Error Estimation with Monte Carlo method

Alberto Guffanti

Albert-Ludwigs-Universität Freiburg

On behalf of the NNPDF Collaboration:
R. D. Ball, L. Del Debbio, M. Ubiali (Edinburgh), S. Forte, A. Piccione (Milano),
J. I. Latorre (Barcelona), J. Rojo-Chacon (LPTHE - Paris)

PDF4LHC Workshop CERN, February 22 - 23, 2008

(Some) Open problems in PDF fitting

- How to fit an unknown function from a set of noisy data
- Treatment of incompatible datasets
- Treatment of non-gaussian error

(Some) Open problems in PDF fitting

- How to fit an unknown function from a set of noisy data
- Treatment of incompatible datasets
- Treatment of non-gaussian error

NNPDF Idea:
Combine of **Monte Carlo techinques** for error estimation
AND
Neural Networks as unbiased interpolants.

The Neural Network Approach

1. Generate N_{rep} Monte-Carlo replicas of the experimental data.
2. Fit a set of Parton distribution functions on each replica, thus defining a sampling of probability density on the space of the PDFs.
3. Expectation values for observables are Monte Carlo integral over nets

$$\langle \mathcal{F}[f_i(x,Q^2)] \rangle = \frac{1}{N_{rep}} \sum_{k=1}^{N_{rep}} \mathcal{F}\left(f_i^{(net)(k)}(x,Q^2) \right)$$

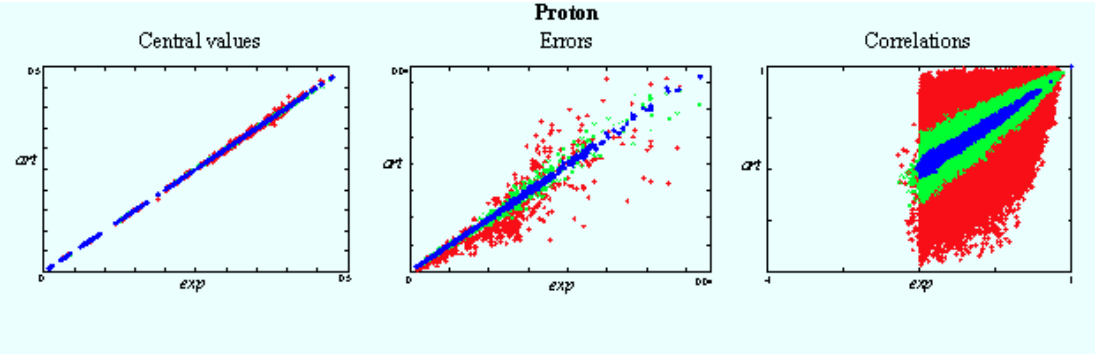
Monte Carlo replicas generation

- Generate artificial data according to distribution

$$O_i^{(art)\,(k)} = (1 + r_N^{(k)}\,\sigma_N)\left[O_i^{(exp)} + \sum_{p=1}^{N_Sys} r_p^{(k)}\,\sigma_{i,p} + r_{i,s}^{(k)}\,\sigma_s^i\right]$$

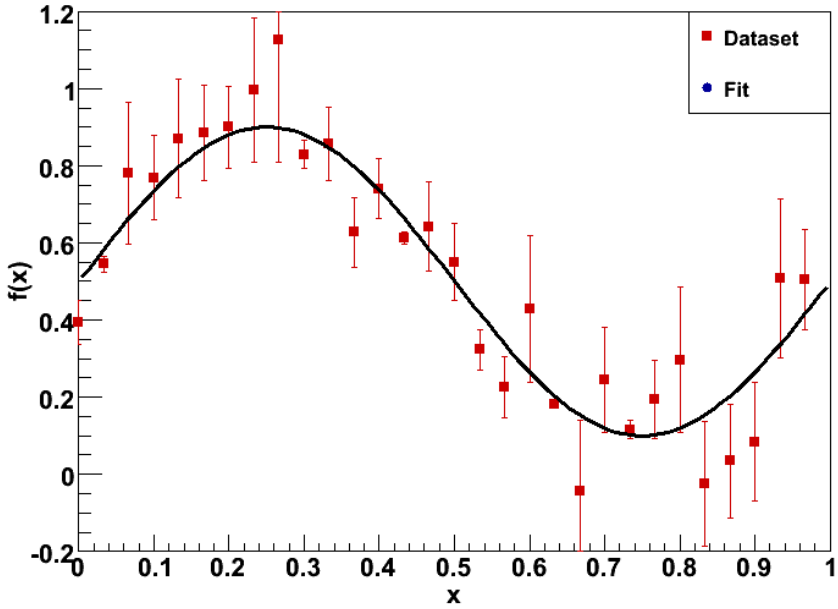
where r_i are univariate gaussian random numbers

- Validate Monte Carlo replicas against experimental data (statistical estimators, faithful representation of errors, convergence rate increasing N_{rep})

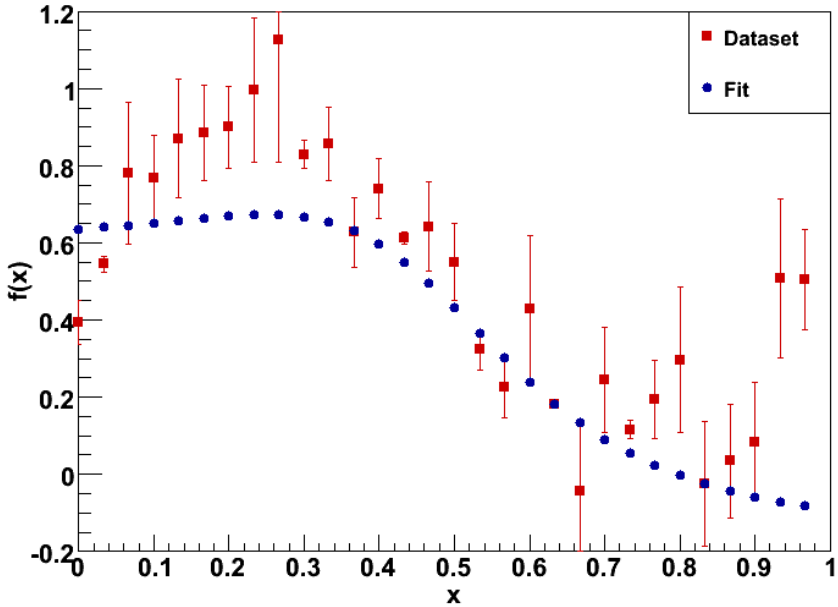


- $\mathcal{O}(1000)$ replicas needed to reproduce correlations to percent accuracy

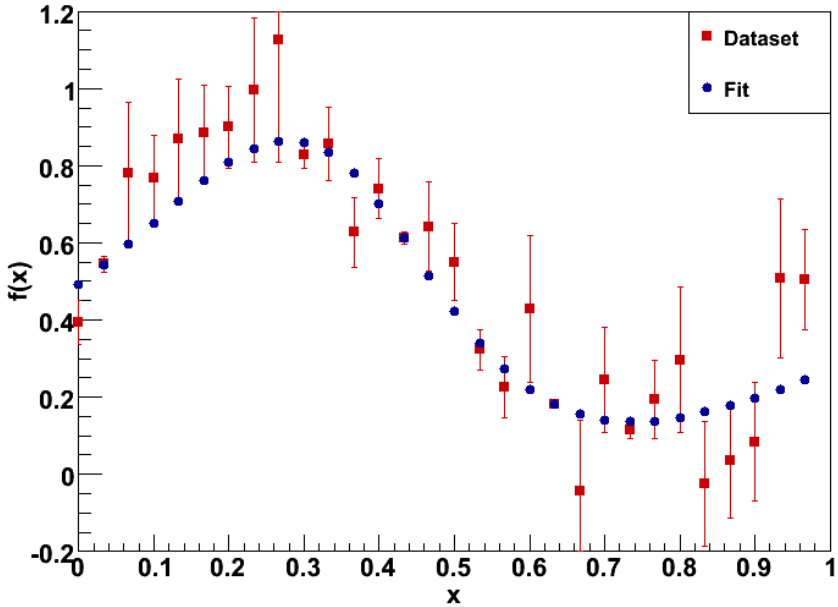
Proper Fitting avoiding Overlearning



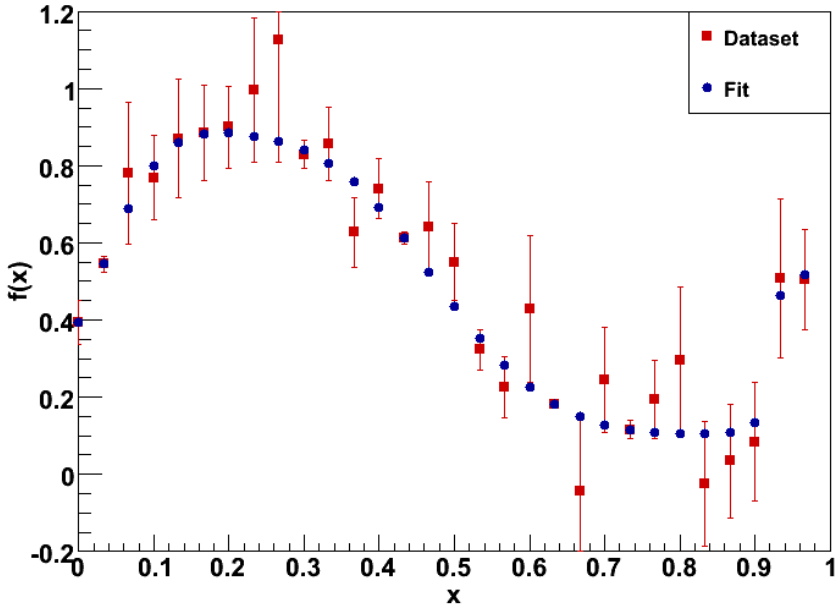
Proper Fitting avoiding Overlearning



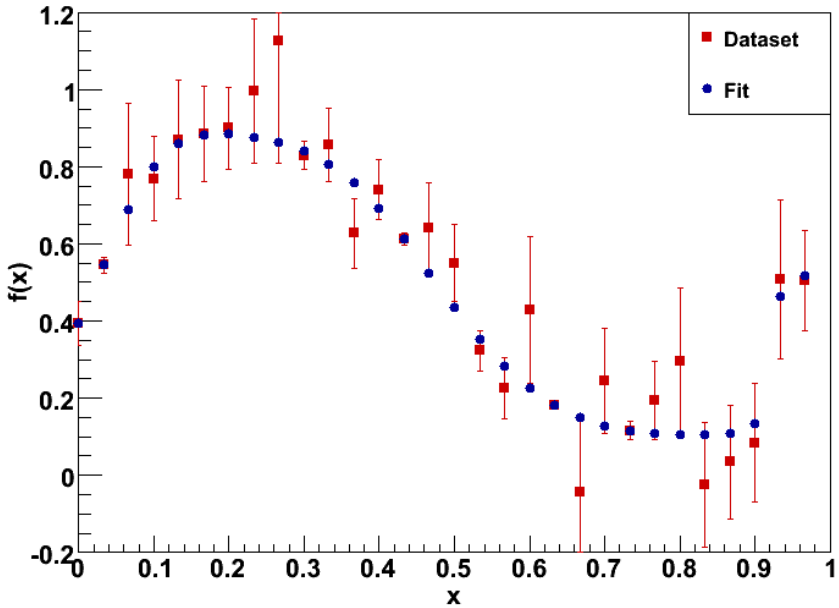
Proper Fitting avoiding Overlearning



Proper Fitting avoiding Overlearning



Proper Fitting avoiding Overlearning



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

How to avoid Overlearning

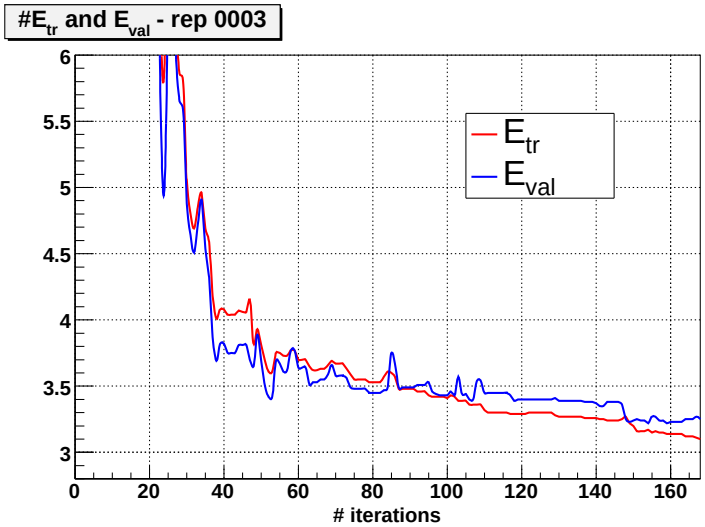
Stopping criterion based on Training-Validation separation

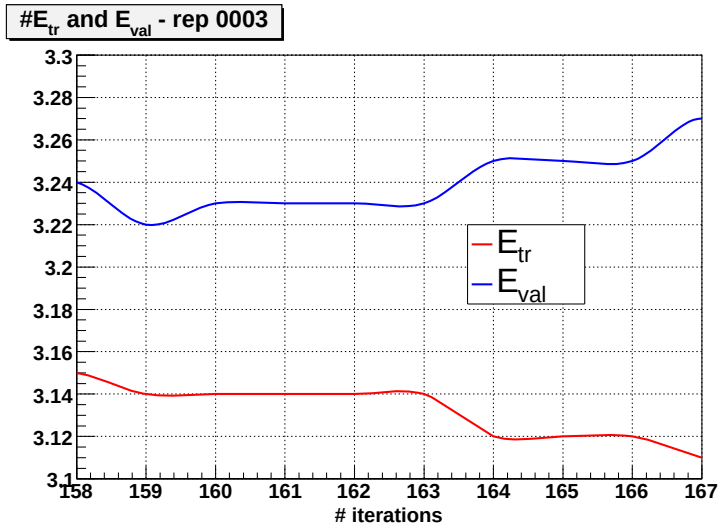
- Divide the data in two sets: Training and Validation
- Minimize the χ^2 of the data in the Training set
- Compute the χ^2 for the data in the Validation set
- When validation χ^2 stops decreasing, **STOP** the fit

How to avoid Overlearning

Stopping criterion based on Training-Validation separation

- Divide the data in two sets: Training and Validation
- Minimize the χ^2 of the data in the Training set
- Compute the χ^2 for the data in the Validation set
- When validation χ^2 stops decreasing, **STOP** the fit

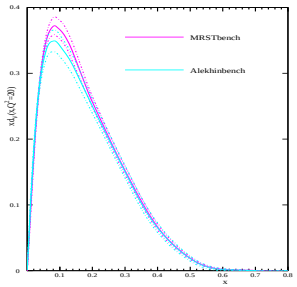


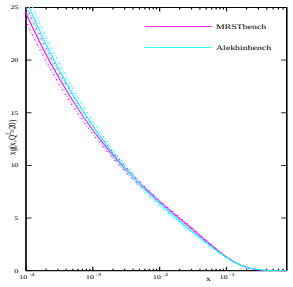


The HERA-LHC Benchmark

What went wrong?

- Benchmark partons agree within relative errors

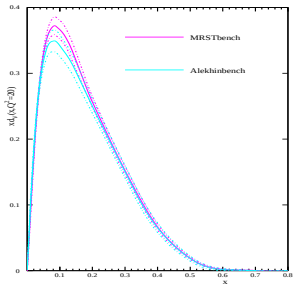


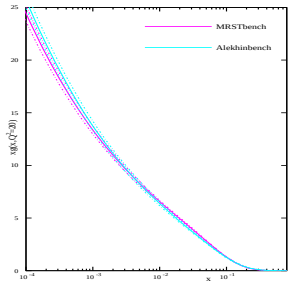


The HERA-LHC Benchmark

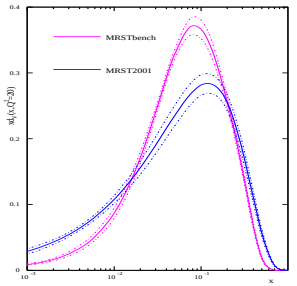
What went wrong?

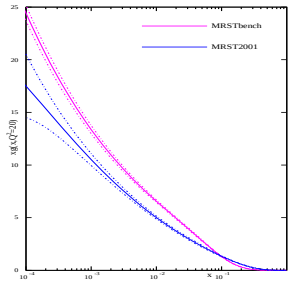
- Benchmark partons agree within relative errors





- Benchmark partons don't agree with global partons

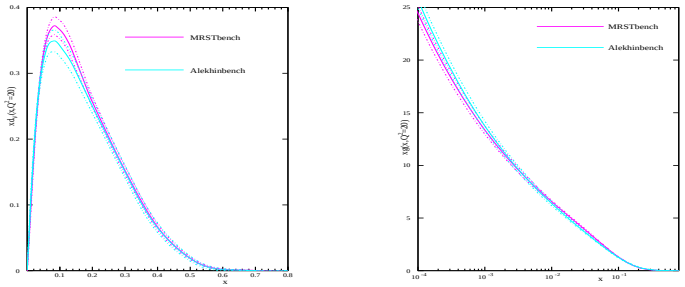




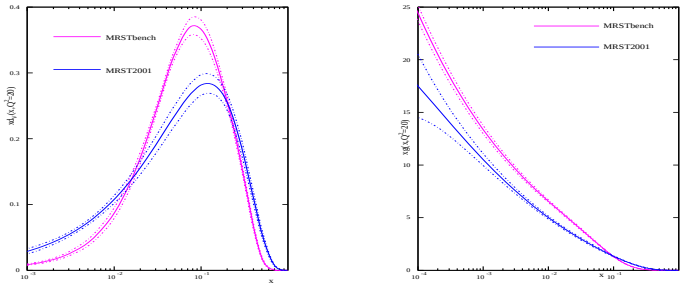
The HERA-LHC Benchmark

What went wrong?

- Benchmark partons agree within relative errors



- Benchmark partons don't agree with global partons

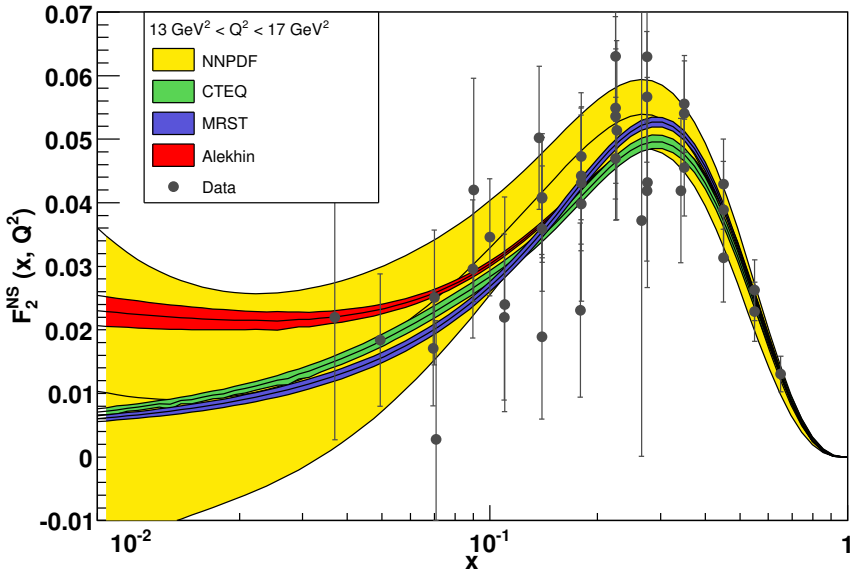


Possible explanations: Underestimated errors?
Incompatible datasets?
A bit of both?

Might Neural MC methods improve the situation?

Example: F_2^{NS} determination

[L. Del Debbio et al., hep-ph/0701127]



- Compatible with results from other PDF determinations
  (even when they are not in agreement)
- Lager uncertainties both in the
  - Data region (MC error estimation)
  - Extrapolation region (functional form bias)