

Domain adaptation with gradient reversal for MC/real data calibration

A. Ryzhikov^{1,2} and A. Ustyuzhanin^{1,2}

¹ National Research University Higher School of Economics, 20 Myasnitskaya st., Moscow 101000, Russia

² Yandex School of Data Analysis, 11/2, Timura Frunze st., Moscow 119021, Russia

E-mail: artemryzhikov@gmail.com

Abstract.

It is quite common part of the data analysis in High Energy Physics to train a classifier for signal and background separation. In case the signal under investigation is a rare process, the signal sample is simulated and background sample is taken from the real data. Such setting create an unnecessary bias: the classifier might learn not the characteristic of the signal but the characteristic of the imperfect simulation. So the challenge is to train the classifier in such way that it picks up signal/background difference and doesn't overfit to the simulation-specific features. The suggested approach is based on cross-domain adaptation technique using neural networks with gradient reversal. The network architecture is a dense multi-branch structure. One branch is responsible for the signal/background discrimination, the second branch helps to avoid the overfitting on the Monte-Carlo training dataset. The tests showed that this architecture is a robust mechanism for choosing trade-offs between discrimination power and overfitting. So the resulting networks successfully distinguishes the signal from the background, but does not distinguish simulated events from the real ones. The third network branch helps to reduce the correlation between the classifier predictions and reconstructed mass of the decay, thereby making such approach highly viable for wide variety of physics searches.

1. Introduction

Vast majority of data analyses in High Energy Physics relies on discrimination between signal and background events. Physics has pioneered and adopted application of the machine learning (ML) approaches for that task. Methods like Decision Trees, Boosting and Neural Networks play significant role in the modern physics discoveries. Today, with the advance of the deep learning, new software frameworks and increased computational power neural networks are becoming more and more popular.

However the application of a large number of machine learning methods for the problems of High Energy Physics need to be considered with care. The reason for that is the low generalization quality and certain number of physical restrictions for discriminative models. Besides demand for high values for corresponding figures of merit models should provide physically-sound result. In this research we propose a new approach for building signal/background discriminator based on neural networks for the high energy physics problem. The method is based on cross-domain adaptation technique with gradient reversal [1]. It suggests the following architecture for the neural network: there is a common part (that is responsible

for generation of important features) and 3 branches growing out of it: the first branch is responsible for signal detection, the second branch helps to avoid overfitting on simulated events and and the third one avoid sculpting a false mass peak.

2. Problem

Usage of the simulated sample is fairly common approach in the High Energy Physics. However it is often hard to include all physics factors into Monte-Carlo simulation. Moreover, not all the variables can be simulated accurately enough, so the discrepancies may lead either to:

- (i) Simulation of both signal and background with high CPU consumption, or to..
- (ii) ML models trained on simulated sample tend to overfit to the simulation artifacts and work poorly on real data.

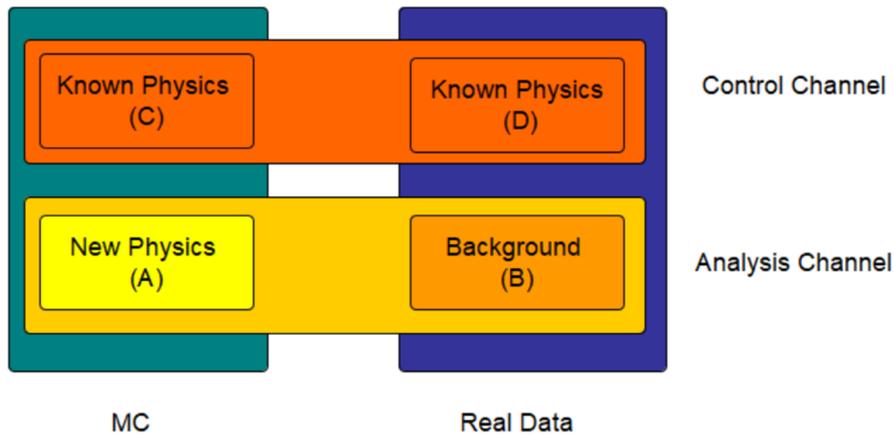


Figure 1. Training on the mixture of the simulated (MC) and the real samples

In this research we consider $\tau \rightarrow 3\mu$ decay that has been published at the Data Science challenge "Flavours of Physics" on *kaggle.com* [3]. The challenge is three-fold:

- (i) The quality of the signal discrimination from the background should be as high as possible. The evaluation metric for the signal discrimination is Weighted Area Under the ROC Curve (truncated AUC) [3]
- (ii) Since the classifier is trained on a mixture of simulated signal and real data background, it is possible to demonstrate high performance by exploiting the features that are specific to the simulation. We require that the classifier should not have a large discrepancy when applied to real and simulated data. To verify this, we use a control channel, $D_s \rightarrow \phi\pi$, that has a similar topology as the say, $\tau \rightarrow 3\mu$ (analysis channel). $D_s \rightarrow \phi\pi$ is a well-known decay, as it happens much more frequently. So the goal is to train a classifier able to separate A from B but not C from D (Figure 1). A Kolmogorov-Smirnov (KS) test is used to evaluate the similarity between the classifier output distributions on C and D. In our problem KS is calculated between predictions distributions for real and simulated data for $D_s \rightarrow \phi\pi$ channel. The KS-value of the test should be less than 0.09.
- (iii) The classifier output should not be correlated with the reconstructed mass feature, i.e. its output distribution should not sculpt artificial peaks that could be interpreted as a (false) signal. To test the flatness we have used Cramer-von Mises (CvM) test that gives the uniformity of the distribution [4].

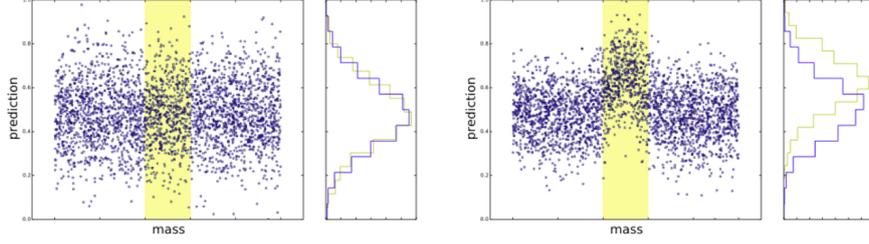


Figure 2. Illustration of the CvM correlation test [4]. On the left side there is no correlation with mass (low CvM values). On the right side models predictions are highly correlated with mass (high CvM values)

3. Baseline

At the moment of this publication the *Data Doping* [2] was the best technique to deal with the classifier overfitting to Monte-Carlo-specific features. We use this technique as the baseline.

The idea of Data Doping is to dope the training sample with a small fraction of Monte-Carlo events from the control channel (denoted by C at Figure 1) labeled as background. Thus, it forbids the classifier to rely on features discriminating real and background events. The optimal fraction of the doping events was taken from [2].

4. Domain adaptation

As an alternative to *Data Doping* we applied method based on *Cross-Domain adaptation with gradient reversal* [1]. The concept is similar to GAN (Generative Adversarial Nets) [6]: we use an additional network branch that is trained to discriminate real from simulated (domain classifier) events. During the training process we reverse the gradient from the domain classifier in order to prevent the whole model from discrimination of the real signal from the simulation (Figure 3).

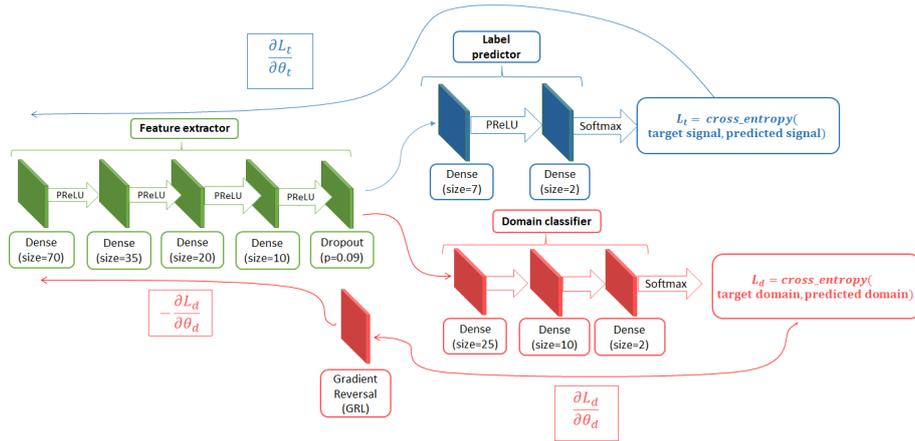


Figure 3. Cross-domain adaptation with gradient reversal

Instead of reversing the gradient it is possible to use the domain classifier loss function with negative sign, since

$$-\frac{\partial L_d}{\partial \Theta_d} = \frac{\partial(-L_d)}{\partial \Theta_d}$$

effectively using the - cross-entropy as the final objective for *Domain classifier*.

The network architecture has a dense 3-branch structure (Figure 3) and consists of the following parts:

Feature extractor - responsible for feature preprocessing;

Label predictor - responsible for the target prediction (signal/background discrimination);

Domain classifier - responsible for cross-domain adaptation and prevents the network from overfitting to MC domain;

Mass predictor - helps to eliminate the correlation between classifier predictions and the reconstructed mass of the decay (It wasn't tested in this research and our architecture was tested without this part. It is omitted from the Figure 3. Theoretically it was designed as additional branch, similar to domain classifier, working along the same principle).

5. Data

Training dataset (Analysis channel) consists of 67000+ events of signal ($\tau \rightarrow 3\mu$) and background events. Control channel consists of 71000+ events of signal ($D_s \rightarrow \phi\pi$) and background. All events are described by 46 features. It is taken from the "Flavours of Physics" challenge [3].

6. Training

The architecture was implemented in **Python 2.7** using *Lasagne (ver. 2.1)* framework. We tuned the following parameters to obtain stable results:

- learning rates ratio between branches (*learning rate multiplier*)
- batch sizes ratio for branches. The best observed values were 1000 and 300 for *Label predictor* and *Domain classifier* respectively
- number of batches per epoch ratio. The best observed ratio between batches number was 6:1 for *Label predictor* and *Domain classifier* respectively

The model was trained for 20 epoches with RMSProp optimizer. To achieve highest stability and reproducibility we trained only *Feature extractor* and *Label predictor* parts several (20) epoches with frozen *Domain classifier* and after that *Domain classifier* was trained (2'nd step).

During the first step only *Feature extractor* and *Label predictor* were trained. Training procedure was 20-epoches RMSProp-optimization of categorical (2-classes) cross-entropy loss. Batch size was 1000. Learning rate at first step was 0.01 and was decayed 10 times each 5 epoches.

During the second step we restored *Feature extractor* and *Label predictor* from previous step and trained all parts. *Feature extractor* and *Label predictor* trained in the same way (20 epoches, same loss, same batch size, same initial learning rate and the same learning rate decay policy). *Domain classifier* was trained with the same parameters during each 6 batches of *Feature extractor* and *Label predictor* training and with the different batch size (batch size=300).

To eliminate *KS*-test difference we increased such Domain classifiers parameters as learning rate, corresponding batch size and batches frequency. **Figure 4** represents such dependency from one of such parameters. It was observed that too small values of *KS* makes *CvM* values higher and *AUC* metric smaller. So the goal was to find balance between *KS*, *CvM* and *AUC* using parameters described above.

7. Results

In the research we've compared the following models: *Baseline* (label predictor from Figure 4 without Domain Adaptation), *Domain Adaptation* (our approach), *Data Doping*. Models were

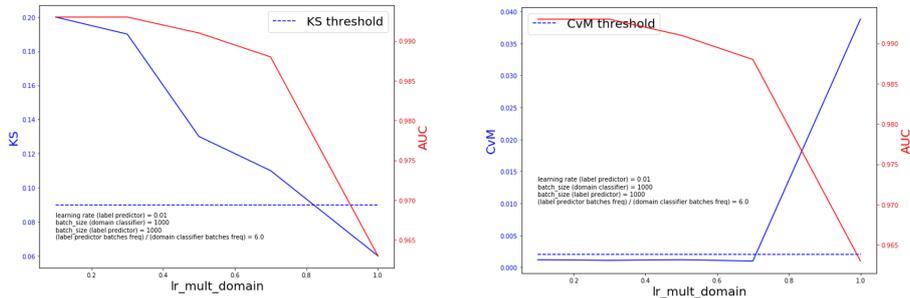


Figure 4. Metrics dependency from domain classifiers *learning rate multiplier*

tested on 85000+ events of signal ($\tau \rightarrow 3\mu$) and background. KS and CvM thresholds were chosen 0.09 and 0.02 respectively

The tests outcome is shown in (Table 1) demonstrates that this architecture is a robust mechanism for choosing trade-off between discrimination power and overfitting.

As shown in the table below our method provides the best solution for signal detection problem ($\tau \rightarrow 3\mu$).

	<i>AUC(truncated)</i>	<i>KS</i>	<i>CvM</i>
<i>Mass-aware Classifier</i>	0.999	0.18	0.0008
<i>Data Doping</i>	0.9744	0.087	0.0011
<i>Domain adaptation</i>	0.979	0.06	0.0008

Table 1. AUC, KS, CvM scores for different approaches

From the table above we can conclude that the Domain Adaptation approach can be used for training models on MC and real data mixture: it avoids using poorly simulated features while keeping the discrimination quality quite high.

8. Conclusion

Domain Adaptation approach addresses important problem of modern data analysis in the High-Energy Physics. It can be used without any special data pre-processing, giving flexibility to control signal/background discrimination quality versus real/simulation mismatch overfitting. The code is published on github [5] that demonstrate training/testing performed on public dataset [3]. This approach is much more flexible than *Data Doping* since it can be easily extended by additional restrictions (like mass-invariant predictions) by further network branching.

References

- [1] Ganin Y and Lempitsky V 2015 Unsupervised domain adaptation by backpropagation (Moscow: Skolkovo Institute of Science and Technology)
- [2] Gaitan V 2016 Data Doping solution for Flavours in Physics challenge *Heavy Flavour Data Mining workshop* (Zurich: University of Zurich, Irchel Campus)
- [3] *Flavours of Physics Competition*
- [4] Rogozhnikov A, Bukva A, Gligorov V, Ustyuzhanin A, Williams M 2015 New approaches for boosting to uniformity (Moscow: Yandex)
- [5] Ryzhikov A, Ustyuzhanin A 2017 Source code for Domain Adaptation research <https://github.com/Leensman/Cross-domain-adaptation-on-HEP-HSE-course-work>
- [6] J. Goodfellow I, Pouget-Abadie J, Mirza M, Xu B, Warde-Farley D, Ozair S, Courville A, Bengio Y 2014 Generative Adversarial Networks (Montreal: Universite de Montreal)