Domain Adaptation via Histogram Loss Component

Hector Casto Noguez

June 4, 2021

IRIS-HEP Fellowship Proposal: June-August 2021
Mentors: Dr. Samuel May (Boston University), Prof. Indara Suarez (Boston University)
Project Keywords: Innovative Algorithms, Domain Adaptation, Systematic Uncertainties

1 Proposal

In recent years, machine learning (ML) algorithms have shown tremendous promise in improving the sensitivity of results delivered by LHC experiments. For instance, heavy-flavour jet identification algorithms like DeepCSV [1, 2] have utilized deep neural networks (DNNs) to significantly increase the performance of heavy-flavour jet identification. The DeepCSV algorithm has proved to be more sensitive and efficient than other identification techniques, increasing the relative efficiency by as much as 15% in conditions with low mis-identification probabilities. Other notable applications of ML in high energy physics (HEP) include the search for for exotic particles with deep learning (DL) [3], b jet energy regression [4], and tagging hadronically decaying boosted objects with DNNs [5]. These algorithms also prove to be useful in physics measurements; for example, one of the DNNs described in Ref. [5] was utilized in the CMS search for the Higgs boson decaying to charm quarks [6].

While these algorithms are applied on data collected by LHC experiments, it is often preferable or even necessary, in the case of fully supervised algorithms, to train them on simulation. For example, training a supervised jet energy regression algorithm requires knowledge of the "true" energy of the jet: this is accessible in simulation, but not in actual data. Developing ML algorithms on simulation can present challenges. If the input features to the algorithm are not perfectly modeled by simulation, the output distributions may also be different between data and simulation. In a HEP analysis, these differences would typically be accounted for through a systematic uncertainty which covers the level of disagreement between data and simulation.

As we move towards the HL-LHC, luminosity increases to unprecedented levels and the statistical uncertainty of measurements will decrease. With smaller statistical uncertainties, systematic uncertainties will play a greater role in the sensitivity of a measurement and minimizing them becomes more important to achieving optimal sensitivity.

The problem of the training sample not being entirely representative of the sample where the algorithm will be applied, is not unique to HEP. More generally, this is referred to as a *domain shift* between the *source domain* and the *target domain*. The field of *domain adaptation* is the study of algorithms which are robust to the domain shift. Solutions to domain adaptation usually fall into one of two categories: pre-training solutions or solutions applied during training. The former focuses on transforming features in the source domain to be more representative of the target domain, while the latter typically adds a component to the loss function which rewards invariance between the source and target domains. Pre-training solutions may involve ML, such as Cycle-Consistent Adversarial Networks [7], but in the context of HEP this often takes the form of deriving corrections for the simulation and is typically done for any HEP analysis, regardless of whether ML methods are even used.

Solutions applied during training are less common in HEP, but have been successfully applied: a CMS search for new long-lived particles (LLPs) decaying to jets [8] utilizes a DNN to identify jets originating from an LLP. A gradient reversal layer [9] is included during training to minimize the DNN's ability to distinguish between events from data and events from simulation. The gradient reversal layer discourages the DNN from learning features which allow it to distinguish between data and simulation, with the hope that doing so will improve the agreement of the output distributions between data and simulation. Indeed, it was found to improve the agreement, as shown in Fig. 1, and is associated with only a small decrease in the algorithm's performance on the original task, identifying jets originating from LLPs.

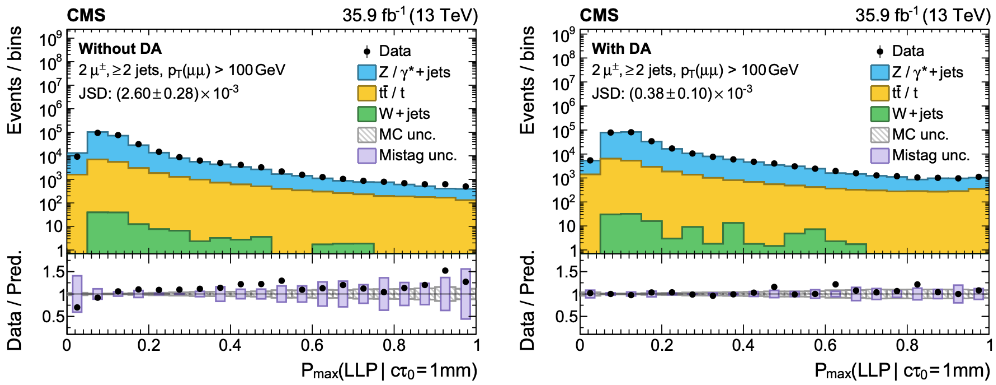

Figure 1: Comparisons of DNN score between data and simulation for the DNN designed to identify jets originating from an LLP, shown without (left) and with (right) the inclusion of a gradient reversal layer [9] during training. Inclusion of the gradient reversal layer both improves agreement between data and simulation and reduces the associated systematic uncertainty. Taken from Ref. [8].

While gradient reversal layers are shown to be successful, it is possible to take a more direct approach to improving the agreement between data and simulation. Rather than discourage the DNN from learning features which allow it to distinguish between examples from the source and target domain (as done for the gradient reversal layer), we propose to explicitly reward the DNN for minimizing differences between distributions in the source and target domains.

One possible implementation is the addition of a *histogram loss* component, originally proposed in the context of dimensionality reduction [10]. For a classification task which is trained on events having label y which is either 0 or 1 and which gives a prediction for each event $\hat{y} \in [0,1]$, probability distribution functions of $\hat{y}$ for events from the source domain (i.e. simulation), denoted S^S and with label $z=0$, and the target domain (i.e. data), denoted S^T and with label $z=1$, can be estimated through histograms H^S and H^T with N bins. The bins are assumed to be uniformly spaced, such that the bin centers are given by $t_1 = 0, t_2, ..., t_N = 1$ and the spacing between bins is given by $\Delta = 1/(N-1)$. The n-th bin of H^T can be constructed as

$$h_n^T = \frac{1}{|S^T|} \sum_{i:z_i=1} \delta_{i,n}, \tag{1}$$

where $\delta_{i,r}$ is a weight defined such that we linearly interpolate for each entry when constructing the histograms:

$$\delta_{i,n} = \begin{cases} (\hat{y} - t_{n-1})/\Delta, & \text{if } \hat{y} \in [t_{n-1}, t_n] \\ (t_{n+1} - \hat{y})/\Delta, & \text{if } \hat{y} \in (t_n, t_{n+1}] \\ 0, & \text{otherwise.} \end{cases} \tag{2}$$

In this way, an event with $\hat{y}$ falling directly between two bins t_n and t_{n+1}, i.e. $\hat{y} = (t_{n+1} - t_n)/2$ would contribute a weight of 0.5 to both h_n and h_{n+1}, and a weight of 0 to all other bins. Similarly, an event with $\hat{y} = t_n$ contributes a weight of 1 to h_n and a weight of 0 to all other bins. A loss function rewarding agreement between the probability distributions for $\hat{y}$ between data and simulation can be constructed as the sum of the squares of differences between each bin of H^S and H^T:

$$L_H(z, \hat{y}) = \sum_{n=1}^{N} (h_n^S - h_n^T)^2. \tag{3}$$

A composite loss function which rewards performance on the original task as well as agreement between H^S and H^T takes the form

$$L(y, z, \hat{y}) = L_C(y, \hat{y}) + \lambda L_H(z, \hat{y}), \tag{4}$$

where L_C would be a typical classification loss (e.g. cross-entropy) and λ is a hyperparameter that dictates the balance between rewarding classification and rewarding data/simulation histogram agreement.

2 Timeline

The proposed timeline is up to six months, starting in mid-June 2021. From mid-June to August, during which I have no other academic or research commitments, I will devote 100% of the time to the project. In the following months, funding can be supplemented from Boston University.

- Month 1: implement on a toy problem with CMS data. Construct a photon ID DNN with a simple set of features describing isolation and shower shape. The classification component is trained with prompt and fake photons from γ + jets simulation, while the histogram loss component takes electrons from data and simulation in a $Z \rightarrow ee$ control region. Deliverable: Functioning DNN with histogram loss component

- Month 2: compare to gradient reversal layer. Implement gradient reversal layer and compare performance between a DNN with a gradient reversal layer and a DNN with a histogram loss component. Deliverable: quantification of performance, both in terms of the original task (AUC) and the data/MC agreement, of DNN with histogram loss vs. DNN with gradient reversal layer

- Months 3-4: Use CERN OpenData to illustrate the idea with public datasets. Deliverable: publication

- Months 4-6: Prepare publication (student will be funded by Boston University).

References

[1] A. M. Sirunyan *et al.* [CMS], "Identification of heavy-flavour jets with the CMS detector in pp collisions at 13 TeV," JINST **13**, no.05, P05011 (2018) doi:10.1088/1748-0221/13/05/P05011 [arXiv:1712.07158 [physics.ins-det]].

[2] M. Stoye, "DeepFlavour in CMS," IML Machine Learning Workshop (2017) [https://cds.cern.ch/record/2256692].

[3] P. Baldi, P. Sadowski and D. Whiteson, "Searching for Exotic Particles in High-Energy Physics with Deep Learning," Nature Commun. **5**, 4308 (2014) doi:10.1038/ncomms5308 [arXiv:1402.4735 [hep-ph]].

[4] A. M. Sirunyan *et al.* [CMS], "A Deep Neural Network for Simultaneous Estimation of b Jet Energy and Resolution," Comput. Softw. Big Sci. **4**, no.1, 10 (2020) doi:10.1007/s41781-020-00041-z [arXiv:1912.06046 [hep-ex]].

[5] A. M. Sirunyan *et al.* [CMS], "Identification of heavy, energetic, hadronically decaying particles using machine-learning techniques," JINST **15**, no.06, P06005 (2020) doi:10.1088/1748-0221/15/06/P06005 [arXiv:2004.08262 [hep-ex]].

[6] A. M. Sirunyan *et al.* [CMS], "A search for the standard model Higgs boson decaying to charm quarks," JHEP **03**, 131 (2020) doi:10.1007/JHEP03(2020)131 [arXiv:1912.01662 [hep-ex]].

[7] J. Zhu, T. Park, P. Isola and A. A. Efros, "Unpaired Image-to-Image Translation Using Cycle-Consistent Adversarial Networks," 2017 IEEE International Conference on Computer Vision (ICCV), 2017, pp. 2242-2251, doi: 10.1109/ICCV.2017.244.

[8] A. M. Sirunyan *et al.* [CMS], "A deep neural network to search for new long-lived particles decaying to jets," Mach. Learn. Sci. Tech. **1**, 035012 (2020) doi:10.1088/2632-2153/ab9023 [arXiv:1912.12238 [hep-ex]].

[9] Y. Ganin and V. Lempitsky, "Unsupervised Domain Adaptation by Backpropagation," *Proc. ICML '15* (Lille, France, 2015), p. 1180, [arXiv:1409.7495 [stat.ML]].

[10] E. Ustinova and V. Lempitsky, "Learning Deep Embeddings with Histogram Loss," Advances in Neural Information Processing Systems **29**, (2016).