The neural network approach to parton fitting

Andrea Piccione

SLAC - Feb, 16th 2006

The NNPDF Collaboration

Luigi Del Debbio[1], Stefano Forte[2],
José I. Latorre[3], A. P.[4] and Joan Rojo[3]

[1] Particle Physics Theory Group, School of Physics, University of Edinburgh
[2] Dipartimento di Fisica, Università di Milano and INFN, Sezione di Milano
[3] Departament d'Estructura i Constituents de la Matèria, Universitat de Barcelona
[4] Dipartimento di Fisica Teorica, Università di Torino and INFN, Sezione di Torino

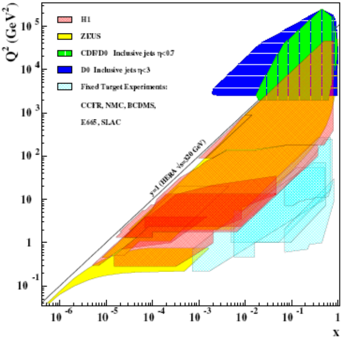
How do we describe hadrons?

- QCD describes interactions between quarks and gluons. Experimentally we observe only hadrons $\rightarrow$ Confinement
- Perturbative QCD is not trustable at low energies ($\sim$ GeV). We can not solve QCD in the non-perturbative region, but on a lattice ...
- We can extract information on the proton structure from a process with only one initial proton (DIS at HERA). Then we can use these as an input for a process where two initial protons are involved (DY at LHC) $\rightarrow$ Factorization

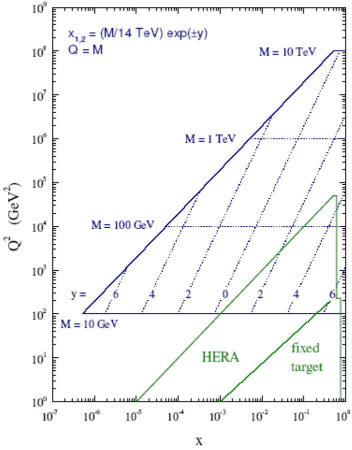
How do we describe hadrons?

- QCD describes interactions between quarks and gluons. Experimentally we observe only hadrons $\rightarrow$ Confinement
- Perturbative QCD is not trustable at low energies ($\sim$ GeV). We can not solve QCD in the non-perturbative region, but on a lattice . . .
- We can extract information on the proton structure from a process with only one initial proton (DIS at HERA). Then we can use these as an input for a process where two initial protons are involved (DY at LHC) $\rightarrow$ Factorization

Kinematics



H1
ZEUS
CDF/D0 Inclusive jets η<0.7
D0 Inclusive jets η<3
Fixed Target Experiments:
CCFR, NMC, BCDMS,
E665, SLAC
y=1 (HERA √s=320 GeV)
Q² (GeV²)
x



x1,2 = (M/14 TeV) exp(±y)
Q = M
M = 10 TeV
M = 1 TeV
M = 100 GeV
M = 10 GeV
y = 6 4 2 0 2 4 6
HERA
fixed target
Q² (GeV²)
x

Deep Inelastic Scattering

- The cross section

$$\frac{d^2\sigma}{dxdQ^2} = \frac{4\pi\alpha^2}{Q^4}\left[[1+(1-y)^2]F_1 + \frac{1-y}{x}(F_2 - 2xF_1)\right]$$

- The structure function

$$F_2(x,\ Q^2) = x\left[\sum_{q=1}^{n_f} e_q^2\, C^q \otimes q_q(x,\ Q^2) + 2n_f\, C^g \otimes g(x,\ Q^2)\right]$$

- Parton distribution evlution is described by DGLAP equations

$$Q^2\frac{d}{dQ^2}q(x,Q^2) = \frac{\alpha_s(Q^2)}{2\pi}(P\otimes q)(x,Q^2)$$

Deep Inelastic Scattering and QCD

- The cross section

$$\frac{d^2\sigma}{dxdQ^2} = \frac{4\pi\alpha^2}{Q^4}\left[[1+(1-y)^2]F_1 + \frac{1-y}{x}(F_2 - 2xF_1)\right]$$

- The structure function

$$F_2(x,\ Q^2) = x\left[\sum_{q=1}^{n_f} e_q^2\,\mathcal{C}^q \otimes q_q(x,\ Q^2) + 2n_f\,\mathcal{C}^g \otimes g(x,\ Q^2)\right]$$

- Parton distribution evlution is described by DGLAP equations

$$Q^2\frac{d}{dQ^2}q(x, Q^2) = \frac{\alpha_s(Q^2)}{2\pi}(P \otimes q)(x, Q^2)$$

Deep Inelastic Scattering and QCD

- The cross section

$$\frac{d^2\sigma}{dxdQ^2} = \frac{4\pi\alpha^2}{Q^4}\left[[1+(1-y)^2]F_1 + \frac{1-y}{x}(F_2 - 2xF_1)\right]$$

- The structure function

$$F_2(x,\ Q^2) = x\left[\sum_{q=1}^{n_f} e_q^2\, \mathcal{C}^q \otimes q_q(x,\, Q^2) + 2n_f\, \mathcal{C}^g \otimes g(x,\, Q^2)\right]$$

- Parton distribution evlution is described by DGLAP equations

$$Q^2 \frac{d}{dQ^2} q(x,\, Q^2) = \frac{\alpha_s(Q^2)}{2\pi}(P \otimes q)(x,\, Q^2)$$

The problem

- For a single quantity $\rightarrow$ 1 sigma error
- For a pair of numbers $\rightarrow$ 1 sigma ellipse
- For a function $\rightarrow$ We need an "error band" in the space of functions (*i.e.* the probability density $\mathcal{P}[f]$ in the space of functions $f(x)$)

Expectation values $\rightarrow$ Functional integrals

$$\langle \mathcal{F}[f(x)] \rangle = \int \mathcal{D}f \, \mathcal{F}[f(x)] \, \mathcal{P}[f(x)]$$

Determine an infinite-dimensional object (a function) from finite set of data points $\rightarrow$ Mathematically ill-posed problem

The standard approach

1. Choose a simple functional form with enough free parameters

$$q(x, Q_0^2) = x^\alpha (1 - x)^\beta P(x; \lambda_1, \ldots, \lambda_n)$$

2. Fit parameters by minimizing χ^2

Open problems:

- Error propagation from data to parameters and from parameters to observables is not trivial
- Theoretical bias due to the choice of a parametrization is difficult to assess (effects can be large if data are not precise or hardly compatible)

The standard approach

1. Choose a simple functional form with enough free parameters

$$q(x, Q_0^2) = x^{\alpha}(1-x)^{\beta} P(x; \lambda_1, \ldots, \lambda_n)$$

2. Fit parameters by minimizing χ^2

Open problems:

- Error propagation from data to parameters and from parameters to observables is not trivial
- Theoretical bias due to the choice of a parametrization is difficult to assess (effects can be large if data are not precise or hardly compatible)

The standard approach

1. Choose a simple functional form with enough free parameters

$$q(x, Q_0^2) = x^\alpha (1-x)^\beta P(x; \lambda_1, \dots, \lambda_n)$$

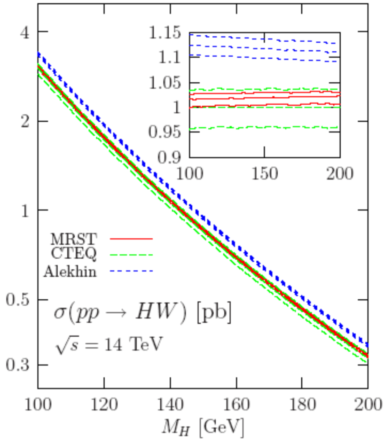
2. Fit parameters by minimizing χ^2

Open problems:

- Error propagation from data to parameters and from parameters to observables is not trivial
- Theoretical bias due to the choice of a parametrization is difficult to assess (effects can be large if data are not precise or hardly compatible)

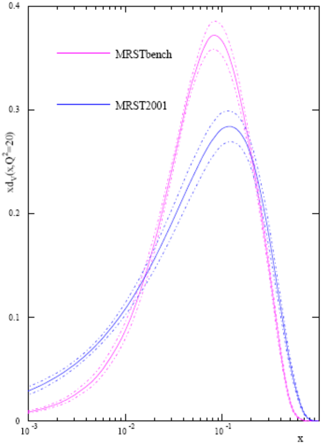
The standard approach - Limitations

[A. Djouadi and S. Ferrag, hep-ph/0310209]



The standard approach - Limitations

[R. S. Thorne, hep-ph/0511119]



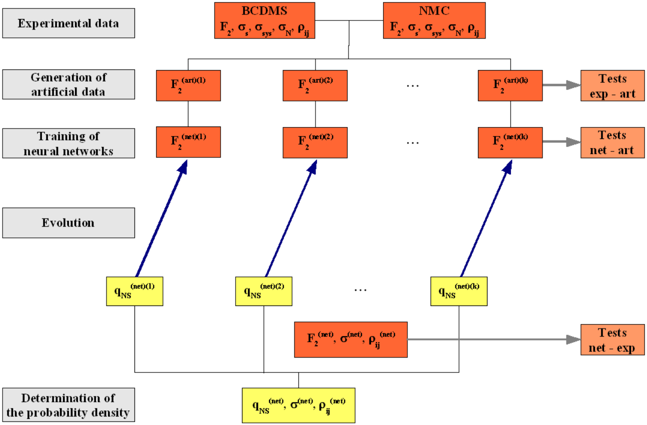
The Bayesian Monte Carlo approach

[W. T. Giele, S. A. Keller, D. A. Kosower, hep-ph/0104052]

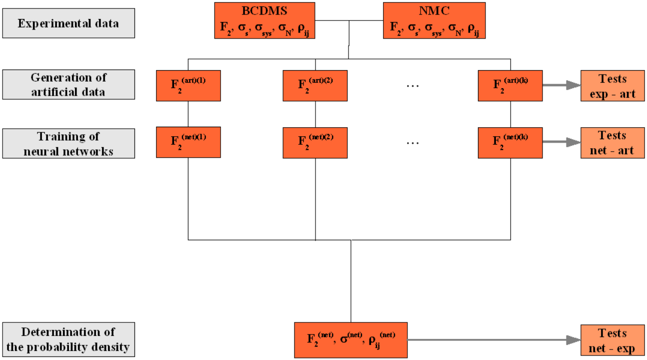
- Generate a Monte-Carlo sample of functions with a "reasonable" prior distribution.
- Calculate observables with functional integral.
- Update probability using Bayesian inference on MC sample.
- Iterate until convergence achieved.

The problem is made finite-dimensional by the choice of a prior, but the result does not depend on the choice if sufficiently general.

The NNPDF approach

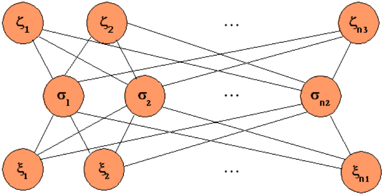


The NNPDF approach



What are Neural Networks?

Neural networks are a class of algorithms very suitable to fit incomplete or noisy data [for HEP applications see ACAT 2005]



Any continuous function can be uniformly approximated by a continuous neural network having only one internal layer, and with an arbitrary continuous sigmoid non-linearity [G. Cybenko (1989)].

Some details on their structure

- Building blocks: neurons, *i. e.* input/output units characterized by sigmoid activation

$$\xi_i^{(l)} = g\left(\sum_{j=1}^{n_l - 1} \omega_{ij}^{(l-1)} \xi_j^{(l-1)} - \theta_i^{(l)}\right) \qquad g(x) = \frac{1}{1+e^{-x}}$$

- In a simple case (1-2-1) we have,

$$\xi_1^{(3)} = \frac{1}{1 + e^{\theta_1^{(3)} - \frac{\omega_{11}^{(2)}}{1+e^{\theta_1^{(2)} - \xi_1^{(1)} \omega_{11}^{(1)}}} - \frac{\omega_{12}^{(2)}}{1+e^{\theta_2^{(2)} - \xi_1^{(1)} \omega_{21}^{(1)}}}}}$$

Some details on their structure

- Building blocks: neurons, *i. e.* input/output units characterized by sigmoid activation

$$\xi_i^{(l)} = g\left(\sum_{j=1}^{n_l-1} \omega_{ij}^{(l-1)} \xi_j^{(l-1)} - \theta_i^{(l)}\right) \qquad g(x) = \frac{1}{1+e^{-x}}$$

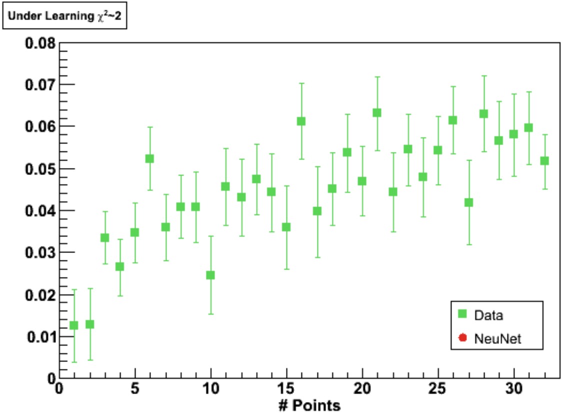
- In a simple case (1-2-1) we have,

$$\xi_1^{(3)} = \frac{1}{1 + e^{\theta_1^{(3)} - \frac{\omega_{11}^{(2)}}{1+e^{\theta_1^{(2)} - \xi_1^{(1)}\omega_{11}^{(1)}}} - \frac{\omega_{12}^{(2)}}{1+e^{\theta_2^{(2)} - \xi_1^{(1)}\omega_{21}^{(1)}}}}}$$

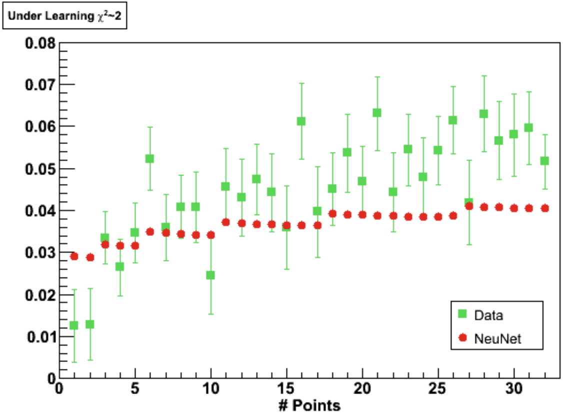
General features

1. Set the parameters randomly.
2. If there are different inputs, normalize them.
3. Define a figure of merit *E* (say χ^2).
4. Define a criterium of convergence (say $\chi^2 \sim 1$).

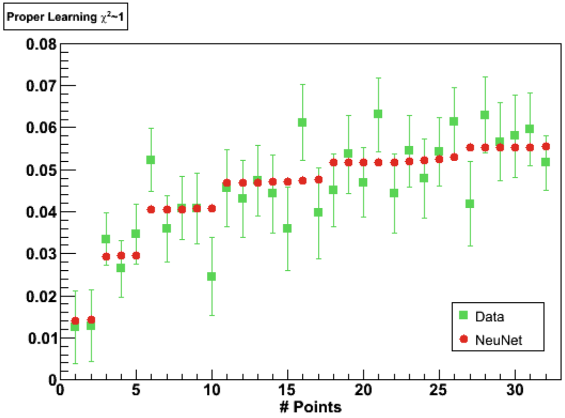
Learning of data



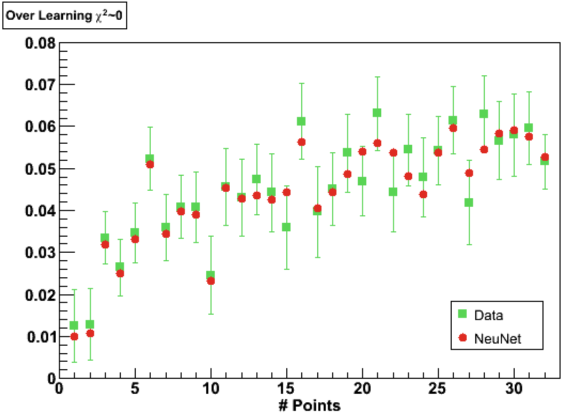
Learning of data



Learning of data



Learning of data



Back Propagation

1. Present an input and calculate the output.
2. Evaluate E.
3. Modify the weights to reinforce correct decisions and discourage incorrect ones:

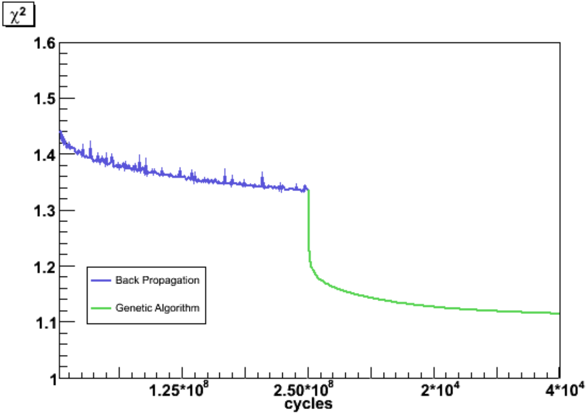
   $$\omega_{ij} \to \omega_{ij} - \eta \frac{\partial E}{\partial \omega_{ij}}$$

   where η is the learning rate.
4. Back to 1, till the stability of E is reached.

Genetic Algorithm

1. Make clones of the set of parameters.
2. Mutate each clone.
3. Evaluate E for all the clones.
4. Select the clone that has the lowest E.
5. Back to 1, till the stability of E is reached.

BP *vs.* GA



χ²
1.6
1.5
1.4
1.3
1.2
1.1
1
Back Propagation
Genetic Algorithm
1.25*10^8
2.50*10^8
2*10^4
4*10^4
cycles

Faithful error propagation: Data $\longrightarrow$ Parametrization

- Monte Carlo sampling of data (generation of replicas of experimental data)

$$F_i^{(art)(k)} = \left(1 + r_N^{(k)}\sigma_N\right)\left[F_i^{(exp)} + r_i^s \sigma_i^{stat} + \sum_{l=1}^{N_{sys}} r^{l,(k)} \sigma_i^{sys,l}\right]$$

where σ_i are the experimantal errors, and r_i are random numbers choosen accordingly to the experimental correlation matrix.

Faithful error propagation: Parametrization → Observables

- Expectation values:

$$\langle \mathcal{F}\left[g(x)\right]\rangle = \frac{1}{N_{rep}} \sum_{k=1}^{N_{rep}} \mathcal{F}\left(g^{(net)(k)}(x)\right)$$

- Errors:

$$\sigma_{\mathcal{F}[g(x)]} = \sqrt{\left\langle \mathcal{F}\left[g(x)\right]^2\right\rangle - \left\langle \mathcal{F}\left[g(x)\right]\right\rangle^2}$$

- Correlations between pairs of different parton distributions at different points:

$$\langle u(x_1) d(x_2)\rangle = \frac{1}{N_{rep}} \sum_{k=1}^{N_{rep}} u^{(net)(k)}(x_1, Q_0^2)\, d^{(net)(k)}(x_2, Q_0^2)$$

Unbiased parametrization

- A Neural Network is trained over each MC replica.
- Neural Network architecture: 2-5-3-1 (37 parameters).
- $x\,q(x, Q_0^2) = NN(x, \log x)(1-x)^a$:
  - we could use as inputs also, say, my age or the atm. pressure;
  - at $x = 1$ we have a kinematical constraint.

Unbiased parametrization

- A Neural Network is trained over each MC replica.
- Neural Network architecture: 2-5-3-1 (37 parameters).
- $x\, q(x, Q_0^2) = NN(x, \log x)(1-x)^a$:
  - we could use as inputs also, say, my age or the atm. pressure;
  - at $x = 1$ we have a kinematical constraint.

PDF Evolution

- We want Mellin space evolution (numerically efficient):

$$q(N, Q^2) = q(N, Q_0^2)\Gamma\left(N, \alpha_s\left(Q^2\right), \alpha_s\left(Q_0^2\right)\right)$$

- We do not want complex neural networks:

$$\Gamma\left(x, \alpha_s\left(Q^2\right), \alpha_s\left(Q_0^2\right)\right) \equiv \frac{1}{2\pi i}\int_{c-i\infty}^{c+i\infty} dN\, x^{-N}\Gamma\left(N, \alpha_s\left(Q^2\right), \alpha_s\left(Q_0^2\right)\right)$$

- $\Gamma(x)$ is a distribution $\longrightarrow$ must be regularized at $x = 1$:

$$q(x, Q^2) = q(x, Q_0^2)\int_x^1 dy\, \Gamma(y) + \int_x^1 \frac{dy}{y}\Gamma(y)\left(q\left(\frac{x}{y}, Q_0^2\right) - yq(x, Q_0^2)\right)$$

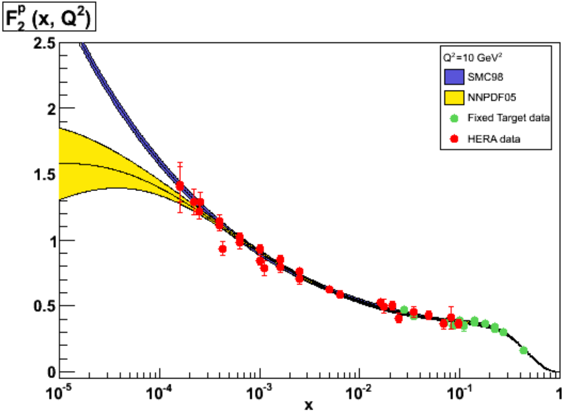
References

- S. Forte, L. Garrido, J. I. Latorre and A. P., *"Neural network parametrization of deep-inelastic structure functions,"* JHEP05 (2002) 062 [arXiv:hep-ph/0204232]
- L. Del Debbio, S. Forte, J. I. Latorre, A. P. and J. Rojo [NNPDF Collaboration], *"Unbiased determination of the proton structure function F_2^p with faithful uncertainty estimation"*, JHEP03 (2005) 080 [arXiv:hep-ph/0501067]

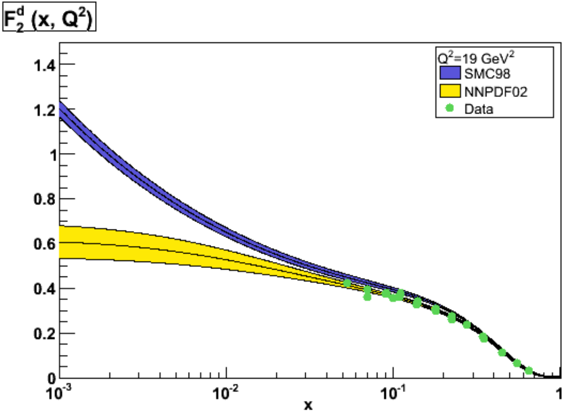
Source code, driver program and graphical web interface for F_2 plots and numerical computations available @

http://sophia.ecm.ub.es/f2neural

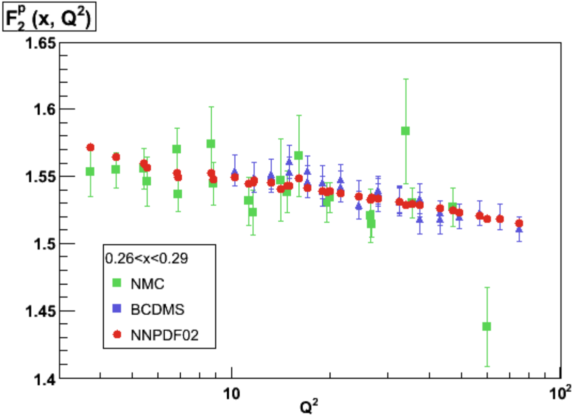
Fit of $F_2^p(x, Q^2)$ *[NNPDF 2005]*



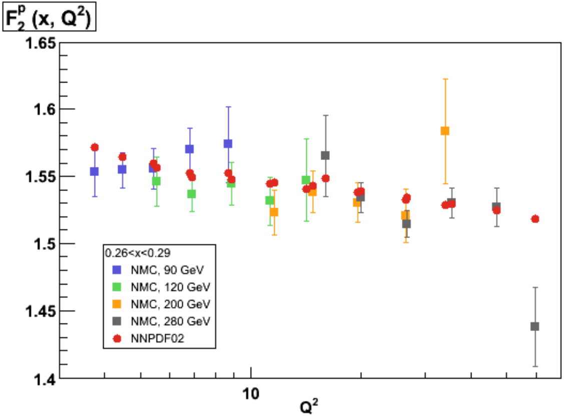
Fit of $F_2^d(x, Q^2)$ *[NNPDF 2002]*



Incompatible data *[NNPDF 2002]*



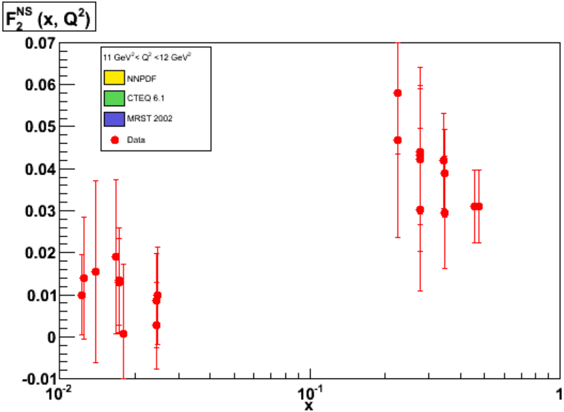
Incompatible data *[NNPDF 2002]*



Details

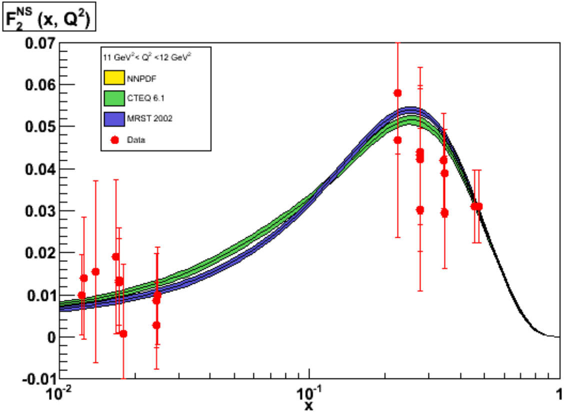
- Experimental data: NMC (229 pts) and BCDMS (254 pts)
- Kinematical cuts: $Q^2 \geq 3\ GeV^2$, $W^2 \geq 6.25\ GeV^2$
- Strong coupling: $\alpha_s\left(M_Z^2\right) = 0.1182$
- Perturbative order: NLO
- VFN: $m_c = 1.5 GeV$, $m_b = 4.5 GeV$, $m_t = 175 GeV$
- TMC: F_2 integral evaluated with NN F_2

Non-Singlet



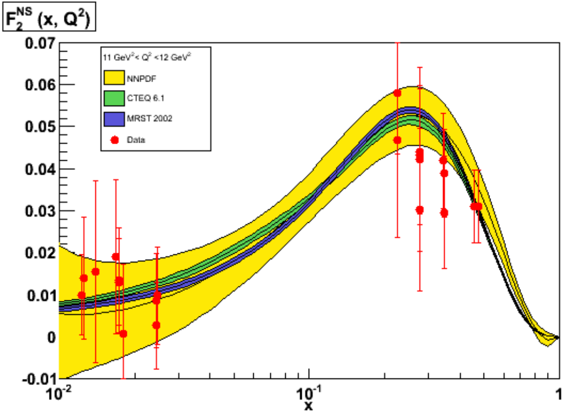
F_2^{NS} (x, Q^2)
11 GeV^2< Q^2 <12 GeV^2
NNPDF
CTEQ 6.1
MRST 2002
Data
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
-0.01
10^{-2}
10^{-1}
1
x

Non-Singlet



F_2^{NS} (x, Q²)
11 GeV² < Q² < 12 GeV²
NNPDF
CTEQ 6.1
MRST 2002
Data
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
-0.01
10⁻²
10⁻¹
1
x

Non-Singlet



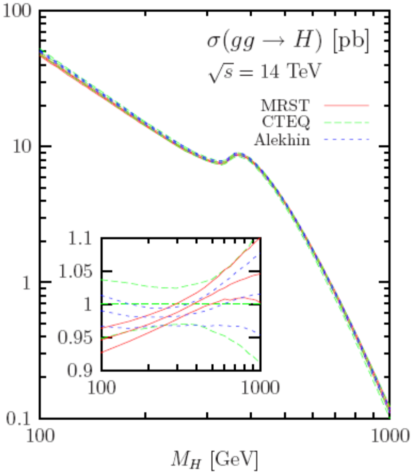
$F_2^{NS}(x, Q^2)$
11 GeV² < Q² < 12 GeV²
NNPDF
CTEQ 6.1
MRST 2002
Data
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0
-0.01
10^{-2}
10^{-1}
1
x

Perspectives

- Construct full set of NNPDF parton distributions from all available data
- Assess impact of uncertainties of PDFs for relevant observables at LHC
- Make formalism compatible with standard interfaces (LHAPDF, PDFLIB) $\rightarrow$ NNPDF partons available for use in Monte Carlo generators

Extra: The standard approach - Limitations

[A. Djouadi and S. Ferrag, hep-ph/0310209]



Extra: The standard approach

- MRST: 15 parms. - $\Delta\chi^2 = 50$ - NC and CC DIS, DY, W-asym, jets

$$xq(x, Q_0^2) = A(1-x)^{\eta}(1+\epsilon x^{0.5}+\gamma x)x^{\delta}, \quad x[\bar{u}-\bar{d}](x, Q_0^2) = A(1-x)^{\eta}(1+\gamma x+\delta x^2)x^{\delta}.$$

$$xg(x, Q_0^2) = A_g(1-x)^{\eta_g}(1+\epsilon_g x^{0.5}+\gamma_g x)x^{\delta_g} - A_-(1-x)^{\eta_-}x^{-\delta_-},$$

- CTEQ: 20 parms. - $\Delta\chi^2 = 100$ - NC and CC DIS, DY, W-asym, jets

$$x\,f(x, Q_0) = A_0\,x^{A_1}\,(1-x)^{A_2}\,e^{A_3 x}\,(1+e^{A_4}\,x)^{A_5}$$

with independent params for combinations $u_v \equiv u-\bar{u}$, $d_v \equiv d-\bar{d}$, g, and $\bar{u}+\bar{d}$, $s=\bar{s}=0.2\,(\bar{u}+\bar{d})$ at Q_0; norm. fixed by sum rules

- Alekhin: 17 parms. - $\Delta\chi^2 = 1$ - NC DIS (+ DY)

$$xu_V(x, Q_0) = \frac{2}{N_u^V}x^{a_u}(1-x)^{b_u}(1+\gamma_2^u x); \qquad xu_S(x, Q_0) = \frac{A_S}{N_S}\eta_u x^{a_s}(1-x)^{b_{su}}$$

$$xd_V(x, Q_0) = \frac{1}{N_d^V}x^{a_d}(1-x)^{b_d}; \qquad xd_S(x, Q_0) = \frac{A_S}{N^S}x^{a_s}(1-x)^{b_{sd}},$$

$$xs_S(x, Q_0) = \frac{A_S}{N^S}\eta_s x^{a_s}(1-x)^{(b_{su}+b_{sd})/2}; \qquad xG(x, Q_0) = A_G x^{a_G}(1-x)^{b_G}(1+\gamma_1^G\sqrt{x}+\gamma_2^G x),$$

Extra: SF Details

Architecture: 4-5-3-1

- Inputs: x, $\log x$, Q^2, $\log Q^2$
- Output: $F_2(x, Q^2)$

Minimization strategy:

- Back Propagation ($\sim 10^8$ training cycles):

$$\chi^{2\,(k)}_{\mathrm{diag}} = \frac{1}{N_{\mathrm{dat}}} \sum_{i=1}^{N_{\mathrm{dat}}} \frac{\left(F_i^{(\mathrm{art})(k)} - F_i^{(\mathrm{net})(k)}\right)^2}{\sigma_{i,t}^{(\mathrm{exp})^2}}$$

- Genetic Algorithm ($\sim 10^4$ generations):

$$\chi^{2\,(k)} = \frac{1}{N_{\mathrm{dat}}} \sum_{i,j=1}^{N_{\mathrm{dat}}} \left(F_i^{(\mathrm{art})(k)} - F_i^{(\mathrm{net})(k)}\right) \mathrm{cov}_{ij}^{-1} \left(F_j^{(\mathrm{art})(k)} - F_j^{(\mathrm{net})(k)}\right)$$

PDF: Extras

Mellin Inversion with the Fixed Talbot algorithm:

$$f(t) = \frac{1}{2\pi i}\int_C ds\ e^{ts}\tilde{f}(s), \quad t = -\ln x$$

$$s(\theta) = r\theta\left(\cot\theta + i\right), \quad -\pi \le \theta \le \pi$$

$$f(t) = \frac{r}{\pi}\int_0^{\pi} d\theta\ Re\left[\exp(ts(\theta))\tilde{f}(s(\theta))(1+i\sigma(\theta))\right]$$

$$\sigma(\theta) = \theta + (\theta\cot\theta - 1)cot\theta$$

$$f(t,M) = \frac{r}{M}\left[\frac{1}{2}\tilde{f}(r)e^{rt} + \sum_{k=1}^{M-1} Re\left[\exp(ts(\theta_k))\tilde{f}(s(\theta_k))(1+i\sigma(\theta_k))\right]\right]$$

$$r = \frac{2M}{5t}, \qquad \theta_k = \frac{k\pi}{M}$$