A factorisation-aware matrix element emulator

[arXiv:2107.06625]

Henry Truong with Daniel Maître

ACAT 2021

November 29 2021

Institute for Particle Physics Phenomenology
Institute for Data Science
Durham University

Table of contents

Introduction

Motivation

- High multiplicity matrix elements are computationally expensive to evaluate.

Motivation

- High multiplicity matrix elements are computationally expensive to evaluate.
- High energy collider experiments are becoming increasingly more precise, meaning theoretical predictions need to improve as well.

Motivation

- High multiplicity matrix elements are computationally expensive to evaluate.
- High energy collider experiments are becoming increasingly more precise, meaning theoretical predictions need to improve as well.
- Successfully emulating matrix elements will provide a fast and accurate alternative to more traditional matrix element providers.

Electron-positron scattering

- We investigate using a neural network model to emulate tree-level matrix elements for $e^+e^- \to Z/\gamma \to q\bar{q} + ng$, up to five jets.

Electron-positron scattering

- We investigate using a neural network model to emulate tree-level matrix elements for $e^+e^- \to Z/\gamma \to q\bar{q} + ng$, up to five jets.
- Matrix elements are plagued with singularities arising because of infrared divergences.

Electron-positron scattering

- We investigate using a neural network model to emulate tree-level matrix elements for $e^+e^- \to Z/\gamma \to q\bar{q} + ng$, up to five jets.
- Matrix elements are plagued with singularities arising because of infrared divergences.
- when gluons are *collinear* to the quark or anti-quark, $s_{qg} \to 0$.

Electron-positron scattering

- We investigate using a neural network model to emulate tree-level matrix elements for $e^+e^- \to Z/\gamma \to q\bar{q} + ng$, up to five jets.
- Matrix elements are plagued with singularities arising because of infrared divergences.
- when gluons are *collinear* to the quark or anti-quark, $s_{qg} \to 0$.
- when gluons are *soft*, $E_g \to 0$.

Electron-positron scattering

- We investigate using a neural network model to emulate tree-level matrix elements for $e^+e^- \to Z/\gamma \to q\bar{q} + ng$, up to five jets.
- Matrix elements are plagued with singularities arising because of infrared divergences.
- when gluons are *collinear* to the quark or anti-quark, $s_{qg} \to 0$.
- when gluons are *soft*, $E_g \to 0$.
- Singularities make it difficult to model over the phase-space effectively.

Method

Dipole factorisation formula

Catani and Seymour introduced [1] universal dipoles that smoothly interpolates between the soft and collinear limits.

[1] S. Catani and M.H. Seymour. “A General algorithm for calculating jet cross-sections in NLO QCD”. In: Nucl. Phys. B 485 (1997), pp. 291–419.

Dipole factorisation formula

Catani and Seymour introduced [1] universal dipoles that smoothly interpolates between the soft and collinear limits. We use these dipoles to factorise out the IR singularity structure from matrix elements

$$|\mathcal{M}_{n+1}|^2 \to |\mathcal{M}_n|^2 \otimes \mathbf{V}_{ij,k}\,, \tag{1}$$

where all divergences are isolated in the process independent factor $\mathbf{V}_{ij,k}$.

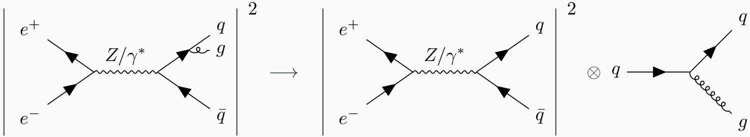


Figure 1: Schematic of dipole factorisation.

[1] S. Catani and M.H. Seymour. "A General algorithm for calculating jet cross-sections in NLO QCD". In: Nucl. Phys. B 485 (1997), pp. 291–419.

Ansatz

We use the dipole factorisation formula to build an ansatz of the colour and helicity summed $n+1$-body matrix element

$$\langle|\mathcal{M}_{n+1}|^2\rangle = \sum_{\{ijk\}} C_{ijk} D_{ij,k} \,, \quad \text{where} \quad D_{ij,k} = \frac{\langle V_{ij,k}\rangle}{s_{ij}} \,. \tag{2}$$

Ansatz

We use the dipole factorisation formula to build an ansatz of the colour and helicity summed $n+1$-body matrix element

$$\langle|\mathcal{M}_{n+1}|^2\rangle = \sum_{\{ijk\}} C_{ijk} D_{ij,k}\,, \quad \text{where} \quad D_{ij,k} = \frac{\langle V_{ij,k}\rangle}{s_{ij}}\,. \tag{2}$$

C_{ijk} are the coefficients we fit using the neural network.

Ansatz

We use the dipole factorisation formula to build an ansatz of the colour and helicity summed $n+1$-body matrix element

$$\langle |\mathcal{M}_{n+1}|^2 \rangle = \sum_{\{ijk\}} C_{ijk} D_{ij,k}\,, \quad \text{where} \quad D_{ij,k} = \frac{\langle V_{ij,k} \rangle}{s_{ij}}\,. \qquad (2)$$

C_{ijk} are the coefficients we fit using the neural network.

The sum over $\{ijk\}$ denotes the sum over relevant permutations of external final state particles.

Neural network architecture

The basis of our neural network model is a dense neural network with eight hidden layers.

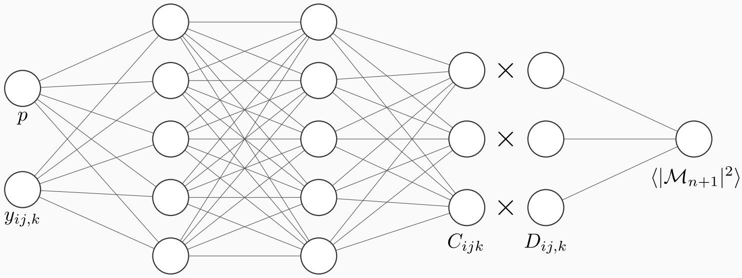


Figure 2: Schematic diagram of our neural network architecture.

Inputs of neural network

Direct inputs to network

- Phase-space points: $p^{(i)} = [E^{(i)}, p_x^{(i)}, p_y^{(i)}, p_z^{(i)}]$

Inputs of neural network

Direct inputs to network

- Phase-space points: $p^{(i)} = [E^{(i)}, p_x^{(i)}, p_y^{(i)}, p_z^{(i)}]$
- Recoil factors: $y_{ij,k} = \dfrac{p_i p_j}{p_i p_j + p_j p_k + p_i p_k}$

Inputs of neural network

Direct inputs to network

- Phase-space points: $p^{(i)} = [E^{(i)}, p_x^{(i)}, p_y^{(i)}, p_z^{(i)}]$
- Recoil factors: $y_{ij,k} = \dfrac{p_i p_j}{p_i p_j + p_j p_k + p_i p_k}$

Global phase-space cuts are applied according to $y_{ij} \geq y_{\text{cut}}$, where we explore three values of $y_{\text{cut}} = [0.01, 0.001, 0.0001]$.

Inputs of neural network

Indirect inputs to network

These inputs are introduced into our model via our custom loss function.

Inputs of neural network

Indirect inputs to network

These inputs are introduced into our model via our custom loss function.

- Spin-averaged dipoles $D_{ij,k} = \frac{\langle V_{ij,k} \rangle}{s_{ij}}$

Inputs of neural network

Indirect inputs to network

These inputs are introduced into our model via our custom loss function.

- Spin-averaged dipoles $D_{ij,k} = \dfrac{\langle V_{ij,k} \rangle}{s_{ij}}$
- Averaging over spin means that we have lost information about the spin-correlation in $g \to gg$ splitting.

Inputs of neural network

Indirect inputs to network

These inputs are introduced into our model via our custom loss function.

- Spin-averaged dipoles $D_{ij,k} = \frac{\langle V_{ij,k} \rangle}{s_{ij}}$
- Averaging over spin means that we have lost information about the spin-correlation in $g \to gg$ splitting.
  - Introduce $S_{ij} \sin(2\phi_{ij}) + C_{ij} \cos(2\phi_{ij})$ in the ansatz for pairs of gluons.
  - ϕ_{ij} is the azimuthal angle of the decay particles in the plane perpendicular to the parent particle momentum.

Outputs of neural network

The output of the neural network is the colour and helicity summed matrix element

$$\langle|\mathcal{M}_{n+1}|^2\rangle = \sum_{\{ijk\}} C_{ijk} D_{ij,k} \,.$$

Outputs of neural network

The output of the neural network is the colour and helicity summed matrix element

$$\langle |\mathcal{M}_{n+1}|^2 \rangle = \sum_{\{ijk\}} C_{ijk} D_{ij,k} \,.$$

During training, the predictions are compared against the target value of the matrix element which we scale according to

$$y = \text{arcsinh}\left(\frac{\langle |\mathcal{M}_{n+1}|^2 \rangle}{S_{\text{pred}}} \right). \tag{3}$$

Custom loss function

We use the mean squared error loss function along with a regularisation term

$$
\begin{aligned}
L &= L_{\text{MSE}} + L_{\text{pen}} \\
L &= \frac{1}{N}\sum_{n=1}^{N}(y - f(\mathrm{x};\theta))^2 + J\sum_i \frac{D_i^{-2}}{\sum_j D_j^{-2}}|C_i D_i|.
\end{aligned}
$$

Custom loss function

We use the mean squared error loss function along with a regularisation term

$$L = L_{\text{MSE}} + L_{\text{pen}}$$

$$L = \frac{1}{N}\sum_{n=1}^{N}(y - f(\mathbf{x};\theta))^2 + J\sum_i \frac{D_i^{-2}}{\sum_j D_j^{-2}}|C_i D_i|.$$

We promote the network to learn about the universal factorisation property in matrix elements, hence becoming *factorisation-aware*.

Results

Results

We present results for our matrix element emulator:

1. Compare results obtained with our method to those in a previous work by Aylett-Bullock and Badger [2].

[2] Simon Badger and Joseph Bullock. "Using neural networks for efficient evaluation of high multiplicity scattering amplitudes". In: JHEP 06 (2020), p. 114.

Results

We present results for our matrix element emulator:

1. Compare results obtained with our method to those in a previous work by Aylett-Bullock and Badger [2].
2. Scaling performance by expanding the network size and number of training samples.

[2] Simon Badger and Joseph Bullock. "Using neural networks for efficient evaluation of high multiplicity scattering amplitudes". In: JHEP 06 (2020), p. 114.

Results

We present results for our matrix element emulator:

1. Compare results obtained with our method to those in a previous work by Aylett-Bullock and Badger [2].
2. Scaling performance by expanding the network size and number of training samples.
3. Generalisation to unseen regions of phase-space by predicting on random phase-space trajectories.

[2] Simon Badger and Joseph Bullock. "Using neural networks for efficient evaluation of high multiplicity scattering amplitudes". In: JHEP 06 (2020), p. 114.

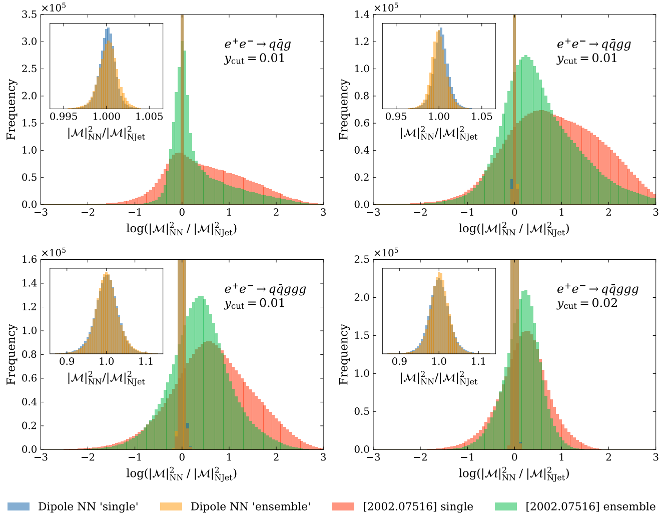
Results

We present results for our matrix element emulator:

1. Compare results obtained with our method to those in a previous work by Aylett-Bullock and Badger [2].
2. Scaling performance by expanding the network size and number of training samples.
3. Generalisation to unseen regions of phase-space by predicting on random phase-space trajectories.
4. Compare MC statistical error to NN accuracy.

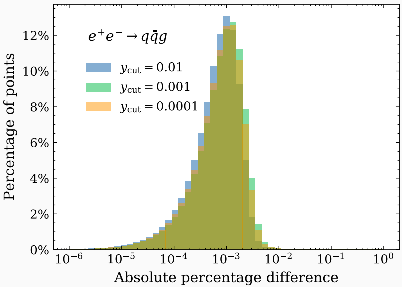
[2] Simon Badger and Joseph Bullock. "Using neural networks for efficient evaluation of high multiplicity scattering amplitudes". In: JHEP 06 (2020), p. 114.

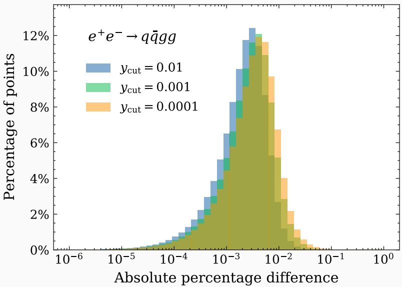
Comparison with n3jet[2]

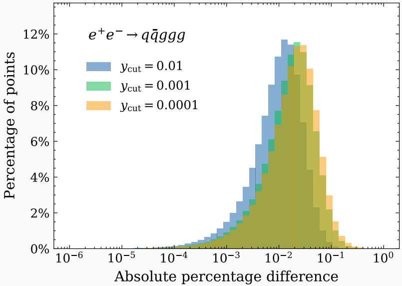


[2] Simon Badger and Joseph Bullock. "Using neural networks for efficient evaluation of high multiplicity scattering amplitudes". In: JHEP 06 (2020), p. 114.

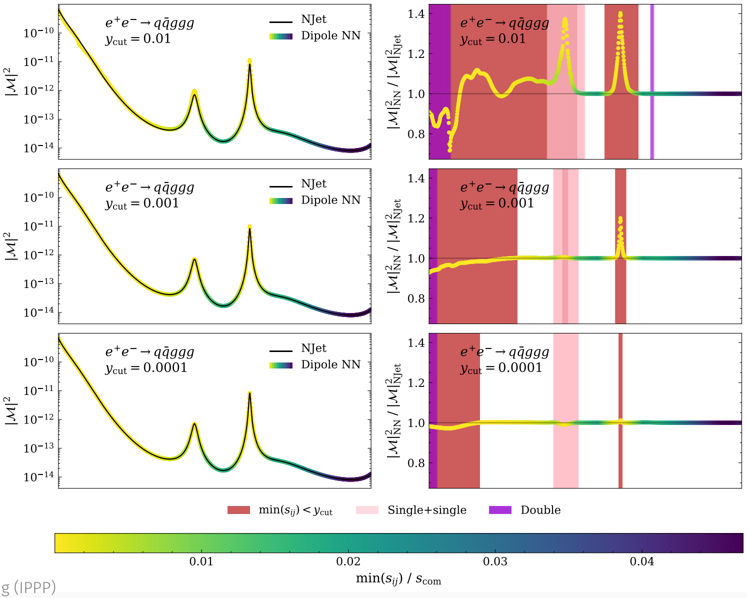
Error distributions



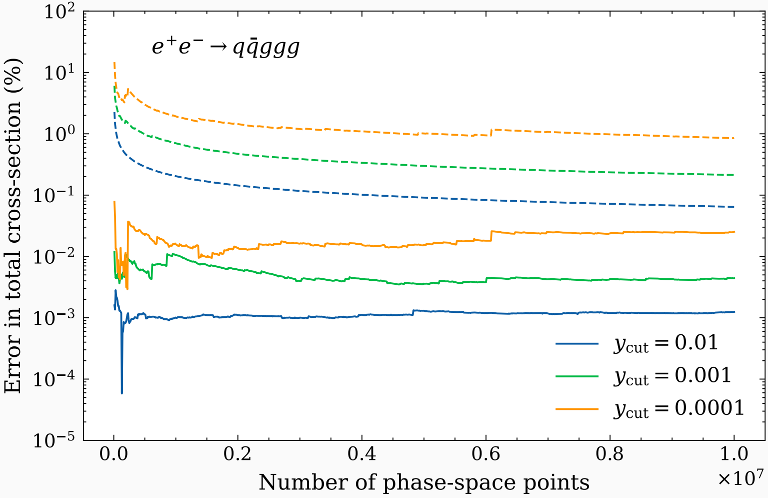




Random phase-space trajectory for 5 jet case



Comparison of errors



Conclusion

Conclusion

We showed that it is possible to build a neural network emulator to accurately model matrix elements by incorporating our physics knowledge on infrared divergences.

Conclusion

We showed that it is possible to build a neural network emulator to accurately model matrix elements by incorporating our physics knowledge on infrared divergences.

With the model trained we can use it to augment datasets by providing extra statistics.

Conclusion

We showed that it is possible to build a neural network emulator to accurately model matrix elements by incorporating our physics knowledge on infrared divergences.

With the model trained we can use it to augment datasets by providing extra statistics.

`github.com/htruong0/fame`