Hyperparameter optimisation of neural networks for proton structure analyses

Based on 2410.16248 with NNPDF and eScience center

Roy Stegeman
The University of Edinburgh

EPS-HEP, Marseille, 7 July 2025

Parton Distribution Functions

Factorization: the LHC master formula

$$\underbrace{\sigma}_{\text{Measurable quantity}} = \sum_{ij} \underbrace{f_i \otimes f_j}_{\text{PDFs}} \otimes \underbrace{\hat{\sigma}_{ij}}_{\text{pQCD}}$$

Parton Distribution Functions (PDFs) are:

- Roughly speaking the probability of sampling a parton (quark or gluon) from a proton

Parton Distribution Functions

Factorization: the LHC master formula

$$\sigma = \sum_{ij} f_i \otimes f_j \otimes \hat{\sigma}_{ij}$$

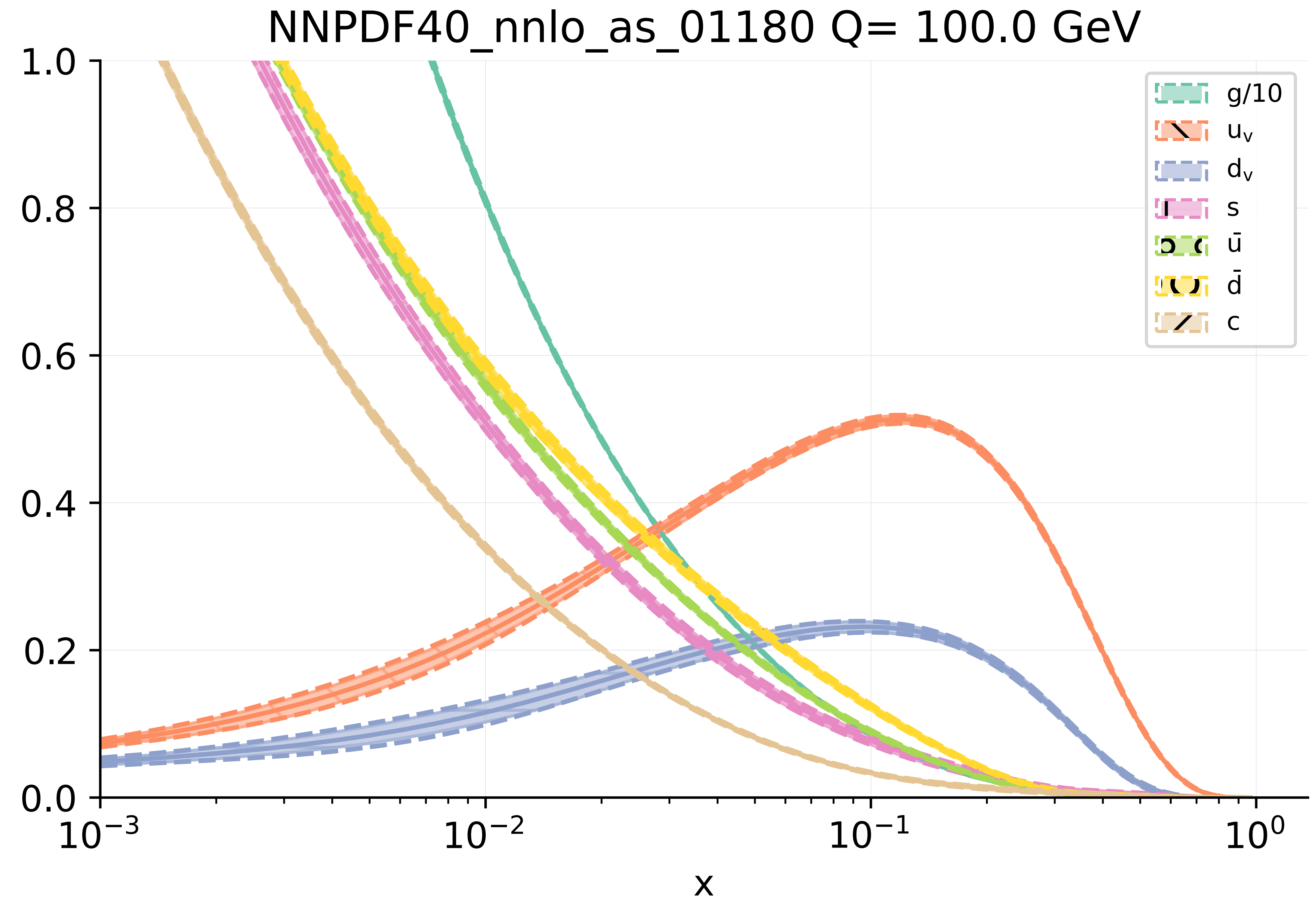
Measurable quantity

PDFs

pQCD

Parton Distribution Functions (PDFs) are:

- Roughly speaking the probability of sampling a parton (quark or gluon) from a proton
- 2 variables:
  - Longitudinal momentum fraction of the parton $x = p_{\text{parton}}/p_{\text{proton}}$
  - Energy scale Q
- Q scaling is known from perturbative QCD (DGLAP)
- **ML problem**: find $f_i(x)$ at a fixed scale Q_0



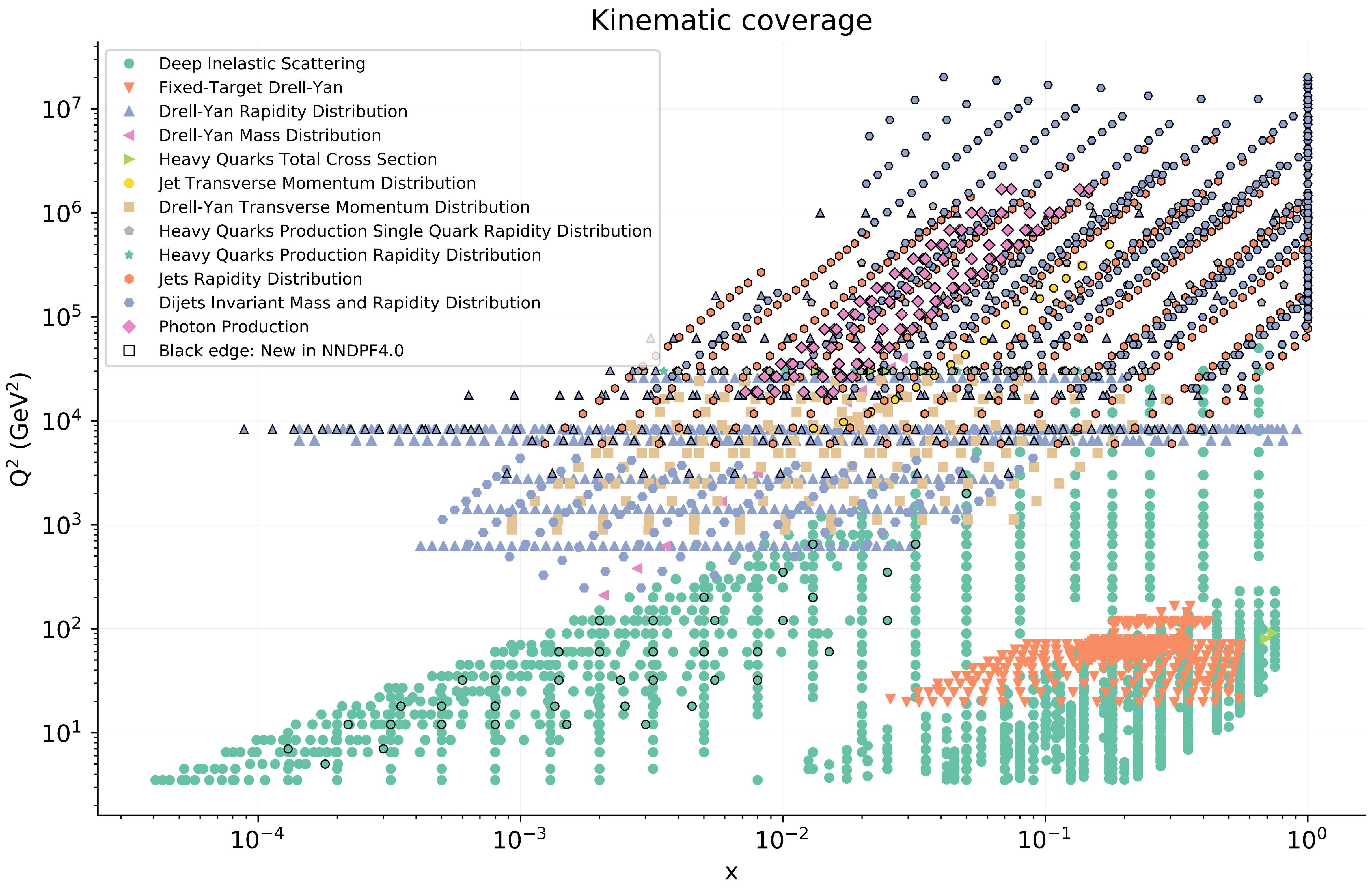
- The NNPDF methodology
- Hyperoptimisation
- Validation of the methodology

The experimental dataset

- **~4500 datapoints** across a wide range of kinematics and processes
- Uncertainties are approximated as **Gaussian**
- For Gaussian data the likelihood estimator is

$$P(\text{model}\,|\,\text{data}) \propto \exp[-\chi^2/2]$$

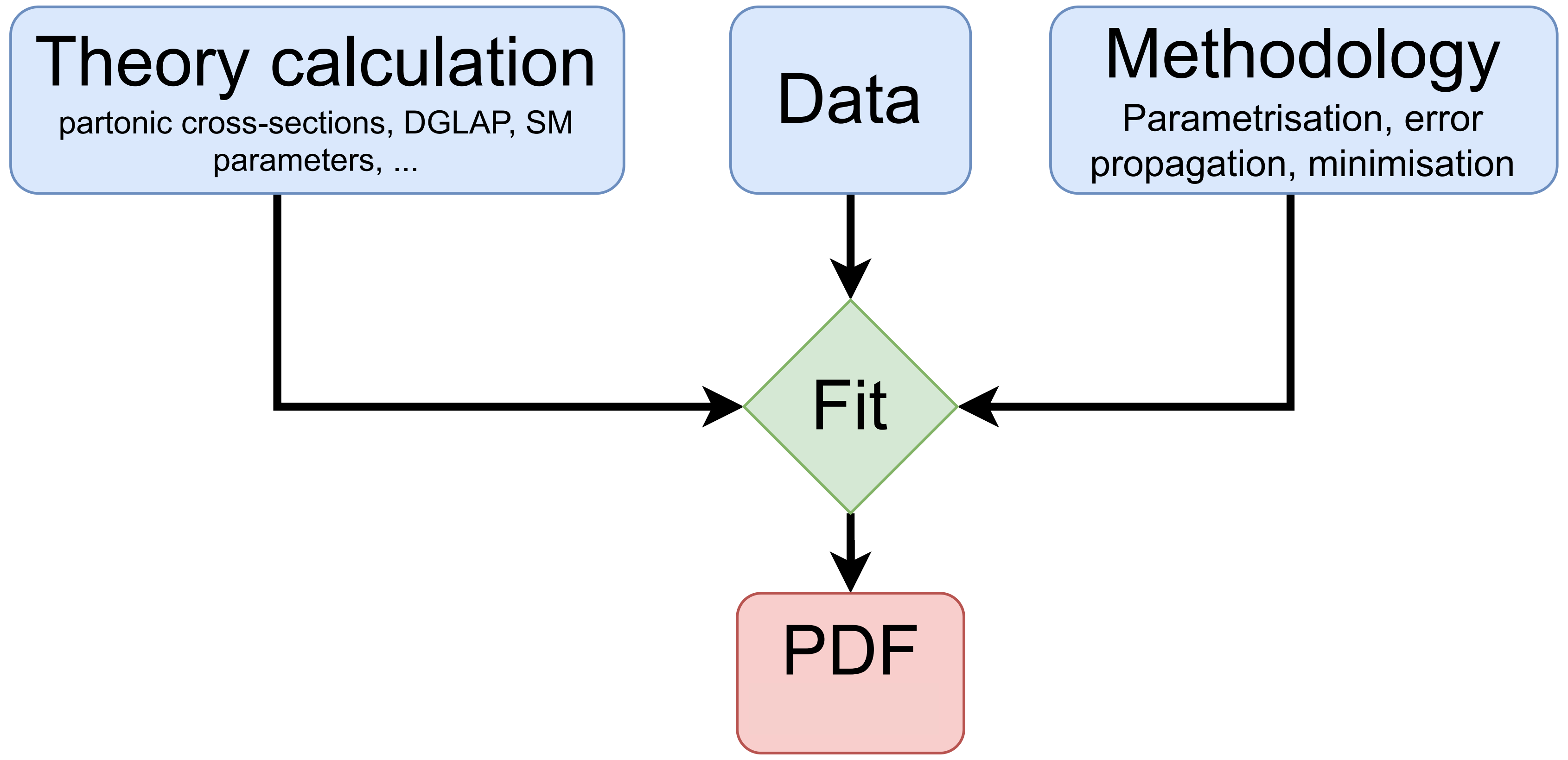
$$\chi^2 = \sum_{i,j}^{N_{\text{dat}}} \left(\text{data} - \text{prediction}\right)_i \text{cov}_{ij}^{-1} \left(\text{data} - \text{prediction}\right)_j$$



PDF determination

Besides data a PDF fit requires theory calculations and a methodology, this talk is about the latter

Different groups, make different choices for each

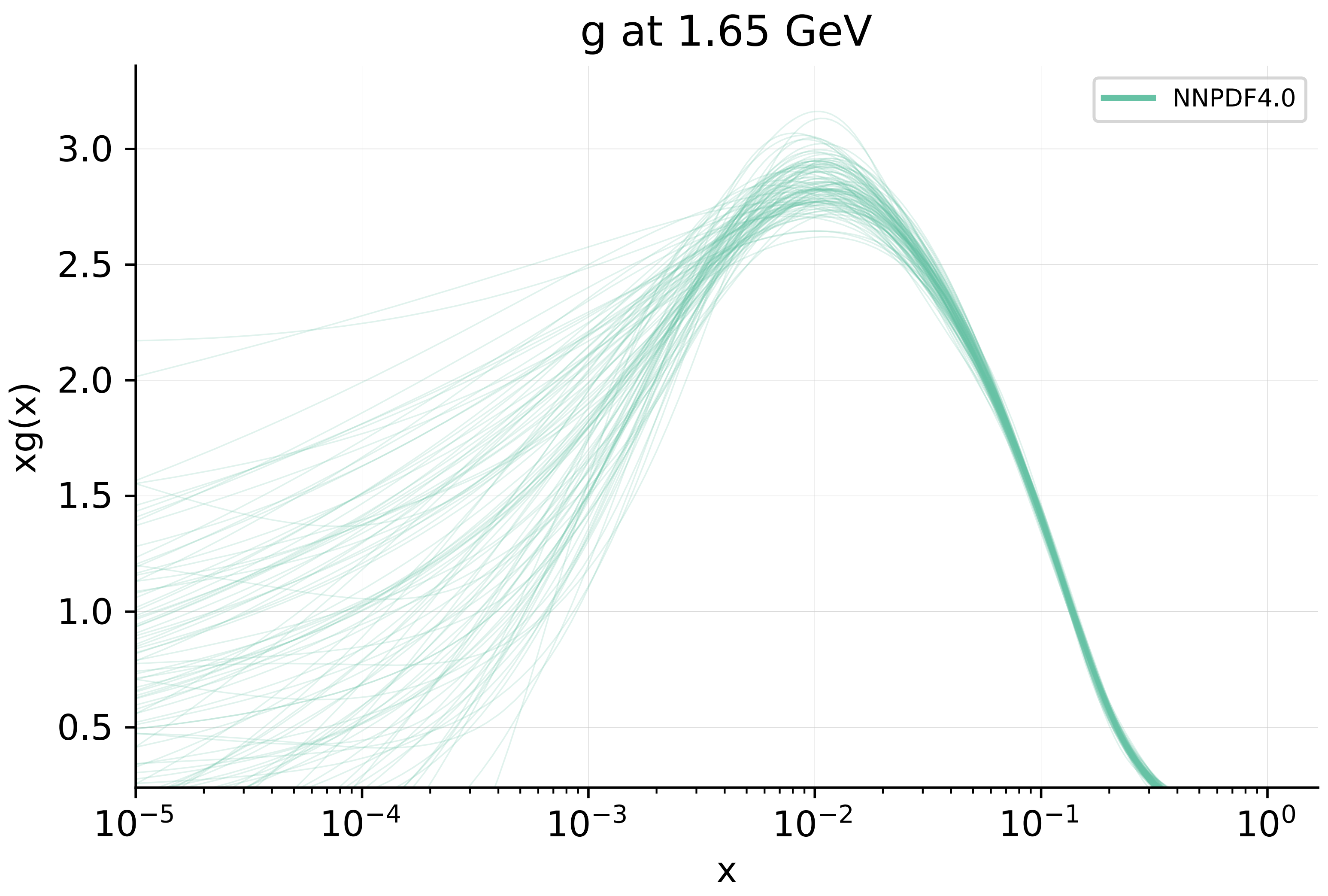


A methodology consists of…

- …a way to **parametrise** the PDFs -> neural network
- …a way to **fit parameters** to data -> gradient descent
- …a way to **propagate uncertainties** from data to functions?

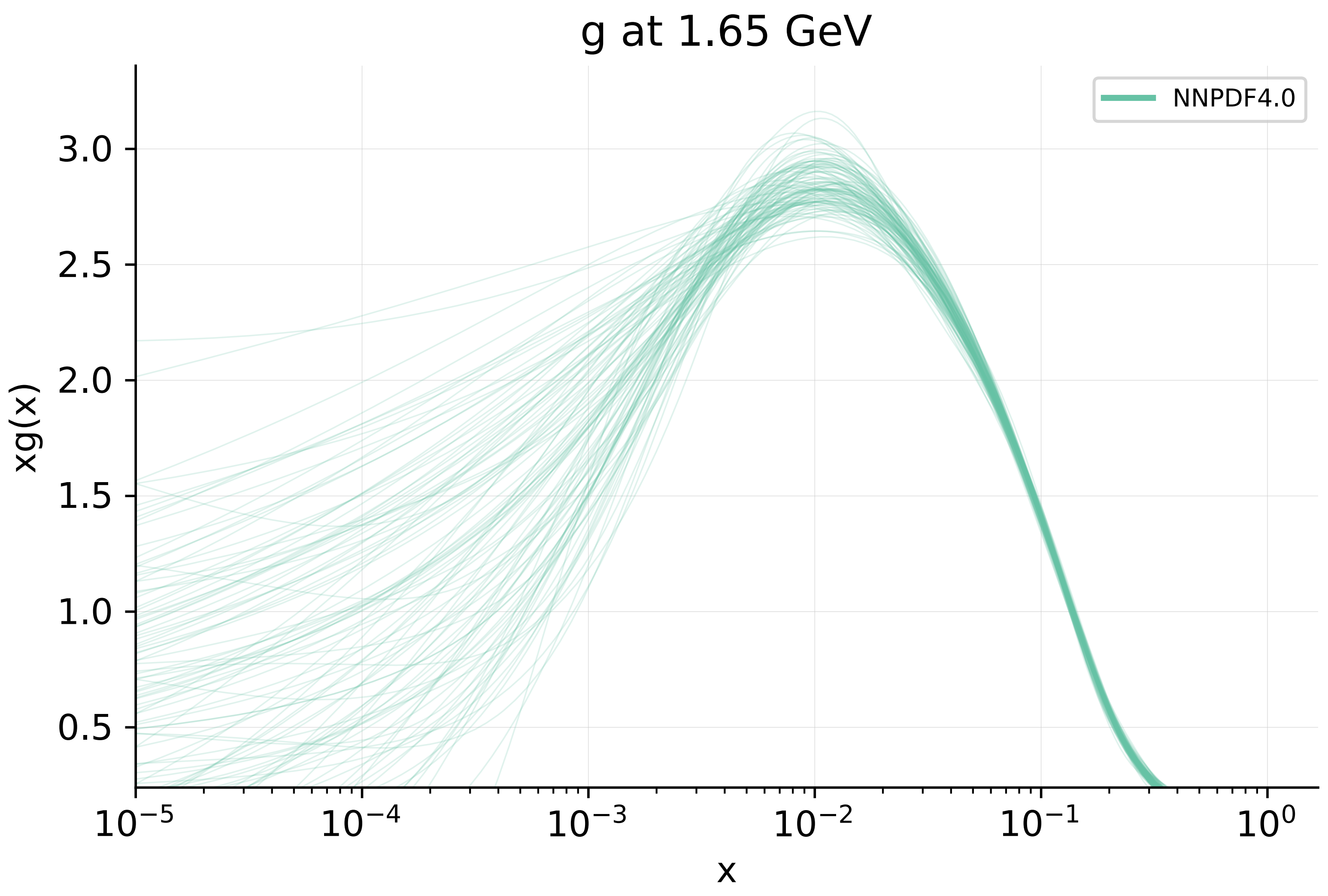
Uncertainty propagation

Create a Monte Carlo samples of “**synthetic data replicas**”
$D^{(k)} \sim \mathcal{N}(D, \mathrm{Cov}_{\mathrm{exp}})$



Uncertainty propagation

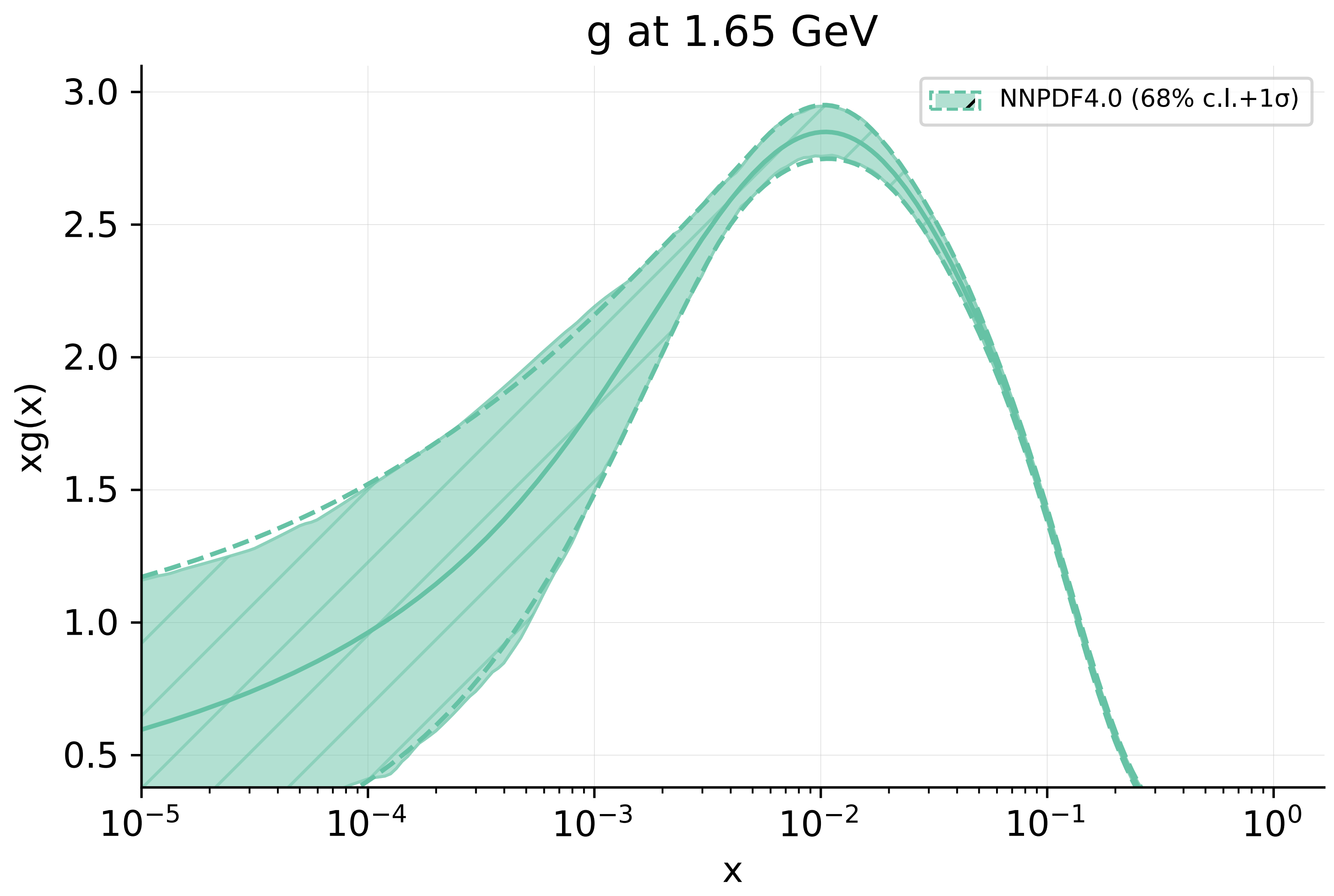
Create a Monte Carlo samples of “**synthetic data replicas**”
$D^{(k)} \sim \mathcal{N}(D, \mathrm{Cov}_{\mathrm{exp}})$



Compute mean and variance of PDF-dependent observables

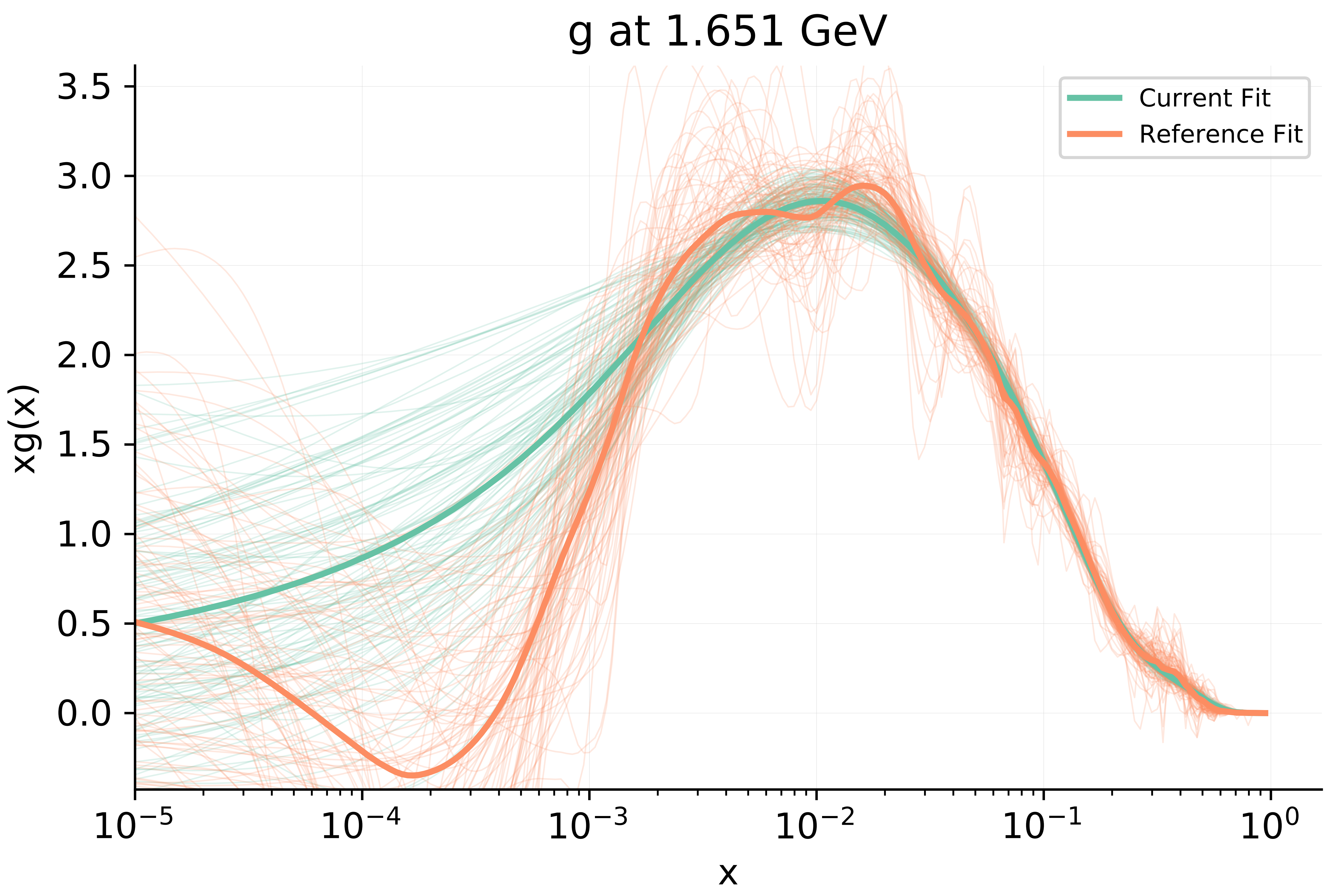
$$\langle \mathcal{O}[f] \rangle \simeq \frac{1}{N} \sum_{k=1}^{N} \mathcal{O}\left[f^{(k)}\right]$$

$$\mathrm{Var}[\mathcal{O}] \simeq \frac{1}{N} \sum_{k=1}^{N} \left(\mathcal{O}\left[f^{(k)}\right] - \langle \mathcal{O} \rangle\right)^2$$



- The NNPDF methodology
- **Hyperoptimisation**
- Validation of the methodology

Fitting PDFs



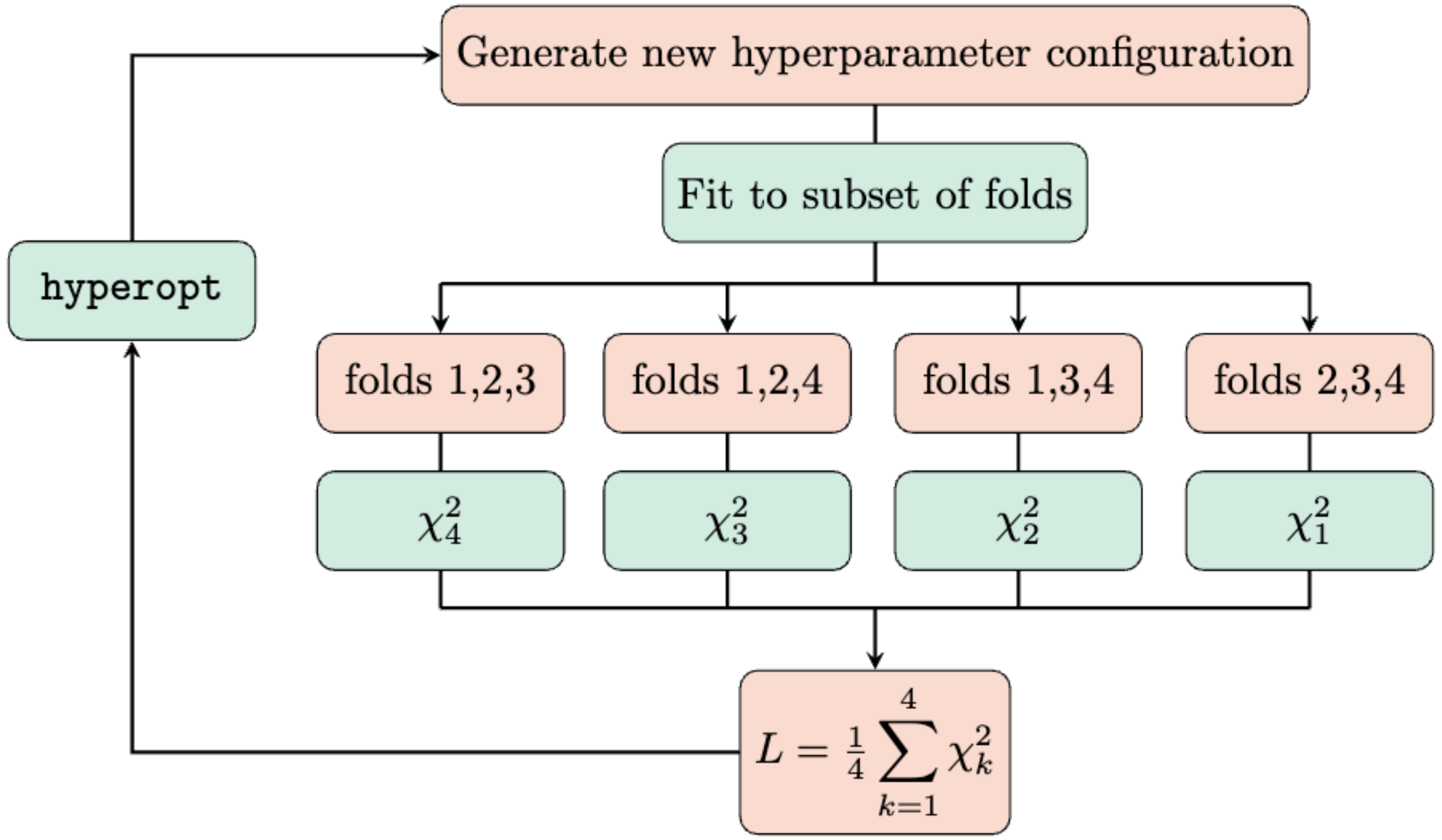
Setting the methodology hyperparameters requires care

The wrong choice may lead to over- or under-fitting

Hyperparameter optimisation

NNPDF4.0 used k-folds cross-validation

1. Partition the dataset into 4 folds
2. Exclude one at a time, perform 4 fits
3. Hyperoptimization metric: best average χ^2 to non-fitted data



Ideally include the PDF uncertainty in the hyperoptimization

- **computationally heavy:**
  4 cpu hours x 4 folds = 16 hours at 16 GB of memory
- This had to be reduced to use higher moments in hyperoptimization

➡solution: GPUs!

Hyperopt using GPUs

# Replicas	10	50	100
Energy reduction	78%	87%	91%
Cost reduction	−45%	47%	55%

NVIDIA H100 GPU vs 16 AMD EPYC Genoa CPU on SURF's SNELLIUS cluster

Technical changes:

- Single NN model for all samples
- Share memory-heavy objects
- Single hyperopt database shared by GPUs

Results:

- Memory usage scales only weakly with number of replicas, enabling a 100 replica fit in a single GPU
- 90% energy reduction: faster and more affordable fits!

Model selection

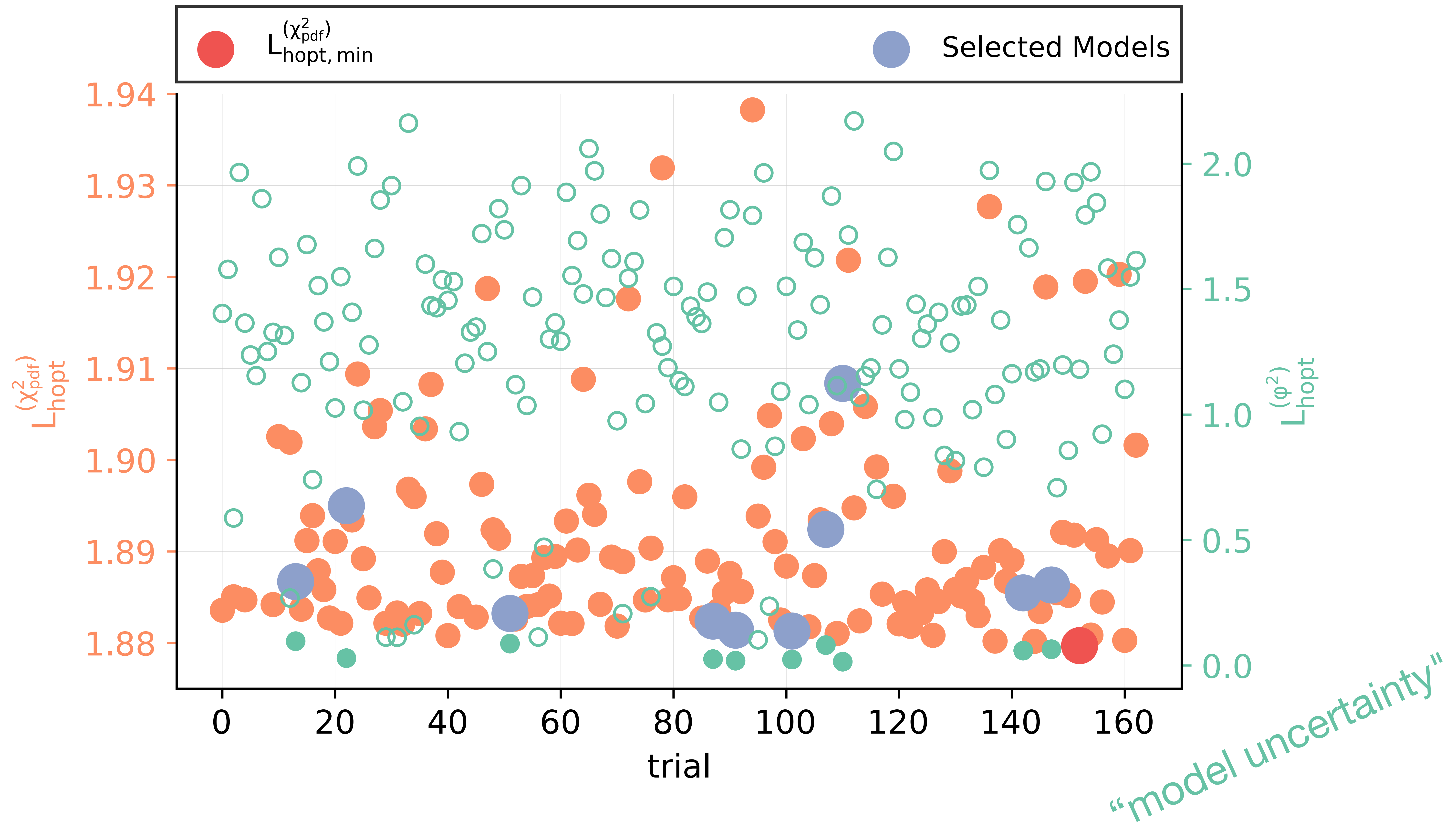
How to define the figure of merit?

Difficult question: what metric should be used to define a “good fit”?

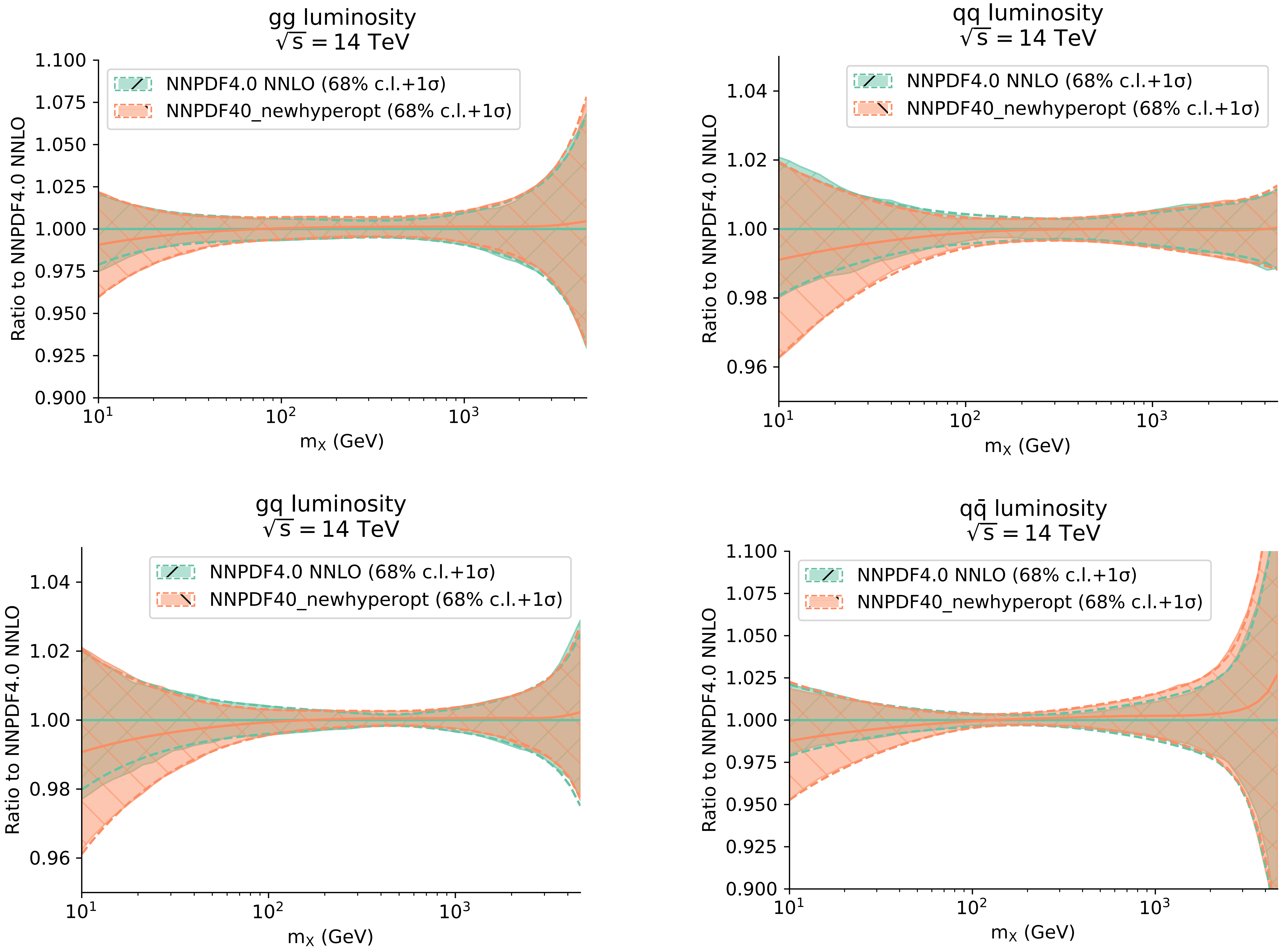
This is actively worked on, but as a first attempt we used the strategy:

1) look at configurations that describe data equally well
2) Pick the ones with the largest uncertainty

In a fit: select not a single setup but randomly **sample over all acceptable configurations**



Results



Large changes to the hyper parameter determination methodology, but results still in **good agreement with NNPDF4.0**

- The NNPDF methodology
- Hyperoptimisation
- Validation of the methodology

Uncertainty validation: closure tests

[Del Debio, Giani, Wilson, 2111.05787]

Basic idea: generate a global pseudo dataset from theory predictions and extract the PDFs from this

Input PDFs $\otimes$ Theory prediction grids $\rightarrow$ pseudodata $\rightarrow$ methodology $\rightarrow$ extracted PDF

Do they agree?*

Uncertainty validation: closure tests

[Del Debio, Giani, Wilson, 2111.05787]

Basic idea: generate a global pseudo dataset from theory predictions and extract the PDFs from this

* Experimental data is sampled from a distribution, therefore pseudodata = prediction + noise

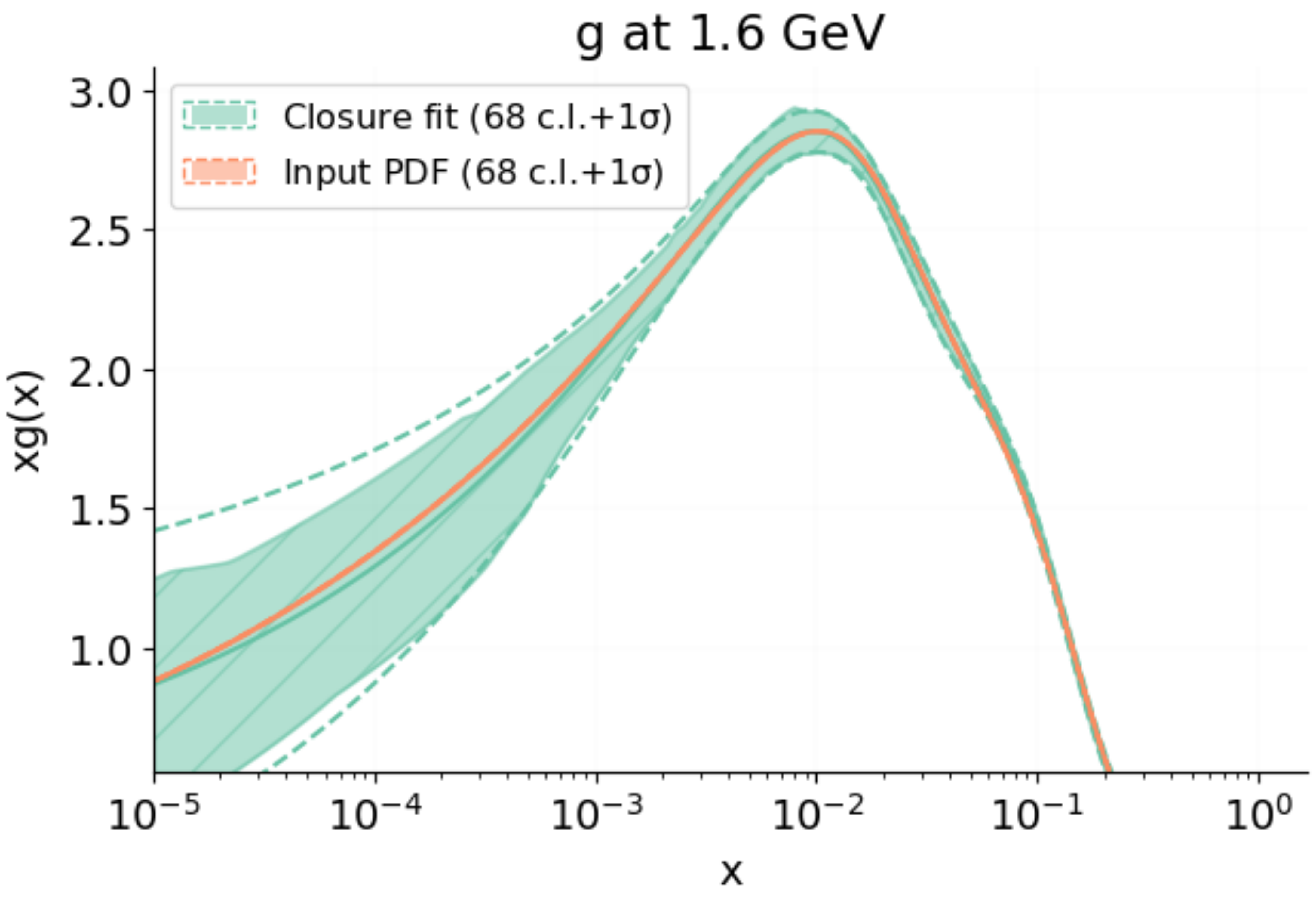
Input PDFs $\otimes$ **Theory prediction grids** → **pseudodata** → **methodology** → **extracted PDF**

Do they agree?*

Uncertainty validation: closure tests

[Del Debio, Giani, Wilson, 2111.05787]

Look at PDF: this seems okay



More quantitative: is the input data within 1 sigma of the prediction 68% of the time?

Use statistical measures to answer this

Recently the impact of **inconsistent data** was studied in a closure test

[Barontini et al., 2503.17447]

Everything is open source!

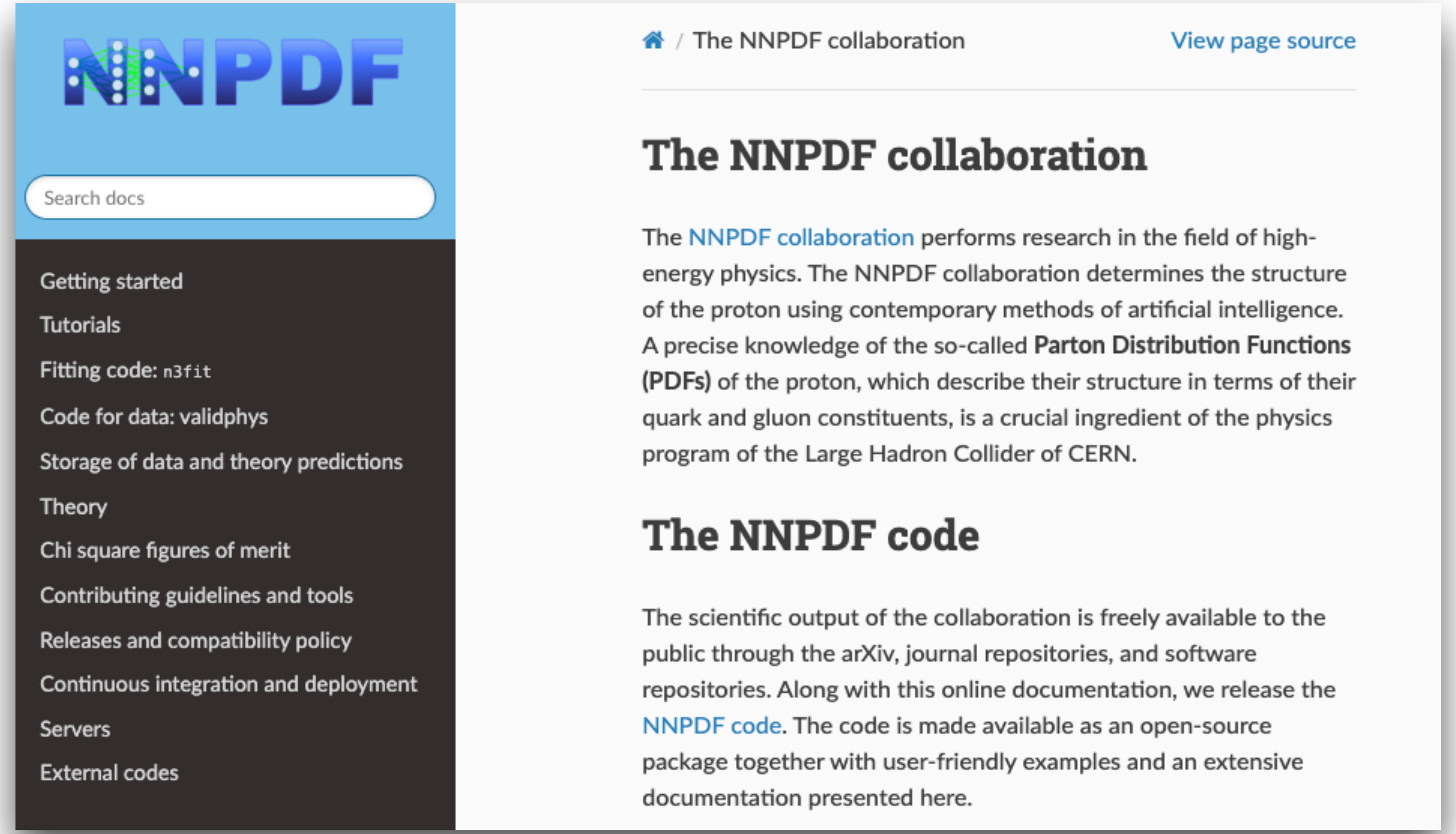
The NNPDF code is developed in a public repository

Everything to do your own PDF fit is open source:

- Data
- Theory grids
- Fitting methodology
- Analysis

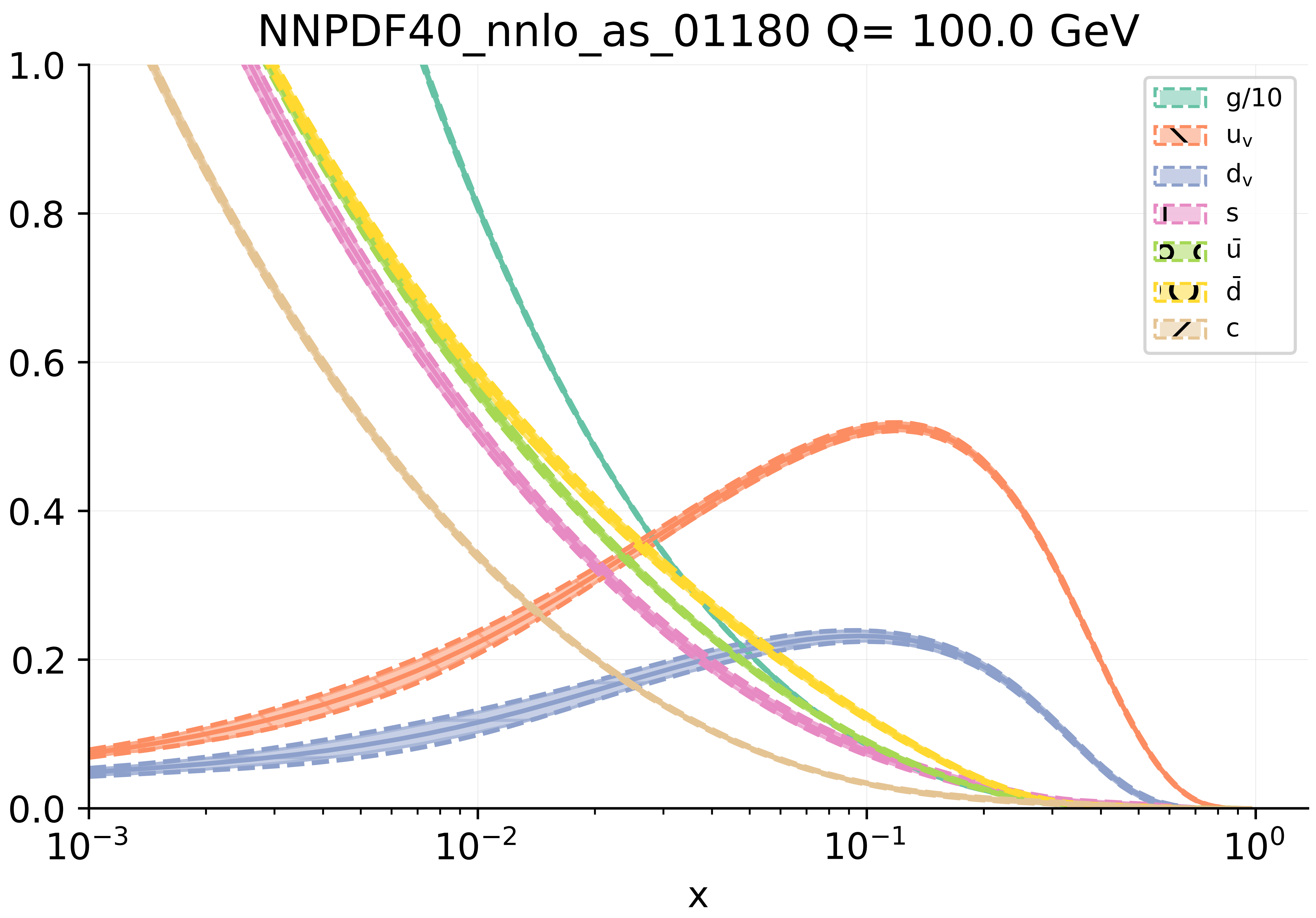
GitHub: github.com/NNPDF/nnpdf

Documentations: docs.nnpdf.science/



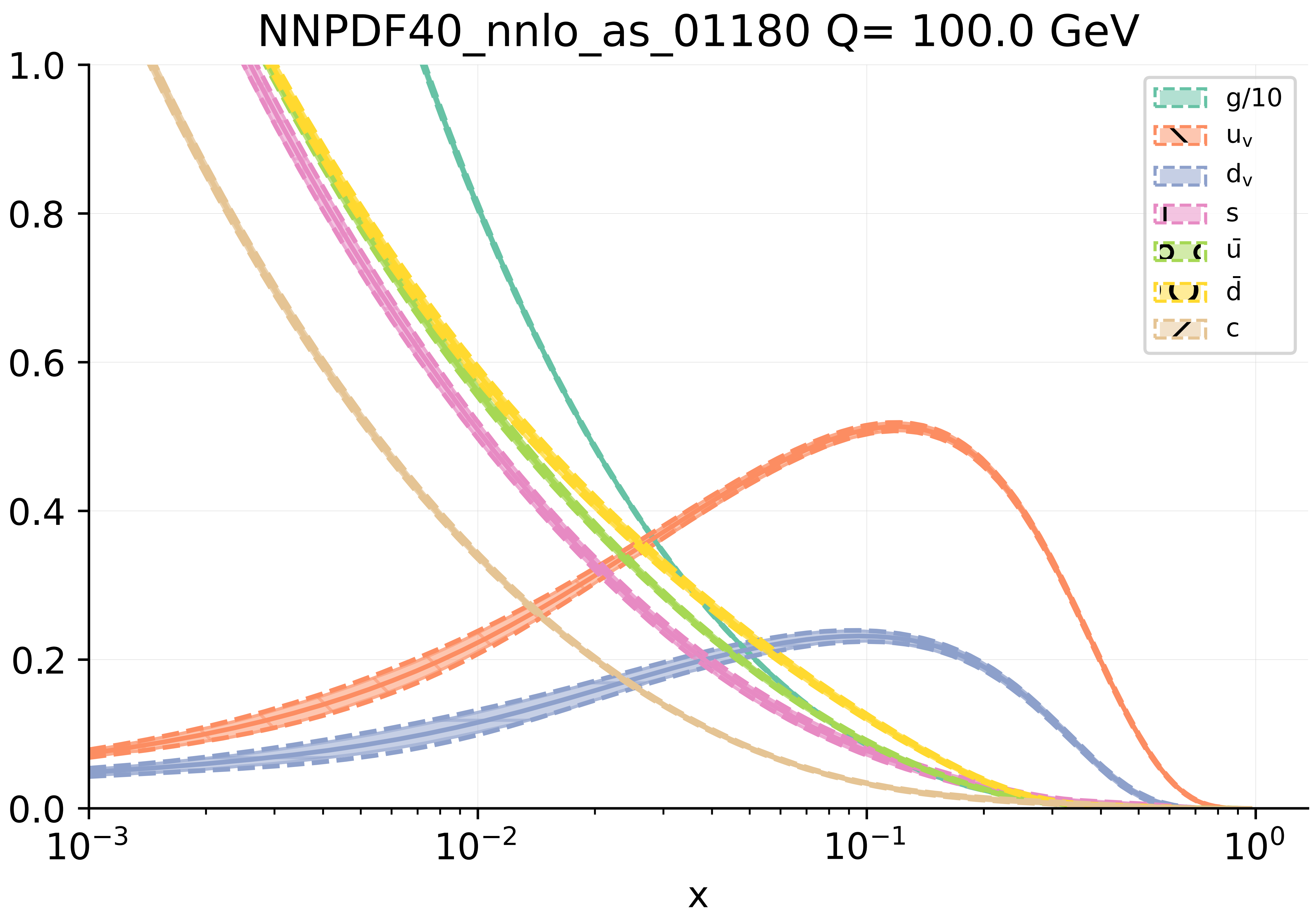
Summary and Outlook

- PDF determination is a Machine Learning challenge
- Hyperparameter tuning is an important step in selecting good ML models
- GPU optimisation has led to 90% reduction in energy cost …
- … and enables us to do hyperoptimisation based on PDF distributions rather than a single replica



Summary and Outlook

- PDF determination is a Machine Learning challenge
- Hyperparameter tuning is an important step in selecting good ML models
- GPU optimisation has led to 90% reduction in energy cost …
- … and enables us to do hyperoptimisation based on PDF distributions rather than a single replica



Thank you for your attention!

Backup slides