Output Correction in HEP using DPGMM

Adil Omari
Computer Science Dept.
Universidad Autonoma de Madrid
Madrid, Spain
adil.omari@inv.uam.es

Roberto Díaz-Morales
Dept. Signal Theory
and Communications
Universidad Carlos III de Madrid
Madrid, Spain
rdiazm@tsc.uc3m.es

Juan J. Choquehuanca-Zevallos
Dept. Signal Theory
and Communications
Universidad Carlos III de Madrid
Madrid, Spain
jchoquehuancaz@tsc.uc3m.es

Abstract

When combining several classifiers, more often than not outputs are considered to be in the same space or range and a direct combination of them is performed (such as simple weighted sums or products). Even if they are results of applying activation functions at the output of each classifier, there is not any guarantee that estimates belong to the same output distributions in order to combine them.
In this paper, a new combination rule to deal with the Kolmogorov-Smirnov and the Cramer-von Mises tests is presented. It takes into account the probability density functions of output estimates from each individual classifier and makes a mapping in a new label space in where the cumulative distributions of sample estimates are maintained (avoiding dissimilarities in distributions). To do so, the probability densities are found by using a Bayesian approach, more precisely, the Dirichlet Process Gaussian Mixture Model technique is used, avoiding tedious validation processes to find correct parameters of the mixture.

1 Introduction

Machine learning has shown to be a very helpful tool in the field of high energy physics (HEP), a field that aims to discover the structure of matter by studying its particles, resulting in an increasing amount of literature that shows the applications of the wide variety of learning algorithms applied to this field. Under this context, the event selection area is an important task that helps physicists in their works. We could mention some examples like [1, 2, 3] in where genetic programming is used. In [4], the authors use gene expression programming. Non evolutionary algorithms like decision trees have been used in [5] to search neutrino oscillations and Neural Networks were used in [6] for top quark selection using the data of the Fermilab Tevatron accelerator. As well, the number of publications related to this area has been greatly increased due to collaborations with the Large Hadron Collider (LHC) and the release of new datasets. It has a direct effect which is the underpinning of the usefulness of machine learning methods in several fields. Here, it is worth to mention some examples such as [7] that uses multivariate techniques or [8] and [9] that use Deep Neural Networks (DNN).

In 2014, a Machine Learning Challenge [10] was organized to encourage the collaboration between high energy physics and data scientists. The result was a solid bridge among both communities, link

that is supported by the winning approaches of the competition such as [11, 12, 13, 14], solutions that were obtained using state of art algorithms such as combinations of DNN and Gradient Boosted Trees.

In the same direction, HEP has opened a new challenge: the Flavour of Physics Challenge [15] where one of the main difficulty is how to deal with scenarios in which the scarcity of data imposes a strong limitation to have a considerable amount of samples that can be used to determine the existence of unknown phenomena, requiring the development of mechanisms to model, simulate and generate data, and so, being able to build classifiers that can distinguish among different phenomena. Unfortunately, some machines can learn discrepancies between those real and simulated samples, so, in order to avoid these unexpected behaviors, machine estimates have to be checked with some tests that are often used by physicist. As part of this challenge two different tests are employed, those are:

- **Cramer-von Mises** (CvM) test is used to ensure the estimation outputs are not correlated with mass estimation and so to avoid signal-like samples to be incorrectly estimated as background.
- **Kolmogorov-Smirnov** (KS) test is used to ensure the resultant output distributions for real and simulated data does not show high discrepancies.

Another important issue that comes with this problem is the lack of prior domain knowledge to correctly pre-process raw data that helps to overcome both tests by making the output signal to be uncorrelated with control channels (i.e. the $D_s^+ \rightarrow \phi(\rightarrow {}^{-+})\pi^+$ channel and hidden mass information). In this work, we present a way to combine classifiers and form an ensemble capable of satisfy the CvM and KS tests at the same time that shows a good classification performance.

This paper is organized as follows: Section 2 shows the architecture of the classification system and the fusion strategy to mix classifiers. Section 3 presents the results of our experiments. Finally, in Section 4, some conclusions and future lines are presented.

2 Architecture

The architecture of the proposed system is mainly composed by two well defined blocks (referred as C1 and C2) and an upper aggregation layer that fuses both outputs.

The first block tries to exploit as much information as possible from the raw data to improve a Weighted Area Under the Receiver Operating Characteristic Curve (WAUC). For that, we divide the training data set and form a plunge of M sets for training and its corresponding validation sets. Each data set $m \in 1, ..., M$ is used to optimize a Gradient Boosting Machine (GBM) [16]. The output for a given sample $\mathbf{x}^{(n)}$ is the simple mean of the M outputs, i.e.

$$y_1^{(n)} = \frac{1}{M} \sum_{m=1}^{M} f_m \left(\mathbf{x}^{(n)} \right) \tag{1}$$

where $y_1^{(n)} \in [0, 1]$ is the output label for this first model (block) composed by classifiers $f_m : R^d \rightarrow [0, 1]$.

In spite that the probability density distributions (pdf) of label estimates show that this first ensemble is highly discriminant among classes (getting a high AUC score), unfortunately there exist regions on the output space where estimations for simulated and real samples do not match the same distributions; and so, estimations from this block don't pass the KS test. Mainly because the latent representations learned by the model could be correlated with the mechanisms to simulate and generate samples with poorly modeled features.

So, we compensate these deficiencies by building a second model to help to correct outputs in those regions where the first block fails. With the intention to eliminate those variables that impinge strongly on the result of KS test, we opt for a dimensionality reduction by a backward procedure to select a subset of features. Thus, one important point to highlight is the fact that our approach does not use any information of the physical meaning of features rather than those found by the learners

we employ. In this way, we avoid exploring and obtaining features that are possibly unique of the process of the simulation when generating samples.

Once a subset of features has been selected, we train a classifier with a simple combination of a GBM and Random Forests [17].

2.1 Output fusion

Since both ensembles are heterogeneous in nature, to correct and improve estimates of the first block classifier, we do not opt for a direct combination of both output estimates (even more, we do not get any considerable improvement), instead we perform a mapping from the output space of the second block into the output space of the first block classifier. For that, we first identify regions on the output space that are possibly related with latent variables that resemble reminiscences of the mechanisms to artificially generate samples, and so making it difficult to pass the tests. Then the next rule is to consider to join both estimates and so get a new corrected output $\hat{y}^{(n)}$ (a function of $y_1^{(n)}$ and $y_2^{(n)}$)

$$\hat{y}^{(n)} = \beta_r y_1^{(n)} + (1-\beta_r) F_{Y|C1}^{-1}\left(F_{Y|C2}\left(y_2^{(n)}\right)\right) \tag{2}$$

where β_r represents the weight for the first block at some region r in the space of y_1. $F_{Y|C2}$ is the cumulative function given the model of the second block (w.r.t. $p(y^{(n)}|C2)$), and $F_{Y|C1}^{-1}$ is the inverse cumulative function given the model of the first block (w.r.t. $p(y^{(n)}|C1)$).

- Evidently, when $\beta_r = 1$, the final output score is just $\hat{y}^{(n)} = y_1^{(n)}$.
- When $\beta_r = 0$ and a map from y_2 into y_1 space is needed in such a way that the cumulative distribution function always remains the same, i.e., $P(Y < y_1^{(n)}|C1) = P(Y < y_2^{(n)}|C2)$. Meaning that $y_2^{(n)}$ and $y_1^{(n)}$ should be in the same quantile respect to their probability density functions.

One important aspect of the above combination is that -for $\beta_r > 0$- the transformation reshapes the pdf for y_2 and as a result it is modified in such a way that its range values suit those of y_1. So, we make the system to be less sensitive on discrepancies of real and simulated data at the same time that it maintains a high classification performance. This is specially useful to treat estimations that falls in the region near 0.6.

2.1.1 Probability density estimation

Regarding the modeling of conditional distributions of the outputs from both blocks $p(y^{(n)}|C1)$ and $p(y^{(n)}|C2)$, there exist several approaches in the literature that help to face this problem, however, in this work we model the output distributions of both ensembles by means of a Gaussian Mixture Model (GMM) defined as in Eq. 3.

$$p\left(y|\pi,\mu,\Sigma\right) = \sum_{i=1}^{K} \pi_i \mathcal{N}\left(y|\mu_i,\Sigma_i\right) \tag{3}$$

where K is the number of components of the mixture and $\pi_i \in [0,1]$ are called the mixing proportions satisfying $\sum_i \pi_i = 1$ for $i = \{1,..,K\}$.

The use of GMM can be a tedious task regarding the selection of an adequate number of components to mix. This issue can easily be jumped by considering a Bayesian treatment of the problem, letting us save computational efforts -avoiding validation routines- when selecting the right hyperparameters. For that, the estimation of the corresponding output pdf's (represented by $p(\cdot)$) is overcome by using a Dirichlet Process GMM (DPGMM) approach [18], that has the benefit of automatically determining the number of mixture components.

To be brief, the DPGMM approach can be written as follows:

$$p\left(y_i|c_i,\pi_i,\mu_i,\Sigma_i\right)=\mathcal{N}\left(\mu_i,\Sigma_i\right) \tag{4}$$

$$p\left(c_i|\boldsymbol{\pi}\right)=Discrete(\pi_1,...,\pi_K) \tag{5}$$

$$p\left(\mu_i,\Sigma_i\right)=G_0 \tag{6}$$

$$p\left(\boldsymbol{\pi}|\alpha\right)=Dir(\alpha/K,...,\alpha/K) \tag{7}$$

where G_0 is the base distribution, and $\boldsymbol{\pi}=\pi_1,...,\pi_K$. c_i are indicator numbers with join distribution defined as

$$p(c_1,..,c_n|\boldsymbol{\pi})=\prod_{k=1}^{K}\pi_k^{n_j} \tag{8}$$

in where n_k is called the occupation number with join probability defined as:

$$p(n_1,...,n_K|\boldsymbol{\pi})=\frac{n!}{n_1!n_2!...n_K!}\prod_{k=1}^{K}\pi_k^{n_j} \tag{9}$$

3 Experiments and results

3.1 Dataset

In order to composite the database, a series of collision events are performed and recorded by the LHCb detector (Large Hadron Collider beauty) one of seven detectors particle accelerator LHC at CERN. The purpose was to find a phenomenon that is a good indicative of "new physics"[15], i.e. the charged lepton flavour violation.

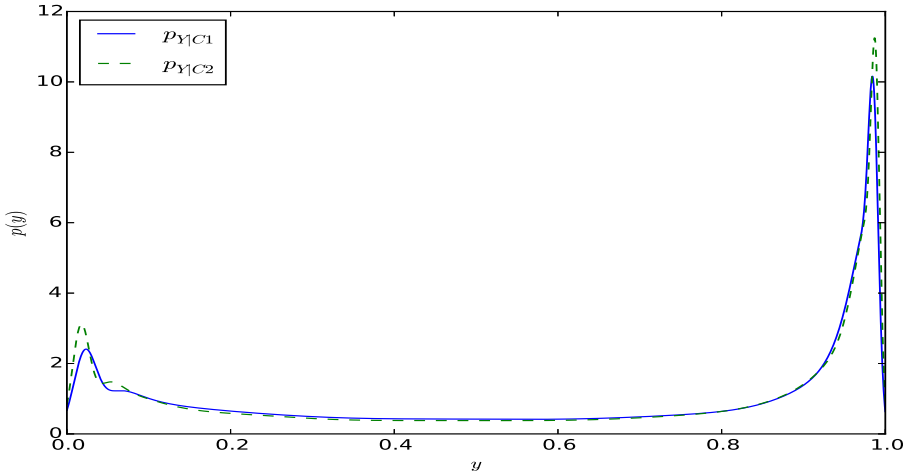
The final dataset (Table 1) is formed by a mixture of real and simulated data for signal events (class 1) and real data for background events (class 0). As well, there exists two separate -called Agreement and Correlation- datasets to check if the final system meets the CvM and KS requirements. The strong demand is not to use neither of them in the training stage. The final score is obtained in a separate test dataset.

Dataset	#Samples
Train (Class 0/Class 1)	67553 (41674/25879)
Agreement	331147
Correlation	5514
Test	855819

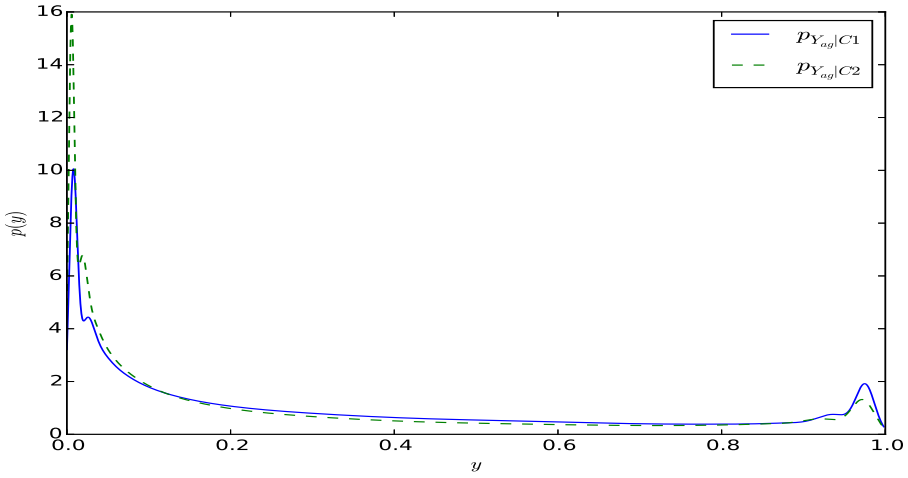
Table 1: Description of challenge dataset.

3.2 Results

This section presents the outcomes of simulations performed on the challenge database. Figs. 1a-1b shows the estimated pdf's for Block 1 and 2. It can be said that on training dataset, we can not observe strong discrepancies since positive samples in the training dataset only contains simulated samples, and so, both distributions seems to be correlated. Results on the formed validation dataset are shown in Table 2, where it can be seen that output estimates of the first ensemble (Block 1) has the highest discrimination perfomance (high WAUC) while the second ensemble (Block 2) has the main property of accomplishing the more demanding test, i.e. KS test. As well, it can be appreciated that



(a)



(b)

Figure 1: Probability density functions for blocks 1 and 2 with DPGMM for: a) train dataset and b) agreement dataset.

when our fusion strategy is employed, we can get a considerable increment in WAUC score while maintaining the results on both tests.

	WAUC	CvM test	KS test
Block 1	0.9881	0.001 (True)	0.202 (False)
Block 2	0.9828	0.001 (True)	0.084 (True)
Fusion	0.9830	0.001 (True)	0.086 (True)

Table 2: Results on validation data set.

Finally, the WAUC score obtained with our proposal was 0.988720 on the test dataset.

4 Conclusions

In this paper, we have proposed a new approach to fuse different classifiers. The method was applied to the Flavours of Physics dataset, challenge that imposes several restrictions on distribution of estimated labels. Our results show that improvements in performance of the final classification system can be obtained when applying our approach while accomplishing the CvM and KS test requirements.

Finally, we would like to mention that are several directions along which this work can be extended. In particular, we are actually studying the possibility to use the proposed combination as a general combination for heterogeneous ensemble -ensemble with learners of a different nature, e.g. Support Vector Machines (SVMs) or semiparametric SVMs [19]-. As well, another interesting idea is to replace β_r by $\beta(\mathbf{x})$, achievable by using algorithms like [20] [21].

Acknowledgements

With partial support from Spain's grants TIN2013-42351-P and S2013/ICE-2845 CASI-CAM-CM and also of the Cátedra UAM–ADIC in Data Science and Machine Learning. The authors also gratefully acknowledge the use of the facilities of Centro de Computación Científica (CCC) at UAM.

References

[1] K. Cranmer and R. S. Bowman, "PhysicsGP: A genetic programming approach to event selection," *Computer Physics Communications*, vol. 167, no. 3, pp. 165–176, 2005.

[2] J. M. Link, P. M. Yager, J. C. Anjos, I. Bediaga, C. Castromonte, C. Gobel, A. A. Machado, J. Magnin, A. Massafferri, J. M. de Miranda, and et al., "Application of genetic programming to high energy physics event selection," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 551, no. 2, pp. 504–527, 2005.

[3] R. Berlich and M. Kunze, "Parametric optimization with evolutionary strategies in particle physics," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 534, no. 1, pp. 147–151, 2004.

[4] L. Teodorescu and D. Sherwood, "High energy physics event selection with gene expression programming," *Computer Physics Communications*, vol. 178, no. 6, pp. 409–419, 2008.

[5] B. P. Roe, H.-J. Yang, J. Zhu, Y. Liu, I. Stancu, and G. McGregor, "Boosted decision trees as an alternative to artificial neural networks for particle identification," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 534, no. 2, pp. 577–584, 2005.

[6] S. Whiteson and D. Whiteson, "Machine learning for event selection in high energy physics," *Engineering Applications of Artificial Intelligence*, vol. 22, no. 8, pp. 1203–1217, 2009.

[7] D. C. O'Neil on behalf of the ATLAS collaboration, "Tau identification using multivariate techniques in ATLAS," *Journal of Physics: Conference Series*, vol. 368, no. 1, p. 012029, 2012.

[8] P. J. Sadowski, D. Whiteson, and P. Baldi, "Searching for Higgs boson decay modes with deep learning," 2014, pp. 2393–2401.

[9] P. Baldi and P. Sadowski and D. Whiteson, "Enhanced Higgs boson to $\tau+\tau-$ search with deep learning," *Physical review letters*, vol. 114, no. 11, p. 111801, 2015.

[10] "HiggsML. Higgs Boson Machine Learning Challenge," 2014. [Online]. Available: http://www.kaggle.com/c/higgs-boson

[11] R. Dıaz-Morales and A. Navia-Vázquez, "Optimization of AMS using Weighted AUC optimized models," vol. 42, pp. 109–127, 2015.

[12] G. Melis, "Dissecting the winning solution of the HiggsML challenge," in *Cowan et al., editor, JMLR: Workshop and Conference Proceedings*, no. 42, 2015, pp. 57–67.

[13] P. Sadowski, J. Collado, D. Whiteson, and P. Baldi, "Deep Learning, Dark Knowledge, and Dark Matter," vol. 42, pp. 81–97, 2015.

[14] T. Chen and T. He, "Higgs boson discovery with boosted trees," in *Cowan et al., editor, JMLR: Workshop and Conference Proceedings*, no. 42, 2015, pp. 69–80.

[15] "Flavours of Physics Challenge," 2015. [Online]. Available: https://www.kaggle.com/c/flavours-of-physics

[16] J. H. Friedman, "Greedy Function Approximation: A Gradient Boosting Machine," *Annals of Statistics*, vol. 29, pp. 1189–1232, 2000.

[17] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[18] D. Görür and C. E. Rasmussen, "Dirichlet Process Gaussian Mixture Models: Choice of the Base Distribution," *J. Comput. Sci. Technol.*, vol. 25, no. 4, pp. 653–664, Jul. 2010.

[19] R. Diaz-Morales and A. Navia-Vázquez, "Efficient parallel implementation of kernel methods," *Neurocomputing*, in press, forthcoming.

[20] A. Omari and A. R. Figueiras-Vidal, "Feature combiners with gate-generated weights for classification," *IEEE Trans. Neural Networks and Learning Sys.*, vol. 24, pp. 158–163, 2013.

[21] A. Omari and A. R. Figueiras-Vidal, "Post-aggregation of classifier ensembles," *Information Fusion*, vol. 26, pp. 96–102, 2015.