Extraction of Compton From Factors with Gepard and PyTorch

Krešimir Kumerički

University of Zagreb, Croatia

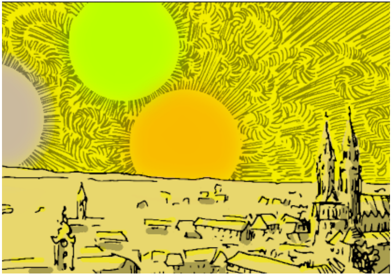
REVESTRUCTURE workshop

10 July, 2023, Zagreb

Croatian Science Foundation

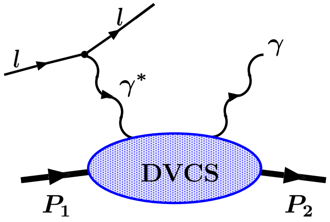
EUROPSKI STRUKTURNI I INVESTICIJSKI FONDOVI

STRONG 2020



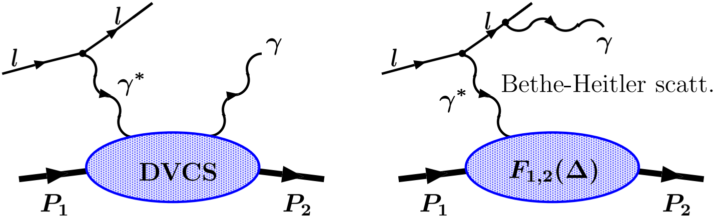
Deeply virtual Compton scattering

- Measured in leptoproduction of a real photon:



Deeply virtual Compton scattering

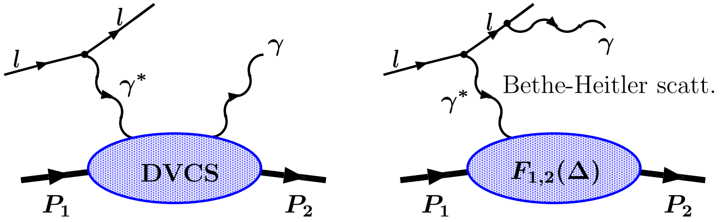
- Measured in leptoproduction of a real photon:



- There is a background process

Deeply virtual Compton scattering

- Measured in leptoproduction of a real photon:



- There is a background process but it can be used to our advantage:

$$\sigma \propto |\mathcal{T}_{\rm DVCS}|^2 + |\mathcal{T}_{\rm BH}|^2 + \mathcal{T}^*_{\rm DVCS}\mathcal{T}_{\rm BH} + \mathcal{T}_{\rm DVCS}\mathcal{T}^*_{\rm BH}$$

- Using $\mathcal{T}_{\rm BH}$ as a referent "source" enables measurement of the phase of $\mathcal{T}_{\rm DVCS}$ → proton "holography" [Belitsky and Müller '02]

DVCS cross section

$$d\sigma \propto |\mathcal{T}|^2 = |\mathcal{T}_{\rm BH}|^2 + |\mathcal{T}_{\rm DVCS}|^2 + \mathcal{I} \, .$$

$$\mathcal{I} \propto \frac{-e_\ell}{\mathcal{P}_1(\phi)\mathcal{P}_2(\phi)} \left\{ c_0^{\mathcal{I}} + \sum_{n=1}^{3} \left[c_n^{\mathcal{I}} \cos(n\phi) + s_n^{\mathcal{I}} \sin(n\phi) \right] \right\},$$

$$|\mathcal{T}_{\rm DVCS}|^2 \propto \left\{ c_0^{\rm DVCS} + \sum_{n=1}^{2} \left[c_n^{\rm DVCS} \cos(n\phi) + s_n^{\rm DVCS} \sin(n\phi) \right] \right\},$$

- Choosing polarizations (and charges) we focus on particular harmonics:

$$c^{\mathcal{I}}_{1,\rm unpol.} \propto \left[F_1 \,\mathfrak{Re}\, \mathcal{H} - \frac{t}{4M_p^2} F_2 \,\mathfrak{Re}\, \mathcal{E} + \frac{x_{\rm B}}{2 - x_{\rm B}} (F_1 + F_2) \,\mathfrak{Re}\, \widetilde{\mathcal{H}} \right]$$

[Belitsky, Müller et. al '01–'14]

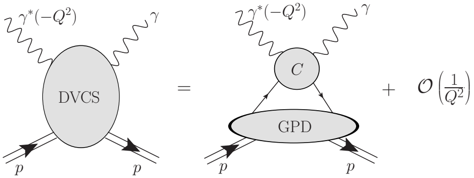
DVCS $\longrightarrow$ CFFs $\longrightarrow$ GPDs

- At leading order DVCS cross-section depends on four complex

Compton form factors (CFFs)

$$\mathcal{H}(\xi,t,Q^2), \quad \mathcal{E}(\xi,t,Q^2), \quad \widetilde{\mathcal{H}}(\xi,t,Q^2), \quad \widetilde{\mathcal{E}}(\xi,t,Q^2)$$

- [Collins et al. '98]



- CFFs are convolution:

$$^a\mathcal{H}(\xi,t,Q^2) = \int \mathrm{d}x \; C^a(x,\xi,\frac{Q^2}{Q_0^2}) \; H^a(x,\eta=\xi,t,Q_0^2)$$

$a=q,G$

- $H^a(x,\eta,t,Q_0^2)$ — Generalized parton distribution (GPD)

[Müller '92, et al. '94, Ji, Radyushkin '96]

Gepard - public code for GPD analysis

gepard

Search docs

CONTENTS:

- Software documentation
- Data sets
- Publications
- Credits

https://gepard.phy.hr/datasets.html

» Tool for studying the 3D quark and gluon distributions in the nucleon

View page source

Gepard

Tool for studying the 3D quark and gluon distributions in the nucleon

Gepard is software for analysis of three-dimensional distribution of quarks and gluons in hadrons, encoded in terms of the so-called Generalized Parton Distributions (GPDs).

This web site has manifold purpose:

- Documentation of the software
- Examples of the use of software
- Interface to various representations of results: numerical and graphical
- Interface to datasets used in analyses: numerical and graphical

Contents:

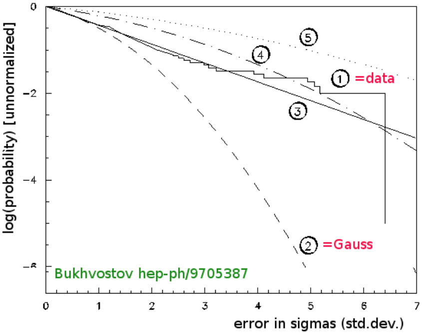
Neural nets and Gepard

- "Old" (Fortran+Python) Gepard (`pyfortran` branch on GitHub) used in-house modified `PyBrain` NNet library — not maintained, difficult to install and work with. [I. Ćorić master's thesis]: `TensorFlow` adaptation
- "Official" (pure Python) Gepard package (`master` branch) — no neural nets
- `torch` branch on the GitHub: new `PyTorch` neural net interface (non-neural models will not work presently)

Neural Nets Method

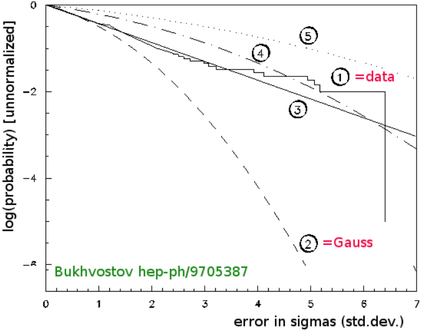
Problems with standard fitting approaches

1. Choice of fitting function introduces theoretical bias leading to systematic error which cannot be estimated (and is likely much larger for GPDs(x, η, t) than for PDFs(x).
2. Propagation of uncertainties from experiment to fitted function is difficult. Errors in actual experiments are not always Gaussian.

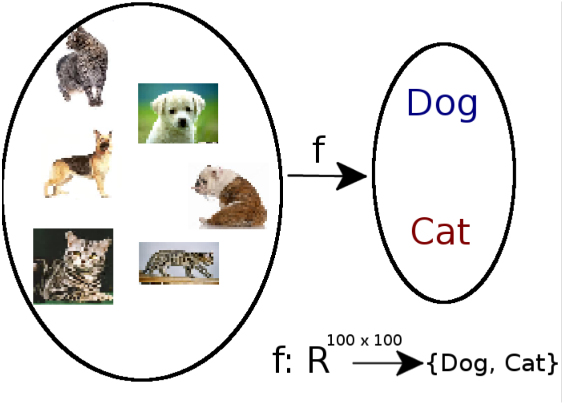


Problems with standard fitting approaches

1. Choice of fitting function introduces theoretical bias leading to systematic error which cannot be estimated (and is likely much larger for GPDs(x, η, t) than for PDFs(x).$\rightarrow$ NNets
2. Propagation of uncertainties from experiment to fitted function is difficult. Errors in actual experiments are not always Gaussian.$\rightarrow$ Monte Carlo error propagation

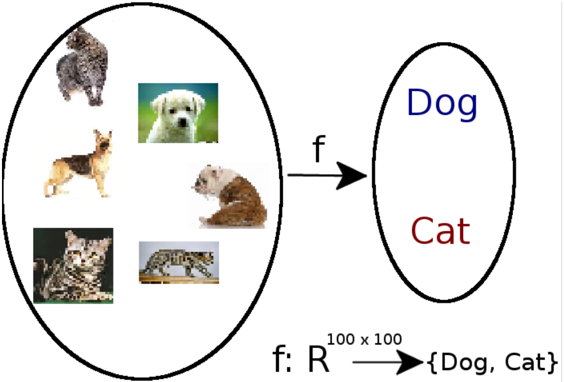


Introduction to neural networks: Cat-or-dog mapping*



*Homage to Vladimir Igorevich Arnold

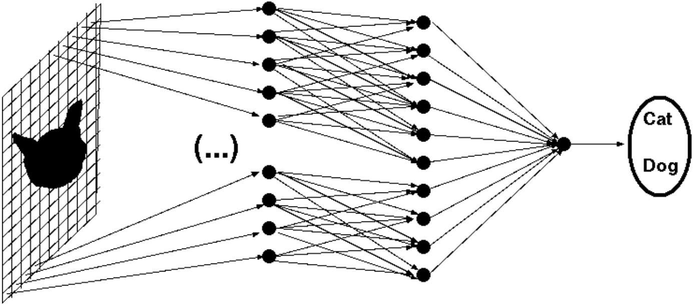
Introduction to neural networks: Cat-or-dog mapping*



- How to represent function f by a computer algorithm?
- $\longrightarrow$ neural networks, learning machines, AI

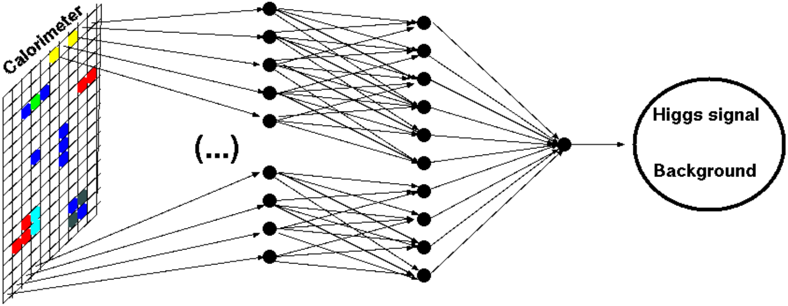
*Homage to Vladimir Igorevich Arnold

Cat-or-dog mapping by neural network

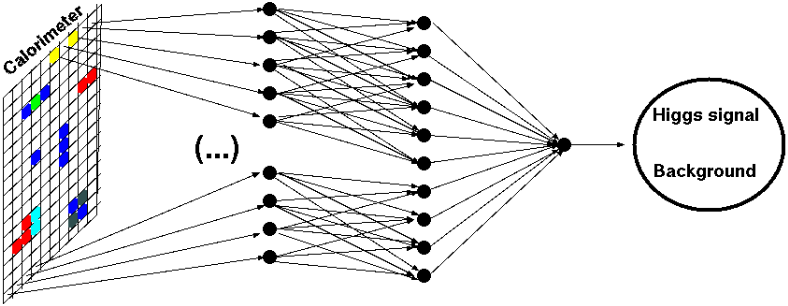


- Parameters ("weights") of neural network adjusted by "training" it on many samples
- Neural network becomes a representation of function f.
- Neural networks are capable of generalization: they successfully classify objects not seen during training

Neural networks in high-energy physics

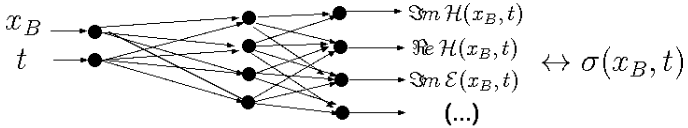


Neural networks in high-energy physics



- Neural networks can be used
  - in place of triggers (hardware NN)
  - in place of simple "cuts" of detektor data (software NN)
- Used by everybody in HEP these days ...
- Training usually done on Monte-Carlo simulated events

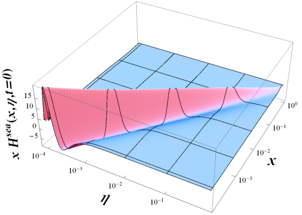
Neural networks as a GPD extraction tool



- Neural network now represents mapping $f : \mathbb{R}^2 \to \mathbb{R}^{n_\mathcal{F}}$.
- We can hope to be able to train neural networks to represent real underlying physical law
- NN approach is successfully applied to PDF fitting by [NNPDF] group and should be even more powerful in GPD fitting with GPDs being less-known functions of more variables.
- [Gepard], [PARTONS], [UVa]

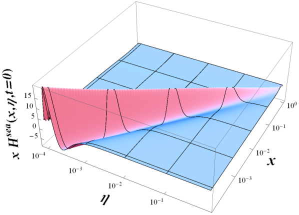
How deep is your net?

- When considering various fancy neural net architectures, keep in mind that we are after this:



How deep is your net?

- When considering various fancy neural net architectures, keep in mind that we are after this:



- . . . and not after this:

Closure tests

Testing the extraction procedure

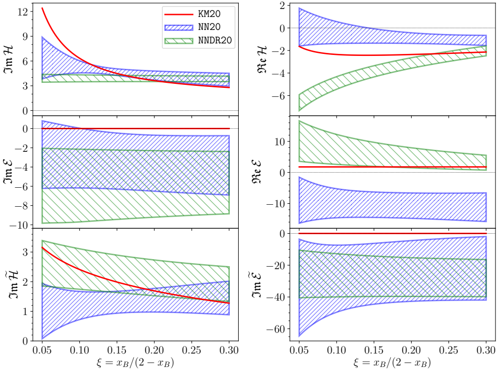
- To each observable many GPDs/CFFs contribute. Can we tell them apart?
- Is the extraction procedure guaranteed to converge to actual underlying physical hadron structure functions?

Testing the extraction procedure

- To each observable many GPDs/CFFs contribute. Can we tell them apart?
- Is the extraction procedure guaranteed to converge to actual underlying physical hadron structure functions?
- Closure [NNPDF] a.k.a. feasibility [PARTONS] test:
  1. Take the known GPD/CFF model - "ground truth"
  2. Generate simulated (mock) data by calculating observables in a certain kinematic range (possibly correspoding to the real measurements of interest)
  3. Apply your fitting/extraction procedure to simulated data
  4. Check that the result is consistent with ground truth

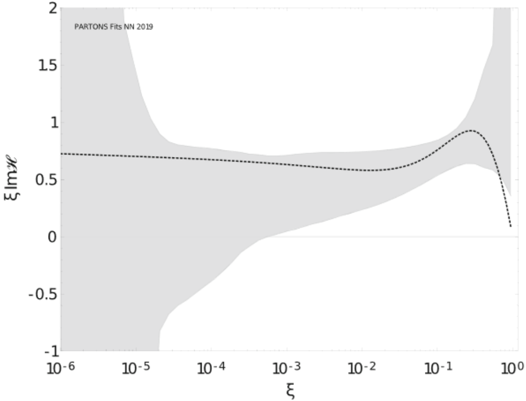
Example of CFF extraction

- [M. Čuić, K.K., A. Schäfer, '20], from JLab data



- Obtained by one-off week-long training session. How reliable are such results?

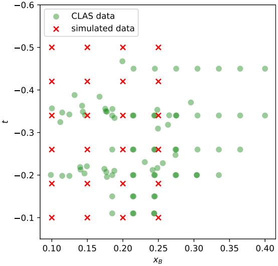
- [NNPDF] group performs closure tests for their PDF neural net fits
- [Moutarde, Sznajder, Wagner '19] showed feasibility of CFF extraction of $\mathfrak{Im}\,\mathcal{H}$ using Goloskokov-Kroll model:



How we tested

- As a ground truth we used KM15 model [K.K. and Müller '15]
- Kinematical points are equidistant, but roughly overlap CLAS6 and CLAS12 kinematics

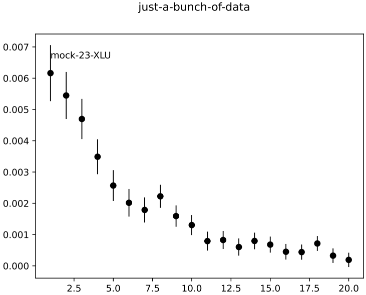
(For speed, $\phi = \pi/4$)



- DVCS observables are a subset of::
  1. helicity dependent and independent cross-sections (X_{LU}, X_{UU})
  2. beam spin asymmetry (A_{LU}) - not an independent observable
  3. beam charge asymmetry (A_{C})
  4. transversal target spin asymmetry ($A_{\mathrm{UT,DVCS}}$)

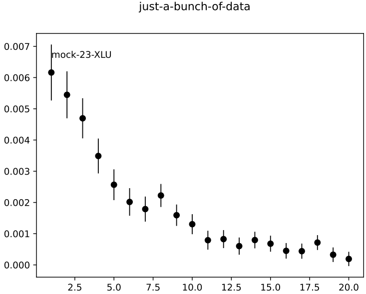
(Almost) toy example (1/2)

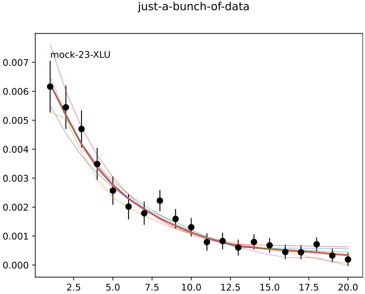
- Only $\mathfrak{Im}\,\mathcal{H}(t)$ (fixed $x_B = 0.2$), only $X_{\rm LU}$.



(Almost) toy example (1/2)

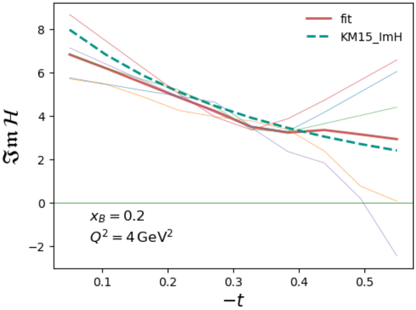
- Only $\mathfrak{Im}\,\mathcal{H}(t)$ (fixed $x_B = 0.2$), only $X_{\rm LU}$.



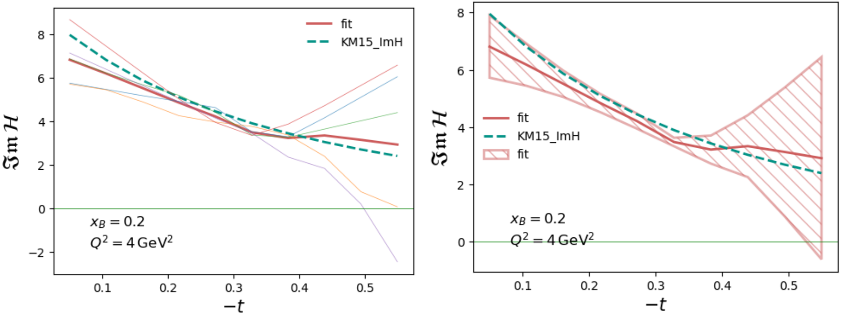


(Unless explicitly specified, x-axis just counts data points, and corresponds to t.)

(Almost) toy example (2/2)

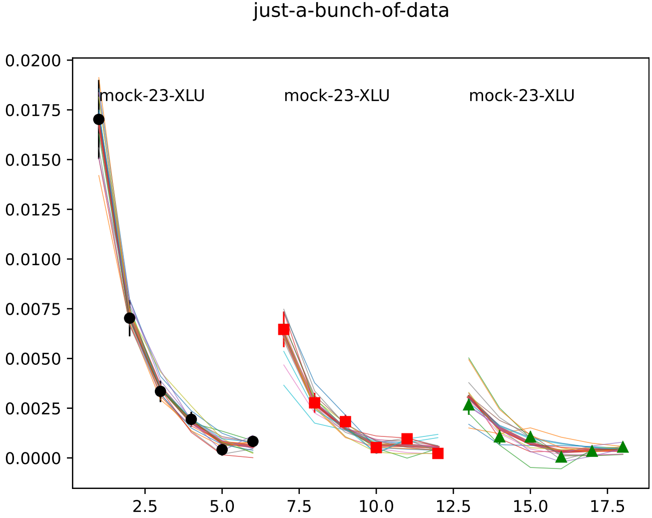


(Almost) toy example (2/2)

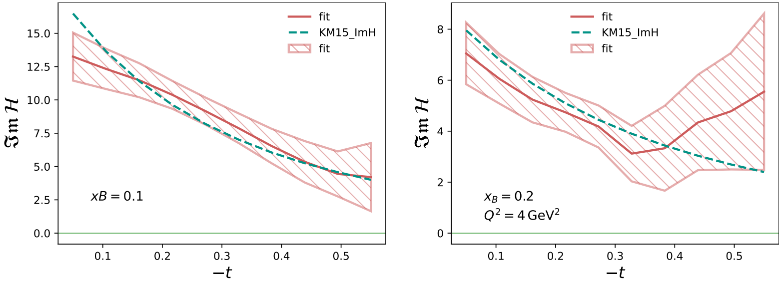


Example 2: $\mathfrak{Im}\,\mathcal{H}$ t and x_B dependence (1/2)

- $\mathfrak{Im}\,\mathcal{H}(x_B, t)$, still only $X_{\rm LU}$.



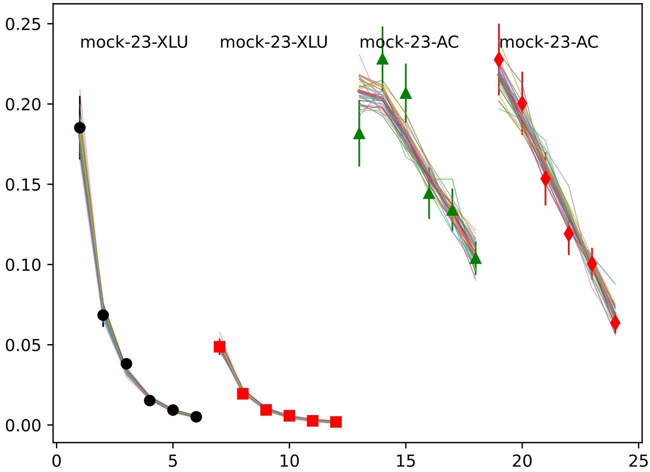
Example 2: $\mathfrak{Im}\,\mathcal{H}$ t and x_B dependence (2/2)



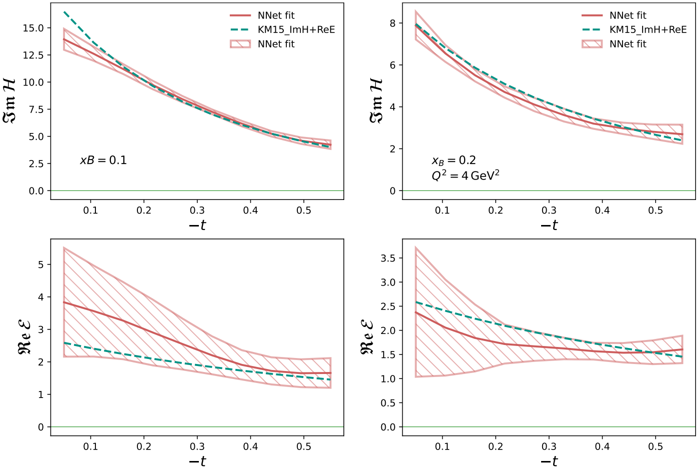
Example 3: $\mathfrak{Im}\,\mathcal{H}$ and $\mathfrak{Re}\,\mathcal{E}$ (1/3)

- $\mathfrak{Im}\,\mathcal{H}(x_B, t)$ and $\mathfrak{Re}\,\mathcal{E}(x_B, t)$ from $X_{\rm LU}$ and $A_{\rm C}$

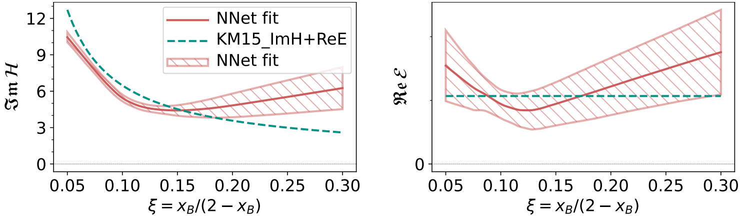
just-a-bunch-of-data



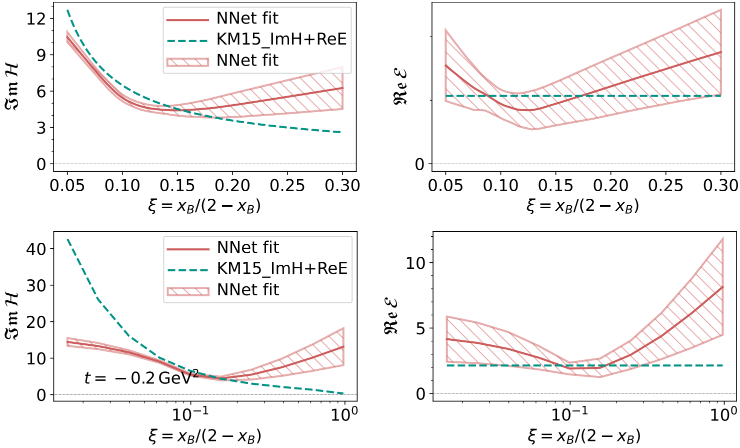
Example 3: $\mathfrak{Im}\,\mathcal{H}$ and $\mathfrak{Re}\,\mathcal{E}$ (2/3)



Example 3: Extrapolation? (3/3)

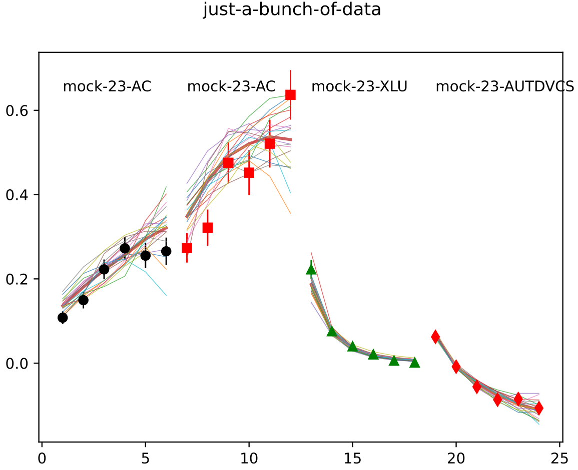


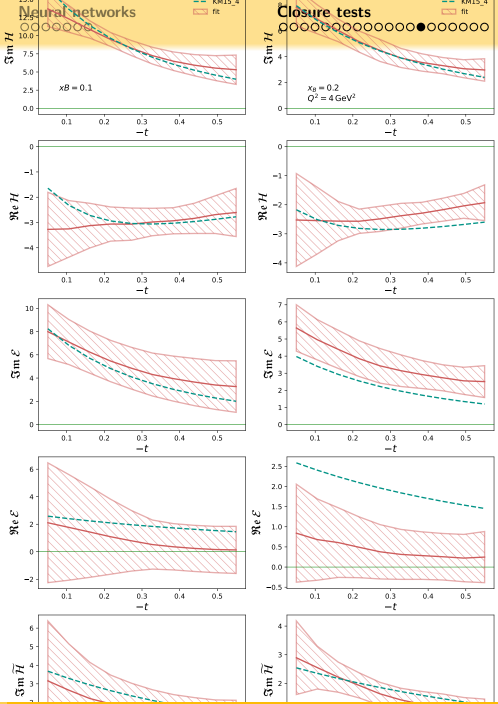
Example 3: Extrapolation? (3/3)

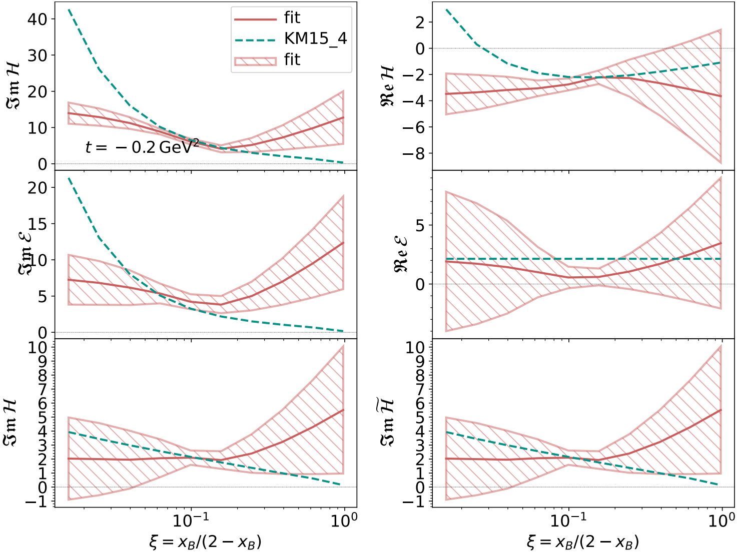


Example 4: Five CFFs (1/3)

- $\mathfrak{Im}\,\mathcal{H}$, $\mathfrak{Re}\,\mathcal{H}$, $\mathfrak{Im}\,\mathcal{E}$, $\mathfrak{Re}\,\mathcal{E}$, and $\mathfrak{Im}\,\widetilde{\mathcal{H}}$, from $X_{\rm UU}$, $X_{\rm LU}$, $X_{\rm UL}$, $A_{\rm C}$, and $A_{UT,DVCS}$

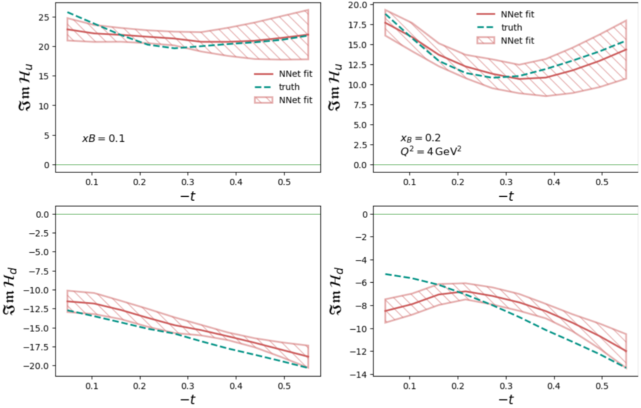



KM15_4
fit
$xB = 0.1$
$x_B = 0.2$
$Q^2 = 4\,\mathrm{GeV}^2$
$\Im\mathfrak{m}\,\mathcal{H}$
$\Re\mathfrak{e}\,\mathcal{H}$
$\Im\mathfrak{m}\,\mathcal{E}$
$\Re\mathfrak{e}\,\mathcal{E}$
$\Im\mathfrak{m}\,\widetilde{\mathcal{H}}$
$-t$

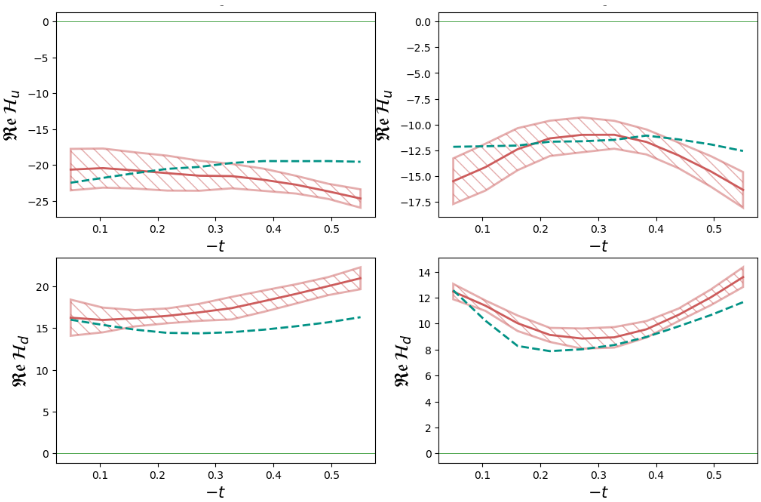

fit
KM15_4
fit
$t = -0.2\,\mathrm{GeV}^2$
$\Im\mathfrak{m}\,\mathcal{H}$
$\Re\mathfrak{e}\,\mathcal{H}$
$\Im\mathfrak{m}\,\mathcal{E}$
$\Re\mathfrak{e}\,\mathcal{E}$
$\Im\mathfrak{m}\,\mathcal{H}$
$\Im\mathfrak{m}\,\widetilde{\mathcal{H}}$
$\xi = x_B/(2 - x_B)$
$\xi = x_B/(2 - x_B)$

Example 5: $\mathcal{H}$ flavor separation (1/2)

- $\mathfrak{Im}\,\mathcal{H}_u$, $\mathfrak{Im}\,\mathcal{H}_d$, $\mathfrak{Re}\,\mathcal{H}_u$, $\mathfrak{Re}\,\mathcal{H}_d$, from $X_{\rm UU}$ and $X_{\rm LU}$ on proton and neutron
- Ground truth is a random smooth single neural net trained on subset of JLab proton and neutron data

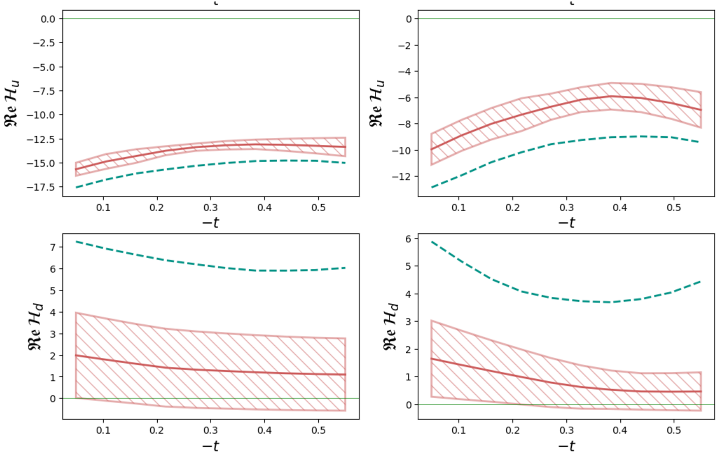


Example 5: $\mathcal{H}$ flavor separation (2/2)



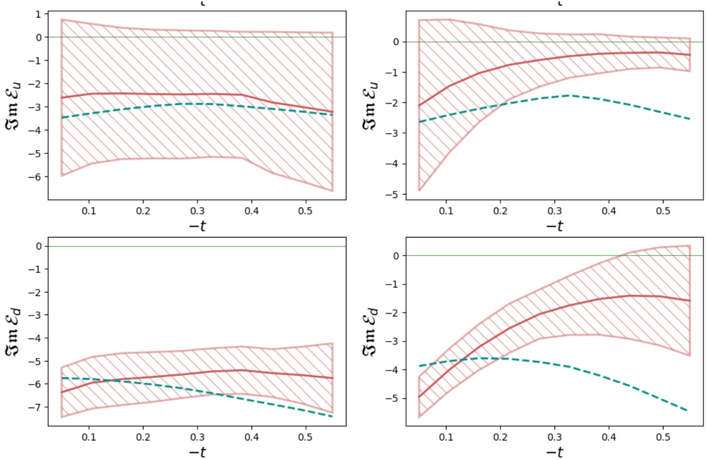
Example 6: $\mathcal{H}$ and $\mathcal{E}$ flavor separation

- from $X_{\rm UU}$, $X_{\rm LU}$ and $A_{\rm C}$ on proton and neutron: fails!

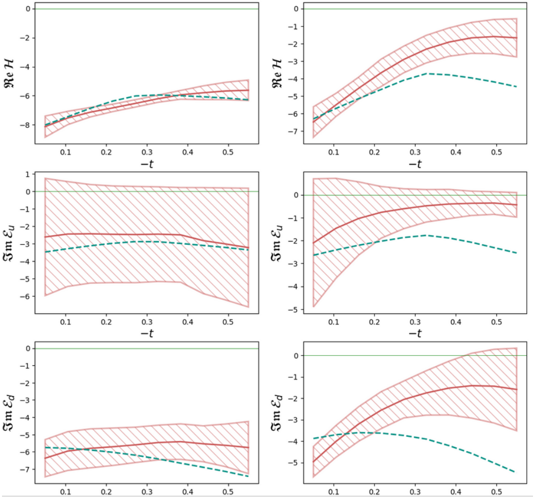


Example 7: $\mathfrak{Im}\,\mathcal{H}$ and $\mathfrak{Im}\,\mathcal{E}$ flavor separation (1/2)

- $\mathfrak{Im}\,\mathcal{H}_u$, $\mathfrak{Im}\,\mathcal{H}_d$, $\mathfrak{Im}\,\mathcal{E}_u$, $\mathfrak{Im}\,\mathcal{E}_d$, $\mathfrak{Re}\,\mathcal{H}$, and $\mathfrak{Re}\,\mathcal{H}$ from $X_{\rm UU}$, $X_{\rm LU}$, $A_{\rm UL}$ and $A_{\rm C}$ on proton and neutron

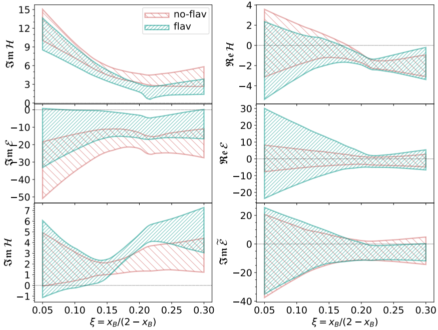


Example 7: $\Im\mathfrak{m}\,\mathcal{H}$ and $\Im\mathfrak{m}\,\mathcal{E}$ flavor separation (2/2)

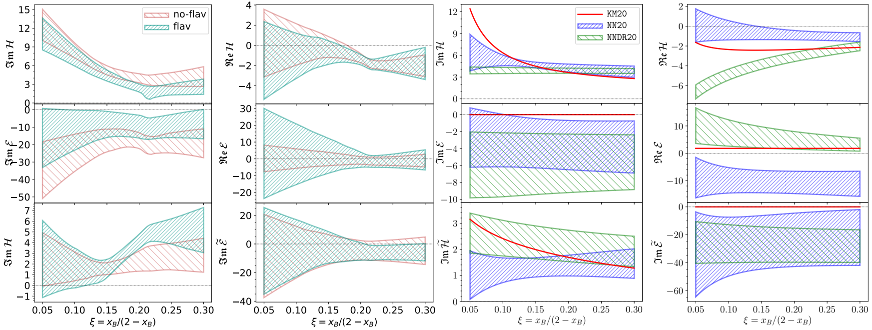


Applications

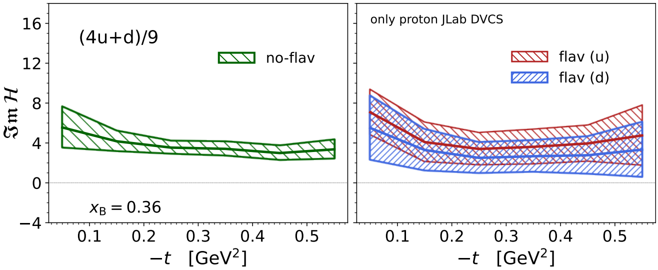
Real JLab data: flavored vs unflavored CFFs



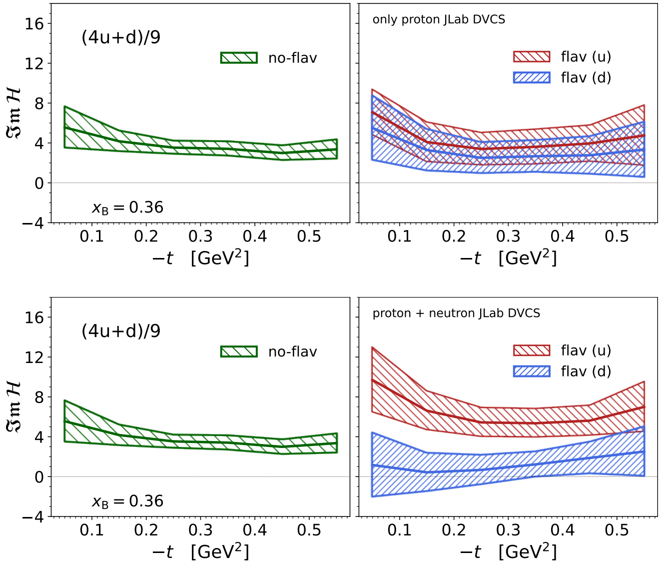
Real JLab data: flavored vs unflavored CFFs



Real JLab data: flavor separation



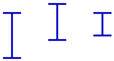
Real JLab data: flavor separation



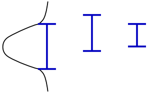
Thank you!

. . . and thanks to the Institute of Modern Physics, Lanzhou, China, where much of this work was done during June 2023

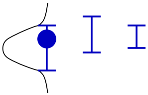
Monte Carlo propagation of errors



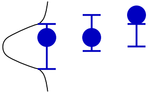
Monte Carlo propagation of errors



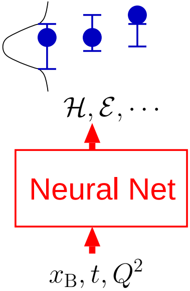
Monte Carlo propagation of errors



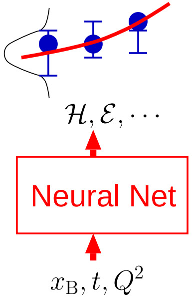
Monte Carlo propagation of errors



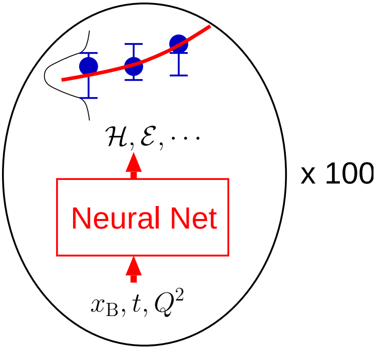
Monte Carlo propagation of errors



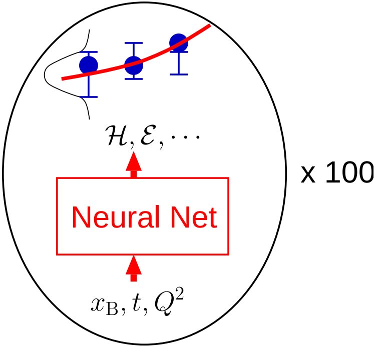
Monte Carlo propagation of errors



Monte Carlo propagation of errors



Monte Carlo propagation of errors



- Set of N_{rep} NNs defines a probability distribution in space of possible CFF functions:

$$\left\langle \mathcal{F}[\mathcal{H}] \right\rangle = \int \mathcal{D}\mathcal{H}\, \mathcal{P}[\mathcal{H}]\, \mathcal{F}[\mathcal{H}] = \frac{1}{N_{rep}} \sum_{k=1}^{N_{rep}} \mathcal{F}[\mathcal{H}^{(k)}] \,, \qquad (1)$$

- Experimental uncertainties and their correlations are preserved [Giele et al. '01]