

Parton Distribution Functions and Neural Networks

Lorenzo Mansi

Table of contents

1. Introduction
2. PDFs
3. Neural Networks
4. NNPDFs
5. Backup Material



Introduction

The aim of this project is to illustrate how neural network algorithms are used to compute parton distribution functions (PDFs henceforth).

Starting from minimal bases the parton model will be illustrated briefly, then it will be explained how a neural network works, concluding with an illustration of their application to compute PDFs.

Further attention will be put in describing the various NNPDF (Neural Network PDF henceforth) developed so far.

PDFs

DIS

Bjorken Scaling

Parton Model

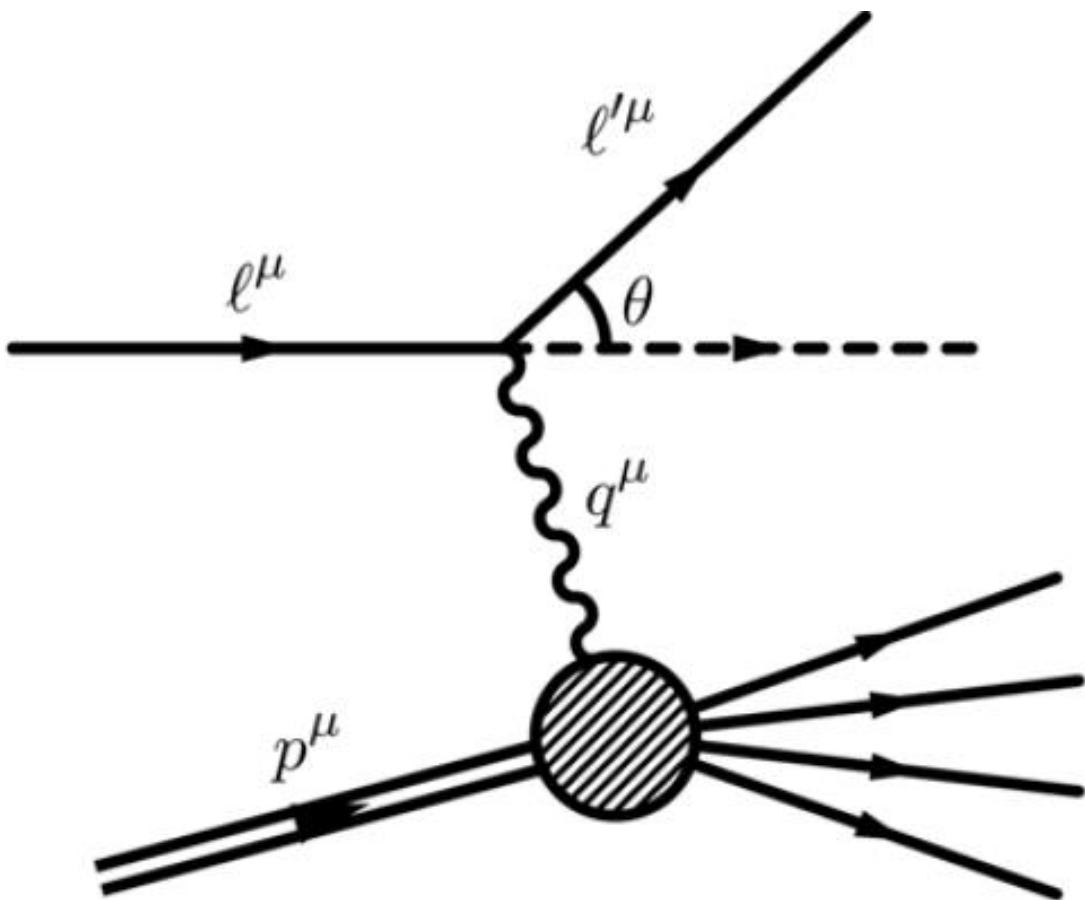
DGLAP

DIS

Electrons used as a probe, at least 2 GeV are required to resolve proton structure.

We seek bound states, the first at 1230 MeV, is the Δ^+ resonance.

Inelastic scattering variables are v and Q^2 , deep inelastic scattering: Q^2 tends to $+\infty$.



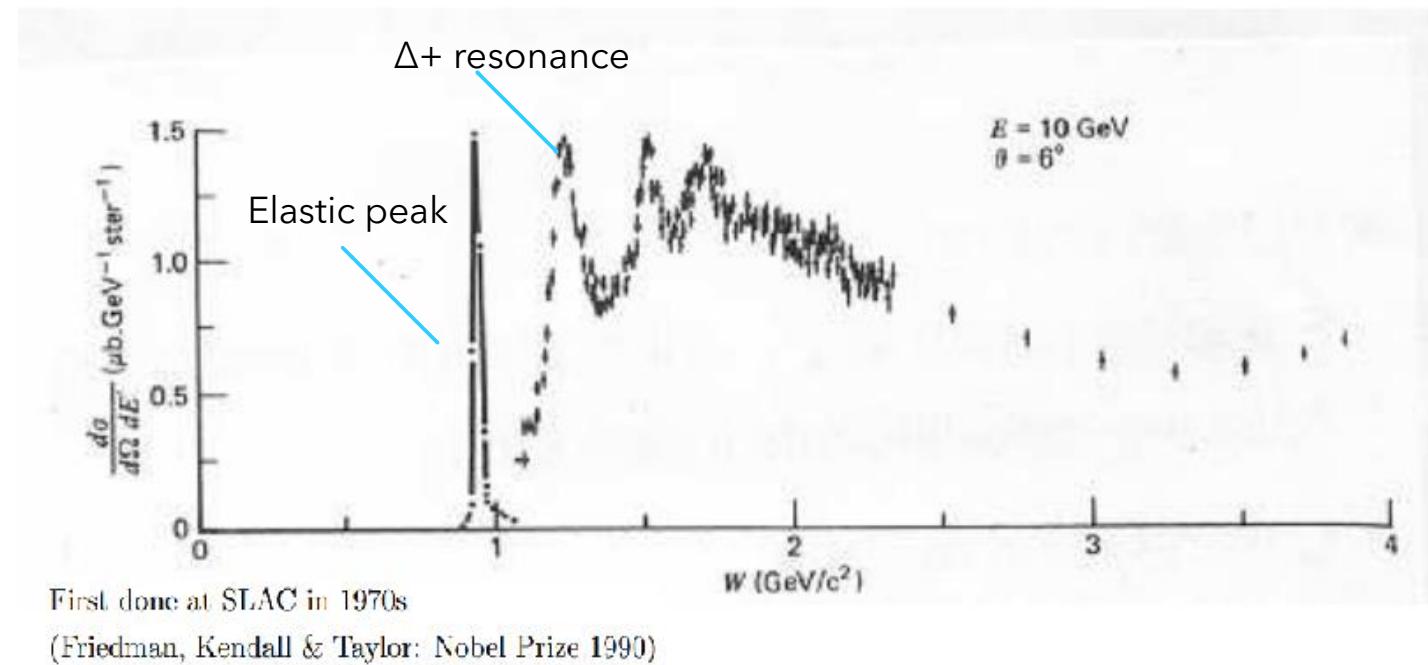
$$\nu = E - E' = \frac{\mathbf{p} \cdot \mathbf{q}}{M}$$

$$Q^2 = -q^2 = 4EE' \sin^2 \frac{\theta}{2}$$

$$W^2 = (\mathbf{p} + \mathbf{q})^2 = M^2 + 2m\nu - Q^2$$

Bjorken Scaling

Cross section for inelastic scattering at high probe energy should depend only on dimensionless parameters once the appropriate energy dimension for the cross section has been factored out.



$$\frac{d\sigma}{d\Omega dE'} = \frac{4\alpha^2 E'^2}{(Q^2)^2} \left[W_2(Q^2, \nu) \cos^2 \frac{\theta}{2} + 2W_1(Q^2, \nu) \sin^2 \frac{\theta}{2} \right]$$

$$0 < x = \frac{Q^2}{2M\nu} < 1$$

$$MW_1(Q^2, \nu) \rightarrow F_1(x)$$

$$\nu W_2(Q^2, \nu) \rightarrow F_2(x)$$

Parton Model

The scaling equation formally defines the PDFs but to give them a physical meaning it is worth to remember that partons:

- Are massless
- Carry zero transversal momentum

So the right frame to understand what PDFs are, is the infinite momentum frame in which the parton after being struck inverts its momentum.

At leading order PDFs are probability distribution: they represent the probability to find a parton «i» with a momenta «xp».

Scaling equation

$$F_2(x) = xF_1(x) = \sum_i Q_i^2 (x f_i(x) + x \bar{f}_i(x))$$

↑
Sum on parton flavors

PDFs

$$x = \frac{Q^2}{2M\nu} = -\frac{q_z^2}{2q_z p_z} = \frac{k_z}{p_z}$$

DGLAP

- Each PDF obeys the Altarelli-Parisi equation.
- “P_{ab}” are called “parton splitting functions” they represent the probability of a parton “b” to become a parton “a”.
- A suitable basis called “Evolution basis” can be chosen to decouple the system.

$$\frac{\partial f_a}{\partial \ln \mu^2} = \frac{\alpha_s(\mu^2)}{2\pi} \sum_b P_{ab} \otimes f_b$$
$$a \otimes b = b \otimes a = \int_x^1 dy \frac{a(y)}{y} b\left(\frac{x}{y}\right)$$

$$\Sigma = \sum_j q_j^+$$

$$V = \sum_j q_j^-$$

$$V_3 = u^- - d^-$$

$$V_8 = u^- + d^- - 2s^-$$

$$V_{15} = u^- + d^- + s^- - 3c^-$$

$$V_{24} = u^- + d^- + s^- + c^- - 4b^-$$

$$V_{35} = u^- + d^- + s^- + c^- + b^- - 5t^-$$

$$T_3 = u^+ - d^+$$

$$T_8 = u^+ + d^+ - 2s^+$$

$$T_{15} = u^+ + d^+ + s^+ - 3c^+$$

$$T_{24} = u^+ + d^+ + s^+ + c^+ - 4b^+$$

$$T_{35} = u^+ + d^+ + s^+ + c^+ + b^+ - 5t^+$$

Neural Networks

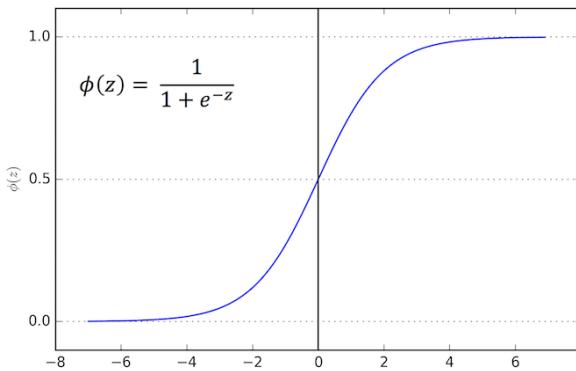
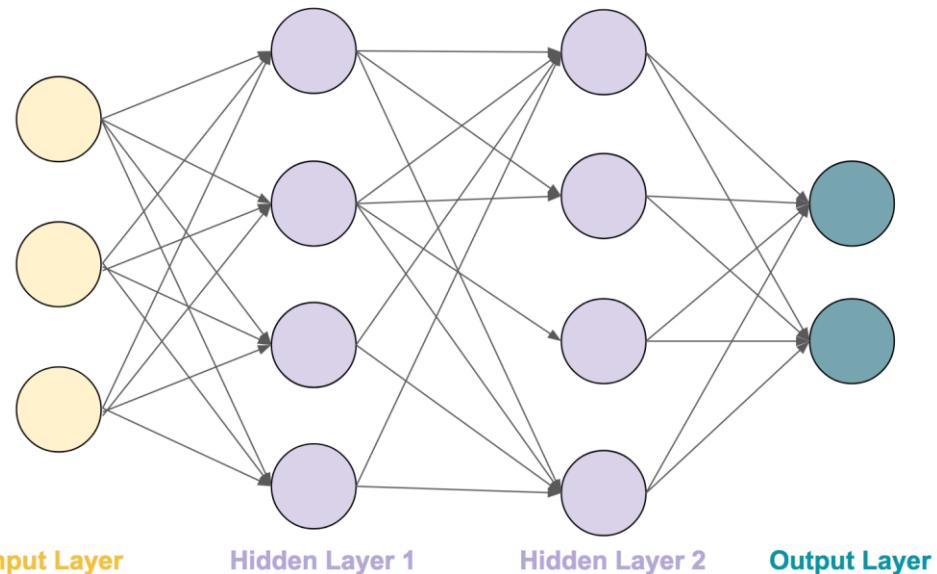
Structure

Training

GA

GD

Structure



$$\sigma(x) = \frac{1}{1+e^{-\frac{(x-\theta)}{\rho}}}$$

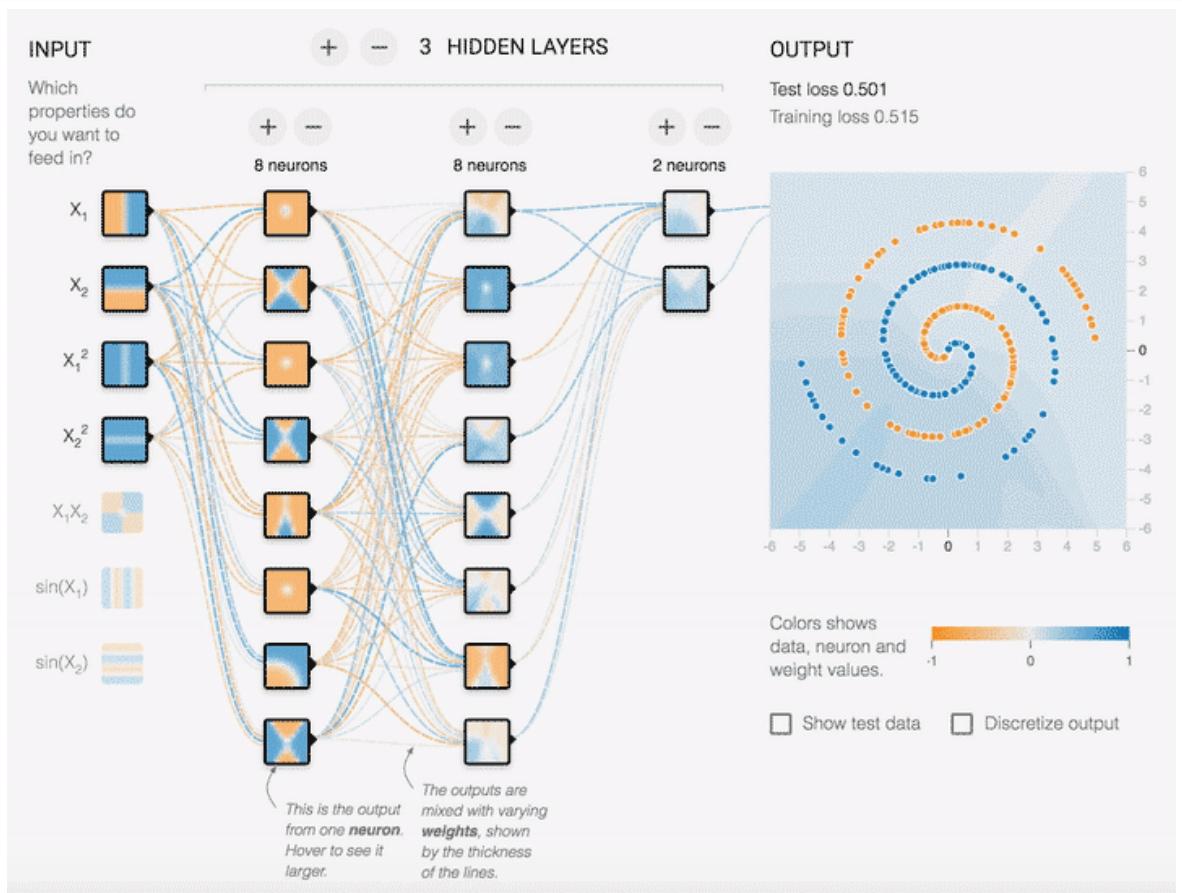
$$\sum_i x_i a_i = \theta \rightarrow \sum_i x_i a_i - \theta = 0$$

- A neural network basically aims to mimic the behaviour of biological neurons and the pattern used by the brain to perform a generic operation.
- Neurons can be either active or inactive if a certain threshold " θ " is reached or not, so "logistic sigmoid function σ " is used as model.
- A neuron which is fed by other neurons' outputs is called a node, each input value in a node has to be weighted suitably.

Training

How does the NN "learn"? The answer can be summarized in these few steps:

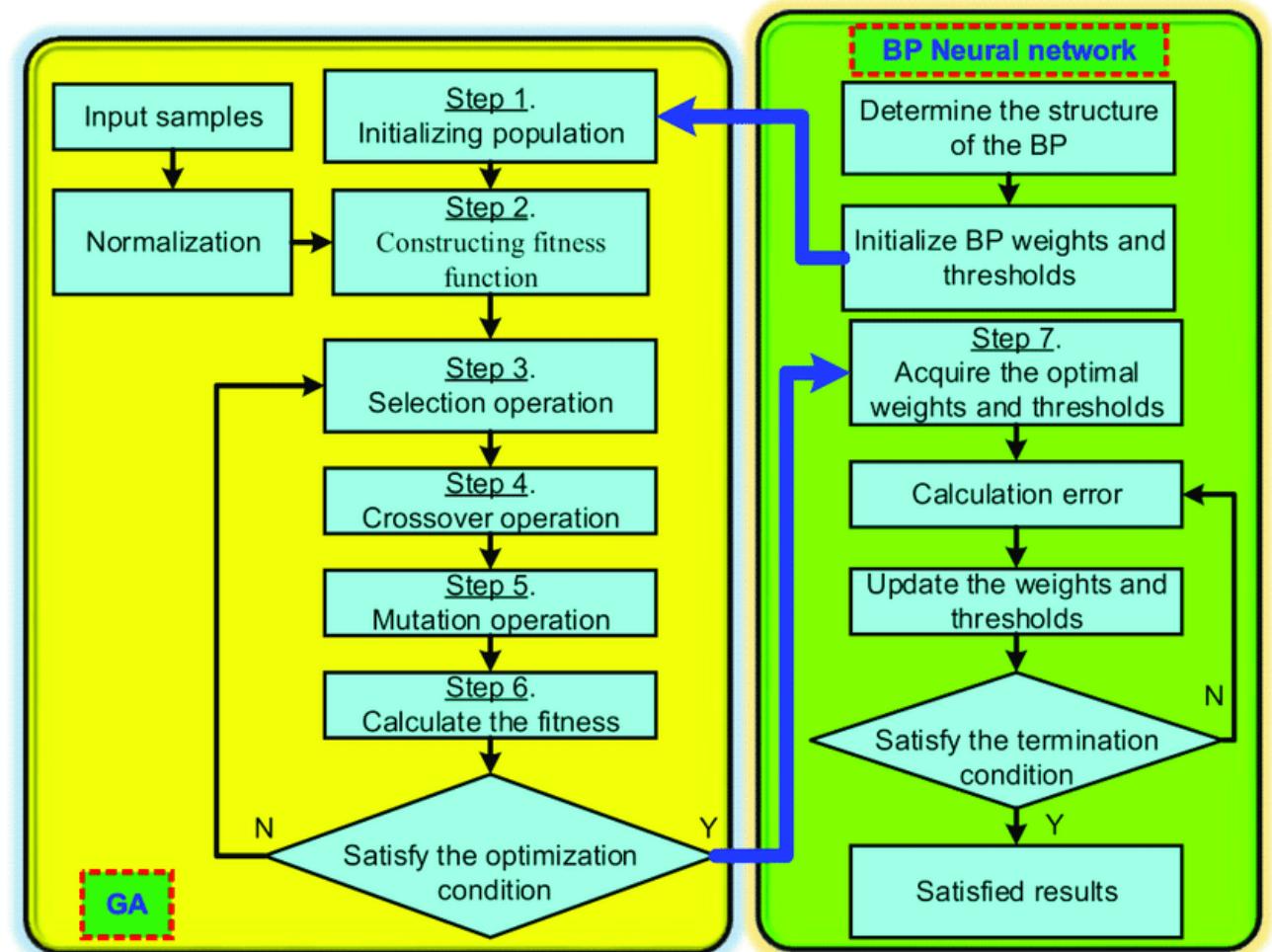
- We require to have some sets of data for which we already know what the neural network output should be. This sort of data are split into training set and validation set.
- We "train" the neural network on the training set requiring the minimization of a certain function χ^2 varying the weights.
- To avoid overtraining we test after a number of iterations the network on the validation set till the χ^2 for the validation set stops improving.



GA: Genetic Algorithm

- Chosen a number of epoch and a number of «mutant nodes»
- At each epoch the mutants vary (it means the weights vary) with a certain probability decided a priori or according to a certain law.
- The process goes on till the minimum χ^2 is obtained or the number of epoch is reached.

Low resource usage but very long time needed.



GD: Gradient Descent

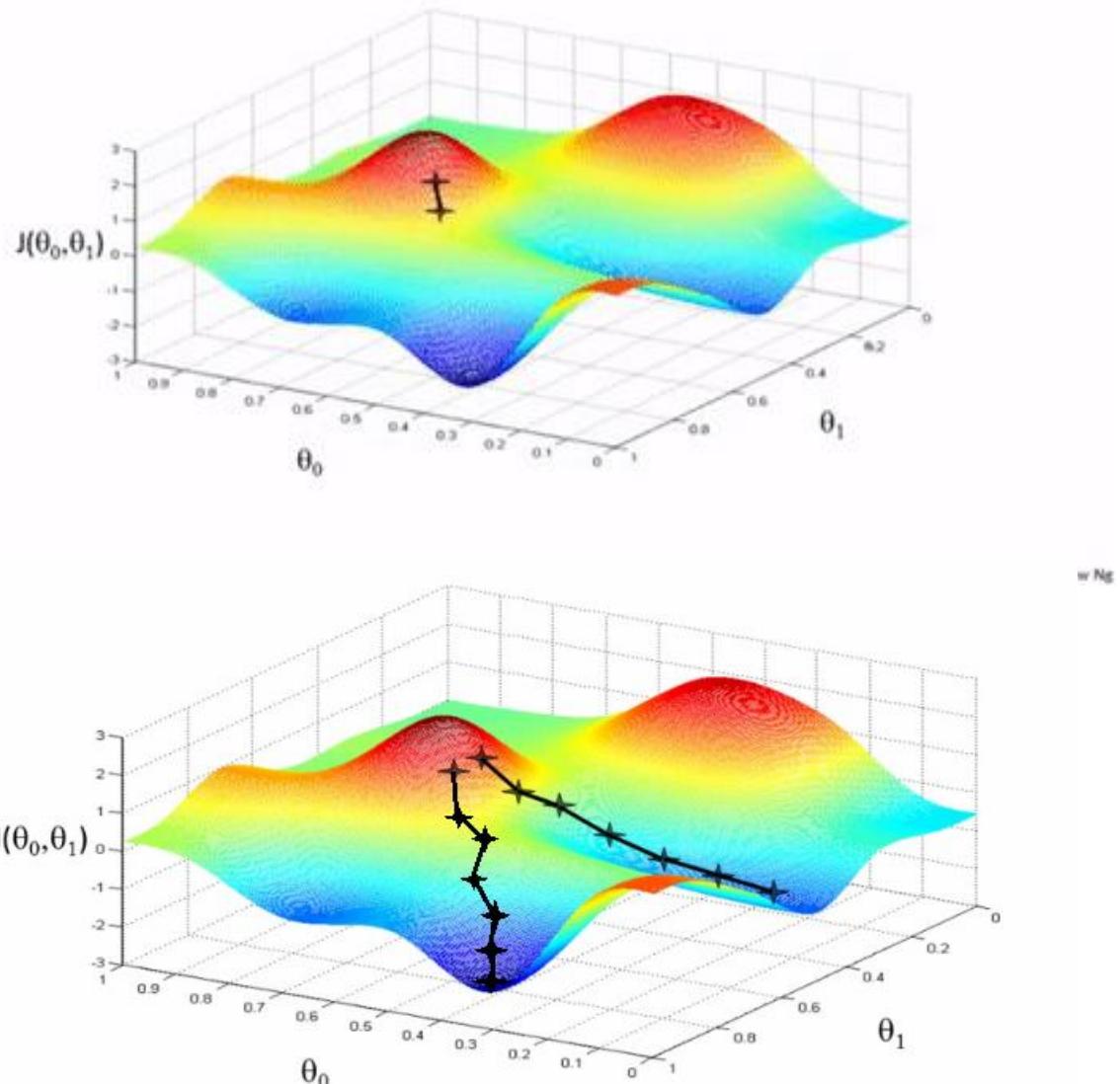
Gradient descent technique consists in:

- given a starting point \vec{x}_0 and a function $y(\vec{x})$, we move from the starting point of a quantity proportional to the gradient of the function in that point:

$$\Delta \vec{x} = -a \vec{\nabla} y|_{\vec{x}_0} \rightarrow \delta y = -a (\vec{\nabla} y|_{\vec{x}_0})^2$$

- Choosing $a > 0$ we actually moved down on the function.
- An error on the path has to be chosen suitably.

High computation power needed but very fast.



NNPDFs

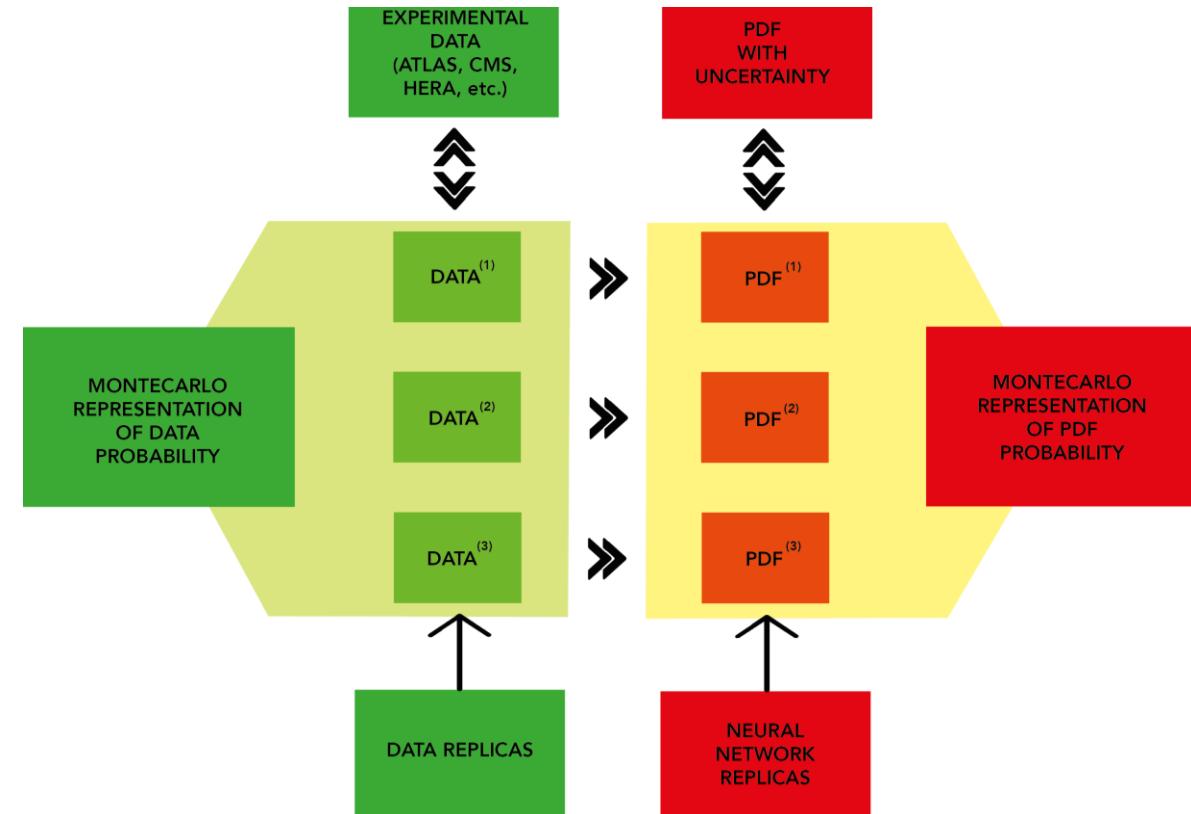
General Aspects
Versions 1,2,3
NNPDF 3.1

General Aspects

Data structure

PDFs as a neural network problem consists in pattern recognition.

1. But what are our datas?
Probability distributions
2. How do we handle them? Turn the input probability distribution of data into a MonteCarlo representation
3. What do we obtain? For each replica a best fit PDF is found (via NN), and then statistics over replicas is done to compute central values, correlation ecc.



LHC cross section p-p:

$$\sigma_X(s, M_X^2) = \sum_{a,b} \int_{x_{min}}^1 dx_1 dx_2 f_{a/h_1}(x_1, M_X^2) f_{b/h_2}(x_2, M_X^2) \hat{\sigma}_{ab \rightarrow X}(x_1 x_2 s, M_X^2)$$

General Aspects

Fit aspects

- Neural Network inputs are x and $\ln(x)$. The behaviour of PDFs at a certain scale is polynomial at high or low x . In fact we seek functions of the form:

$$f_i(x.Q_0) = A_i x^{-\alpha_i} (1 - x)^{\beta_i} \mathbf{NN}_i(x)$$

Where $\mathbf{NN}(x)$ is the part computed through Neural Network.

- The χ^2 is defined as:

$$\chi^2 = \sum_{i,j}^{N_{dat}} (D - P)_i \sigma_{ij}^{-1} (D - P)_j$$

Where D_i is the i-th data point, P_i is the convolution product between the FastKernel tables for point "i" and the PDF model, and σ_{ij} is the covariance matrix between data point "i" and "j".

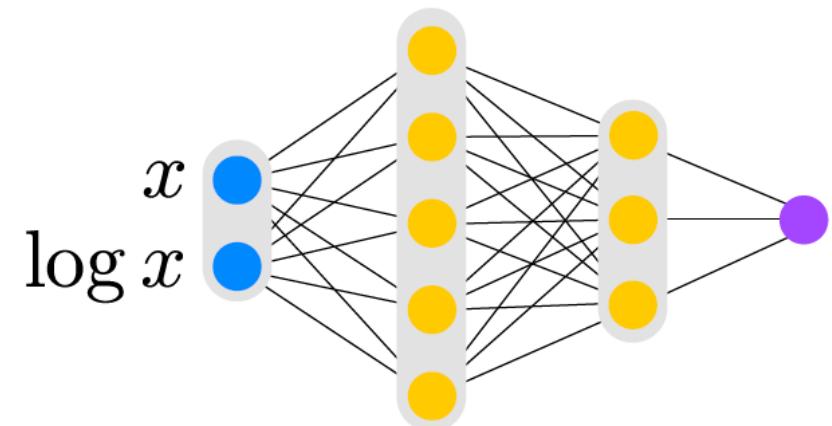
Versions 1,2,3

Structure

Each PDF is parametrized by a preprocessed NN. The values of x and $\ln x$ are taken as input, and the value of the PDF is given as output. The number of independently parametrized PDFs has increased over time but the architecture has remained the same.

The output node activation function is linear, the hidden nodes use a sigmoid activation function. The input is logarithmic per $10^{-4} < x < 0.03$, linear per $0.03 < x < 0.5$.

- NNPDF 1.0: 5 independent PDFs (up and down quarks and antiquarks and the gluon)
- NNPDF 1.1 onwards: 7 independent PDFs (up, down, strange quarks and antiquarks and the gluon)
- NNPDF 3.1: 8 independent PDFs (up, down, strange quarks and antiquarks, total charm, gluon), evolution basis used.



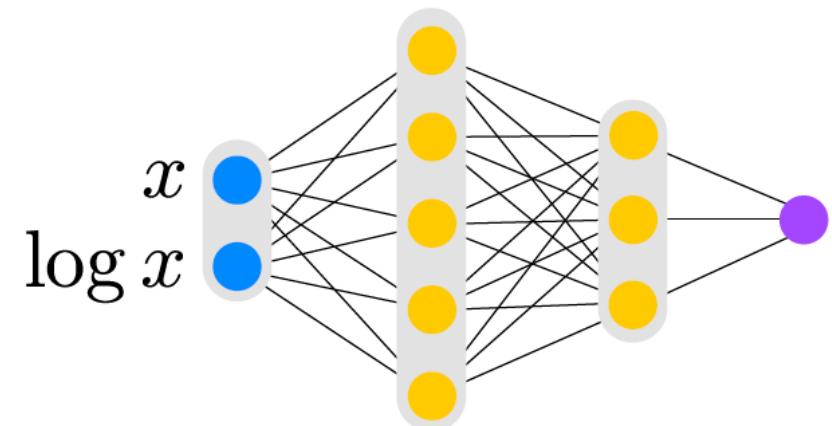
$$f_i(x.Q_0) = A_i x^{-\alpha_i} (1 - x)^{\beta_i} \mathbf{NN}_i(x)$$

Versions 1,2,3

Fit function

The preprocessing exponents α_i and β_i were chosen:

- NNPDF 1.0: to be fixed.
- NNPDF 1.2: were randomly selected in a range decided a priori for each PDF. The range in which exponents were chosen was decided viewing the stability of the fit.
- NNPDF 2.0: the range was quantitatively determined by computing correlation coefficients between χ^2 figures of merit and verifying it remains small.
- NNPDF 3.0: the range is now determined self-consistently: the effective exponents are computed for each independent combination of PDFs and for each PDF replica, the 68% confidence level range is determined for each combination, the fit is repeated with the exponents varied in a range taken equal to twice this range, and the procedure is iterated until the range stops changing.



$$f_i(x.Q_0) = A_i x^{-\alpha_i} (1 - x)^{\beta_i} \mathbf{NN}_i(x)$$

Versions 1,2,3

Fitting method

Genetic algorithms (GA) are used up to NNPDF3, weights are initialized using a random Gaussian distribution for each PDF.

Weights are mutated at random according to the rule:

$$w_i \rightarrow w_i + \eta_i r_i$$

The mutation rates were dynamically adjusted as a function of the number of iterations according to:

$$\eta_i = \frac{\eta_i^0}{N_{ite}^p}$$

Versions 1,2,3

Fitting method

- NNPDF1.0:
 - 2 mutations per PDF,
 - Number of mutants $N_{mut} = 120$
 - $p = 1/3$
 - Maximum number of generations $N_{max} = 5000$.
- NNPDF2.0:
 - the minimization was divided in two epochs
 - Transition at $N_{ite} = 2500$ generations
 - $N_{mut} = 80$ of mutants in the first epoch, decreased ($N_{mut} = 10$) in the second epoch
 - p was now randomly varied between 0 and 1 at each generation
 - maximum number of generations $N_{max} = 30000$.
- NNPDF2.3:
 - the number of mutations was increased to 3 for several PDFs.
- NNPDF3.0, a GA based on nodal mutation has been adopted:
 - Each node in each network is assigned an independent probability of being mutated.
 - If a node is selected, its threshold and all of the weights are mutated according to the rule mentioned before, with now η fixed, and p a random number between 0 and 1 shared by all of the weights.
 - The values $\eta = 15$ and mutation probability 15% per node have been selected as optimal based on closure tests.

Versions 1,2,3

Closure Test

A critical issue in PDF determination is making sure that PDF uncertainties are faithful.

Because the true PDFs are not known, this can only be done through closure testing (implemented from NNPDF 3.0).

- Assume true PDFS to be known: generate fake experimental data
- Can decide data uncertainty (level "0", level "1", level "2")
- Fit PDFS to fake data
- Check whether fit reproduces the underlying truth:
 - Check whether true values gaussianly distribute around the fit
 - Check whether uncertainties are faithful
 - Check stability (Indep. of methodological details)

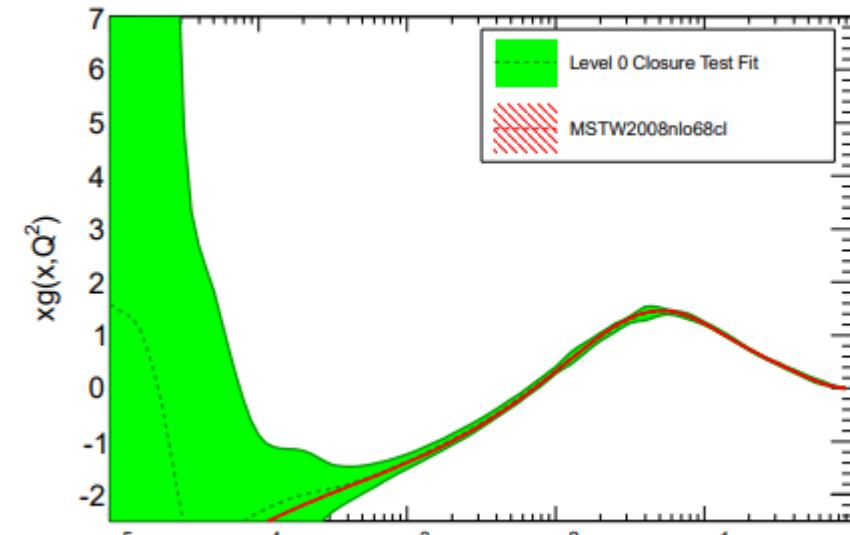
Versions 1,2,3

Closure Test

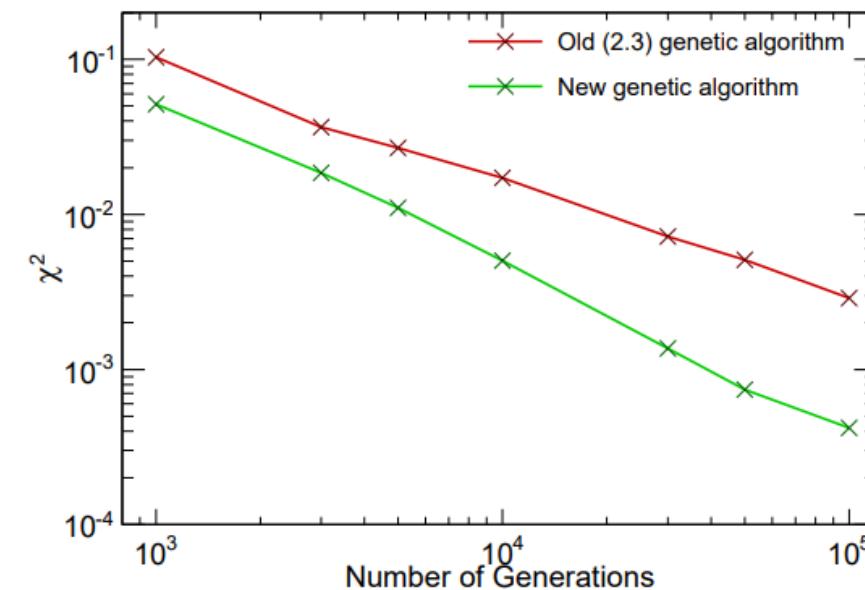
- Level “0” consists of data with no uncertainties.
- Theoretically a perfect χ^2 fit can be performed.
- Uncertainties at data points tend to zero.

Only interpolation and extrapolation uncertainties.

THE GLUON Level 0 closure test vs. MSTW



Effectiveness of Genetic Algorithm in Level 0 Closure Tests



Versions 1,2,3

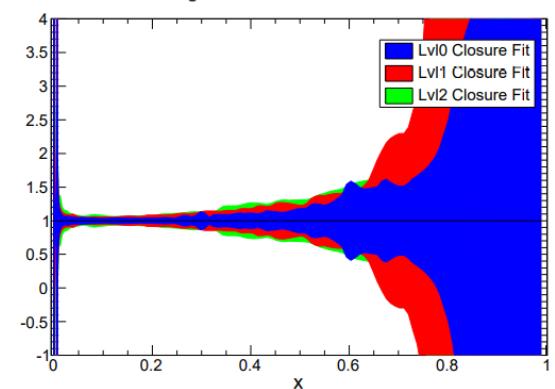
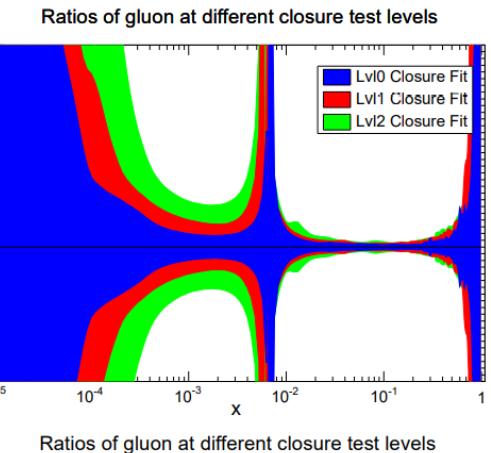
Closure Test

- Level “1” is generated by assuming the probability distribution which corresponds to the published experimental covariance matrix.
- These data correspond to a hypothetical set of experimental results for which the experimental covariance matrix is exactly correct.
- Replicas fitted to same data over and over again.

Functional uncertainty due to infinity equivalent minima.

- Level “2” data are then the Monte Carlo replicas produced out of the level 1 data, as if the latter were actual experimental data.

Functional uncertainty due to data uncertainty.



NNPDF 3.1

- New python based framework: Keras and Tensor Flow libraries.
- Every piece of the NN is subjected to change.
- All fits use GD method instead of GA.
- Only one NN used with 8 outputs (one per flavour) with linear activation function, the first layer is fixed and used to split x in x and $\ln(x)$.
- Introduction of «hyperopt»: systematically scan over many combinations of hyperparameters to find the optimal configuration for the NN.

NNPDF 3.1

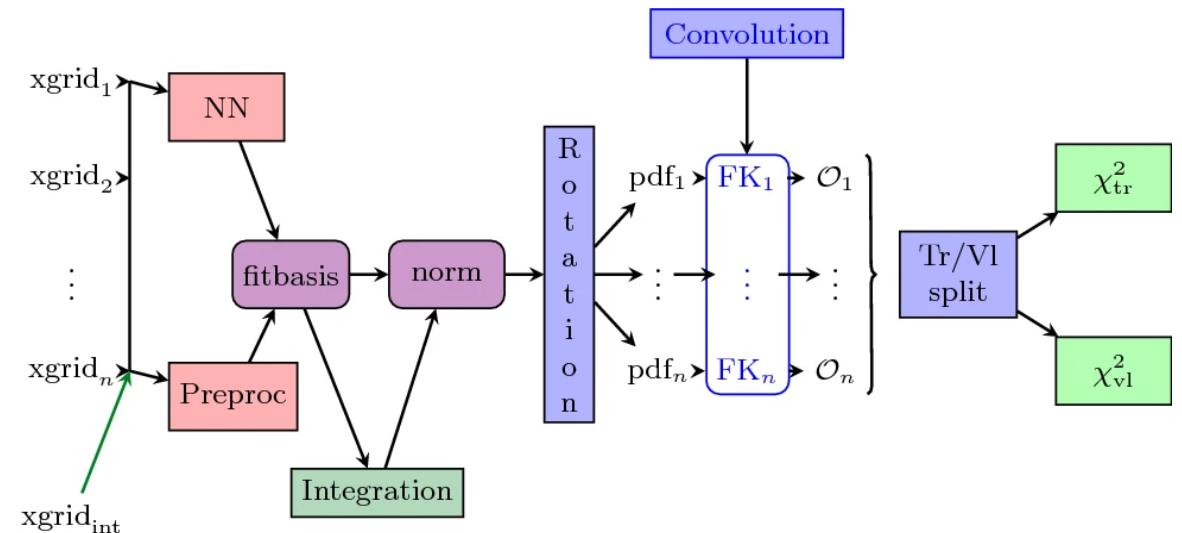
«n3fit»

- «xgrid» contains the x-inputs for each experiment entering the fit
- «Preproc» computes the exponents
- «NN» is for $NN(x)$
- «norm» normalizes the PDFs
- «Rotation» take PDFs from Evolution Basis to Flavor Basis
- «Convolution» creates a rank 4 luminosity tensor:

$$\mathcal{L}_{i\alpha j\beta} = f_{i\alpha} f_{j\beta}$$

- The luminosity tensor is contracted with FK table to obtain observables:

$$\mathcal{O}^n = \mathbf{FK}_{i\alpha j\beta}^n \mathcal{L}_{i\alpha j\beta}$$



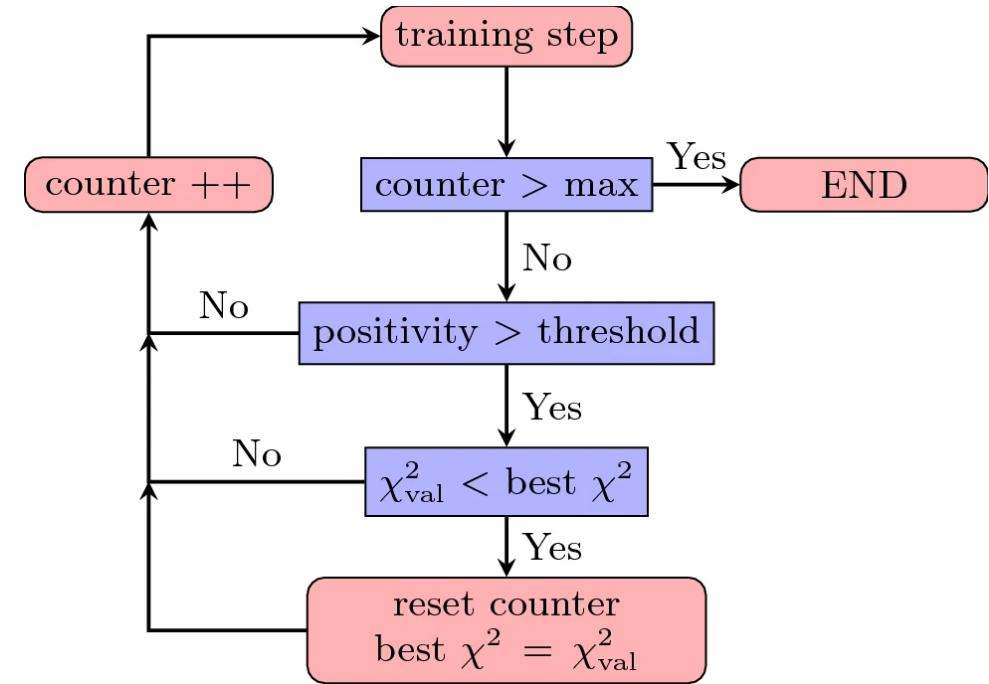
NNPDF 3.1

«n3fit»

- Data are split in Training and Validation set
- A benchmark between the old and the new fitting methodology is performed on DIS fit (3092 datapoints) and Global Fit (4285 datapoints).

Table 1. Comparison of the average computing resources consumed by the old and new methodologies for the DIS and Global setups.

DIS fit	CPU h.	Mem. Usage (GB)	Good replicas
n3fit (new)	0.2	2	95%
nnfit (old)	4	4	70%
Global fit	CPU h.	Mem. Usage (GB)	Good replicas
n3fit (new)	1.5	4	95%
nnfit (old)	30	5	70%



NNPDF 3.1

«hyperopt»

Scans over these parameters:

Table 2. Parameters on which the hyperparameter scan is performed from.^[42]

Neural Network	Fit options	
Number of layers	Optimizer	
Size of each layer	Initial learning rate	
Dropout	Maximum number of epochs	
Activation functions	Stopping Patience	
Initialization functions	Positivity multiplier	

Parameter	DIS only	Global
Hidden layers	2	3
Architecture	35-25-8	50-35-25-8
Activation	tanh	sigmoid
Initializer	glorot_normal	glorot_normal
Dropout	0.0	0.006
Optimizer	Adadelta	Adadelta
Max epochs	40000	50000
Stopping patience	30%	30 %

Table 4. Best models found by our hyperparameter scan for the DIS and global setups using the new n3fit methodology.

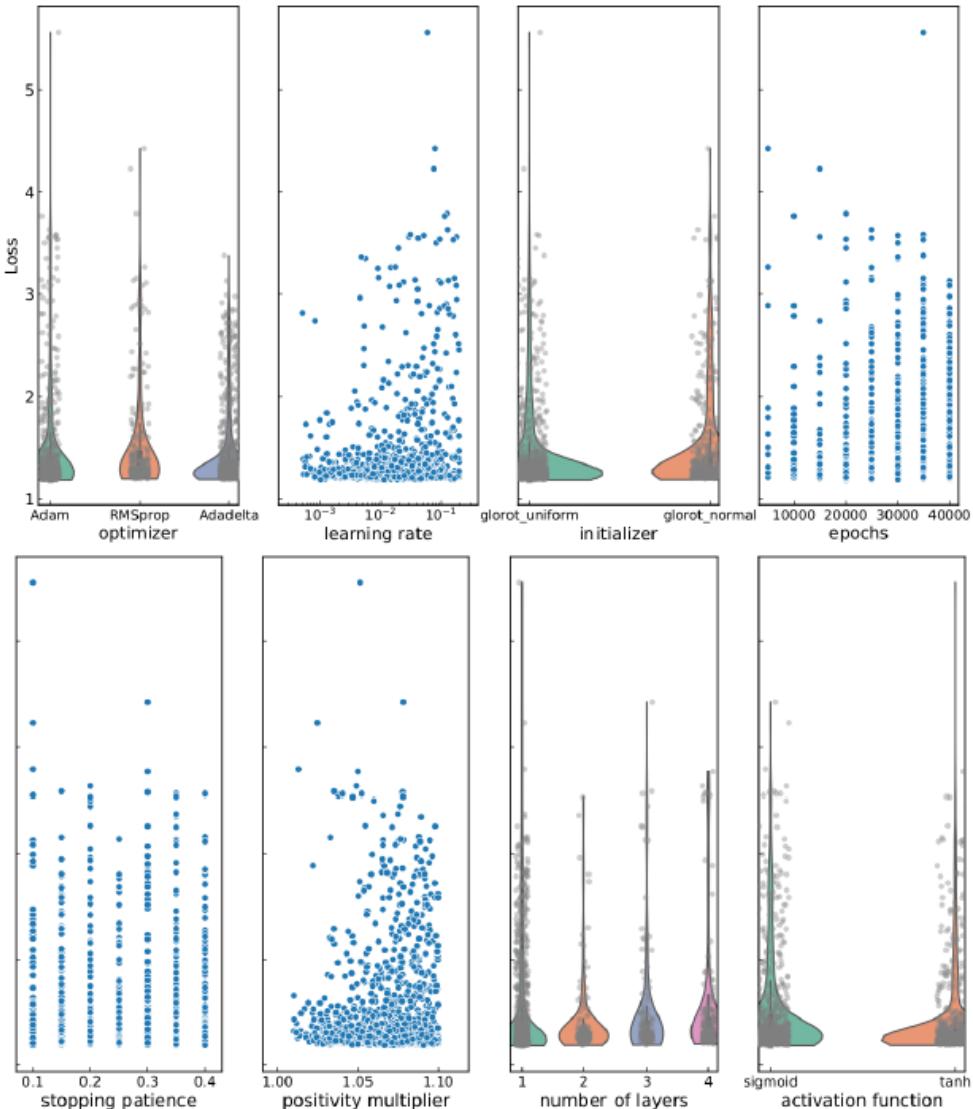
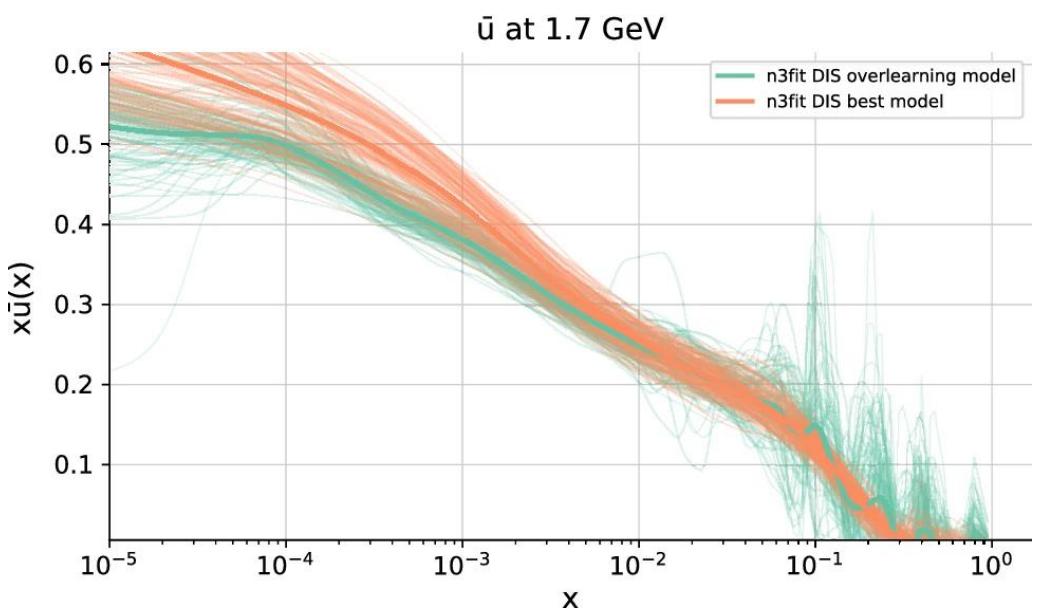


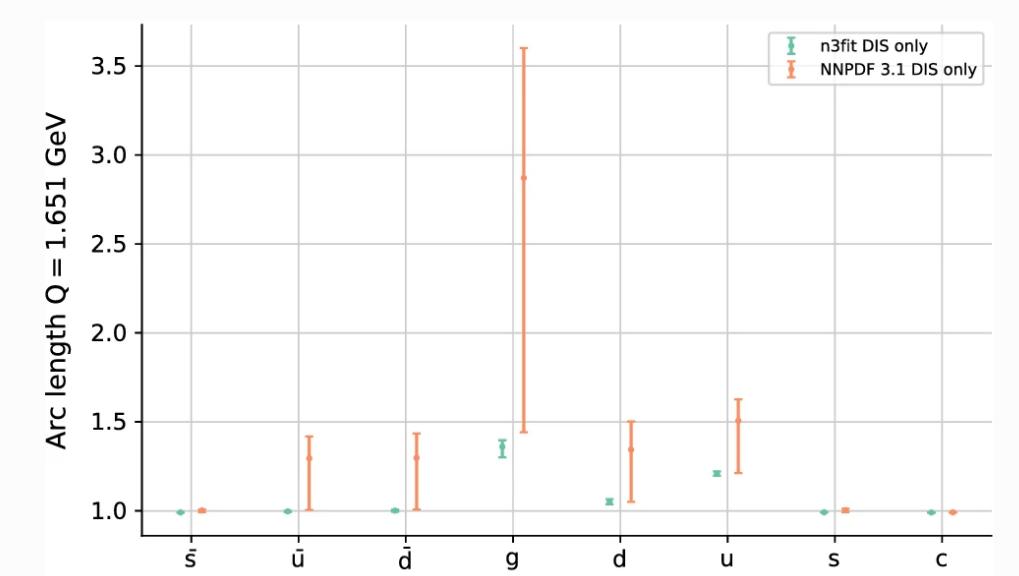
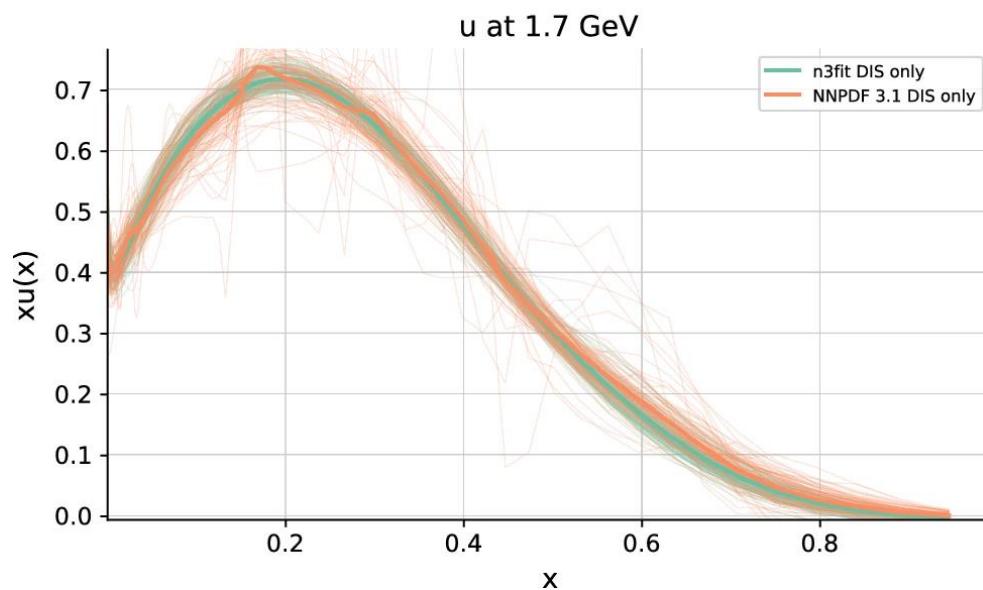
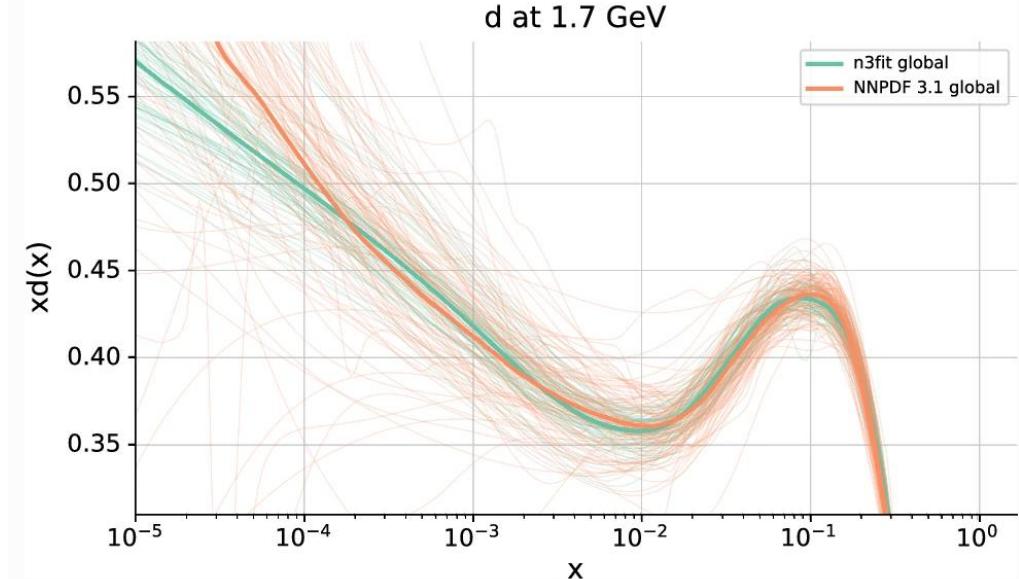
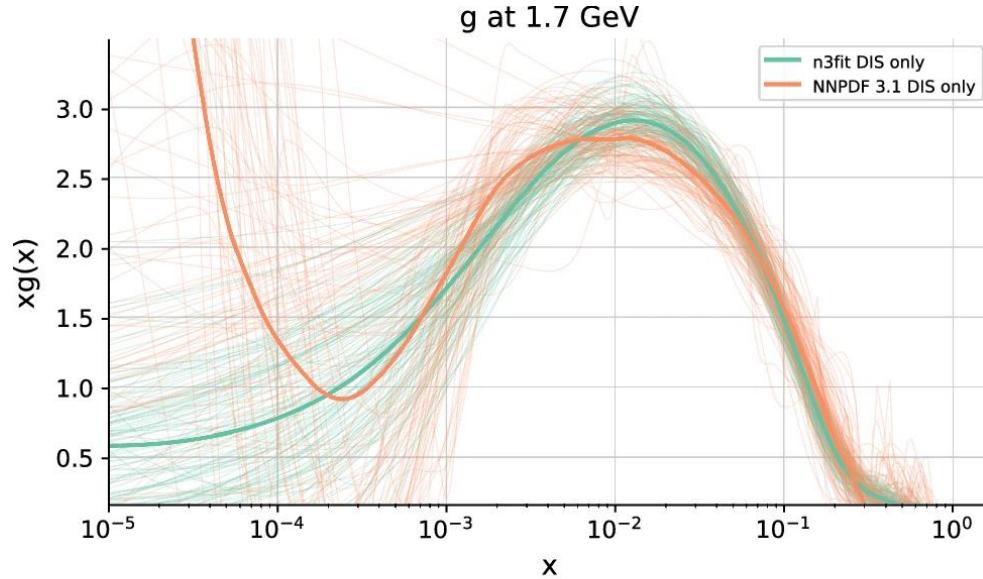
Fig. 10. Graphical representation of a hyperparameter scan for a DIS only fit with 2000 trials (from Ref.^[42]). The loss function presented in the y-axis is an average of the validation and testing χ^2 . The shape of the violin plots represent a visual aid on the behavior of the fit as a function of the free parameter. Fatter plots represent better stability, i.e., configurations which are less likely to produce outliers.

NNPDF 3.1

Overlearning

- The cross-validation it is not enough to prevent overfitting due to correlations within points in a same dataset when using "hyperopt" with n3fit.
- To eliminate architectures that allowed overlearning it was included a testing set where the model generalization power is tested.
- This is a set of datasets where none of the points are used in the fitting either for training or validation.
- The test set is defined by removing from the training set datasets with duplicate process type and smaller leading-order kinematic range coverage.
- The "testing loss" and is used as a third criterion to discard combinations of hyperparameters.





Thanks for your attention!



Backup



By isospin symmetry $SU(2)$, we have :

$$f_u^{(p)} = f_d^{(n)} \equiv u$$

$$f_d^{(p)} = f_u^{(n)} \equiv d$$

$$f_s^{(p)} = f_s^{(n)} \equiv s$$

Moreover, the partons distribution function should respect the sum rules, corresponding to the conservation of quantum number :

- Charge conservation:

$$Q = \int_0^1 dx \left[\frac{2}{3}(u - \bar{u}) - \frac{1}{3}(d - \bar{d}) \right] = 1 \quad \text{for proton}$$

$$Q = \int_0^1 dx \left[\frac{2}{3}(d - \bar{d}) - \frac{1}{3}(u - \bar{u}) \right] = 0 \quad \text{for neutron}$$

- Isospin number :

$$I_3 = \frac{1}{2} \int_0^1 dx [(u - \bar{u}) - (d - \bar{d})] = \frac{1}{2} \quad \text{for proton}$$

$$I_3 = \frac{1}{2} \int_0^1 dx [(d - \bar{d}) - (u - \bar{u})] = -\frac{1}{2} \quad \text{for neutron}$$

- Strangeness :

$$S = \int_0^1 dx (s - \bar{s}) = 0$$

- Baryonic number :

$$B = \frac{1}{3} \int [u - \bar{u} + d - \bar{d} + s - \bar{s}] = 1$$

The sum rule on B can be obtained using the relation $Q = I_3 + \frac{Y}{2}$ with $Y = B + S$.

Finally, the last sum rule to consider is energy-momentum conservation for proton and neutron:

$$\int_0^1 dx x [u(x, Q^2) + d(x, Q^2) + s(x, Q^2) + \bar{u}(x, Q^2) + \bar{d}(x, Q^2) + \bar{s}(x, Q^2) + g(x, Q^2)] = 1,$$

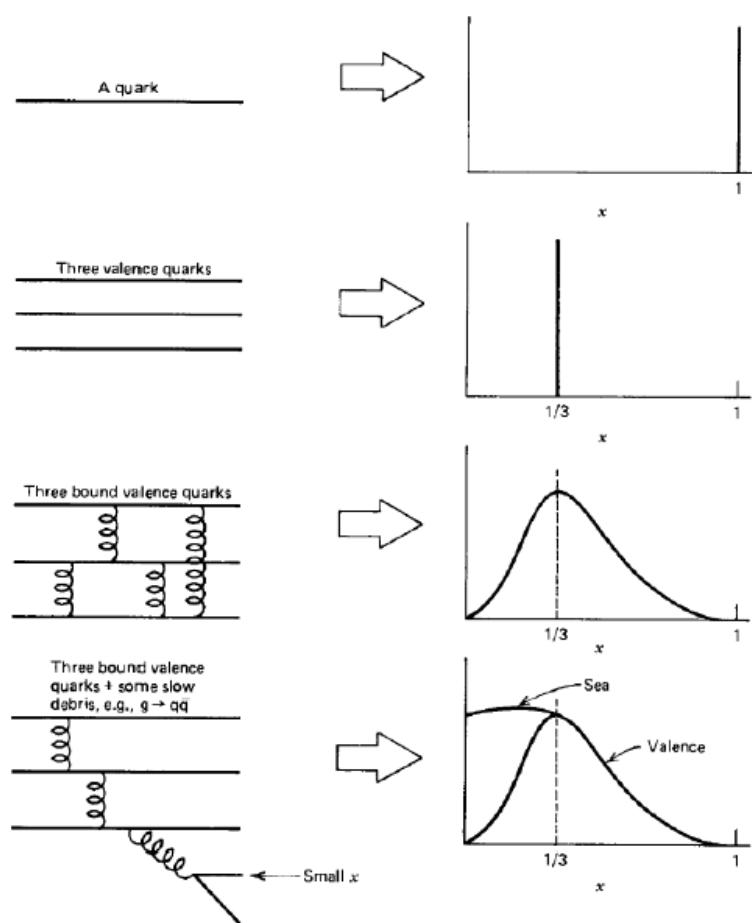


Figure 3.8: Qualitative shape of F_2 depending on the model of the proton structure. Figure from [20]

Table 18.1: The main processes relevant to global PDF analyses, ordered in three groups: fixed-target experiments, HERA and the $p\bar{p}$ Tevatron / pp LHC. For each process we give an indication of their dominant partonic subprocesses, the primary partons which are probed and the approximate range of x constrained by the data.

Process	Subprocess	Partons	x range
$\ell^\pm \{p, n\} \rightarrow \ell^\pm X$	$\gamma^* q \rightarrow q$	q, \bar{q}, g	$x \gtrsim 0.01$
$\ell^\pm n/p \rightarrow \ell^\pm X$	$\gamma^* d/u \rightarrow d/u$	d/u	$x \gtrsim 0.01$
$pp \rightarrow \mu^+ \mu^- X$	$u\bar{u}, d\bar{d} \rightarrow \gamma^*$	\bar{q}	$0.015 \lesssim x \lesssim 0.35$
$pn/pp \rightarrow \mu^+ \mu^- X$	$(u\bar{d})/(u\bar{u}) \rightarrow \gamma^*$	\bar{d}/\bar{u}	$0.015 \lesssim x \lesssim 0.35$
$\nu(\bar{\nu}) N \rightarrow \mu^-(\mu^+) X$	$W^* q \rightarrow q'$	q, \bar{q}	$0.01 \lesssim x \lesssim 0.5$
$\nu N \rightarrow \mu^- \mu^+ X$	$W^* s \rightarrow c$	s	$0.01 \lesssim x \lesssim 0.2$
$\bar{\nu} N \rightarrow \mu^+ \mu^- X$	$W^* \bar{s} \rightarrow \bar{c}$	\bar{s}	$0.01 \lesssim x \lesssim 0.2$
<hr/>			
$e^\pm p \rightarrow e^\pm X$	$\gamma^* q \rightarrow q$	g, q, \bar{q}	$10^{-4} \lesssim x \lesssim 0.1$
$e^+ p \rightarrow \bar{\nu} X$	$W^+ \{d, s\} \rightarrow \{u, c\}$	d, s	$x \gtrsim 0.01$
$e^\pm p \rightarrow e^\pm c\bar{c}X, e^\pm b\bar{b}X$	$\gamma^* c \rightarrow c, \gamma^* g \rightarrow c\bar{c}$	c, b, g	$10^{-4} \lesssim x \lesssim 0.01$
$e^\pm p \rightarrow \text{jet}+X$	$\gamma^* g \rightarrow q\bar{q}$	g	$0.01 \lesssim x \lesssim 0.1$
<hr/>			
$p\bar{p}, pp \rightarrow \text{jet}+X$	$gg, qg, qq \rightarrow 2j$	g, q	$0.00005 \lesssim x \lesssim 0.5$
$p\bar{p} \rightarrow (W^\pm \rightarrow \ell^\pm \nu) X$	$ud \rightarrow W^+, \bar{u}\bar{d} \rightarrow W^-$	u, d, \bar{u}, \bar{d}	$x \gtrsim 0.05$
$pp \rightarrow (W^\pm \rightarrow \ell^\pm \nu) X$	$u\bar{d} \rightarrow W^+, d\bar{u} \rightarrow W^-$	$u, d, \bar{u}, \bar{d}, g$	$x \gtrsim 0.001$
$pp(p\bar{p}) \rightarrow (Z \rightarrow \ell^+\ell^-)X$	$uu, dd, ..(u\bar{u}, ..) \rightarrow Z$	$u, d, ..(g)$	$x \gtrsim 0.001$
$pp \rightarrow W^- c, W^+ \bar{c}$	$gs \rightarrow W^- c$	s, \bar{s}	$x \sim 0.01$
$pp \rightarrow (\gamma^* \rightarrow \ell^+\ell^-)X$	$u\bar{u}, d\bar{d}, .. \rightarrow \gamma^*$	\bar{q}, g	$x \gtrsim 10^{-5}$
$pp \rightarrow (\gamma^* \rightarrow \ell^+\ell^-)X$	$u\gamma, d\gamma, .. \rightarrow \gamma^*$	γ	$x \gtrsim 10^{-2}$
$pp \rightarrow b\bar{b} X, t\bar{t} X$	$gg \rightarrow b\bar{b}, t\bar{t}$	g	$x \gtrsim 10^{-5}, 10^{-2}$
$pp \rightarrow \text{exclusive } J/\psi, \Upsilon$	$\gamma^*(gg) \rightarrow J/\psi, \Upsilon$	g	$x \gtrsim 10^{-5}, 10^{-4}$
$pp \rightarrow \gamma X$	$gq \rightarrow \gamma q, g\bar{q} \rightarrow \gamma\bar{q}$	g	$x \gtrsim 0.005$

Backup

For QCD, since gluons also emit additional gluons and similar charges attract themselves, the effect is anti-screening, the effective color charge decreases as one probes the original quark. Therefore, this phenomena of asymptotic freedom is given by the QCD running coupling,

$$\alpha_s(Q^2) = \frac{\alpha_s(\mu^2)}{1 + \frac{\alpha_s(\mu^2)}{12\pi}(11N_c - 2N_f) \log\left(\frac{Q^2}{\mu^2}\right)}$$

where N_c is the number of color charges, N_f the number of quarks flavors and μ the renormalization scale. To make α_s independent of the renormalization scheme, we introduce a scale Λ_{QCD}^2 ,

$$\Lambda_{QCD}^2 = \mu^2 e^{\frac{-12\pi}{(11N_c - 2N_f)\alpha_s(\mu)}},$$

which experimentally is ~ 200 . In this QCD scale, the coupling constant can be written as

$$\alpha_s(Q^2) = \frac{12\pi}{(11N_c - 2N_f)\log(Q^2/\Lambda_{QCD}^2)},$$

where we see $\alpha_s \rightarrow 0$ when $Q^2 \rightarrow \infty$ and when $Q^2 \gg \Lambda_{QCD}^2$ we have hard interactions.

$$\frac{\partial f_a}{\partial \ln \mu^2} = \frac{\alpha_s(\mu^2)}{2\pi} \sum_b P_{ab} \otimes f_b$$

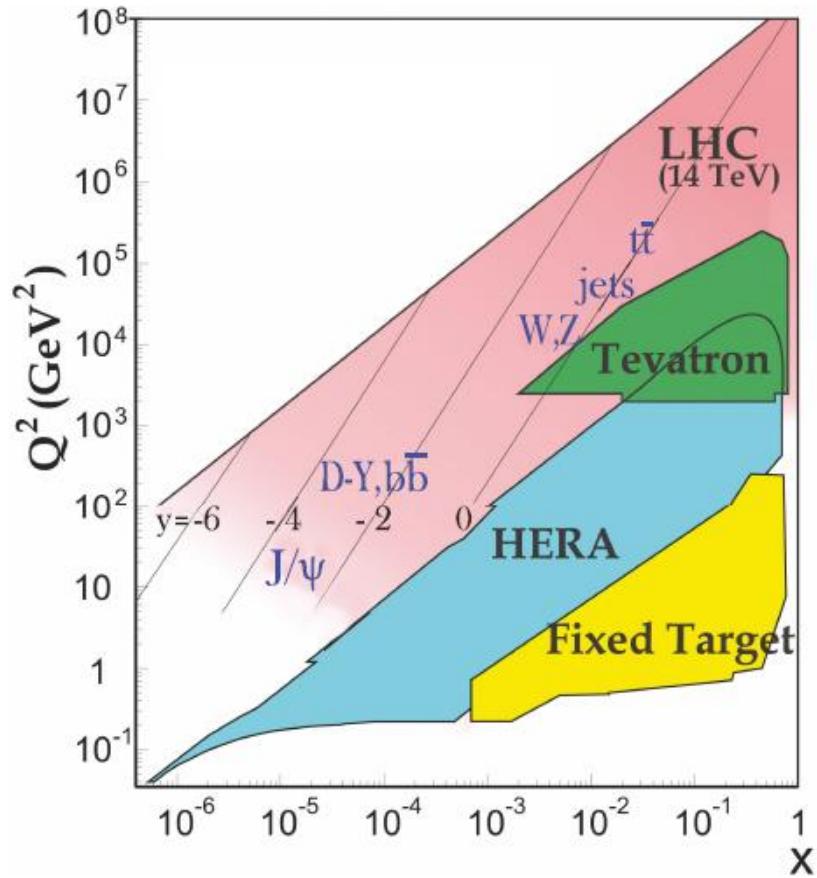


Figure 18.3: Kinematic domains in x and Q^2 probed by fixed-target and collider experiments, where here Q^2 can refer either the literal Q^2 for deep inelastic scattering, or the hard scale of the process in hadron-hadron collisions, e.g. invariant mass or transverse momentum p_T^2 . Some of the final states accessible at the LHC are indicated in the appropriate regions, where y is the rapidity. The incoming partons have $x_{1,2} = (Q/14 \text{ TeV})e^{\pm y}$ where Q is the hard scale of the process shown in blue in the figure. For example, open charm production [23] and exclusive J/ψ and Υ production [24] at high $|y|$ at the LHC may probe the gluon PDF down to $x \sim 10^{-5}$.

Backup

PDFs were found under the assumptions:

- No transvers momenta is carried by the parton
- Partons are massless
- Partons are pointlike
- Polarization of the targeted proton is fixed
- Targeted proton cannot be scattered in another baryon (remember the Δ resonance found before)

How do we improve the model? Using Generalized Parton Distributions (GPDs)

Backup

In a DIS process with transversal momenta the Bjorken x is not the only dimensionless quantity that we can construct. There is also:

$$\xi = -\frac{\Delta \cdot n}{P \cdot n}$$

Where $\Delta = p' - p$ and $P = \frac{p+p'}{2}$, n is a light-like 4-vector. A new phenomena can arise with the virtual photon comes onshell and it results in $ep \rightarrow epy$ (DVCS). (It is only possible with transvers momenta)

M. Diehl / Physics Reports 388 (2003) 41–277

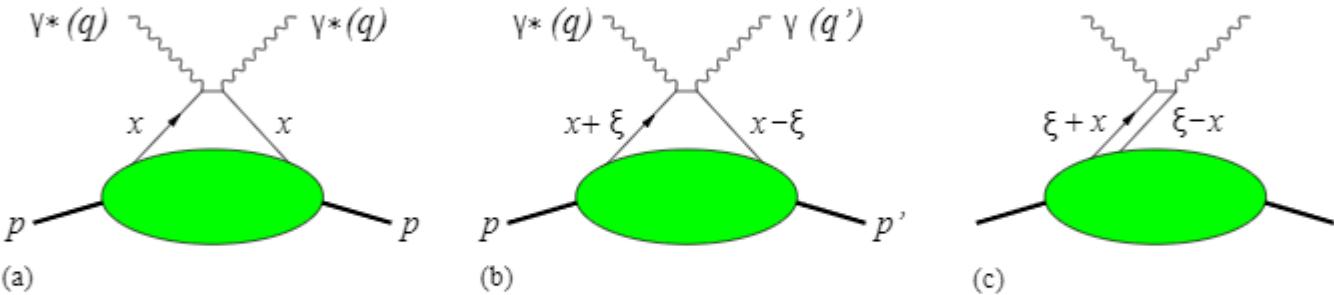


Fig. 1. (a) Handbag diagram for the forward Compton amplitude $\gamma^* p \rightarrow \gamma^* p$, whose imaginary part gives the DIS cross section. (b) Handbag diagram for DVCS in the region $\xi < x < 1$. (c) The same in the region $-\xi < x < \xi$. Momentum fractions x and ξ refer to the average hadron momentum $\frac{1}{2}(p + p')$. A second diagram is obtained in each case by interchanging the photon vertices.

Shown in Fig. 1c a new phenomena is presented: instead of a parton being emitted and reabsorbed by the target, we have emission of a quark-antiquark (or gluon) pair. GPDs are “the yellow box in figure”.

References

- [1] P.G. Ratcliffe, An Introduction to Elementary Particle Phenomenology, Chapter 3, <http://dx.doi.org/10.1088/978-0-7503-1072-7ch3>
- [2] G. Altarelli and G. Parisi, Nucl. Phys. B126, 298 (1977), [https://doi.org/10.1016/0550-3213\(77\)90384-4](https://doi.org/10.1016/0550-3213(77)90384-4)
- [3] R. G. Roberts, The Structure of the proton: Deep inelastic scattering. Cambridge University Press, 1990.
- [4] Kevin Gurney, 1997, An Introduction to Neural Networks, Taylor & Francis, Inc., USA.
- [5] <https://n3pdf.github.io/eko/theory/FlavorSpace.html#evolution-basis>
- [6] <http://nnpdf.mi.infn.it/wp-content/uploads/2017/10/Forte-Orsay.pdf>
- [7] [DGLAP.pdf \(sunysb.edu\)](#)
- [8] [Emilie-Li-Dissertation.pdf \(imperial.ac.uk\)](#)
- [9] M. Diehl, Generalized parton distributions, Physics Reports, Volume 388, Issues 2-4, 2003, Pages 41-277, ISSN 0370-1573, <https://doi.org/10.1016/j.physrep.2003.08.002>
- [10] Parton distribution functions, Stefano Forte and Stefano Carrazza, 2020, [\[2008.12305\] Parton distribution functions \(arxiv.org\)](#)
- [11] Carrazza, S., Cruz-Martinez, J. Towards a new generation of parton densities with deep learning models. *Eur. Phys. J. C* **79**, 676 (2019). <https://doi.org/10.1140/epjc/s10052-019-7197-2>