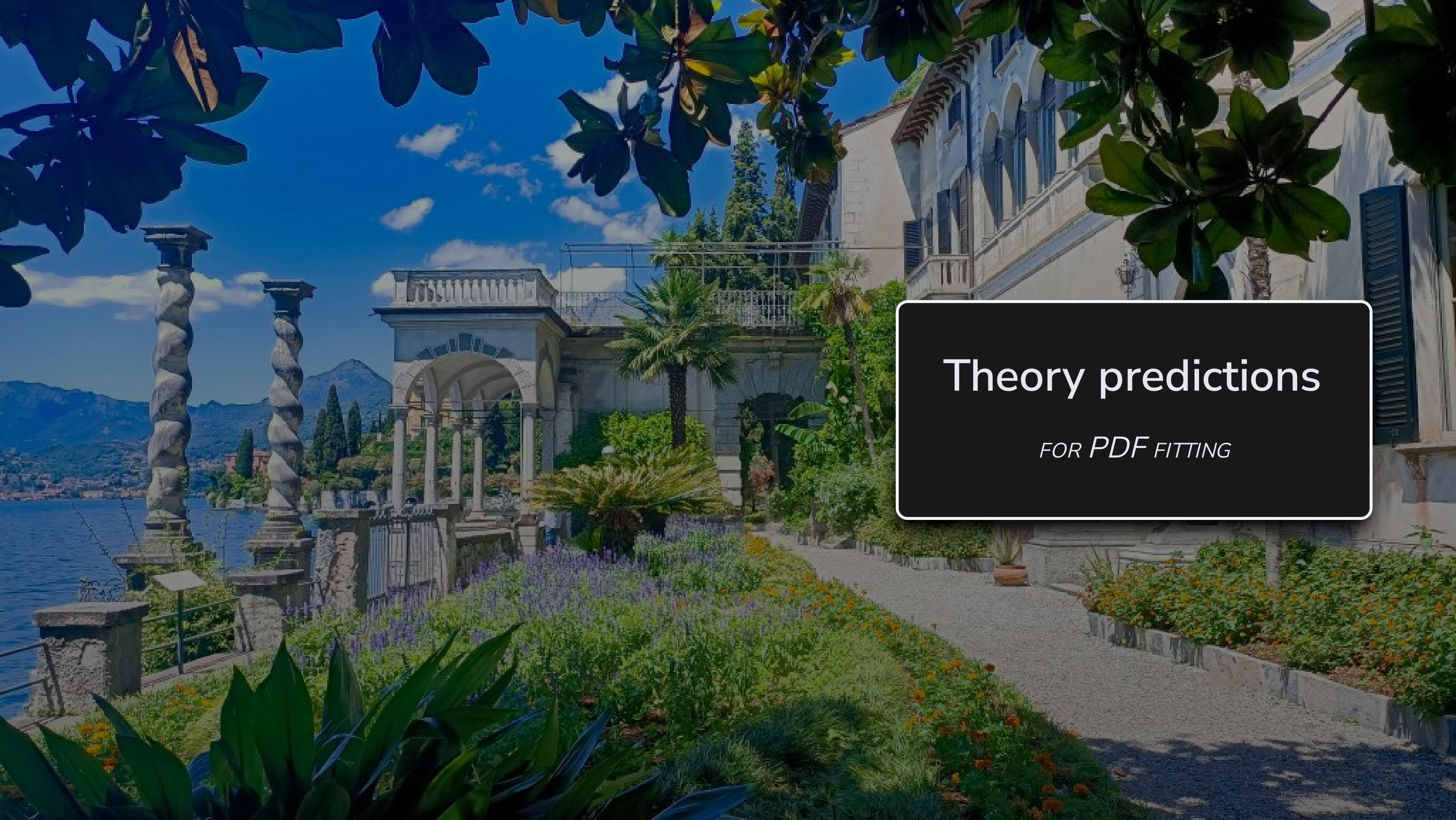


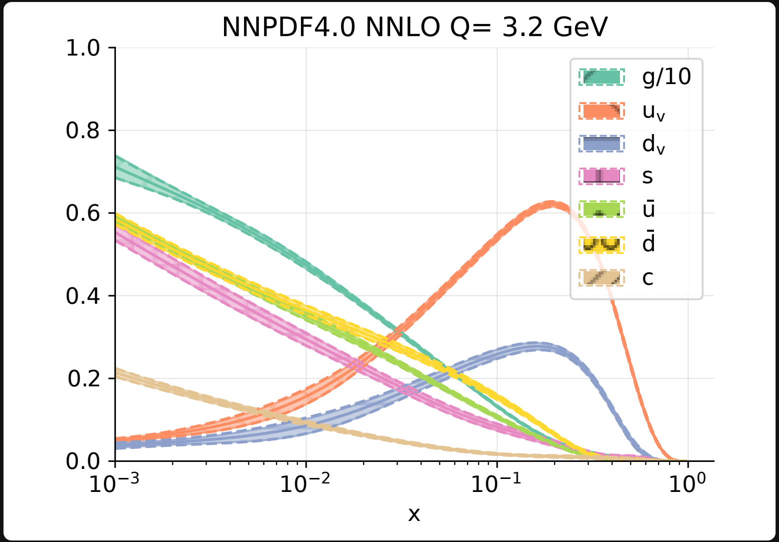
Theory predictions

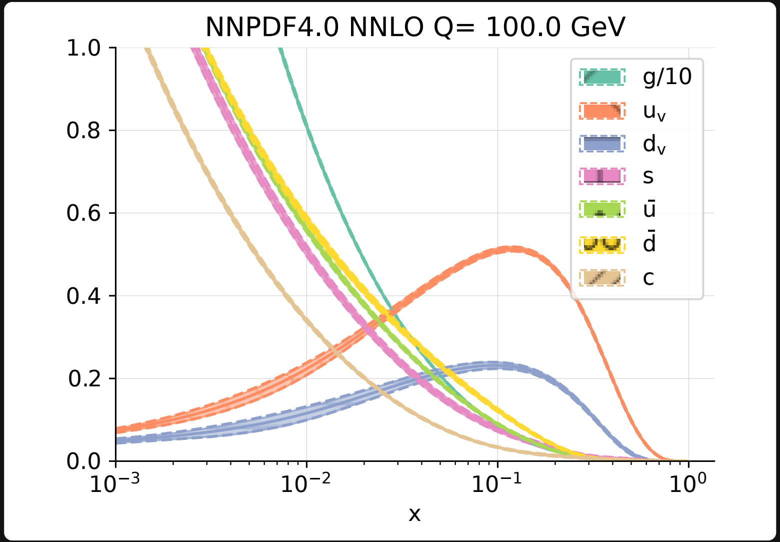
FOR PDF FITTING

Contents

NNPDF 4.0

[arXiv: 2109.02653]

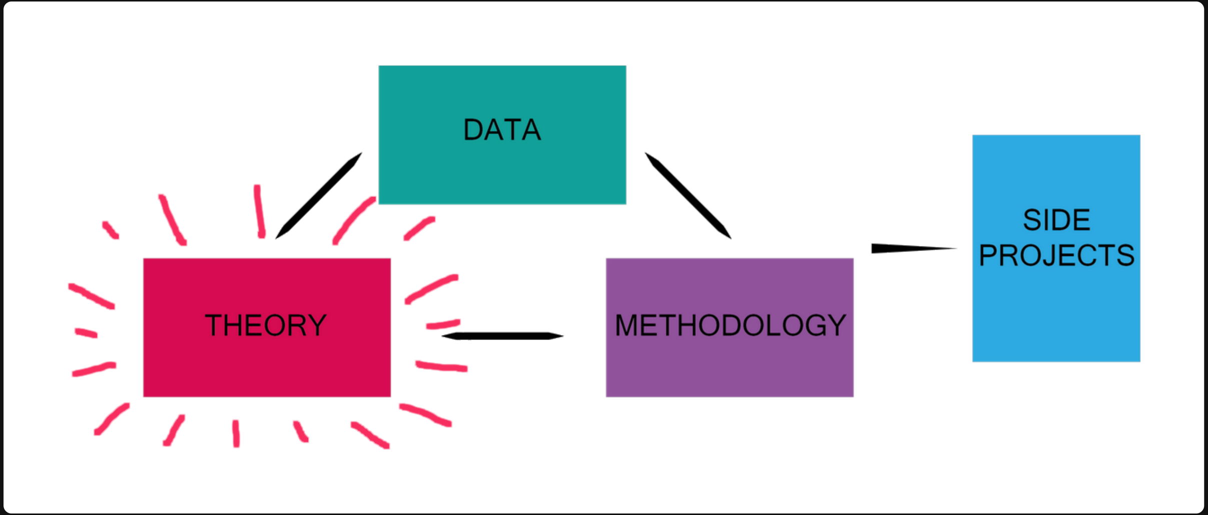




To get to a full PDF set many ingredients are required...

The two most relevant novelties of NNPDF 4.0 have been:

- the Machine Learning improved methodology
- the great amount of new LHC data sets.



My main work has not yet been used for a main fit, and it concerns the remaining part: *theory predictions*.

Theory predictions

Factorization

The QCD property allowing PDFs definition

$$\sigma_{dis} = \hat{\sigma} \otimes f + \textit{non-leading powers}$$
$$\sigma_h = \hat{\sigma} \otimes f \otimes f + \ldots \equiv \hat{\sigma} \otimes \mathcal{L} + \ldots$$

where:

- σ is the prediction for the physical observable, to be compared with the experimental measurement
- f is the PDF
- $\hat{\sigma}$ is the *coefficient function*

$\otimes$ is a convolution

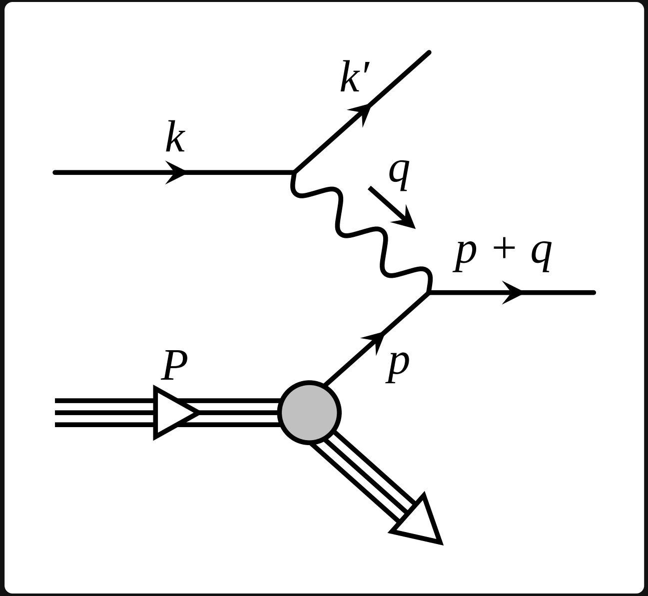
$$a \otimes b(x) \equiv \int_x^1 \mathrm{d}z \, a(z) b(x/z)$$

and a sum over flavors is implicit

Factorization ensures PDFs universality: the PDF determined from some processes is the same for any other involving the same initial-state hadron.

Deep Inelastic Scattering

Lepton-hadron scattering



The ideal process to probe PDFs: a point-like particle scanning the composite one.

DIS DIMENSIONS

Many options to be supported

Observables:

The physical object to be computed

Process
- EM
- NC
- CC

×

Kind
- σ
- F_2
- F_L
- F_3

×

Heavyness
- Light
- Charm
- Bottom
- Top

×

Flavor Number Scheme
- FFNS
- ZM-VFNS
- GM-VFNS

Coefficients:

The encoded expression

Perturbative Order
- LO
- NLO
- NNLO
- N^3LO

×

Process
- EM
- NC
- CC

×

Kind
- F_2
- F_L
- F_3

×

Heavyness
- Light
- Heavy
- Asymptotic
- Intrinsic

×

Channel
- Non-singlet*
- Gluon
- Singlet

Coefficients Availability

NLO	light	heavy	asymptotic	intrinsic
NC	✓	✓	✓	✓
CC	✓	✓	✓	✓
NNLO				
NC	✓	⊞*	✓	✗
CC	✓	⊞**	✓	✗
N^{3}LO				
NC	✓	✗†	✗‡	✗
CC	✓	✗†	✗‡	✗

Yadism

Yet another DIS module

There were already mainly two available DIS providers: *APFEL* and *QCDNUM*.

But we needed:

- an improved/rechecked scale variations implementation
- new analytical components
  - intrinsic NLO CC
  - improved heavy NNLO NC
  - heavy NNLO CC *(yet to come)*
- re-examine FNSs
- more TMC options
- extended and automated benchmarks
- integration in the pineline
  - produce PineAPPL grids
  - registered PineFarm provider
- curated docs
- improved maintainability (more modular, organic design, CI/CD, packaging and distribution)

Yadism is also a coefficients database: they are implemented in such a way to be directly used by 3rd parties!

FONLL STATUS

[arXiv: 1001.2312]

$$F_{FONLL}(Q^2) = F^{(n)}(m_q) - F^{(n,0)}(m_q) + F^{(n+1)}$$

This GM-VFNS is matching two calculations:

- one done in n flavors scheme, accounting for mass effects
- another in $n+1$ scheme, resumming collinear logs

To use PDFs in a single scheme, $f^{(n+1)}$, matching conditions (and an entire EKO) are encoded in the coeffs, $B^{(n)}$.

For consistency, Yadism at the moment implements FONLL with one mass at a time:

3 ⇢ 3c + 3b $\xrightarrow{\mu c}$ 4 ⇢ FONLL-c + 4b $\xrightarrow{\mu b}$ 5 ⇢ FONLL-b $\xrightarrow{\mu t}$ 6 ⇢ ZM6

Lifting the limitation of using PDFs in a single scheme, FONLL can be directly implemented at the observable level, including consistently many different mass effects differences: $F^{(d,n)} = F^{(n)} - F^{(n,0)}$

$$F_{FONLL}(m_c, m_b) = F^{(d,3)}(m_c) + F^{(d,4)}(m_b) + F^{(5)}$$

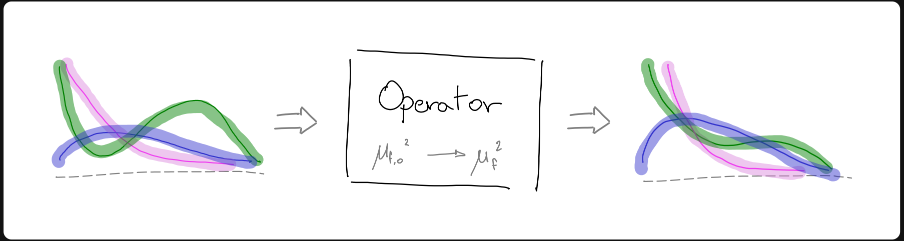
DGLAP EVOLUTION

Define PDFs dependency on unphysical scale μ_F

$$\mu^2 \frac{\mathrm{d}\mathbf{f}}{\mathrm{d}\mu^2}(\mu^2) = \mathbf{P}(a_s(\mu^2), \mu^2) \otimes \mathbf{f}(\mu^2)$$

These equations define a set of linear operators $\mathbf{E}(\mu^2 \leftarrow \mu_0^2)$ on PDF sets:

$$\mathbf{f}(\mu^2) = \mathbf{E}(\mu^2 \leftarrow \mu_0^2) \otimes \mathbf{f}(\mu_0^2)$$



EKO

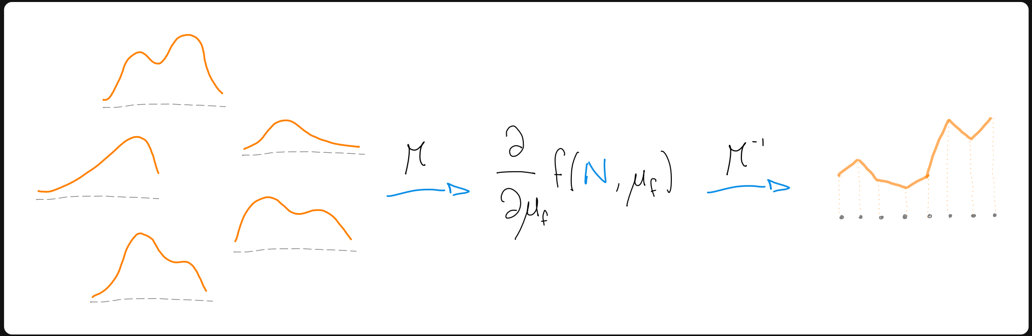
Evolutionary Kernel Operator

[arXiv: 2202.02338]

The main goal of EKO is to compute **reusable evolution operators** (i.e. EKOs).

runcard 📄 —run 🦓→ EKO 📦

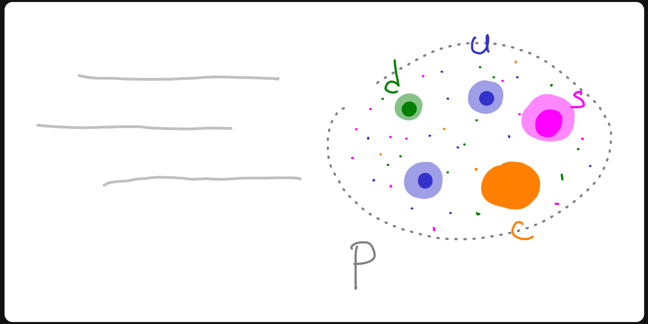
It does so, by solving DGLAP equations in N-space, but providing an x-space output for compatibility with existing PDF sets.



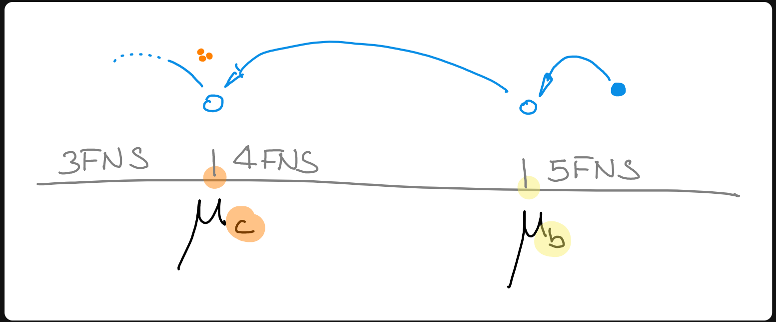
ORIGINAL FEATURES

Pineline integration

- Intrinsic evolution



- Full backward VFNS (including intrinsic)



… AND MORE TO COME

More than anything else, EKO is an evolution framework, still rapidly growing collecting many contributions, from new and existing applications.

The goal is to have all of them in a single place, for anyone interested in solving DGLAP.

- Factorization scale vars
  - first truly expanded
  - full *resummation* scales `[arXiv: 2205.15900]`
- QED implementation
  - up to NLO QCD-QED and NNLO QED
  - fixed (non-running) QED coupling
- N^3LO evolution and matching
  - *in-house* splitting functions
- Polarized evolution
- Time-like evolution

- storage
- distributed computation
- N space expressions database
- curated docs
- improved maintainability (more modular, organic design, CI/CD, packaging and distribution)

Of course, everything on top of reimplemented features: up to NNLO evolution and matching, EXA/EXP/TRN (and more flexible variations), α_s evolution, interpolation, …

PineAPPL

PineAPPL Is Not an Extension of APPL

[arXiv: 2008.12789]

This is an interpolation grids format, mainly designed and developed by Christopher Schwan.

We decide to adopt as the base interface between the different building blocks for the whole theory predictions infrastructure.

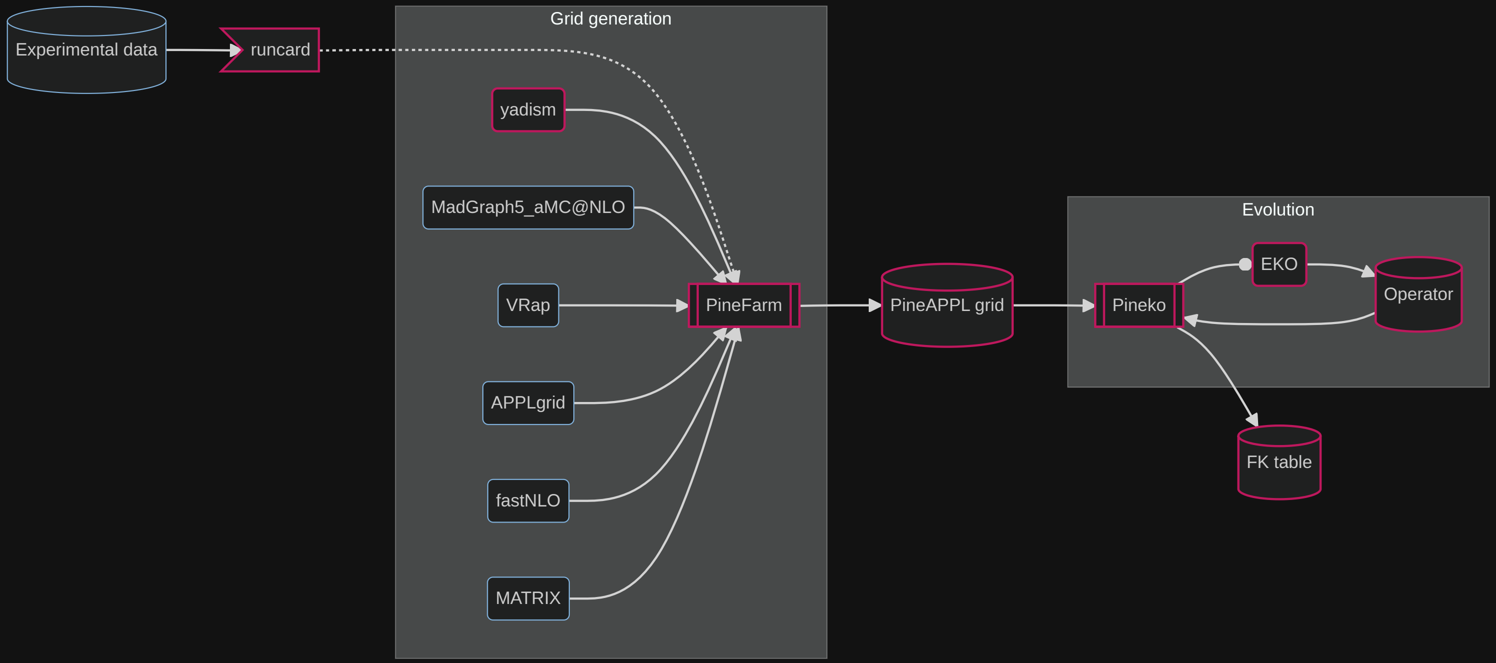
In order to this, it ships:

- a performant Rust library
    - maintaining a dedicated file format
- a versatile CLI
    - for quick inspection and frequent tasks execution, e.g.:
    - convolution, conversion, optimization, pull calculation, plotting, ...
- Python bindings
    - powering the integration with the theory pineline
- C/C++/Fortran interface
    - for direct grid filling by MC generators
- converters from other common formats (APPLgrid, fastNLO)

PINELINE

Industrialized theory predictions

[arXiv: 2211.10447]



Special focus on reproducibility.

Alternative Methodology

Function space

Actually a large one

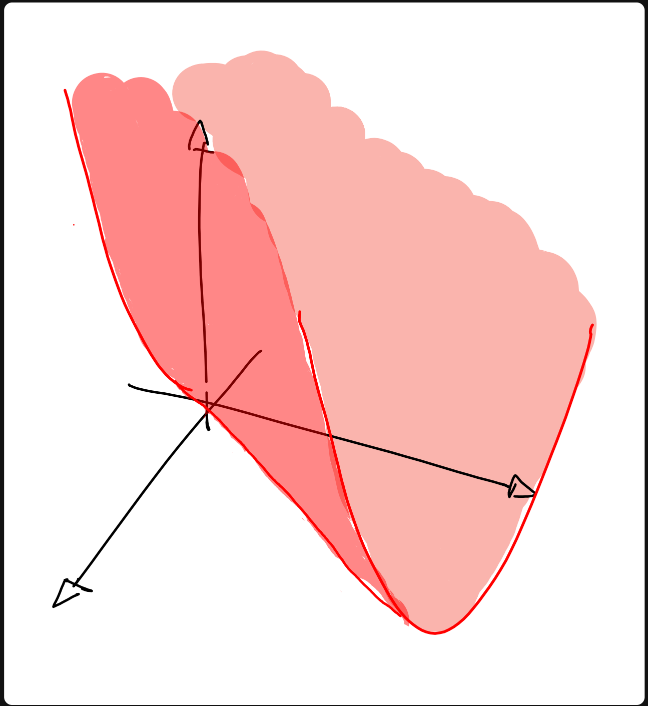
A function $f : \mathbb{R} \to \mathbb{R}$ (or suitable intervals) lives in an infinite-dimensional space.

This has a simple consequence:

Under-determination

Fitting an **unknown function** on a finite number of data is always an **under-determined** problem.

How to choose a solution, when **many** are available and **equivalent**?



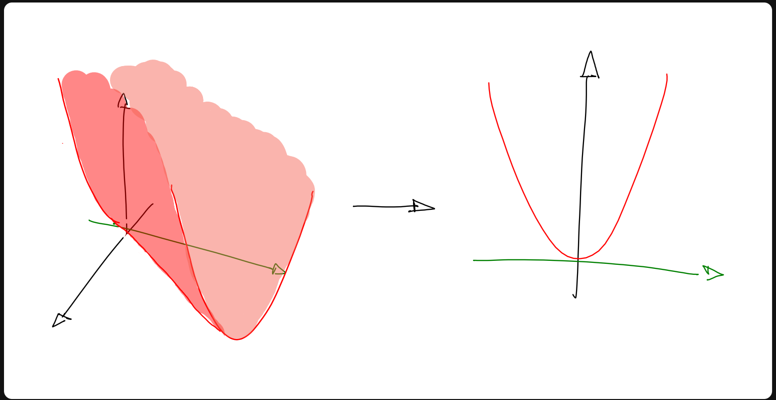
Assumptions

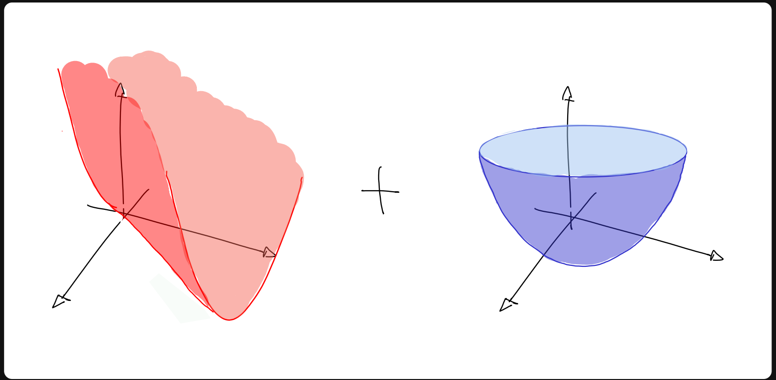
Declined in two main options

Slicing	Regularization
One consists in reducing the number of parameters, by slicing a suitable hyperplane	The second approach removes the zero-direction by adding some regularization.

In PDF language this corresponds to the choice of a fixed parametrization. A suitable one can also remove loss-function zero-directions from the space.

This is what the Neural Network (and its training algorithm) is doing under the hood.





Bayes

Something conceptually simple, but still powerful

Typical examples of ML are:

- image and speech recognition
- generative tasks
- style transfer
- and so on...

All these problems have in common:

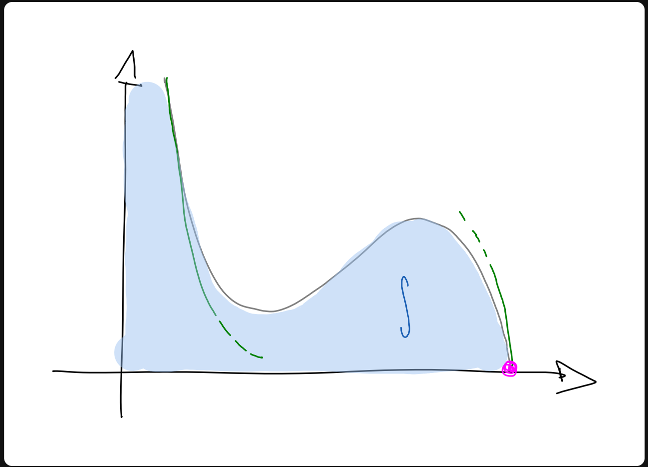
very high-dimensional objects, with poor analytical/algorithmic insight on their structure

Working out an explicit and effective representation for them would be difficult.

This is not the case for PDFs!

Math language description and clear analytic properties are at hand:

sum rules, power-like behavior, ...

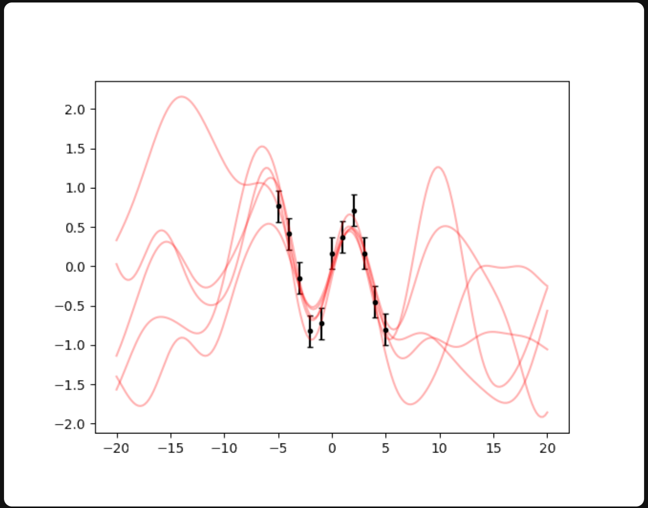


$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

Gaussian Processes

The prior

Like with NN we can limit the slicing*, using a suitable regularization, here coming from the prior.



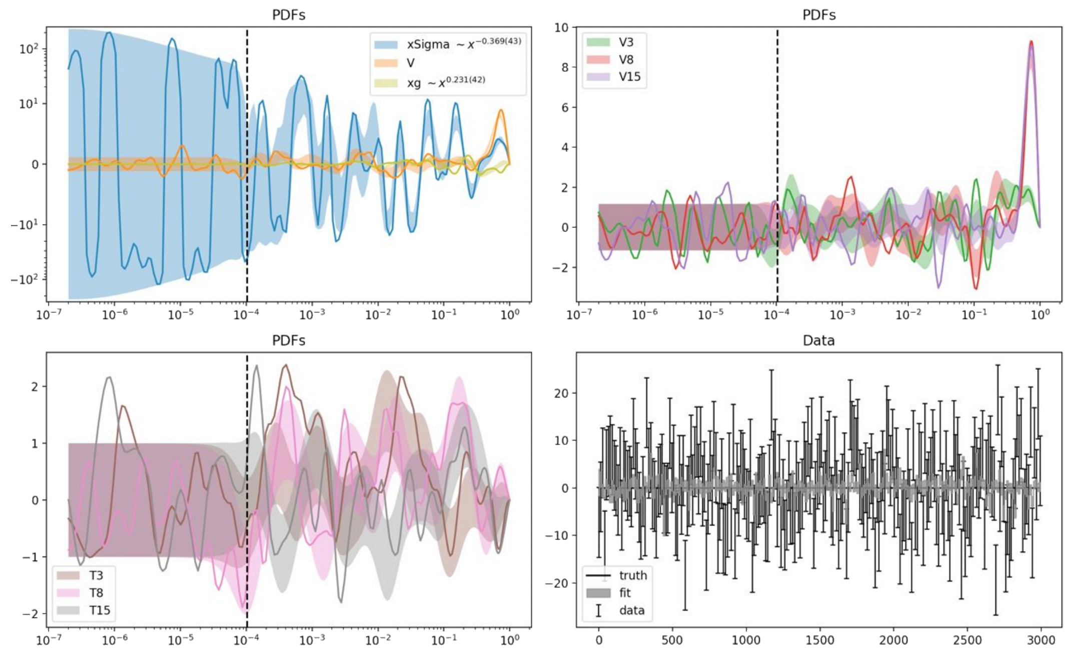
Essentially a multi-Gaussian with a metric-driven kernel, with the motivation that is simple, and sufficiently flexible.

Basic ideas:

- *parametrization exactly our delivery: we use PDF values at grid points (we would no expose more degrees of freedom anyhow, so no need to use them)
- transformations data are not in the PDF space, but we can use linear and non-linear (quadratic) transformations
- sum rules the Gaussian process allow us to impose them analytically (in practice, it is easier to impose them as zero-error data, but it is only a technicality)
- integrability integrability and extrapolation behavior it is implemented as constraints on hyper-parameters

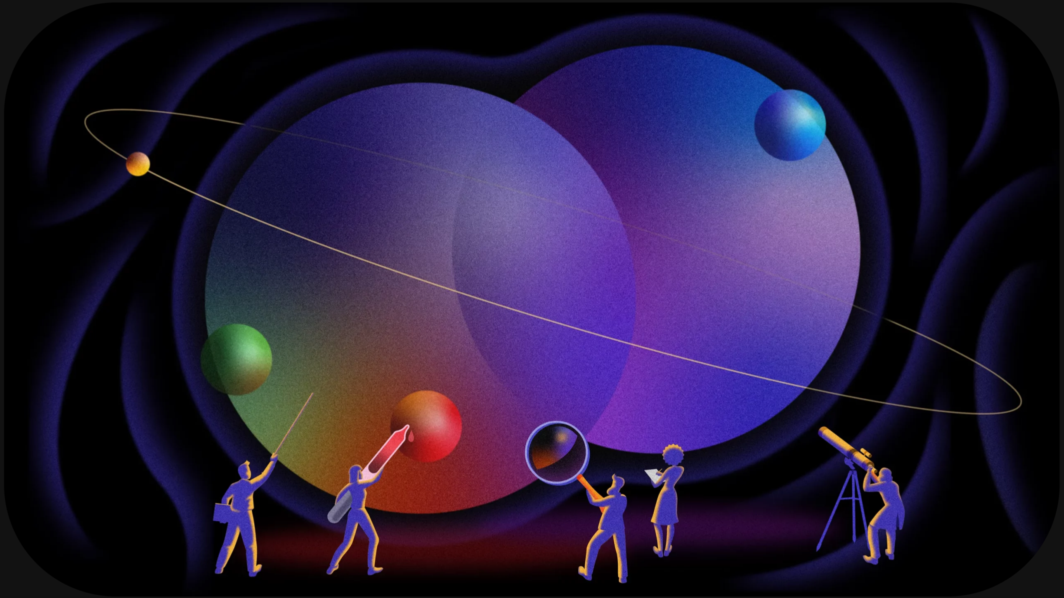
Prototype

Just a POC on completely fake data.



Applications

Intrinsic Charm



Charm in the proton

Heavier but there

[arXiv: 2208.08372]

A charm component in the proton is not a novelty, since it is generated perturbatively by DGLAP evolution, in a FNS with 4 flavors or more (by gluon splitting).

But it is possible to also have a charm of different origin

- perturbative: DGLAP generated
- intrinsic: generated by non-perturbative dynamics
- fitted: the component arising in the boundary condition from the fit

The NNPDF4.0 charm component it is not directly *intrinsic*, since the fit is done in the 4FNS.

UNVEILING THE INTRINSIC COMPONENT

Experimental data and predictions

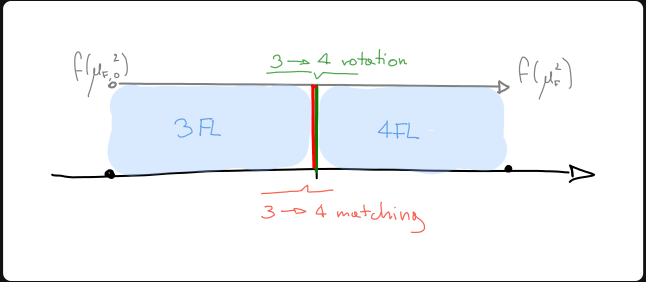
NNPDF4.0 parametrization in 4FNS

Matching: 4FNS ➤ 3FNS
@ NNLO or N3LO

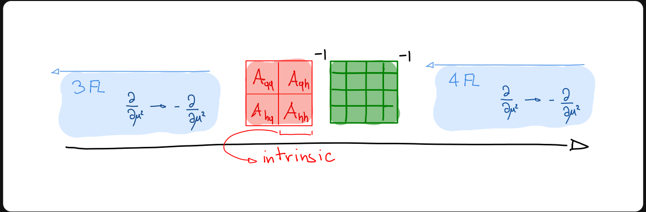
3FNS PDFs: including ***scale-independent*** **intrinsic charm component**

PDF and MHO uncertainties

The Operator Matrix Element (OME) $\mathbf{A}^{(n_f)}(\mu_h^2)$ is partially known up to N^{3}LO.

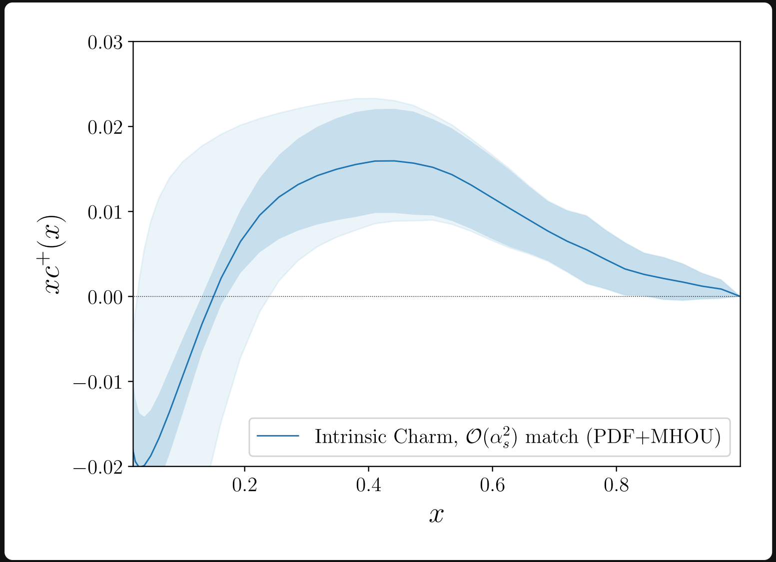


Inverse operator (the OME can be inverted either *perturbatively* or *numerically*)



$$\mathbf{f}^{(n_f+1)}(\mu_{F,1}^2) = \left[\mathbf{E}^{(n_f+1)}(\mu_{F,1}^2 \leftarrow \mu_h^2)\mathbf{R}^{(n_f)}\mathbf{A}^{(n_f)}(\mu_h^2)\mathbf{E}^{(n_f)}(\mu_h^2 \leftarrow \mu_{F,0}^2)\right] \times \mathbf{f}^{(n_f)}(\mu_{F,0}^2)$$

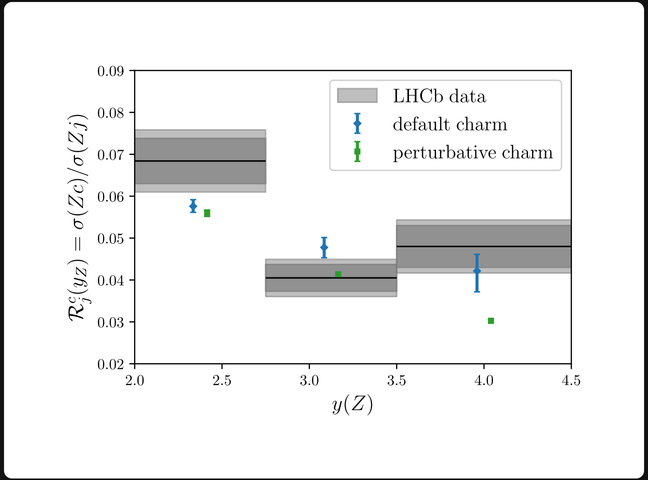
EVIDENCE



In 3FNS a valence-like peak is present.

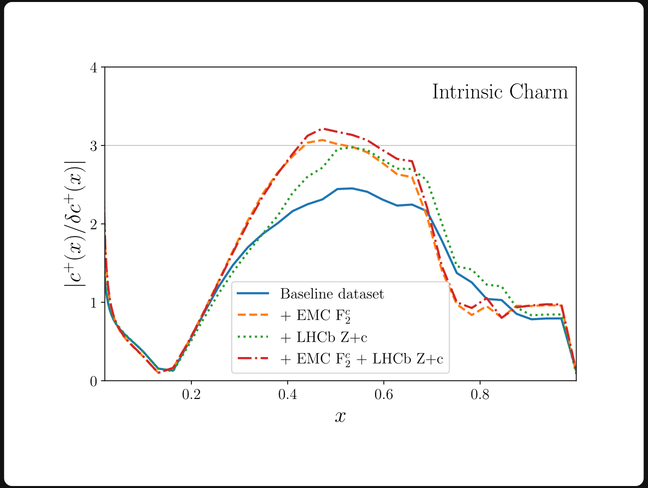
- for $x \leq 0.2$ the perturbative uncertainties are quite large
- the carried momentum fraction is within 1%

Predictions and stability



The intrinsic charm fit produces **better predictions** for some charm-sensitive datasets, *not included in the fit*.

The **evidence is stable** under dataset variations, *including* charm-sensitive datasets.



Conclusions

Summary 📜

All my work has been focused on Parton Distribution Functions, and most of it as a member of the NNPDF collaboration.

My main research topic has been the development of a new generation of fast and reproducible theory predictions for HEP processes, especially focused on the PDF fitting tasks, including:

- a Deep Inelastic Scattering library, `yadism`
- a DGLAP evolution library, `eko`
    - and keep benchmarking the two of them (`banana`)
- the development of the integrated Pineline, involving:
    - contributions to PineAPPL
    - the development of two further integrations, `pineko` and `pinefarm`

Moreover, I worked on a series of applications:

- intrinsic charm evidence in NNPDF4.0
- study of the Drell–Yan forward-backward asymmetry

And other more methodology-related topics, like investigation of PDF positivity beyond LO.

OUTLOOK 🔭

From now on...

The major ongoing efforts are focused on:

- putting into production the whole Pineline, replacing the old theory predictions providers
- adding new* features to the baseline fit:
    - MHOU @ NNLO, with *theory covariance matrix* formalism `[arXiv: 1906.10698]`
    - QED evolution & luxQED
    - N^3LO addition

**with respect to NNPDF4.0*

But they are not the only ones:

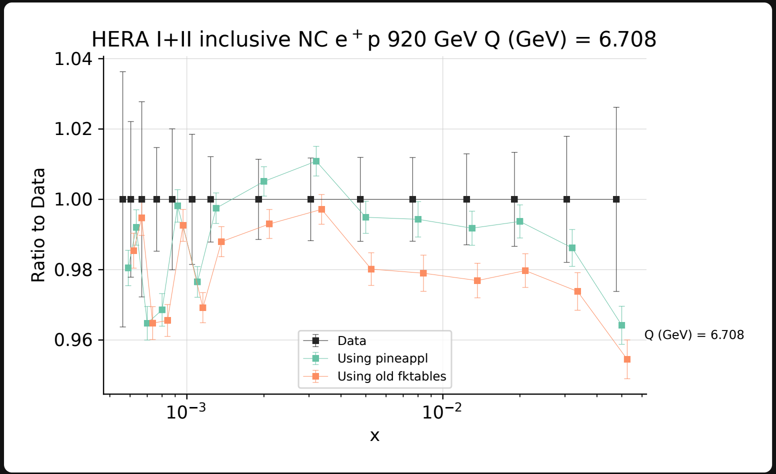
- studies on alternative methodologies keep going
- more theory predictions: polarized, time-like, NNLO grids and CC-DIS
- more phenomenological investigations (ν structure functions)
- technical improvements (speed-up, distributed computation)
- user-facing improvements (docs, tutorials, public API, bundling)

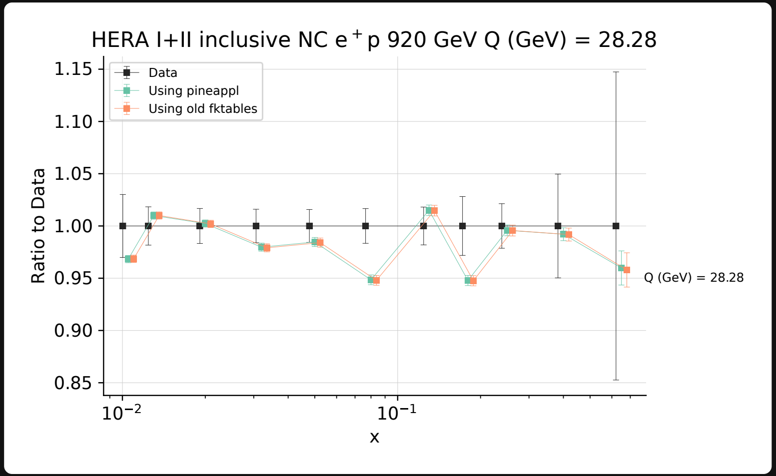
Many thanks to the whole NNPDF collaboration!

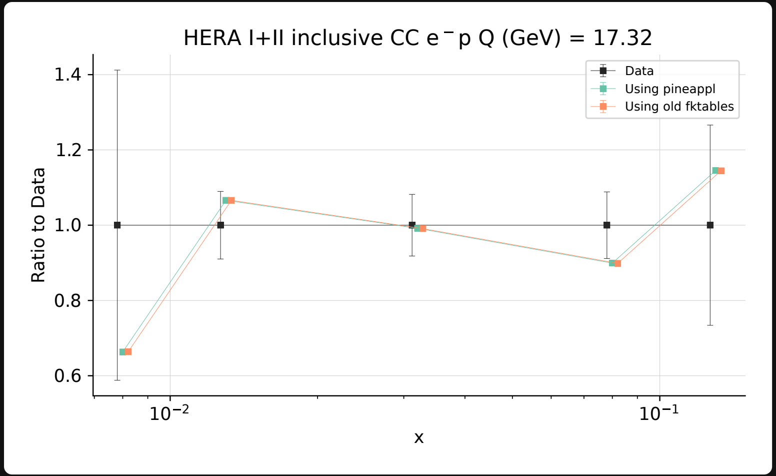
… and especially to our Milan group!!

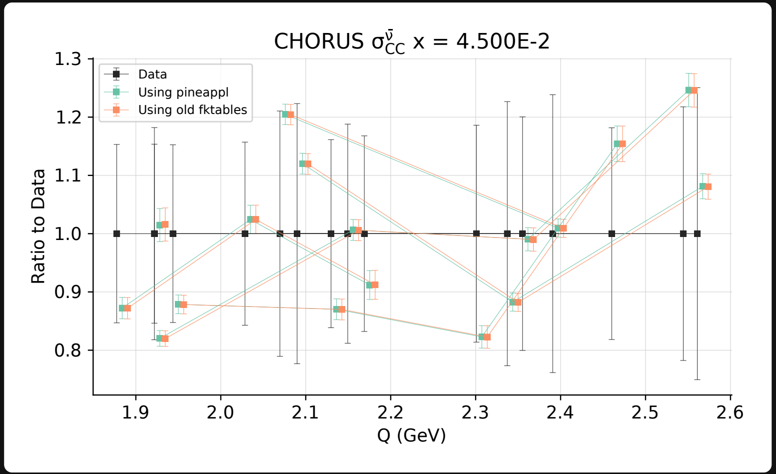
Backup

Backup Index

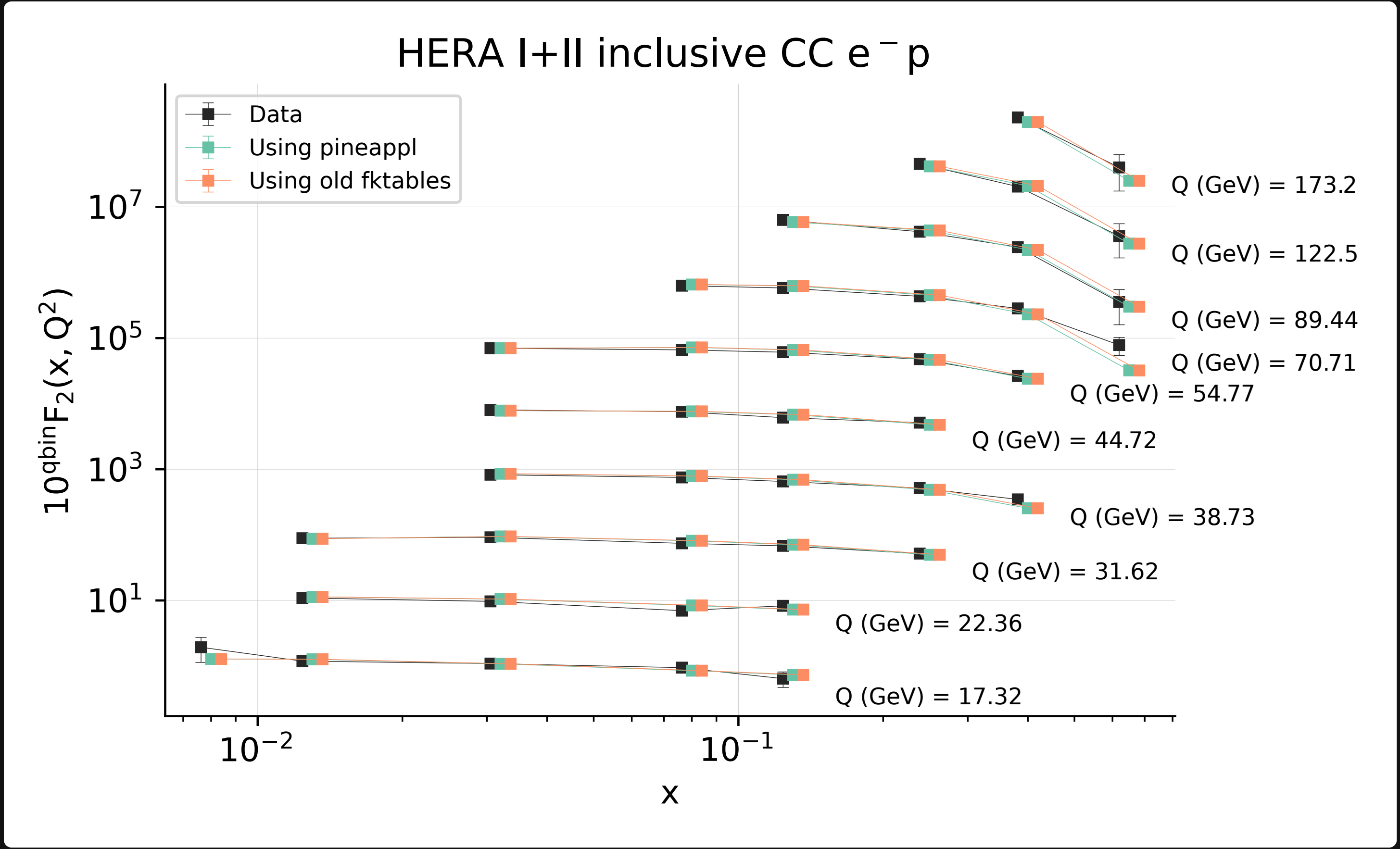

HERA I+II inclusive NC e$^+$p 920 GeV Q (GeV) = 6.708
Ratio to Data
x
Data
Using pineappl
Using old fktables
Q (GeV) = 6.708


HERA I+II inclusive NC e$^+$p 920 GeV Q (GeV) = 28.28
Ratio to Data
x
Data
Using pineappl
Using old fktables
Q (GeV) = 28.28

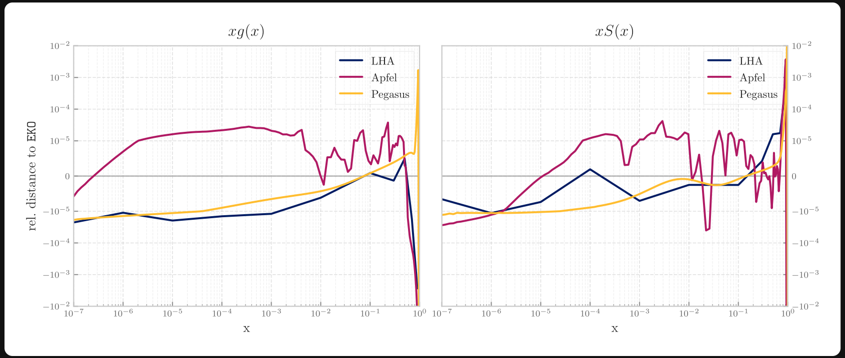

HERA I+II inclusive CC e$^-$p Q (GeV) = 17.32
Ratio to Data
x
Data
Using pineappl
Using old fktables

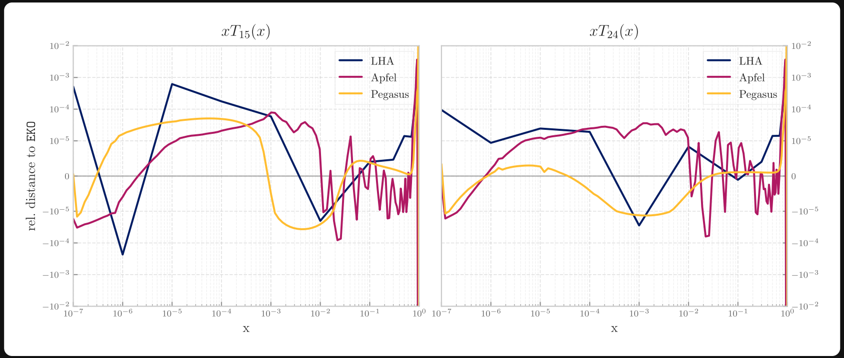

CHORUS $\sigma_{CC}^{\bar{\nu}}$ x = 4.500E-2
Ratio to Data
Q (GeV)
Data
Using pineappl
Using old fktables

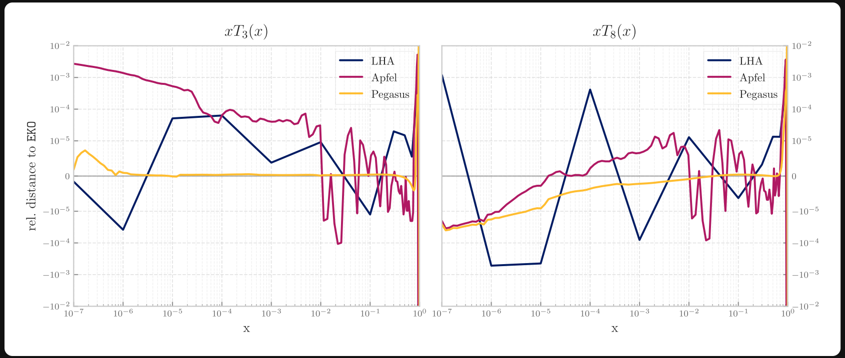
Yadism vs APFEL

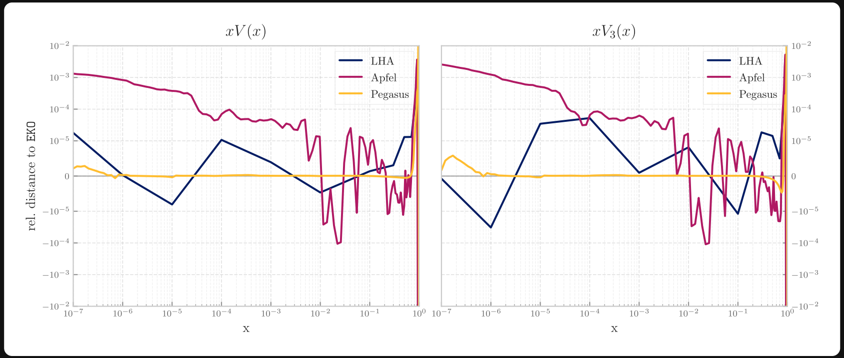


EKO vs LHA BENCHMARK

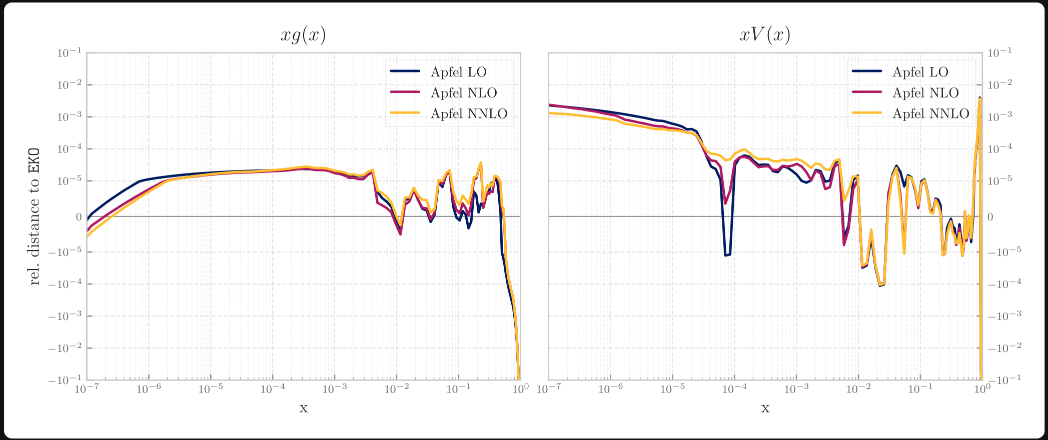


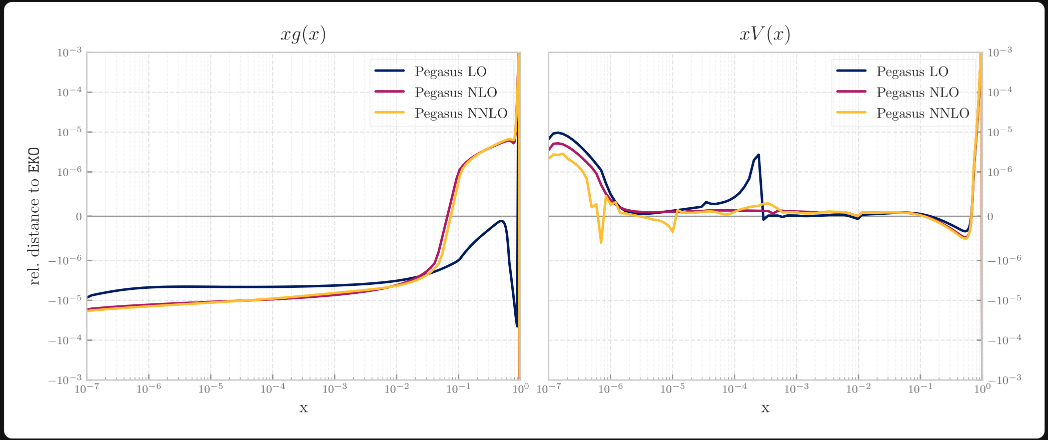




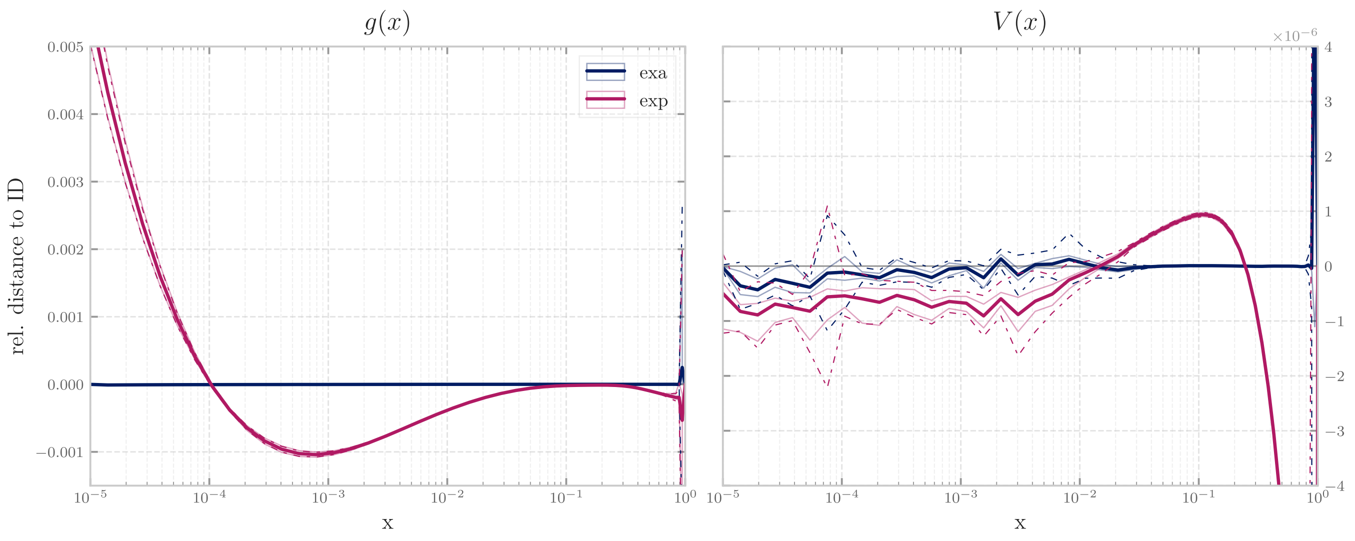


EKO vs APFEL & PEGASUS

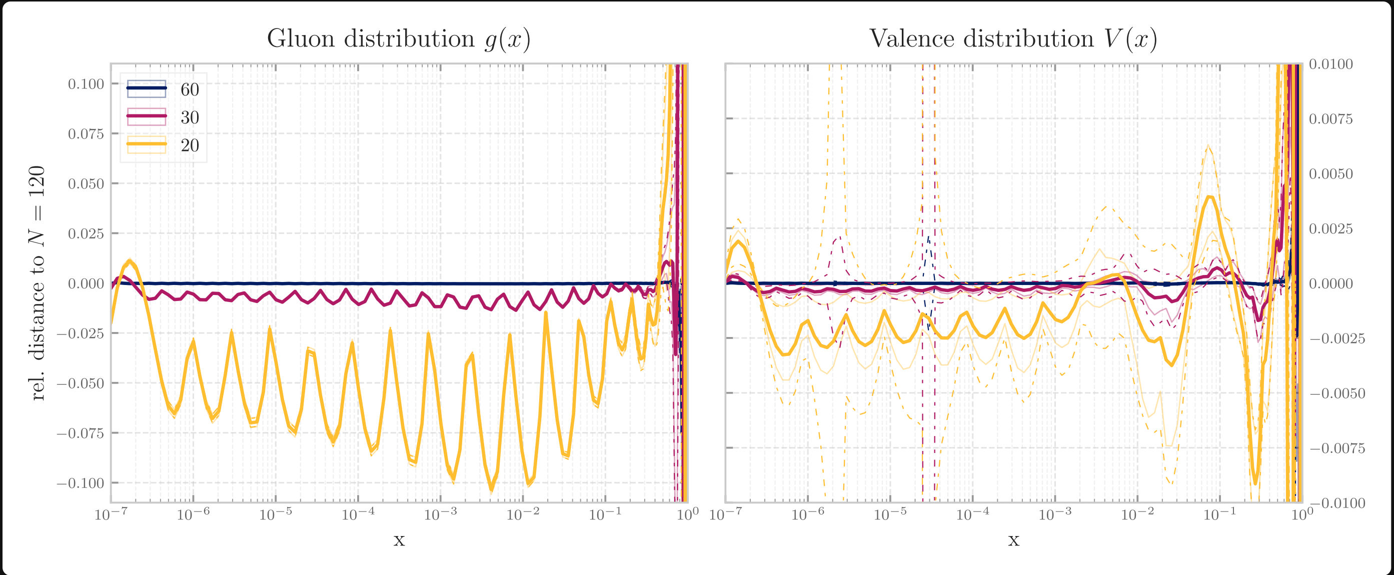




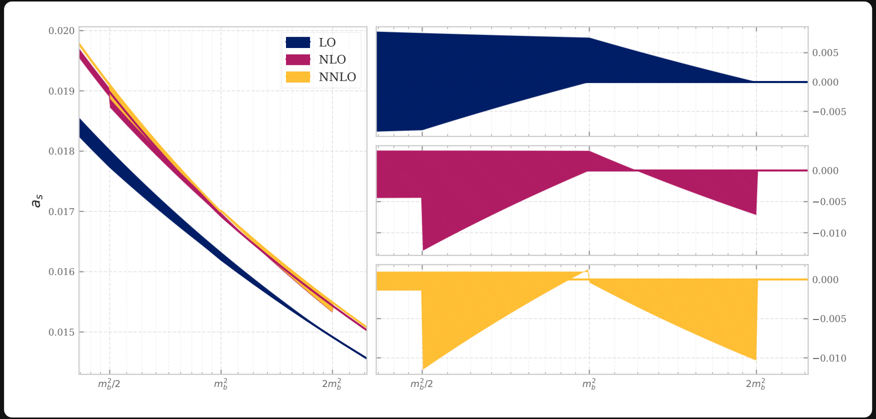
EKO BACKWARD CLOSURE

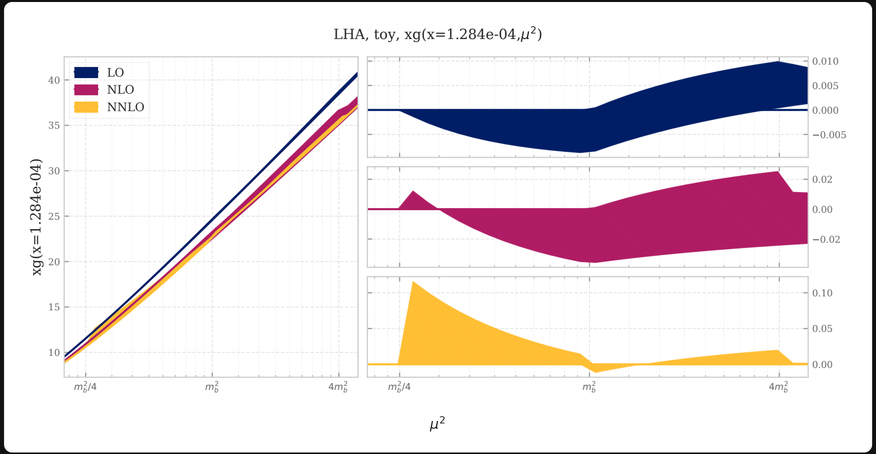


EKO INTERPOLATION

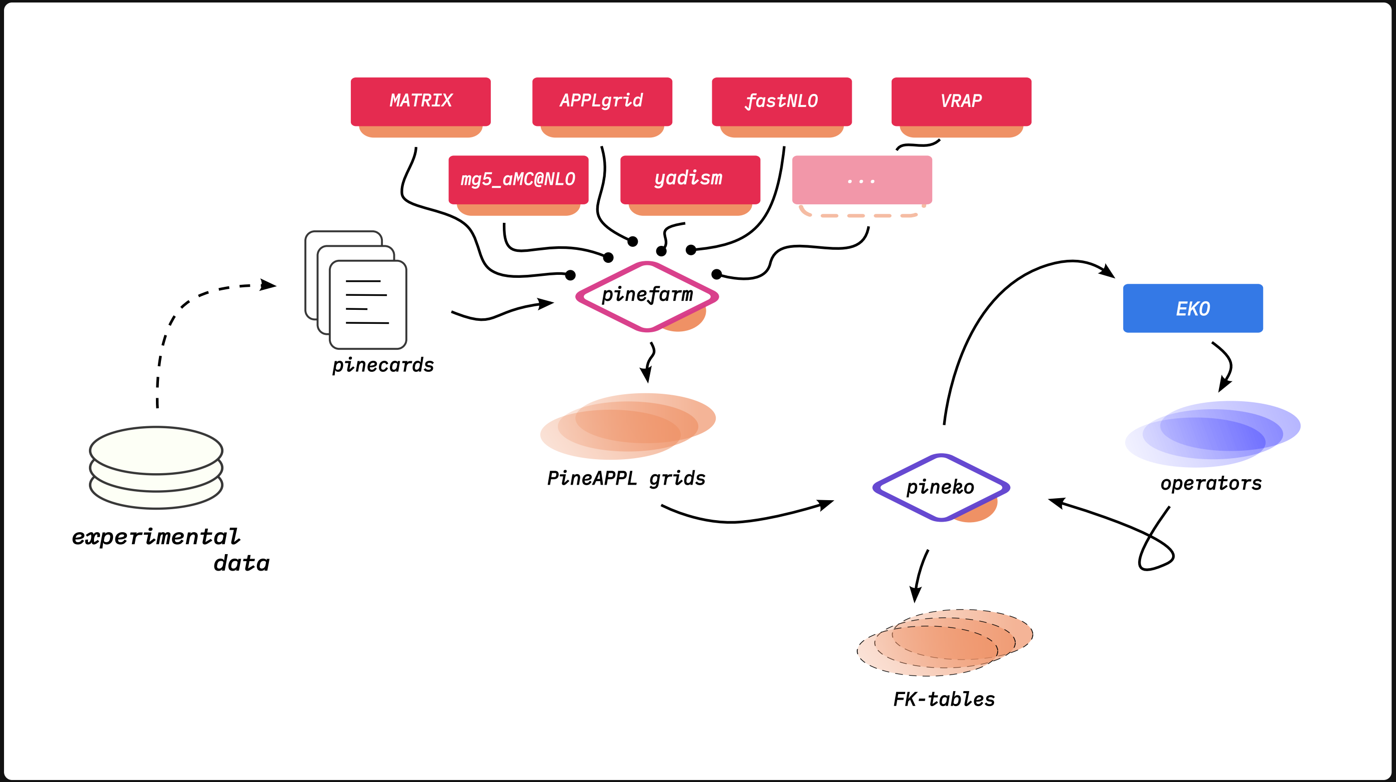


EKO MATCHING: α_S AND PDFs





Old diagram



MATRIX
APPLgrid
fastNLO
VRAP
mg5_aMC@NLO
yadism
...
pinecards
pinefarm
EKO
PineAPPL grids
operators
pineko
experimental data
FK-tables

Theory covariance matrix - requisites

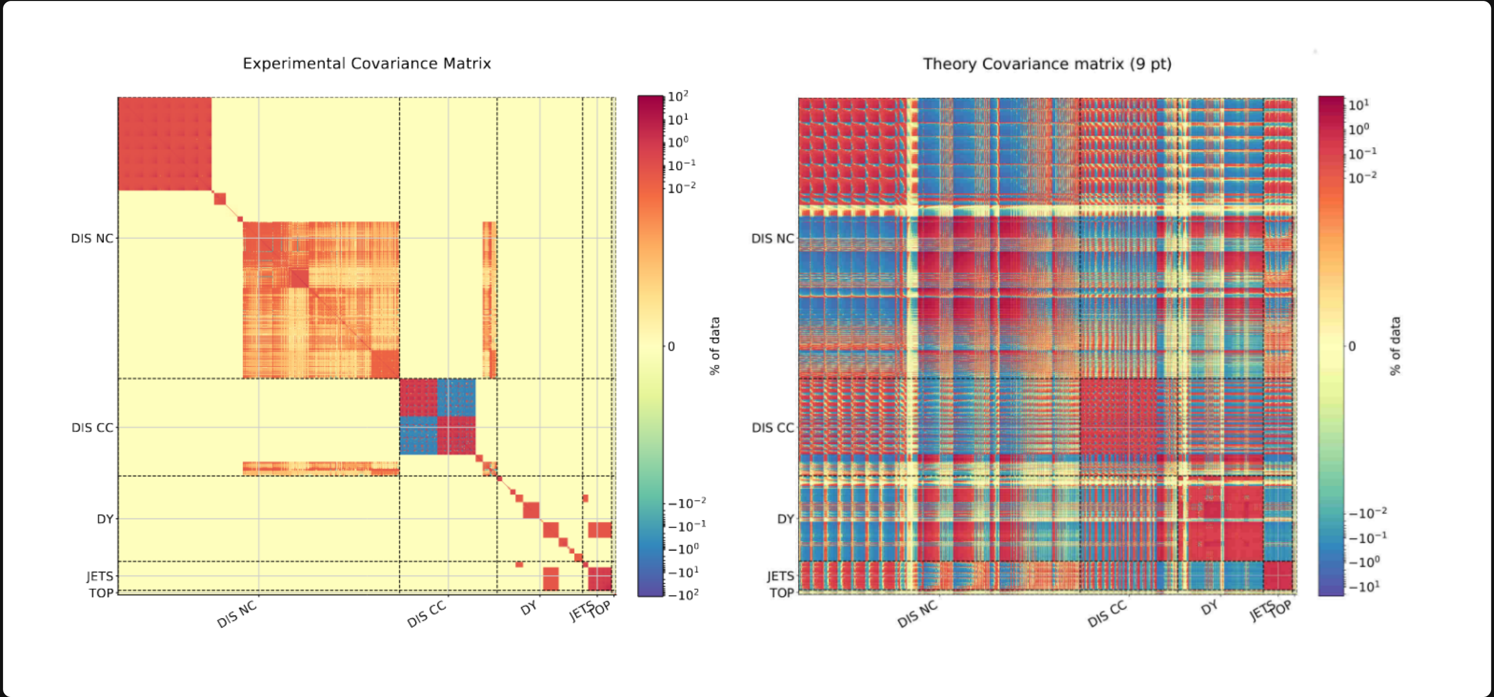
- the theory covariance has to be *generated by some shift vectors* $\Delta_i(\vec{\kappa})$ - the vectors should be *proportional to the difference of predictions* obtained by a theory variation $T_i(\vec{\kappa})$ and the default theory in which $\vec{\kappa} = \vec{\kappa}_0$

$$\Delta_i(\vec{\kappa}) = c_i(\vec{\kappa}) \left(T_i(\vec{\kappa}) - T_i(\vec{\kappa}_0)\right)$$
$$S_{ij} = \sum_{\vec{\kappa} \in \mathcal{V}_{ij}} \Delta_i(\vec{\kappa}) \Delta_j(\vec{\kappa})$$

- it has to be also *positive semi-definite*, as required for any covariance matrix

$$v_i S_{ij} v_j > 0 \qquad \forall v \in \mathbb{R}^{n_{\mathrm{data}}}$$

Theory covariance matrix - example



Alternative prescriptions

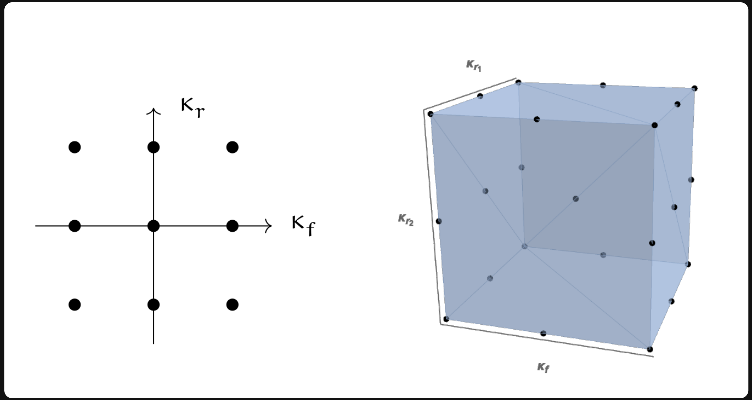
It is possible to restrict the arbitrary normalization $c_i(\vec{\kappa})$ with a sets of reasonable constraints:

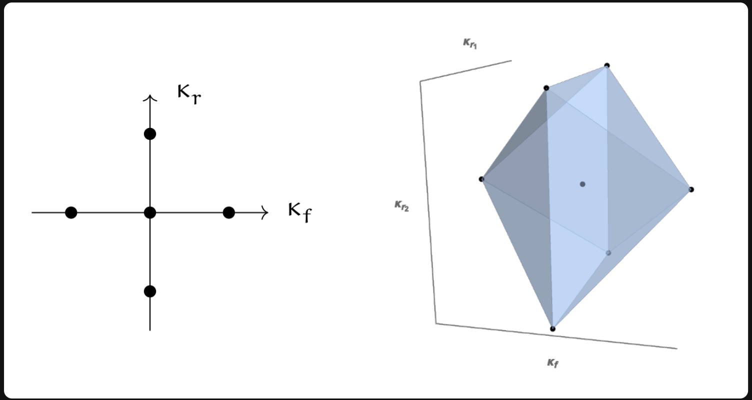
- isomorphic spaces: for each pair of disjoint data sets, the prescription space should be the same (it might be different for on/off-diagonal, but always independent on the data set)
- normalization: the normalization of the theory covmat should not scale (i.e. depend) with the size of the prescription space (the cardinality of the finite set)

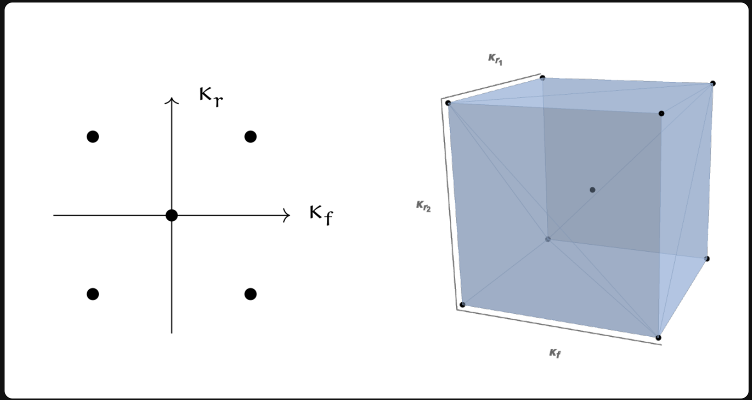
However, there is a unique choice with two sensible options:

- fully factorized space: together with the other requisites, it generates a unique set of prescriptions (one for each n-points)
- κ_F-sliced space: the factorization scale is rather special, so instead of considering a space that is factorized with that as well, it is possible to just accept that the space is fully factorized for each value of the factorization scale κ_F

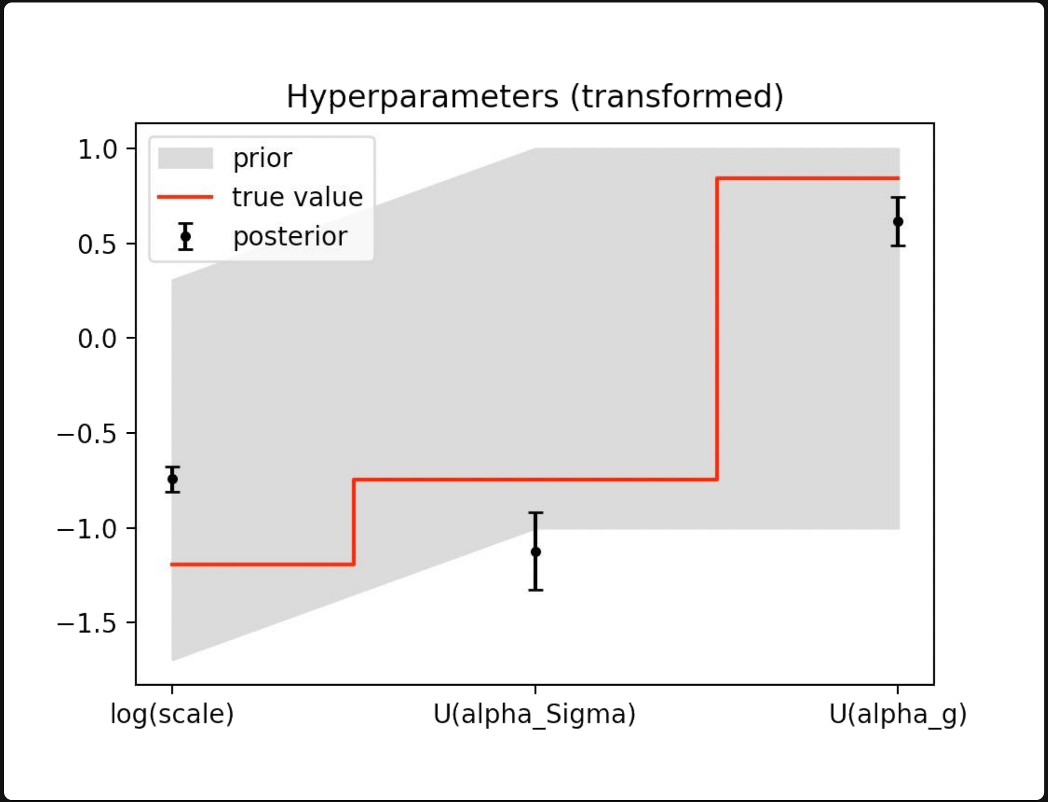
Point prescriptions



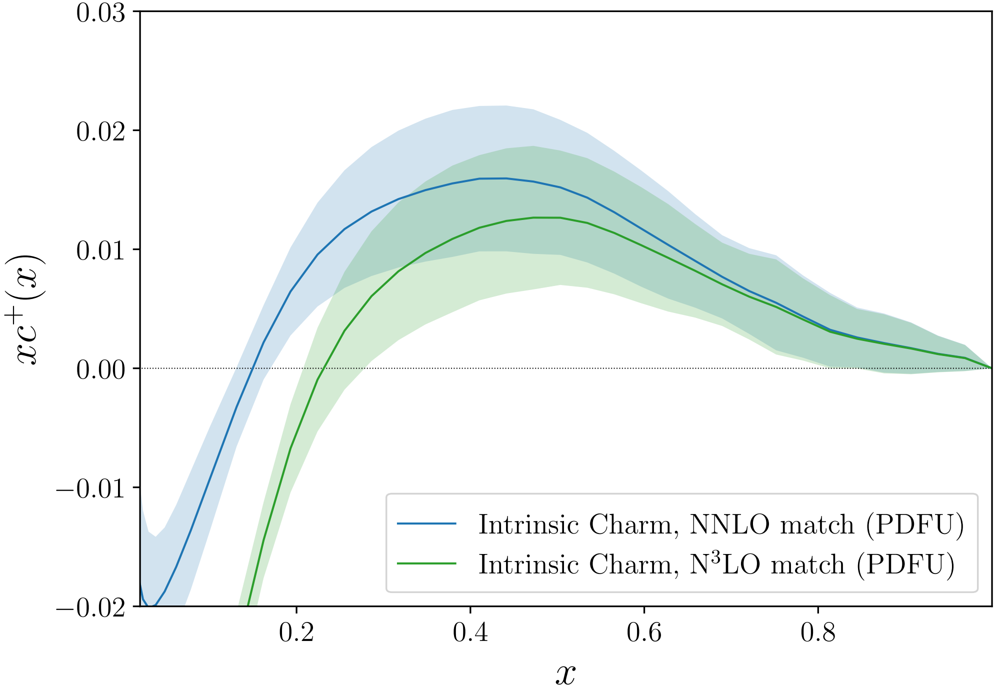




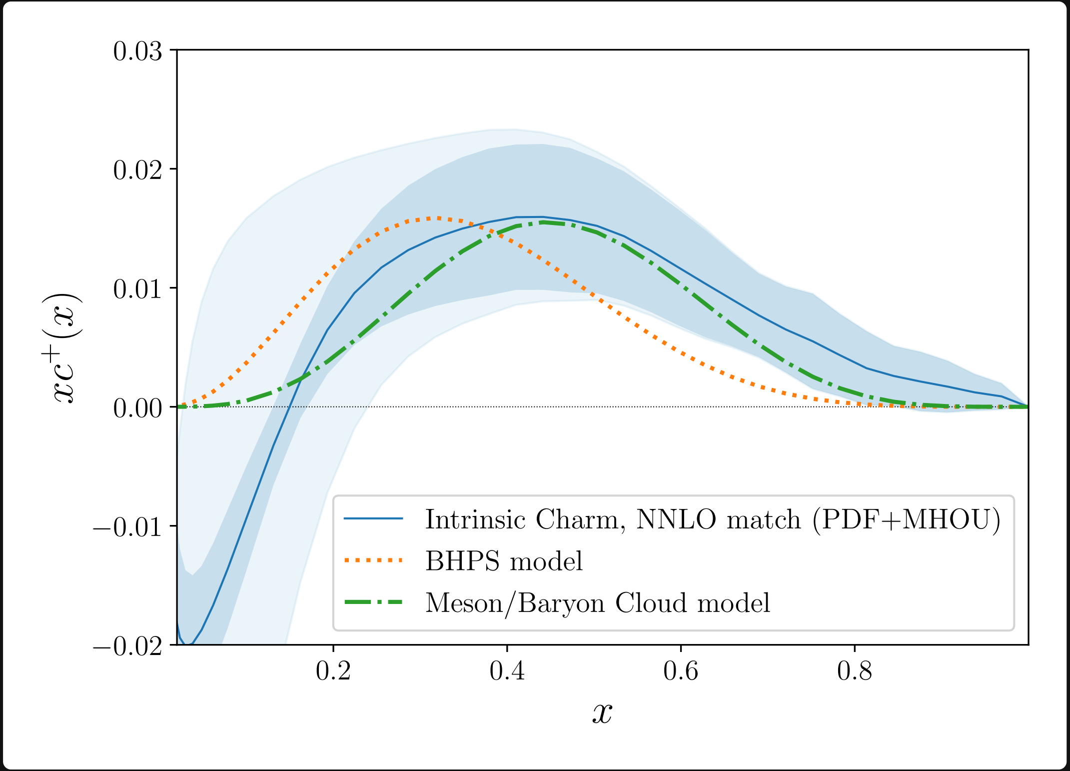
Hyperparameters



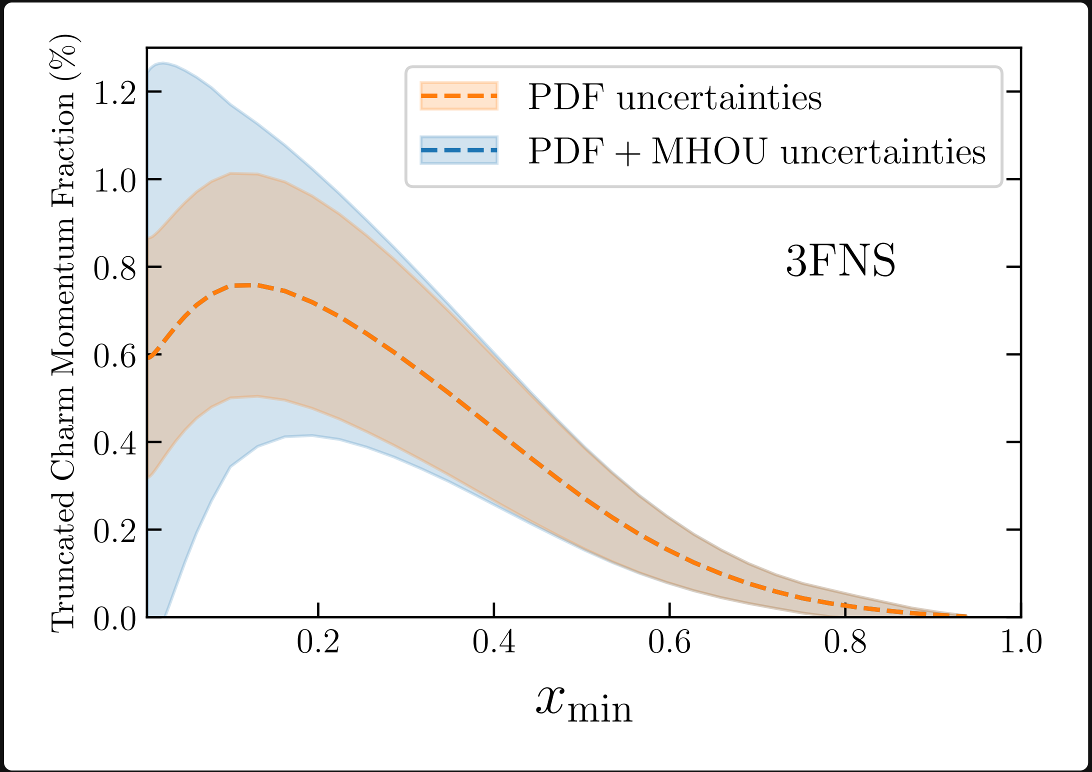
Intrinsic matching effect: NNLO vs N^3LO



Models comparison



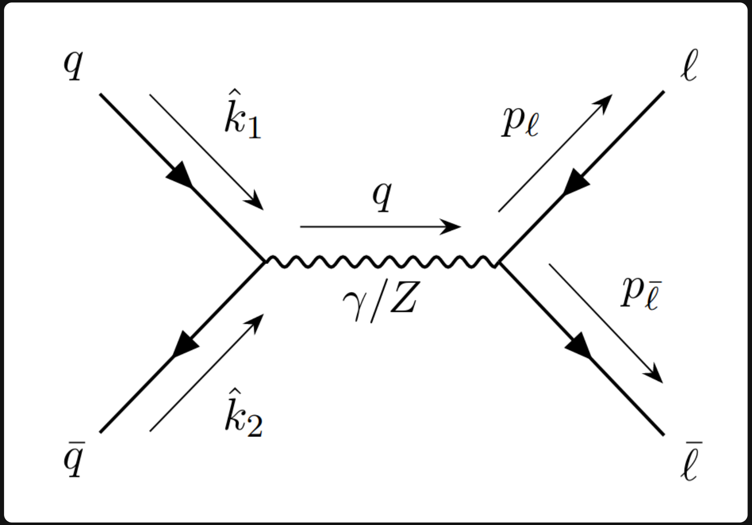
Charm truncated momentum fraction



Drell–Yan Forward-Backward Asymmetry

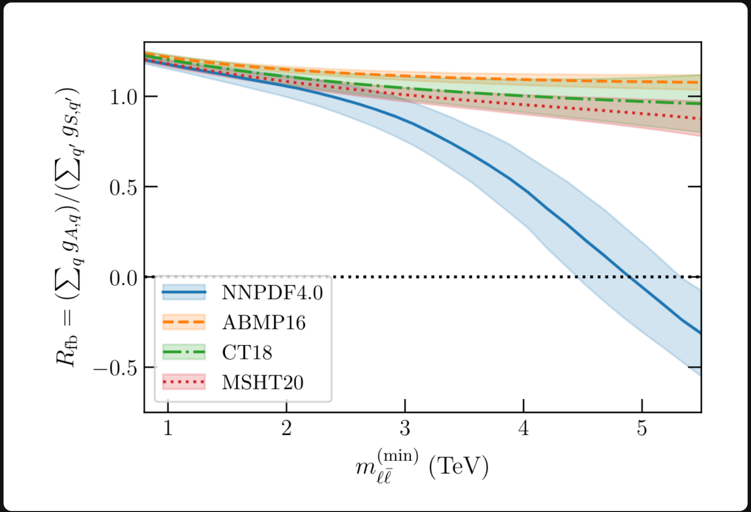
[arXiv: 2209.08115]

This is an interesting case study for PDF extrapolation.



$$A_{\mathrm{fb}}(\cos\theta^*) \equiv \frac{\frac{\mathrm{d}\sigma}{\mathrm{d}\cos\theta^*}(\cos\theta^*) - \frac{\mathrm{d}\sigma}{\mathrm{d}\cos\theta^*}(-\cos\theta^*)}{\frac{\mathrm{d}\sigma}{\mathrm{d}\cos\theta^*}(\cos\theta^*) + \frac{\mathrm{d}\sigma}{\mathrm{d}\cos\theta^*}(-\cos\theta^*)}, \qquad \cos\theta^* > 0$$

Extrapolation

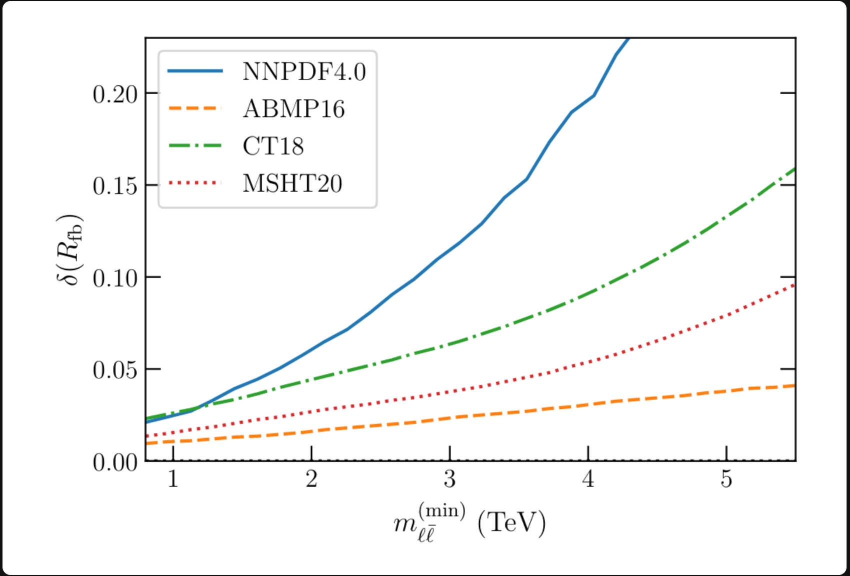


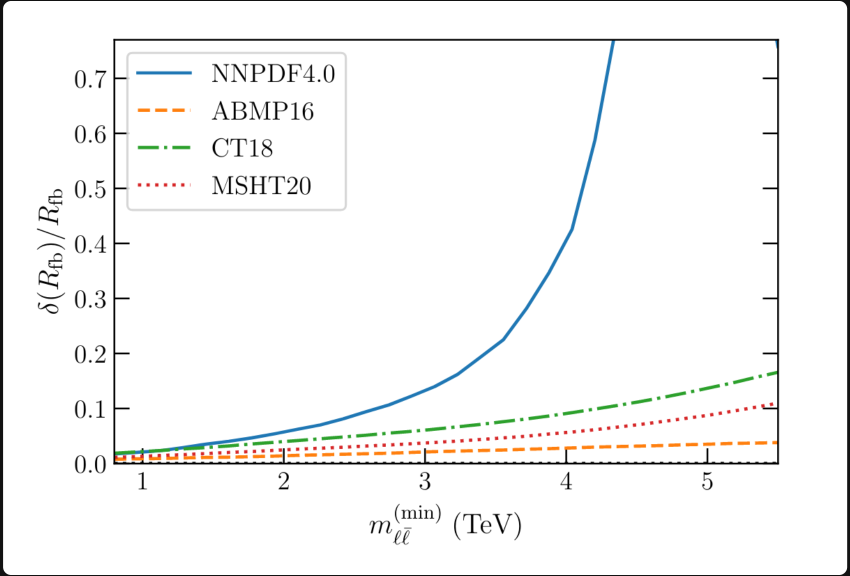
LO expression:

$$A_{\rm fb}(\cos\theta^*) = \frac{\cos\theta^*}{(1+\cos^2(\theta^*))} R_{\rm fb} \qquad R_{\rm fb} \equiv \frac{\sum_q g_{A,q}}{\sum_{q'} g_{S,q'}}$$

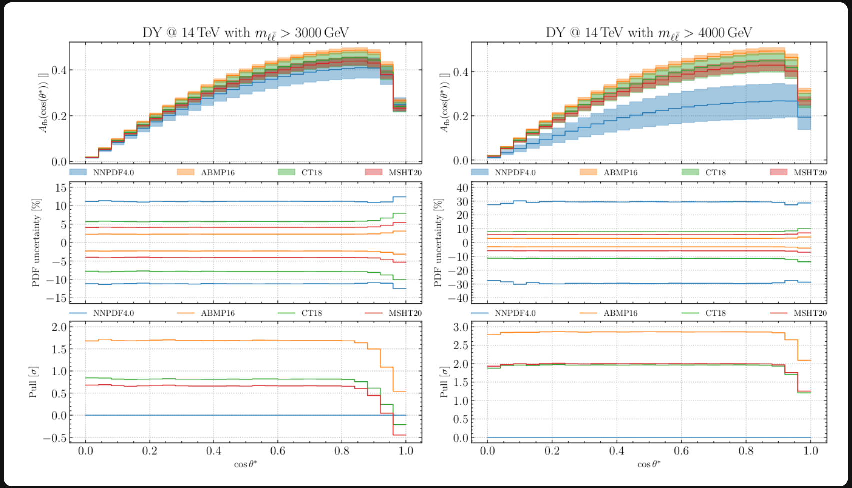
$$g_{A,q} = \frac{\pi\alpha^2}{3s} \int_{m_{\ell\bar{\ell}}^{\min}}^{\sqrt{s}} \frac{{\rm d}m_{\ell\bar{\ell}}}{m_{\ell\bar{\ell}}} A_q(m) \int_{\log(m_{\ell\bar{\ell}}/\sqrt{s})}^{\log(\sqrt{s}/m_{\ell\bar{\ell}})} {\rm d}y_{\ell\bar{\ell}}\, \mathcal{L}_{A,q}(m_{\ell\bar{\ell}}, y_{\ell\bar{\ell}})$$

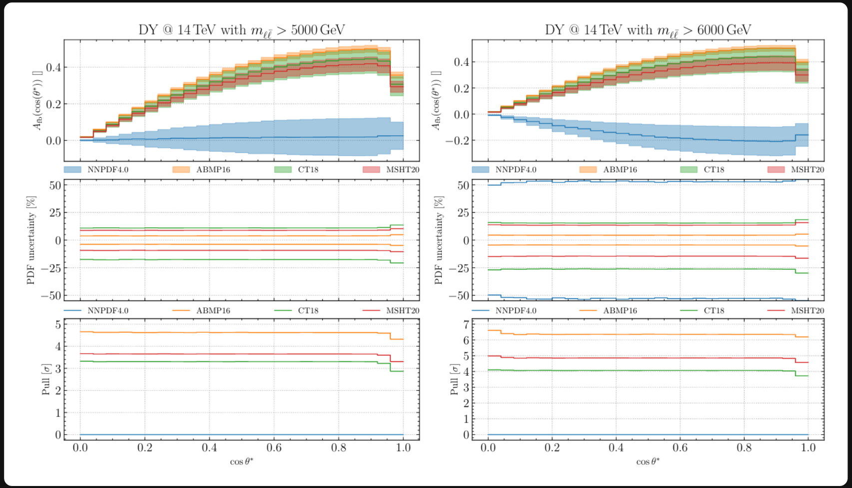
R_{fb} RELATIVE UNCERTAINTY





The NLO picture





Thanks Anthony Fu, and the other contributors, for the wonderful Slidev framework.