Improving Λ Signal Extraction with Domain Adaptation via Normalizing Flows

Rowan Kelleher [1] Anselm Vossen [1 2] Matthew McEneaney [1]

[1]Duke University [2]Jefferson Lab

Problem Statement

Studying the Λ hyperon channel at CLAS12 can provide helpful insight into the spin structure of the proton and hence the strong force. However, Λ hyperons decay before reaching the detector, meaning studies must first reconstruct the Λs using the decay products, often a proton π^- pair. **Machine learning can help classify which proton π^- pairs decayed from a Λ (signal)** and which come from other processes (background), but suitable simulation data is necessary to train such a classifier. Monte-Carlo (MC) simulation is used for this purpose, **but differences exist between the MC and data distributions**, making hurting classifier performance.

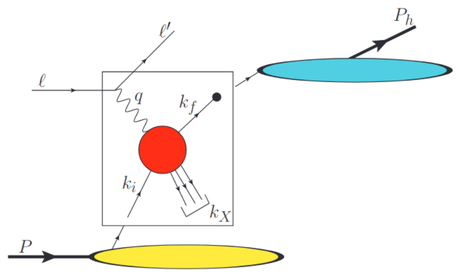


Figure 1. SIDIS Process [1]

Idea

- Use Normalizing flows (NF) to try to **improve classification** of Λ hyperons **through domain adaptation**.
- **Data is transformed to the MC domain**, allowing a classifier trained on MC to more accurately classify data events.
- Need probability density function (PDF) of base distribution: use normal distribution
- **Train two separate models**: data and MC
- Data is passed forward through the trained data model, transforming it into normal domain, and then passed backwards through the MC model, transforming it to the MC domain.

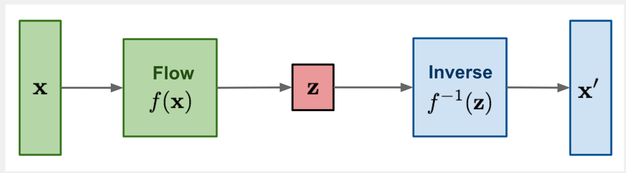


Figure 2. The data is transformed from an initial state x to a normalized state z then back through an inverse transformation to x' which is the data in the MC domain[4]

References

[1] M. Boglione, M. Diefenthaler, S. Dolan, L. Gamberg, W. Melnitchouk, D. Pitonyak, A. Prokudin, N. Sato, and Z. Scalyer, 2022.

[2] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio, 2017.

[3] Ivan Kobyzev, Simon J.D. Prince, and Marcus A. Brubaker. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(11):3964–3979, nov 2021.

[4] Phillip Lippe, 2022.

[5] M. McEneaney and A. Vossen. *Journal of Instrumentation*, 18(06):P06002, jun 2023.

Implementation and Model

- NFs are a series of bijective (and hence invertible) functions that **transform a base distribution to a more complex target distribution** via change of variables:

$$p_X(x) = p_Z(z) \left| det \left(\frac{\partial g(z)}{\partial z^T} \right) \right|^{-1}$$

- The **RealNVP** architecture [2] was used for this study which utilizes a series of affine coupling layers to improve computation efficiency.
- We split the inputs into two components and parameterize the output of the first component by the second input (but leaving the output of the second input the same as the input). This allows for quick log-likelihood calculations.
- Train by **minimizing the negative log-likelihood** using the change of variables formula.

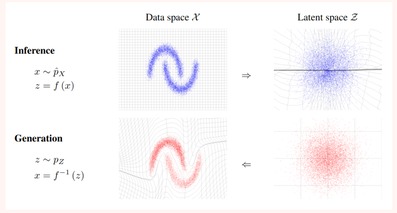


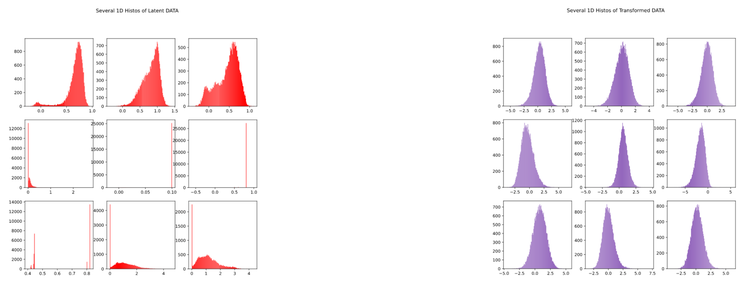
Figure 3. Flowing forward from initial distribution (data) to a base normal distribution, then backwards to the target (MC) distribution. Diagram from [2]

Data

- The data used in this study was taken from the fall 2018 run at the CLAS12 detector.
- MC simulation was produced with the same configuration as the data using a Pepsi-Lund based event generator.
- The graph neural network used in [5] was used to extract latent representations of the data and MC in a fixed dimension of 71.

Results

- First need to verify ability of NF models to learn PDFs of data and MC



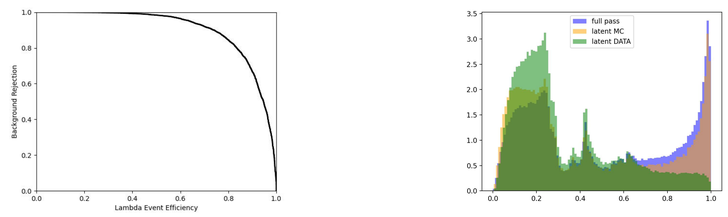
(a) Latent Data (b) Transformed Data

Figure 4. A few dimensions are shown for the latent data (a) and the normalized data (b) (passed through the first NF model, but not the second).

- Figure 4 shows that the **transformed data appears normal, but not perfect** as some are skewed and have non-zero means.

Classifier

- 15 layers, input dim of 71, output dim of 2 (confidence of event being signal)
- Reached an **accuracy of 82%** on MC. The receiver operating characteristic (ROC) curve is plotted in figure 5; the **area under the curve was 0.90**.



(a) ROC Curve (b) Classifier Output

Figure 5. The Classifier's ROC Curve (a). The classifier output for the data, MC, and transformed data are shown in (b).

- The mass spectrum signal peak was fit using a **crystal ball fit over a quadratic background**.
- The **figure of merit** (FOM = $N_{signal}/\sqrt{N_{tot}}$) and **purity** ($N_{signal}/N_{tot}$) were calculated for 20 different cuts on the roc curve to illustrate performance as a function of the cut (Figure 6).

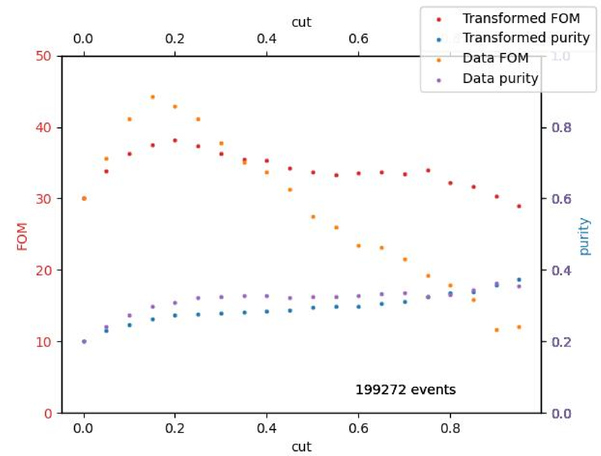


Figure 6. Caption

- **FOM remains flatter** for the transformed data compared to the latent data which falls sharply.
- Figure 5b shows that the **classifier output matches much better** between the transformed data and MC.

Conclusions

- Domain adaptation via NFs **improves generalizability** as FOM depends less on cut
- Different NF models/configurations can be investigated to improve signal extraction further

rck32@duke.edu