Neural Network Determination of Parton Distribution Functions

Maria Ubiali

School of Physics, University of Edinburgh

Moriond QCD and High Energy Interactions
La Thuile, 13 March 2008

The NNPDF Collaboration
R.D.Ball[1], L.Del Debbio[1], S.Forte[2], A.Guffanti[3], J.I.Latorre[4], A. Piccione[2], J. Rojo[5], M.U.[1]

[1] PPT Group, School of Physics, University of Edinburgh
[2] Dipartimento di Fisica, Università di Milano
[3] Physikalisches Institut, Albert-Ludwigs-Universität Freiburg
[4] Departament d'Estructura i Constituents de la Matèria, Universitat de Barcelona
[5] LPTHE, Université Paris VI et Paris VII

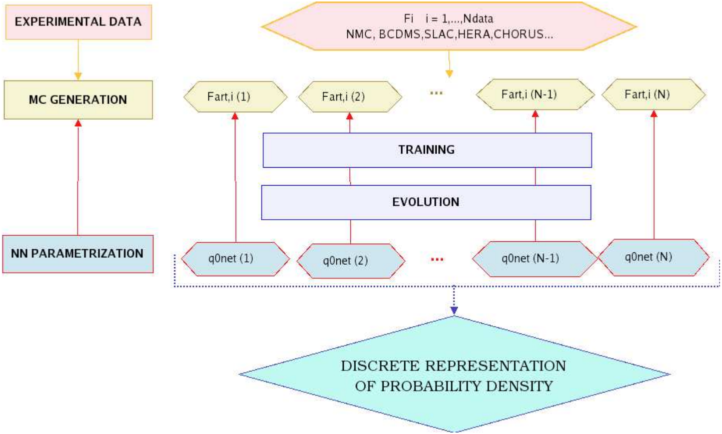
Issue and Standard Approach

- Given a set of data points we must determine a set of functions with error.
- We need an error band in the space of functions, i.e. a probability density $\mathcal{P}[q(x)]$ in the space of PDFs, q(x). For an observable $\mathcal{F}$ depending on PDFs :

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

- Standard approach, choose a basis of functions and project PDFs on it: the ∞-dimensional space of function reduces to a finite-dimensional space of parameters.
- Issues:
  - Non trivial propagation of errors: non-gaussian errors and incompatible data.
  - The error associated to the choice of parametrisation is difficult to assess.

NNPDF approach



$$\langle \mathcal{F}[q(x)] \rangle = \int [\mathcal{D}q]\, \mathcal{F}[q(x)] \mathcal{P}[q(x)] \quad \longrightarrow \quad \langle \mathcal{F}[q(x)] \rangle = \frac{1}{N_{\mathrm{rep}}} \sum_{k=1}^{N_{\mathrm{rep}}} \mathcal{F}[q^{(k)(\mathrm{net})}(x)]$$

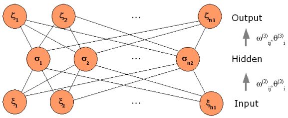
Main Ingredients

- Monte Carlo determination of errors:

After fitting, the error of an observable depending on PDFs $\longrightarrow$

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

- Neural Networks as redundant and unbiased parametrisation of PDFs:



* Each neuron receives input from neurons in preceding layer.

* Activation determined by weights and thresholds according to a non linear function:

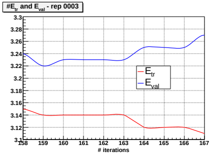
$$\xi_i = g\Big(\sum_j \omega_{ij}\xi_j - \theta_i\Big), \qquad g(x) = \frac{1}{1+e^{-x}}$$

- Dynamical stopping criterion in order to fit data and not statistical noise.

* Divide data in two sets: training and validation.

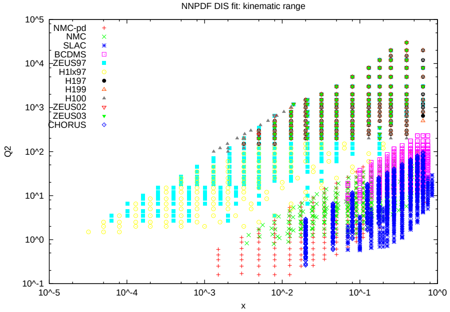
* Minimisation is performed only on the training set. The validation χ^2 for the set is computed.

* When the training χ^2 still decreases while the validation χ^2 stops decreasing $\longrightarrow$ STOP.



Singlet fit

- NLO fit.
- ZM-VFN treatment of heavy quarks.
- All DIS data included.
- Flavor Assumptions:
  - Symmetric strange sea $s(x) = \bar{s}(x)$
  - Strange sea proportional to non-strange sea $\bar{s}(x) = \frac{C}{2}(\bar{u}(x) + \bar{d}(x))$ (C = 0.5)

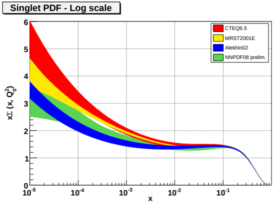


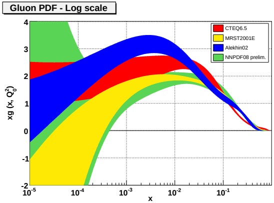
- Parametrization of 4+1 combinations of PDFs at $Q_0^2 = 2\ \text{GeV}^2$:

Singlet : $\Sigma(x)$	$\longmapsto$	$NN_\Sigma(x)$	2-3-2-1 20 pars
Gluon : $g(x)$	$\longmapsto$	$NN_g(x)$	2-3-2-1 20 pars
Total valence : $V(x) \equiv u_V(x) + d_V(x)$	$\longmapsto$	$NN_V(x)$	2-3-2-1 20 pars
Non-singlet triplet : $T_3(x)$	$\longmapsto$	$NN_{T3}(x)$	2-3-2-1 20 pars
Sea asymmetry : $\Delta_S(x) \equiv \bar{d}(x) - \bar{u}(x)$	$\longmapsto$	$NN_\Delta(x)$	2-3-1 13 pars

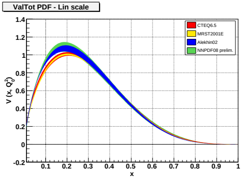
93 parameters

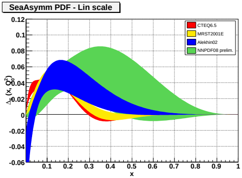
Some Very Preliminary Results

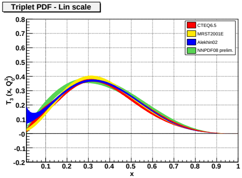




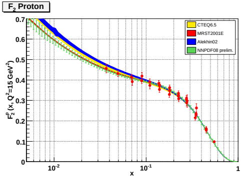
pdfs

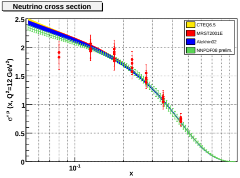


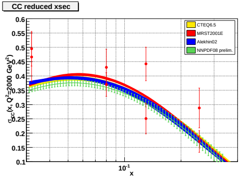




observables







Conclusions

- Standard approaches to PDFs fitting might lead to underestimation of errors associated with parton densities.
- Combination of Monte Carlo techniques and Neural Networks as unbiased interpolating functions has proved to be a fast and robust alternative method.
- A non singlet fit has been published [hep-ph/0701127] and a full DIS fit will be published very soon.

BACKUP SLIDES

MC replicas of experimental data

- Generate a $N_{\rm rep}$ Monte Carlo sets of artificial data, or "pseudo-data" of the original N_{data} data points

$$F_i^{(art)(k)} \qquad \begin{aligned} i &= 1, ..., N_{\rm data} \\ k &= 1, ..., N_{\rm rep} \end{aligned}$$

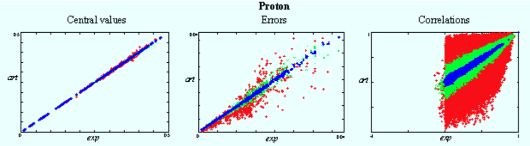
according to:

$$F_i^{(art)(k)} = (1 + r_N^{(k)})\,[F_i^{\rm exp} + r_S^{(K)} \sigma_i^{\rm stat} + \sum_{j=1}^{N_{\rm sys}} r_{j,\rm SY}^{(k)} \sigma_{ij}^{\rm sys}]$$

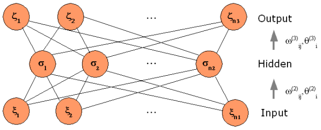
σ_i, experimental errors;
r_i, zero mean gaussian random numbers distributed according to the experimental correlation matrix.

- Validate MC replicas according to experimental data (statistical estimators, faithful representation of errors, convergence rate increasing $N_{\rm rep}$).

How many replicas do we need?
1000 replicas are enough to reproduce correlation to percent accuracy.



Explicit functional form of a NN



- Each neuron receives input from neurons in preceding layer.
- Activation determined by weight and threshold according to a non linear function:

$$\xi_i = g\Big(\sum_j \omega_{ij}\xi_j - \theta_i\Big), \qquad g(x) = \frac{1}{1+e^{-x}}$$

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

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

- NNs are just another set of basis functions.
- Thanks to non linear behaviour, any function can be represented by a sufficiently big NN.

Statistical Estimators I: observables

- Central value of the i-th experimental point

$$\left\langle F_i^{(\mathrm{art})} \right\rangle_{\mathrm{rep}} = \frac{1}{N_{\mathrm{rep}}} \sum_{k=1}^{N_{\mathrm{rep}}} F_i^{(\mathrm{art})(k)} \ .$$

- Variance of the i-th experimental point

$$\sigma_i^{(\mathrm{art})} = \sqrt{\left\langle \left(F_i^{(\mathrm{art})} \right)^2 \right\rangle_{\mathrm{rep}} - \left\langle F_i^{(\mathrm{art})} \right\rangle_{\mathrm{rep}}^2} \ .$$

- Associated covariance:

$$\rho_{ij}^{(\mathrm{art})} = \frac{\left\langle F_i^{(\mathrm{art})} F_j^{(\mathrm{art})} \right\rangle_{\mathrm{rep}} - \left\langle F_i^{(\mathrm{art})} \right\rangle_{\mathrm{rep}} \left\langle F_j^{(\mathrm{art})} \right\rangle_{\mathrm{rep}}}{\sigma_i^{(\mathrm{art})} \sigma_j^{(\mathrm{art})}} \ .$$

$$\mathrm{cov}_{ij}^{(\mathrm{art})} = \rho_{ij}^{(\mathrm{art})} \sigma_i^{(\mathrm{art})} \sigma_j^{(\mathrm{art})} .$$

Statistical Estimators II: replicas vs data

- Mean variance and percentage error on central values over the $N_{\rm dat}$ data points.

$$\left\langle V\left[\left\langle F^{(\rm art)}\right\rangle_{\rm rep}\right]\right\rangle_{\rm dat} = \frac{1}{N_{\rm dat}}\sum_{i=1}^{N_{\rm dat}}\left(\left\langle F_i^{(\rm art)}\right\rangle_{\rm rep} - F_i^{(\rm exp)}\right)^2 ,$$

$$\left\langle PE\left[\left\langle F^{(\rm art)}\right\rangle_{\rm rep}\right]\right\rangle_{\rm dat} = \frac{1}{N_{\rm dat}}\sum_{i=1}^{N_{\rm dat}}\left[\frac{\left\langle F_i^{(\rm art)}\right\rangle_{\rm rep} - F_i^{(\rm exp)}}{F_i^{(\rm exp)}}\right] .$$

- $\left\langle V\left[\left\langle \sigma^{(\rm art)}\right\rangle_{\rm rep}\right]\right\rangle_{\rm dat}, \left\langle V\left[\left\langle \rho^{(\rm art)}\right\rangle_{\rm rep}\right]\right\rangle_{\rm dat}, \left\langle V\left[\left\langle {\rm cov}^{(\rm art)}\right\rangle_{\rm rep}\right]\right\rangle_{\rm dat}$

  $\left\langle PE\left[\left\langle \sigma^{(\rm art)}\right\rangle_{\rm rep}\right]\right\rangle_{\rm dat}, \left\langle PE\left[\left\langle \rho^{(\rm art)}\right\rangle_{\rm rep}\right]\right\rangle_{\rm dat}, \left\langle PE\left[\left\langle {\rm cov}^{(\rm art)}\right\rangle_{\rm rep}\right]\right\rangle_{\rm dat}$

  relative to errors, correlations and covariances are defined in the same way.
- These estimators indicate how close are the averages over generated data and the experimental values.

Stability estimators III: replicas vs data

- Scatter correlation:

$$r\left[F^{(\mathrm{art})}\right]=\frac{\left\langle F^{(\mathrm{exp})}\left\langle F^{(\mathrm{art})}\right\rangle_{\mathrm{rep}}\right\rangle_{\mathrm{dat}}-\left\langle F^{(\mathrm{exp})}\right\rangle_{\mathrm{dat}}\left\langle\left\langle F^{(\mathrm{art})}\right\rangle_{\mathrm{rep}}\right\rangle_{\mathrm{dat}}}{\sigma_s^{(\mathrm{exp})}\sigma_s^{(\mathrm{art})}},$$

  where the scatter variances are defined as

$$\sigma_s^{(\mathrm{exp})}=\sqrt{\left\langle\left(F^{(\mathrm{exp})}\right)^2\right\rangle_{\mathrm{dat}}-\left(\left\langle F^{(\mathrm{exp})}\right\rangle_{\mathrm{dat}}\right)^2},$$

$$\sigma_s^{(\mathrm{art})}=\sqrt{\left\langle\left(\left\langle F^{(\mathrm{art})}\right\rangle_{\mathrm{rep}}\right)^2\right\rangle_{\mathrm{dat}}-\left(\left\langle\left\langle F^{(\mathrm{art})}\right\rangle_{\mathrm{rep}}\right\rangle_{\mathrm{dat}}\right)^2}.$$

- $r\left[\sigma^{(\mathrm{art})}\right]$ $r\left[\rho^{(\mathrm{art})}\right]$ $r\left[\mathrm{cov}^{(\mathrm{art})}\right]$ are defined in the same way.
- The scatter correlation indicates the size of the spread of data around a straight line. Specifically $r\left[\sigma^{(\mathrm{art})}\right]=1$ implies that $\left\langle\sigma_i^{(\mathrm{art})}\right\rangle$ is proportional to $\sigma_i^{(\mathrm{exp})}$.

Stability estimators and theoretical error

- Difficult to give a statistical measure of theoretical error: check that the final result depend within 2σ on theoretical assumptions.
- E.g. choice of the initial parametrisation:

$$d[q] = \sqrt{\left\langle \frac{\left(q_i^{(1)} - q_i^{(2)}\right)^2}{(\sigma_i^{(1)})^2 + (\sigma_i^{(2)})^2} \right\rangle_{\mathrm{dat}}},$$

  $q_i^{(1)}$, $q_i^{(2)}$ predictions for the i-th data point in the two fits, $\sigma_i^{(1)}$, $\sigma_i^{(2)}$ predictions for the corresponding statistical uncertainties.
- The results of the first and second fit are statistically equivalent if $d[q] = 1$ on average.
- The same must be done for the choice of kinematical cuts, random seeds, preprocessing exponents...

Neural Network and Training Algorithm

- Set neural network parameters randomly.
- Make clones of the parameter vector and mutate them.
- Evaluate the figure of merit for each clone:

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

- Select the best ones and iterate the procedure until a stability is reached.

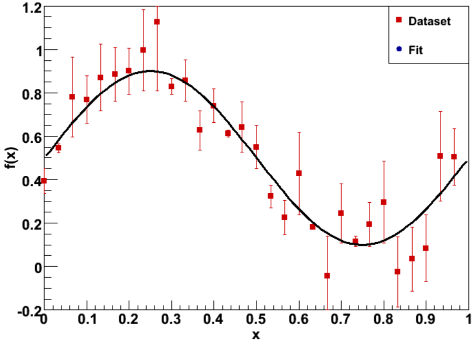
PROs

- The possibility of getting trapped in a local minimum is reduced.
- Allows to minimise the fully correlated χ^2.

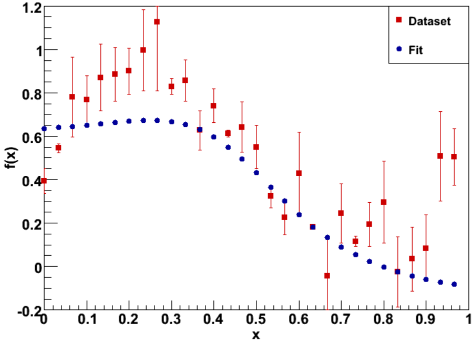
CONs

- It is monotonically decreasing by construction.
- It risks to converge slowly if the parameters ar not properly tuned.

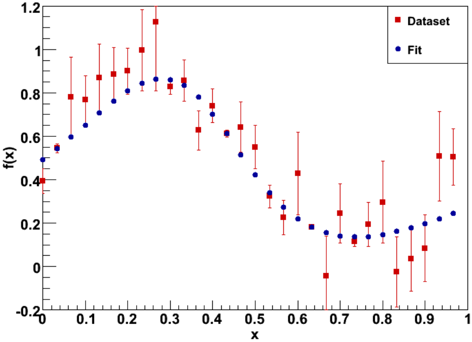
Proper Fitting avoiding Overlearning: an example



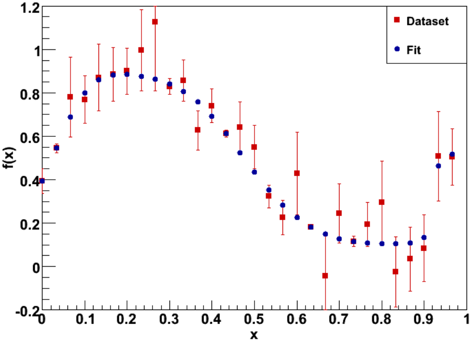
Proper Fitting avoiding Overlearning: an example



Proper Fitting avoiding Overlearning: an example



Proper Fitting avoiding Overlearning: an example

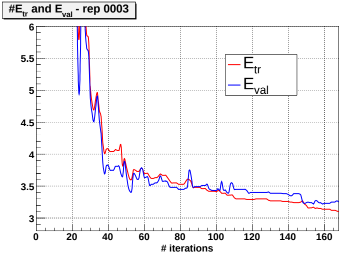


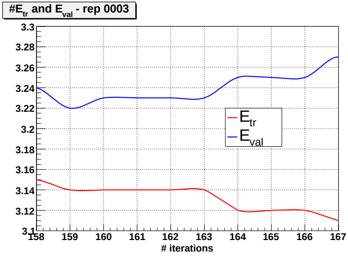
- Need a redundant parametrization to avoid excessive constraining
- Need a way of stopping the fit before overlearning sets in

How to avoid Overlearning?

Stopping criterion based on Training-Validation separation

* Divide data in two sets: training and validation.
* Minimisation is performed only on the training set. The validation χ^2 for the set is computed.
* When the training χ^2 still decreases while the validation χ^2 stops decreasing → STOP.





The Evolution Code

- Observables are a convolution over x of PDFs and Coefficient Functions.
- Each observable is a particular linear combination of $(2n_f + 1)$ parton distributions.
- Data are given at various scales $\rightarrow$ Solve DGLAP eqns and evolve from the initial parametrisation scale Q_0^2 to the experimental one.
- Theory: higher perturbative orders, resummations, higher twists, nuclear corrections, heavy quark threshold...

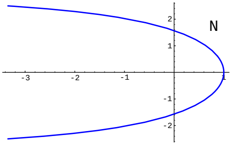
We want $\rightarrow$ Mellin space evolution.

$$\tilde{\Gamma}(N, \alpha_s(Q^2), \alpha_s(Q_0^2)) = C(N, \alpha_s(Q^2))\, \Gamma(N, \alpha_s(Q^2), \alpha_s(Q_0^2))$$

We do not want $\rightarrow$ Complex neural networks.

$$\begin{aligned} q(x, Q^2) &= \int_x^1 \frac{dy}{y}\, \tilde{\Gamma}(y, \alpha_s(Q^2), \alpha_s(Q_0^2))\, q(\frac{x}{y}, Q_0^2) \\ \tilde{\Gamma}(y, \alpha_s(Q^2), \alpha_s(Q_0^2)) &= \frac{1}{2\pi i} \int_C dN\, x^{-N}\, \tilde{\Gamma}(N, \alpha_s(Q^2), \alpha_s(Q_0^2)) \end{aligned}$$

Evolution



$$q(x, Q^2) = \gamma\, q(x, Q_0^2) + \int_x^1 \frac{dy}{y}\, \Gamma(y, a_s, a_0) \left[q\left(\frac{x}{y}, Q_0^2\right) - y\, q(x, Q_0^2) \right]$$

$$\gamma = \int_{c-i\infty}^{c+\infty} \frac{dN}{2\pi i} \frac{\Gamma(N)}{1-N} - \int_0^x dy\, \Gamma(y, a_s, a_0).$$

x	$e_{\rm rel}(u_v)$	$e_{\rm rel}(d_v)$	$e_{\rm rel}(\Sigma)$	$e_{\rm rel}(g)$
$1 \cdot 10^{-7}$	$7.6 \cdot 10^{-5}$	$4.5 \cdot 10^{-7}$	$1.3 \cdot 10^{-4}$	$1.2 \cdot 10^{-4}$
$1 \cdot 10^{-6}$	$8.3 \cdot 10^{-6}$	$3.2 \cdot 10^{-6}$	$1.2 \cdot 10^{-5}$	$1.7 \cdot 10^{-5}$
$1 \cdot 10^{-5}$	$4.7 \cdot 10^{-6}$	$1.4 \cdot 10^{-5}$	$1.5 \cdot 10^{-6}$	$2.2 \cdot 10^{-5}$
$1 \cdot 10^{-4}$	$3.3 \cdot 10^{-6}$	$3.3 \cdot 10^{-6}$	$1.4 \cdot 10^{-5}$	$4.8 \cdot 10^{-6}$
$1 \cdot 10^{-3}$	$1.3 \cdot 10^{-1}$	$9.7 \cdot 10^{-6}$	$2.6 \cdot 10^{-6}$	$1.5 \cdot 10^{-5}$
$1 \cdot 10^{-2}$	$2.9 \cdot 10^{-6}$	$1.6 \cdot 10^{-5}$	$5.5 \cdot 10^{-6}$	$4.9 \cdot 10^{-6}$
$1 \cdot 10^{-1}$	$7.9 \cdot 10^{-6}$	$1.1 \cdot 10^{-5}$	$5.2 \cdot 10^{-6}$	$3.8 \cdot 10^{-6}$
$3 \cdot 10^{-1}$	$1.4 \cdot 10^{-5}$	$2.7 \cdot 10^{-5}$	$1.8 \cdot 10^{-6}$	$3.6 \cdot 10^{-6}$
$5 \cdot 10^{-1}$	$2.8 \cdot 10^{-7}$	$1.0 \cdot 10^{-5}$	$1.7 \cdot 10^{-6}$	$6.4 \cdot 10^{-6}$
$7 \cdot 10^{-1}$	$9.0 \cdot 10^{-6}$	$7.3 \cdot 10^{-6}$	$8.7 \cdot 10^{-6}$	$7.6 \cdot 10^{-6}$
$9 \cdot 10^{-1}$	$1.1 \cdot 10^{-5}$	$6.0 \cdot 10^{-6}$	$1.0 \cdot 10^{-5}$	$7.9 \cdot 10^{-6}$

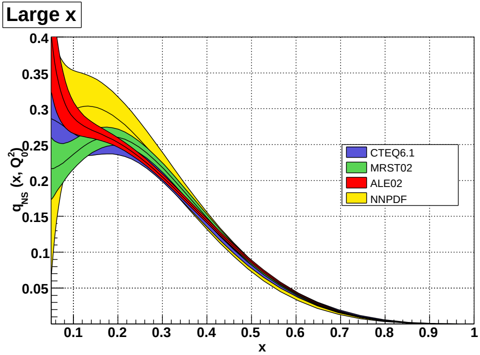
Table: LH benchmark vs NNPDF output for u_v, d_v, Σ and g distributions.
NLO accuracy, **VFN** scheme, **truncated** solution. Inversion with FT algorithm.

Non singlet fit

- Determination of

$$T_3(x, Q_0^2) \equiv (u + \bar{u} - d - \bar{d})(x, Q_0^2)$$

  at $Q_0^2 = 2\text{GeV}^2$ at LO, NLO, NNLO.
- DATA SETS: $F_2^p(x, Q^2) - F_2^d(x, Q^2)$ BCDMS and NMC



See hep-ph/0701127 for all technical details