QCD at the Femtoscale in the Era of Big Data

Shaping PDFs analysis with Neural Networks

Chiara Bissolotti

Argonne National Laboratory

Argonne NATIONAL LABORATORY

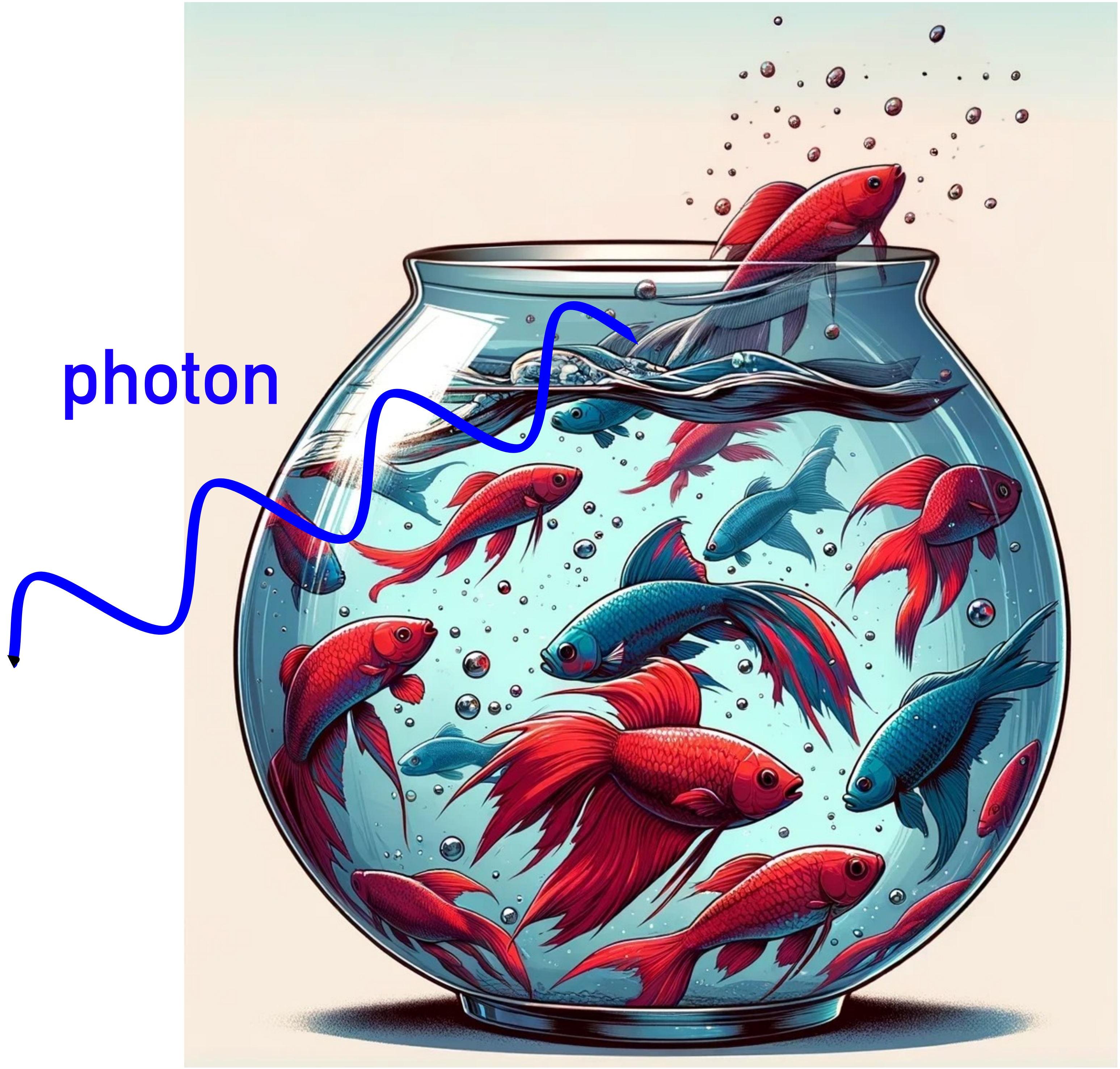
In collaboration with

Valerio Bertone, Radja Boughezal, Frank Petriello

Problem statement

QCD at the femtoscale

can not directly observe partons due to confinement,
a fundamental property of QCD



instead, we rely on indirect methods,
such as deep inelastic scattering experiments,
to infer their distributions

inverse problem

Parton distributions

momentum

position

$$\rho\left(x, \mathbf{k}_T, \mathbf{b}_T\right)$$

$$\int d^2\mathbf{b}_T$$

Transverse Momentum Distributions

TMDs

$$\int d^2\mathbf{k}_T$$

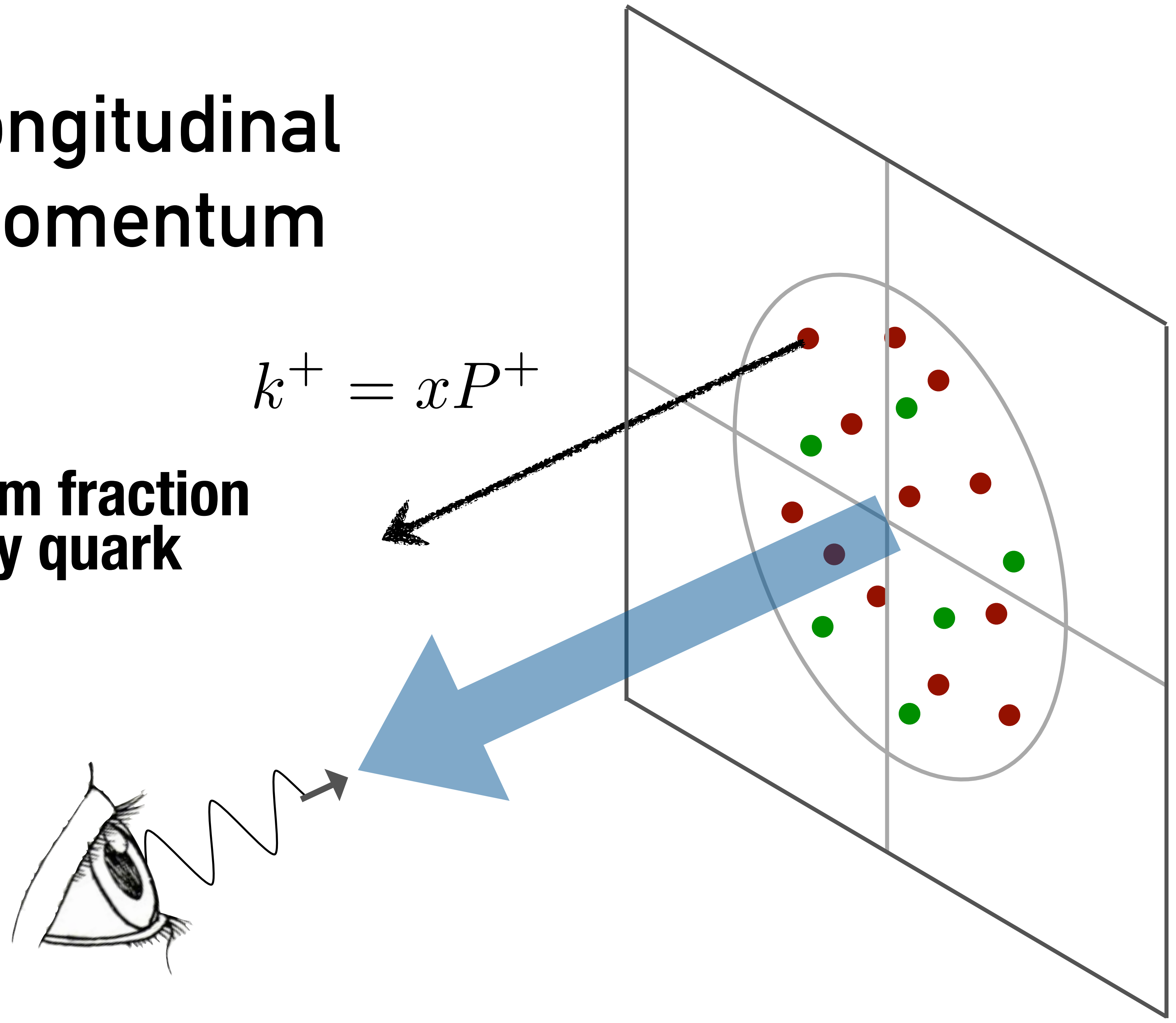
Parton Distribution Functions

PDFs

Longitudinal momentum

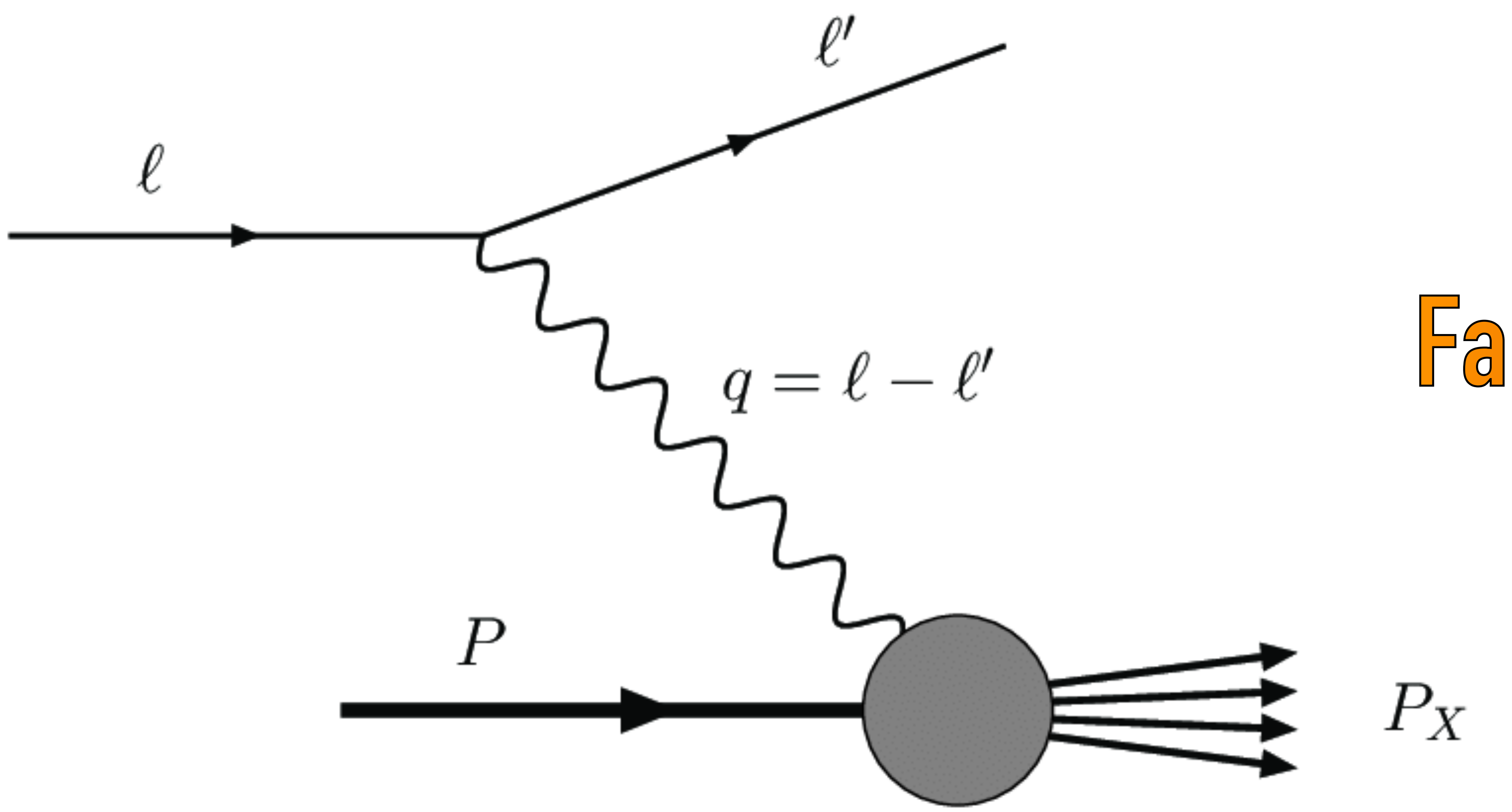
$k^+ = xP^+$

x: momentum fraction carried by quark



Collinear Parton Distribution Functions

Deep Inelastic Scattering



PDF

convolution integral $\otimes = \int_0^1 dx$

$$\sigma = \text{PDFs} \otimes C$$

Factorization theorem

Short-distance,
process dependent coefficient function

$$\sigma = \sum_{i=q,\bar{q},g} \int dx \, \boxed{f_i(x;Q^2)} \, \hat{\sigma}_i(x;Q^2)$$

partonic cross section

fraction of proton's momentum carried by the parton x

momentum transfer squared,
related to the resolution at which the proton's structure is probed Q^2

PDF fit scheme

parameterize PDFs

at a starting scale Q_0^2

evolve to $Q^2 > Q_0^2$ corresponding to Q_0^2 of data

QCD evolution

compute cross section

calculate σ_{DIS}

[for the purpose of this talk QCD evolution and perturbative calculations are treated as a black box]

compare with

available data

compute χ^2

minimize with respect to PDF parameters

Data and observables

we aim to
replicate the results of **HERAPDF2.0**

❋ reduced cross sections

- neutral current DIS
- charged current DIS

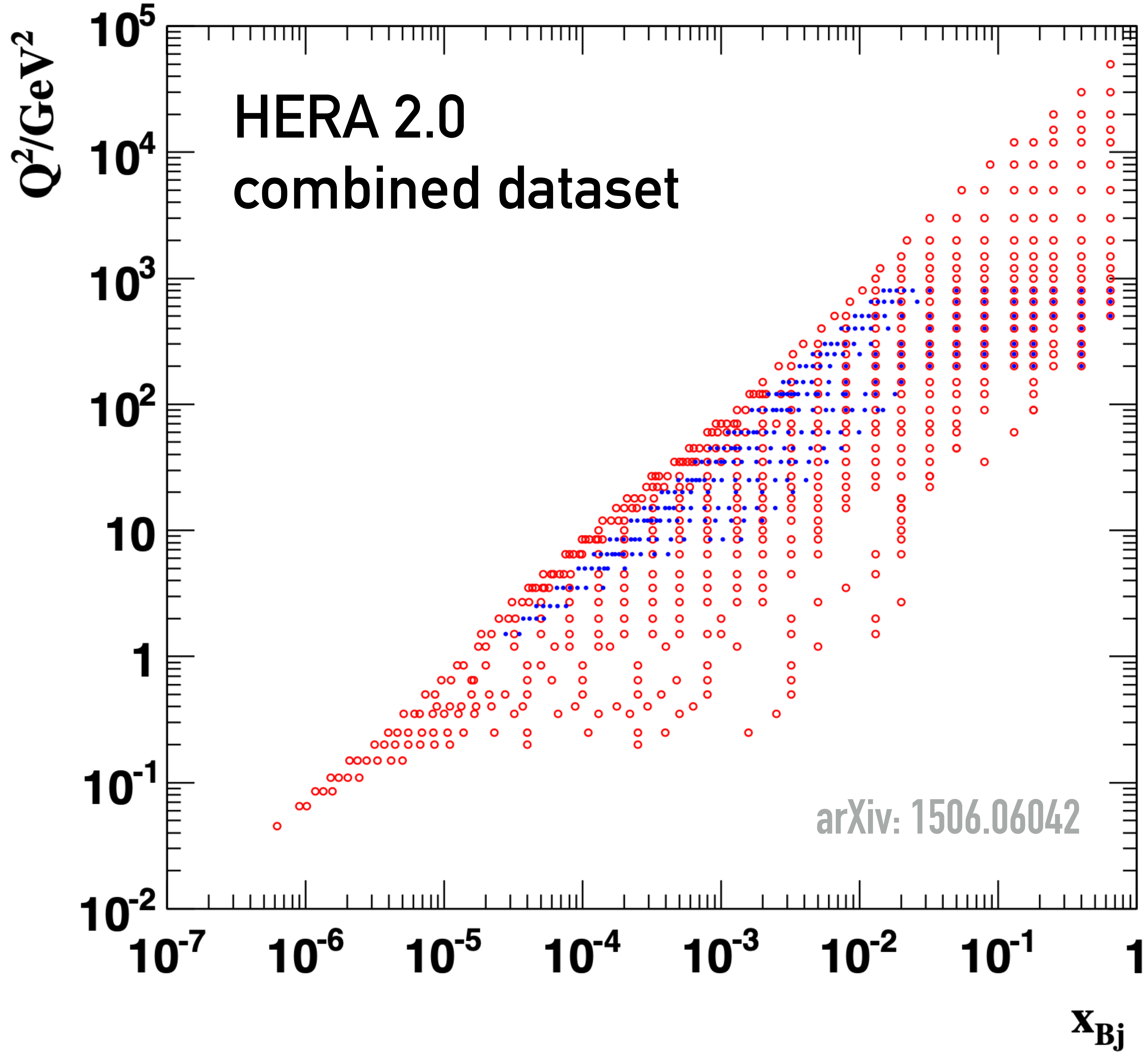
this fit

cuts: $Q > 3\,\mathrm{GeV}\,,\ W > 1.5\,\mathrm{GeV}$

conservative cuts

Total number of points: 1016

H1 and ZEUS



The points of the two grids used for the combination

data with $\sqrt{s} = 318\,\mathrm{GeV}$
data with $\sqrt{s} = 251\,\mathrm{GeV}$ or $\sqrt{s} = 225\,\mathrm{GeV}$

Statistics

Monte Carlo replicas

Gaussian assumption

$$\mathcal{G}(\mathbf{x}^{(k)}) \propto \exp\left[-(\mathbf{x}^{(k)} - \boldsymbol{\mu})^T \mathbf{C}^{-1} (\mathbf{x}^{(k)} - \boldsymbol{\mu})\right]$$

Covariance matrix

$$C_{ij} = \delta_{ij}\sigma^2_{i,\mathrm{unc}} + \sum_\beta \sigma^{(\beta)}_{i,\mathrm{corr}} \sigma^{(\beta)}_{j,\mathrm{corr}}$$

Cholesky decomposition

$$\mathbf{C} = \mathbf{L} \cdot \mathbf{L}^T$$

Replica generation

$$\mathbf{x}^{(k)} = \boldsymbol{\mu} + \mathbf{L} \cdot \mathbf{r}^{(k)}$$

r standard normal random variables

each replica fitted independently

$$\frac{1}{N_{\mathrm{rep}}} \sum_k x_i^{(k)} \simeq \mu_i \qquad \frac{1}{N_{\mathrm{rep}}} \sum_k x_i^{(k)} x_j^{(k)} \simeq \mu_i \mu_j + C_{ij}$$

Statistics

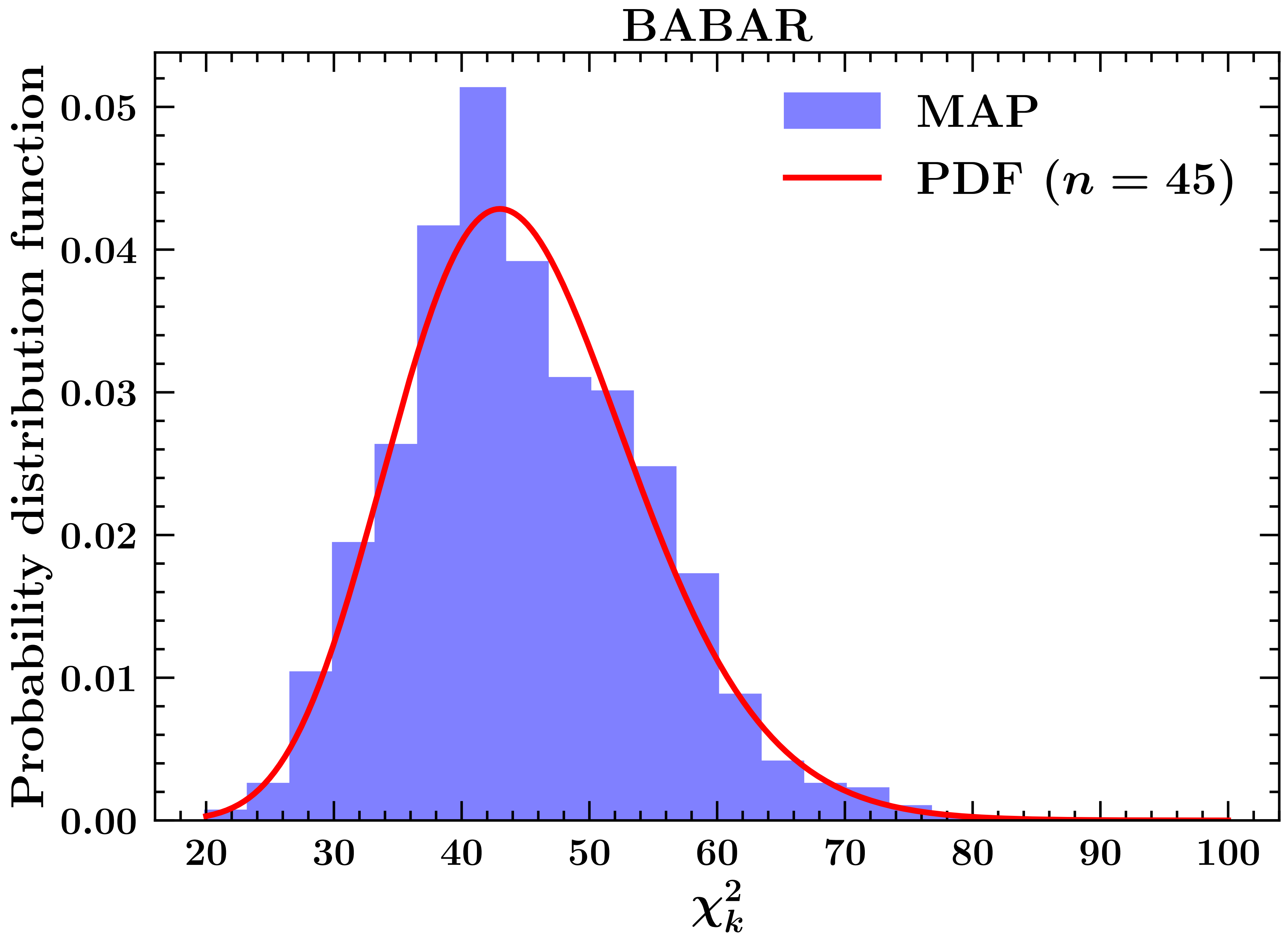
Monte Carlo replicas

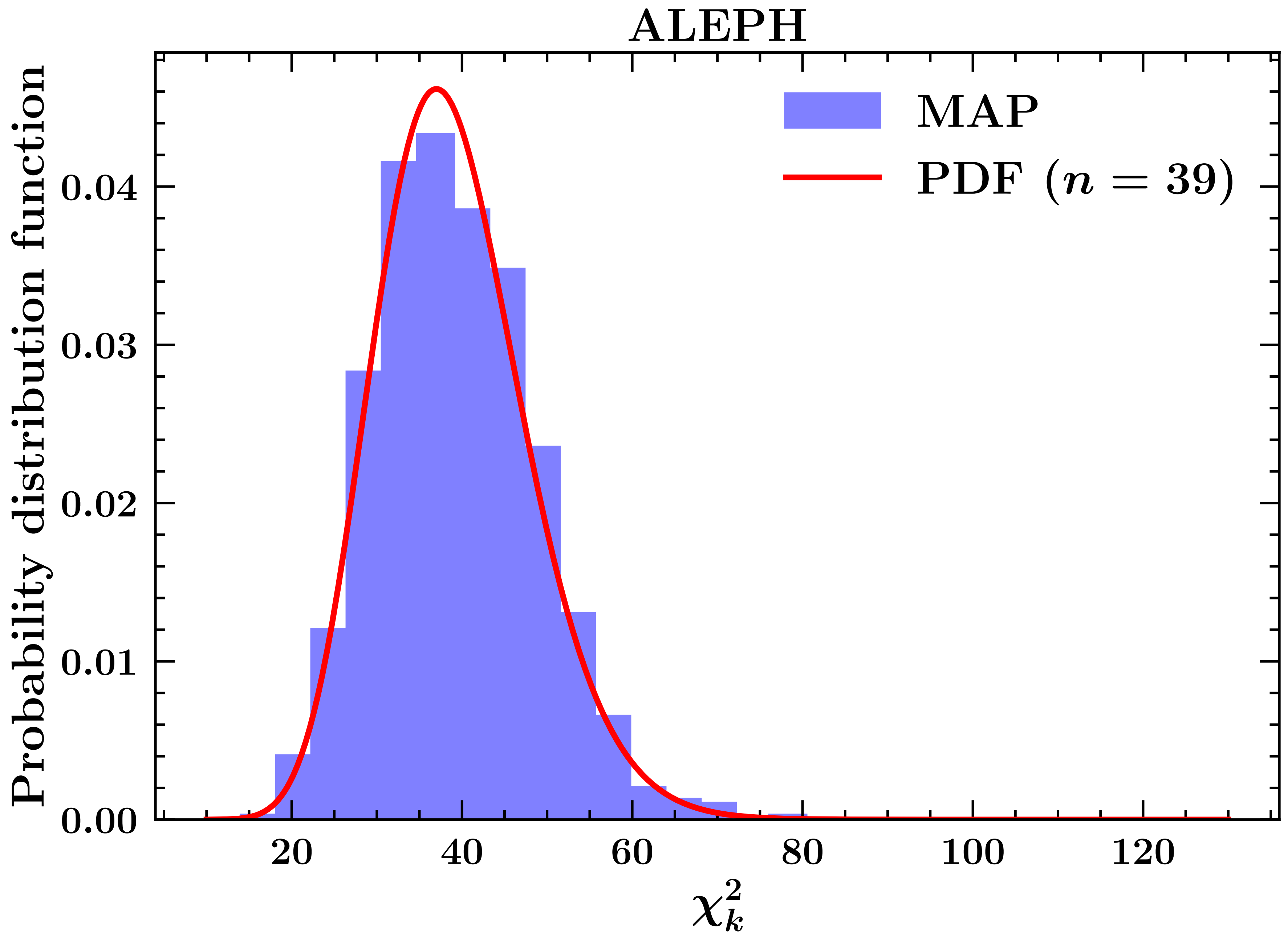
Gaussian assumption

$$\mathcal{G}(\mathbf{x}^{(k)}) \propto \exp\left[-(\mathbf{x}^{(k)}-\boldsymbol{\mu})^T\mathbf{C}^{-1}(\mathbf{x}^{(k)}-\boldsymbol{\mu})\right]$$

Replica generation

$$\mathbf{x}^{(k)} = \boldsymbol{\mu} + \mathbf{L}\cdot\mathbf{r}^{(k)} \longrightarrow \frac{1}{N_{\mathrm{rep}}}\sum_k x_i^{(k)} \simeq \mu_i \qquad \frac{1}{N_{\mathrm{rep}}}\sum_k x_i^{(k)}x_j^{(k)} \simeq \mu_i\mu_j + C_{ij}$$





Methodology - MLE

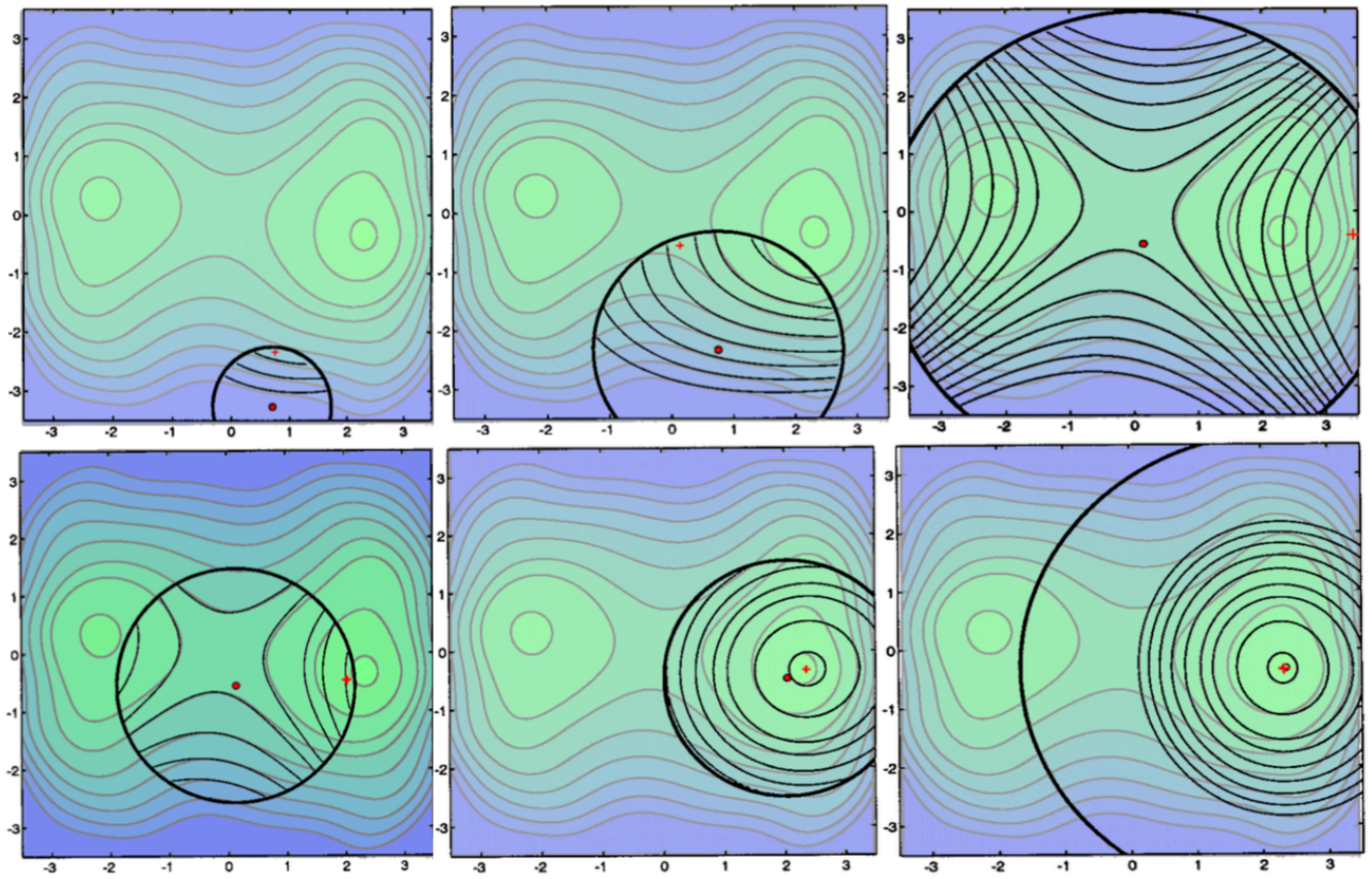
Maximum Likelihood Estimation

likelihood function

$$\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{d}) \equiv \mathcal{P}(\boldsymbol{d}|\boldsymbol{\theta}) = \frac{1}{\sqrt{(2\pi)^n|\mathbf{C}|}} \exp\left(-\frac{1}{2}[\boldsymbol{t}(\boldsymbol{\theta}) - \boldsymbol{d}]^T \mathbf{C}^{-1} [\boldsymbol{t}(\boldsymbol{\theta}) - \boldsymbol{d}]\right)$$

$$\max_{\boldsymbol{\theta}} \ln\left(\mathcal{L}(\boldsymbol{\theta}|\boldsymbol{d})\right) \rightarrow \min_{\boldsymbol{\theta}} \left(\frac{1}{2}[\boldsymbol{t}(\boldsymbol{\theta}) - \boldsymbol{d}]^T \mathbf{C}^{-1} [\boldsymbol{t}(\boldsymbol{\theta}) - \boldsymbol{d}]\right) \equiv \frac{1}{2} \min_{\boldsymbol{\theta}} \chi^2(\boldsymbol{\theta})$$

minimization carried out by **ceres-solver**



trust region

Levenberg–Marquardt

combines the gradient-descent algorithm to the Gauss-Newton method

need the knowledge of the derivatives of the χ^2 with respect to the parameters $\boldsymbol{\theta}$

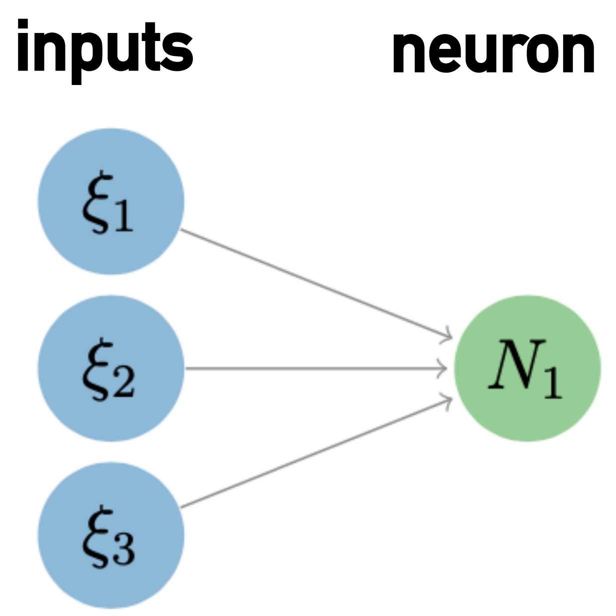
analytic derivatives

provided by NNAD

Methodology - Neural Network

NNAD C++ Library: arXiv:2005.07039

Basis



weights ↑ activation function ↑

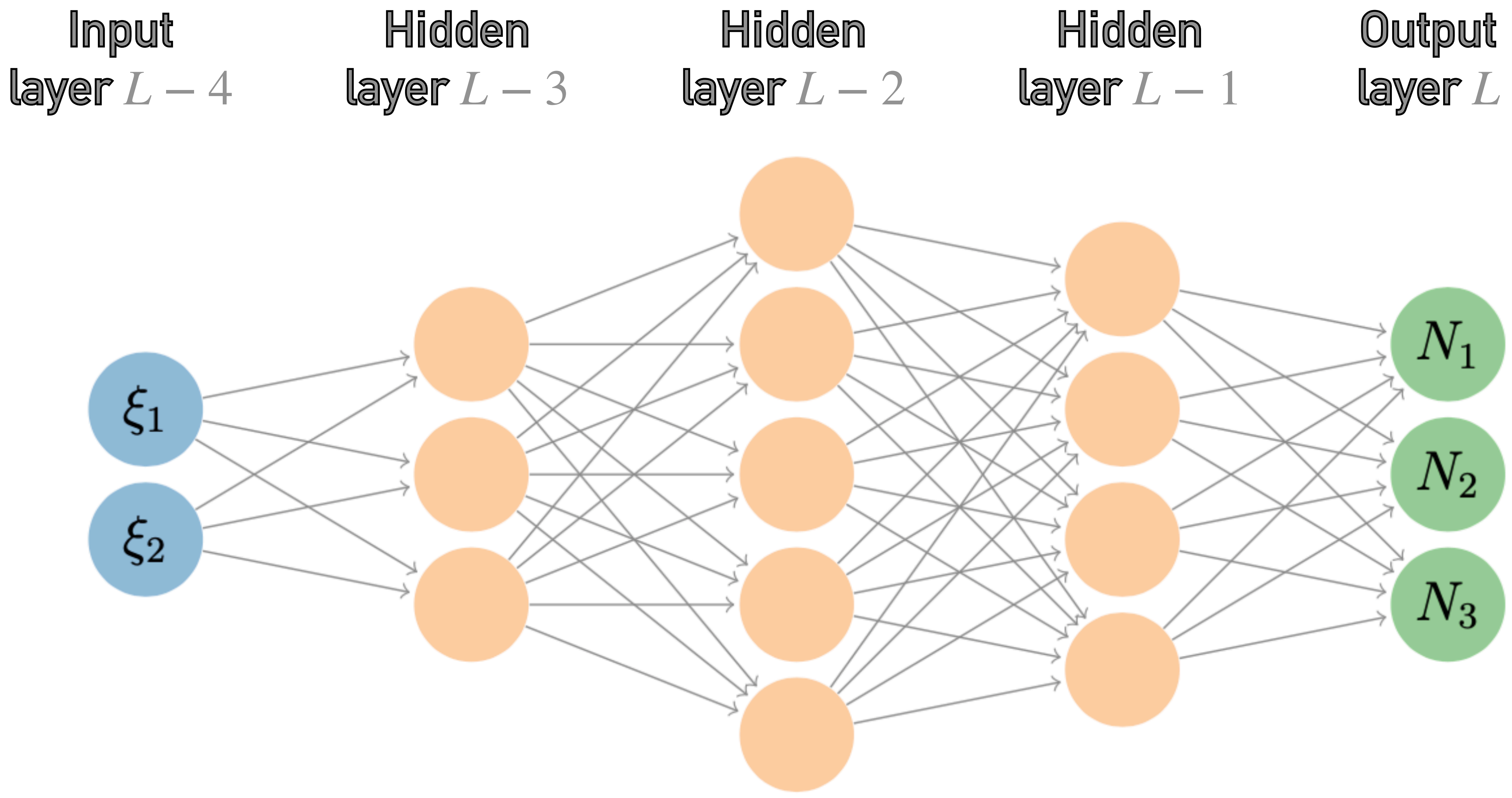
$$N_1(\boldsymbol{\xi};\{w_{1j},\theta_1\}) = \phi\left(\sum_{j=1}^{3} w_{1j}\xi_j + \theta_1\right)$$

↓ bias

Defining:

$$x_i^{(\ell)} = \sum_{j=1}^{N_{\ell-1}} \omega_{ij}^{(\ell)} y_j^{(\ell-1)} + \theta_i^{(\ell)}$$

$$z_i^{(\ell)} = \Phi'_\ell\left(x_i^{(\ell)}\right) \qquad y_i^{(\ell)} = \phi_\ell\left(x_i^{(\ell)}\right)$$



Forward propagation

in a fully connected NN

$$N_k(\boldsymbol{\xi};\{\omega_{ij}^{(\ell)},\theta_i^{(\ell)}\}) = \phi_L\left(\sum_{j^{(1)}}^{N_{L-1}} \omega_{kj^{(1)}}^{(L)} y_{j^{(1)}}^{(L-1)} + \theta_k^{(L)}\right)$$

$$= \phi_L\left(\sum_{j^{(1)}=1}^{N_{L-1}} \omega_{kj^{(1)}}^{(L)} \phi_{L-1}\left(\sum_{j^{(2)}=1}^{N_{L-2}} \omega_{j^{(1)}j^{(2)}}^{(L)} y_{j^{(2)}}^{(L-2)} + \theta_{j^{(1)}}^{(L-1)}\right) + \theta_k^{(L)}\right) \qquad = \dots$$

NN Analytic Derivatives - NNAD

Simplified case

theory prediction as a convolution

$$\hat{f}(\xi,\zeta) = \sum_{i=1} [C_i \otimes_\xi N_i(\xi,\zeta)]$$

chi-square formula

$$\chi^2\left\{\omega_{ij}^{(\ell)},\theta_i^{(\ell)}\right\} = \sum_{k=1}^{N_{\rm data}} \left(\frac{\hat{f}(\xi_k,\zeta_k;\{\omega_{ij}^{(\ell)},\theta_i^{(\ell)}\}) - d_k}{\sigma_k}\right)^2$$

the computation of the gradient of the χ^2 requires the derivatives

$$\frac{\partial\chi^2}{\partial\omega_{ij}^{(\ell)}} \qquad \frac{\partial\chi^2}{\partial\theta_i^{(\ell)}}$$

thanks to the feed forward nature of the NN
it's possibile to write these derivatives in a closed form

NN Analytic Derivatives

computation of the gradient of the χ^2

$$\hat{f}(\xi,\zeta) = \sum_{i=1} [C_i \otimes_\xi N_i(\xi,\zeta)]$$

$$\frac{\partial \chi^2}{\partial \omega_{ij}^{(\ell)}} = 2\sum_{k=1}^{N_{\mathrm{data}}} \left(\frac{[\mathbf{C} \otimes_\xi \mathbf{N}]\,(\xi_k,\zeta_k) - d_k}{\sigma_k^2} \right) \left[\mathbf{C} \otimes_\xi \frac{\partial \mathbf{N}}{\partial \omega_{ij}^{(\ell)}} \right] (\xi_k,\zeta_k),$$

$$\frac{\partial \chi^2}{\partial \theta_{i}^{(\ell)}} = 2\sum_{k=1}^{N_{\mathrm{data}}} \left(\frac{[\mathbf{C} \otimes_\xi \mathbf{N}]\,(\xi_k,\zeta_k) - d_k}{\sigma_k^2} \right) \left[\mathbf{C} \otimes_\xi \frac{\partial \mathbf{N}}{\partial \theta_{i}^{(\ell)}} \right] (\xi_k,\zeta_k),$$

reduce the computation of the derivatives of the χ^2 to the computation of the derivatives of the NN

backward propagation

apply chain rule, starting from the output layer all the way back

NN Analytic Derivatives

computation of the gradient of the χ^2

$$\hat{f}(\xi,\zeta) = \sum_{i=1} [C_i \otimes_\xi N_i(\xi,\zeta)]$$

$$\frac{\partial \chi^2}{\partial \omega_{ij}^{(\ell)}} = 2 \sum_{k=1}^{N_{\rm data}} \left(\frac{[\mathbf{C} \otimes_\xi \mathbf{N}](\xi_k,\zeta_k) - d_k}{\sigma_k^2} \right) \left[\mathbf{C} \otimes_\xi \frac{\partial \mathbf{N}}{\partial \omega_{ij}^{(\ell)}} \right] (\xi_k,\zeta_k),$$

$$\begin{aligned}
\frac{\partial N_k}{\partial \omega_{ij}^{(\ell)}} &= \frac{\partial y_k^{(L)}}{\partial \omega_{ij}^{(\ell)}} \\
&= z_k^{(L)} \frac{\partial x_k^{(L)}}{\partial \omega_{ij}^{(\ell)}} \\
&= \sum_{j^{(1)}=1}^{N_{L-1}} \left[z_k^{(L)} \omega_{kj^{(1)}}^{(L)} \right] \frac{\partial y_{j^{(1)}}^{(L-1)}}{\partial \omega_{ij}^{(\ell)}} \\
&= \sum_{j^{(1)}=1}^{N_{L-1}} \sum_{j^{(2)}=1}^{N_{L-2}} \boxed{\left[z_k^{(L)} \omega_{kj^{(1)}}^{(L)} \right]} \left[z_{j^{(1)}}^{(L-1)} \omega_{j^{(1)}j^{(2)}}^{(L-1)} \right] \frac{\partial y_{j^{(2)}}^{(L-2)}}{\partial \omega_{ij}^{(\ell)}} \\
&= \dots
\end{aligned}$$

can be computed explicitly

backward propagation

apply chain rule, starting from the output layer all the way back

defining:

$$\mathbf{\Sigma}^{(\ell)} \equiv \prod_{\alpha=L}^{\ell+1} \boldsymbol{S}^{(\alpha)}, \quad S_{ij}^{(\ell)} \equiv \boxed{z_i^{(\ell)} \omega_{ij}^{(\ell)}}$$

NN Analytic Derivatives

computation of the gradient of the χ^2

$$\frac{\partial \chi^2}{\partial \omega_{ij}^{(\ell)}} = 2 \sum_{k=1}^{N_{\rm data}} \left(\frac{\left[\mathbf{C} \otimes_\xi \mathbf{N}\right](\xi_k, \zeta_k) - d_k}{\sigma_k^2} \right) \left[\mathbf{C} \otimes_\xi \frac{\partial \mathbf{N}}{\partial \omega_{ij}^{(\ell)}} \right] (\xi_k, \zeta_k),$$

$$\frac{\partial \chi^2}{\partial \theta_{i}^{(\ell)}} = 2 \sum_{k=1}^{N_{\rm data}} \left(\frac{\left[\mathbf{C} \otimes_\xi \mathbf{N}\right](\xi_k, \zeta_k) - d_k}{\sigma_k^2} \right) \left[\mathbf{C} \otimes_\xi \frac{\partial \mathbf{N}}{\partial \theta_{i}^{(\ell)}} \right] (\xi_k, \zeta_k),$$

chain rule

$$\frac{\partial N_k}{\partial \omega_{ij}^{(\ell)}} = \Sigma_{ki}^{(\ell)} z_i^{(\ell)} y_j^{(\ell-1)}$$

$$\frac{\partial N_k}{\partial \theta_{i}^{(\ell)}} = \Sigma_{ki}^{(\ell)} z_i^{(\ell)}$$

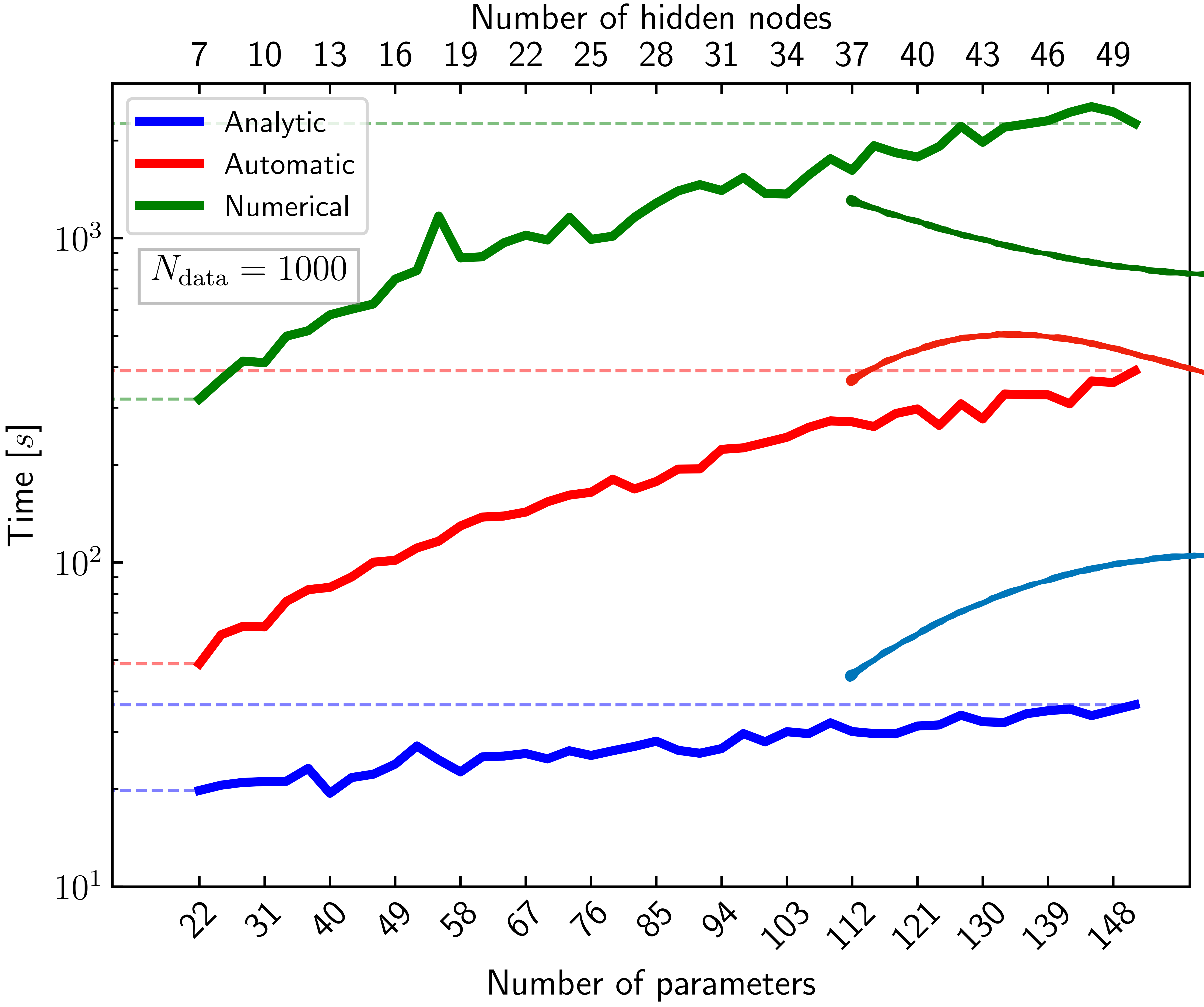
backward propagation

matrix $\mathbf{\Sigma}$ can be computed recursively moving backwards

allows to compute the derivatives w.r.t. all free parameters of a NN

NN Analytic Derivatives

Performance



differentiation methods as implemented in ceres-solver

NNAD

analytic back-propagation formula

performance advantage

Framework

C++ code - ‘NavyPier’

NavyPier

data, input cards with choices for NN,
main code that runs the fit

ceres-solver

minimization

NangaParbat

data handling,
χ^2, covariance matrix, uncertainties treatment
Monte Carlo replica generation, statistical analysis…

NNAD

Neural Network

$$\chi^{2(k)} \equiv \left(\mathbf{T}(\theta^{(k)}) - \mathbf{x}^{(k)}\right)^T \cdot \mathbf{C}^{-1} \cdot \left(\mathbf{T}(\theta^{(k)}) - \mathbf{x}^{(k)}\right)$$

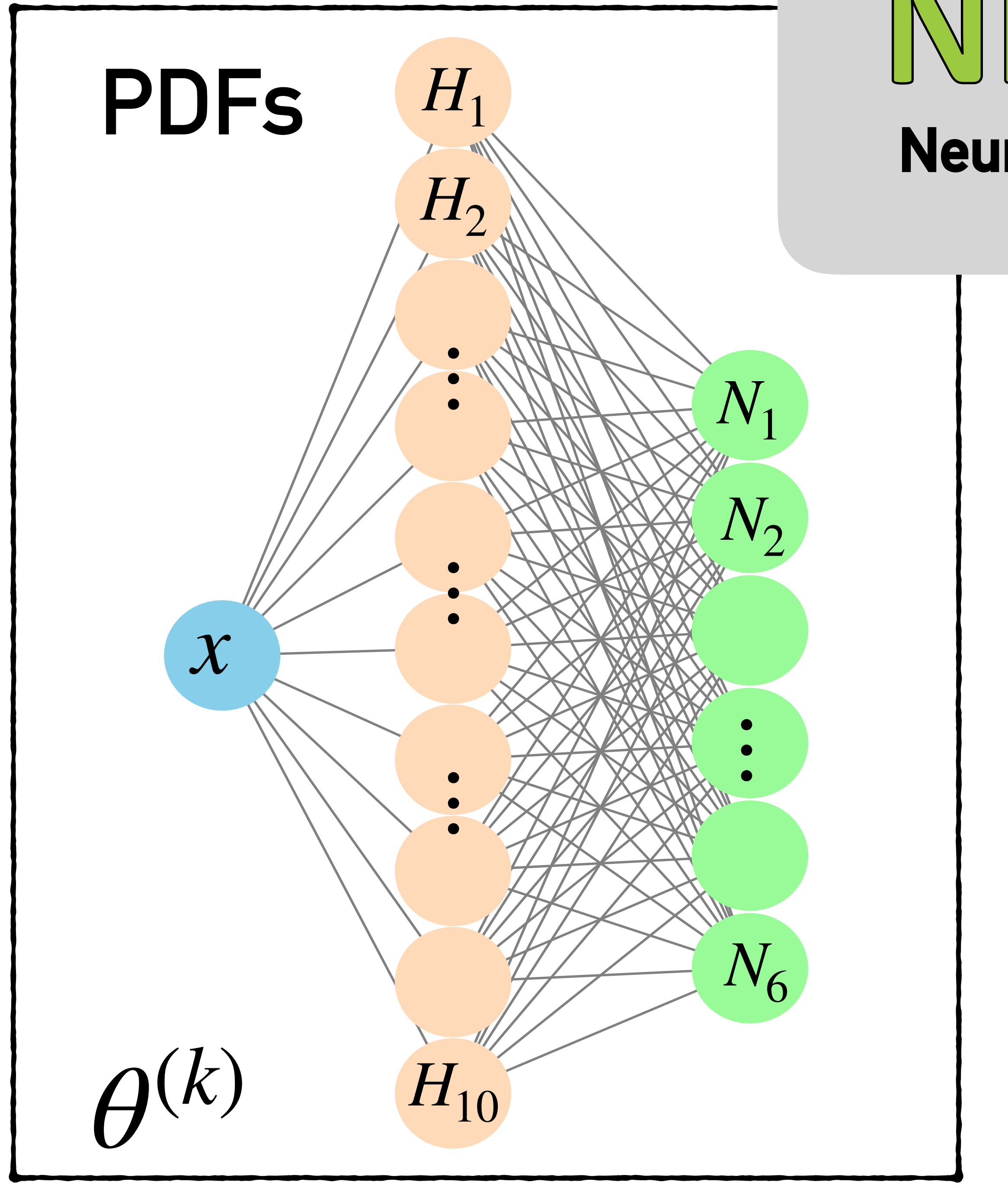
k: replica

$$\mathbf{T}(\theta^{(k)}) = \boxed{\hat{\sigma}} \otimes$$

theoretical prediction

apfel++

perturbative computations,
NNLO



Reproducing HERAPDF2.0 with NavyPier

‘Sort of’ benchmark

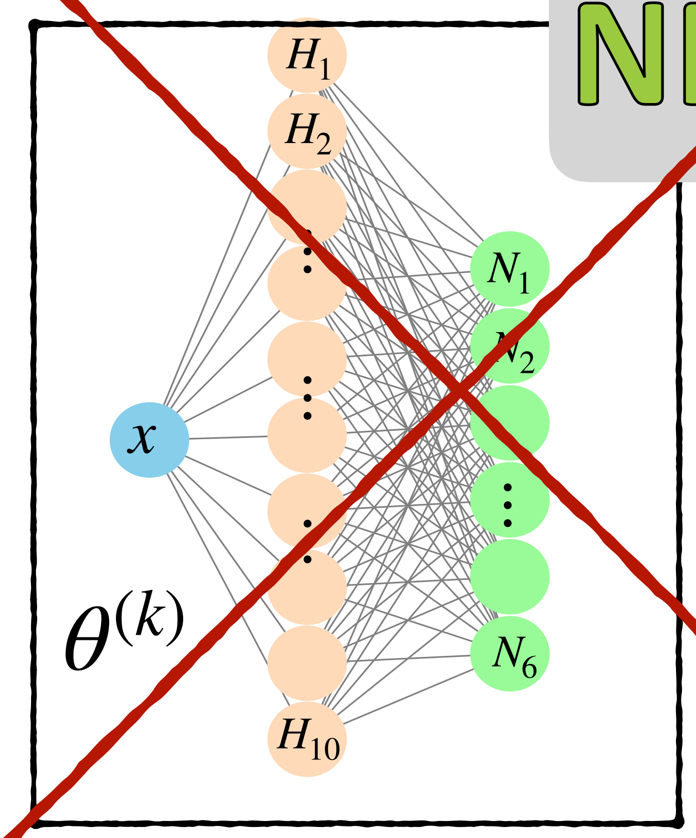
NavyPier

NangaParbat

$$\mathbf{T}(\theta^{(k)}) = \boxed{\hat{\sigma}} \otimes$$

theoretical prediction

apfelxx



NNAD

Reproducing HERAPDF2.0 with NavyPier

‘Sort of’ benchmark

NavyPier

NangaParbat

$$\mathbf{T}(\theta^{(k)}) = \boxed{\hat{\sigma}} \otimes$$

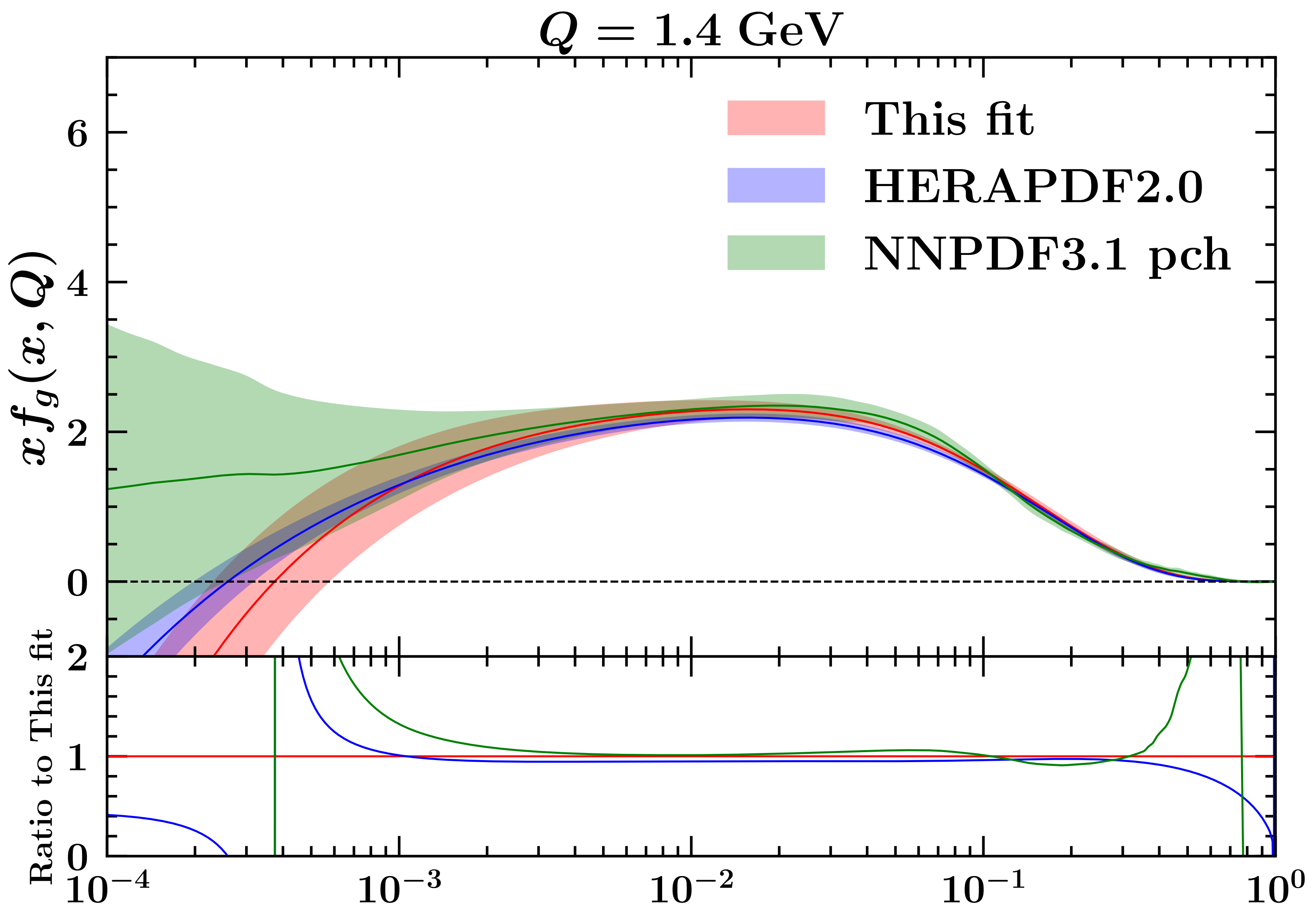
apfelxx

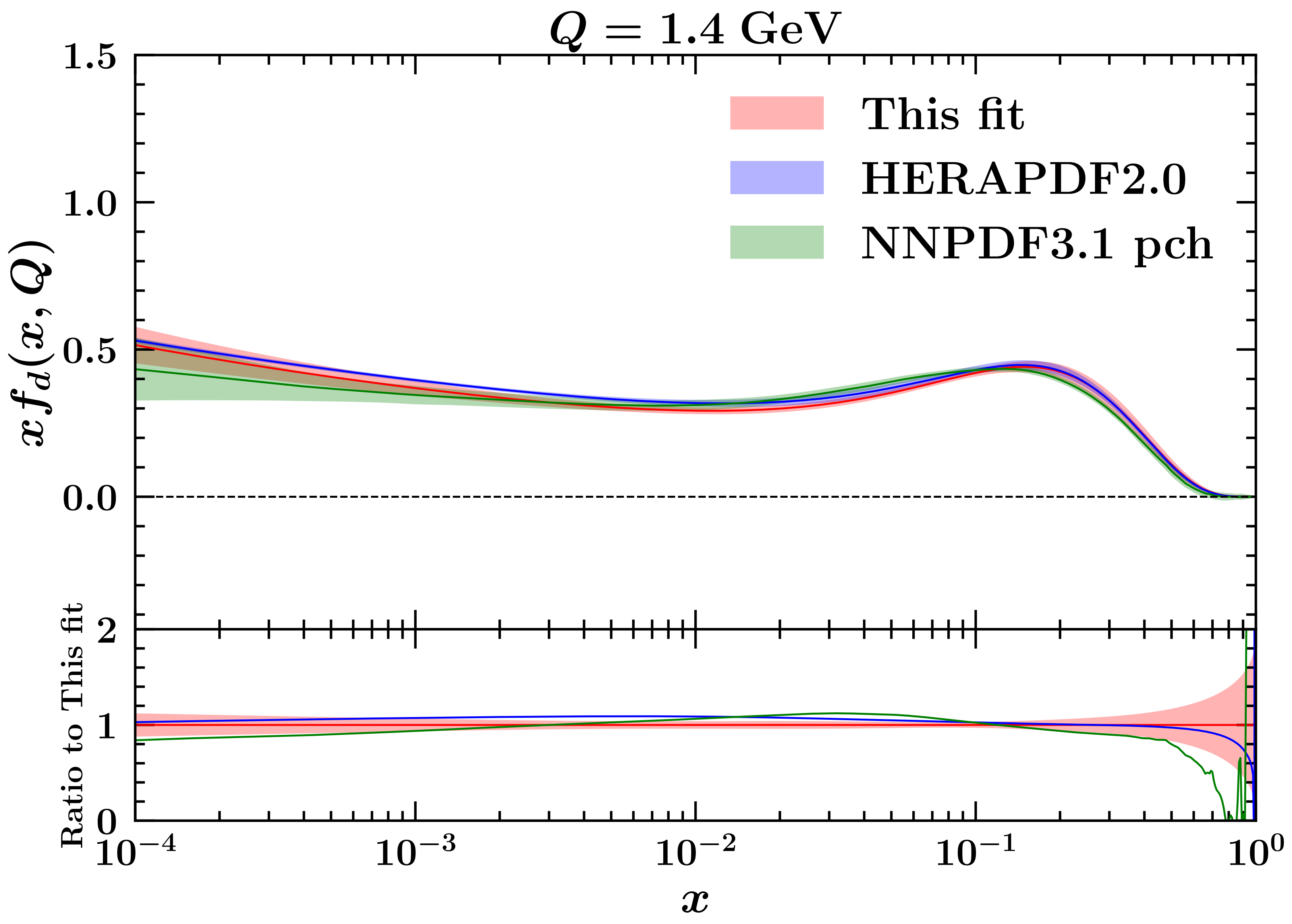
theoretical prediction

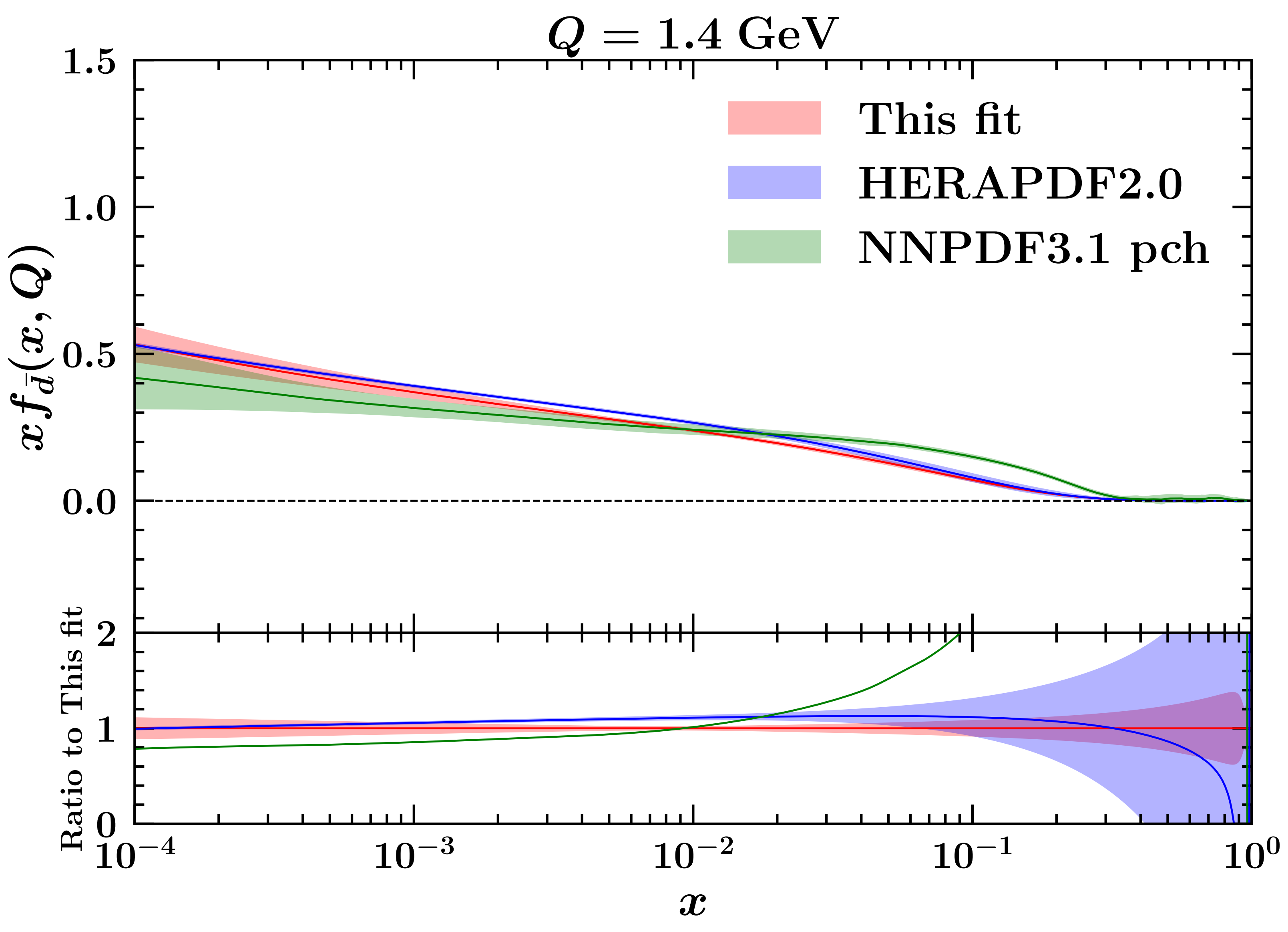
HERA parameterization

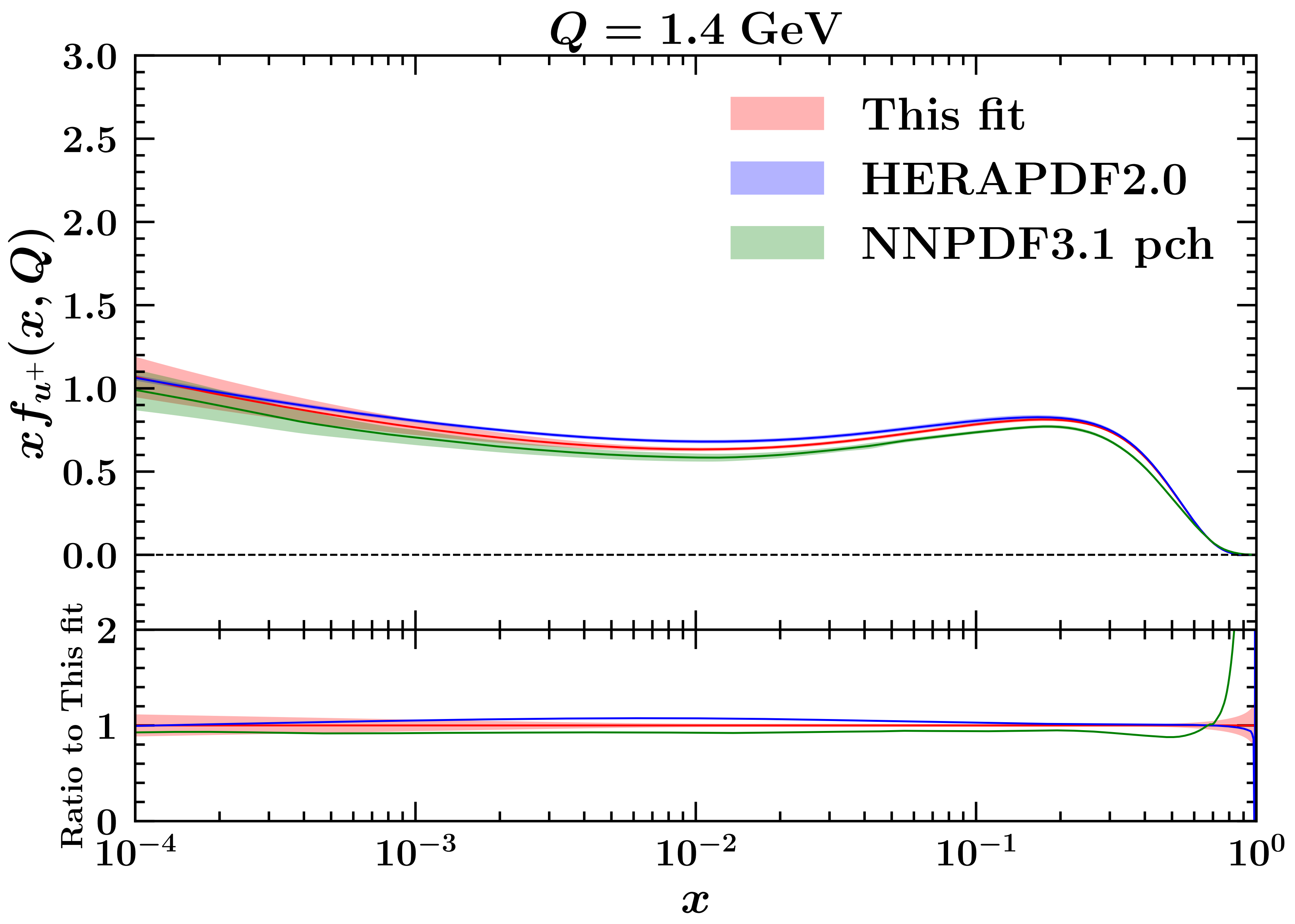
$$\begin{aligned}
xg(x) &= A_g x^{B_g}(1-x)^{C_g} - A_{gp}x^{B_{gp}}(1-x)^{C_{gp}}\,, \\
xu_v(x) &= A_{uv}x^{B_{uv}}(1-x)^{C_{uv}}\left(1+E_{uv}x^2\right)\,, \\
xd_v(x) &= A_{dv}x^{B_{dv}}(1-x)^{C_{dv}}\,, \\
x\bar{U}(x) &= A_{\bar{U}}x^{B_{\bar{U}}}(1-x)^{C_{\bar{U}}}(1+D_{\bar{U}}x)\,, \\
x\bar{D}(x) &= A_{\bar{D}}x^{B_{\bar{D}}}(1-x)^{C_{\bar{D}}}\,.
\end{aligned}$$

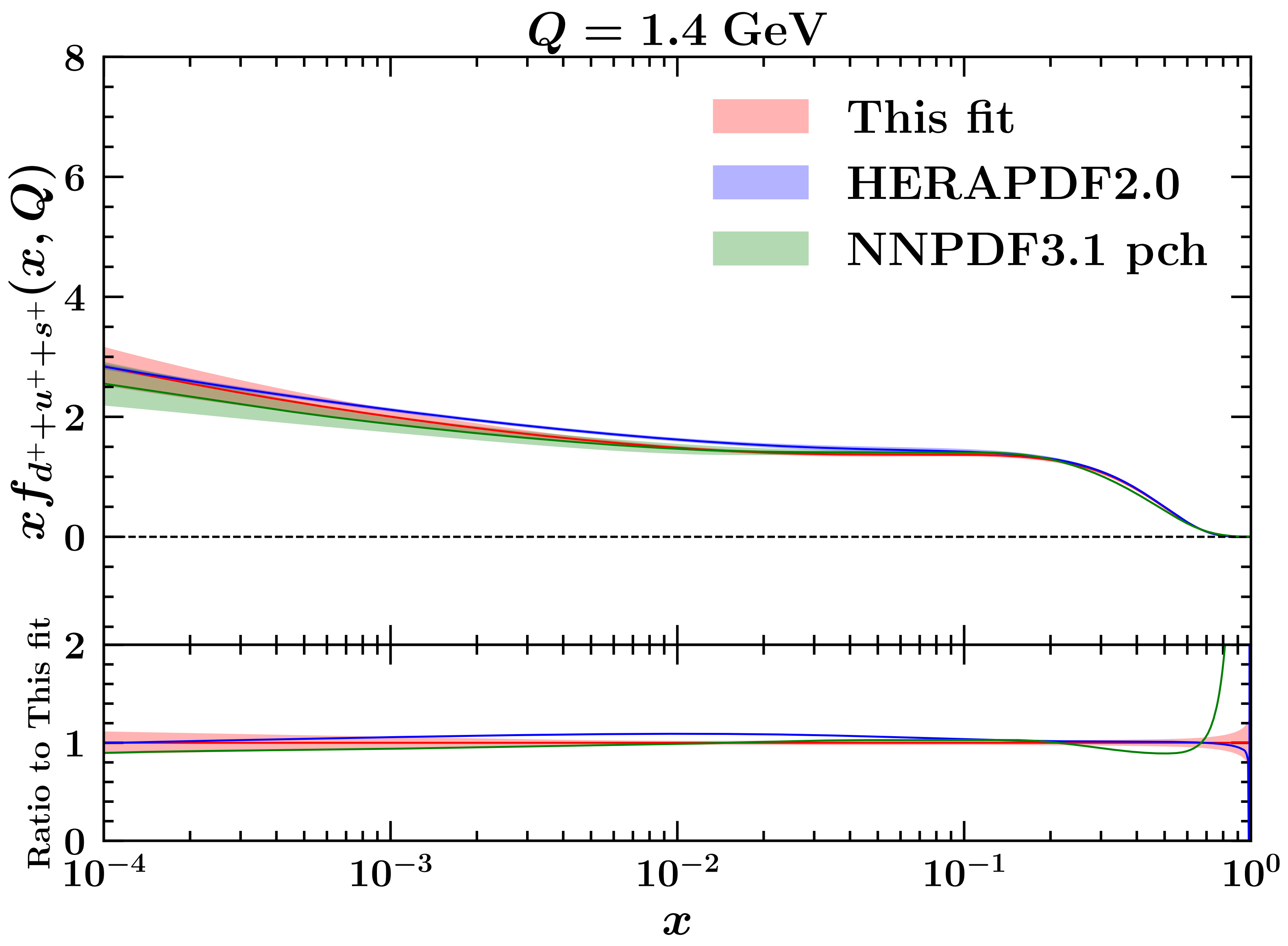
No NN, HERAPDF2.0-like









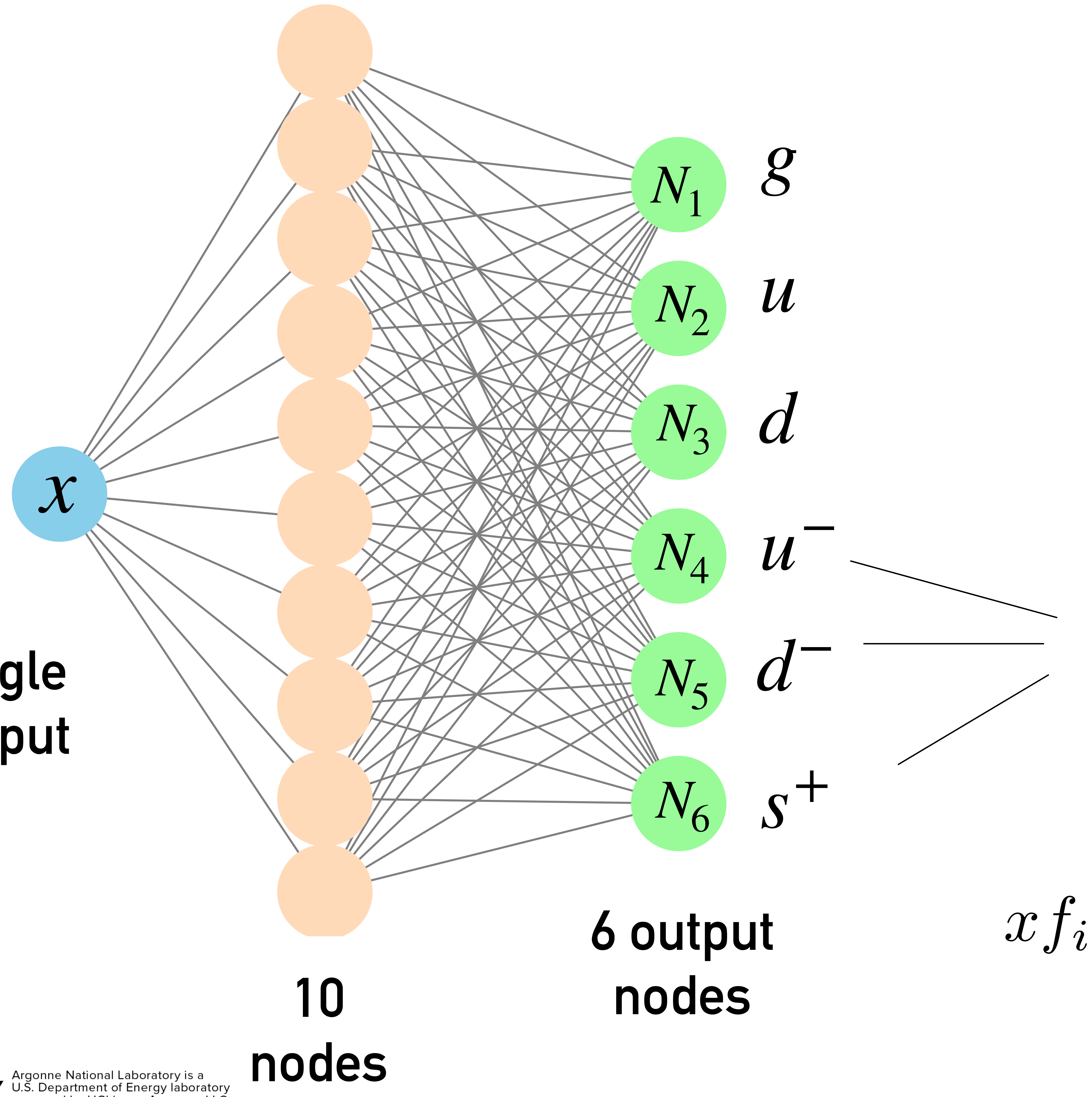


NNLO PDFs

Neural network structure

tentative 1 of n...

for PDF extraction



$$q^\pm \equiv q \pm \bar{q}$$

parameterization
no preprocessing function

$$x f_i\,(x, Q_0) = N N_i(x; \theta)$$

output of the NN, no factor in front

for $x \to 1$

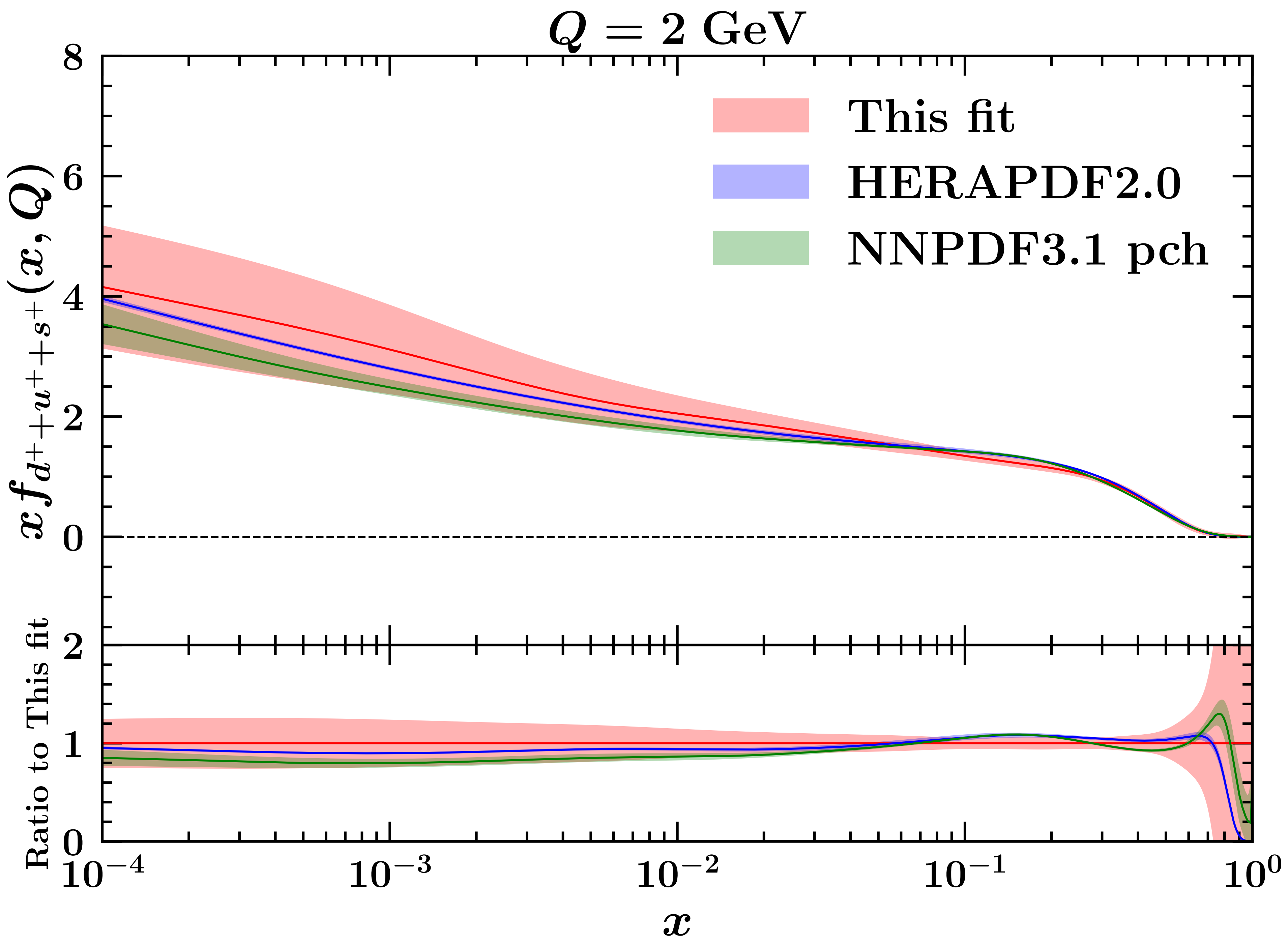
$$x f_i\,(x, \mu_0 = 1.4\ \mathrm{GeV}) = \left(N_i(x;\theta) - N_i(1;\theta)\right)^2$$

PDF extraction

NNLO

$$xf_i\,(x, Q_0) = NN_i(x;\theta)$$

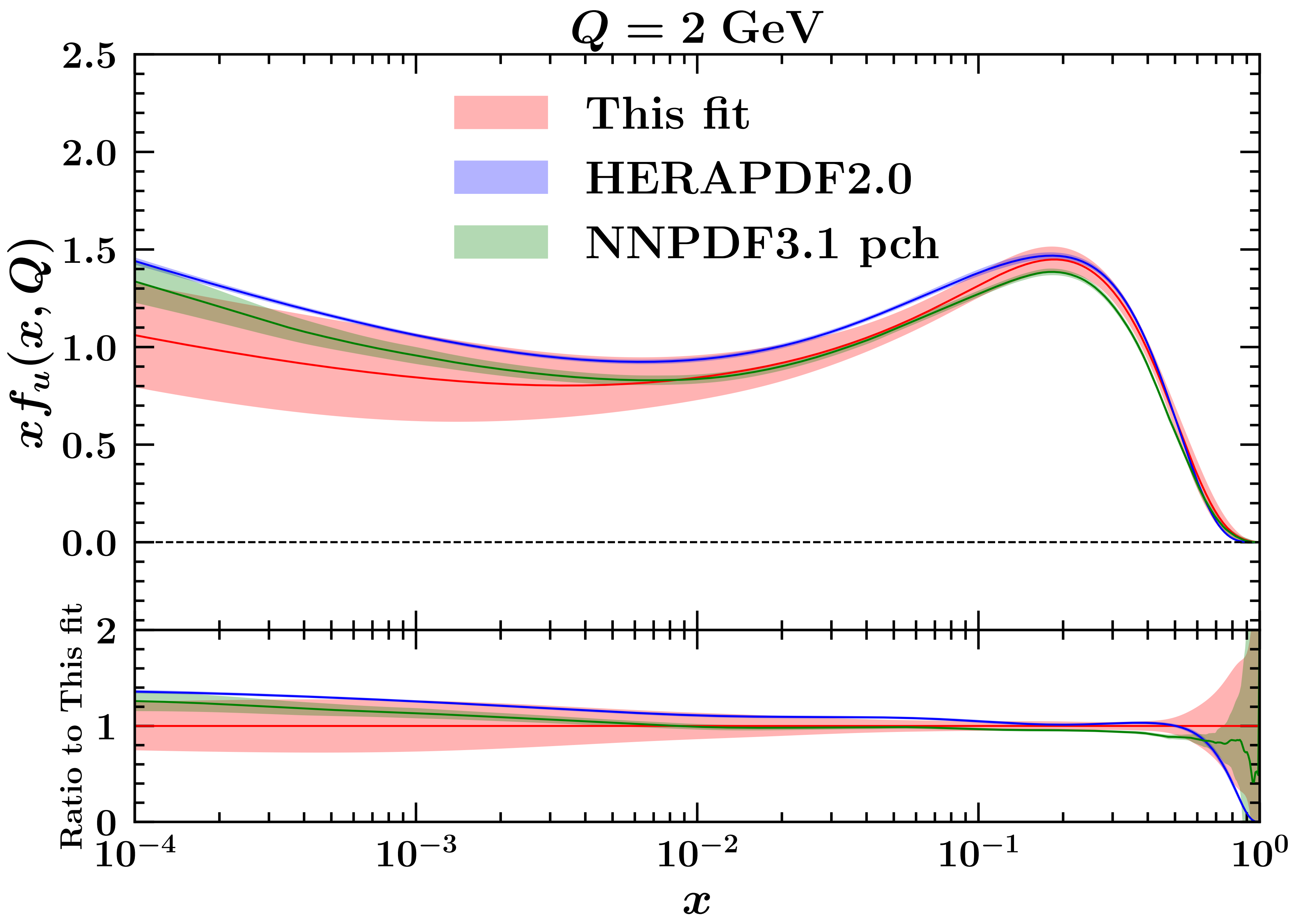
stable results even when varying NN structure

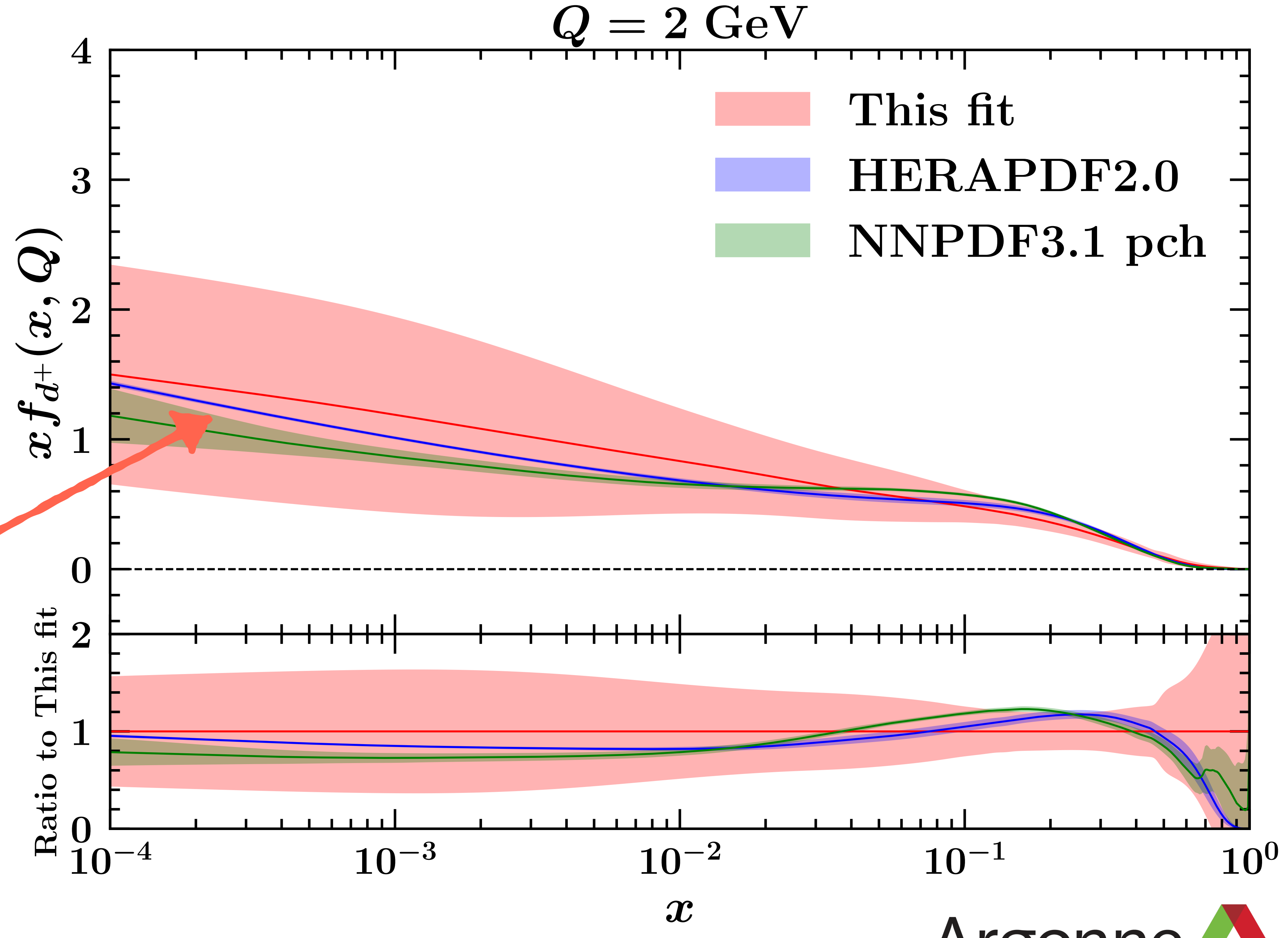


overall,
it's not bad

BUT

large uncertainties and
central values not so close to other extractions

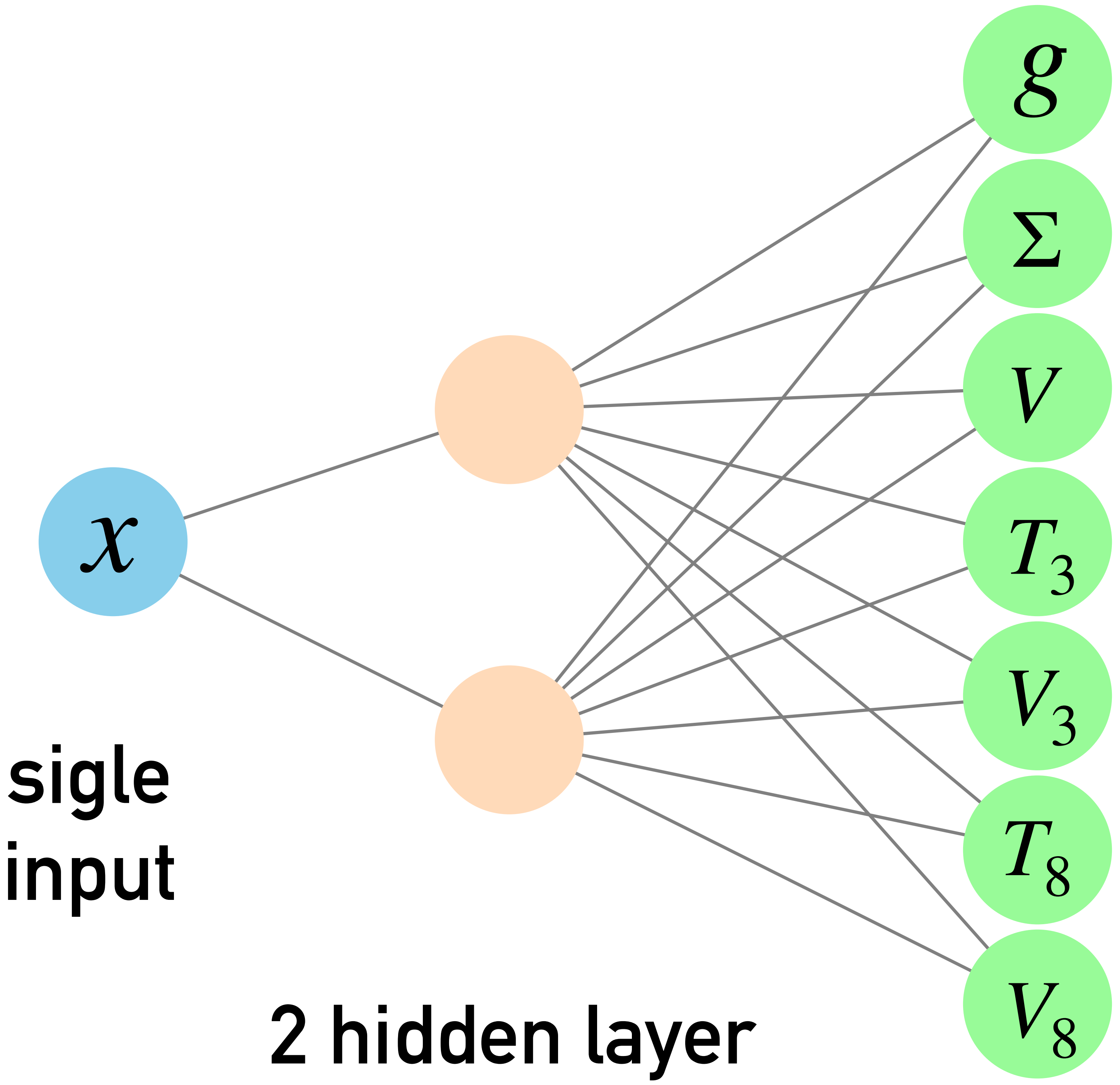




Neural network structure

for PDF extraction

parameterization with preprocessing function



$$g$$
$$\Sigma = \sum_q q^+$$
$$V = \sum_q q^-$$
$$T_3 = u^+ - d^+$$
$$V_3 = u^- - d^-$$
$$T_8 = u^+ + d^+ - 2s^+$$
$$V_8 = u^- + d^- - 2s^-$$

sigle input

2 hidden layer nodes

7 output nodes

❋ we changed the NN structure

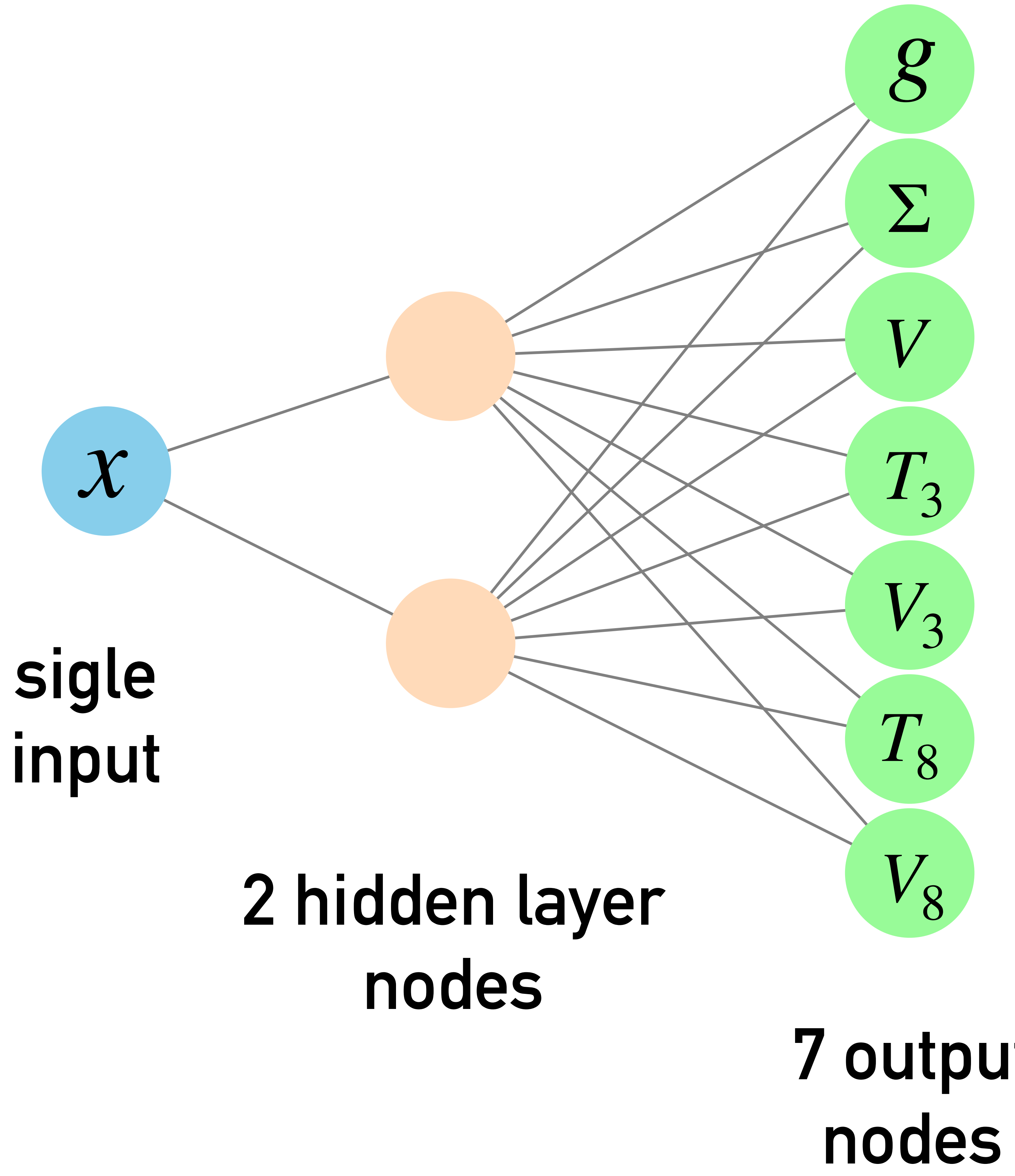
- one more output node
- reduce the number of parameters

❋ we changed the parameterization of the output nodes

Neural network structure

for PDF extraction

parameterization with preprocessing function



as preprocessing functions we choose the

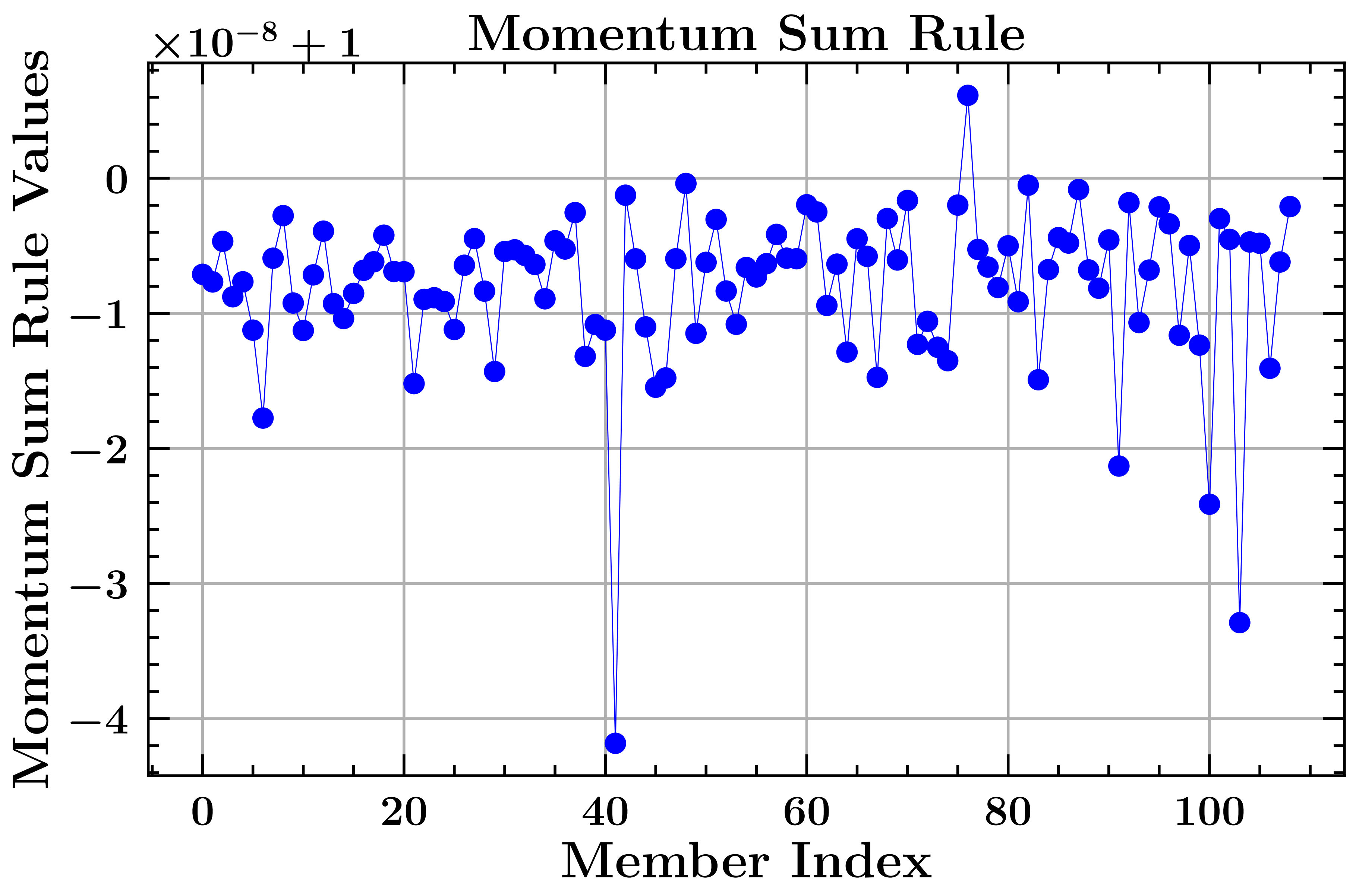
HERA parameterization in the form $\{g, u, \bar{u}, d, \bar{d}, s = \bar{s}\}$

$$
\begin{aligned}
xg(x) &= A_g x^{B_g}(1-x)^{C_g} - \cancel{A_{g_p} x^{B_{g_p}}(1-x)^{C_{g_p}}}\,,\\
xu(x) &= A_{U_b} x^{B_{U_b}}(1-x)^{C_{U_b}}(D_{U_b}x+1) + A_{u_v} x^{B_{u_v}}(1-x)^{C_{u_v}}(E_{u_v}x^2+1),\\
xd(x) &= A_{D_b}(1-f_s)x^{B_{D_b}}(1-x)^{C_{D_b}} + A_{d_v} x^{B_{d_v}}(1-x)^{C_{d_v}}\,,\\
x\bar{u}(x) &= A_{U_b} x^{B_{U_b}}(1-x)^{C_{U_b}}(D_{U_b}x+1),\\
x\bar{d}(x) &= A_{D_b}(1-f_s)x^{B_{D_b}}(1-x)^{C_{D_b}}\,,\\
xs(x) &= A_{D_b} f_s x^{B_{D_b}}(1-x)^{C_{D_b}} = x\bar{s}(x)\,.
\end{aligned}
$$

14 free params + NN params

Sum rules

as integral constraints on extracted PDFs



- momentum sum rule $\Sigma = \sum_q q^+ \equiv 1$

$$\longrightarrow \int_0^1 x \left[f_q(x, Q^2) + f_{\bar{q}}(x, Q^2) + f_g(x, Q^2) \right] \, dx = 1$$

- total valence sum rule $V = \sum_q q^- \equiv 3$

$$\longrightarrow \int_0^1 \left[f_u(x) - f_{\bar{u}}(x) + f_d(x) - f_{\bar{d}}(x) \right] dx = 3$$

- … and analogously for other combinations:

$$T_3 = u^+ - d^+ \qquad V_3 = u^- - d^- \equiv 1$$

$$T_8 = u^+ + d^+ - 2s^+ \qquad V_8 = u^- + d^- - 2s^- \equiv 3$$

notation

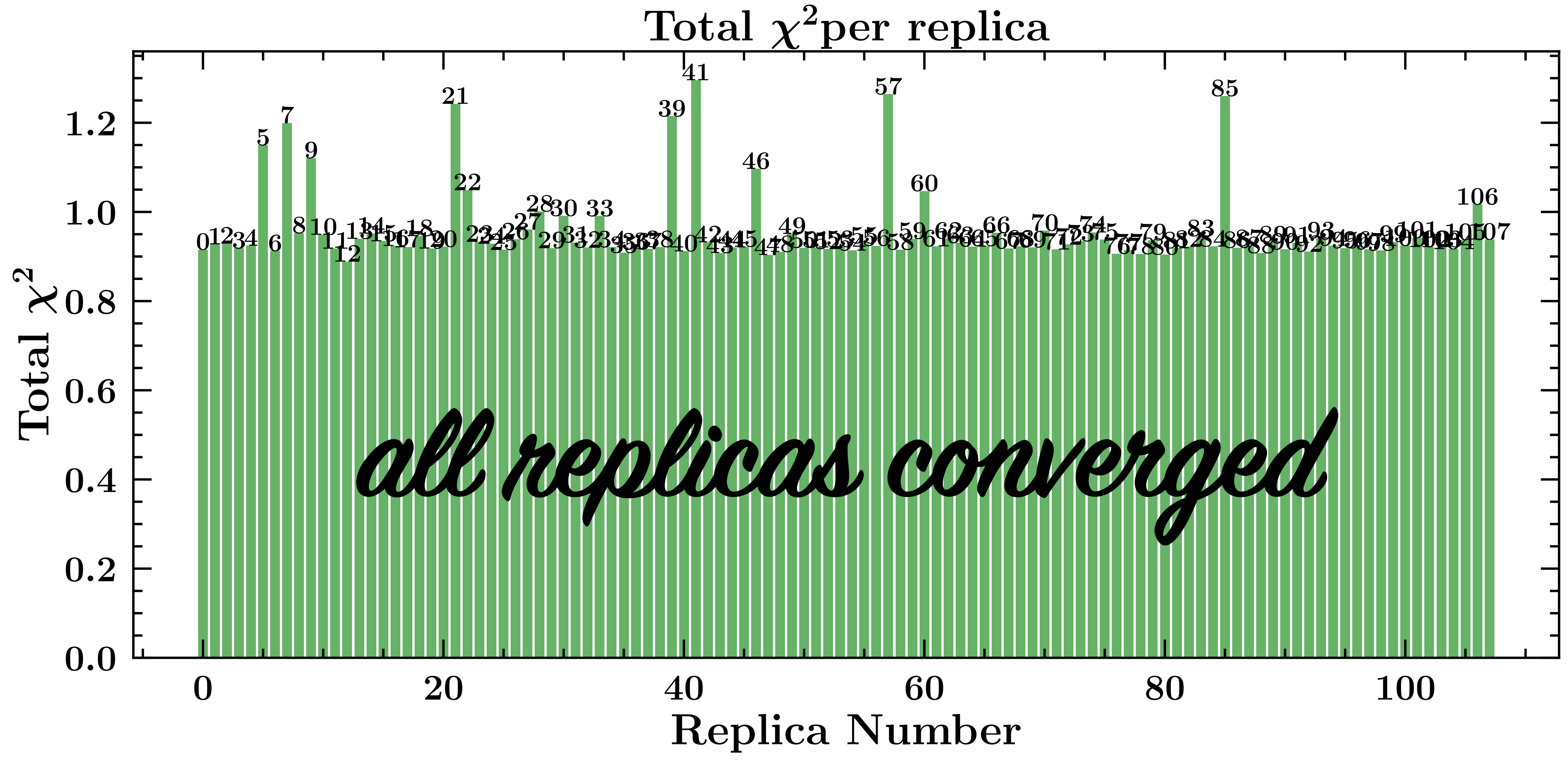
$$q^\pm \equiv q \pm \bar{q}$$

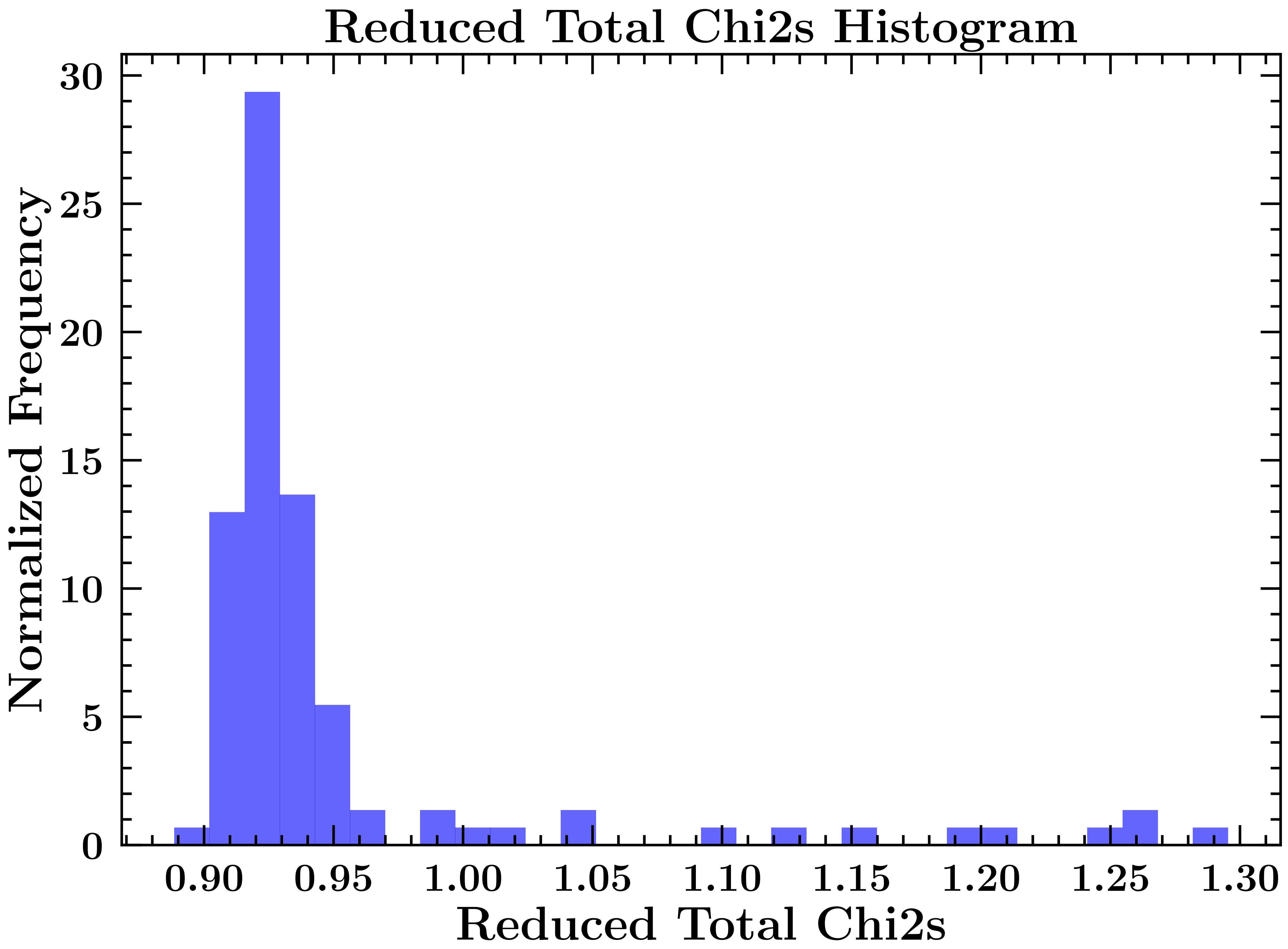
Results

NN with preprocessing functions

in the spirt of 'proof of concept'

107 replicas





d.o.f = 1016

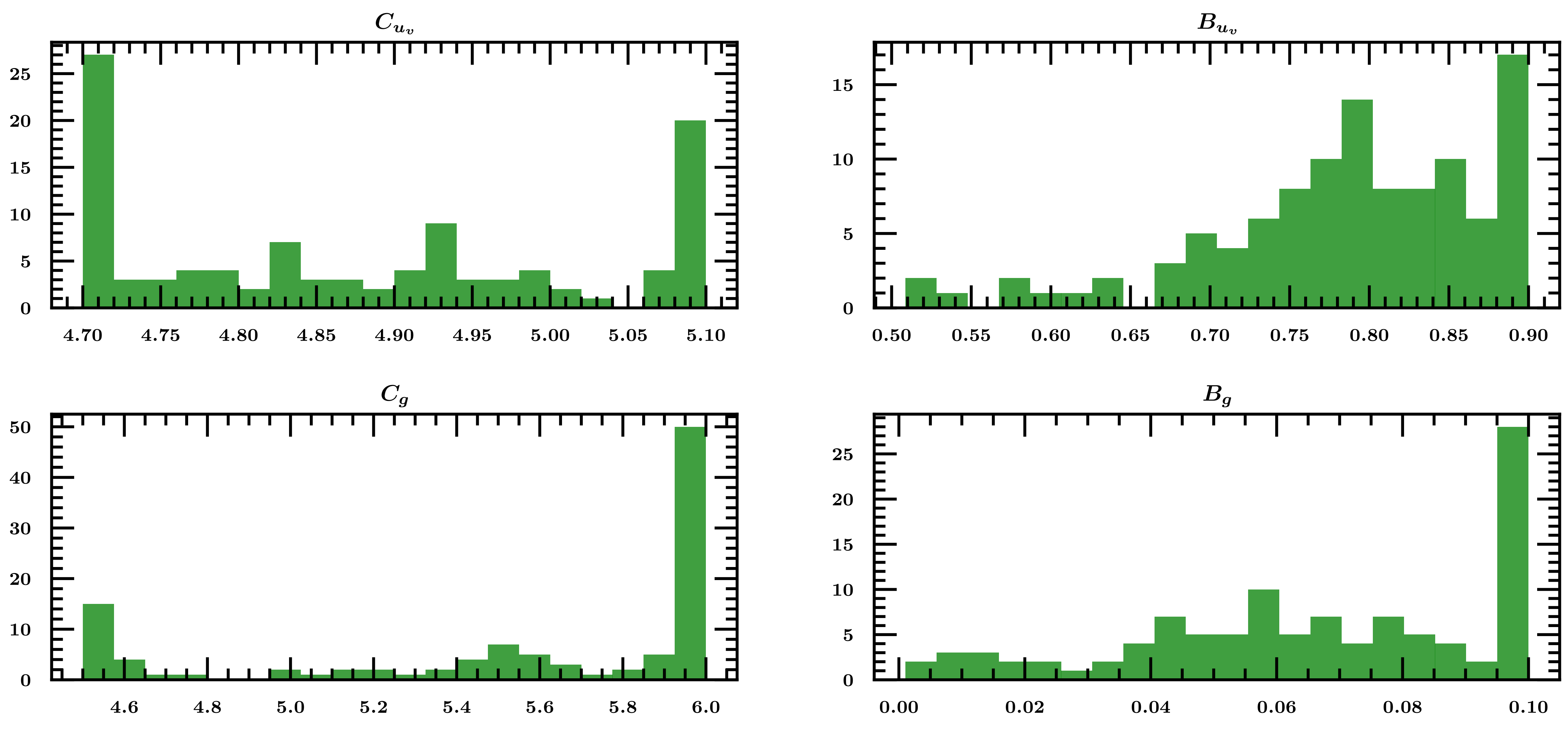
= # of points: 1016

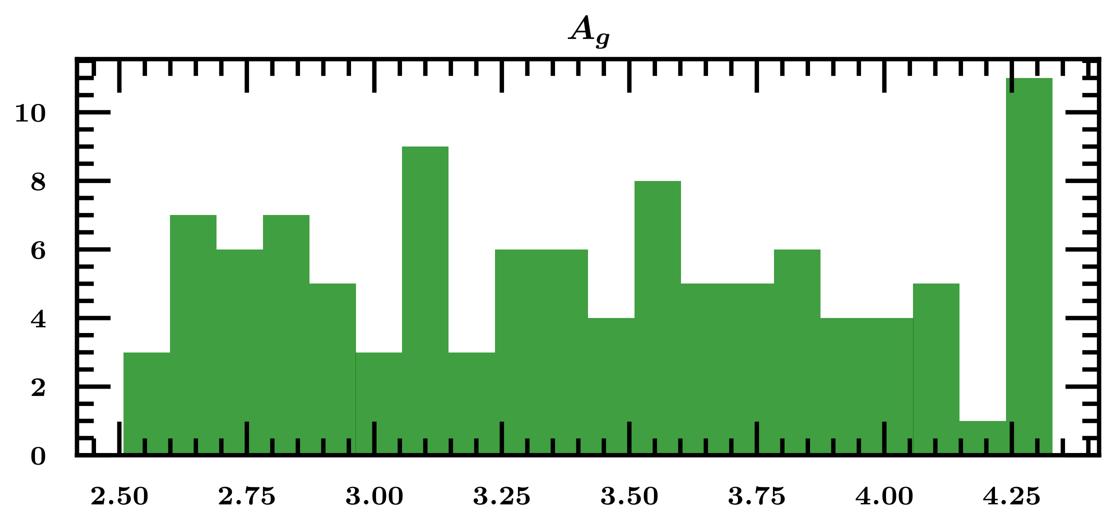
total χ^2 = 0.9

Results

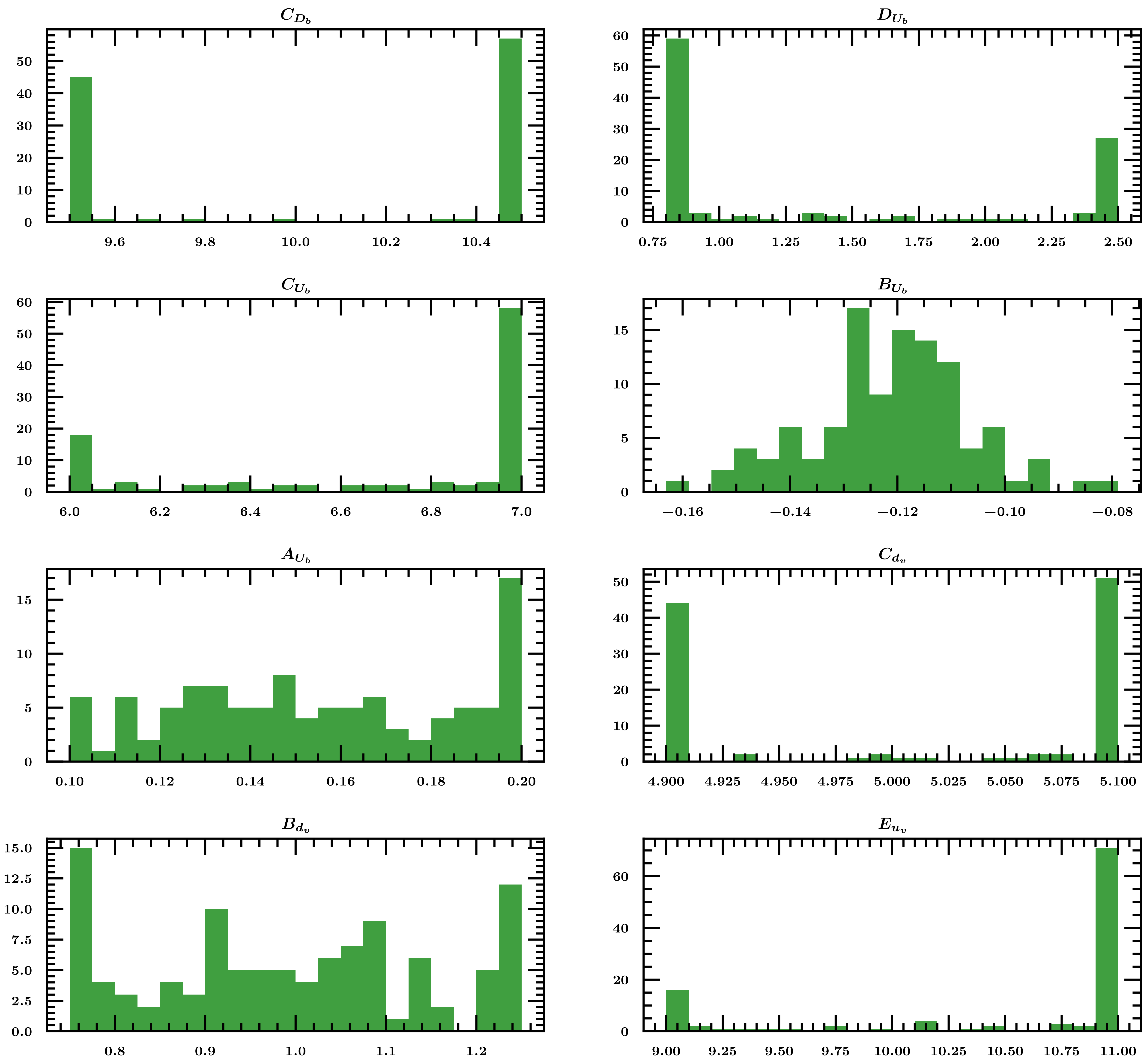
NN with preprocessing functions

* preprocessing function parameters histograms





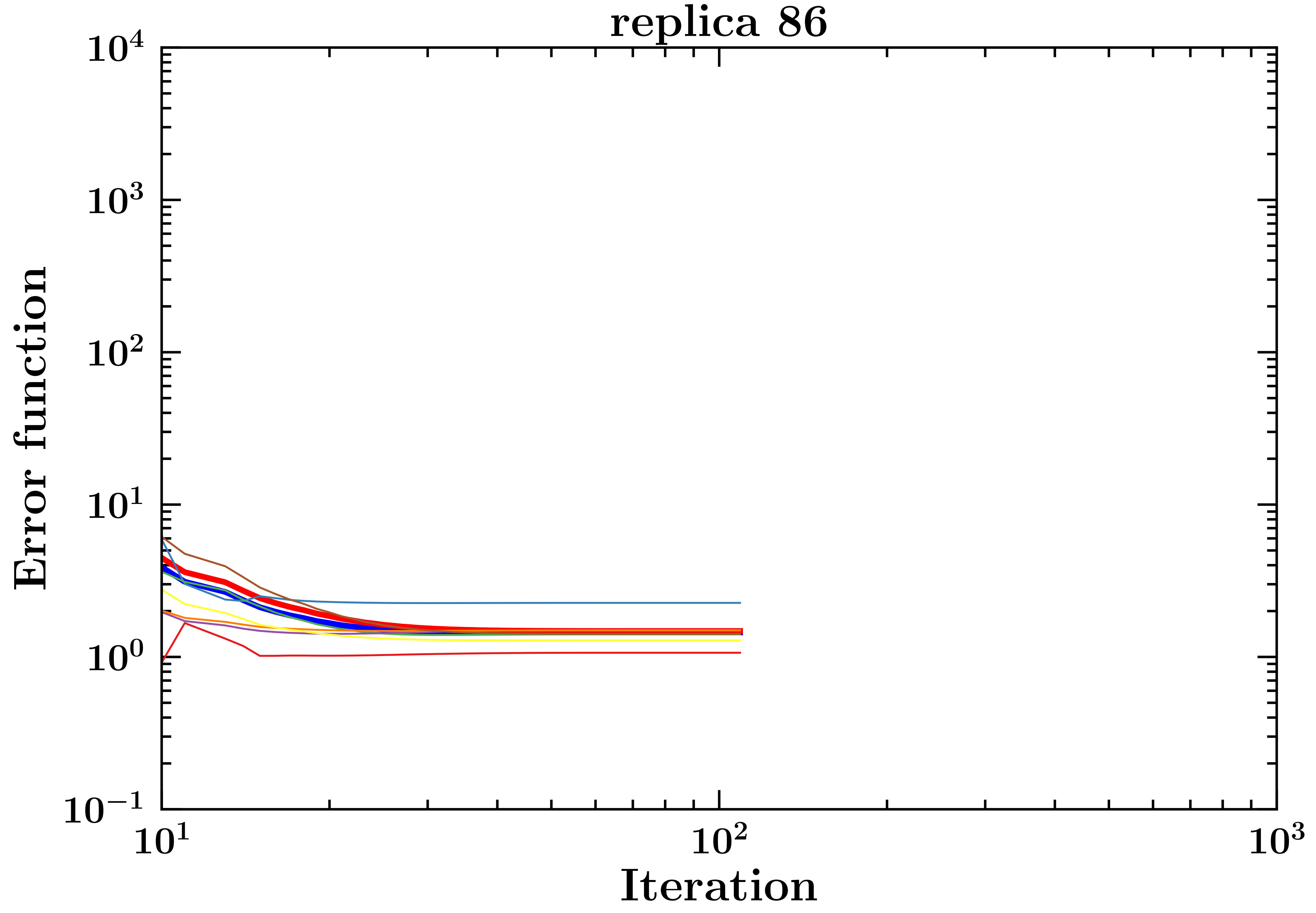
... still work in progress ...

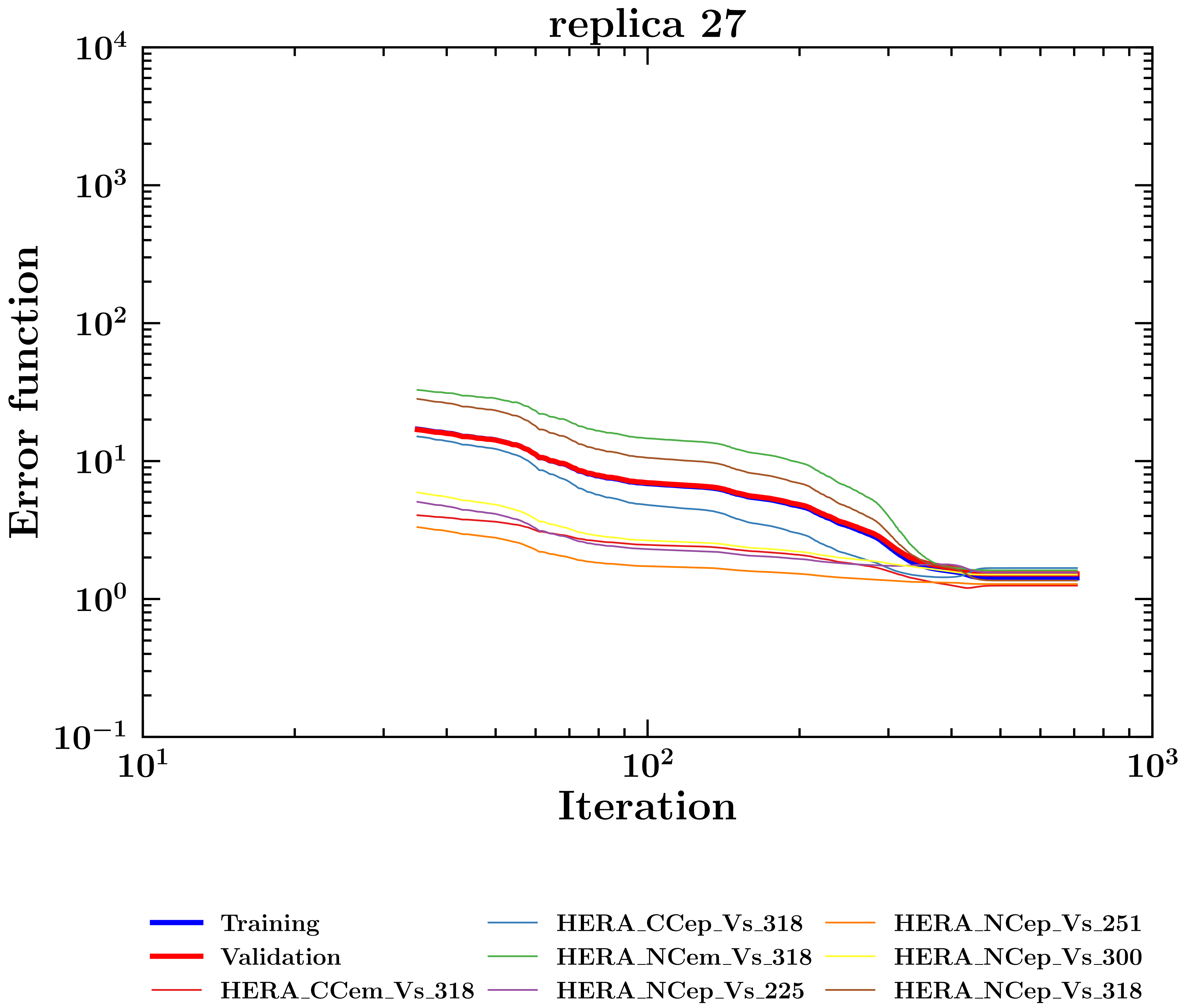


Training and validation

- datasets split in
  50% training, 50% validation

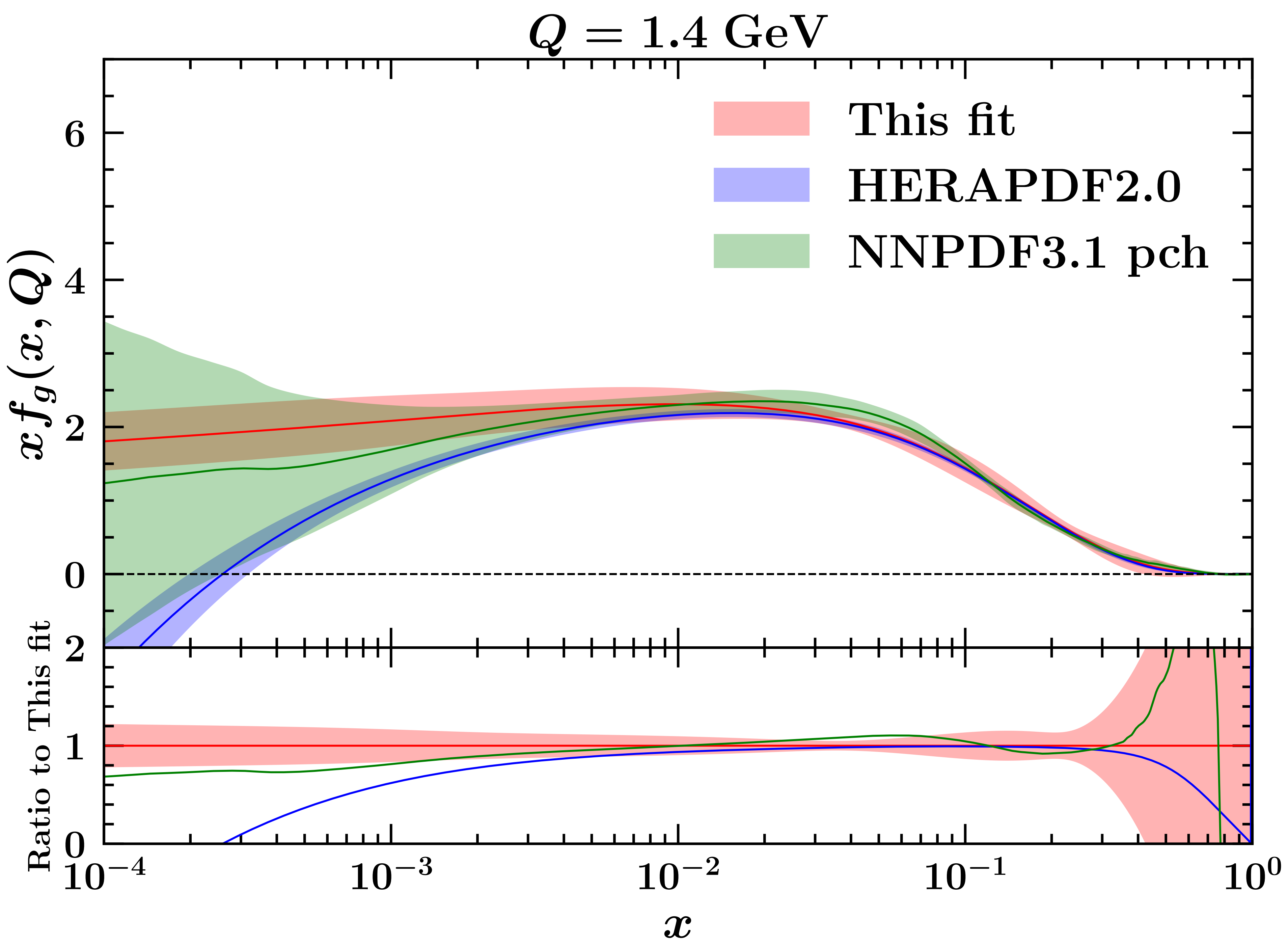
same choice as MontBlanc
FF extraction with NN

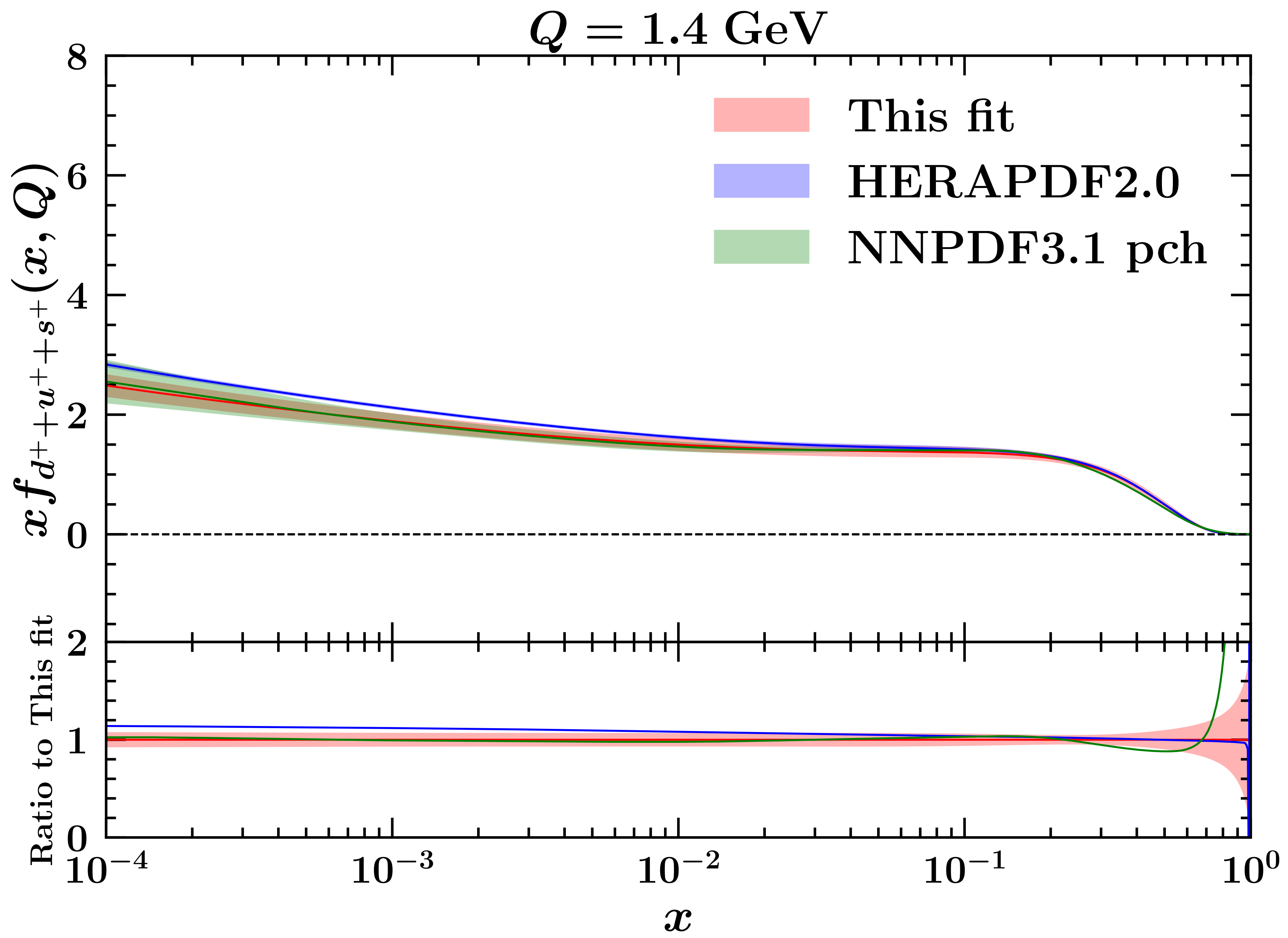


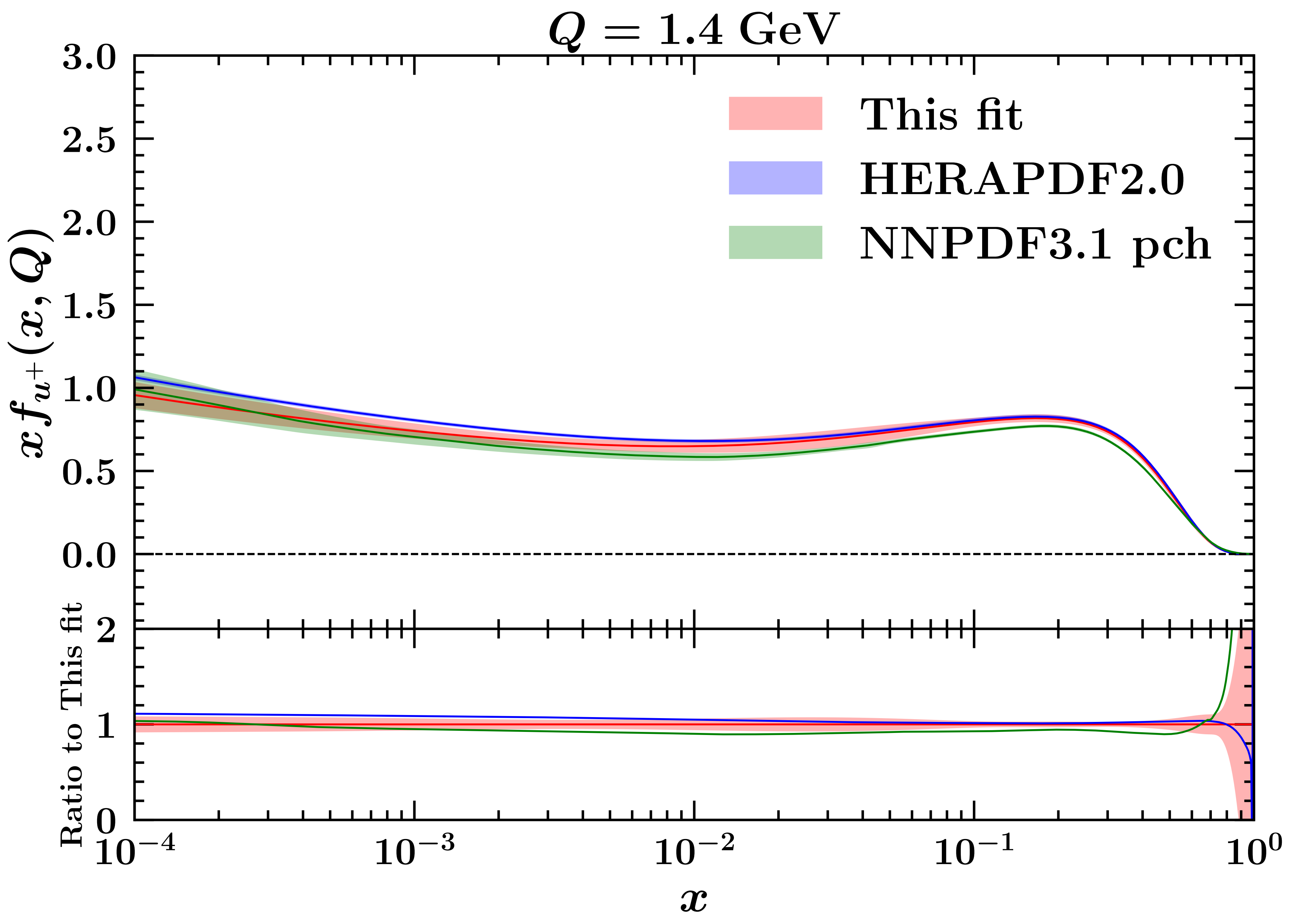


PDF bands

68% c.l.

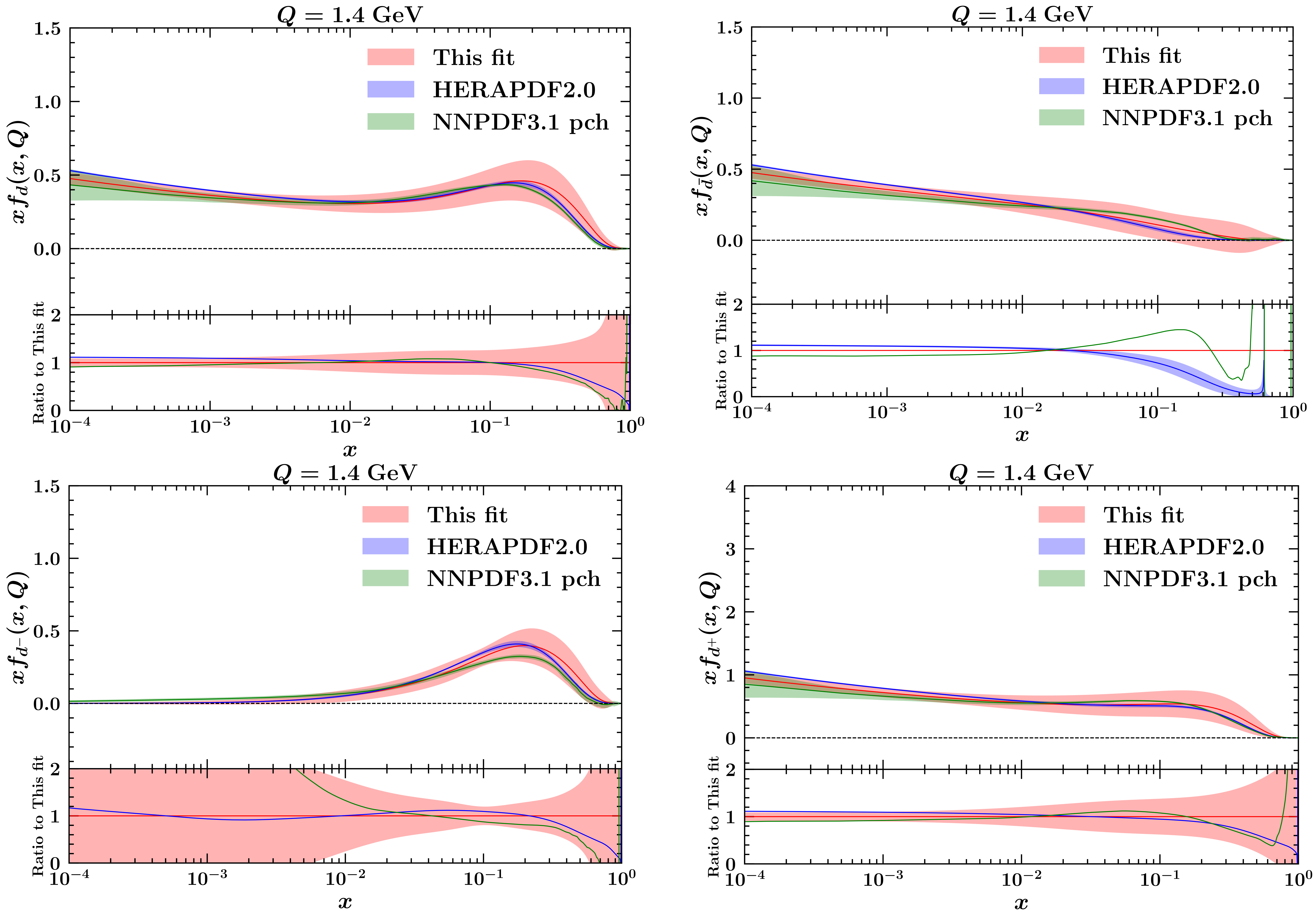






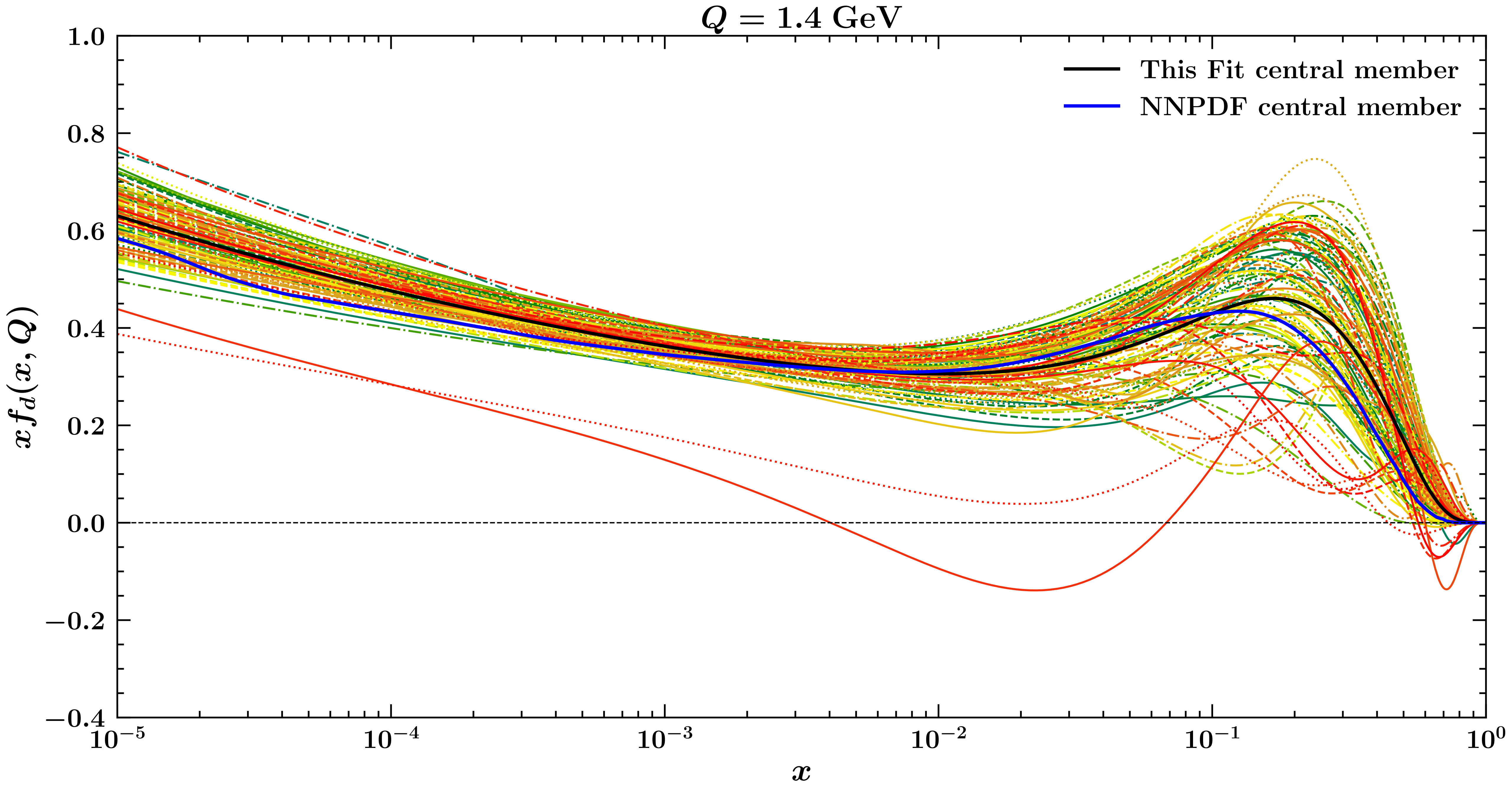
PDF bands

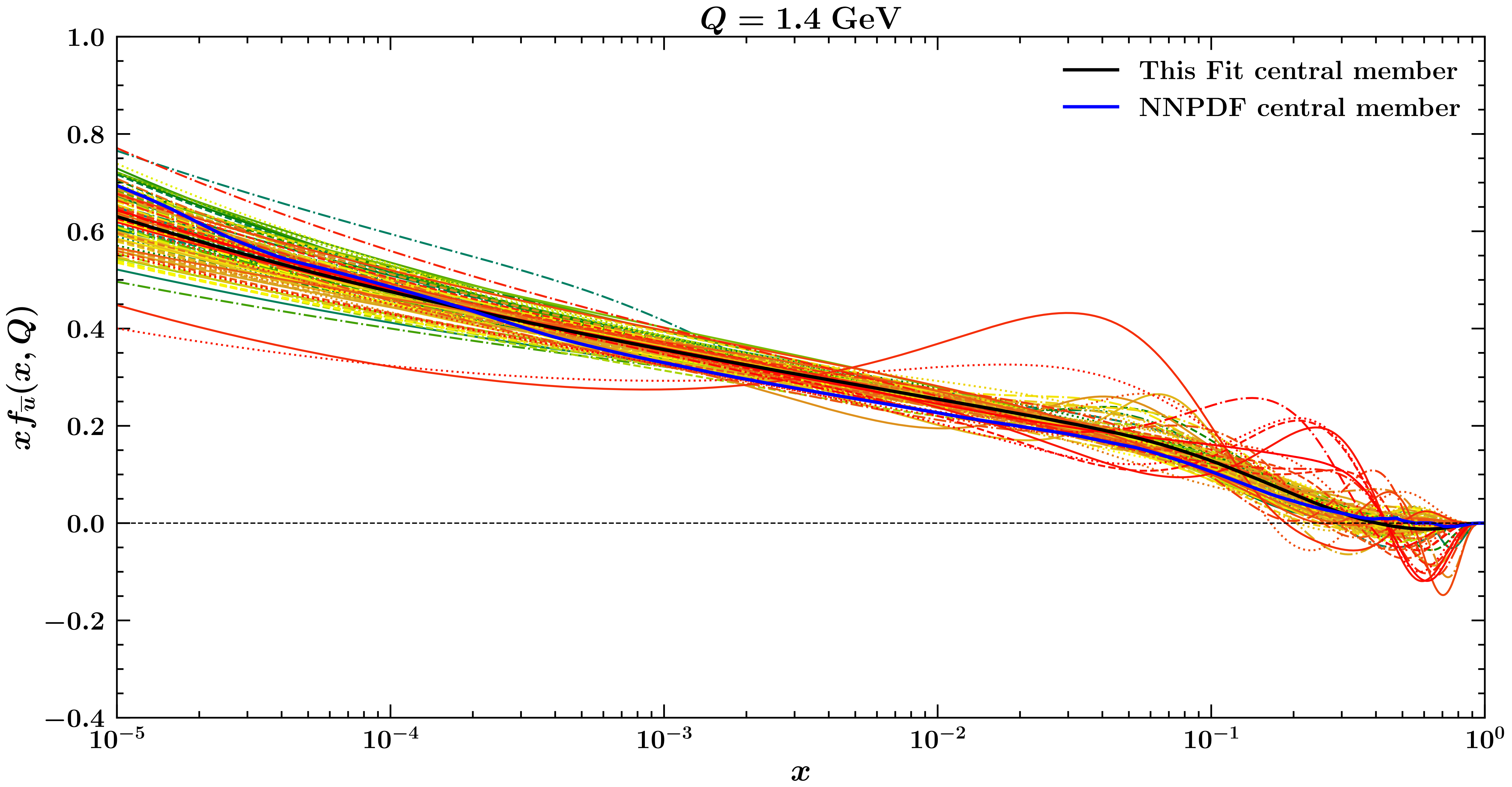
68% c.l.



All replicas

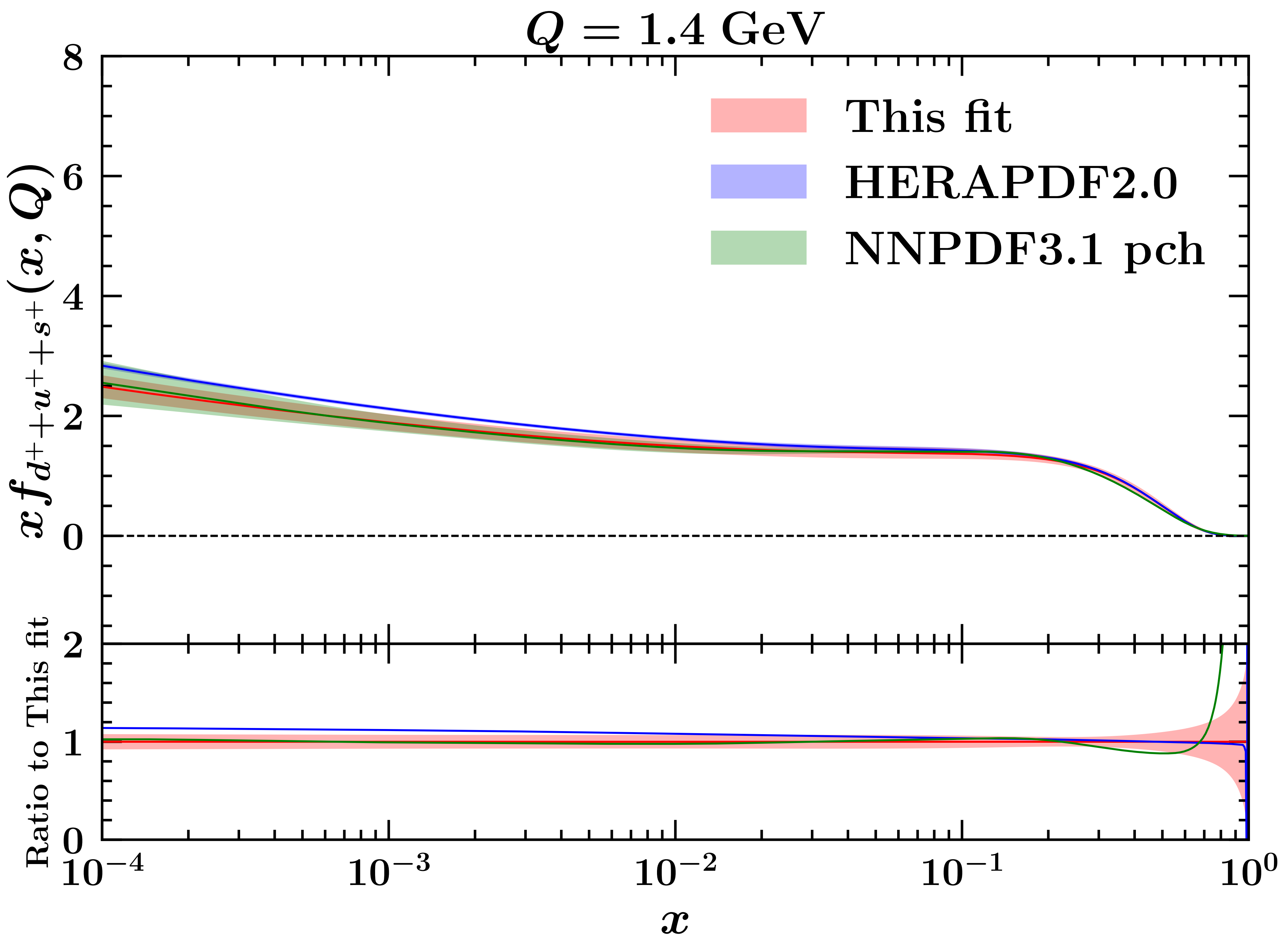
colors rank from
worst chi2 (red) to best chi2 (green)





Conclusions and outlook

(… it's still work in progress)



- ❋ we managed to get a PDF extraction compatible with HERAPDF2.0

- ❋ encouraging results with preprocessing function + NN

  … we are dealing with the challenges, e.g. positivity

- ❋ the **NavyPier – NNAD** framework is extendible to

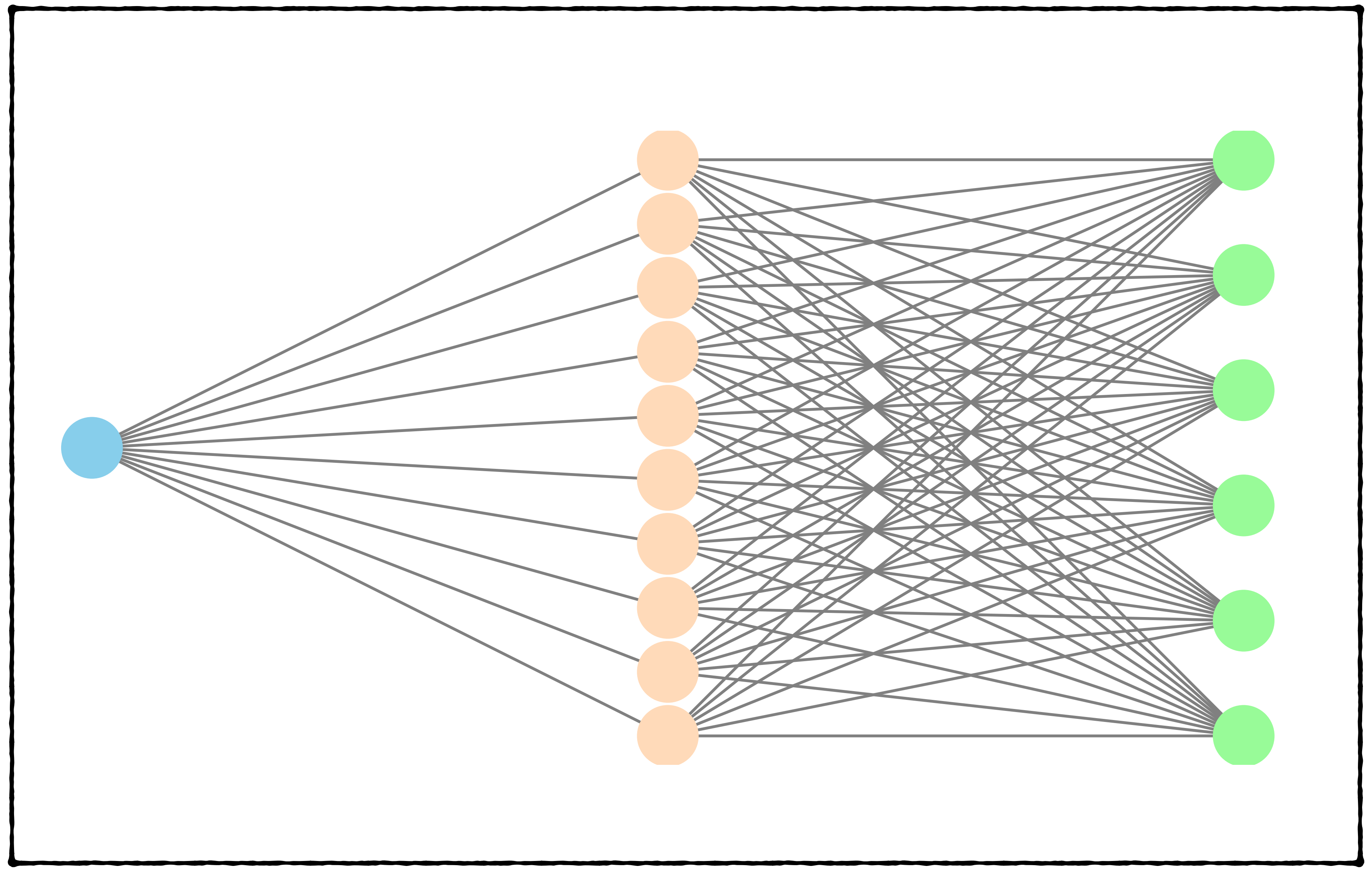
  fit TMDs with **NangaParbat**

  Beyond the Standard Model studies

Next step:

add Beyond the Standard Model to the framework

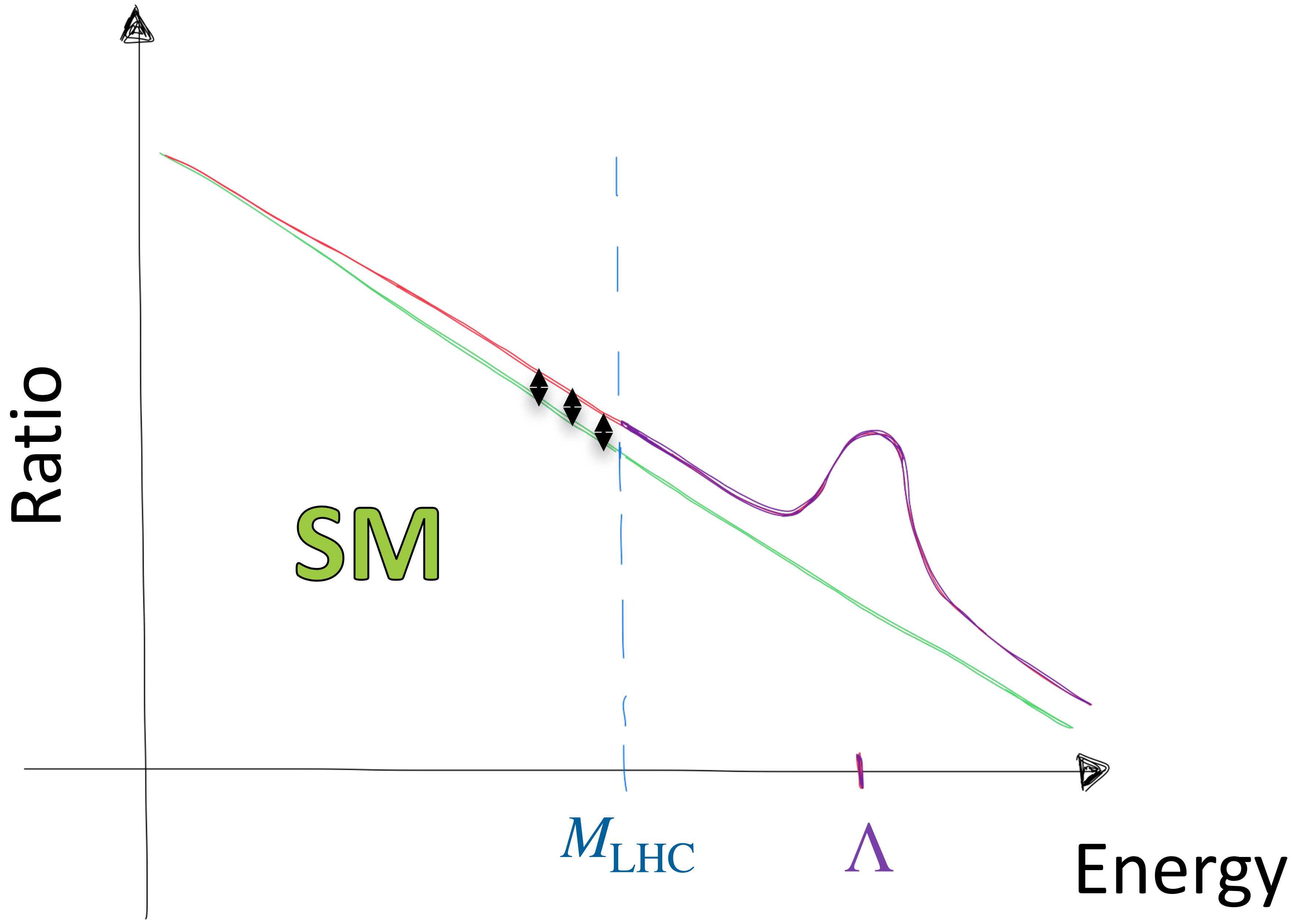
PDFs



+

SMEFT

Standard Model Effective Field Theory



Backup

BSM framework

Standard Model Effective Field Theory

$$\mathscr{L}_{\text{SMEFT}} = \mathscr{L}_{\text{SM}} + \sum_{n>4} \frac{1}{\Lambda^{n-4}} \sum_k \mathcal{C}_k^{(n)} O_k^{(n)}$$

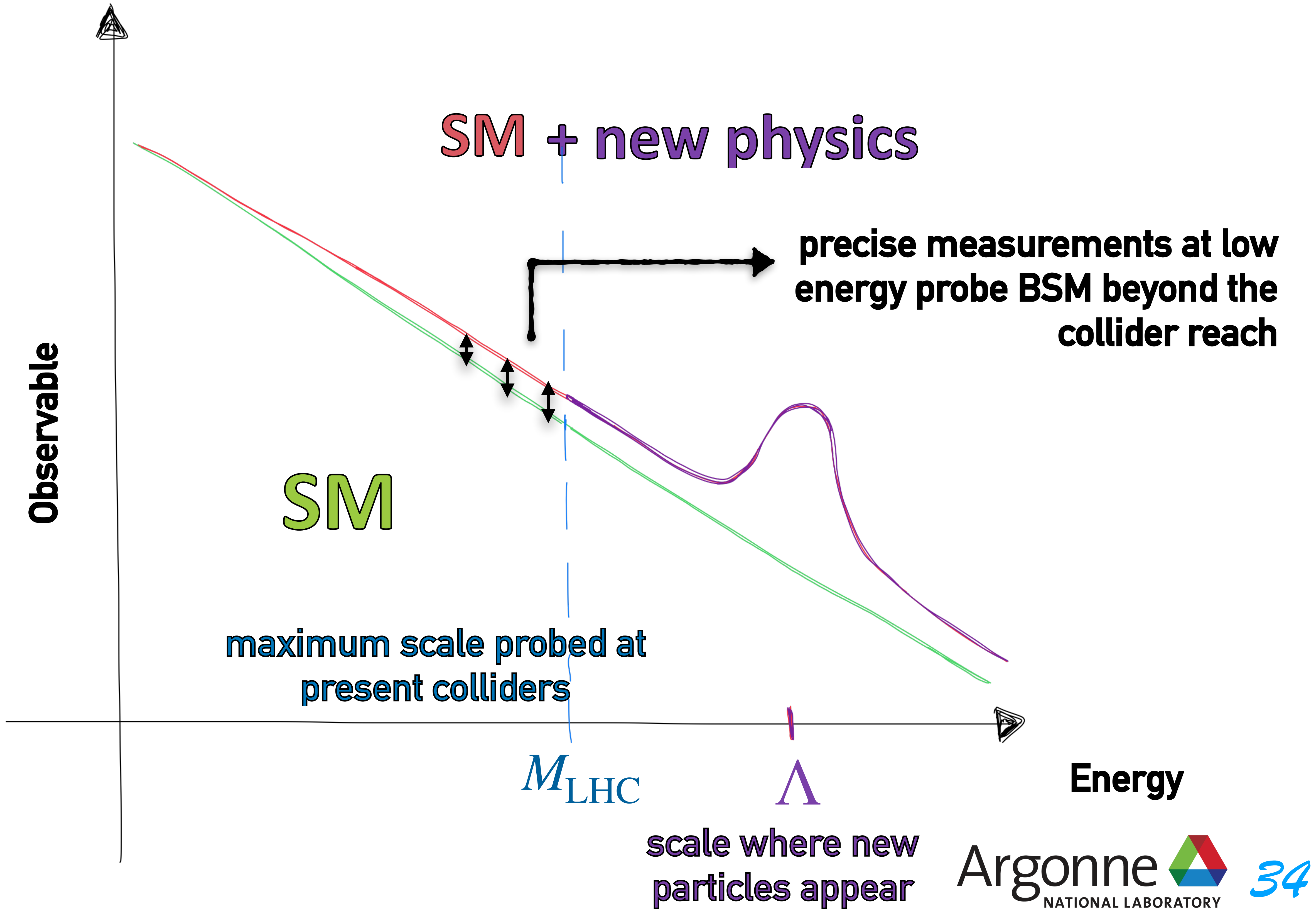
operators
interactions terms

Wilson coefficients

- The EFT constructed with SM fields and symmetries
- Model independent
- Connect data from different processes
- Number of parameters known for all orders

$$\mathscr{L}_{\text{SMEFT}} = \mathscr{L}_{\text{SM}} + \frac{1}{\Lambda}\mathscr{L}_5 + \frac{1}{\Lambda^2}\mathscr{L}_6 + \frac{1}{\Lambda^3}\mathscr{L}_7 + \frac{1}{\Lambda^4}\mathscr{L}_8 + \ldots$$

2499 **36971**



EFT pathway to new physics

precise SM predictions

$$\Delta Obs_n = Obs_n^{EXP} - Obs_n^{SM} = \frac{1}{\Lambda^2}\sum_k \mathcal{C}_k^{(6)}(\mu)a_{n,k}^{(6)}(\mu) + \mathcal{O}\left(\frac{1}{\Lambda^4}\right)$$

precise experimental measurements

precise EFT predictions

huge efforts to improve each one of these steps