Machine learning for QCD global analysis

CENTER *for* NUCLEAR FEMTOGRAPHY

Yaohang Li

Old Dominion University
Research: Machine learning, Monte Carlo methods

Nobuo Sato

Jefferson Lab Theory Center
Research: QCD global analysis (JAM) of hadron structure and hadronization

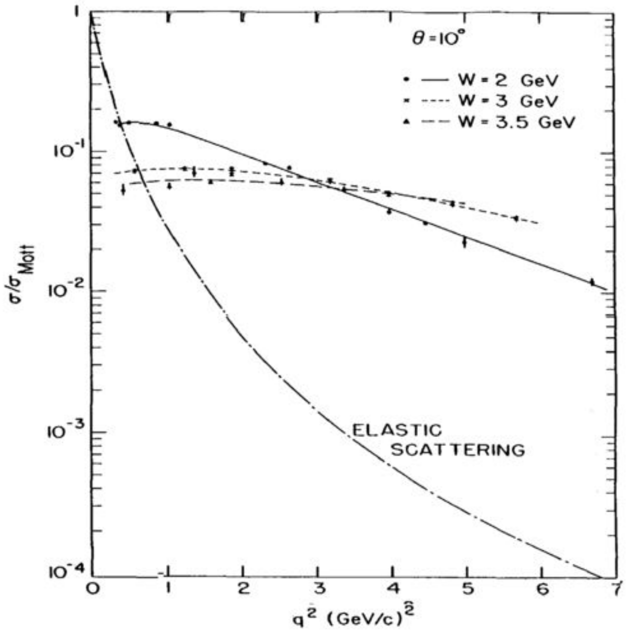
Outline

PART I: (N. Sato)

- Motivations
- Bayesian inference
- Examples
- Why machine learning?

PART II: (Y. Li)

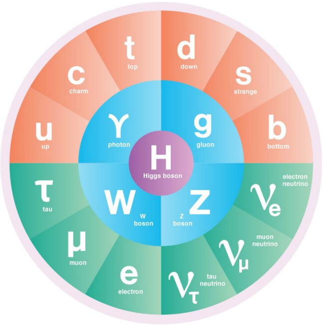
- End-to-end inverse problems
- VAIM Architecture
- Results in Toy Problems
- Results in DIS



Discovery of point-like particles inside proton

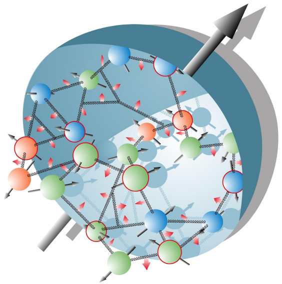
Motivations

Understanding the emergent phenomena of QCD

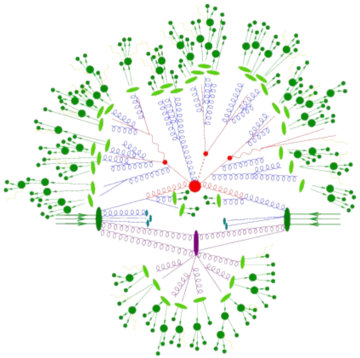


parts

"In philosophy, systems theory, science, and art, emergence occurs when an ***entity is observed*** *to have properties* ***its parts*** *do not have on their own, properties or behaviors which emerge only when the parts interact in a wider whole."* Wiki



Hadron Structure



Hadron formation

Observed entity

What do we mean by “hadron structure” ? (1D)

$$\xi = \frac{k^+}{P^+}$$

Parton momentum fraction relative to parent hadron

$$f_i(\xi) = \int \frac{\mathrm{d}w^-}{4\pi} e^{-i\xi p^+ w^-} \boxed{\langle N|\bar{\psi}_i(0, w^-, \mathbf{0}_\mathrm{T})\gamma^+\psi_i(0)|N\rangle}$$

parton distribution function (PDF)

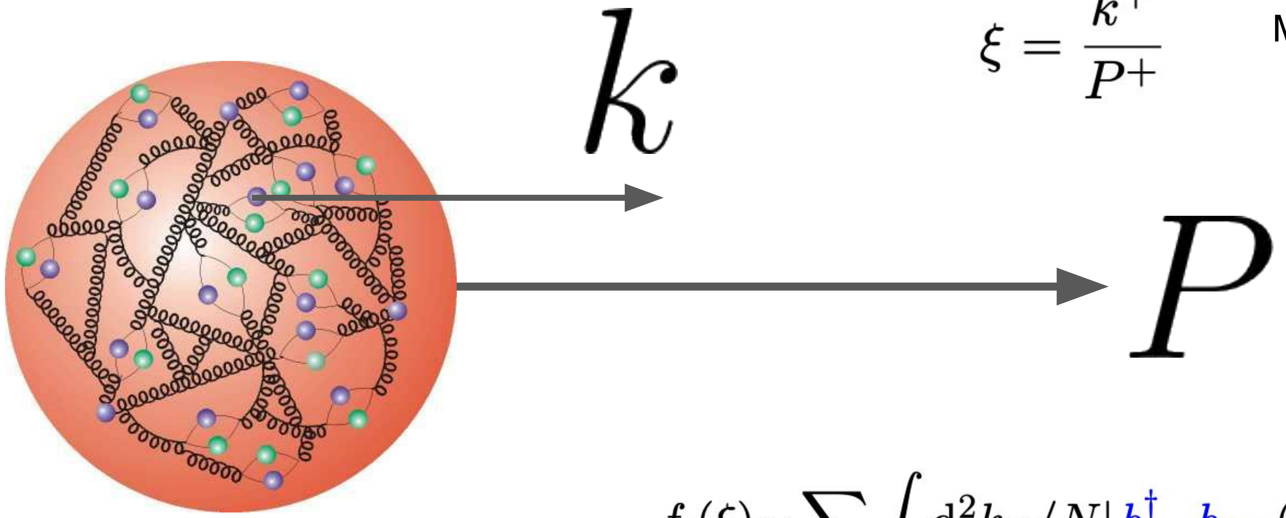
Interpretation in non-interacting QCD

$$\psi_i(x) = \sum_{k,\alpha} b_{k,\alpha}(x^+) u_{k,\alpha} e^{-ik^+x^- + ik_\mathrm{T}\cdot x_\mathrm{T}} + d^\dagger_{k,\alpha}(x^+) u_{k,-\alpha} e^{ik^+x^- - ik_\mathrm{T}\cdot x_\mathrm{T}}$$

$$f_i(\xi) \sim \sum_\alpha \int \mathrm{d}^2 k_\mathrm{T} \, \langle N| \underbrace{b^\dagger_{k,\alpha} b_{k,\alpha}(\xi p^+, k_\mathrm{T}, \alpha)}_{\text{number operator}} |N\rangle$$

How quarks and gluons are distributed?

$$\xi = \frac{k^+}{P^+}$$ Momentum fraction



$$f_i(\xi) \sim \sum_\alpha \int \mathrm{d}^2 k_\mathrm{T} \, \langle N | \underbrace{b^\dagger_{k,\alpha} b_{k,\alpha}(\xi p^+, k_\mathrm{T}, \alpha)}_{\text{number operator}} | N \rangle$$

What do we mean by "hadronization" ? (1D)

$$\zeta = \frac{p_h^+}{k^+}$$

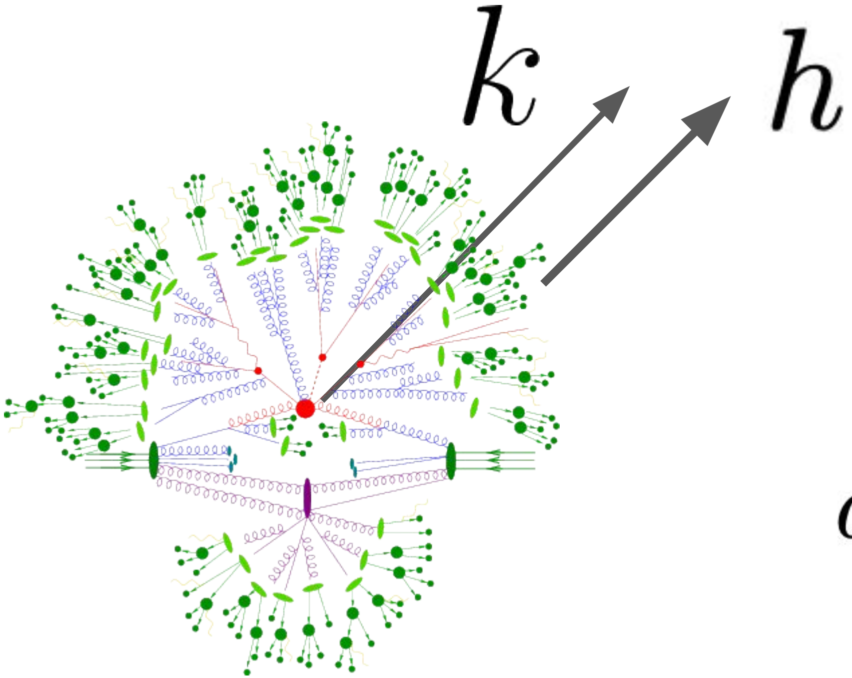
hadron momentum fraction relative to parent parton

$$d_{h/j}(\zeta) = \frac{\mathrm{Tr}_{\text{color,Dirac}}}{4N_{c,j}} \sum_X \zeta \int \frac{dw^+}{2\pi} e^{i(p_h^-/\zeta)w^+}$$
$$\times \gamma^- \langle 0 | \bar{\psi}_j(0, w^+, \mathbf{0}_\mathrm{T}) \, |p_h, X\rangle\langle p_h, X| \, \psi_j(0) | 0 \rangle$$

Fragmentation functions (FFs)

X = all states except detected hadron ***h***

How hadrons takes energy from quarks and gluons ?



$$\zeta = \frac{p_h^+}{k^+}$$

$$d_{h/j}(\zeta)$$

Number density of hadrons from parent parton

Hadron structure in interacting theory

Definition of PDFs in field theory requires renormalization

PDFs will depend on renormalization scale and its RGEs are the famous DGLAP equations

UV singularity when the field separation is zero

$$f_i(\xi) \stackrel{!}{=} \int \frac{\mathrm{d}w^-}{4\pi} e^{-i\xi p^+ w^-} \langle N|\bar{\psi}_i(0, w^-, \mathbf{0}_\mathrm{T})\gamma^+\psi_i(0)|N\rangle$$

Renormalization

$$f = Z_F \otimes f_{\text{bare}}$$

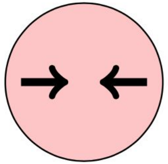
$$f(\xi) \to f(\xi, \mu)$$

Dokshitzer–**G**ribov–**L**ipatov–**A**ltarelli–**P**arisi

$$\frac{\mathrm{d} f_i(\xi,\mu^2)}{\mathrm{d}\ln\mu^2} = \sum_j \int_\xi^1 \frac{\mathrm{d}y}{y} P_{ij}(\xi,\mu^2)\, f_j\left(\frac{y}{\xi},\mu^2\right)$$

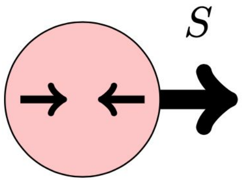
aka **DGLAP**

Spin structures



$$f = f_{\rightarrow} + f_{\leftarrow}$$

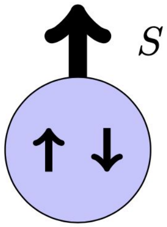
$$\langle N|\bar{\psi}_i(0, w^-, \mathbf{0}_\mathrm{T})\gamma^+\psi_i(0)|N\rangle$$



$$\Delta f = f_{\rightarrow} - f_{\leftarrow}$$

Helicity distribution

$$\langle N|\bar{\psi}_i(0, w^-, \mathbf{0}_\mathrm{T})\gamma^+\gamma_5\psi_i(0)|N\rangle$$

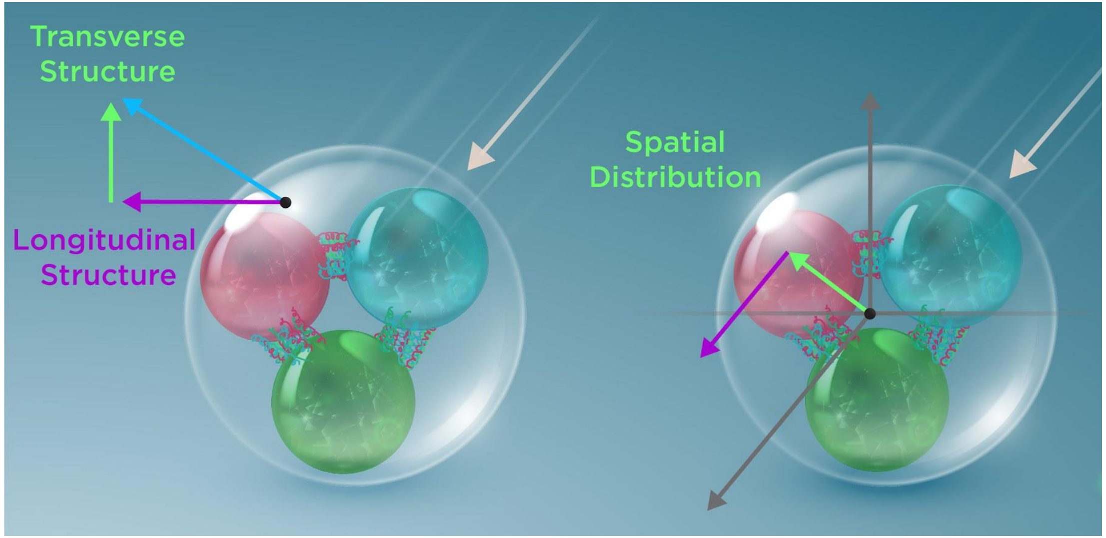


$$\delta_\mathrm{T} f = f_{\uparrow} - f_{\downarrow}$$

Transversity

$$\langle N|\bar{\psi}_i(0, w^-, \mathbf{0}_\mathrm{T})\gamma^+\gamma_\perp\gamma_5\psi_i(0)|N\rangle$$

Extensions to 3D



$$f(\xi)$$

PDFs

$$f(\xi, k_{\mathrm{T}})$$

Transverse momentum distribution -> **TMDs**

$$f(\xi, b_{\mathrm{T}})$$

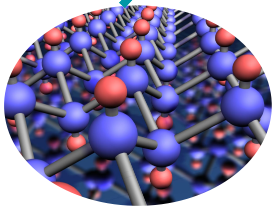
Impact parameter distribution -> **GPDs**

So how do we get hadron structure from experimental data?

What part of this is the
"**internal structure**"?

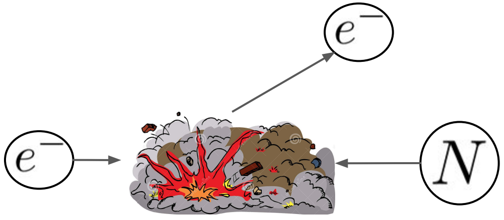
Factorization

Want to see
internal structure



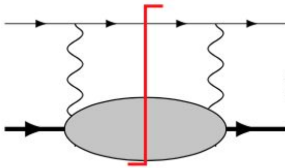
But we only see **debris**

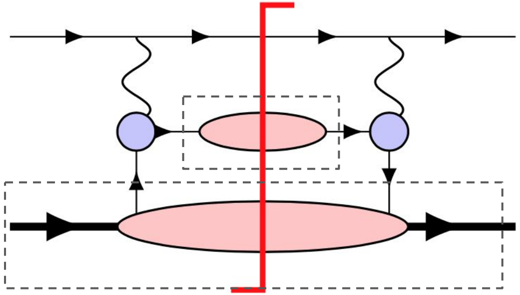
Factorization in deep-inelastic scattering (DIS)

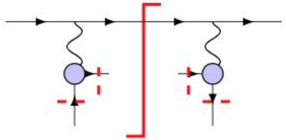


$$E' \frac{d\sigma}{d^3 l'}$$

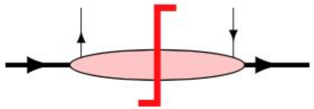
Approximations







Collision dependent factor



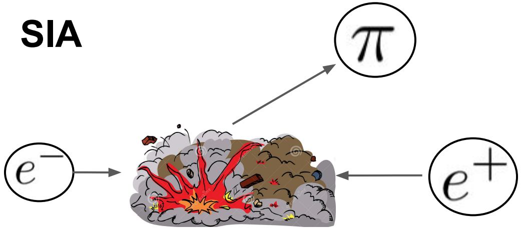
Internal structure

$$\sum_i \int_x^1 \frac{d\xi}{\xi} H_i(\xi) f_i\left(\frac{x}{\xi}\right) + \mathcal{O}\left(\frac{m^2}{Q^2}\right)$$

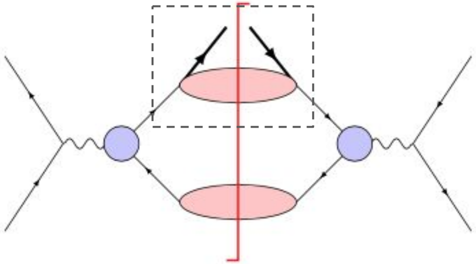
Error of approximations

Factorization in other reactions

SIA

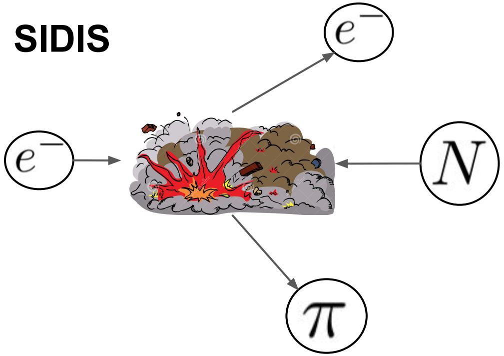


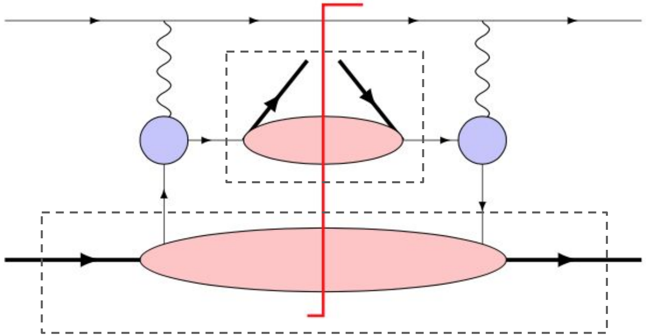
hadronization



$$d\sigma = \sum_i H_i^{\mathrm{SIA}} \otimes d_i$$

SIDIS



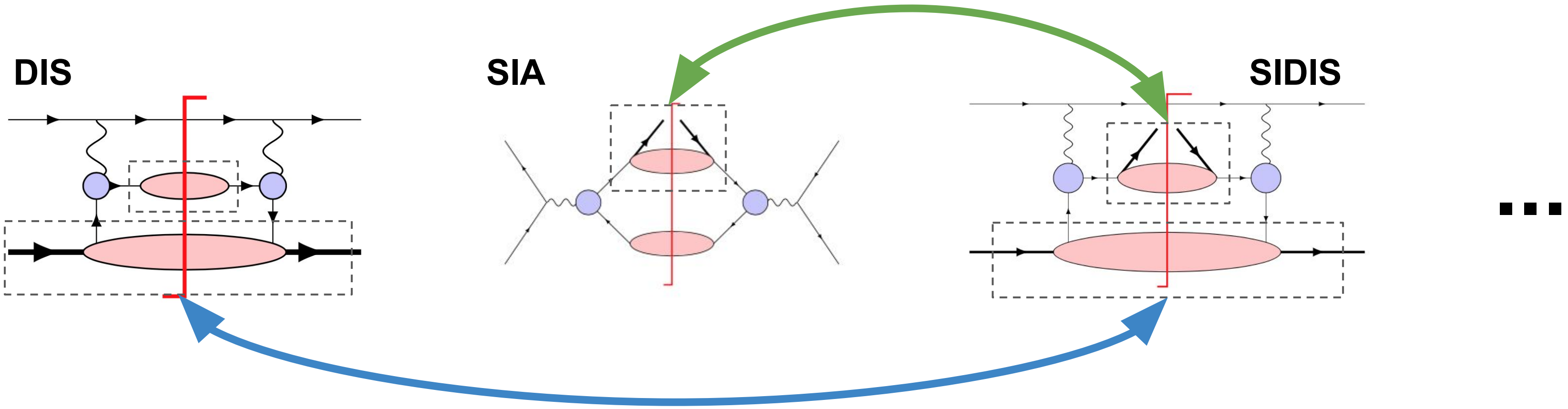


structure + hadronization

$$d\sigma = \sum_{ij} H_{ij}^{\mathrm{SIDIS}} \otimes f_i \otimes d_j$$

..and many more

Universality

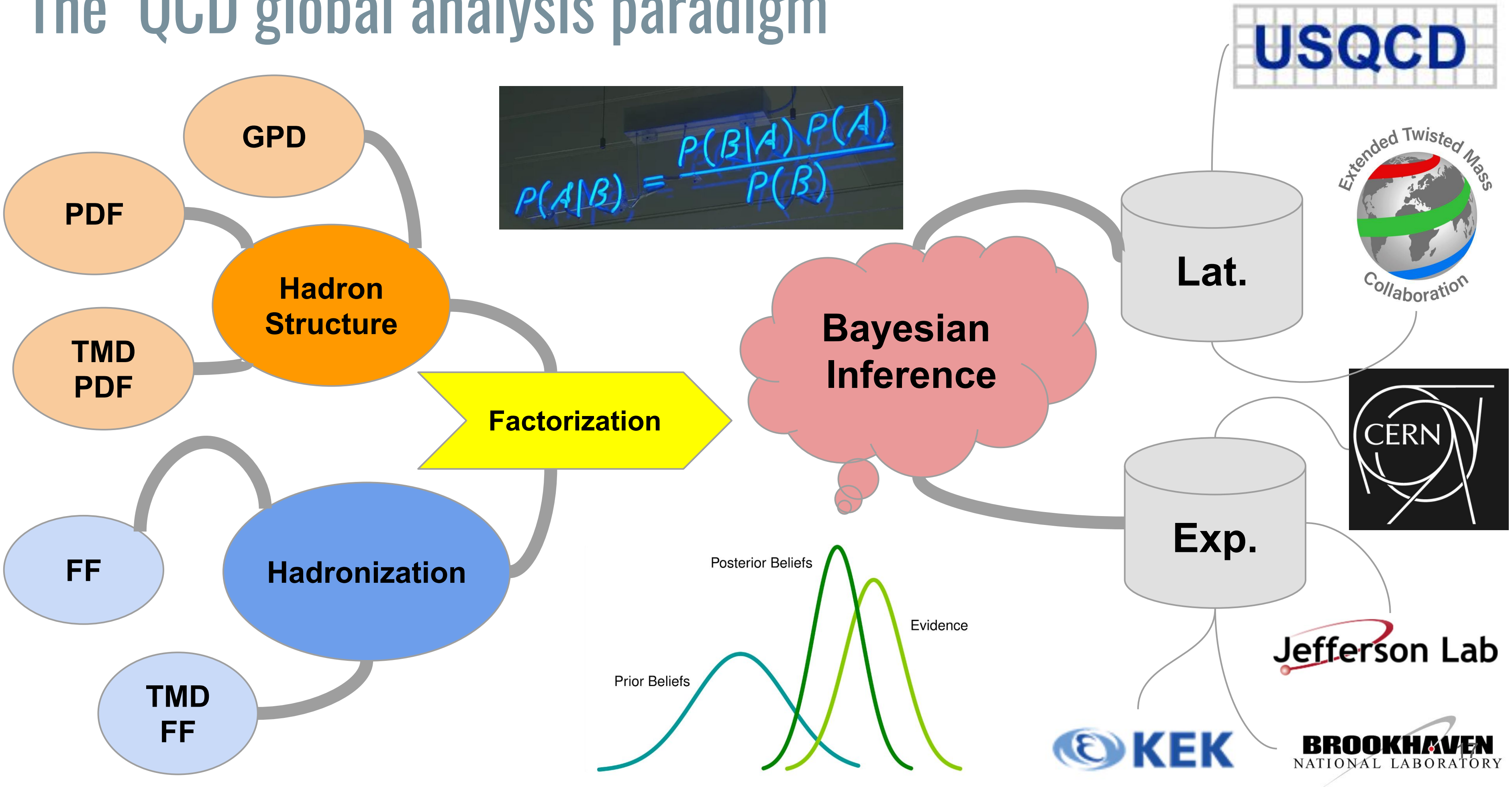


cross sections described by **universal non-perturbative** functions, e.g. PDFs, FFs

Scientific method

Question	Hadron structure
Hypothesis	Factorization theorems
Prediction	Theoretical observables
Experiment	JLab, EIC, LHC, RHIC, (lattice)
Analysis	Global Analysis

The QCD global analysis paradigm



The Bayesian inference

Experiments = theory + errors

$$d\sigma_{\mathrm{DIS}} = \sum_i H_i^{\mathrm{DIS}} \otimes f_i$$

$$d\sigma_{\mathrm{DY}} = \sum_{ij} H_{ij}^{\mathrm{DY}} \otimes f_i \otimes f_j$$

$$d\sigma_{\mathrm{SIA}} = \sum_i H_i^{\mathrm{SIA}} \otimes d_i$$

$$d\sigma_{\mathrm{SIDIS}} = \sum_{ij} H_{ij}^{\mathrm{SIDIS}} \otimes f_i \otimes d_j$$

Hadron Structure

Hadronization

RGE boundary conditions

$$f_i(\xi, \mu_0^2) = N_i \xi^{a_i}(1-\xi)^{b_i}(1+\ldots)$$

$$d_i(\zeta, \mu_0^2) = N_i \zeta^{a_i}(1-\zeta)^{b_i}(1+\ldots)$$

$$\mathbf{a} = (N_i, a_i, b_i, \ldots)$$

Posterior distribution

Prior distribution

$$\rho(\mathbf{a}|\mathrm{data}) \sim \mathcal{L}(\mathbf{a}, \mathrm{data})\pi(\mathbf{a})$$

Likelihood

$$\mathcal{L}(\boldsymbol{a}, \mathrm{data}) = \exp\left[-\frac{1}{2}\chi^2(\boldsymbol{a}, \mathrm{data})\right]$$

$$\mathrm{E}[f_i(\xi, \mu^2)] = \int d^n\mathbf{a}\, \rho(\mathbf{a}|\mathrm{data}) f_i(\xi, \mu^2; \mathbf{a})$$

$$\mathrm{V}[f_i(\xi, \mu^2)] = \int d^n\mathbf{a}\, \rho(\mathbf{a}|\mathrm{data}) \left[f_i(\xi, \mu^2; \mathbf{a}) - \mathrm{E}[f_i(\xi, \mu^2)]\right]^2$$

T. Bayes.

Bayesian inference

The Bayes theorem

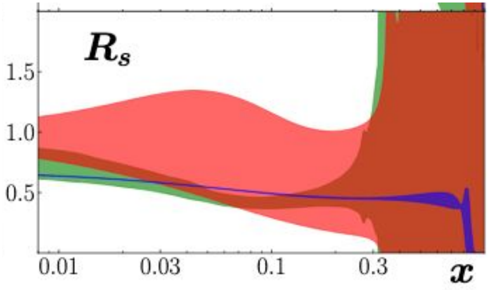
$$\mathcal{L}(\boldsymbol{a}, \mathrm{data}) = \exp\left[-\frac{1}{2}\chi^2(\boldsymbol{a}, \mathrm{data})\right]$$

This is a choice

Min, Max, penalties, regulators etc

$$\rho(\boldsymbol{a}|\mathrm{data}) \sim \mathcal{L}(\boldsymbol{a}, \mathrm{data})\pi(\boldsymbol{a})$$

This is impractical



$$\mathrm{E}[\mathcal{O}] = \int d^n a \ \rho(\boldsymbol{a}|\mathrm{data}) \ \mathcal{O}(\boldsymbol{a})$$

$$\mathrm{V}[\mathcal{O}] = \int d^n a \ \rho(\boldsymbol{a}|\mathrm{data}) \ \left[\mathcal{O}(\boldsymbol{a}) - \mathrm{E}[\mathcal{O}]\right]^2$$

How do we deal with the curse of dimensionality ?

$$\mathrm{E}[\mathcal{O}] = \int d^n a \; \rho(\boldsymbol{a}|\mathrm{data}) \; \mathcal{O}(\boldsymbol{a})$$

$$\mathrm{V}[\mathcal{O}] = \int d^n a \; \rho(\boldsymbol{a}|\mathrm{data}) \; \left[\mathcal{O}(\boldsymbol{a}) - \mathrm{E}[\mathcal{O}]\right]^2$$

Option 1: Maximum likelihood

Asummes symmetric likelihood, unique solution

$$\mathrm{E}[\mathcal{O}] \simeq \mathcal{O}(\boldsymbol{a}_0)$$

Assumes Gaussian behavior around ML

$$\mathrm{V}[\mathcal{O}] = \mathrm{Hessian}, \;\; \mathrm{Lagrange\ multipliers}$$

How do we deal with the curse of dimensionality ?

$$\mathrm{E}[\mathcal{O}] = \int d^n a\ \rho(\boldsymbol{a}|\mathrm{data})\ \mathcal{O}(\boldsymbol{a})$$

$$\mathrm{V}[\mathcal{O}] = \int d^n a\ \rho(\boldsymbol{a}|\mathrm{data})\ \left[\mathcal{O}(\boldsymbol{a}) - \mathrm{E}[\mathcal{O}]\right]^2$$

Option 2: MC approach

Build an MC ensemble ($$$)

$$\mathrm{E}[\mathcal{O}] \simeq \frac{1}{N}\sum_k \mathcal{O}(\boldsymbol{a}_k)$$

$$\mathrm{V}[\mathcal{O}] = \simeq \frac{1}{N}\sum_k \left[\mathcal{O}(\boldsymbol{a}_k) - \mathrm{E}[\mathcal{O}]\right]^2$$

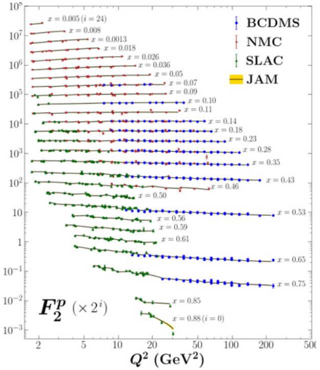
Many algorithms

- MCMC
- HMC
- Data resampling
- ...

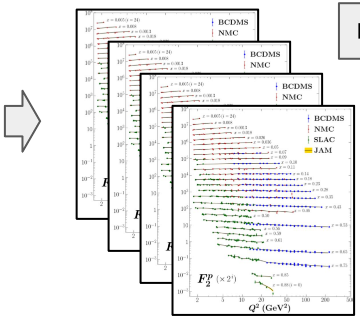
Data resampling

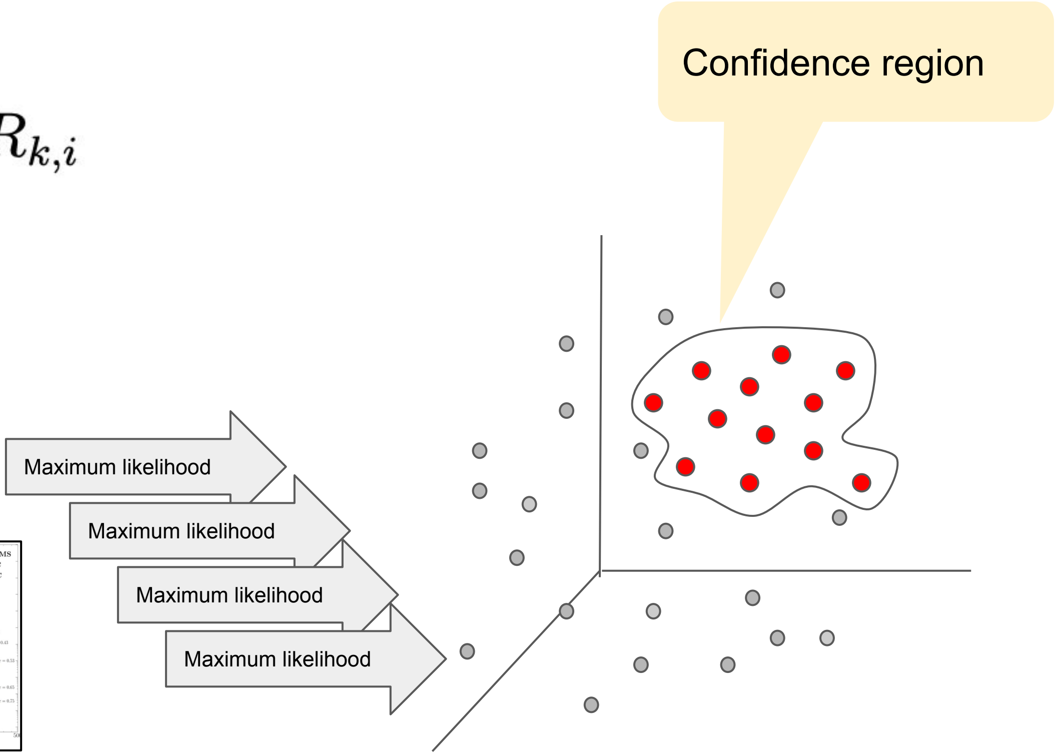
$$d_{k,i}^{(\mathrm{pseudo})} = d_i^{(\mathrm{original})} + \alpha_i \; R_{k,i}$$

Original data



Replica data





Parameter space

Jefferson Lab Angular Momentum Collaboration

Staff / Faculty

W. Melnitchouk (JLab), T. Rogers (ODU/JLab), A. Prokudin (PSU), D. Pitonyak (LVC), L. Gamberg (PSU), Z. Kang (UCLA) J. Qiu (JLab), A. Accardi (Hampton/JLab), A. Metz (Temple), C.-R. Ji (NCSU), M. Constantinou (Temple), F. Steffens (Bonn), M. White (Adelaide) , ...

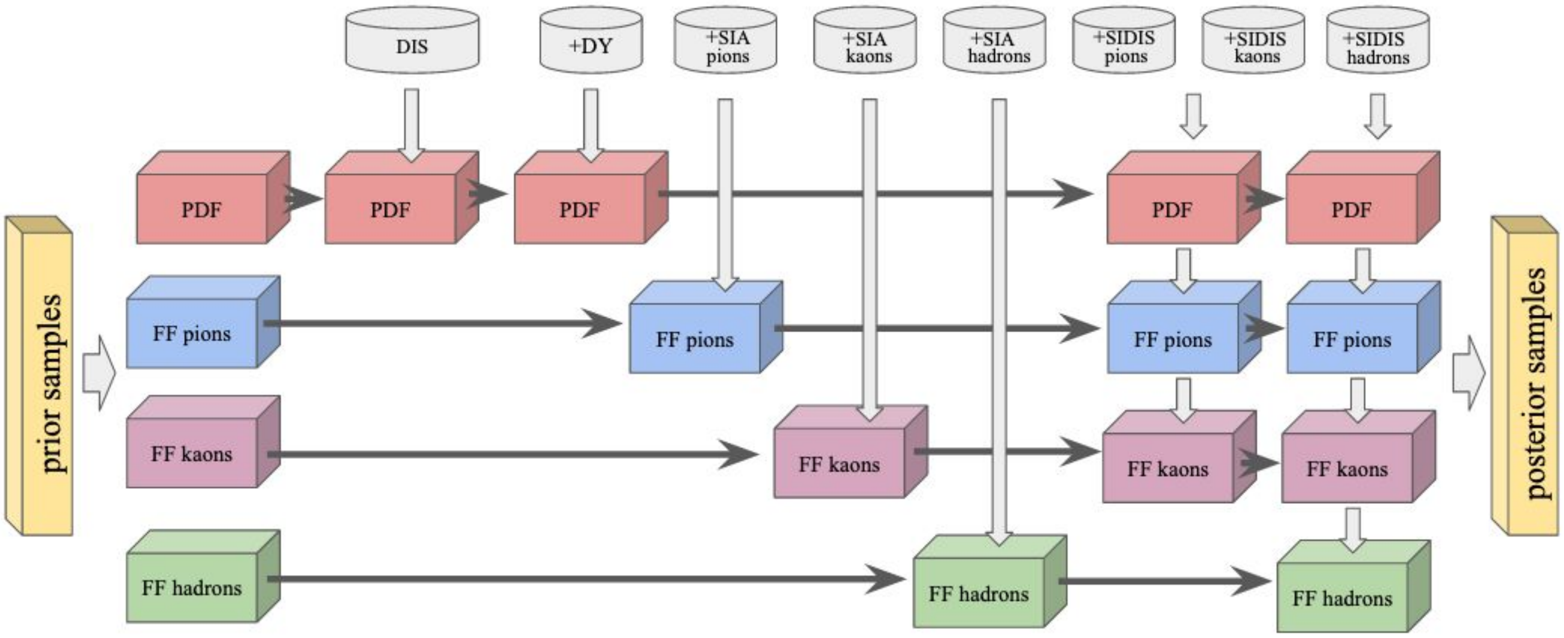
Students / Postdocs

C. Cocuzza (Temple), Y. Zhou (W&M), P. Barry (NCSU), E. Moffat (ODU), J. Bringewatt (UMD), J. Ethier (Nikhef), C. Andres (JLab), F. Delcarro (JLab), A. Hiller-Blin (JLab), Z. Searle (Adelaide)

Examples

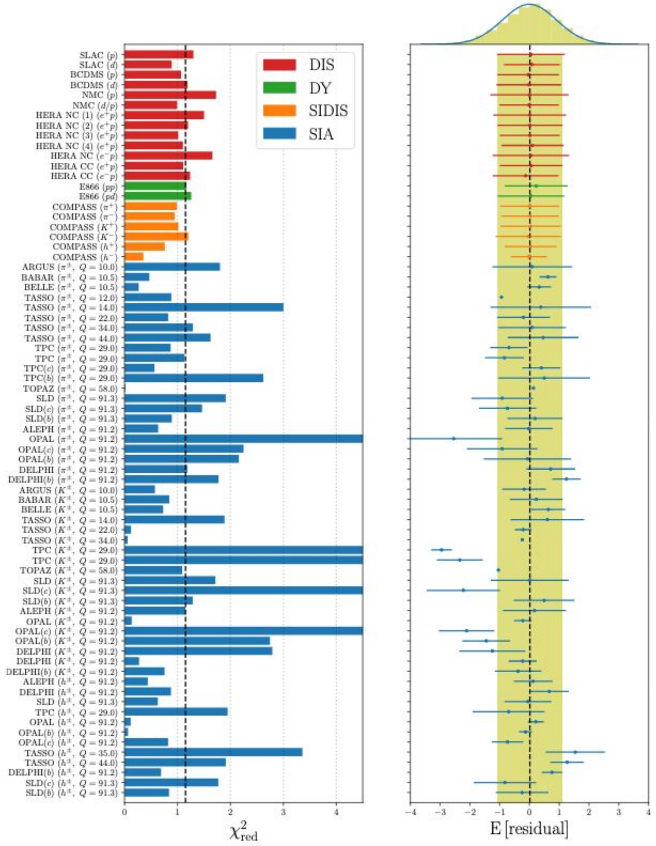
JAM1D: PDFs + FFs

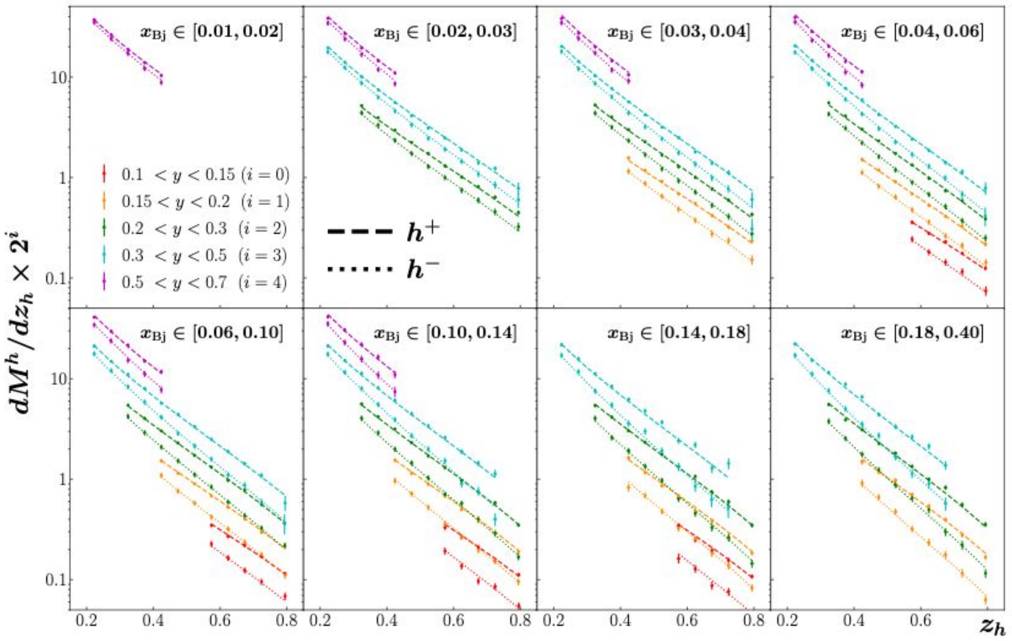
Moffat, Melnitchouk, Rogers, NS
arXiv:2101.04664



JAM1D: PDFs + FFs

Moffat, Melnitchouk, Rogers, NS
arXiv:2101.04664

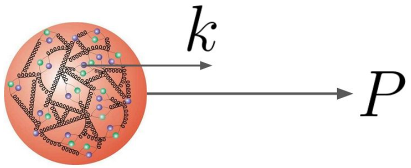


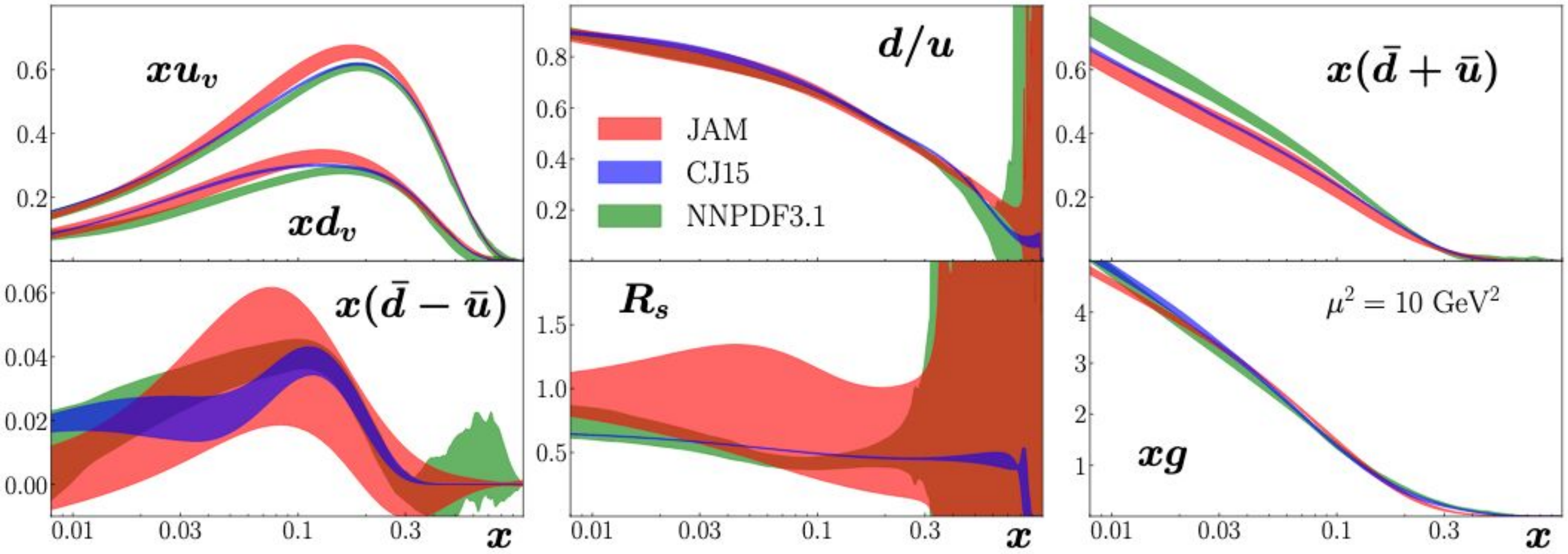


JAM1D: PDFs + FFs

Moffat, Melnitchouk, Rogers, NS
arXiv:2101.04664

PDFs

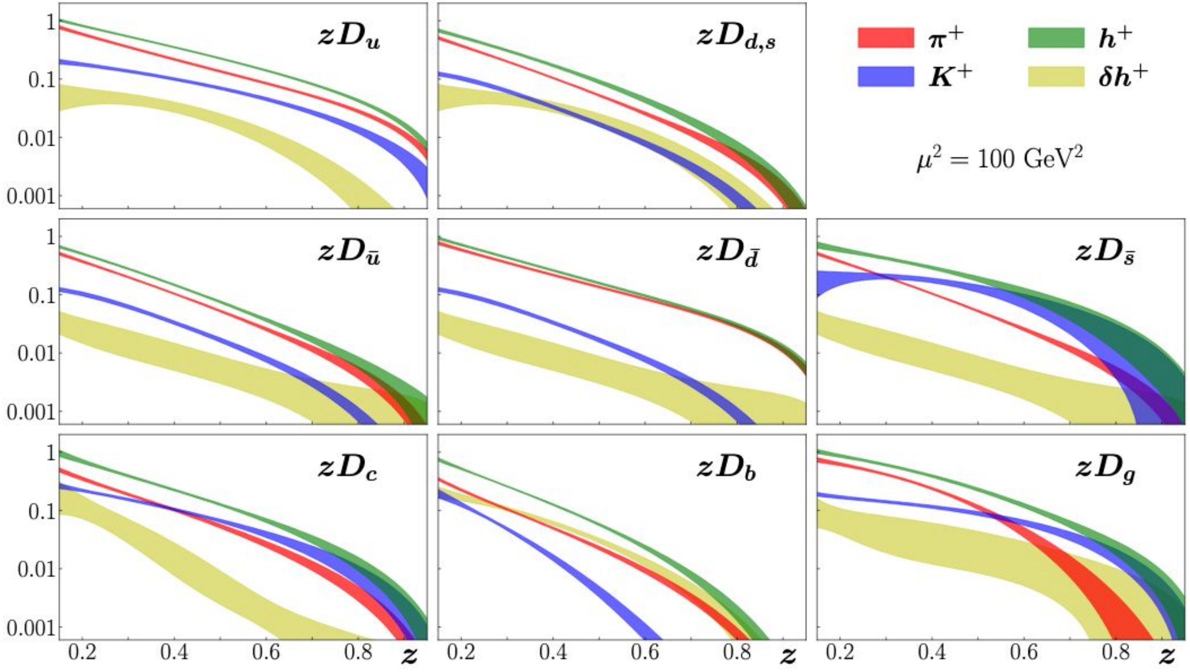


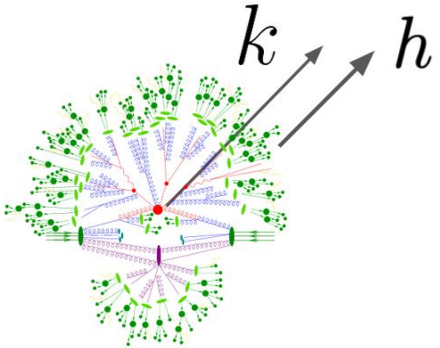


JAM1D: PDFs + FFs

Moffat, Melnitchouk, Rogers, NS
arXiv:2101.04664

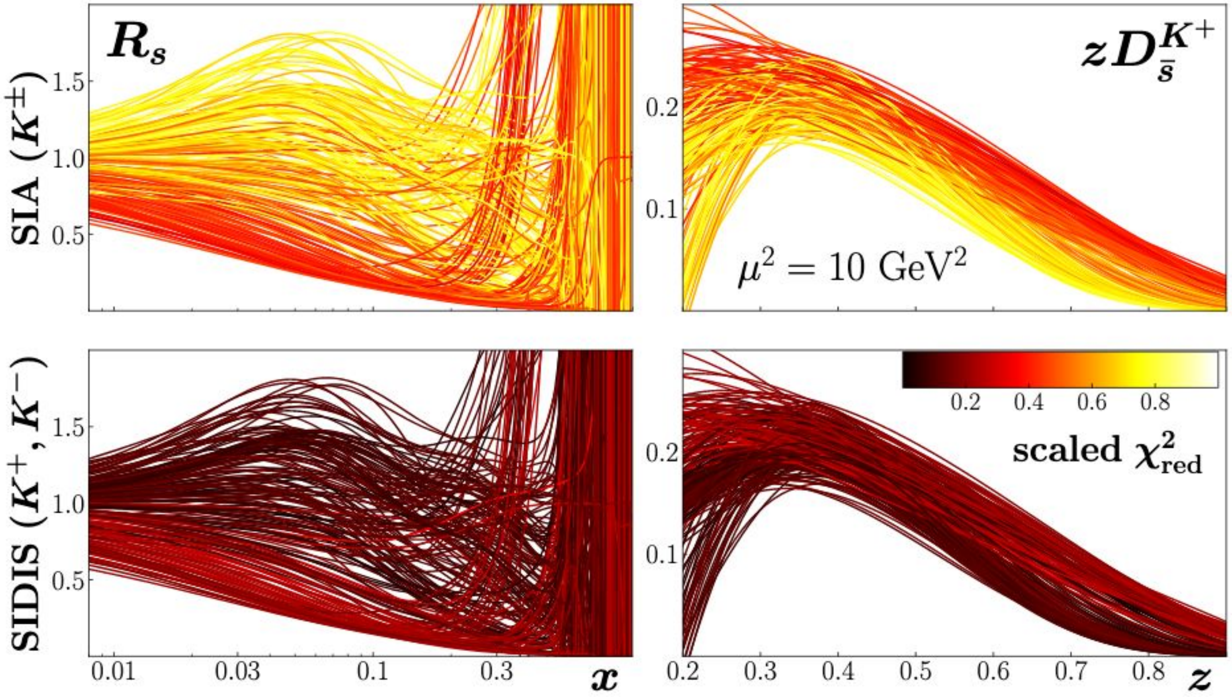
FFs





JAM1D: PDFs + FFs

Moffat, Melnitchouk, Rogers, NS
arXiv:2101.04664

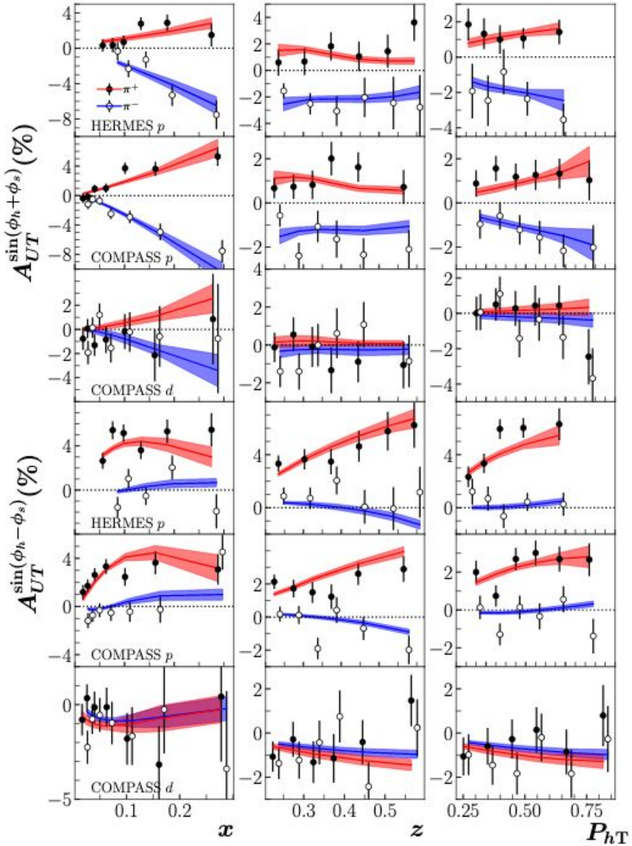


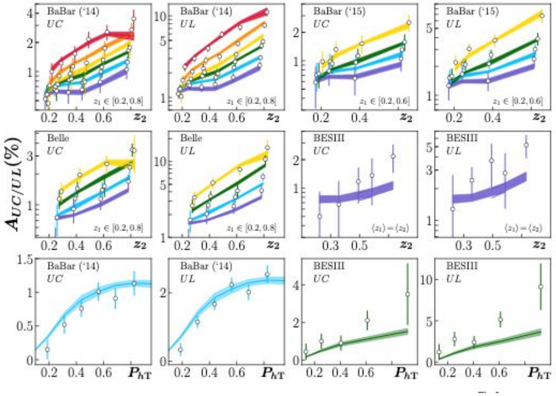
$$R_s = \frac{s + \bar{s}}{\bar{u} + \bar{d}}$$

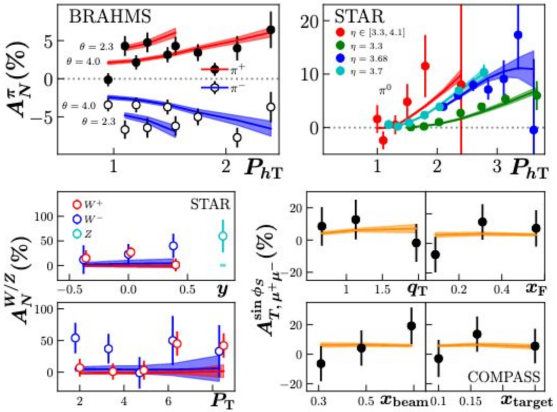
The simultaneous fit of PDFs and FFs provides new insights on nucleon strangeness

JAM3D: TMDs +CT3

Cammarota, Gamberg, Kang, Miller, Pitonyak, Prokudin, Rogers, NS

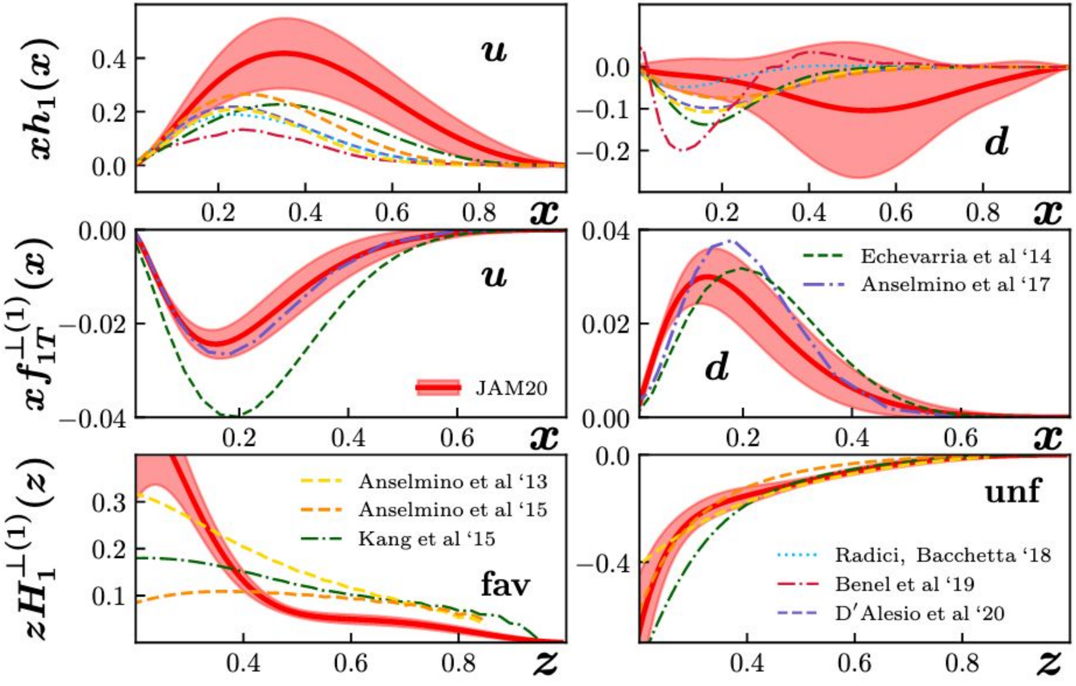




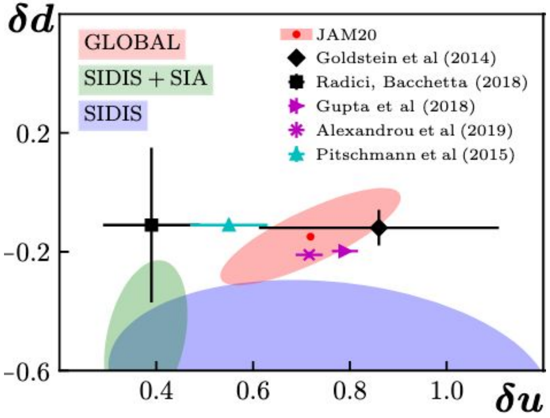


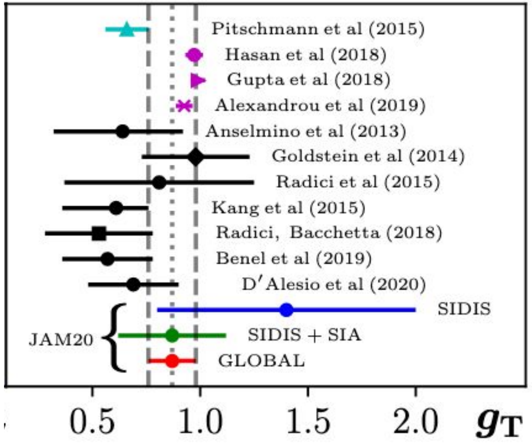
Observable	Reactions
$A^{\text{Siv}}_{\text{SIDIS}}$	$e+(p,d)^{\uparrow} \to e+(\pi^+,\pi^-,\pi^0)+X$
$A^{\text{Col}}_{\text{SIDIS}}$	$e+(p,d)^{\uparrow} \to e+(\pi^+,\pi^-,\pi^0)+X$
$A^{\text{Col}}_{\text{SIA}}$	$e^+ + e^- \to \pi^+\pi^-(UC,UL)+X$
$A^{\text{Siv}}_{\text{DY}}$	$\pi^- + p^{\uparrow} \to \mu^+\mu^- + X$
$A^{\text{Siv}}_{\text{DY}}$	$p^{\uparrow}+p \to (W^+,W^-,Z)+X$
A^h_N	$p^{\uparrow}+p \to (\pi^+,\pi^-,\pi^0)+X$

JAM3D: TMDs + CT3



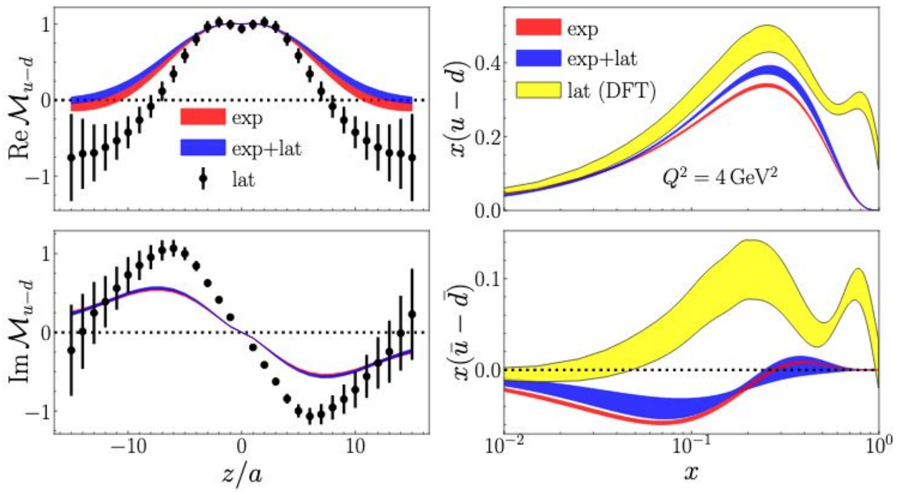
Cammarota, Gamberg, Kang, Miller, Pitonyak, Prokudin, Rogers, NS

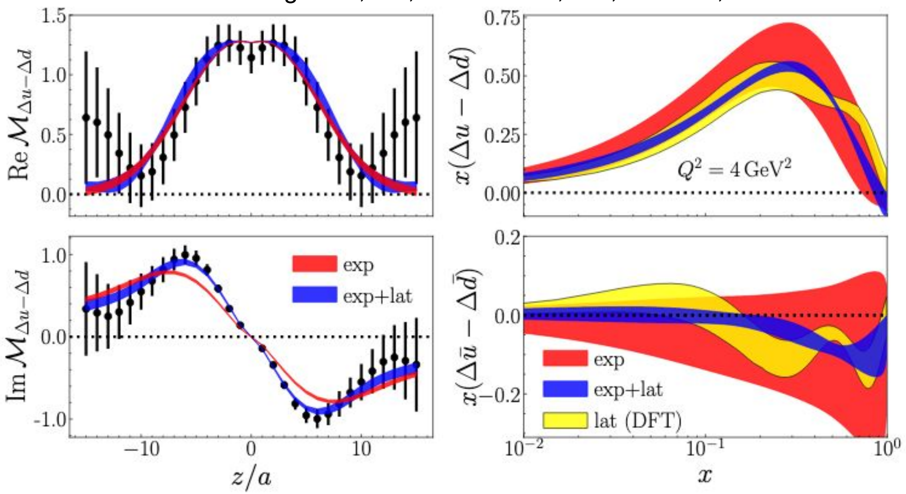




JAM1D: experimen + lattice

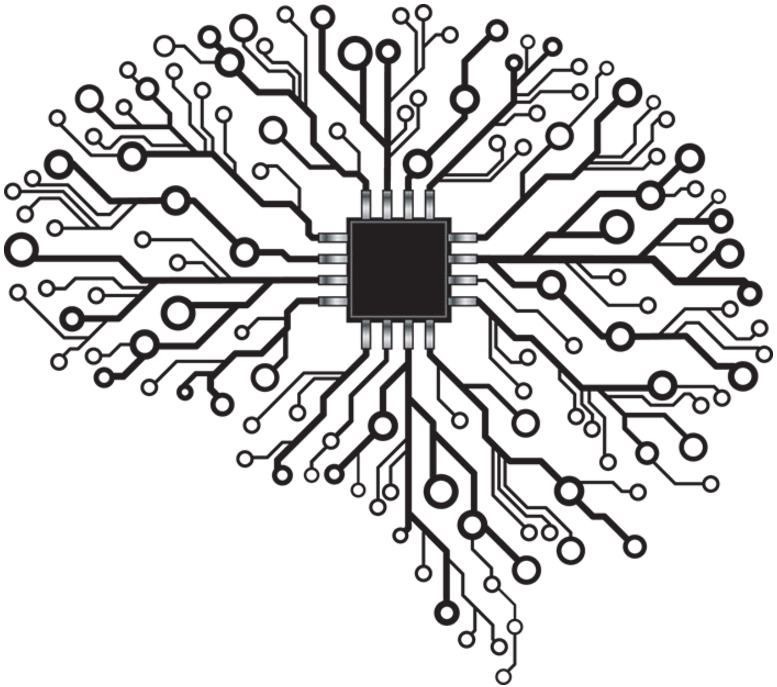
Bringewatt, NS, Melnitchouk, Qiu, Steffens, Constantinou





$$\mathcal{M}_q(z,\mu) = \int_{-\infty}^{\infty} dx\, e^{-ixP_3 z} \int_{-1}^{1} \frac{d\xi}{|\xi|} C_q\left(\frac{x}{\xi}, \frac{\mu}{\xi P_3}\right) f_q(\xi,\mu)$$

$$\mathcal{M}_{\Delta q}(z,\mu) = \int_{-\infty}^{\infty} dx\, e^{-ixP_3 z} \int_{-1}^{1} \frac{d\xi}{|\xi|} C_{\Delta q}\left(\frac{x}{\xi}, \frac{\mu}{\xi P_3}\right) \Delta f_q(\xi,\mu)$$



Why Machine Learning?

The inverse problem

Forward mapping

Parametrization d.o.f.

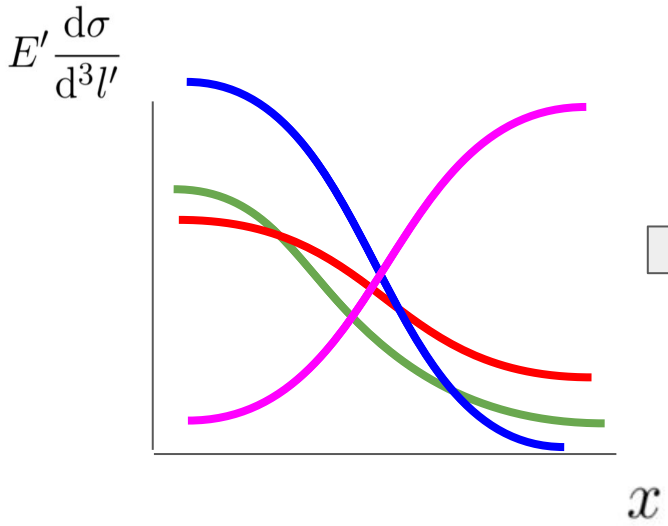
$$E' \frac{\mathrm{d}\sigma^{\mathbf{DIS}}}{\mathrm{d}^3 l'} = \sum_i \int_x^1 \frac{\mathrm{d}\xi}{\xi} H_i(\xi) f_i\left(\frac{x}{\xi}; \boldsymbol{a}\right)$$

Experimental observable

Backward mapping

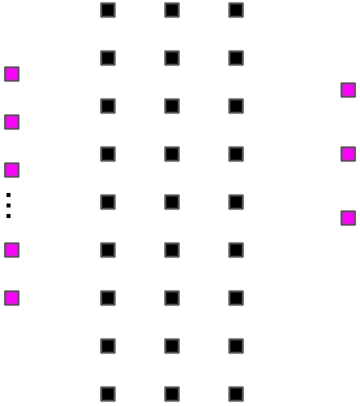
An idea: parametrize the inverse function

Observable space



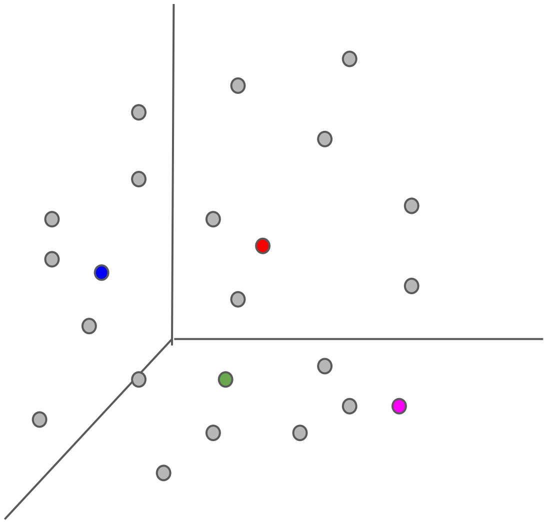
Theory

$$E' \frac{\mathrm{d}\sigma}{\mathrm{d}^3 l'} = \sum_i \int_x^1 \frac{\mathrm{d}\xi}{\xi} H_i(\xi) f_i\left(\frac{x}{\xi}; \boldsymbol{a}\right)$$



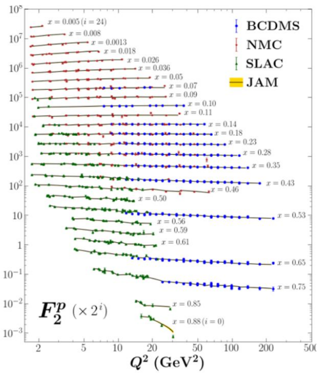
Neural Nets

Parameter space

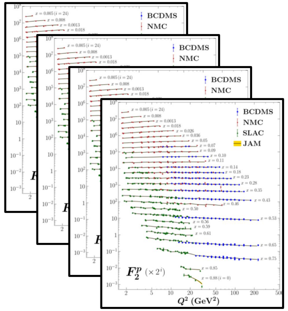


Parameter inference

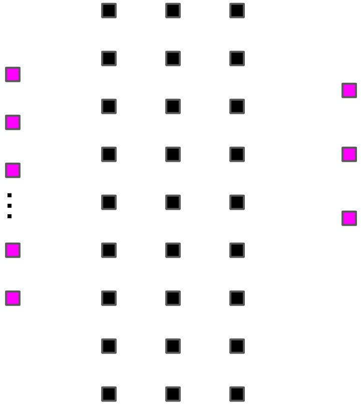
Original data



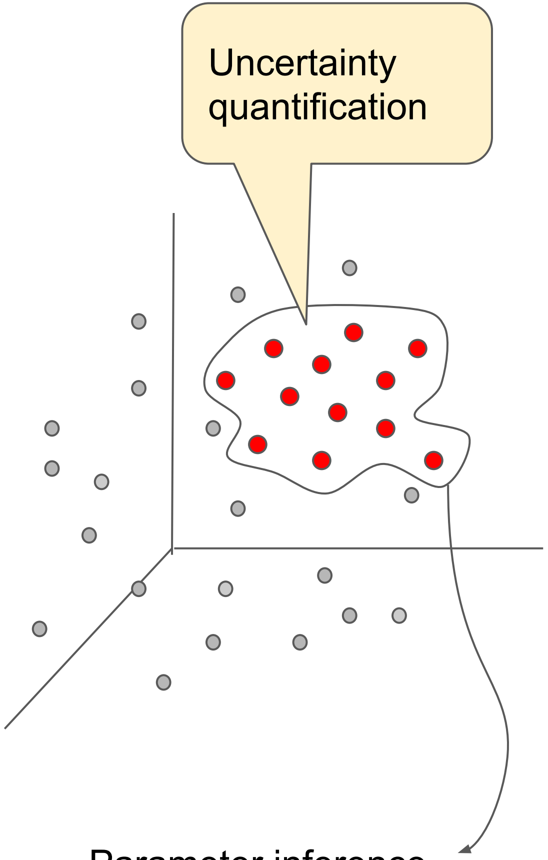
Replica data



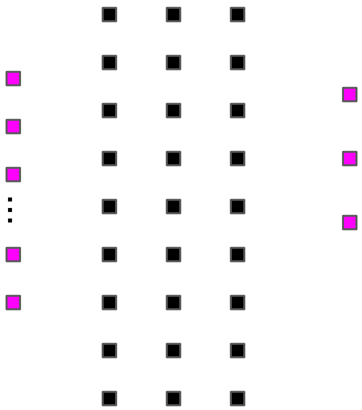
$$\rho(\mathbf{a}|\text{data}) \sim \mathcal{L}(\mathbf{a}, \text{data})\pi(\mathbf{a})$$



Trained inverse mapper



Parameter inference

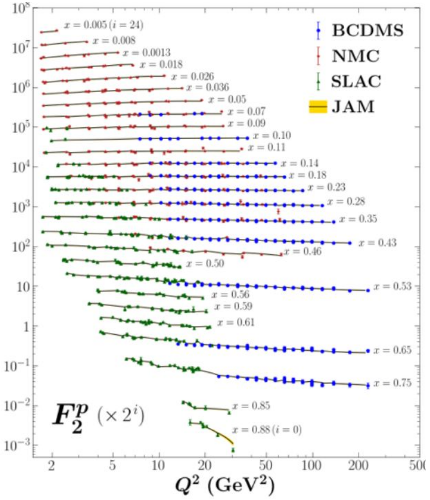


So why do we need **inverse mappers**?

1) Manipulate data input

Collecting MC samples is too expensive

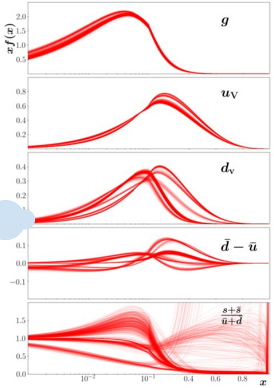
$$\rho(\mathbf{a}|\mathrm{data}) \sim \mathcal{L}(\mathbf{a}, \mathrm{data})\pi(\mathbf{a})$$



What happens if we remove … data ?

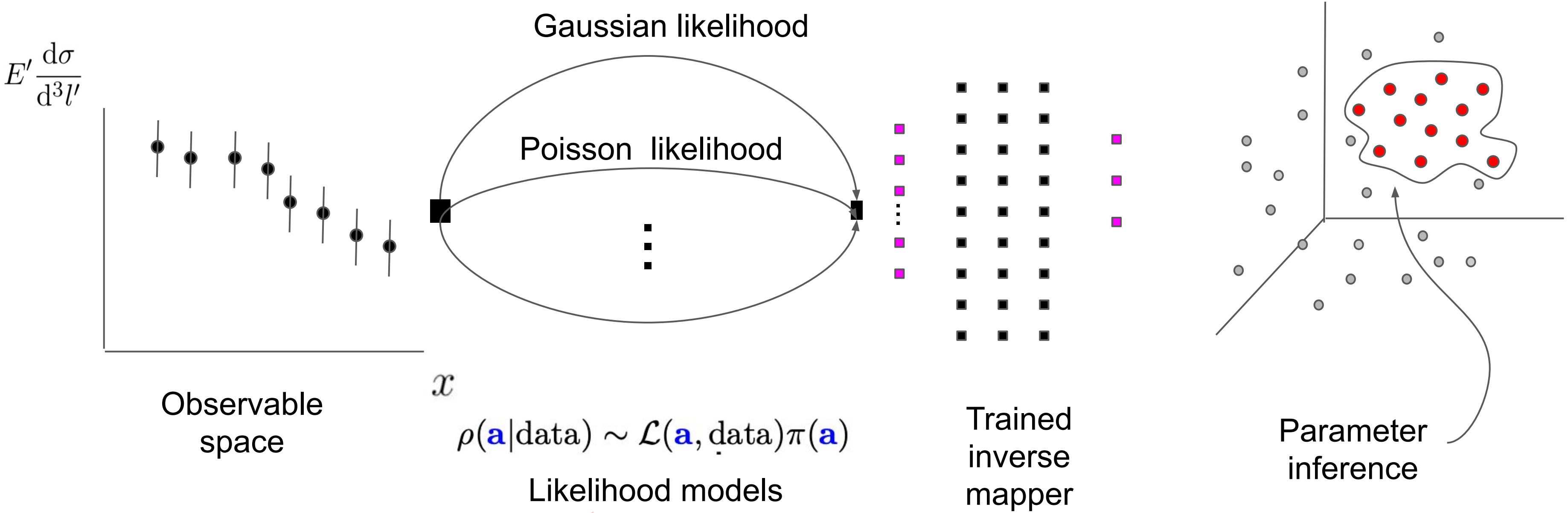
Where do we need more experiments?

“Global analysis is a kind of a sausage” … **how to unpack it**?



What data are forcing … to be …?

2) Bayesian inference modeling



Existing methodologies are prohibitively expensive for such studies

3) Change how we view global analysis efforts

LHAPDF 6.3.0

Main page | PDF sets | Class hierarchy | Functions | Examples | Mor

LHAPDF Documentation

Introduction

LHAPDF is a general purpose C++ interpolator, used for evaluating PDFs from discretised data files. Previous versions of **LHAPDF** were written in Fortran 77/90 and are documented at http://lhapdf.hepforge.org/lhapdf5/.

LHAPDF ID	Set name	Number of set members	Latest data version	Notes
251	GRVPI0	1	1	alpha_s was broken in LHAPDF5. This version uses approximate 1st order running from reported Lambda4,5 values.
252	GRVPI1	1	1	
270	xFitterPI_NLO_EIG	8	1	
280	xFitterPI_NLO_VAR	6	1	
10000	cteq6	41	-1	Corresponds to LHAPDF5's cteq6 or cteq6mE; central member equivalent to cteq6m.
10042	cteq6l1	1	4	
10150	cteq61	41	1	
10550	cteq66	45	1	
10770	CT09MCS	1	1	
10771	CT09MC1	1	1	
10772	CT09MC2	1	1	
10800	CT10	53	4	
10860	CT10as	11	2	
10900	CT10w	53	1	
10960	CT10was	11	2	
10980	CT10f3	1	1	
10981	CT10f4	1	1	
10982	CT10wf3	1	1	
10983	CT10wf4	1	1	
11000	CT10nlo	53	4	
11062	CT10nlo_as_0112	1	1	
11063	CT10nlo_as_0113	1	1	

Many groups around the world

- Global efforts consolidated in interpolation tables - static version of hadron structures
- Inverse mapper can be a new representation for hadron structure data - dynamical version
- Towards a unified quality control for hadron structure inference

Citation: P.A. Zyla *et al.* (Particle Data Group), Prog. Theor. Exp. Phys. **2020**, 083C01 (2020)

$\boxed{\Lambda_c(2595)^+}$ $\qquad I(J^P) = 0(\frac{1}{2}^-)$ Status: $\boxed{***}$

The $\Lambda_c^+\pi^+\pi^-$ mode is largely, and perhaps entirely, $\Sigma_c\pi$, which is just at threshold; since the Σ_c has $J^P = 1/2^+$, the J^P here is almost certainly $1/2^-$. This result is in accord with the theoretical expectation that this is the charm counterpart of the strange $\Lambda(1405)$.

PDG particle data group

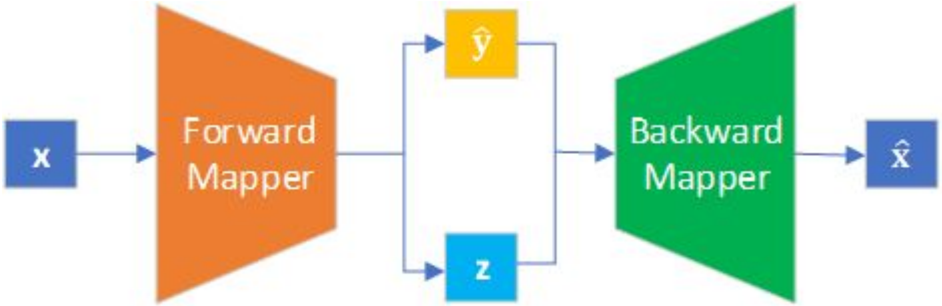
Outline

PART I: (N. Sato)

- Motivations
- Bayesian inference
- Examples
- Machine learning

PART II: (Y. Li)

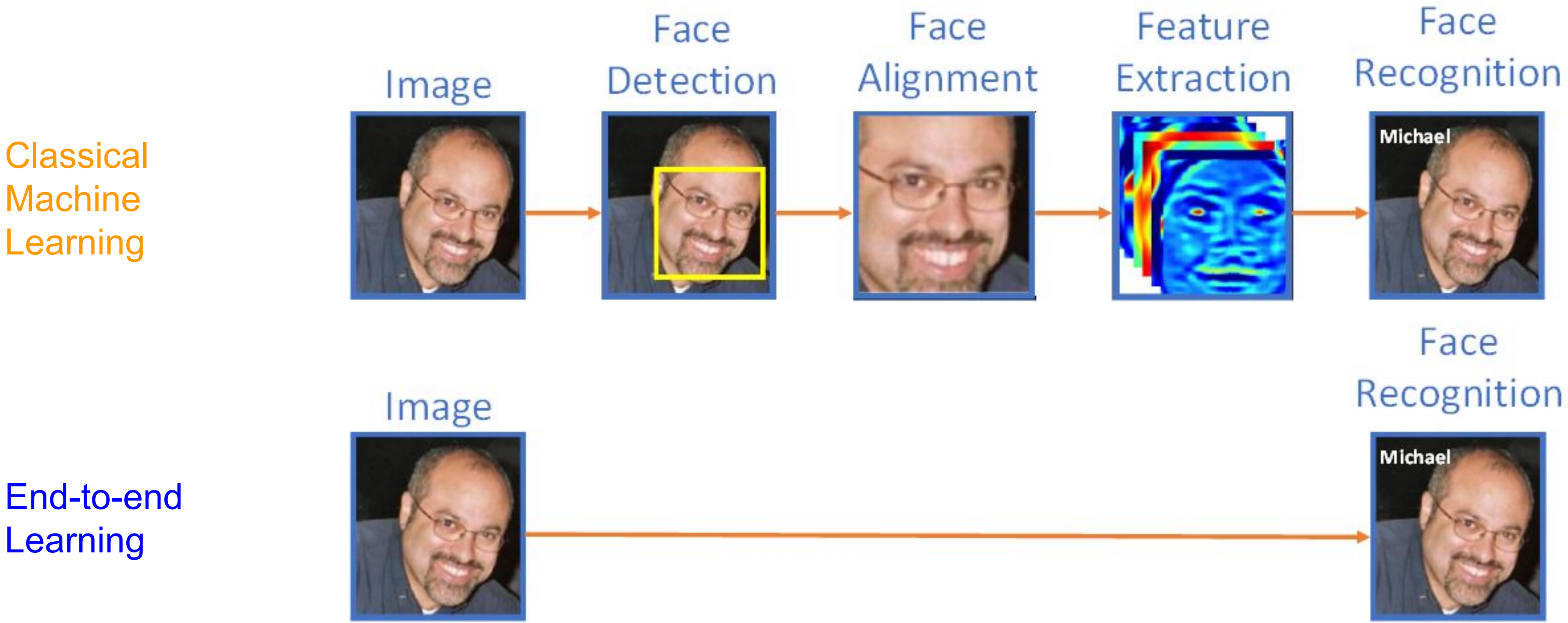
- End-to-end inverse problems
- VAIM Architecture
- Results in Toy Problems
- Results in DIS



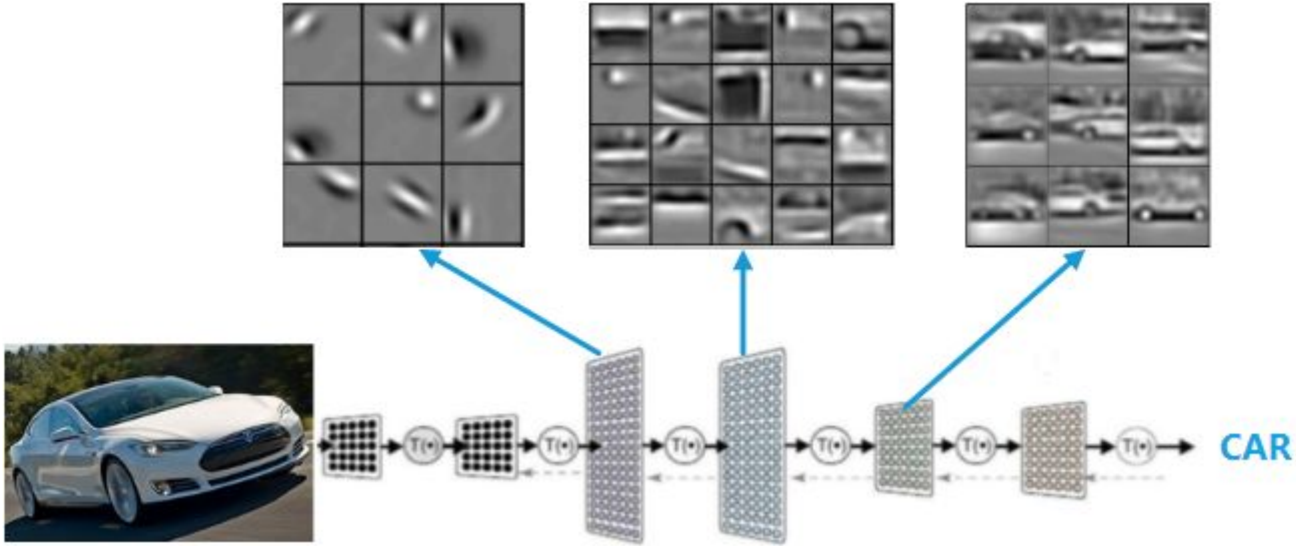
Inverse mapper architectures

End-to-End Learning

- End-to-End Learning
  - Machine learning model automatically learns all features
  - Directly convert input data into output prediction



End-to-End Learning in Image Recognition



Source: https://goo.gl/1KsWvF

End-to-End Learning for Inverse Problems

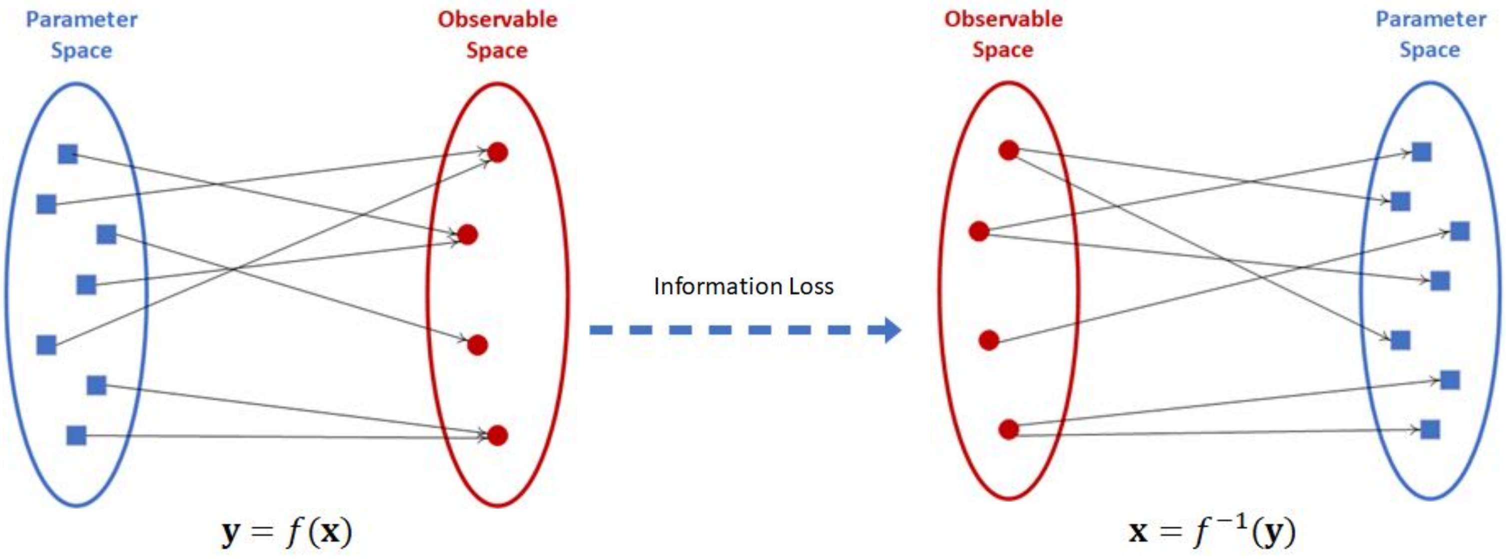
Ill-posedness of Inverse Problems

Forward Mapper

Backward Mapper

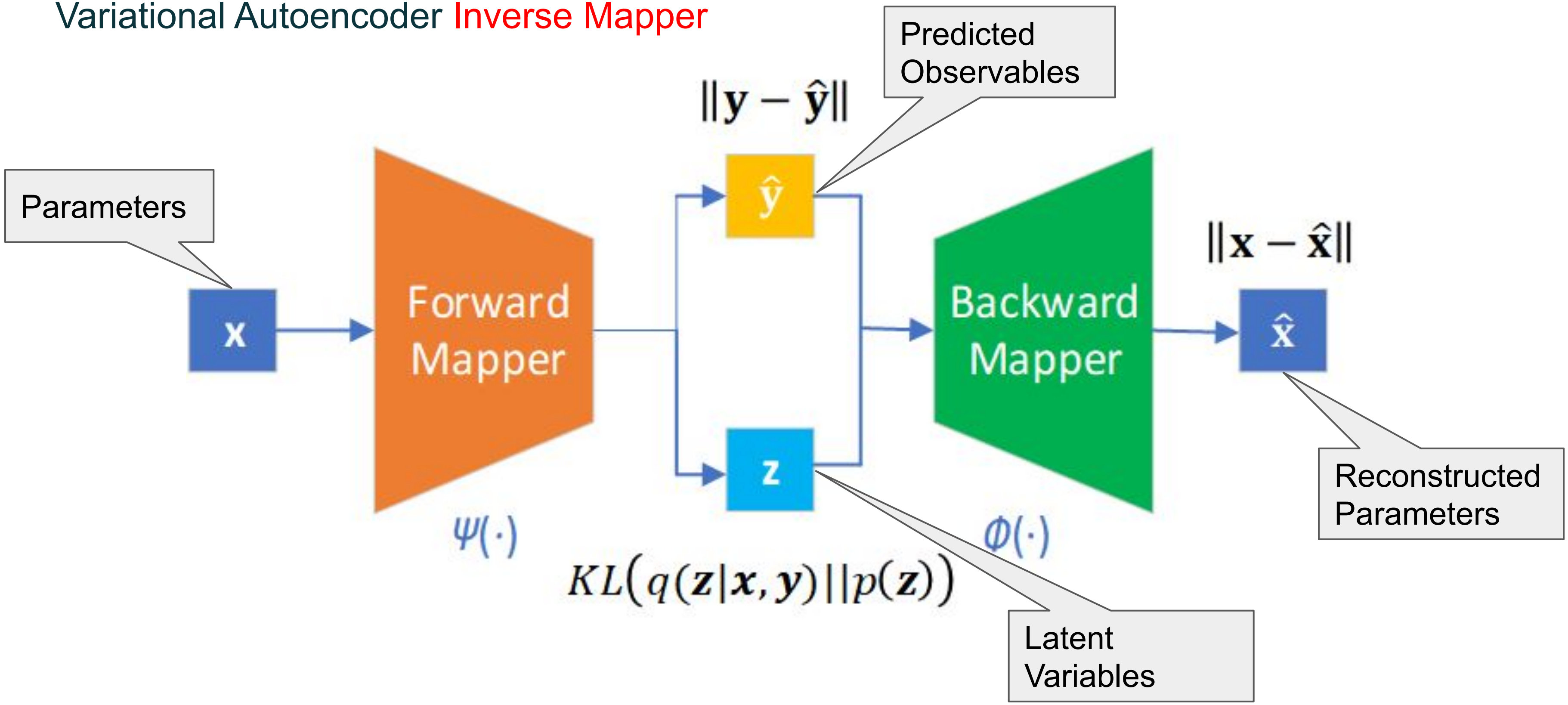
well-posed

ill-posed



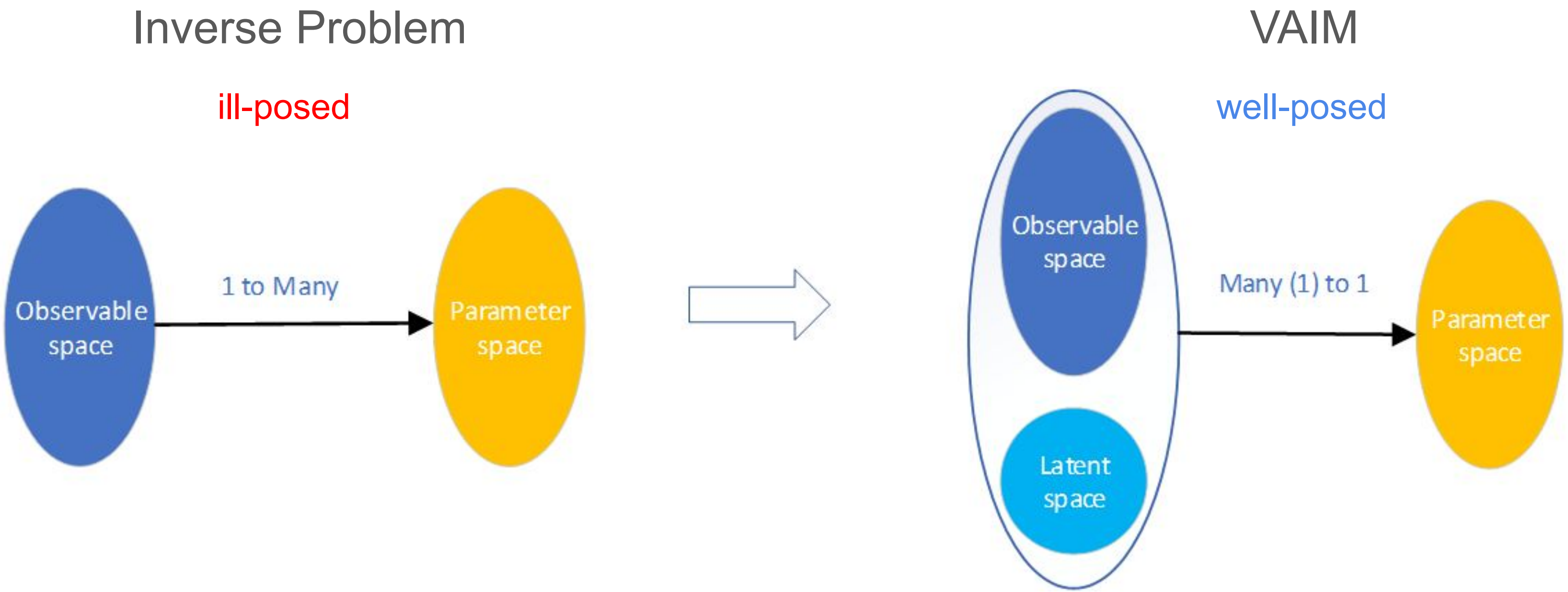
Machine Learning Architecture

Variational Autoencoder Inverse Mapper

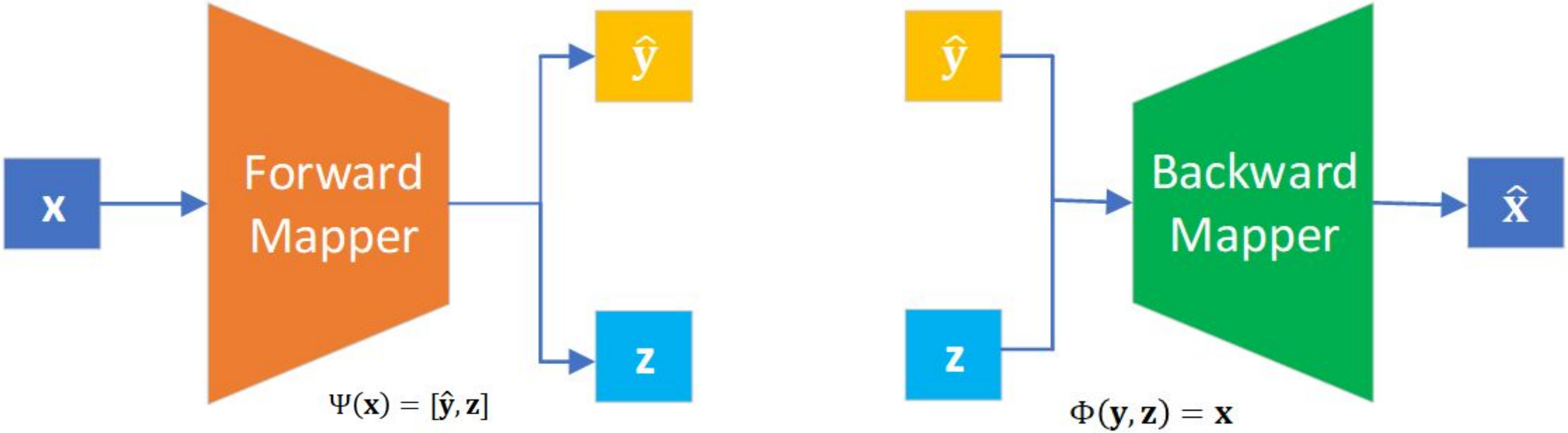


Fundamental Idea

Variational Autoencoder Inverse Mapper



Forward Mapper and Backward Mapper



Learn posterior distribution
$p(\mathbf{z}|\mathbf{x}, \mathbf{y})$

Learn likelihood distribution
$p(\mathbf{x}, \mathbf{y}|\mathbf{z})$

Math behind Variational Autoencoder Inverse Mapper

- Approximate
  - True posterior distribution $p(\mathbf{z} \mid \mathbf{x}, \mathbf{y})$
- Variational Inference
  - Learn an approximate distribution $q(\mathbf{z} \mid \mathbf{x}, \mathbf{y})$ such that

$$q(\mathbf{z} \mid \mathbf{x}, \mathbf{y}) \sim p(\mathbf{z} \mid \mathbf{x}, \mathbf{y})$$

  - Minimize the Kullback-Leibler (KL) divergence

$$\min KL(q(\mathbf{z} \mid \mathbf{x}, \mathbf{y}) \,||\, p(\mathbf{z} \mid \mathbf{x}, \mathbf{y}))$$

Math behind Variational Autoencoder Inverse Mapper (cont.)

- Variational Autoencoder Theory

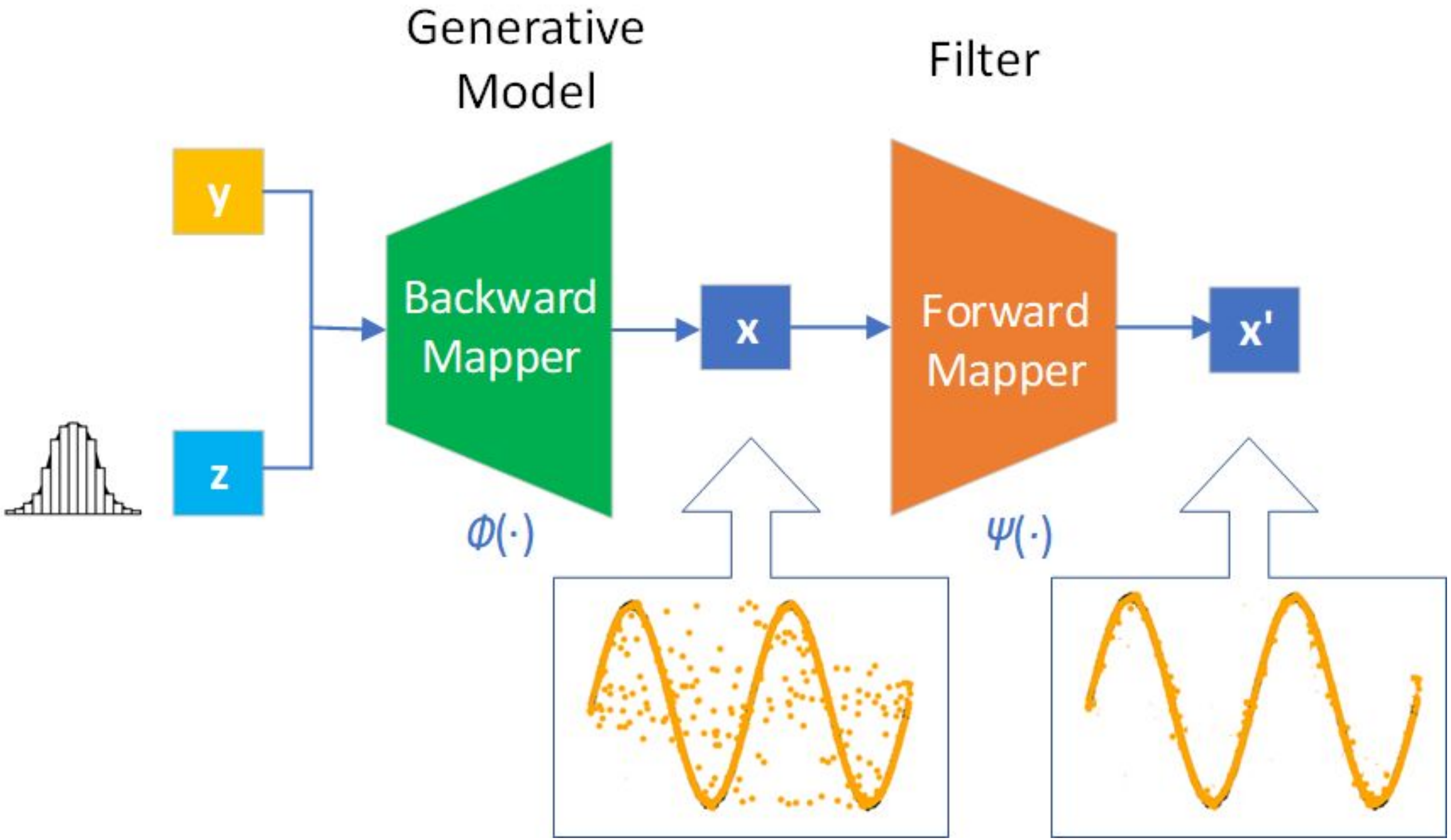
  $$\min KL(q(\mathbf{z} \mid \mathbf{x}, \mathbf{y}) \mid\mid p(\mathbf{z} \mid \mathbf{x}, \mathbf{y}))$$

  Equivalent to

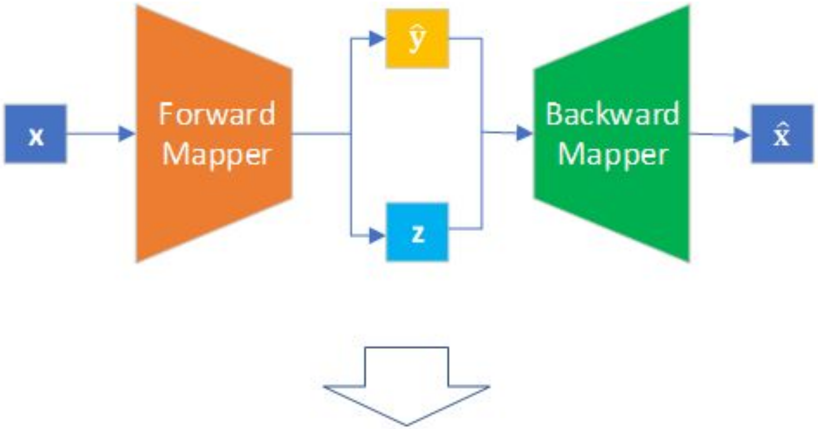
  $$\min \|\mathbf{y} - \hat{\mathbf{y}}\|_2^2 + \|\mathbf{x} - \hat{\mathbf{x}}\|_2^2 + KL(q(\mathbf{z} \mid \mathbf{x}, \mathbf{y}) \mid\mid p(\mathbf{z}))$$

  - True prior distribution
    - Select tractable $p(\mathbf{z})$ distribution easy to generate
      - Gaussian
      - Uniform

Variational Autoencoder Inverse Mapper Production Mode

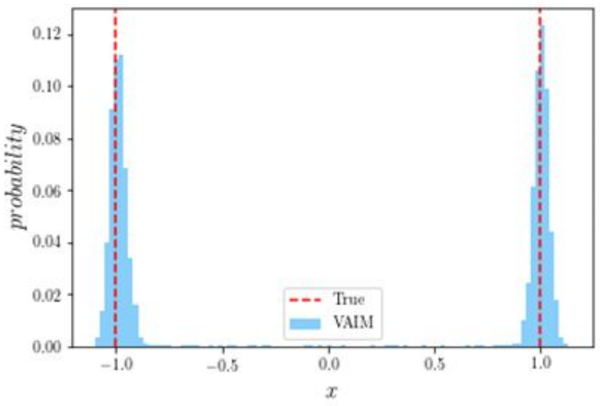


Toy Inverse Problems

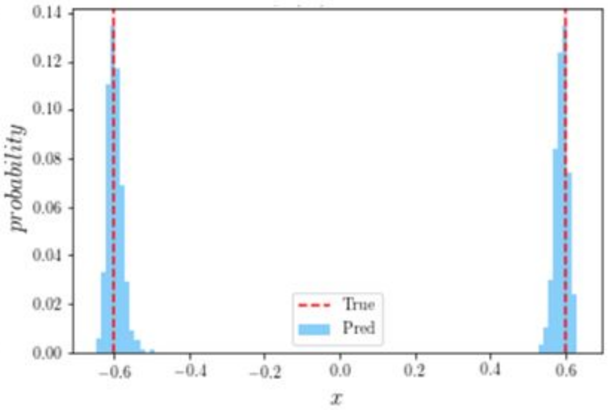


Toy Problems	Solution Characteristics
1) $f(x) = x^2$	2 solutions (except x = 0)
2) $f(x) = \sin(x)$	Finite but different number of solutions
3) $f(x) = {x_0}^2 + {x_1}^2$	Infinite solutions

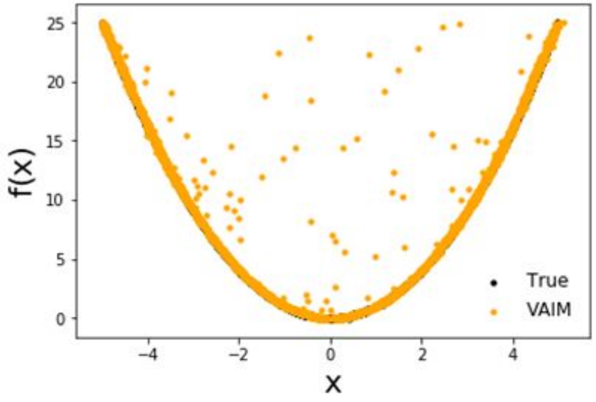
Toy Problem 1) $f(x) = x^2$



Parameter x distribution for $f(x) = 1.0$



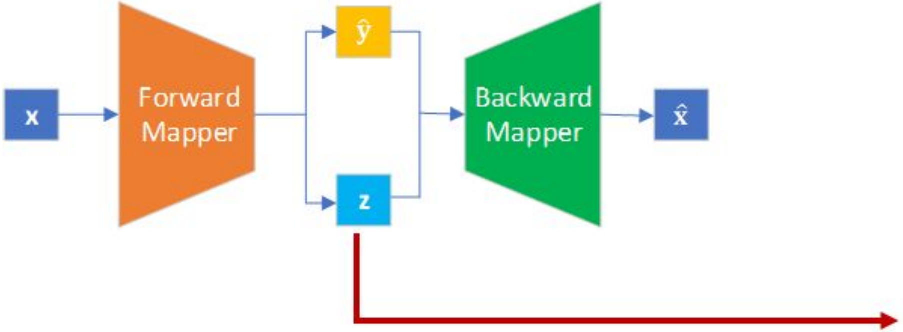
Parameter x distribution for $f(x) = 0.36$

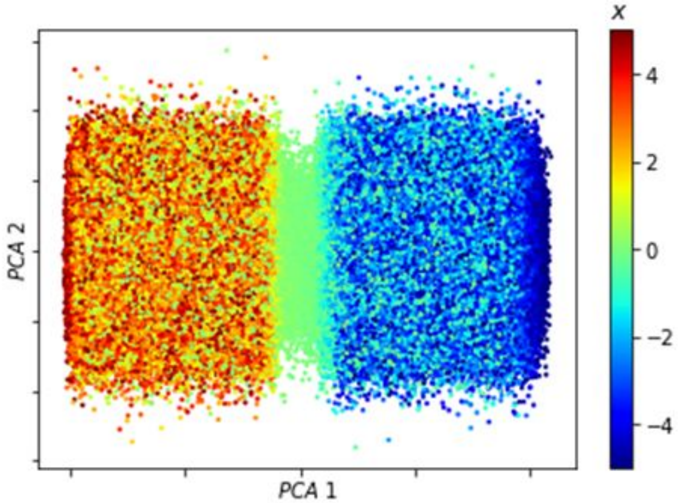


Uniformly sampling $f(x)$

Toy Problem 1) $f(x) = x^2$

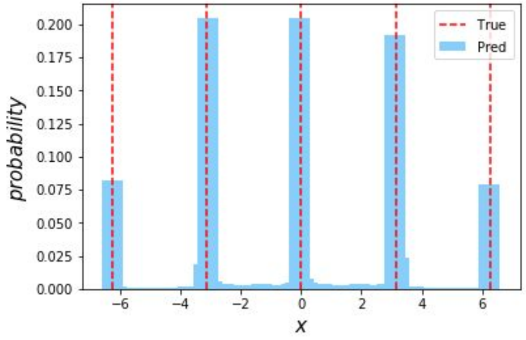
Latent Space Analysis



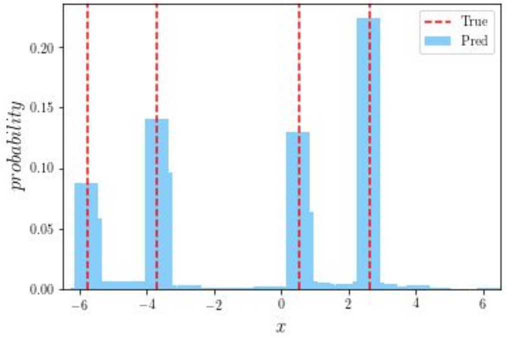


Latent space captures the sign information

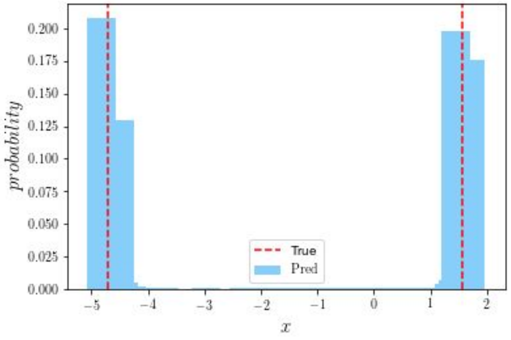
Toy Problem 2) $f(x) = \sin(x), x \in [-2\pi, 2\pi]$



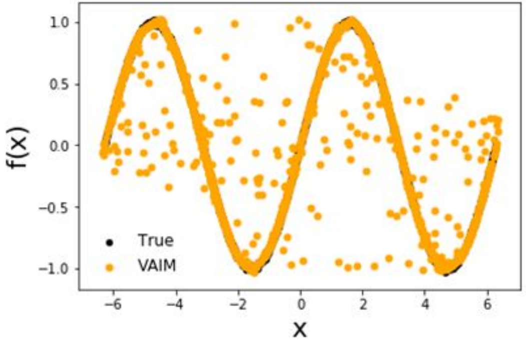
$f(x) = 1.0$



$f(x) = -0.5$



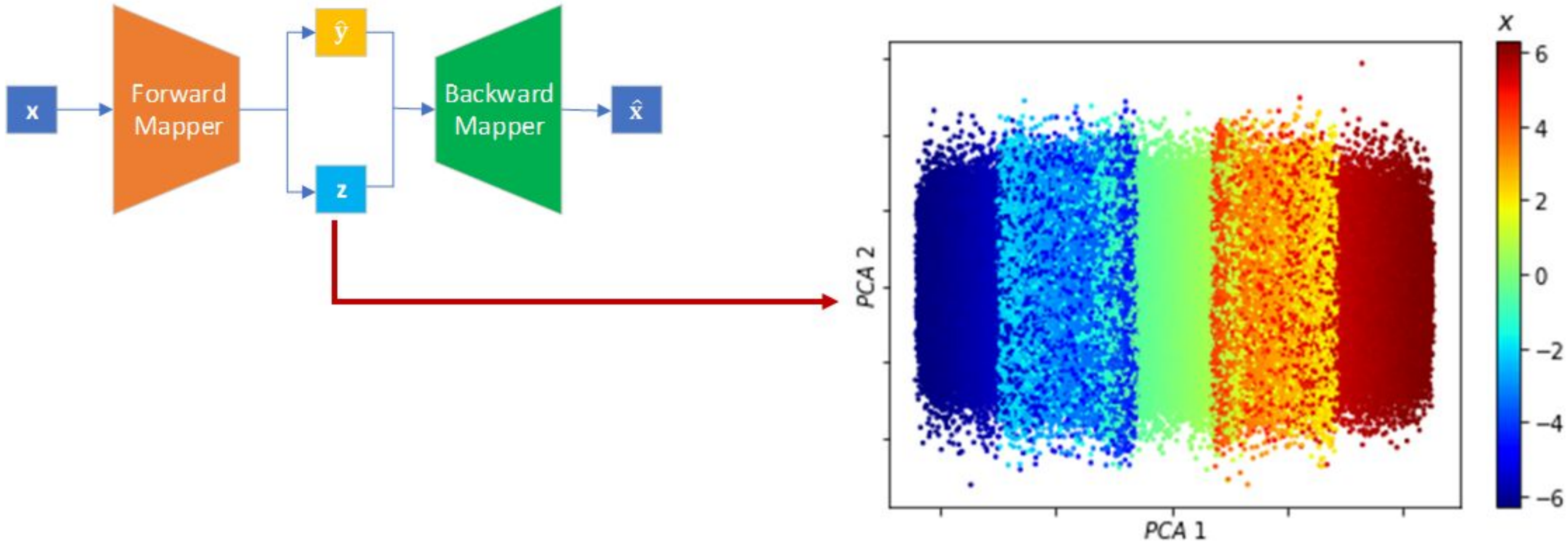
$f(x) = 0.0$



Uniformly sampling $f(x)$

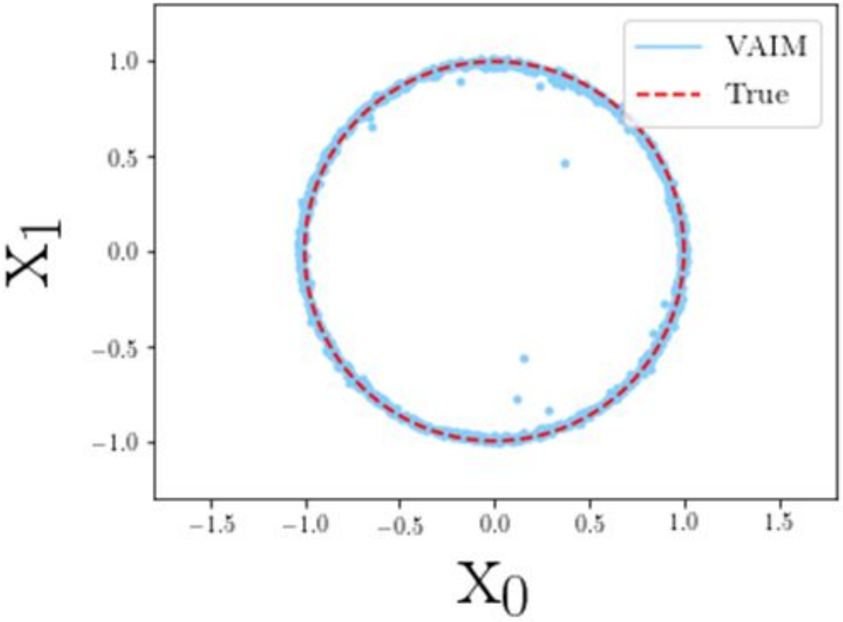
Toy Problem 2) $f(x) = \sin(x), x \in [-2\pi, 2\pi]$

Latent Space Analysis



Latent space captures the period information

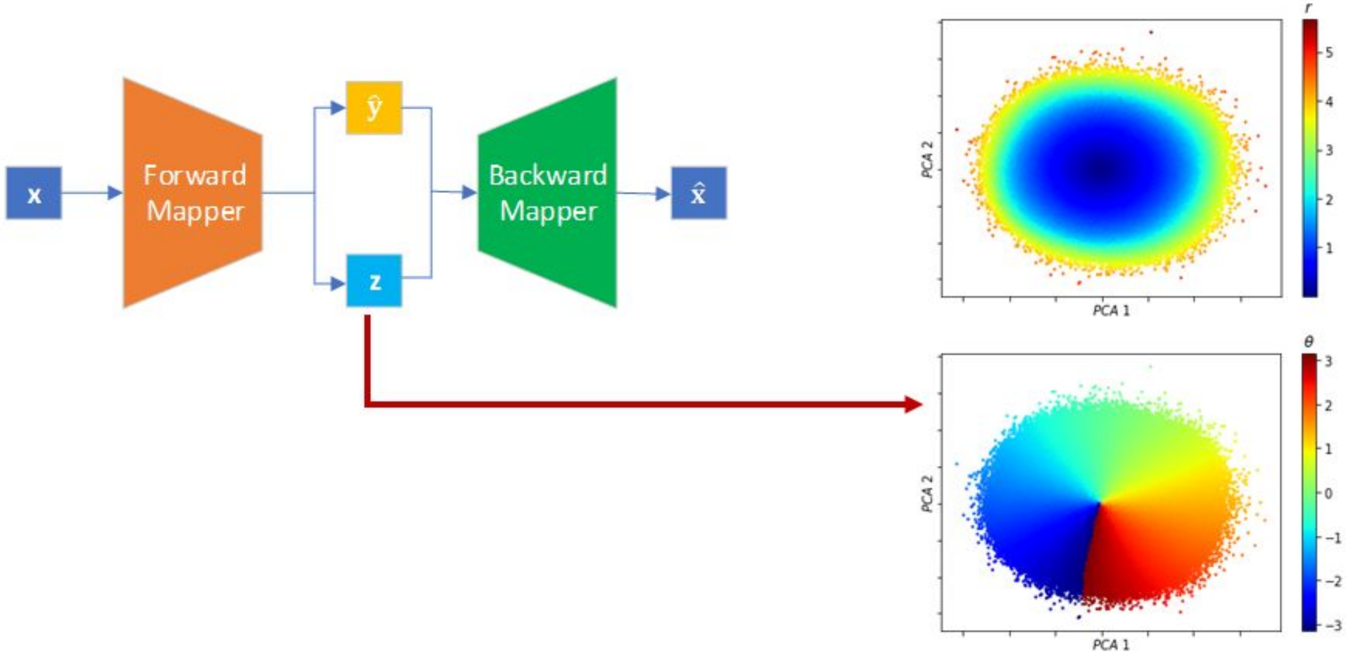
Toy Problem 3) $f(x) = x_0^2 + x_1^2$



Parameter (x_0, x_1) distribution for $f(x) = 1.0$

Toy Problem 3) $f(x) = x_0^2 + x_1^2$

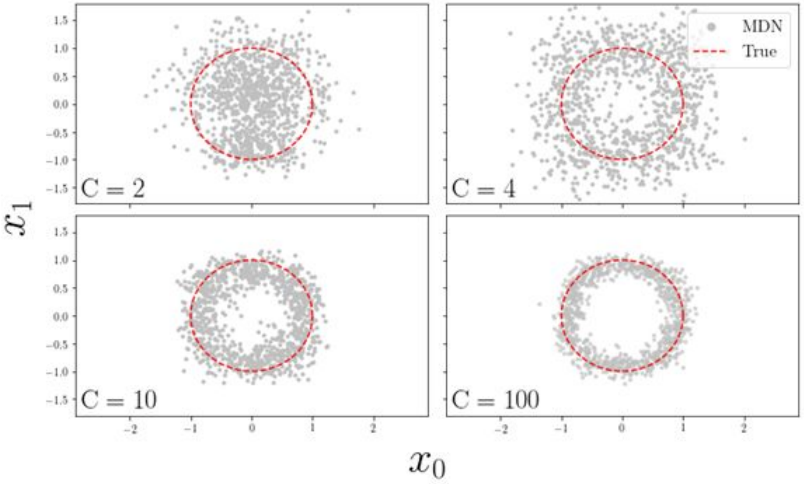
Latent Space Analysis



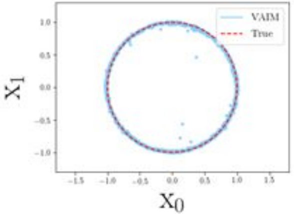
Latent space captures the radius and polar angle information

Comparison with Mixture Density Network (MDN)

- Fundamental Idea of MDN
  - Construct a conditional probability $p(\mathbf{y}|\mathbf{x})$
  - Approximated with mixing Gaussian components
  - Assumption
    - (Finite) Gaussian Mixture
    - Poor approximation when the inverse problem is significantly non-Gaussian
- Advantage of VAIM
  - No Gaussian Assumption

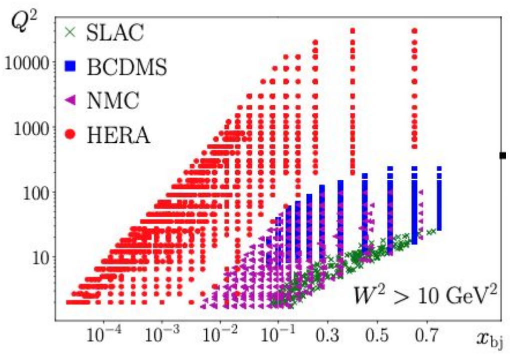


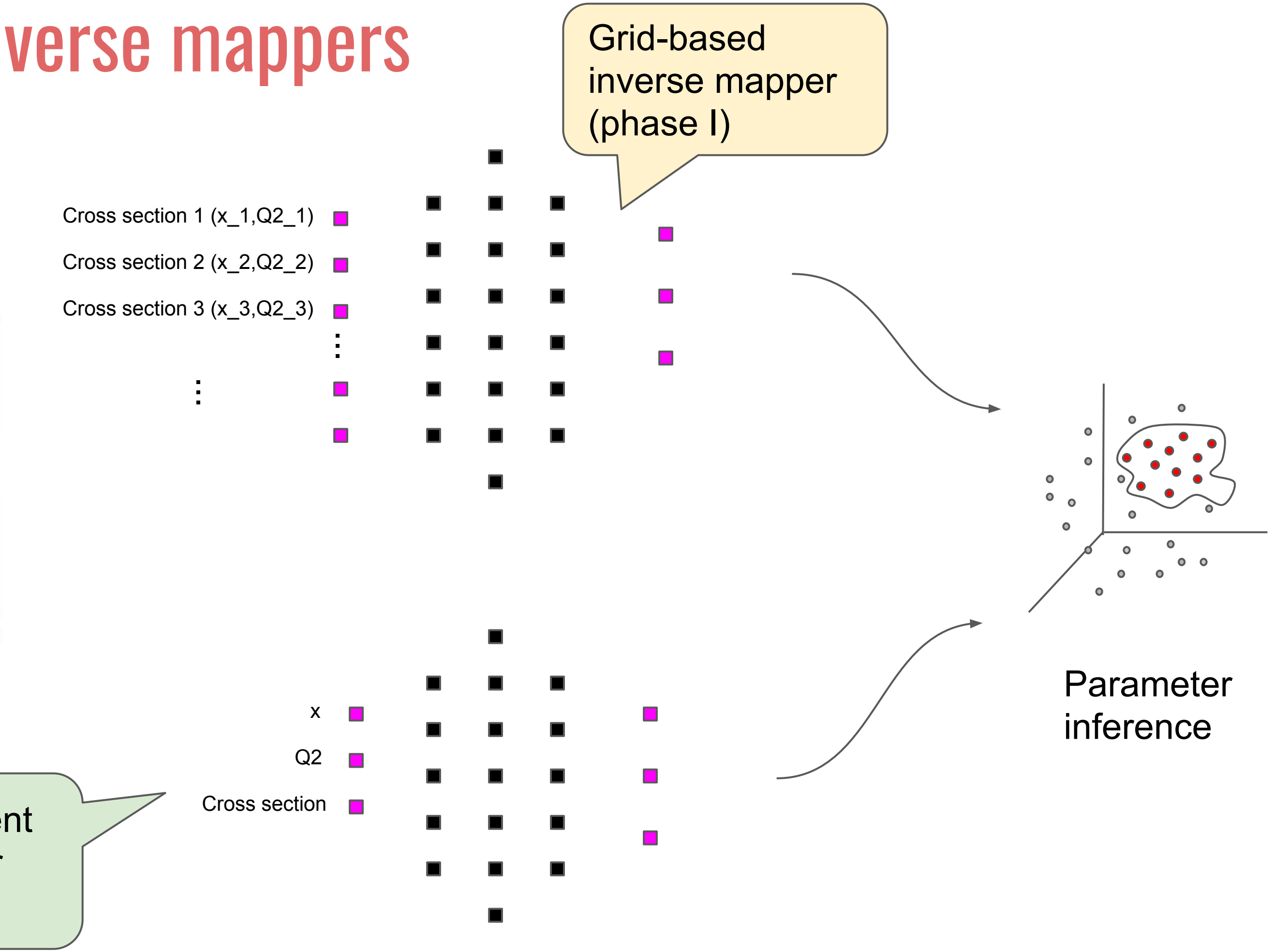
MDN predictions for toy problem $f(x) = {x_0}^2 + {x_1}^2$



VAIM predictions for toy problem $f(x) = {x_0}^2 + {x_1}^2$

Designing the inverse mappers



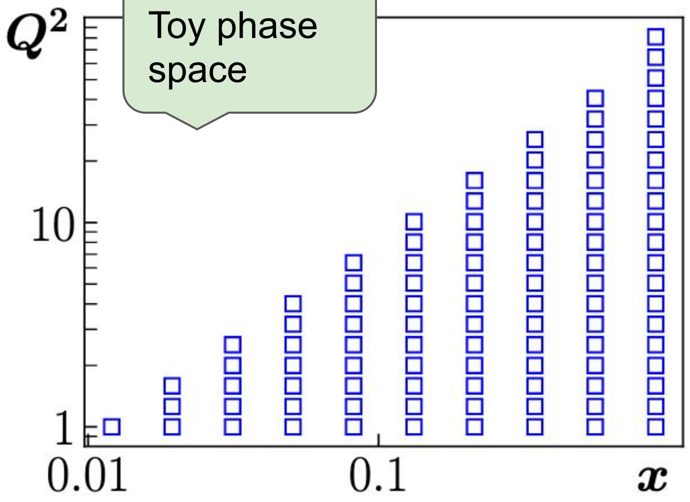


Toy DIS Problem

Toy setup

Toy observables

$$\sigma_p(x,Q^2) = 4u(x,Q^2) + d(x,Q^2),$$
$$\sigma_n(x,Q^2) = 4d(x,Q^2) + u(x,Q^2).$$

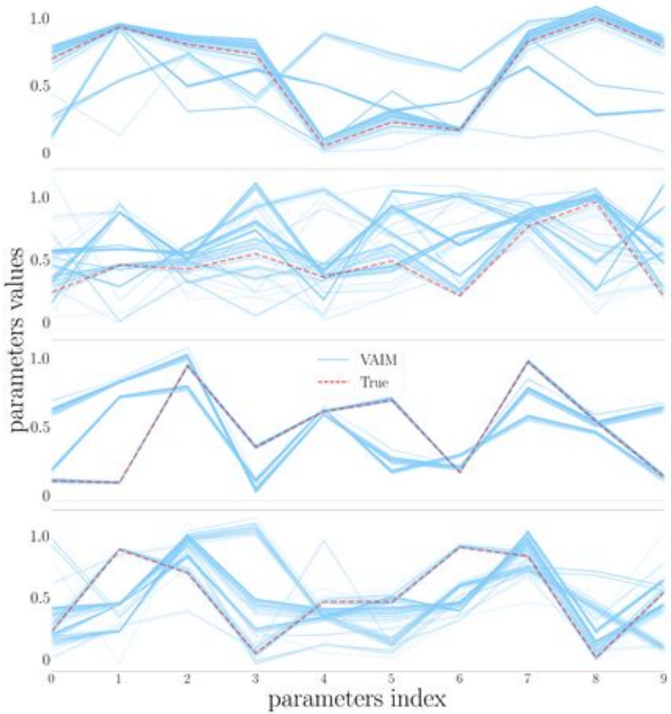


$$u(x,Q^2) = N_u(Q^2)\, x^{\alpha_u(Q^2)}(1-x)^{\beta_u(Q^2)}(1+\gamma_u(Q^2)\sqrt{x}+\delta_u(Q^2)\,x),$$
$$d(x,Q^2) = N_d(Q^2)\, x^{\alpha_d(Q^2)}(1-x)^{\beta_d(Q^2)}\,(1+\gamma_d(Q^2)\sqrt{x}+\delta_d(Q^2)\,x),$$

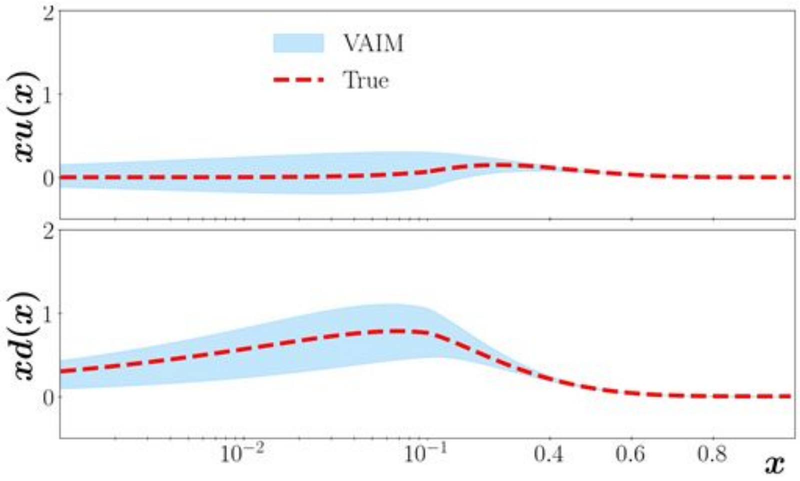
Toy PDFs with DGLAP like behaviour

$$p(Q^2) = p^{(0)} + p^{(1)} s(Q^2), \quad s(Q^2) = \log\left(\frac{\log(Q^2/\Lambda^2_{\rm QCD})}{\log(Q_0^2/\Lambda^2_{\rm QCD})}\right)$$
$$p = \{N_{u,d},\ \alpha_{u,d},\ \beta_{u,d},\ \gamma_{u,d},\ \delta_{u,d}\}$$

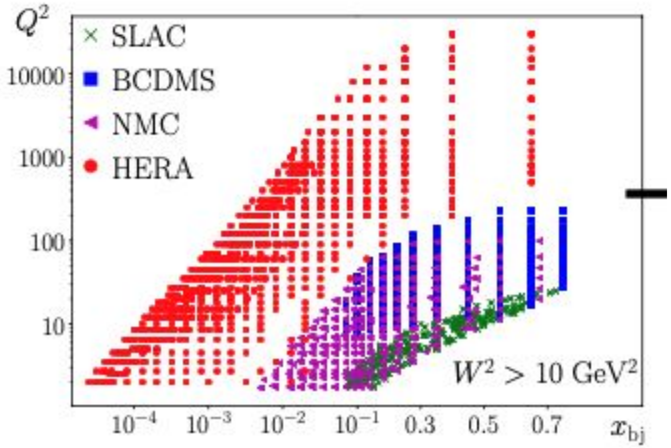
Toy DIS Problem



Parameter distributions generated by VAIM in four control samples



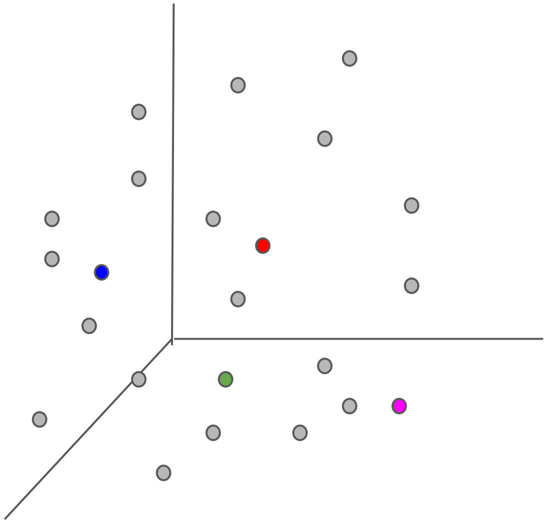
Reconstructed PDF using a control sample



Application to real QCD

Step 1: training samples from JAM

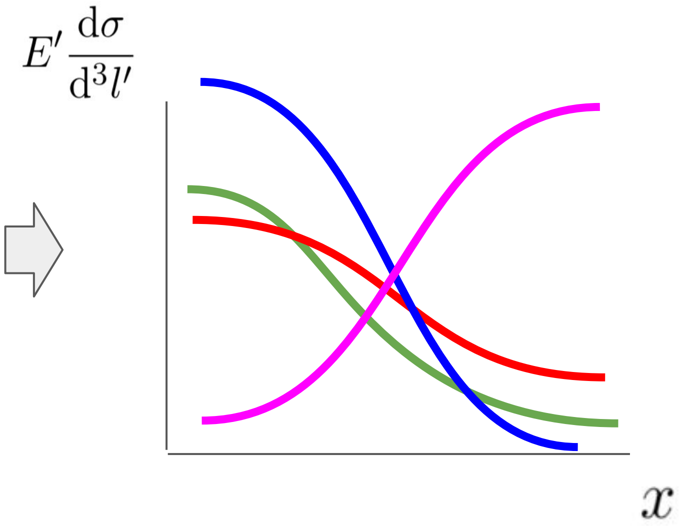
Parameter space



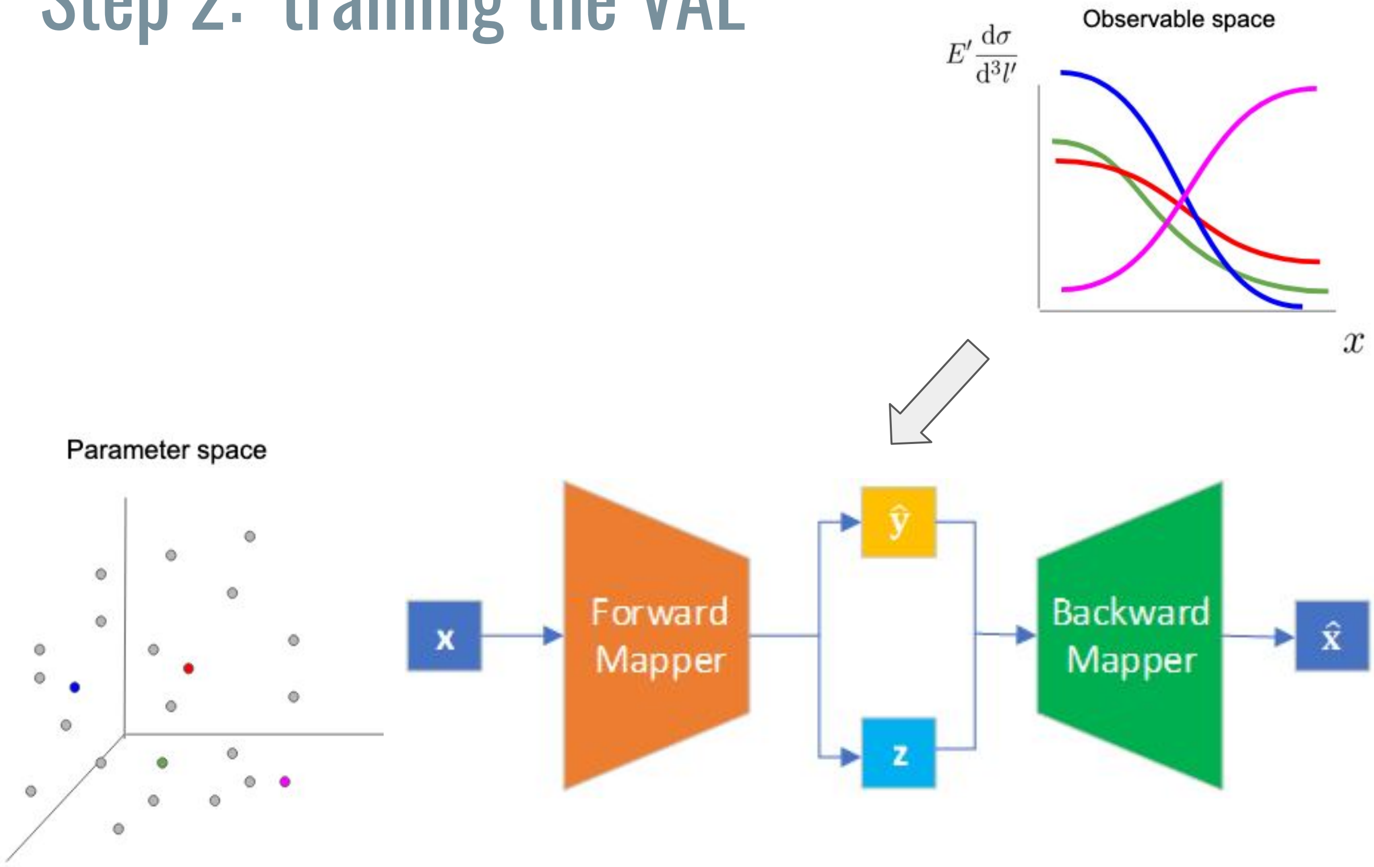
Theory

$$E' \frac{\mathrm{d}\sigma}{\mathrm{d}^3 l'} = \sum_i \int_x^1 \frac{\mathrm{d}\xi}{\xi} H_i(\xi) f_i\left(\frac{x}{\xi}; \boldsymbol{a}\right)$$

Observable space

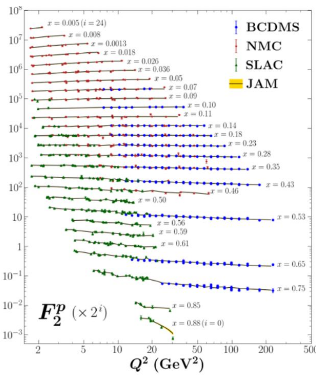


Step 2: training the VAE

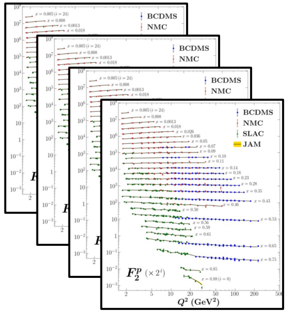


Step 3: Parameter inference

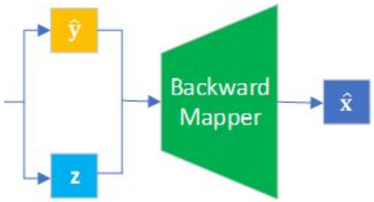
Original data

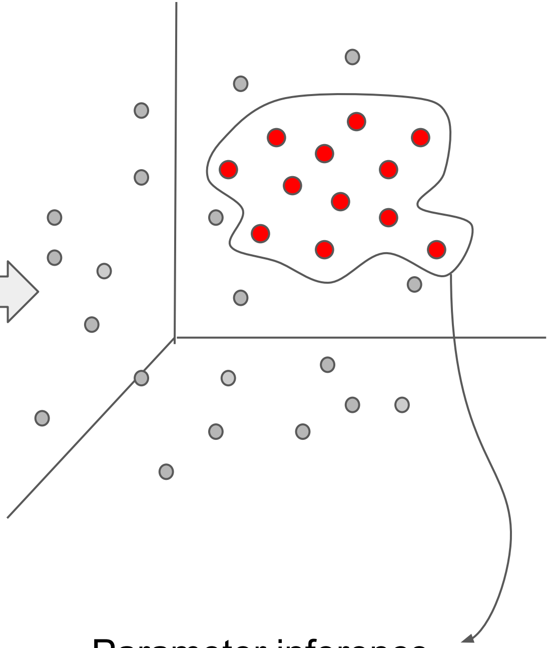


Replica data



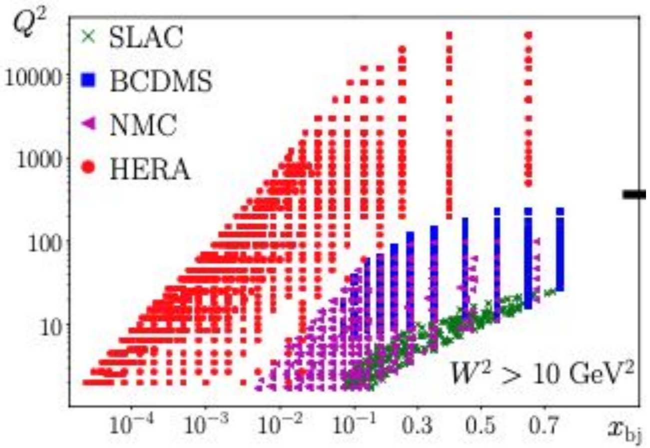
Trained inverse mapper

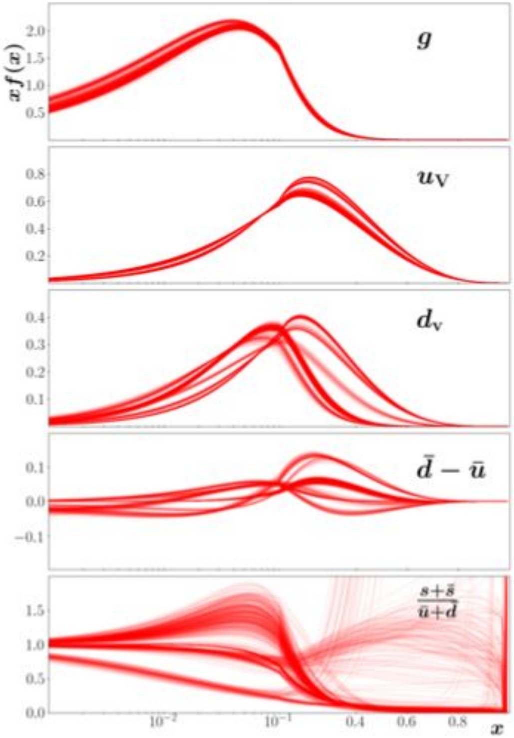




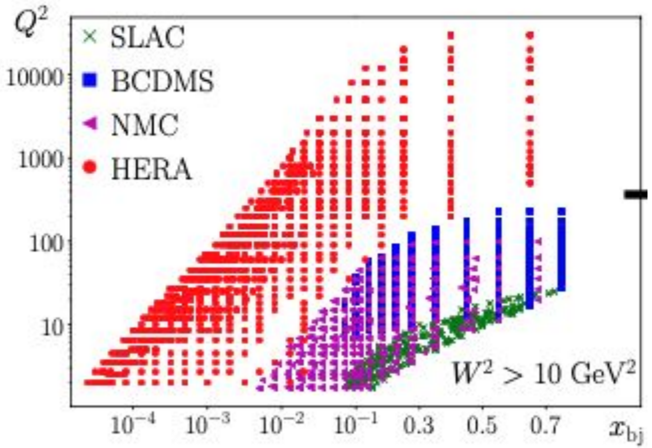
Parameter inference

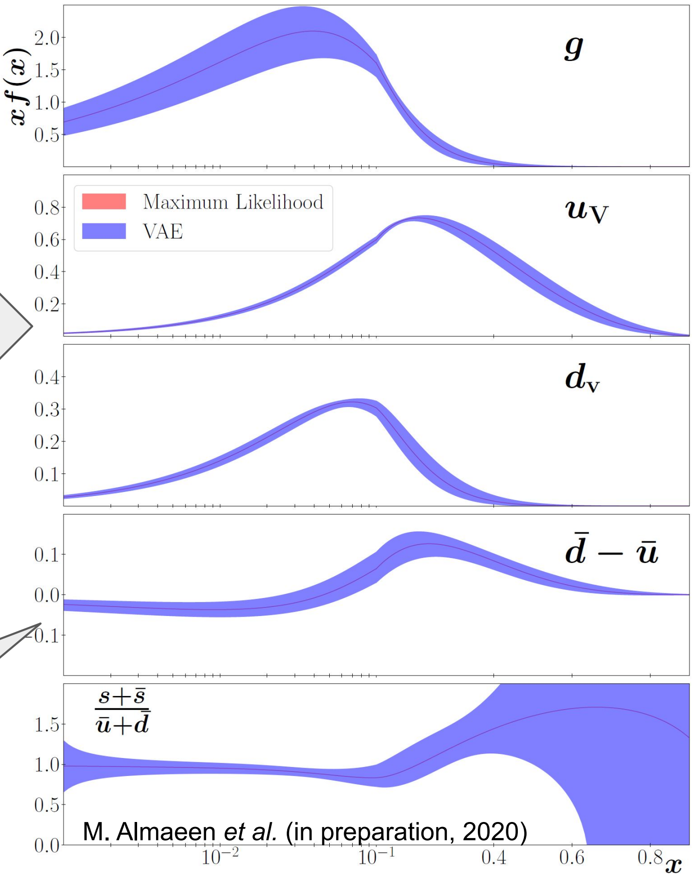
Upol PDF inference using JAM MC framework





Upol PDF inference using JAM inverse mapper

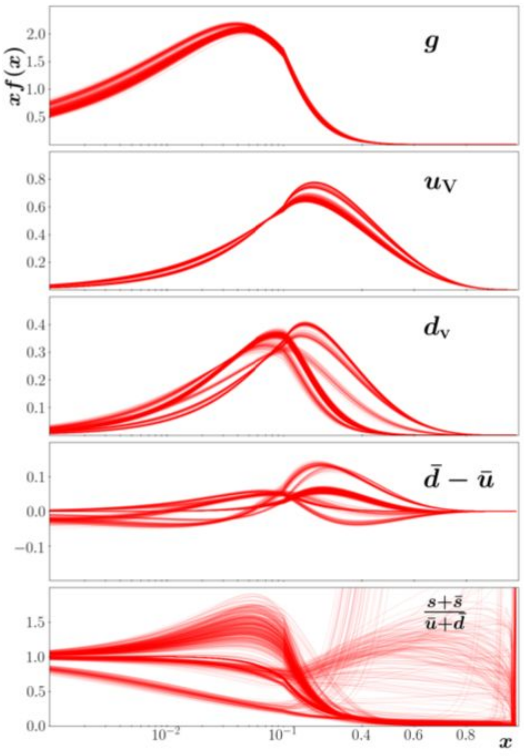




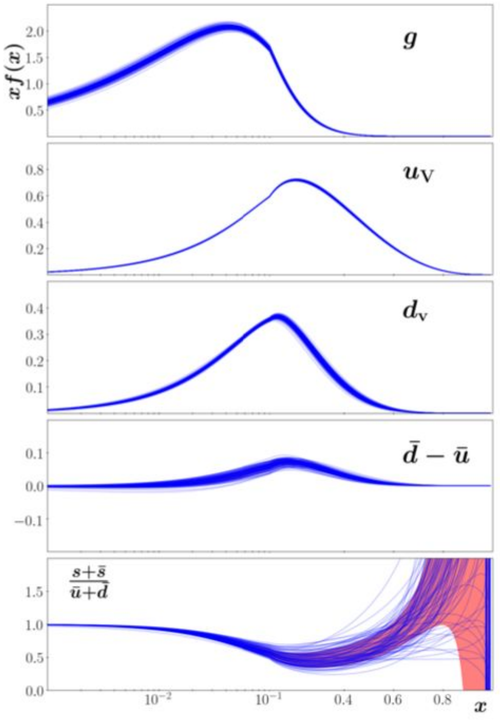
Inverse mappers results very close to JAM methodology

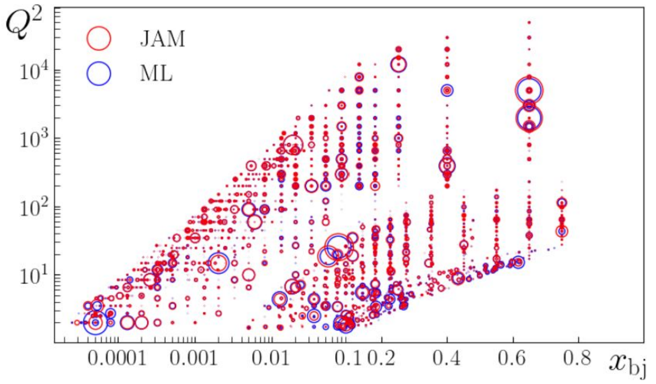
Does it work?

JAM MC approach



Inverse mapper approach





$$\frac{\chi^2_{\mathrm{JAM}}}{N_{\mathrm{pts}}} = 1.25$$

$$\frac{\chi^2_{\mathrm{ML}}}{N_{\mathrm{pts}}} = 1.36$$

Preliminary

Summary

- VAIM – an end-to-end deep learning model for inverse problems
  - Works well in three toy inverse problems with very different solution patterns
  - Preliminary success in toy DIS problems
  - Preliminary success in real PDFs from actual global QCD analysis of experimental data on inclusive DIS
- Challenges
  - Stability
  - Robustness
  - Programming physics into the deep learning framework

Publication: Almaeem et al., Variational Autoencoder Inverse Mapper: An End-to-End Deep Learning Framework for Inverse Problems, IJCAI2021, to be submitted today.

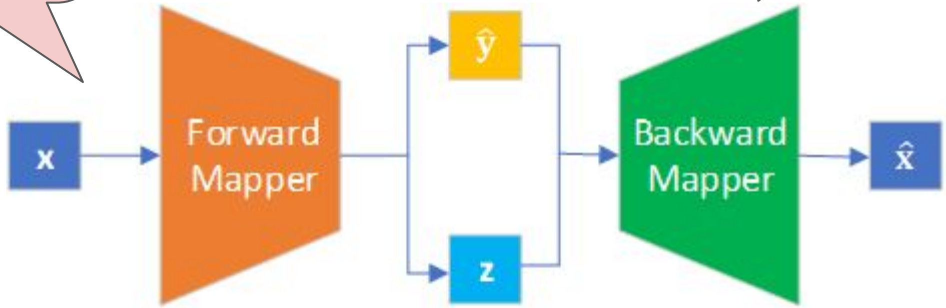
Acknowledgement: The work is supported by CNF.

Future directions

Where do we go from here?

Quantitative demonstration that ML can replace current paradigm

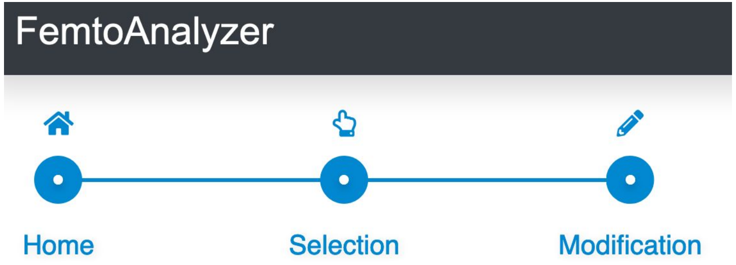
Inclusion of more observables: *DY, jets, SIDIS*



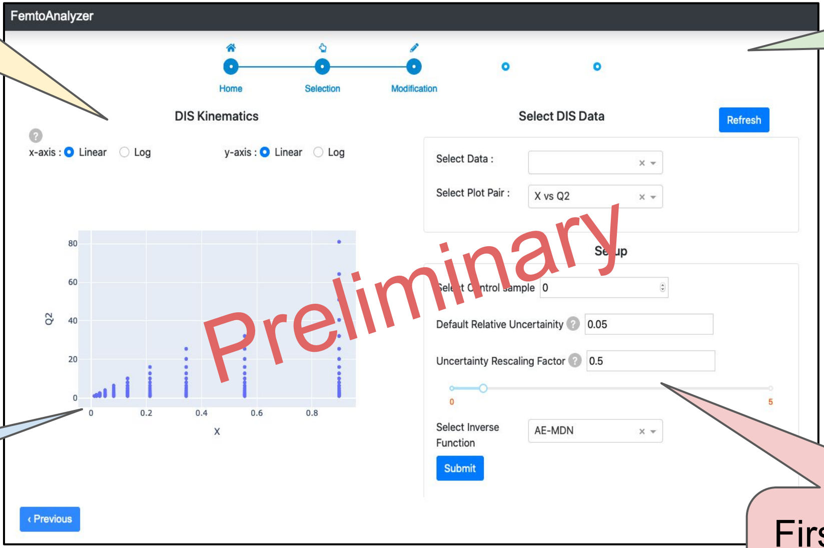
Integration with web-frameworks

Extensions to TMDs, GPDs

Status of web framework



Currently using toy DIS-like with grid-based models



Deployed on *Heroku* web service

First prototypes for data manipulation tools

First prototypes for 1D hadron structure visualization

Where do we go from here?

Where can we host the web application $$$?

Build an AI-based database

A tool for JLab and EIC community

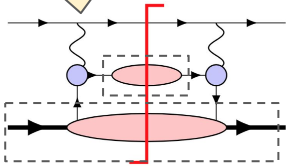
FemtoAnalyzer as an AI agent

The ML workforce

Jefferson Lab

co-PI

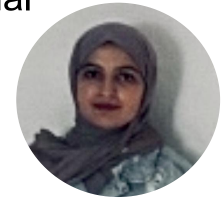
QCD Theory

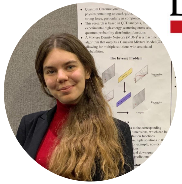


ODU

PI

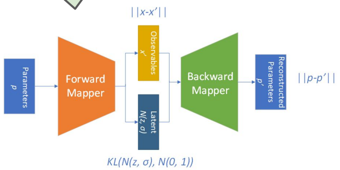
ODU
Manal





Eleni
DAVIDSON

Inverse mappers

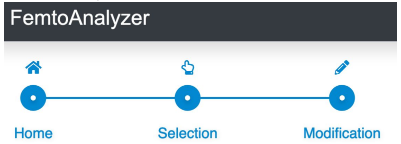


Rida
DAVIDSON

DAVIDSON

ODU
Heramb

Web-interface



Raghu
DAVIDSON

Annabel
DAVIDSON