Symbolic Extraction of Non-Perturbative TMD from Drell–Yan Data

Cole Granger · Cristiano Fanelli

William & Mary — School of Computing, Data Sciences and Physics

with the MAP Collaboration — Bacchetta, Bertone, Bissolotti, Cerutti, Radici, Rodini, Rossi

CAREER

The result, up front

An analytic non-perturbative TMD extracted directly from data

A neural network fit to global Drell–Yan data, distilled by symbolic regression into a compact closed-form expression.

9

free numerical constants

1.040

χ^2/ndf over 482 data points

31

total expression complexity

Comparable to the parent neural-network fit (χ^2/ndf = 0.941) — at a tiny fraction of the parameters.

PART I

A challenging problem

TMDs, the non-perturbative problem, and a tension between flexibility and interpretability

Transverse-momentum-dependent distributions

- TMDs give a three-dimensional, momentum-space picture of the nucleon — by describing the joint dependence of parton distributions on the longitudinal momentum fraction x and intrinsic transverse momentum $\mathbf{k}_\perp$
- We focus on the unpolarized quark TMD PDF $f_1(x, k_\perp)$.
- It contains a **perturbative** part (computable in QCD) and a **non-perturbative** part f_{NP} encoding long-distance physics.
- f_{NP} cannot be calculated perturbatively — it must be inferred from data, typically Drell–Yan measurements.

Key idea

f_{NP} is the object we want to determine — but it is not directly observable. It enters measured cross sections only through TMD evolution, flavor sums, and kinematic convolutions.

WILLIAM & MARY | DATA SCIENCE
CHARTERED 1693

Fitting f_{NP}: flexibility vs. interpretability

Traditional parametrizations

Gaussian / exponential forms with low-dimensional kinematic dependence. Practical and widely used — but they impose functional assumptions that can bias the extraction.

Neural-network fits

Flexible; they reduce model dependence. But the fit is high-dimensional and hard to interpret or compare directly with traditional analyses.

Can we combine the flexibility of a network with the interpretability of a formula?

Symbolic regression as a bridge

1

FIT

Train a factorized neural network directly on global Drell–Yan data at N^3LL accuracy.

2

DISTILL

Apply symbolic regression (PySR) to each network component to obtain compact analytic expressions.

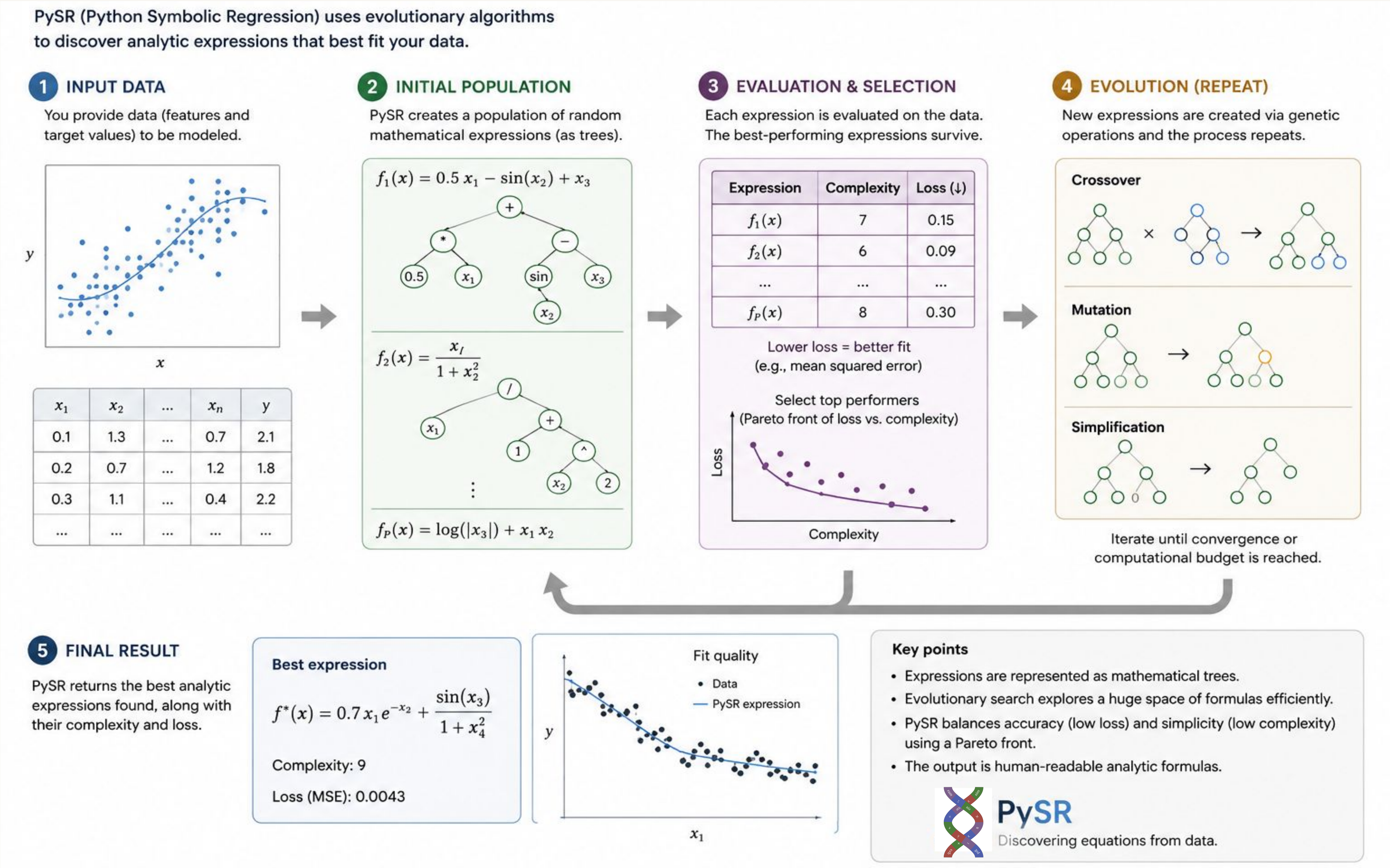
3

SELECT

Choose the final formula from the Pareto front balancing experimental χ^2 and complexity

Result: the first f_{NP} obtained directly from data rather than assumed.

PySR


PySR (Python Symbolic Regression) uses evolutionary algorithms
to discover analytic expressions that best fit your data.
1 INPUT DATA
You provide data (features and target values) to be modeled.
y
x
x_1 x_2 ... x_n y
0.1 1.3 ... 0.7 2.1
0.2 0.7 ... 1.2 1.8
0.3 1.1 ... 0.4 2.2
... ...
2 INITIAL POPULATION
PySR creates a population of random mathematical expressions (as trees).
$f_1(x) = 0.5\,x_1 - \sin(x_2) + x_3$
+ * − 0.5 x_1 sin x_3 x_2
$f_2(x) = \frac{x_l}{1 + x_2^2}$
/ x_1 + 1 ^ x_2 2
$f_P(x) = \log(|x_3|) + x_1\,x_2$
3 EVALUATION & SELECTION
Each expression is evaluated on the data. The best-performing expressions survive.
Expression Complexity Loss (↓)
$f_1(x)$ 7 0.15
$f_2(x)$ 6 0.09
... ...
$f_P(x)$ 8 0.30
Lower loss = better fit
(e.g., mean squared error)
Select top performers
(Pareto front of loss vs. complexity)
Loss
Complexity
4 EVOLUTION (REPEAT)
New expressions are created via genetic operations and the process repeats.
Crossover
Mutation
Simplification
0
Iterate until convergence or computational budget is reached.
5 FINAL RESULT
PySR returns the best analytic expressions found, along with their complexity and loss.
Best expression
$f^*(x) = 0.7\,x_1 e^{-x_2} + \frac{\sin(x_3)}{1 + x_4^2}$
Complexity: 9
Loss (MSE): 0.0043
Fit quality
Data
PySR expression
y
x_1
Key points
• Expressions are represented as mathematical trees.
• Evolutionary search explores a huge space of formulas efficiently.
• PySR balances accuracy (low loss) and simplicity (low complexity) using a Pareto front.
• The output is human-readable analytic formulas.
PySR
Discovering equations from data.

Symbolic Regression in Hadron Physics and Beyond

SR has already recovered interpretable structures across several areas:

- GPDs from lattice + phenomenology (Dotson et al., EXCLAIM Coll.);
- Precision LHC pheno with rediscovery of known QED angular distributions (Bendavid et al)
- Fragmentation forms from SIDIS (Makke et al.)

positions SR — paired with their new ECC method — as a practical, interpretable tool for extracting physical laws, and a step toward SR-assisted nucleon tomography

A. Dotson et al., "Generalized Parton Distributions from Symbolic Regression," arXiv:2504.13289 [hep-ph] (2025).

M. Morales-Alvarado, D. Conde, J. Bendavid, V. Sanz, M. Ubiali, "Symbolic Regression for Precision LHC Physics," NeurIPS 2024 (ML & Physical Sciences), arXiv:2412.07839 [hep-ph].

J. Bendavid, D. Conde, M. Morales-Alvarado, V. Sanz, M. Ubiali, "Angular Coefficients from Interpretable Machine Learning with Symbolic Regression," JHEP 02 (2026) 081 [arXiv:2508.00989 [hep-ph]].

N. Makke, S. Chawla, "Inferring Interpretable Models of Fragmentation Functions using Symbolic Regression," Mach. Learn. Sci. Tech. 6, 025003 (2025) [arXiv:2501.07123 [hep-ph]].

Symbolic Regression in Hadron Physics and Beyond

SR has already recovered interpretable structures across several areas:

- GPDs from lattice + phenomenology (Dotson et al., EXCLAIM Coll.);
- Precision LHC pheno with rediscovery of known QED angular distributions (Bendavid et al)
- Fragmentation forms from SIDIS (Makke et al.)

A. Dotson et al., "Generalized Parton Distributions from Symbolic Regression," arXiv:2504.13289 [hep-ph] (2025).

M. Morales-Alvarado, D. Conde, J. Bendavid, V. Sanz, M. Ubiali, "Symbolic Regression for Precision LHC Physics," NeurIPS 2024 (ML & Physical Sciences), arXiv:2412.07839 [hep-ph].

J. Bendavid, D. Conde, M. Morales-Alvarado, V. Sanz, M. Ubiali, "Angular Coefficients from Interpretable Machine Learning with Symbolic Regression," JHEP 02 (2026) 081 [arXiv:2508.00989 [hep-ph]].

N. Makke, S. Chawla, "Inferring Interpretable Models of Fragmentation Functions using Symbolic Regression," Mach. Learn. Sci. Tech. 6, 025003 (2025) [arXiv:2501.07123 [hep-ph]].

SR has not been used to extract non-perturbative TMD directly from data

*Other techniques

- Z. Kang, AI-assisted modeling and Bayesian inference of unpolarized quark transverse momentum distributions from Drell-Yan data (2026)

PART II

Method

From TMD factorization to a factorized network to a distilled formula

Where f_{NP} enters TMD factorization

$$f_1(x, b_T, \mu_f, \zeta_f) = [\, C \otimes f_1 \,] \times \exp\{\text{ TMD evolution }\} \times f_{NP}(x, b_T; \zeta_f)$$

- For **$q_T \ll Q$**, the Drell–Yan cross section factorizes into a perturbatively calculable hard-scattering kernel and the Fourier transform (b_T)-space TMD PDFs, whose long-distance behavior is encoded in f_{NP}
- TMD (Collins–Soper) evolution resums large logarithms; a **b*-prescription** avoids the QCD Landau pole.
- The prescription leaves spurious power corrections at large b_T — these are absorbed into f_{NP}.
- Constraints: **$f_{NP} \to 1$ as $b_T \to 0$** (perturbative limit); non-perturbative large b_T behavior consistent with the expected Sudakov suppression and TMD evolution

A factorized, distillable ansatz

- f_{NP} is parametrized as

$$f_{\mathrm{NP}}(x, b_T; \zeta) = \exp\left[-g_2^2(b_T)\, b_T^2 \ln\left(\zeta/Q_0^2\right)\right] \cdot R(x, b_T)$$

where

$$R(x, b_T) = \exp\left[h_x(x)\, h_b(b_T) + h_{xb}(x, b_T)\right]$$

- **Why factorize?** Symbolic regression scales poorly with dimensionality — a full 2D fit yields expressions too complex to distill (no candidate reached $\chi^2/N_{dat} < 1.3$).
- Decompose into two 1D pieces $h_x(x)$, $h_b(b_T)$ plus a cross term $h_{xb}(x, b_T)$ that captures x–b_T correlation.
- Boundary conditions $h_b(0)=0$, $h_{xb}(x,0)=0$ are enforced by construction ⇒ **$R(x,0)=1$ exactly.**

Keeping h_{xb} "honest"

- L1 sparsity prior
- Capacity cap (4 neurons)
- Subset-balanced χ^2

A small network, trained on the observable

- Each component (h_x, h_b, g_2, h_{xb}) is a shallow MLP with only 4 or 8 neurons — **O(100) parameters total.**
- Trained directly against the experimental χ^2 via a precomputed **convolution grid**: quadrature weights, momentum fractions and scales stored per data bin.

  The NNs learn only the non-perturbative function. All perturbative QCD ingredients are frozen into a precomputed NangaParbat convolution skeleton, allowing efficient training with PyTorch while preserving the full MAP theory calculation. [Phys. Rev. Lett. 135, 021904 (2025)]

- A balanced loss across 59 experiment-specific subsets stops low-statistics sets from being sacrificed.
- Overfitting guards: tiny network, Pareto-knee selection, and symbolic compression.

O(100) → 9

network parameters distilled to analytic constants

… while the global χ^2 changes from 0.941 (NN) to 1.040 (SR)

WILLIAM & MARY | DATA SCIENCE
CHARTERED 1693

Total Loss

$$\mathcal{L} = \mathcal{L}_{\chi^2}^{\rm bal} + \lambda_{L_1} \|\theta_{h_{xb}}\|_1 + \lambda_{L_2} \|\theta_{\rm shape}\|^2.$$

loss term that pushes every subset χ^2 proxy value towards global mean

$$\mathcal{L}_{\chi^2}^{\rm bal} = \frac{1}{N_{\rm exp}} \sum_s w_s \frac{\chi_s^2}{N_s} + \lambda_{\rm bal} \frac{1}{N_{\rm exp}} \sum_s \left(\frac{\chi_s^2}{N_s} - \langle \chi^2/N \rangle \right)^2,$$

L2 weight decay on the parameters of the remaining shape networks (h_x, h_b, g_2), discouraging arbitrarily large weights without strongly constraining the fit.

L1 sparsity penalty applied exclusively to the hxb network parameters

The global Drell–Yan dataset

Source	Experiments	Ndat
Fermilab	E288, E605, E772	233
Tevatron	CDF, D0	71
RHIC	STAR	7
LHC	ATLAS, CMS, LHCb	171
Total	**—**	**482**

482 kinematic bins · 9 experiments · 59 subsets

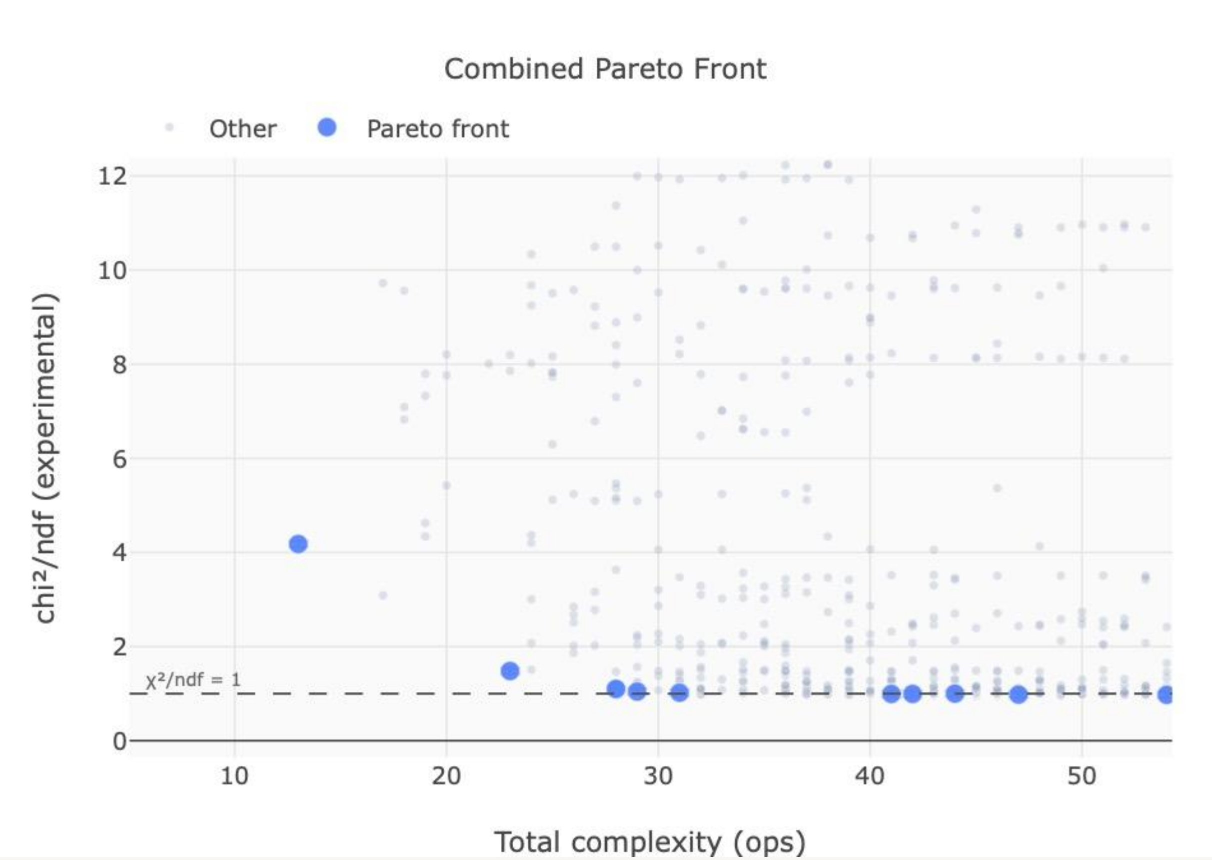
Uniform selections:

- $q_T/Q < 0.2$ — ensure TMD factorization applies
- Υ-resonance veto (9–11 GeV)

Discovering the formula: PySR + Pareto selection

- PySR evolves populations of candidate expressions (crossover, mutation, simplification), returning a **Pareto front**: complexity vs. accuracy.
- SR loss is MSE vs a sample grid from the NN
- SR is restricted to b_T<2.5 GeV^{-1} — beyond which TMD evolution leaves the data essentially unable to constrain f_{NP}
- Per-component candidates are combined; each full f_{NP} is evaluated on the real data.

Selection: minimum complexity with reduced χ^2 compatible with 1, worst experimental subset χ^2 < 3 for points on the Pareto front→ **complexity 31.**



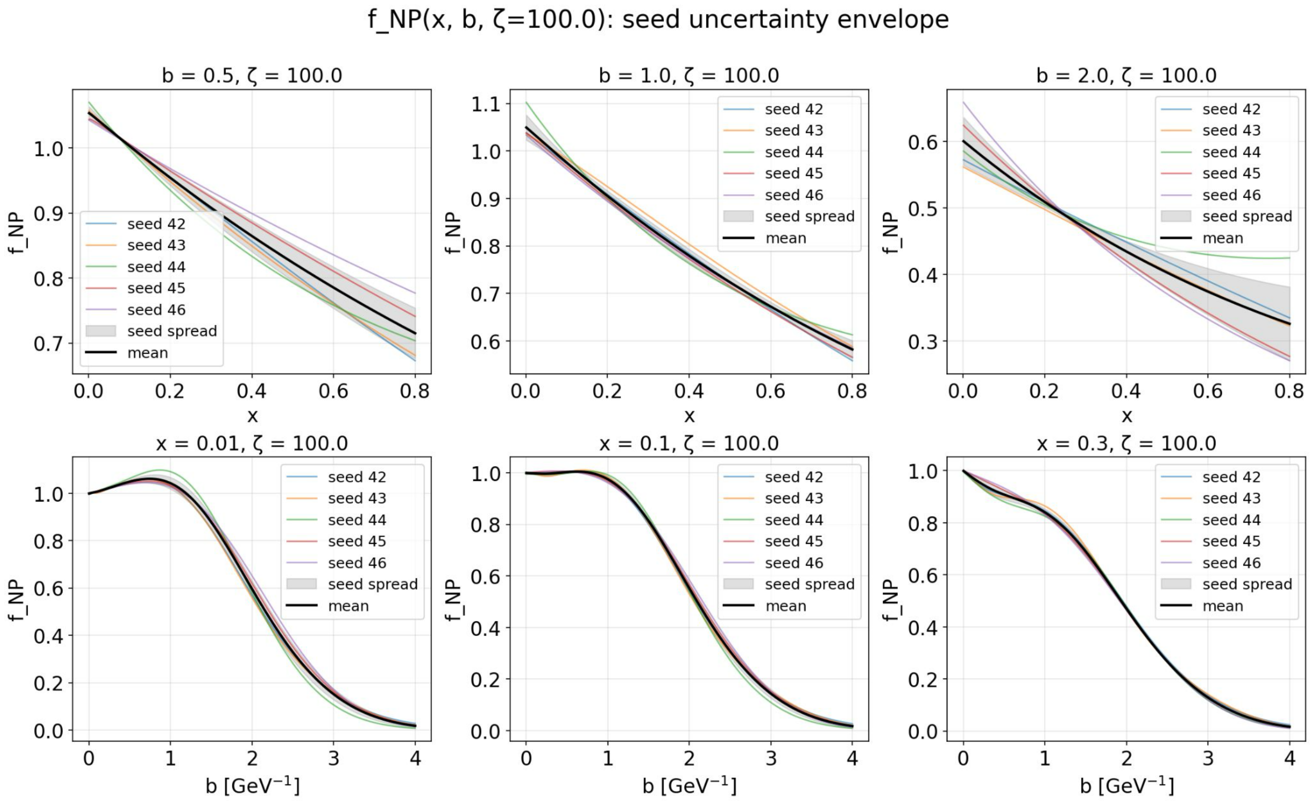
PART III

Results

The formula, its fit quality, and what it means

Robustness across network seeds

- The pipeline was run over multiple independent NN seeds (different random initializations).
- Narrow spread ⇒ the recovered functional form is stable; wide spread ⇒ genuinely distinct solutions of comparable quality.
- The band widens at large b_T ($b_T \gtrsim 4$ GeV^{-1}): there the χ^2 gradient is Sudakov-suppressed, so the network is poorly constrained.



The SR-driven formula

$$f_{NP}(x, b_T, \zeta) = \exp[\, -g_2^{\,2}(b_T)\, b_T^{\,2} \ln(\zeta/Q_0^2)\,] \cdot \exp[\, h_x(x)\, h_b(b_T) + h_{xb}(x, b_T)\,]$$

9 constants · complexity 31 · $\chi^2/\mathrm{ndf} = 1.040$

$h_x(x)$

$$h_x(x) = -0.854 - 1.45\,x$$

$h_b(b_T)$

$$h_b(b_T) = \frac{b_T}{s}\left(0.919 - 0.368\,\sqrt{b_T/s}\right)$$

$h_{xb}(x, b_T)$

$$h_{xb}(x, b_T) = \frac{b_T}{s}\left(0.766 - 0.0567\,(b_T/s - x)^2\right)$$

$g_2(b_T)$

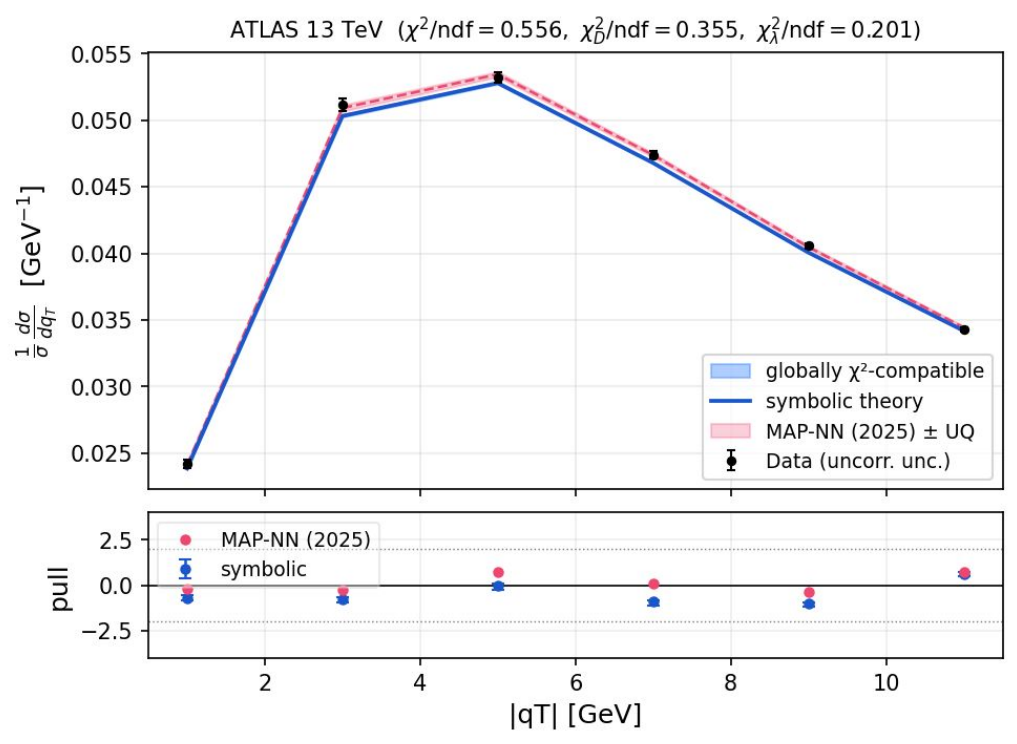
$$g_2(b_T) = 0.169 + 0.0328\,\sqrt{b_T/s + 0.636}$$

Here s is a scale factor of 1 GeV^{-1} to ensure all components are dimensionless

The formula describes the data

Global χ²/ndf = 1.040 ± 0.065 (473 dof) — compatible with 1, and close to the parent NN (0.941).

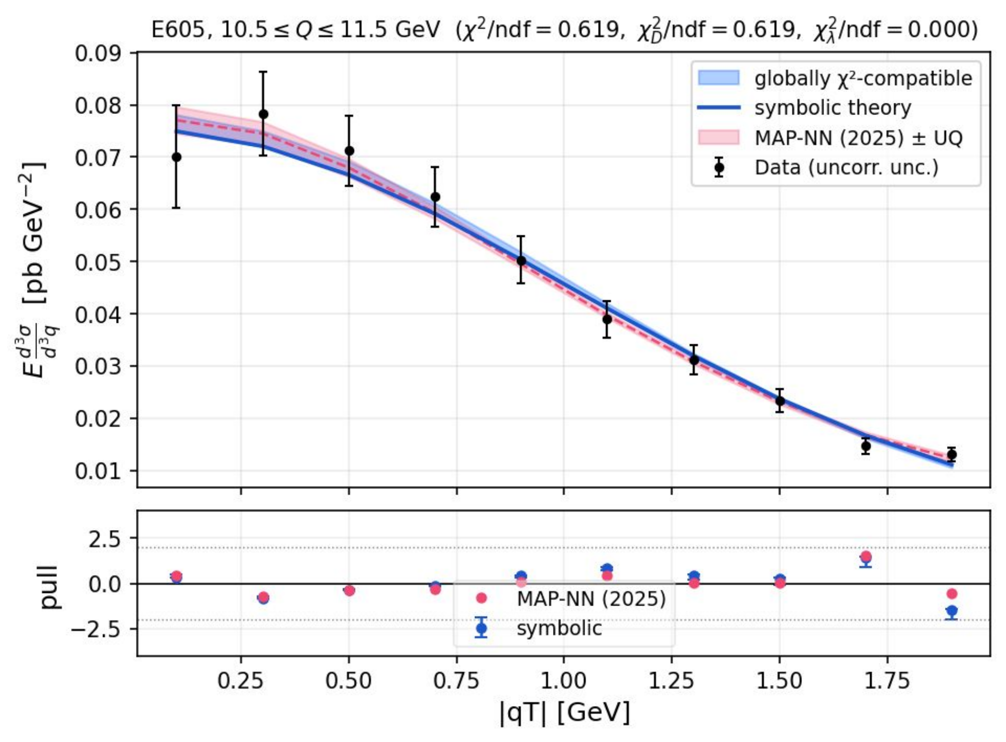
Experiment	Ndat	NN	Symbolic
Fermilab	233	1.057	1.179
RHIC	7	1.070	1.101
Tevatron	71	0.803	0.864
LHCb	21	1.083	1.074
CMS	78	0.420	0.429
ATLAS	72	1.213	1.274
Total	**482**	**0.941**	**1.040**



The formula describes the data

We show good agreement with our reference fit except for a few hard-to-fit subsets in the low-Q range

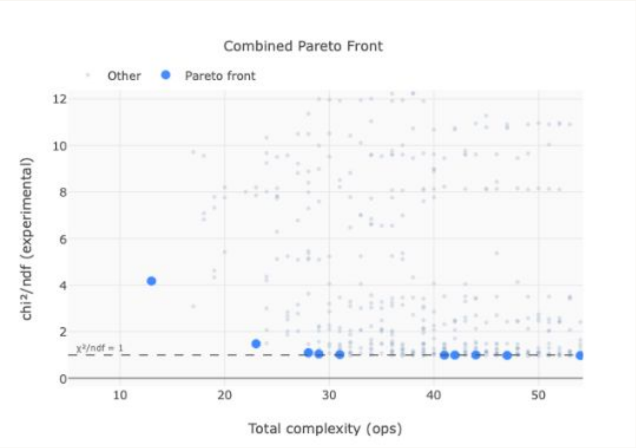
Experiment	Ndat	NN	Symbolic
Fermilab	233	1.057	1.179
RHIC	7	1.070	1.101
Tevatron	71	0.803	0.864
LHCb	21	1.083	1.074
CMS	78	0.420	0.429
ATLAS	72	1.213	1.274
Total	**482**	**0.941**	**1.040**

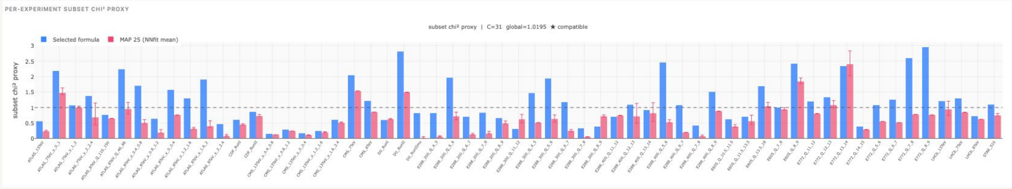


The “Pareto Explorer”

https://wmdataphys.github.io/Results/Symbolic_TMD_PDF/pareto_explorer_global_maxexp_front.html

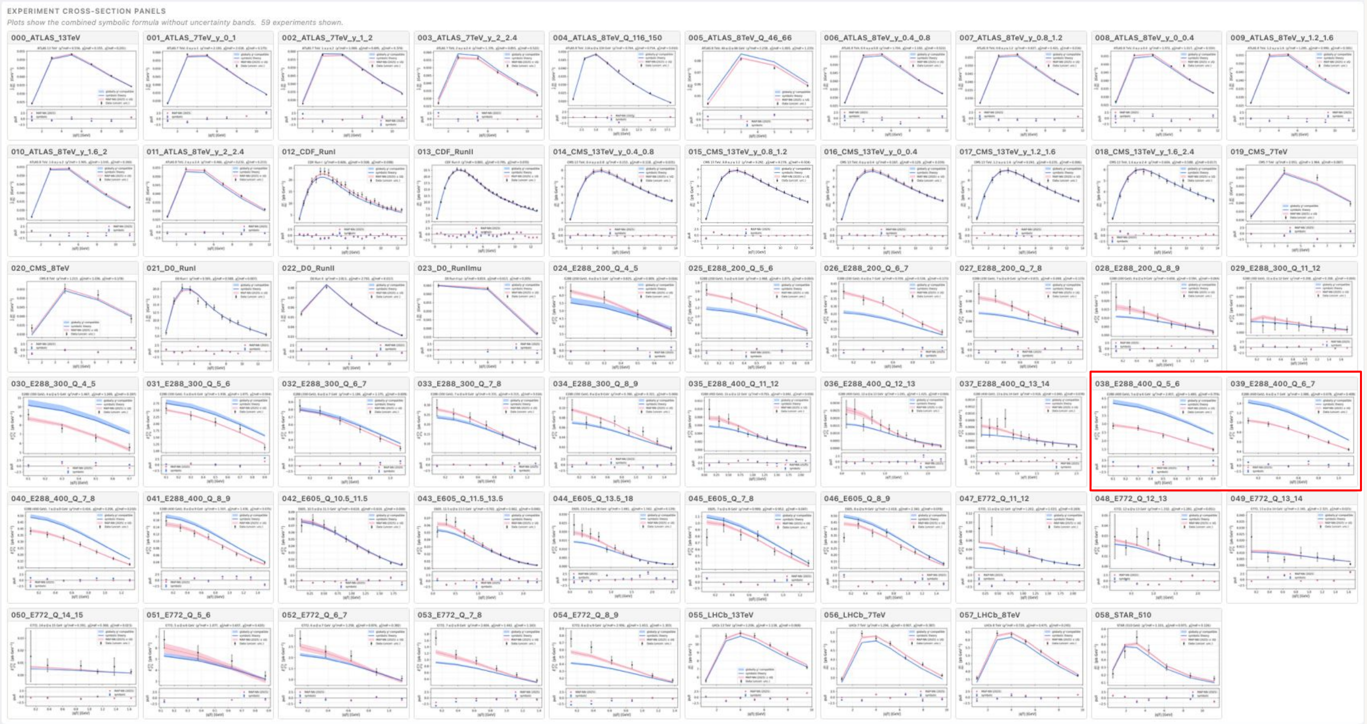
Choose a point in the Pareto front





→ Display fit to experimental cross-sections

Tension in low-Q fixed target



Tensions concentrate in two regions: low-Q fixed-target (E772, E605) and central-rapidity collider bins (ATLAS 7 TeV |y|<1, D0 Run II). All other subsets fall below χ² ≈ 2.5.

What the components mean

$h_x(x)$
A simple linear function, negative across all x — a smooth modulation of the b_T-space suppression amplitude.

$h_b(b_T)$
A √-corrected linear ramp × b_T; vanishes at $b_T = 0$, peaks ≈ 0.85 near $b_T \approx 2.77\ \mathrm{GeV}^{-1}$.

$h_{xb}(x, b_T)$
Contributes at leading order. At $(x, b_T) = (0.1, 1)$: $h_{xb} = +0.72$ vs. backbone $h_x * h_b = -0.55$ — the cross term partially cancels the factorized suppression. Survives an L1 prior actively pushing it to zero.

$g_2(b_T)$
The Collins–Soper coupling grows mildly (0.195 → 0.227); its small-b_T value matches prior global fits.

Data prefer a mild but genuine x–b_T correlation over strict separability.

WILLIAM & MARY | DATA SCIENCE
CHARTERED 1693

Uncertainty Quantification

- Each constant suggested by PySR is treated as a free parameter
- We first vary these parameters independently to find a minimum and maximum value with which the global χ^2 is compatible with 1, and the worst experimental χ^2 is < 3
- From this hypercube, we sample parameters jointly, and keep samples with the same criteria
- This captures constant ranges consistent with global $\chi^2 \approx 1$ within statistical uncertainty. It does not yet capture full theory/experimental uncertainties.

Limitations & what's next

Caveats

- Result is conditioned on the factorized enveloping ansatz; alternatives may give different components at similar fit quality.
- The value of h_{xb} is partly imposed (sparsity (L1,L2 penalties) + capacity cap (4 hidden unit)), not purely data-driven.
- f_{NP} is flavor-universal
- Constrained only for $b_T < 2.5$ GeV^{-1}; the h_b node (crossing 0) at $b_T \approx 6.2$ GeV^{-1} is an extrapolation.

What's next

- Replica-based uncertainty quantification (MCMC).
- Flavor dependence with semi-inclusive DIS data.
- Extension to polarized TMDs.

Conclusions - Data-Driven Symbolic Extraction

- A flexible neural-network representation, distilled by symbolic regression, yields a compact analytic parametrization without prescribing its functional form.
- The extracted non-perturbative TMD achieves a $\chi^2/\mathrm{ndf} = 1.040$ with only 9 constants while matching the reference global fit
- The data prefer a genuine x–b_T correlation beyond purely separable models.
- This establishes symbolic regression as a practical tool for extracting interpretable non-perturbative QCD from experimental data.

Interactive Pareto explorer on GitHub · *Supported by NSF CAREER #2443510*
wmdataphys.github.io

WILLIAM & MARY | DATA SCIENCE
CHARTERED 1693

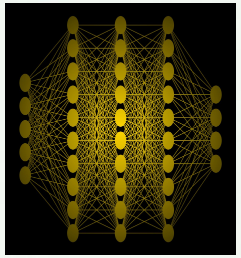
Backup

NangaParbat Convolution

Neural network provides f_{NP}; NangaParbat provides the convolution skeleton to predict cross sections.

1. Neural network: $f_{NP}(x, b_T, \zeta)$

Input: (x_1, x_2, b_T, ζ)

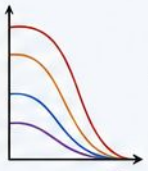


NN outputs f_{NP} at the grid points

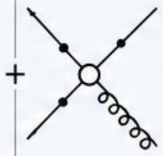
2. NangaParbat convolution skeleton (precomputed)

Computed once using NangaParbat (N^{3}LL accuracy) and stored.

Collinear PDFs



\+ Perturbative coefficients



\+ TMD evolution (Sudakov)

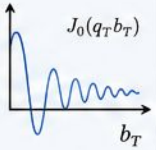
$e^{-S(b_T,\mu,\zeta)}$

Collins–Soper + RG evolution

\+ Kinematics & phase space

Jacobian factors, flux, etc.

\+ b_T integration (Ogata quadrature)



For each contribution to each experimental bin i, store weights and grid points (x_1, x_2, b_T, ζ):

$$\{W_{ik}^{\text{NP-indep}}, x_{1k}, x_{2k}, b_{Tk}, \zeta_k\}_{k=1}^{N_k}$$

These weights are independent of f_{NP} and the NN.

3. Predicted cross sections

At each training step:

- Evaluate NN $f_{NP}(x_1, x_2, b_T, \zeta)$ at stored grid points.
- Multiply by precomputed weights and sum:

$$t_i(\boldsymbol{\theta}) = \sum_k W_{ik}^{\text{NP-indep}} \times f_{NP}^{NN}(x_{1k}, x_{2k}, b_{Tk}, \zeta_k; \boldsymbol{\theta})$$

- Obtain predicted cross section $t_i(\boldsymbol{\theta})$ for each data bin i.

4. Compare to data (χ^2)

Compare predictions with experimental measurements y_i using the experimental covariance matrix (or equivalent nuisance profiling).

$$\chi^2(\boldsymbol{\theta}) = (t(\boldsymbol{\theta}) - y)^T V^{-1} (t(\boldsymbol{\theta}) - y)$$

PyTorch autograd computes $\nabla_\theta \chi^2$ and updates NN parameters $\boldsymbol{\theta}$.

Key points

- **NangaParbat** provides the **theory "skeleton"**: all perturbative ingredients, evolution, PDFs, kinematics, and the b_T-space convolution are precomputed and stored.
- **The NN supplies only the non-perturbative function** f_{NP} on the fixed grid.
- **Predictions are confronted directly with the original experimental data** (no transformation of the data).
- **Precomputing the skeleton** avoids propagating gradients through the NangaParbat code and enables efficient training with PyTorch autograd.

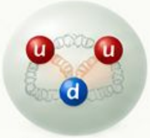
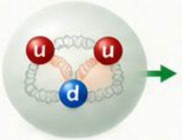
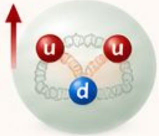
In short

NN: $f_{NP}(x, b_T, \zeta)$ → NangaParbat skeleton (fixed weights) → Predicted cross sections → compare to data

NN is trained on Drell–Yan data from 9 experiments (fixed-target, Tevatron, RHIC, LHC) at N^{3}LL accuracy within the *NangaParbat* framework.

TMDs

TMDs depend on the spin of both the nucleon and the quark (at leading twist, 8 quark TMDs)

Nucleon (spin)	Quark (spin)	Example TMD	Physical meaning	Symbol
 **Unpolarized** No preferred spin direction. Valence content (proton): u u d. Sea quarks ($\bar{u}$, $\bar{d}$, $s\ \bar{s}$, …) and gluons (g)	**Unpolarized** No preferred spin direction	**Unpolarized TMD PDF** (number density)	Probability to find an unpolarized quark with x and $k_\perp$	f_1
	Transversely polarized Spin perpendicular to momentum	**Boer–Mulders function** $h_1^\perp$ (T-odd, chiral-odd)	Correlation between quark spin and $k_\perp$ in an unpolarized nucleon	$h_1^\perp$
 **Longitudinally polarized** Spin along momentum. Valence content (proton): u u d. Sea quarks ($\bar{u}$, $\bar{d}$, $s\ \bar{s}$, …) and gluons (g)	**Unpolarized** No preferred spin direction	**Helicity (or g_{1L})** (chiral-even)	Distribution of unpolarized quarks in a longitudinally polarized nucleon	g_{1L}
	Transversely polarized Spin perpendicular to momentum	**Worm-gear** $h_{1L}^\perp$ (chiral-odd)	Correlation between quark transverse spin and $k_\perp$ in a longitudinally polarized nucleon	$h_{1L}^\perp$
 **Transversely polarized** Spin perpendicular to momentum. Valence content (proton): u u d. Sea quarks ($\bar{u}$, $\bar{d}$, $s\ \bar{s}$, …) and gluons (g)	**Unpolarized** No preferred spin direction	**Sivers function** $f_{1T}^\perp$ (T-odd)	Correlation between nucleon spin and quark $k_\perp$	$f_{1T}^\perp$
	Longitudinally polarized Spin along momentum	**Worm-gear** g_{1T} (chiral-even)	Correlation between quark helicity and $k_\perp$ in a transversely polarized nucleon	g_{1T}
	Transversely polarized Spin perpendicular to momentum	**Transversity** h_1 (chiral-odd)	Distribution of transversely polarized quarks in a transversely polarized nucleon	h_1
	Transversely polarized Spin perpendicular to momentum	**Pretzelosity** $h_{1T}^\perp$ (chiral-odd)	Quadrupole-like (pretzel-shaped) correlations in a transversely polarized nucleon	$h_{1T}^\perp$

Here, x is the longitudinal momentum fraction of the quark and $k_\perp$ is its intrinsic transverse momentum.

u up quark (u)
d down quark (d)
Unpolarized
Longitudinal polarization (spin along momentum)
Transverse polarization (spin perpendicular to momentum)
Gluons (force carriers) and sea quarks