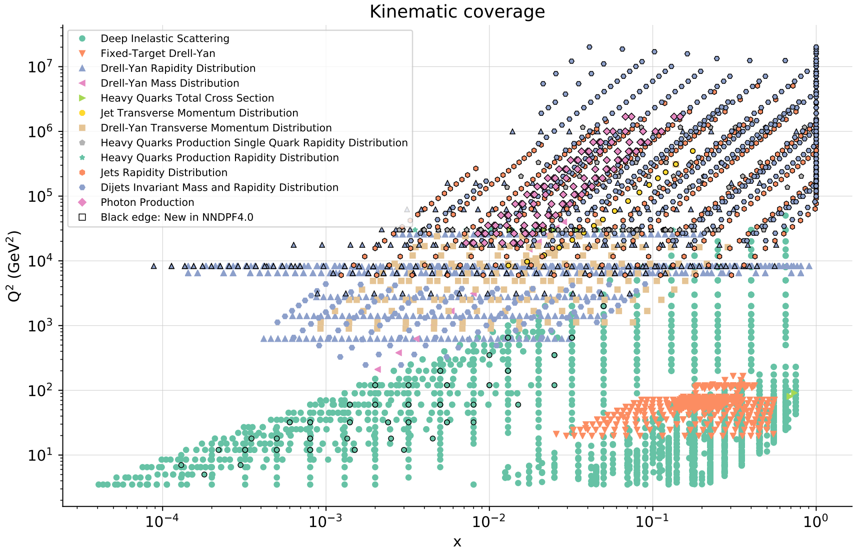
minimisation

error propagation

validation

Each of these ingredients is a source of uncertainty in the PDF determination

Data: kinematic coverage



NNPDF4.0 (NNLO) $\quad N_{\mathrm{dat}} = 4618 \quad N_{\mathrm{proc}} = 13$

PDF determination in Bayesian language

Inverse Problem

Given a data set D, determine $p(f|D)$ in the space of PDFs $f:[0,1]\to\mathbb{R}$

For any observable $\mathcal{O}$ depending on a set of PDFs $[f]$

its expectation value and uncertainty are functional integrals over the space of PDFs

$$\langle\mathcal{O}[f]\rangle = \int \mathcal{D}f\, p(f|D)\, \mathcal{O}[f] \qquad \text{expectation value}$$

$$\sigma_{\mathcal{O}}[f] = \left[\int \mathcal{D}f\, p(f|D)\, \left(\mathcal{O}[f] - \langle\mathcal{O}[f]\rangle\right)^2\right]^{1/2} \qquad \text{uncertainty}$$

ILL-DEFINED PROBLEM

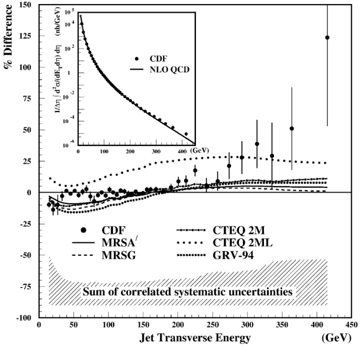
determine a set of infinite-dimensional objects (the PDFs of each parton)
from a finite piece of information (the experimental data points)

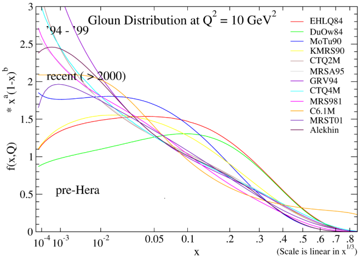
Solution:

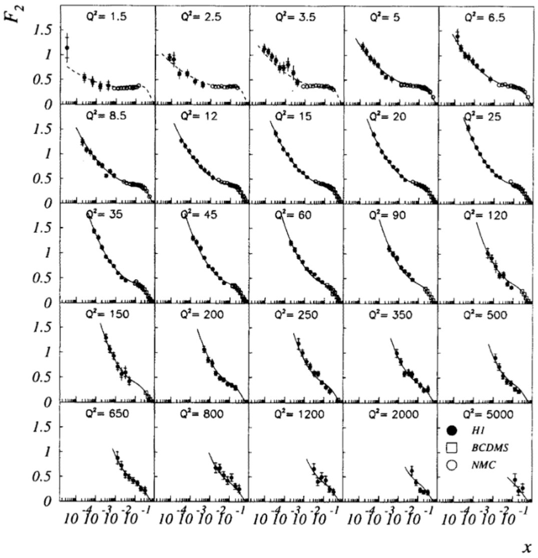
Approximate $p(f|D)$ with its projection on the space of parameters $p(\theta|D,\mathcal{H})$

Determine $p(\theta|D,\mathcal{H}) \propto p(D|\theta,\mathcal{H})p(\theta|\mathcal{H})$ from MAP $\theta^* = \arg\max_\theta p(\theta|D,\mathcal{H})$

Uncertainty estimation







circa 1995: rise of HERA F_2^p and CDF jet inconsistency

3. The NNPDF methodology

A machine learning and machine learned methodology

Why does the NNPDF methodology stand out?

uncertainty representation
bootstrap of experimental uncertainties
unambiguous statistical characterisation of uncertainties

parametrisation
neural network
reduction of parametrisation bias

optimisation
stocastic gradient descent
efficient exploration of the parameter space

validation
closure and future tests
characterisation of the interpolation and extrapolation uncertainties

PDF uncertainties are statistically sound, minimally biased, and completely characterised
The systematic use of machine learning and artificial intelligence plays a crucial role

Uncertainty representation: the functional Monte Carlo

1. Generate (art) replicas of (exp) data according to the distribution

$$\mathcal{O}_i^{(art)(k)} = \mathcal{O}_i^{(exp)} + r_i^{(k)}\sigma_{\mathcal{O}_i}\,, \qquad i = 1, \dots N_{\rm dat}\,, \qquad k = 1, \dots, N_{\rm rep}$$

   where $r_i^{(k)}$ are (Gaussianly distributed) random numbers for each k-th replica ($r_i^{(k)}$ can be generated with any distribution, not neccesarily Gaussian)

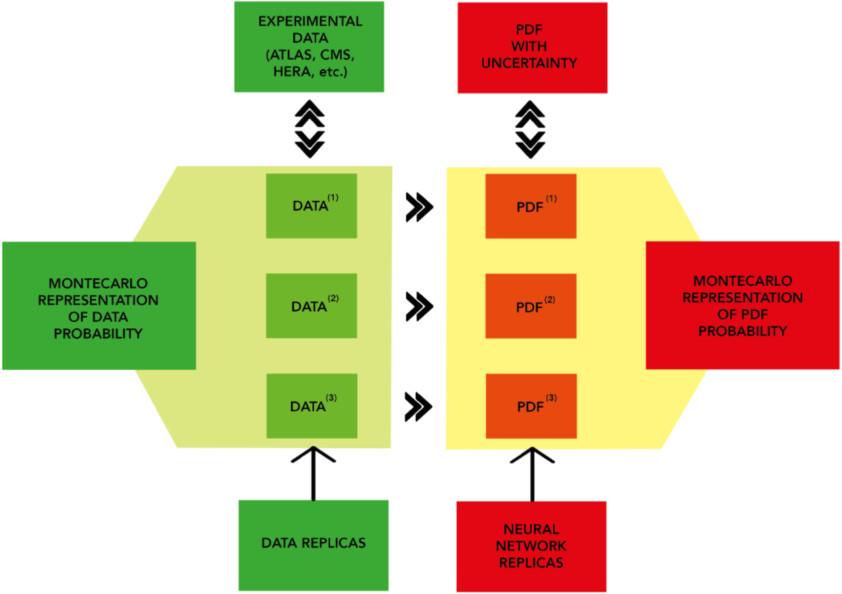
2. Perform a fit for each replica $k = 1, \dots, N_{\rm rep}$
3. Compact computation of observables and their uncertainties
   (PDF replicas are equally probable members of a statistical ensemble)

$$\langle \mathcal{O}[f(x,Q^2)] \rangle = \frac{1}{N_{\text{rep}}} \sum_{k=1}^{N_{\text{rep}}} \mathcal{O}[f^{(k)}(x,Q^2)]$$

$$\sigma_{\mathcal{O}}[f(x,Q^2)] = \left[\frac{1}{N_{\text{rep}}-1} \sum_{k=1}^{N_{\text{rep}}} \left(\mathcal{O}[f^{(k)}(x,Q^2)] - \langle \mathcal{O}[f(x,Q^2)] \rangle \right)^2 \right]^{1/2}$$

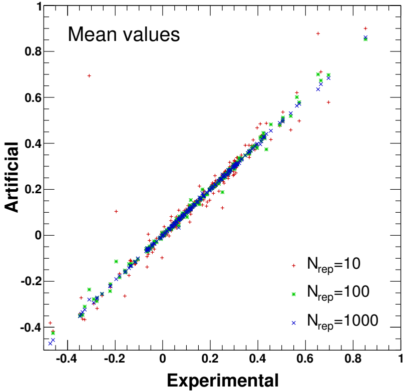
$\Rightarrow$ **knowledge of likelihood shape not necessary**
$\Rightarrow$ **computational expensive: need to perform $N_{\rm rep}$ fits instead of one**

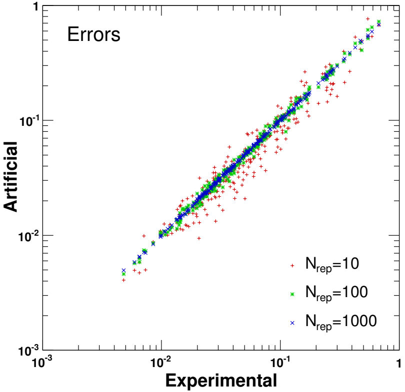
Uncertainty representation: the functional Monte Carlo



The Monte Carlo method: determining the sample size

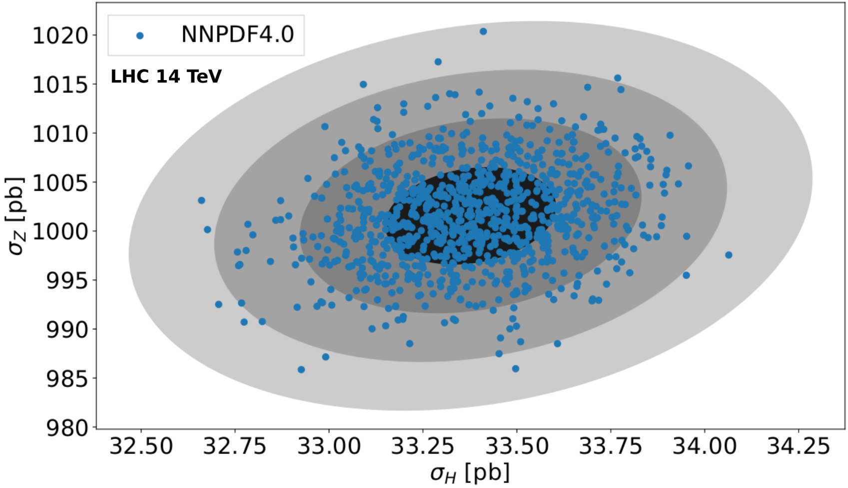
Require that the average over the replicas reproduces the central value of the original experimental data to a desired accuracy (the standard deviation reproduces the error and so on)





Accuracy of few % requires ~ 100 replicas

The Monte Carlo method: importance sampling



Importance sampling is reproduced correctly (14 replicas out of 1000 vs 98.9% C.I.)
Outliers in the distribution of fitted replicas are good fits to unlikely fluctuations of the data

Parametrisation: general features

Problem projected onto the finite-dimensional space of parameters

Choose a parametrisation at an initial scale Q_0^2
for each independent parton i (or a combination of them)

$$\boxed{xf_i(x,Q_0^2) = A_{f_i}\, x^{a_{f_i}}\, (1-x)^{b_{f_i}}\, \mathscr{F}_i(x,\{c_{f_i}\})}$$

small x

$$xf_i(x,Q_0^2) \xrightarrow{x\to 0} x^{a_{f_i}}$$

$$\mathscr{F}_i(x,\{c_{f_i}\}) \xrightarrow[x\to 1]{z\to 0} \text{finite}$$

$$\xrightarrow[\text{interpolation in between}]{}$$

large x

$$xf_i(x,Q_0^2) \xrightarrow{x\to 1} (1-x)^{b_{f_i}}$$

The problem is reduced to the determination of the finite set of parameters $\{c_{f_i}\}$

The interpolating function $\mathscr{F}_i(x,\{c_{f_i}\})$ should be sufficiently

GENERAL (the possible PDF behaviours in the space of functions should not be limited)

SMOOTH (PDFs are implicitly assumed ot be smooth functions)

FLEXIBLE (it should be able to adapt to a variety of data and processes)

to describe the data with minimal bias

Parametrisation: two alternative choices

1. Standard parametrisation, *e.g.*

$$\mathscr{F}_i(x,\{c_{f_i}\}) = 1 + a_{f_i}\sqrt{x} + b_{f_i}x + c_{f_i}x^2$$

in terms of a (relatively) small set of parameters ($\mathcal{O}(30)$ per PDF set)

$$\{\mathbf{a}\} = \{a_{f_i}, b_{f_i}, c_{f_i}, d_{f_i}\}$$

$\Rightarrow$ **smooth behavior (a desirable feature for a PDF)**
$\Rightarrow$ **potential source of bias if the parametrization is too rigid**

2. Redundant parametrisation, *e.g.*

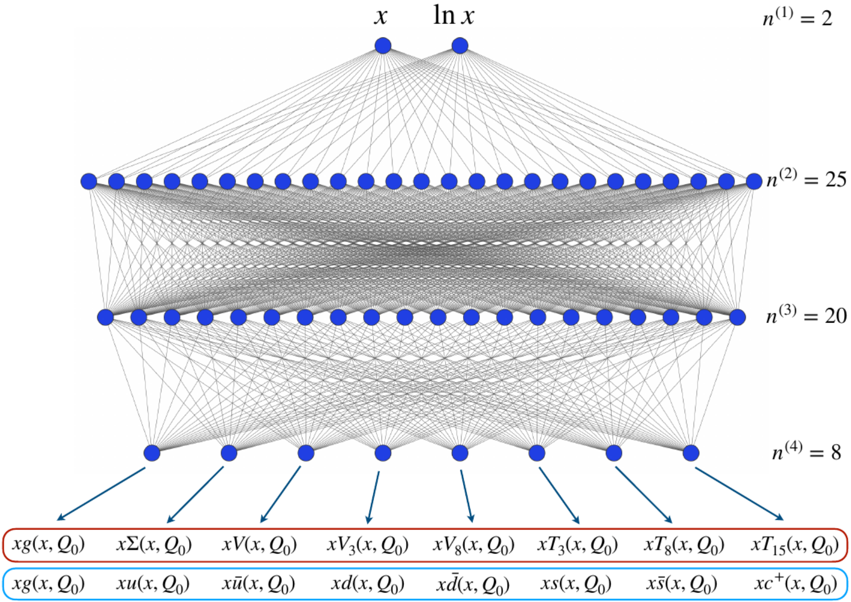
certain classes of polynomials (*e.g.* Bernstein, Chebyschev) or a neural network

in terms of a huge set of parameters ($\mathcal{O}(200)$ per PDF set)

$$\{\mathbf{a}\} = \{\omega_{ij}^{(L-1),f_i}, \theta_i^{(L),f_i}\}$$

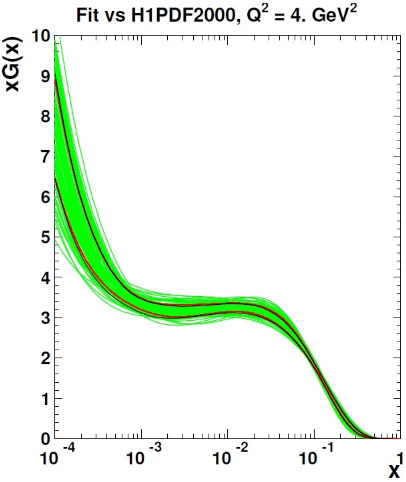
$\Rightarrow$ **potentially non-smooth**
$\Rightarrow$ **bias due to the parametrization reduced as much as possible**

The NNPDF4.0 parametrisation

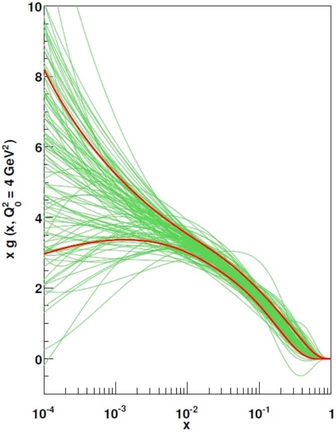


Parametrisation: standard *vs* redundant

HERA-LHC 2009 PDF benchmark

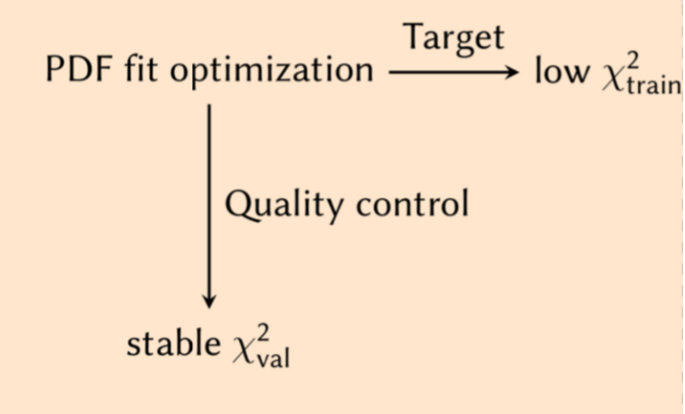


simple parametrization



redundant parametrization (NN)

Optimisation: training and validation



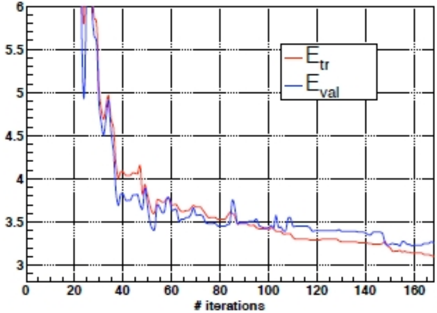
$$\chi^2 = \sum_{i,j}^{\mathrm{N_{dat}}} \left(T_i[\{\mathbf{a}\}] - D_i\right) \left(\mathrm{cov}^{-1}\right)_{ij} \left(T_j[\{\mathbf{a}\}] - D_j\right)$$

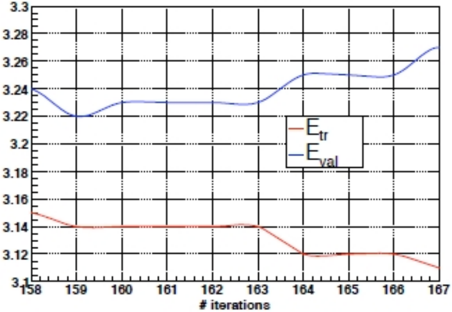
$$(\mathrm{cov})_{ij} = \delta_{ij} s_i^2 + \left(\sum_{\alpha}^{N_c} \sigma_{i,\alpha}^{(c)} \sigma_{j,\alpha}^{(c)} + \sum_{\alpha}^{N_{\mathcal{L}}} \sigma_{i,\alpha}^{(\mathcal{L})} \sigma_{j,\alpha}^{(\mathcal{L})} \right) D_i D_j$$

Optimisation: stopping criterion

CROSS-VALIDATION METHOD

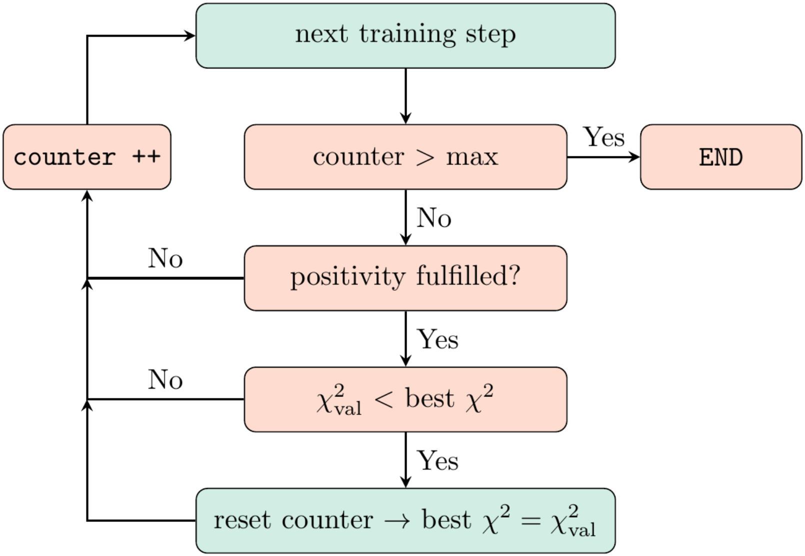
- divide the data into two subsets (training & validation)
- train the NN on training subset and compute χ^2 for each subset
- stop when the χ^2 of validation subset no longer decreases (NN are learning noise!)





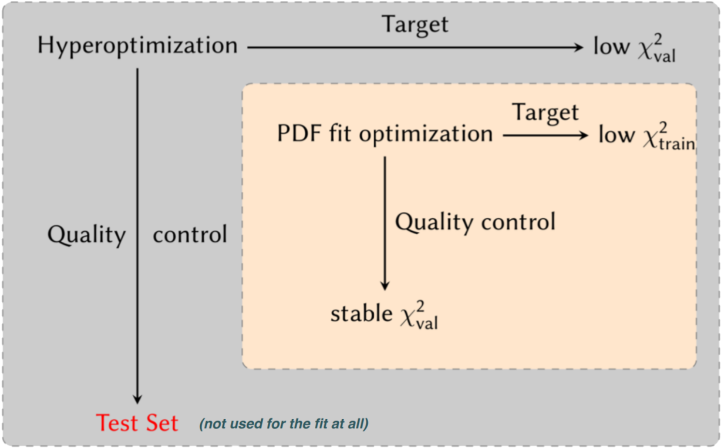
The best fit does not coincide with the χ^2 absolute minimum

Optimisation: iterative algorithm



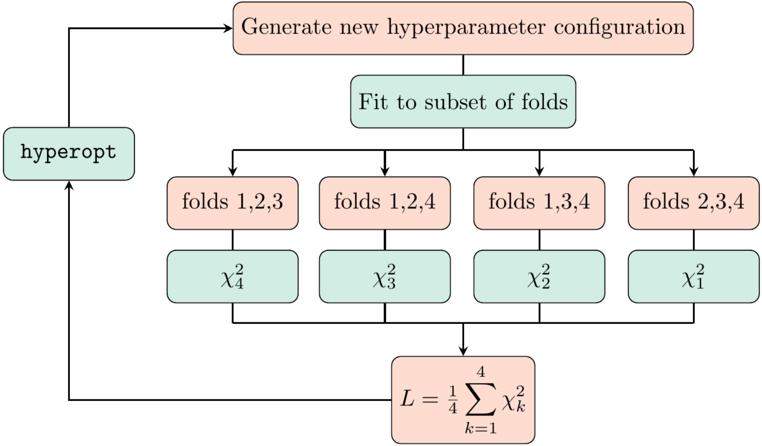
stochastic gradient descent with backpropagation

Hyperoptimisation: fitting the methodology



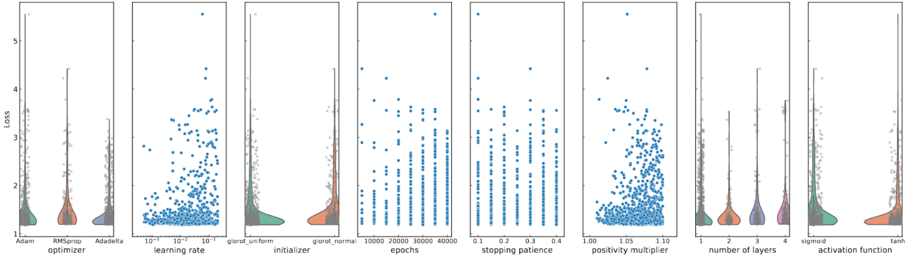
Compare to a Test Set (new set of data previously not used at all)
Who picks the Test Set? Automatic generalisation based on K foldings
Divide the data into n representative sets, fit $n-1$ sets and use n-th set as test set
Hyperoptimise on mean and standard deviation of $\chi^2_{\text{test},i}$, $i = 1 \dots n$

Hyperoptimisation: K-folding



Compare to a Test Set (new set of data previously not used at all)
Who picks the Test Set? Automatic generalisation based on K foldings
Divide the data into n representative sets, fit $n-1$ sets and use n-th set as test set
Hyperoptimise on mean and standard deviation of $\chi^2_{\text{test},i}$, $i = 1 \ldots n$

Hyperoptimisation: scan the hyperparameter space



SCAN parameter space; **OPTIMISE** $\chi^2_{\rm val}$; **BAYESIAN UPDATING**

Hyperoptimisation requires to define a reward (or loss) function to grade each model

This is different from the cost function (optimised separately for each model)

cost function: $C = E_{\rm tr}$ reward function: $R = \frac{1}{2}(E_{\rm val} + E_{\rm test})$

In a hyperparameter scan one compares the performance of hundreds parameter combinations

Some parameters are discrete (type of minimiser), other are continuous (learning rate)

One should visualise which parameters are relevant and which parameters are immaterial

The *violin* plots are the KDE-reconstructed probability distributions for the hyperparameters

Hyperoptimisation successfully validated in closure tests and in future tests

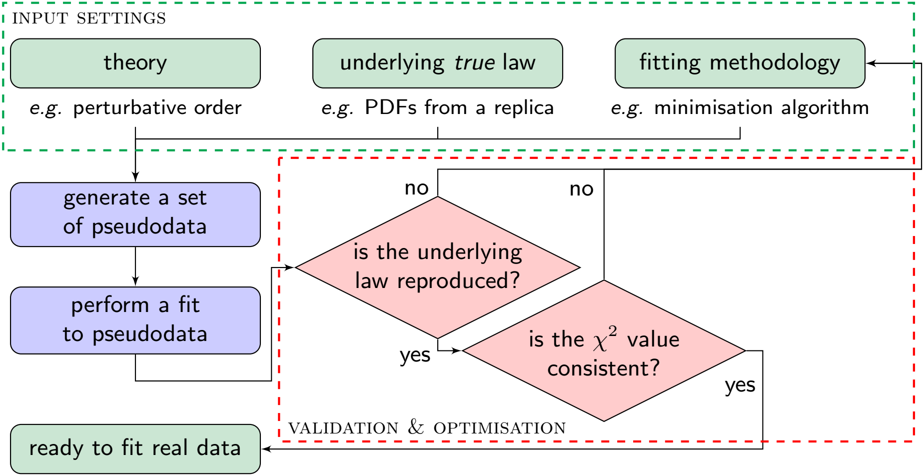
Hyperoptimisation: a machine learned methodology

Parameter	NNPDF4.0	Flavor basis
Architecture	2-25-20-8	2-7-26-27-8
Activation function	hyperbolic tangent	sigmoid
Initializer	`glorot_normal`	`glorot_normal`
Optimizer	`Nadam`	`Nadam`
Clipnorm	6.0×10^{-6}	2.3×10^{-5}
Learning rate	2.6×10^{-3}	2.6×10^{-3}
Maximum # epochs	17×10^{3}	45×10^{3}
Stopping patience	10% of max epochs	16% of max epochs
Initial positivity $\Lambda^{(\mathrm{pos})}$	185	2
Initial integrability $\Lambda^{(\mathrm{int})}$	10	10

	NNPDF4.0 (CPU)	NNPDF4.0 (GPU)
Fit timing per replica	38 min	6.6 min
RAM use	6.1 GB	N/A

Validation: closure tests

Validation of the fitting strategy with known underlying physical law



Full control of procedural uncertainties

Validation: closure test levels

1. Level 0: generate pseudodata D_i^0 with zero uncertainty
   (but $(\mathrm{cov})_{ij}$ in the χ^2 is the data covariance matrix)
   $\rightarrow$ fit quality can be arbitrarily good, if the fitting methodology is efficient: $\chi^2/N_{\mathrm{dat}} \sim 0$
   $\rightarrow$ validate fitting methodology (parametrisation, minimisation)
   $\rightarrow$ interpolation and extrapolation uncertainty
2. Level 1: generate pseudodata D_i^1 with stochastic fluctuations (no replicas)

$$D_i^1 = (1 + r_i^{\mathrm{nor}}\sigma_i^{\mathrm{nor}}) \left(D_i^0 + \sum_p^{N_{\mathrm{sys}}} r_{i,p}^{\mathrm{sys}}\sigma_{i,p}^{\mathrm{sys}} + r_i^{\mathrm{stat}}\sigma_i^{\mathrm{stat}} \right)$$

   $\rightarrow$ experimental uncertainties are not propagated into PDFs: $\chi^2/N_{\mathrm{dat}} \sim 1$
   $\rightarrow$ functional uncertainty (a large number of functional forms with equally good χ^2)
3. Level 2: generate N_{rep} Monte Carlo pseudodata replicas $D_i^{2,k}$ on top of Level 2

$$D_i^{2,k} = (1 + r_i^{\mathrm{nor},k}\sigma_i^{\mathrm{nor}}) \left(D_i^1 + \sum_p^{N_{\mathrm{sys}}} r_{i,p}^{\mathrm{sys},k}\sigma_{i,p}^{\mathrm{sys}} + r_i^{\mathrm{stat},k}\sigma_i^{\mathrm{stat}} \right)$$

   $\rightarrow$ propagate the fluctuations due to experimental uncertainties into PDFs: $\chi^2/N_{\mathrm{dat}} \sim 1$
   $\rightarrow$ input PDFs lie within the 1-sigma band of the fitted PDFs with a probability of $\sim$ 68%

Validation: closure tests at work

Data region: closure tests

Fit PDFs to pseudodata generated assuming a known underlying law

Define bias and variance

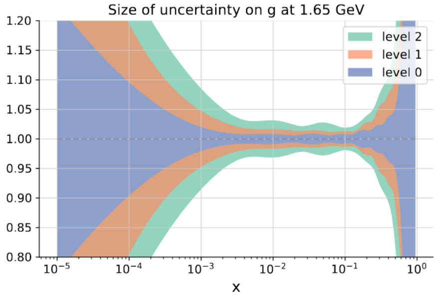
bias difference of central prediction and truth

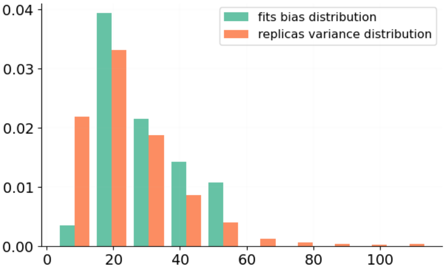
variance uncertainty of replica predictions

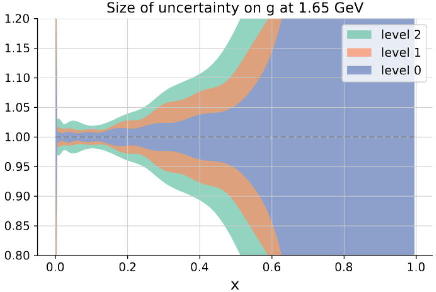
If PDF uncertainty faithful, then

E[bias] = variance

25 fits, 40 replicas each

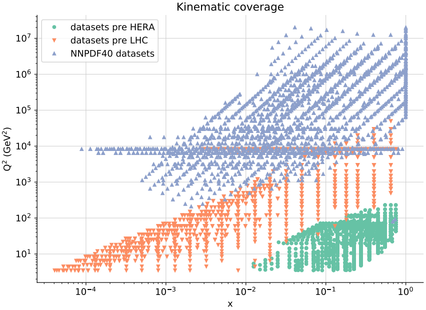


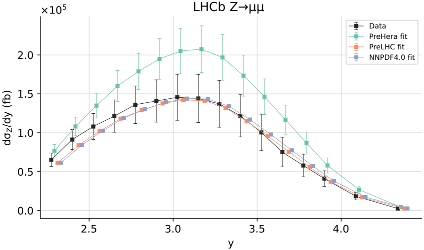




[EPJ C77 (2017) 663; arXiv:2111.05787]

Validation: future tests



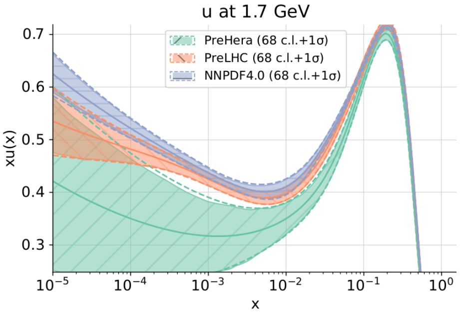


Extrapolation regions: future test

Test PDF uncertainties on data sets not included in a given PDF fit that cover unseen kinematic regions

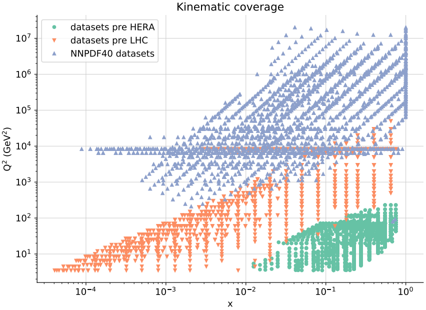
Data set	NNPDF4.0	pre-LHC	pre-HERA
pre-HERA	1.09	1.01	0.90
pre-LHC	1.21	1.20	23.1
NNPDF4.0	1.29	3.30	23.1

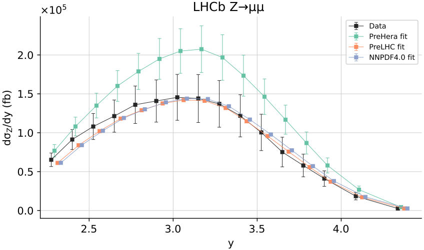
Only exp. cov. matrix



[Acta Phys.Polon. B52 (2021) 243]

Validation: future tests



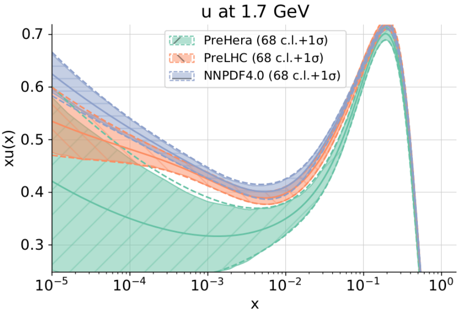


Extrapolation regions: future test

Test PDF uncertainties on data sets not included in a given PDF fit that cover unseen kinematic regions

Data set	NNPDF4.0	pre-LHC	pre-HERA
pre-HERA			0.86
pre-LHC		1.17	1.22
NNPDF4.0	1.12	1.30	1.38

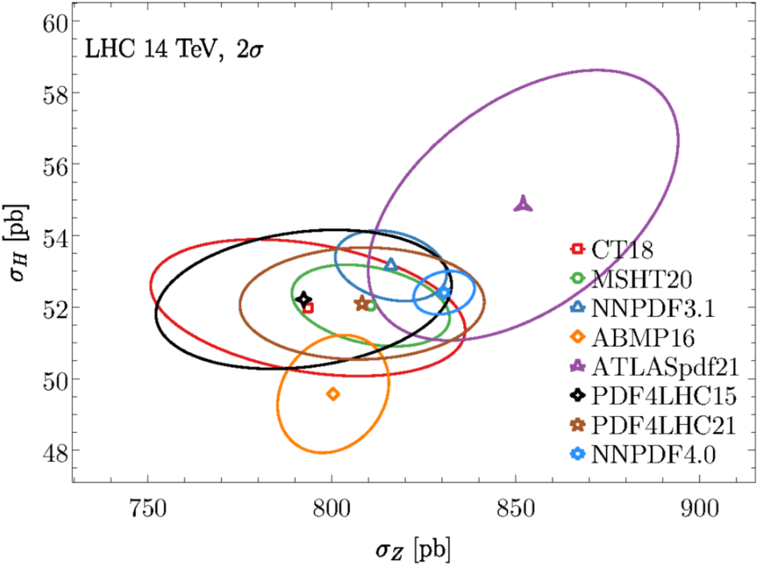
Exp+PDF cov. matrix



[Acta Phys.Polon. B52 (2021) 243]

4. To conclude

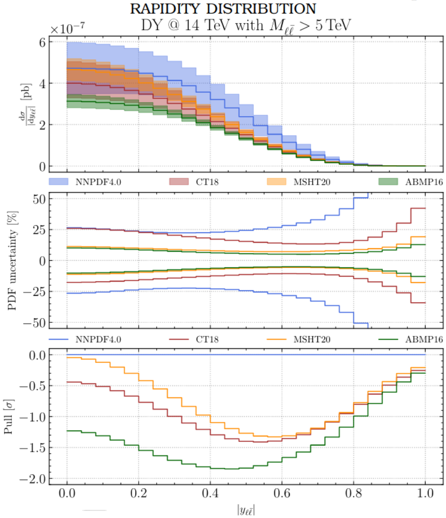
Uncertainties in the data region

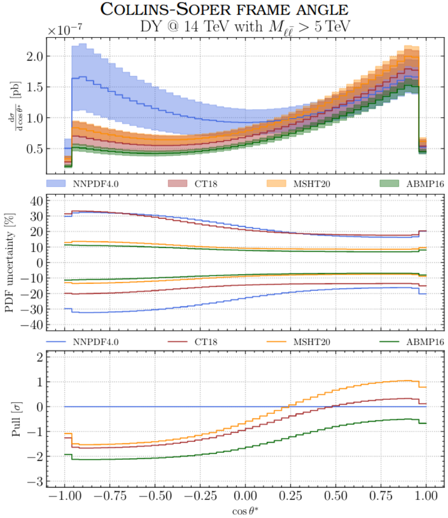


PDFs from a machine learned methodology are more precise

Uncertainties in the extrapolation region

Z' production





PDFs from a machine learned methodology are more accurate

Public code

NNPDF

Tests passing DOI 10.5281/zenodo.6542572

NNPDF: An open-source machine learning framework for global analyses of parton distributions

The NNPDF collaboration determines the structure of the proton using Machine Learning methods. This is the main repository of the fitting and analysis frameworks. In particular it contains all the necessary tools to reproduce the NNPDF4.0 PDF determinations.

Documentation

The documentation is available at https://docs.nnpdf.science/

Install

See the NNPDF installation guide for the conda package, and how to build from source.

Please note that the conda based workflow described in the documentation is the only supported one. While it may be possible to set up the code in different ways, we won't be able to provide any assistance.

We follow a rolling development model where the tip of the master branch is expected to be stable, tested and correct. For more information see our releases and compatibility policy.

NNPDF

Search docs

- Getting started
- Fitting code: `n3fit`
- Code for data: validphys
- Handling experimental data: Buildmaster
- Storage of data and theory predictions
- Theory
- Chi square figures of merit
- Contributing guidelines and tools
- Releases and compatibility policy
- Continuous integration and deployment
- Servers
- External codes
- Tutorials

`https://github.com/NNPDF`