Augmented Recursive Neural Networks for Jet Physics

Team Voyagers: Yurui Mu, Sudhir Nallam, Xiaoyu Wang, Wenqing Zhu

NYU

NYU Data Science

Problem Statement

'Jets' are the most common structures seen in collisions at Large Hadron Collider (LHC). These jets are produced from the fragmentation and hadronization of quarks and gluons as described by quantum chromodynamics(QCD). In paper [1], it leverages techniques of sentences' semantic structure to represent the jet substructure as a "jet embedding", which maps a set of 4-momenta (vector representing the particle) into R^q. Yet there is a large background from jets produced by more mundane QCD processes which complicates with the process of source determination. In our project we propose a few methods to augment the model proposed in [1] more robust to systematic uncertainties.

Method

The central idea of our modeling approach is recursive embedding. Individual jets, particles are topologically structured as a binary tree t_j. The embedding of h_k^{jets} of node k is recursively defined as following:

$$h_k^{jet} = \begin{cases} u_k & \text{if } k \text{ is a leaf} \\ \sigma\left(W_h \begin{Bmatrix} h_{KL}^{jet} \\ h_{KR}^{jet} \\ u_k \end{Bmatrix} + b_h \right) & \text{otherwise} \end{cases}$$

$$o_k = \begin{cases} v_{i(k)} & \text{if } k \text{ is a leaf} \\ o_{k_L} + o_{k_R} & \text{otherwise} \end{cases}$$

To make the classification model pivotal to nuisance parameters, we add adversarial loss:

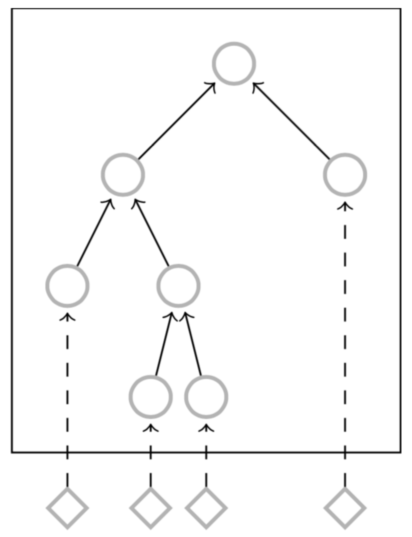
$$E_\lambda(\theta_f, \theta_r) = \mathcal{L}_f(\theta_f) - \lambda \mathcal{L}_r(\theta_f, \theta_r)$$

$$\hat{\theta}_f, \hat{\theta}_r = \arg \min_{\theta_f} \max_{\theta_r} E(\theta_f, \theta_r).$$

Data

In this project, we used jets level data with two binary tree topology structures: a)Sequential recombination jet clustering algorithm, referred to as k_t b)Simple sequential descending sorting in p_T, referred to as desc-p_T In each of our training and testing datasets, we have $10,000$ data entries, each contains 4-momenta vectors of particles, and features of jets like mass, η, p_T, energy and tree edges. The label $(y = 1)$ corresponds to a hadronically decaying W boson with $200 < p_T < 500 GeV$, while $(y = 0)$ corresponds to QCD jet with the same range of p_T. Moreover, to build models with robustness, we used the pileup data with four pileup levels: 25, 40, 50, 60 under k_t architecture.

Model Details



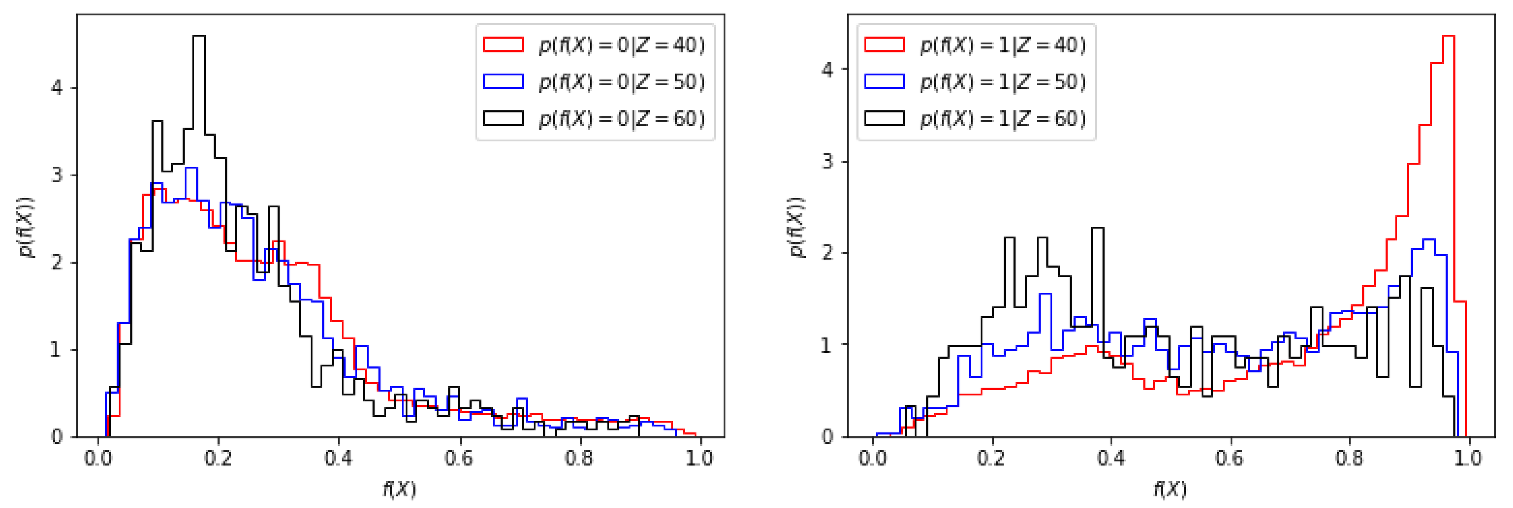
While jets recombination from a physics point of view is a far more complex process, we hold a suspicion that using such activation function is not expressive enough and further proposed adding 1 by 1 convolution layer to increase the level of complexity of nonlinearity. Furthermore, we seek to provide robustness in such classification tasks to pileup interactions [1]. This is achieved through pitting decision function f against an adversary model. Adding this adversarial loss amounts to a regularization term that further eliminate correlation between prediction and pileup interaction levels.

Results

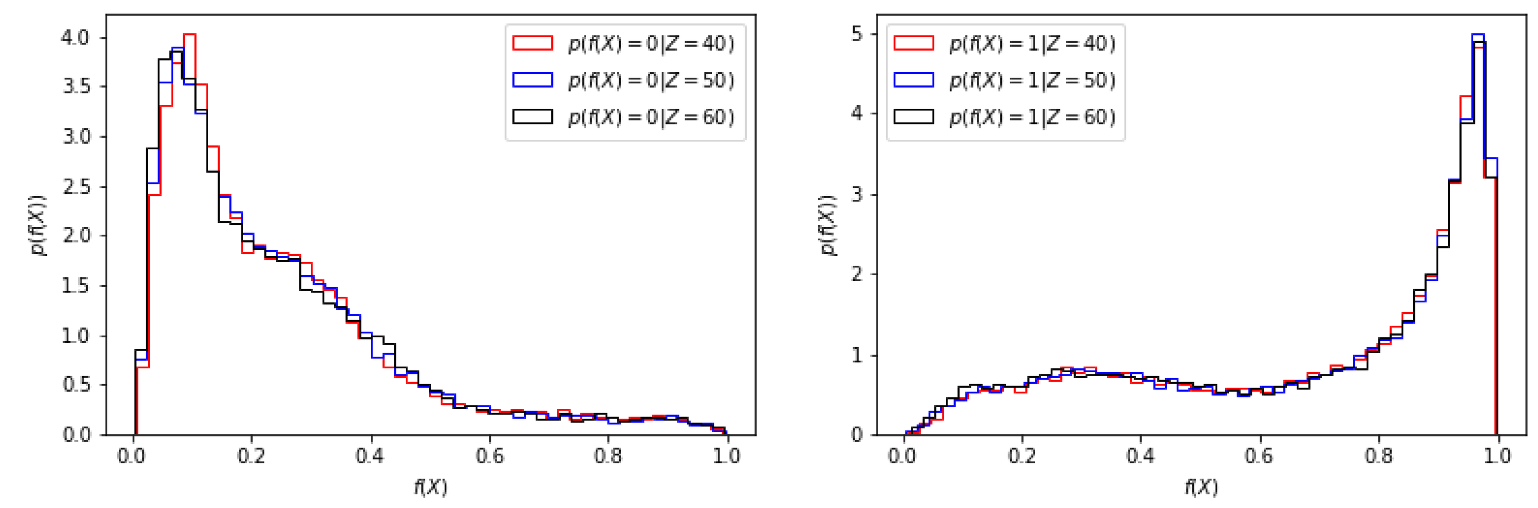
From the table results, we can see that using 1 by 1 convolutional layer as activation function has improved test accuracy on k_t (non-pileup) data. However we didn't witness a significant improvement on desc-p_T (non-pileup) data.

Table 1: Summary of Jet classification Performance

Input	Architectures	Model	ROC AUC	$R_{\epsilon=50\%}$
particles	k_t	RNN	0.9009	58.75
particles	k_t	RNN + 1x1 Convolution	0.9156	56.45
particles	desc-pt	RNN	0.9202	70.71
particles	desc-pt	RNN + 1x1 Convolution	0.9026	41.06



From the plot we can see that previously, the training of Recursive Neural Network without adversarial loss results in having three different distribution curves for each level of pileup interaction.



After our implementation of co-training with adversarial loss, we see that the three probability distribution curves are overlapping, showing that each decision function is invariant of the levels of pileup, a stable and robust classification model is indeed formed.

Conclusion

In this work, we modified the complexity level of non-linearity and witness an increase on performance. Moreover, we trained a stable decision function robust to variate levels of pileup interactions. Yet there are still more work we can do in the future:

a. As the recombination happens at last step is often considered more important, one possible future direction is to change the formation of root node: activation function and weight, and explore the resulting effects.

b. In practice a both optimal and pivotal classifier may not exist. In this work we used a hyper-parameter λ and set it equal to 1 to control trade-off between performance optimality and its independence with pileup interaction levels. A future work could experiment on the λ values to reach optimum balance.

Source Code

The source code and compiled executables for this project are available at `https://github.com/NYU-CDS-Capstone-Project/Voyagers`

References

[1] G. Louppe, K. Cho, C. Becot and K. Cranmer, QCD-Aware Recursive Neural Networks for Jet Physics, 1702.00748.

[2] Gilles Louppe, Michael Kagan, and Kyle Cranmer. Learning to Pivot with Adversarial Networks. 2016, 1611.01046.

Acknowledgement

We would like to thank Professor Kyle Cranmer, Johann Brehmer and Isaac Henrion for their patient guidance, and Professor Michael Gill and Professor Claudio Silva for instructions and efforts on the Capstone course.