Baku State University: Journal of Physics & Space Sciences, **2024**, v1 (3), p. 27-33

journal homepage: http://bsuj.bsu.edu.az/en

Study of the associative production of the Higgs boson with the Z-boson using MVA methods

Faig N. Ahmadov*

1. Institute of Physics Ministry of Science and Education, Baku, Azerbaijan

Received 16-Jun-2024; Accepted 05-Aug-2024

DOI: https://doi.org/10.30546/209501.2024.1.3.027

Abstract

In the work, two multivariate analysis methods Neural Network (NN) and Boosted Decision Tree (BDT) were used to separate the $ZH(b\bar{b})$ signal from the background and the results obtained from them were compared. The list of input variables for BDT and NN is similar to those used in the analysis in the ATLAS experiment. Up to 0.8 million signals and the same number of background events were used for training and testing. The settings used in the ATLAS analysis, which has the best performance, were chosen to tune the BDT hyperparameters. Various number of events (0.1M, 0.2M and 0.8M) are trained and different settings for NN are obtained, providing performance that exceeds that of BDT. It turns out that for any number of training events, it is possible to find corresponding NN settings with better performance than BDT. The problem with NN training is that it is computationally intensive compared to BDT.

***Keywords**: Higgs boson, associative production, hyperparameter, Neural Network, Boosted Decision Tree*
PACS: 14.80.Bn; 02.50.Sk; 07.05.Mh

1. Introduction

After the discovery of the Higgs boson in experiments at the LHC, its mass was measured to be 125 GeV [1, 2]. With such a mass, the probability of its decay into bb is greater than the sum of the probabilities of all other decay channels [3]. Therefore, this channel makes a great contribution to the study of the Higgs boson. To study the Higgs boson in the bb decay channel, a more suitable production channel is associative production with a vector boson. Since the decay of the Higgs boson into

* e-mail: fahmadov@jinr.ru, ORCID ID: 0000-0003-3644-540X.

a pair of b-quarks was observed for the first time in this production channel. Therefore, we can say with confidence that the $\mathrm{VH}(\mathrm{b\bar{b}})$ (V is Z or W) process is one of the most important channels for studying the properties of the Higgs boson. Theoretical and experimental data can be analyzed using different methods. Currently, a more promising method is the Multivariate Analysis method (MVA). The MVA method itself is divided into a large number of sub methods or algorithms.

The analysis of the data using the multivariate analysis method in LHC experiments began after the experimental confirmation of the Higgs boson. Two multivariate analysis methods are commonly used in LHC experiments to study pp collision events. These methods are Boosted Decision Tree (BDT) [4] and Neural Network (NN) [5]. Until 2013, the cut-based analysis method was used to study many processes of the Standard Model and beyond the Standard Model.

2. Event selection

Before using MVA, some pre-selection cuts were applied to events and objects. We use $\mathrm{ZH}(\mathrm{b\bar{b}})$ only as a signal process, where the Z boson decays into charged leptons (electrons or muons) and the Higgs boson decays into a pair of bb quarks. At the level of event generators, we get four objects, two oppositely charged leptons and b-anti-b-quarks. If we take into ac-count the initial and final state radiation, then in the final state we can get more than two quarks. But in order for the results to be used in the future when analyzing experimental data, it is necessary to take into account the effects of the detector (in this work, the effects of the ATLAS detector were used). Since quarks are detected as jets in the detector, we need to distinguish jets of b-quarks from jets of other light quarks. For this purpose, a b-tagging algorithm based on a neural network is used, which takes into account the kinematic parameters of the jet.

To isolate and reject some background and fake objects, the following cuts are applied to selected objects:

- The events should contain two charged leptons of the same flavor with transverse momentum $p_T > 25$ GeV and pseudorapidity $|\eta| < 2.5$ and there should not be more than two leptons with $p_T > 7$ GeV and $|\eta| < 2.7$.
- To suppress non-resonant backgrounds, the invariant mass of these two leptons should be in the range 81 GeV < m_{ll} < 101 GeV (closer to the Z-boson mass).
- The transverse momentum of the Z boson reconstructed from these two leptons should be $p_T^Z > 75$ GeV.
- Also, the events should contain two b-jets with transverse momentum $p_T > 20$ GeV and $|\eta| < 2.5$ where the Higgs boson is reconstructed from these two b-jets. It is required that the b-jet with the highest transverse momentum have $p_T > 45$ GeV.
- Third or more jets with $p_T > 30$ GeV are allowed, but they should not be b-jet.

Typically, in analyzes of experimental data, events are divided into two categories,

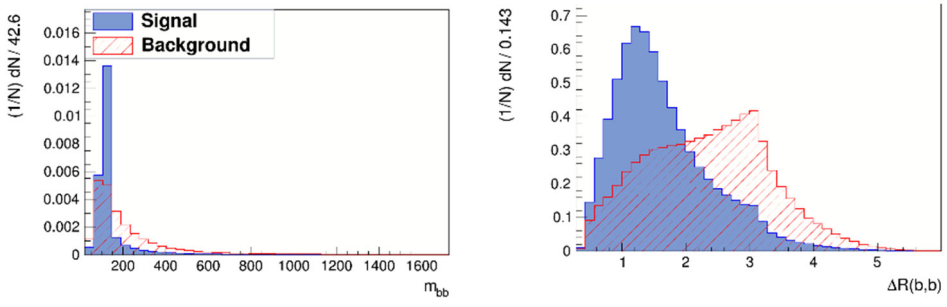
events with two jets and events with three or more jets. For simplicity, this work considers one category, which includes events with two or more jets.

3. BDT and NN analysis techniques

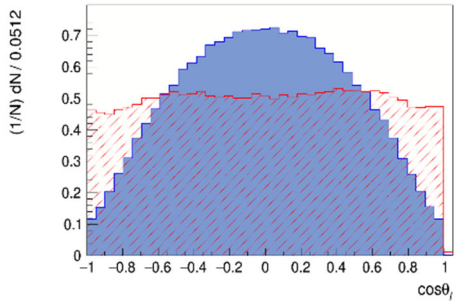
In this work, two well-known machine learning algorithms BDT and NN are used to sepa-rate the $ZH(b\bar{b})$ signal from backgrounds such as Z+jet, W+jet, $t\bar{t}$ and di-boson. The BDT pa-rameters were sufficiently optimized when analyzing the ATLAS data [6], so we used the algo-rithm with already optimized parameters. The main focus will be on NN optimization. The same input variables are used for BDT and NN, and they are as follows:

- $\mathbf{m_{bb}}$ – invariant mass of two b-jets,
- **ΔR(b,b)** – angular distances between the two b-jets,
- **Δϕ(Z,bb)** – azimuthal distance between the Z-boson and two b-jets,
- **dη(Z,bb)** – pseudorapidity distances between the Z-boson and the two b-jets,
- $\mathbf{m_{ll}}$ – invariant mass of the 2 leptons system,
- $\mathbf{p_T^Z}$ – transverse momentum of the Z-boson (the vector sum of the 2 charged leptons pT),
- $\mathbf{p_T^{b1}}$ – transverse momentum of 1st b-jet,
- $\mathbf{p_T^{b2}}$ – transverse momentum of 2nd b-jet,
- $\mathbf{cos\theta_l}$ – cosine of the angular distance between the direction of the negatively charged lepton in the Z boson rest frame and the flight direction of the Z boson in the laboratory frame.

The distribution of some of these input variables that have higher separating power is shown in Figure 1. As can be seen from the figure, the shape of the signal distribution differs from the shape of the background. Due to this difference, the signal to background ratio can be improved. In addition, BDT and NN take into account the correlation between input variables, shown in Figure 2 for signal (a) and background (b).

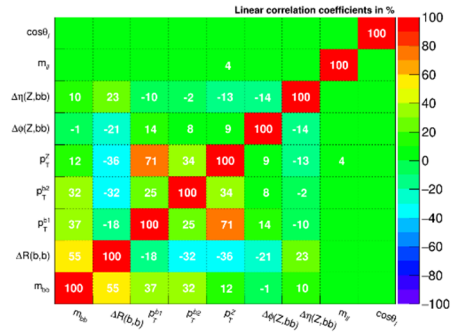


(**a**) (**b**)

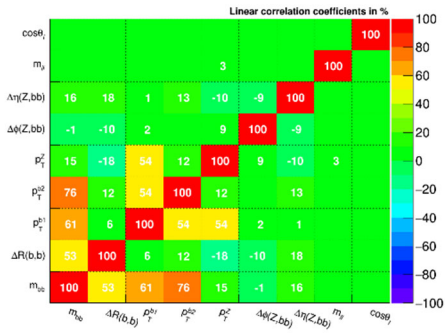


(c)

Fig. 1. Distributions of the invariant mass of the two b-jets (a), the transverse momentum of the Z-boson (b) and the cosine of the angle defined above (c) for the signal (blue histogram) and for the background (red histogram).



(a)



(b)

Fig. 2. Correlation matrices for signal (a) and background (b).

4. Optimization of multivariate analysis algorithms

We use TMlpANN as the NN and Gradient BDT (BDTG), from the Toolkit for Multivariate Data Analysis (TMVA) [7] in ROOT.

As mentioned above, we use an optimised BDT with the following hyperparameters:

- **NTrees = 200** (The number of trees in the forest),
- **MaxDepth = 4** (The maximum depth of cell tree),
- **BoostType = Grad** (Boosting type for the trees in the forest. GradientBoost uses the binomial log-likelihood loss function ln(1 + exp(−2F(x)y)) for classification),
- **Shrinkage = 0.5** (Learning rate for GradBoost algorithm),
- **SeparationType = GiniIndex** (Separation criterion for node splitting. For a given cell the gain is defined as p(1 − p), where p = nS/(nS + nB) in the considered cell, nS and nB are the sum of the weights of signal and background events in that node, respectively),
- **nCuts = 100** (Number of grid points in variable range used to find an optimal cut for a node splitting),

- **MinNodeSize = 5** (Minimum percentage of training events required in a leaf node),
- **PruneMethod = NoPruning** (Used for removal of statistically insignificant branches. If MaxDepth is small, there is no need to use a pruning method).

For NN, we first check the default hyperparameter setting from TMVA. Default hyperparameters:

- **Types = TMlpANN** (MVA method, ROOT's own Artificial Neural Network),
- **Learning Method = BFGS** (The Broyden-Fletcher-Goldfarb-Shannon method [8]),
- **Ncycles = 200** (Number of training cycles),
- **Hidden Layers = N, N-1** (Hidden layer architectures. N is the number of input variables. In this case, there are two hidden layers with the number of neurons N and N-1.),
- **Validation Fraction = 0.3** (Fraction of events in the training tree used for cross-validation).

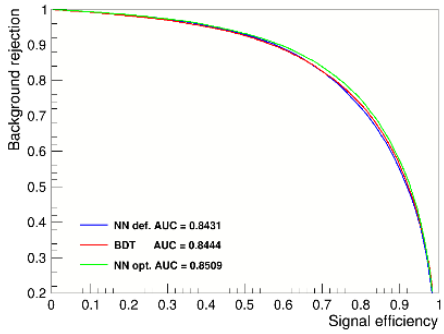
We first compare the performance of NN with the default hyperparameter setting with the performance of BDT. We use the area under the ROC curve (AUC) as a performance parameter. The performance of NN and BDT is affected not only by the hyperparameter configuration, but also by the number of events and the number of input variables. Since both NN and BDT use the same input variables, in addition to tuning the hyperparameters, we can change the number of events to improve performance. For this purpose, three training options are used with the number of signal events: 50k, 100k and 400k and the same number of background events. For each training case, the ROC curve was obtained and the AUC was determined. Fig. 3 shows the comparison of ROC curves and AUC values for BDT and NN. From the comparison of AUC values, we can see that BDT outperforms NN with default settings in every training case.

The following combinations of hyperparameter values were checked to improve the performance of the NN:

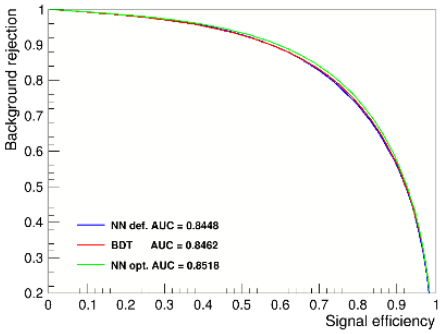
- **Ncycles** = 200, 500, 1000, 2000, 3000, 4000
- **Hidden Layers** = N,N-1; N,N; N,N+1; N,N-1,N-2
- **Validation Fraction** = 0.3, 0.5.

The rest of the hyperparameters remain unchanged in training.

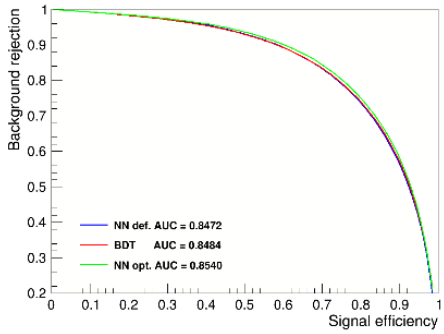
Among these hyperparameter combinations, the following combination was chosen to provide the best NN performance for all three cases: Ncycles = 2000, Hidden Layers = N, N-1, Learning Method = BFGS, Validation Fraction = 0.5. As can be seen from Fig. 3, with these optimal hyperparameter combinations, the performance of NN outperforms BDT in all cases. In this case, overtraining is not observed, this can be seen from Fig. 4, the overtraining histograms that compares the training and test distributions of NN and BDT responses.



(a)

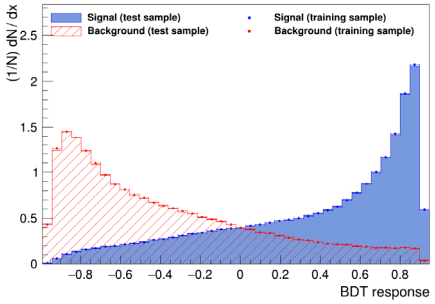


(b)

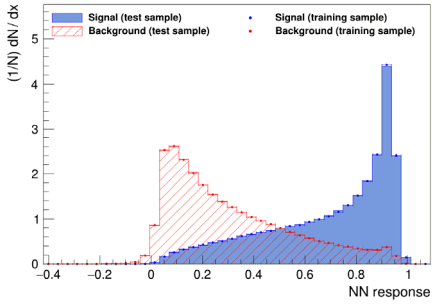


(c)

Fig. 3. ROC curves and AUC values for the NN with the default hyperparameter setting (blue curve), for the BDT (red curve), and for the NN with the optimized hyperparameter setting (green curve). During the training, 10^5 (a), $2 \cdot 10^5$ (b) and $8 \cdot 10^5$ (c) events were used.



(**a**)



(**b**)

Fig. 4. Correlation matrices for signal (a) and background (b).

5. Conclusion

The $ZH(b\bar{b})$ process was analyzed using two multivariate analysis methods BDT and NN. More than one and a half million signal and background Monte Carlo events were used for analysis. The most effective BDT hyperparameter settings optimized in the analysis of ATLAS data were used. Different numbers of events

were trained and different settings for NN were tested to obtain performance exceeding BDT performance. It was found that as the hyperparameters of NN, when we take the number of training cycles to be 2000, two hidden layers with the number of nodes equal to the number of input variables in the first and one less in the second, use BFGS as the training method, and the fraction of events for validation take 0.5, then we get the maximum AUC value. After NN optimization, the AUC values are 0.8509, 0.8518, and 0.8540 for the 100k, 200k, and 800k training events, respectively, which are higher than the AUC values for the BDT (0.8444, 08462, 0.8484). Even though NN is superior to BDT, the only problem with training in NN is that it requires more computing resources than BDT.

References

[1] ATLAS Collaboration, Phys. Lett. B 716, 1 (2012). https://doi.org/10.1016/j.physletb.2012.08.020

[2] CMS Collaboration, Phys. Lett. B 716, 30 (2012). https://doi.org/10.1016/j.physletb.2012.08.021.

[3] A. Djouadi, J. Kalinowski, M. Spira, Comput. Phys. Commun. 108 (1998) 56.

[4] Y. Freund and R. Schapire, Experiments with a new boosting algorithm, in Machine Learning, Proceedings of the Thirteenth International Conference (ICML), M. Kaufmann, (1996).

[5] Warren McCulloch and Walter Pitts, 1943, Bulletin of Mathematical Biophysics 5:115–133.

[6] ATLAS Collaboration, Phys. Lett. B 786, 59 (2018). https://doi.org/10.1016/j.physletb.2018.09.013.

[7] A. Hoecker et al., TMVA - Toolkit for Multivariate Data Analysis, PoS(ACAT)040 [physics/0703039].

[8] C.G. Broyden, J. Inst. of Math. and App. 6, 76 (1970); R. Fletcher, Computer J. 13, 317 (1970); D. Goldfarb, Math. Comp. 24, 23 (1970); D.F. Shannon, Math. Comp. 24, 647 (1970).