



Reconstruction of missing resonances combining nearest neighbors regressors and neural network classifiers

Alexandre Alves^{2,1,a}, C. H. Yamaguchi^{2,b}

¹ Departamento de Física, Universidade Federal de São Paulo, Diadema 09913-030, Brazil

² Instituto de Física, Universidade de São Paulo, R. do Matão 1371, São Paulo 05508-090, Brazil

Received: 16 March 2022 / Accepted: 12 August 2022 / Published online: 24 August 2022
© The Author(s) 2022

Abstract Neutrinos, dark matter, and long-lived neutral particles traverse the particle detectors unnoticed, carrying away information about their parent particles and interaction sources needed to reconstruct key variables like resonance peaks in invariant mass distributions. In this work, we show that a k -nearest neighbors regressor algorithm combined with deep neural network classifiers, a k NNNN, is able to accurately recover binned distributions of the fully leptonic WW mass of a new heavy Higgs boson and its Standard Model backgrounds from the observable detector level information at disposal. The output of the regressor can be used to train even stronger classifiers to separate signals and backgrounds in the fully leptonic case and guarantee the selection of on-mass-shell Higgs bosons with enhanced statistical significance. The method assumes previous knowledge of the event classes and model parameters, thus suitable for post-discovery studies.

1 Introduction

Uncovering the nature of dark matter and neutrinos will undoubtedly reveal a deeper structure of fundamental physics. If dark matter exists, it should permeate the universe and fly by our detection devices, just like the neutrinos do; however, detecting them already proved to be a challenging task. An alternative is to produce dark matter and neutrinos in large colliders and design detectors to infer their properties to get clues about the underlying structure of the physical laws.

Multi-purpose detectors, like ATLAS [1] and CMS [1], can accurately detect many types of particles like photons, electrons, muons, and hadrons, but not neutral weakly interacting particles, like neutrinos and dark matter. This fact poses a problem to the particular quest for new physics man-

ifesting as dark states. The escape of neutrinos out of the detectors prevents us from performing some key observations that could benefit from low backgrounds. For example, the Higgs boson mass and width could be even more accurately measured if the information from fully leptonic $WW, ZZ \rightarrow \ell^+ \ell'^- \nu_\ell \bar{\nu}_{\ell'}$, $\ell(\ell') = e, \mu$, modes were recoverable. Instead, apart from $ZZ \rightarrow 4\ell$, we need to rely upon the semi-leptonic or fully hadronic modes to perform those measurements with a significantly higher level of backgrounds. Identifying bumps and sharp thresholds in the invariant mass distribution of observable and dark states would also help disentangle new physics signals like heavy Higgs bosons [2], Higgs pair production with one invisible Higgs [3,4], sleptons and charginos [5,6], and new gauge bosons decays neutrinos and/or dark matter from their associate backgrounds [7,8], to name a few possibilities. Another important example where a fully-leptonic mode benefits from a clean environment is the measurement of the scattering angles for W, Z bosons in polarization studies [9].

In processes where N_ν neutrinos are produced in the hard scattering, there are $4N_\nu$ unknowns that should be recovered to reconstruct the parent particles. The missing transverse momentum, calculated from the imbalance in the sum of visible transverse momenta of reconstructed physics objects, furnishes two constraints, despite not exactly equal the sum of neutrinos transverse momentum due to detector effects, contamination from neutrinos, and other missing particles from hadronic jets, for example. Mass constraints must provide the complementary information necessary for reconstruction. The number of mass constraints N_m other than N_ν : $p_V^2 = 0$, from neutrinos, is process dependent though, and in many cases, they do not suffice to recover the four momenta of the neutrinos if $3N_\nu \geq N_m + 2$. Even in cases where sufficient mass constraints exist, like fully leptonic $t\bar{t}$ signals [10,11], the misreconstruction of the neutrinos transverse momentum, combinatorial particle assignment, and ambiguities aris-

^a e-mail: aalves@unifesp.br (corresponding author)

^b e-mail: carlos.yamaguchi@usp.br

ing from the quadratic nature of the equations do not guarantee meaningful solutions for all events.

In a process-independent way, one approach to circumvent the impossibility of recovering the four-momenta of all the escaping particles is to design kinematic variables and methods that correlate with the lost information, for example, with the masses of the parent particles. Many such variables are smartly crafted to provide useful hints about decaying particles in many situations [12–22]. Yet, none of them, by construction, is capable of recovering a resonance peak.

Another possible approach is using a regression algorithm to predict the neutrinos four-momenta or some variable of interest from the observed information. One might tackle tasks of that type by training an algorithm to parameterize a multivalued function $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$, a neural network, for example [23–26]. Methods of density estimation [27] might also be useful.¹ As a matter of fact, in Ref. [31], neural network regressors were used, in the framework of the standard model, to reconstruct the lepton polar angle defined in the W rest frame from leptonic or semi-leptonic decays in vector boson scattering at the LHC. The fractions of transversely and longitudinally polarized W bosons can then be inferred from the measurement of the lepton polar angle distribution. An additional difficulty in the case where we are interested in detecting a new particle, is that we need to infer the signal parameters, mainly its mass, prior to the training of the regressors. Despite this inference could be performed at the same time of the kinematic variable regression using some conditional model, we will assume that it is performed directly from data.²

Assuming the previous knowledge about the signal resonance, its mass and possibly its width, the most straightforward approach to reconstructing a mass variable involving escaping neutrinos is by interpolating a support set of events from simulations instead of adjusting the parameters of some function that should generalize from training to test datasets. Such an accurate and efficient algorithm for supervised regression is the k -nearest neighbors algorithm, as we will demonstrate in this work. As we argued, the caveat of this approach, like any other supervised regression algorithm, is that we need to know what type of event is produced in the collisions beforehand to select the correct support set for interpolation of the variable. Our approach takes advantage of the exquisite power of neural networks to classify the events. In principle, it is possible to identify signal events without any previous knowledge using outliers detection and unsupervised methods; however, as we discussed, without

knowing the mass parameters, reconstructing a mass peak is challenging.³

In this work, we show how to combine neural networks for classification and k NN for regression is useful in reconstructing a new heavy Higgs boson decaying to $W^+W^- \rightarrow \ell^+\ell'^- + \nu_\ell\bar{\nu}_{\ell'}$, $\ell(\ell') = e, \mu$, a fully leptonic final state with two escaping neutrinos, and its main SM backgrounds. We will show that the predicted mass of the charged leptons and neutrinos can be reliably used as a powerful new attribute to clean up the backgrounds further while enabling the selection of on-mass shell Higgs bosons.

The work is organized as follows. In Sect. 2, we describe the k NN regression algorithm; in Sect. 3, we provide details of the combined construction of regressors and classifiers to identify the heavy Higgs boson and its main SM backgrounds, while in Sect. 4 we present our final results in terms of improvement of the statistical significance of the signal hypothesis; Sect. 5 is devoted to conclusions and prospects.

2 Details of the k NN regression

The k -nearest neighbors regressor [34] is a simple but effective algorithm for interpolation. First of all, we define a support dataset $\mathcal{S} = \{(\mathbf{X}_i, F(\mathbf{X}_i)), i = 1, \dots, N_s\}$, these are the exemplars which will be used to predict the value of the function of interest. Second, we define a distance metric, $Dist(\mathbf{X}, \mathbf{Y})$, to decide which exemplars of \mathcal{S} are closer to a new point, \mathbf{X}_{new} , where \mathbf{X} , in our case, is a \mathbb{R}^n vector. Third, we choose how many nearest neighbors to \mathbf{X}_{new} will be used to compute $F(\mathbf{X}_{new})$, the target of our regression, according to a weighted mean

$$F(\mathbf{X}_{new}) = \frac{\sum_{m=1}^k F(\mathbf{X}_m)/Dist(\mathbf{X}_{new}, \mathbf{X}_m)}{\sum_{m=1}^k 1/Dist(\mathbf{X}_{new}, \mathbf{X}_m)}. \quad (1)$$

Substituting $Dist(\mathbf{X}_{new}, \mathbf{X}_m) = 1$ in the formula above corresponds to an arithmetic mean estimator for F . The weighted or arithmetic option will be decided in the tuning stage of the analysis.

In principle, once we have chosen the distance metric, the number of nearest neighbors, k , used to compute $F(\mathbf{X}_{new})$ is the only hyperparameter of the algorithm. Note that this model has no parameters to be adjusted contrary to a neural network. This is the reason we do not need a training phase. However, the distance metric, k , and possibly other hyperparameters should be adjusted to get a good regressor by minimizing some error function. All $F(\mathbf{X}_m)$, $m = 1, \dots, N_s$ are known thus, we are in the realm of supervised learning.

¹ For more regression algorithms and applications, see [28–30].

² An interesting possibility is using a Variational Autoencoder for Regression as in Ref. [32].

³ Regression with unlabeled data is possible when the marginal distribution of the target is known [33].

In our case, the target function of the regression, F , is the leptonic $\ell^+\ell'^-\nu_\ell\bar{\nu}_{\ell'}$, invariant mass, $M_{\ell\ell\nu\nu}$. The input of this function is the observable information obtained from the electrons and muons four-momenta, p_e and p_μ , respectively. The representation of the events was chosen as the energies and 3-momentum of the charged leptons plus high level functions construed from that low level information: $\mathbf{X} = (f_{ij}(p_\ell, p_{\bar{\ell}}), i = 1, \dots, N_{ev}, j = 1, \dots, M)$ representing N_{ev} events with M features.

If the number of dimensions of the features space is large, distance-based models like k NN might perform poorly. For that reason, it is usual to project the features space onto a latent space of reduced dimensionality. There are various ways to do that. We chose to linearly transform the original features using a principal component analysis (PCA) [35] and looking for the nearest neighbors in the transformed space of the first $P < M$ variables which best explain the variance of the data, $\mathbf{X}^{pca} = T_P(\mathbf{X})$. We also adjust P to obtain the best regressors. Let us now construct the regressors for the signal and the backgrounds.

3 Reconstruction of fully leptonic resonances

The dataset consists of 400,000 simulated signal events $pp \rightarrow H_2 \rightarrow W^+W^- \rightarrow \ell^+\ell'^- + \nu_\ell\bar{\nu}_{\ell'}$, $\ell(\ell') = e, \mu$, where H_2 is a new Higgs boson produced via gluon fusion, for each one of the three different mass values: 1, 1.5 and 2 TeV and two fixed total H_2 width, 1% and 10% of the mass parameter, totaling 2.4 million signal events. The dataset also contains 5.2 million of the corresponding SM backgrounds.

Our goal is twofold: (1) to show that the resonance can be reliably reconstructed, and (2) that using it can boost both ML classifiers' accuracy (and other metrics), and more importantly, the signal significance compared to a baseline classifier without the $M_{\ell\ell\nu\nu}$ regression. The actual value of the statistical significance depends on the number of signal events, which is model dependent. Because we are interested in showing how much the signal significance increases after using k NNNN compared to a baseline analysis, we fix the number of signal events to illustrate our method. Our sole supposition is that the leptons plus neutrinos signals are dominated by the WW mode with negligible interference with the corresponding SM backgrounds.⁴

We consider the following background sources in our analysis: (1) the dominant irreducible component, $pp \rightarrow W^+W^-$, (2) the subdominant irreducible, $pp \rightarrow ZZ(\gamma^*)$, (3) the dominant reducible contribution, $pp \rightarrow t\bar{t} \rightarrow W^+W^-b\bar{b}$. All the signals and backgrounds partonic events

are simulated at leading order using MadGraph5 [36]. Hadronization is simulated with Pythia8 [37], while detectors effects are simulated with Delphes3 [38]. We generate around 1.3 million events for each one of those background classes.

The partonic events are used to obtain the ground truth $M_{\ell\ell\nu\nu}$ distributions once the neutrinos momenta are available. Note that this distribution explicitly assumes that missing energy is all due to escaping neutrinos produced in the hard scattering, but not the misreconstruction of observable momenta or the missing of other particles. However, the leptons momenta and the event's missing energy, which feed the algorithms, include all the simulated effects. Therefore, part of the mismatch between the k NN prediction and the true distributions can be explained this way. As we are going to see, however, the partonic predictions present good precision.

We adopt the following basic acceptance cuts to select events with two opposite charge leptons and missing energy

$$p_{T,\ell} > 20 \text{ GeV}, \quad |\eta_\ell| < 2.4, \quad \Delta R_{\ell\ell} > 0.4, \\ M_{\ell\ell} > 30 \text{ GeV}, \quad \cancel{E}_T > 40 \text{ GeV}, \quad |\Delta\eta_{\ell\ell}| < 3.0, \quad (2)$$

where $p_{T,\ell}$ and η_ℓ denote the transverse momentum and pseudo-rapidity of the leptons, respectively, while $M_{\ell\ell}$, \cancel{E}_T and $\Delta R_{\ell\ell}$ denote the invariant mass of the charged leptons, the missing transverse energy and the distance in the $\eta \times \phi$ plane of the event. The last cut, on the rapidity gap between the charged leptons, was imposed to suppress weak boson fusion backgrounds, which are neglected in the subsequent analysis. The $M_{\ell\ell}$ helps to suppress the low mass leptons backgrounds from $Z\gamma^*$, which showed to be a source of contamination among the events classes.

In Fig. 1, we show the distributions of some features chosen to represent the events and predict their classes and $M_{\ell\ell\nu\nu}$. Along with the energies and the components of the 3-momenta of the charged leptons, we also include their transverse momentum, and the following variables:

- $M_{\ell\ell}$, the charged leptons invariant mass,
- \cancel{E}_T , the missing transverse energy,
- $\Delta R_{\ell\ell} = \sqrt{(\Delta\eta_{\ell\ell})^2 + (\Delta\phi_{\ell\ell})^2}$, where $\Delta\eta_{\ell\ell}$ and $\Delta\phi_{\ell\ell}$ represent the rapidity and azimuthal angle differences between the charged leptons,
- $\cos\theta^* = \tanh\left(\frac{\Delta\eta_{\ell\ell}}{2}\right)$, proposed in Ref. [39],
- $\sqrt{\hat{s}}(0) = \sqrt{E_{\ell\ell}^2 - p_{T,\ell\ell}^2} + \cancel{E}_T$, proposed in Ref. [15],
- $M_{Reco} = \sqrt{M_{\ell\ell}^2 + 2p_{T,\ell\ell}\sqrt{M_{\ell\ell}^2 + p_{T,\ell\ell}^2} - p_{T,\ell\ell}}$, where $p_{T,\ell\ell}$ is the transverse momentum of the charged lepton pair,
- the number of jets tagged as a bottom jet to suppress $t\bar{t}$ events.

⁴ A non-negligible interference with the SM Higgs boson is expected with wide scalar resonances of masses below 1 TeV or so. This should not pose any difficulties for the k NN regression, however.

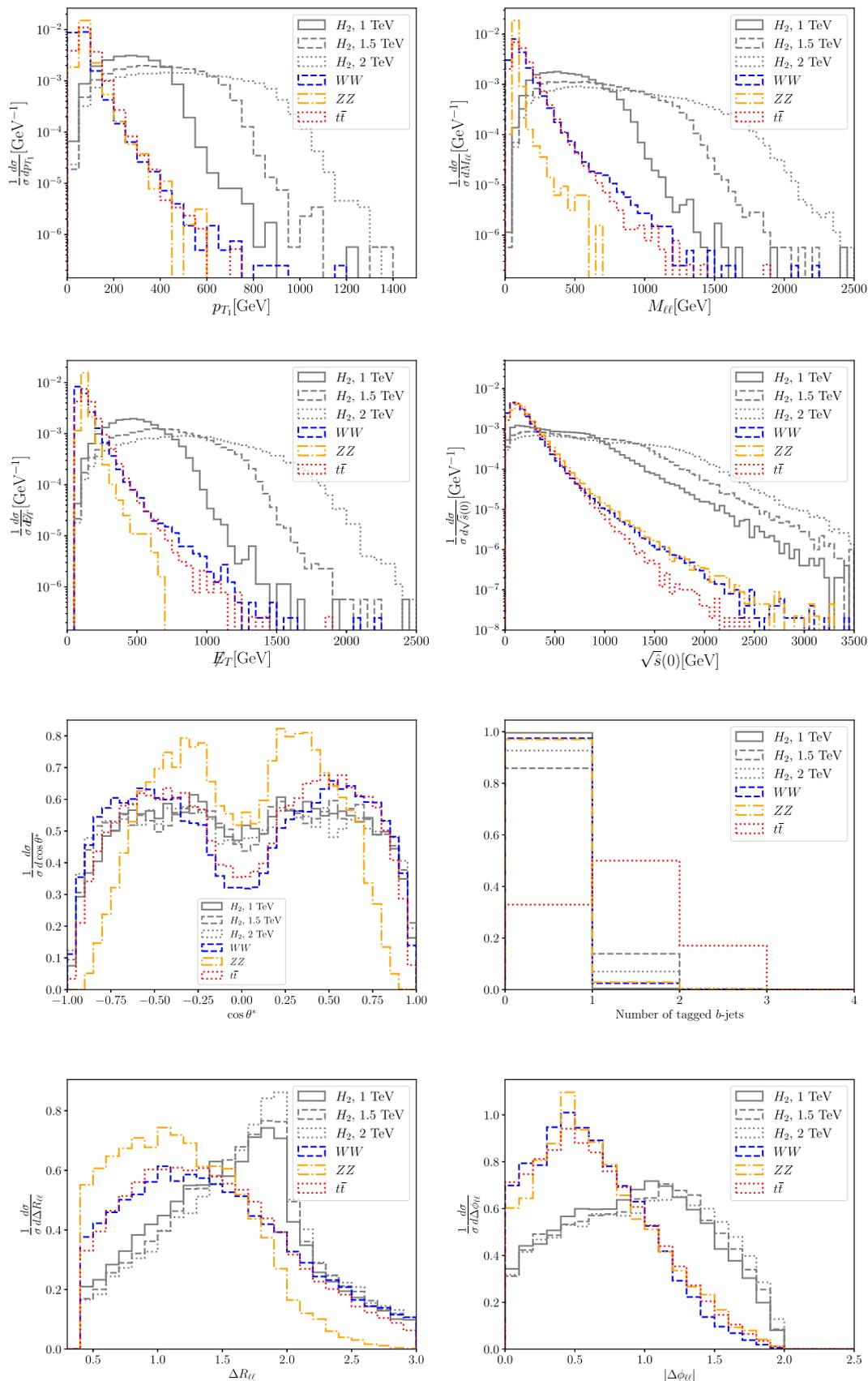


Fig. 1 Some of the kinematic distributions of Higgs bosons and its corresponding SM backgrounds chosen to represent the events for regression and classification

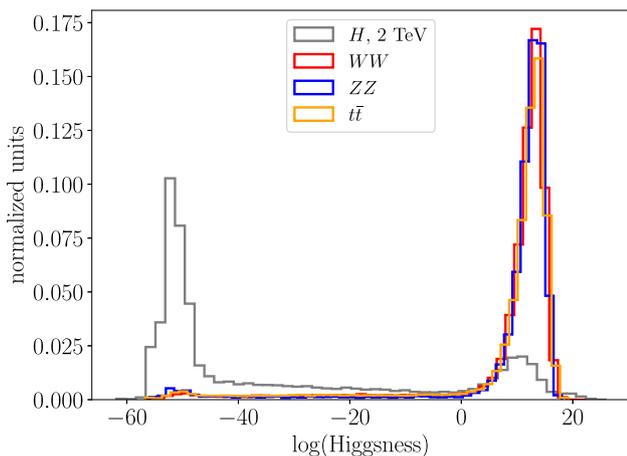


Fig. 2 The logarithm of the Higgsness variable defined in Eq. (3) for a 2 TeV Higgs boson and its SM backgrounds

Beside these kinematic variables, we also constructed the *Higgsness*⁵ variable [40] to denounce the presence of a heavy Higgs boson decaying to $W^\pm W^{*\mp} \rightarrow \ell^+ \ell'^- + \nu_\ell \bar{\nu}_{\ell'}$. The idea is to search for the neutrinos 4-momenta of an event which minimize

$$\text{Higgsness} \equiv \underset{p_{\nu}, p_{\bar{\nu}}}{\text{argmin}} \left[\frac{(M_{\ell^+ \ell'^- \nu \bar{\nu}}^2 - m_H^2)^2}{\delta_H^4} + \min \left(\frac{(M_{\ell^+ \nu}^2 - m_W^2)^2}{\delta_W^4}, \frac{(M_{\ell'^- \bar{\nu}}^2 - m_W^2)^2}{\delta_W^4} \right) \right], \tag{3}$$

where δ_H and δ_W , in principle, represent experimental uncertainties, but for our purposes, they can be treated as free parameters. In fact, the value of these parameters matters for the Higgsness distributions, and we adjust them for maximum discernment among the classes.

In Fig. 2, we show the distribution of the logarithm of Higgsness for a 2 TeV Higgs boson and the *WW*, *ZZ* and *t \bar{t}* backgrounds. We used a simplex algorithm from SciPy [41] to search for the minimum of the Higgsness variable. As expected, Higgsness is very small for signal events, while it is much bigger for a background event.

For each class, we construct a regressor function according to Eq. (1). At this stage, we employed 0.9 and 1.2 million events for signals and backgrounds, respectively. To ensure that the dataset’s size would not play a role in the results, we separated 80% of that data for tuning the regressors. We adjusted, with a grid search, the number of nearest neighbors, *k*, the distance metric⁶, *Dist*, the number of PCA trans-

formed variables, *P*, and the weighted or arithmetic option in Eq. (1) in order to minimize the mean square error between the predicted and the true binned $M_{\ell\ell\nu\nu}$ distributions. The space of hyperparameters in the grid search is the following

$$k \in [1, 5], \quad \text{Dist} \in \{\text{Minkowsky, Manhattan, Chebyshev, Canberra, Braycurtis}\}, \\ P \in [1, 8], \quad \text{weight} \in \{\text{uniform, weighted}\}. \tag{4}$$

We display, in Fig. 3, some results of the tuning of the number of nearest neighbors, *k*, and the number of principal components to demonstrate the quality of the *k*NN regression for the cases of the SM *WW* background and a 2 TeV Higgs boson. The other backgrounds and signals present very similar behavior. The best hyperparameters were chosen as those with the smaller mean squared error (MSE) between the true and predicted histograms of the target variable.

The *k*NN regressor is robust against most parameter variations while being very accurate for predictions. Overall, for all backgrounds and the signals, the nearest neighbor to a new point in the latent space of PCA transformation is the most accurate prediction for our target variable. We tested various alternatives to *k*NN as gradient boosting and neural networks regressors, and the nearest neighbors approach showed itself superior in approximating the true distribution of masses. We also found that neural networks present an improved generalization performance across classes compared to other algorithms, especially the *k*NN algorithm, which is very dependent on the class of the event. For example, we found that training a neural network with *WW* backgrounds might be useful to obtain $M_{\ell\ell\nu\nu}$ for the other classes, especially the backgrounds, but its performance on signal events is still not competitive with much simpler proxy variables that correlate with the resonance mass, as $\sqrt{\hat{s}}(0)$ [15] and other transverse mass variables. The significant advantage of algorithms with good generalization performance is being agnostic towards the other classes, depending less on the previous knowledge of the types of events.

The number of PCA dimensions where the original data representation is projected onto showed a more significant variation. While for the *ZZ* background and the 1 TeV Higgs, the smaller MSE could be reached with just a one-dimensional latent space, the *WW* background performed better in a two-dimensional PCA space, the *t \bar{t}* and a 1.5 TeV Higgs with 3 PCA dimensions, and the 2 TeV Higgs with 6 PCA dimensions. We thus observe that as the particles get heavier, the higher should be the dimension of the PCA space. The choice of the distance metrics has no impact on the performance of the algorithms once the uniform weights performed better than the weighted option in all experiments.

⁵ Topness was another variable proposed in this reference to better tag *t \bar{t}* events.

⁶ For a good account on *k*NN and its options, including the distance metrics available, see sklearn page. Our results are, in fact, insensitive

Footnote 6 continued
to the distance metric option as explained in the text, we show them just for completeness.

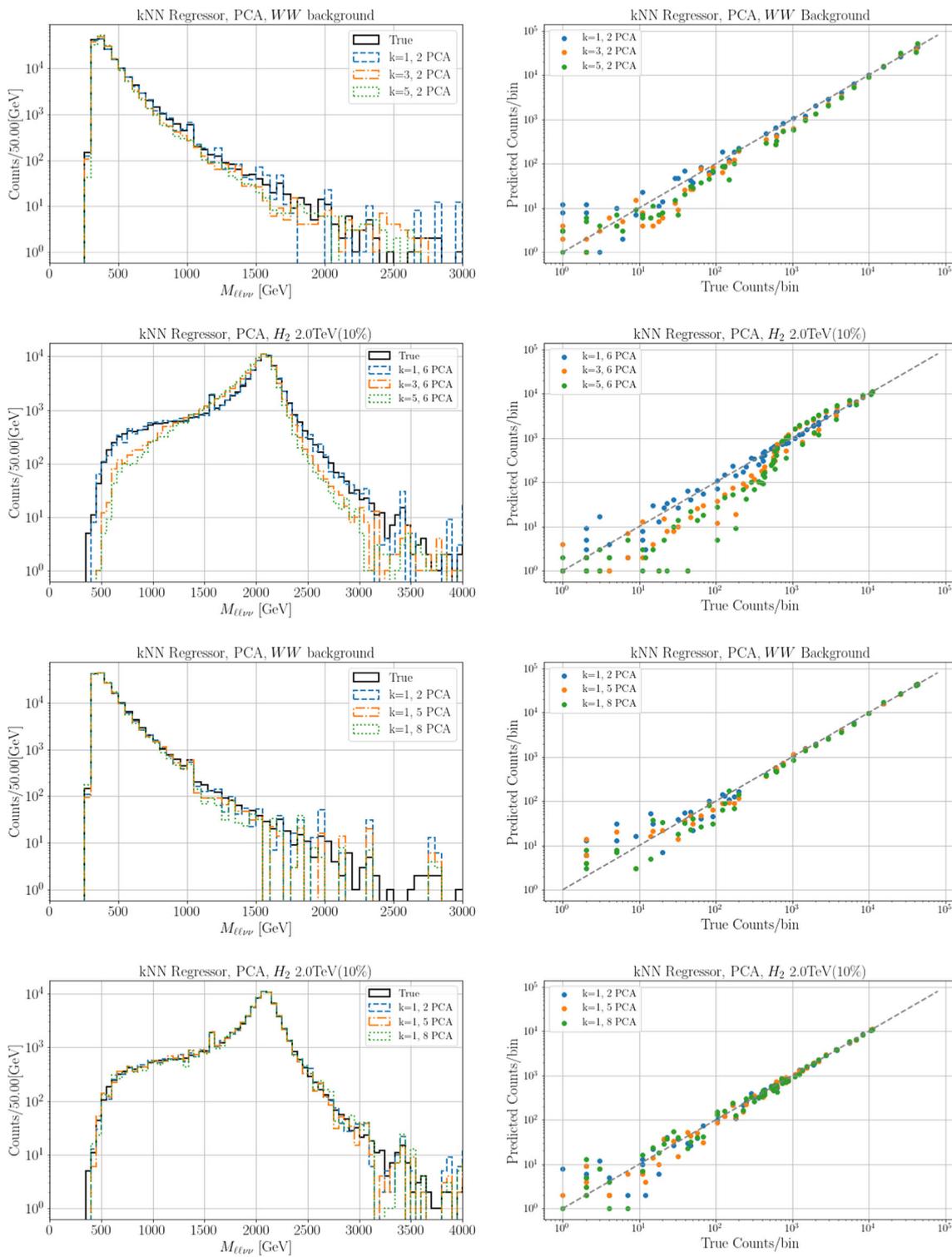


Fig. 3 Results for the tuning of the number of nearest neighbors, k , and number of principal components (PCA) of the latent space. In the four upper panels we display $k = 1, 3$, and 5 , keeping PCA fixed at its best value. In the four lower panels we display PCA = $2, 5$, and 8 , keeping k fixed at its best value

It means that the prediction is a simple arithmetic mean of the nearest neighbors of a given point projected on the principal component space of the events. We also tested non-

linear transformations to the latent space as TSNE, but with marginal gains at the cost of much longer computation time.

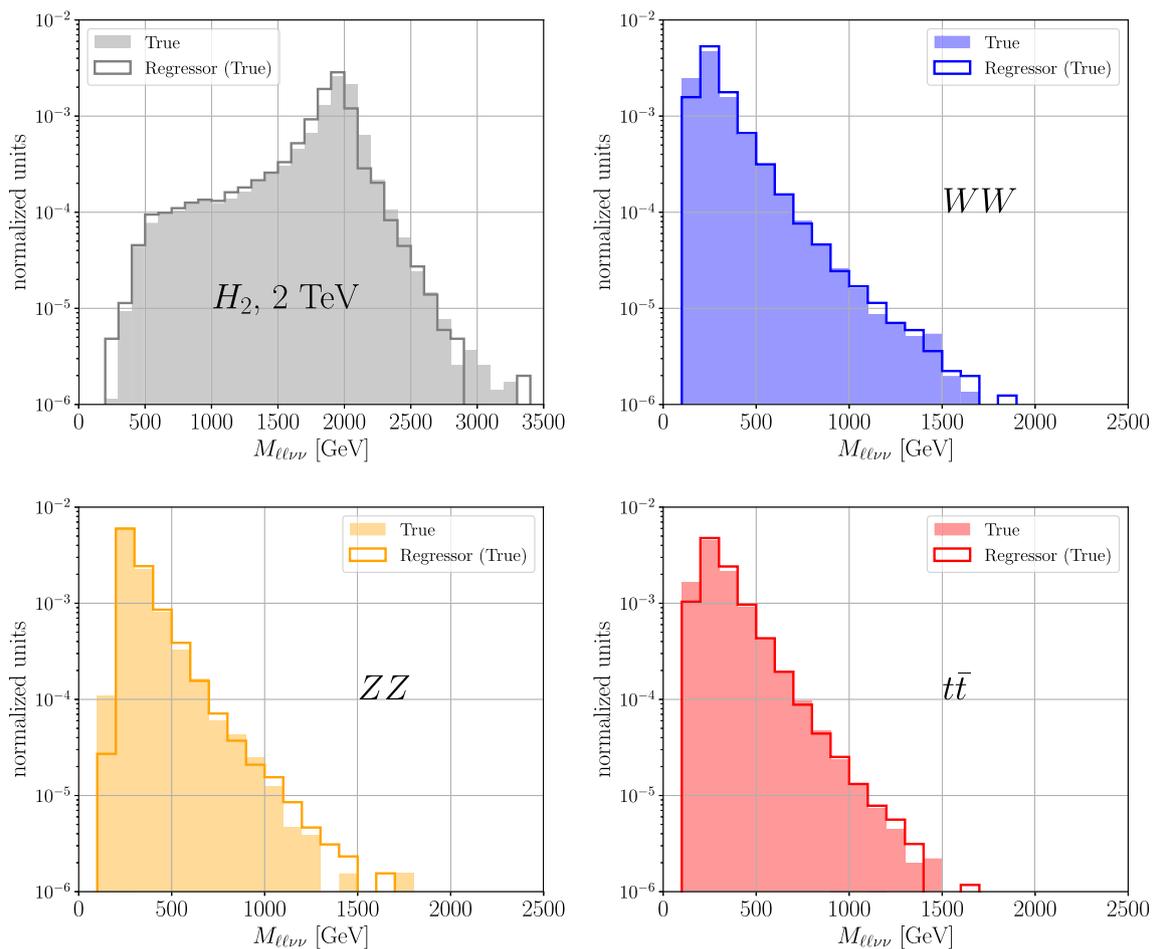


Fig. 4 The true (shaded areas) and predicted (solid lines) $M_{\ell\nu\nu}$ distributions for the 2 TeV Higgs (upper left), WW (upper right), $ZZ(\gamma^*)$ (lower left) and $t\bar{t}$ background (lower right). The regression is based on true samples, in this case

In Fig. 4, we display the true and the predicted $M_{\ell\nu\nu}$ masses for a 2 TeV Higgs boson, with $\Gamma_H/m_H = 10\%$, and the WW , $ZZ(\gamma^*)$, and $t\bar{t}$ backgrounds. As we see, the regressors work very well for each class of events. We checked that the width of the resonances affects too little the accuracy of the regression from $\Gamma_H/m_H = 1\%$ up to 10% .

3.1 Pre-regression classification

The construed $M_{\ell\nu\nu}$ regressor of a given class can predict the target distribution of events that pertain to that class exclusively. If one feeds a background regressor with a signal event, for instance, the background regressor will find the target value of the background distribution, which is closer to the signal event. In order to predict the classes' targets correctly, we need first to predict the classes as accurately as possible. We also need to know the mass of the resonance.

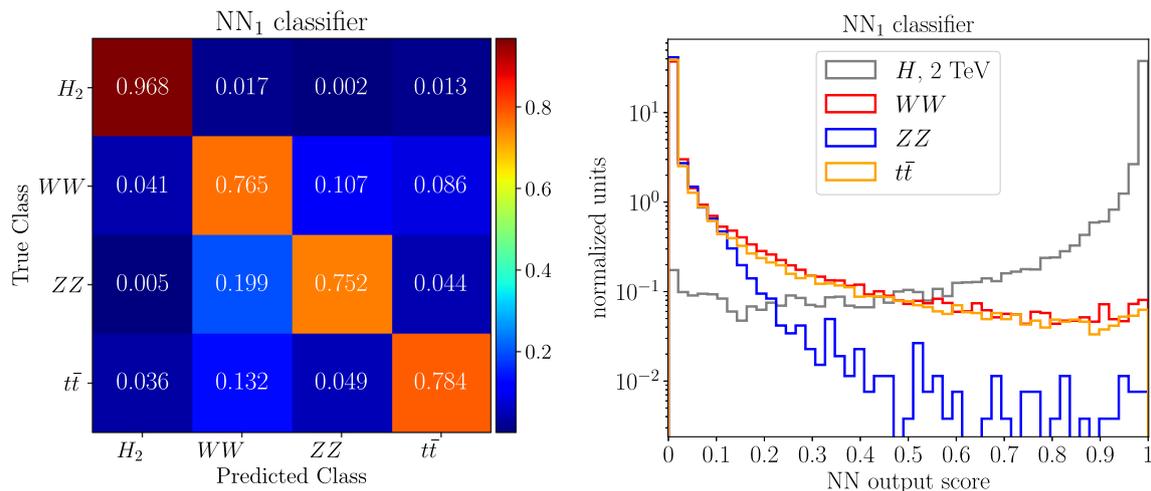
The classification of events was performed with neural networks (NN) [30, 42] based on the same features used for regression. We took 1.5 million signals and 4 million back-

grounds events to tune, train and test the algorithms. As we will discuss later, this body of data was further split to independently adjust, train, and test a second neural network; that is why we need such a large number of simulations.

We used Keras [43] with the Tensorflow2.0 [44] backend to build multiclass NN classifiers. The tuning of the architecture and hyperparameters were done with 30 Hyperopt [45] runs. An initial learning rate was adjusted following a schedule halving every ten epochs. The training was halted if no improvements on the validation loss were observed over 20 epochs or a maximum of 100 epochs was reached. The model delivering the smaller validation loss was selected during the training phase. We trained different models to identify the Higgs boson of 1, 1.5, and 2 TeV masses. The hyperparameters and the neural network architectures are shown in Table 1. We split the data in proportion to 70%, 20%, and 10% for training, testing, and validation of the classifiers, respectively. What we learn is that the 1 TeV Higgs bosons need a more regularized model to be discerned from backgrounds in the test samples compared to heavier masses

Table 1 Hyperparameters and architecture of the neural network classifiers to clean Higgs boson signals of 1, 1.5 and 2 TeV masses from its SM backgrounds. No dropout layers were needed in the 1.5 and 2 TeV cases

Hyperparameter/arch	Search space	1 TeV	1.5 TeV	2 TeV
L2 regularization	$[10^{-8}, 10^{-4}]$	5.5×10^{-6}	4.7×10^{-8}	3.3×10^{-8}
Initial learning rate	$[10^{-8}, 10^{-2}]$	2×10^{-3}	8.2×10^{-3}	9.6×10^{-3}
Batch size	[32, 64, 96, 128, 160, 256, 384, 512, 1024]	160	160	128
Dropout×rate	[True,False]×[0.0, 0.9]	0.03	–	–
Weight initialization	[Uniform,Normal]	Normal	Uniform	Uniform
Layer activation	[ReLU,Tanh,Softplus]	Tanh	ReLU	Tanh
Number of layers	[1, 2, 3, 4]	4	4	4
Number of neurons/layer	[32, 64, 96, 128, 160, 256, 384, 512, 1024]	(64, 32, 16, 8)	(160, 80, 40, 20)	(128, 64, 32, 16)
Total of parameters		4188	20,544	13,748

**Fig. 5** Confusion matrix of the classification and the output scores of the neural network before $M_{\ell\ell\nu\nu}$ regression at the left and right panels, respectively, for a 2 TeV Higgs boson and its main SM backgrounds

with a stronger L2 regularization, dropout layers, and a less complex architecture. It reflects the fact that it is harder to separate lighter resonances from the SM backgrounds.

In Fig. 5, we display the confusion matrix of the NN classifier (let's call it NN₁) trained to recognize the signals of a broad 2 TeV Higgs boson resonance, with $\Gamma_H = 200$ GeV, against the WW , ZZ and $t\bar{t}$ events at the left panel, and the output scores of each class at the right panel. As expected, WW and $t\bar{t} \rightarrow W^+W^- + b\bar{b}$ events are more frequently mistagged by the classifier, with 13(11)% of the $t\bar{t}(WW)$ sample tagged as a $WW(t\bar{t})$ event. Looking at Fig. 1, we indeed see that WW and $t\bar{t}$ events look similar once the decay of the top quark to a W boson plus a b -jet. On the other hand, around 1/3 of all $t\bar{t}$ events have no tagged b -jets, the most important discriminant against W pair production. This similarity is summarized in the right panel of Fig. 5 where we see that the scores distributions of WW and $t\bar{t}$ events overlap.

However, the most mistagged class is ZZ , where 19% of the sample is classified as a WW event. On the other hand, only 3% of signal events are wrongly assigned to background classes. The same behaviour was observed for the other two mass values. This somewhat large misidentification of $ZZ(\gamma^*)$ events might be explained by the introduction of Higgsness as a feature of the dataset. As we see in Fig. 2, while being very powerful to discern the signals, Higgsness is very similar for background classes. Withdrawing Higgsness from the data representation decreases the true positive rate of the 2 TeV Higgs boson from 97 to 94%, while also decreasing the proportion of ZZ events to be labeled as WW events from 19 to 6%. Apparently, singling out signal events with Higgsness make the background classes less discernible among themselves.

With the NN classifier in hand, we can reconstruct the $M_{\ell\ell\nu\nu}$ mass of the events. We emphasize that it is necessary to know the class of the events before the regression once the target variable can only be correctly estimated when interpo-

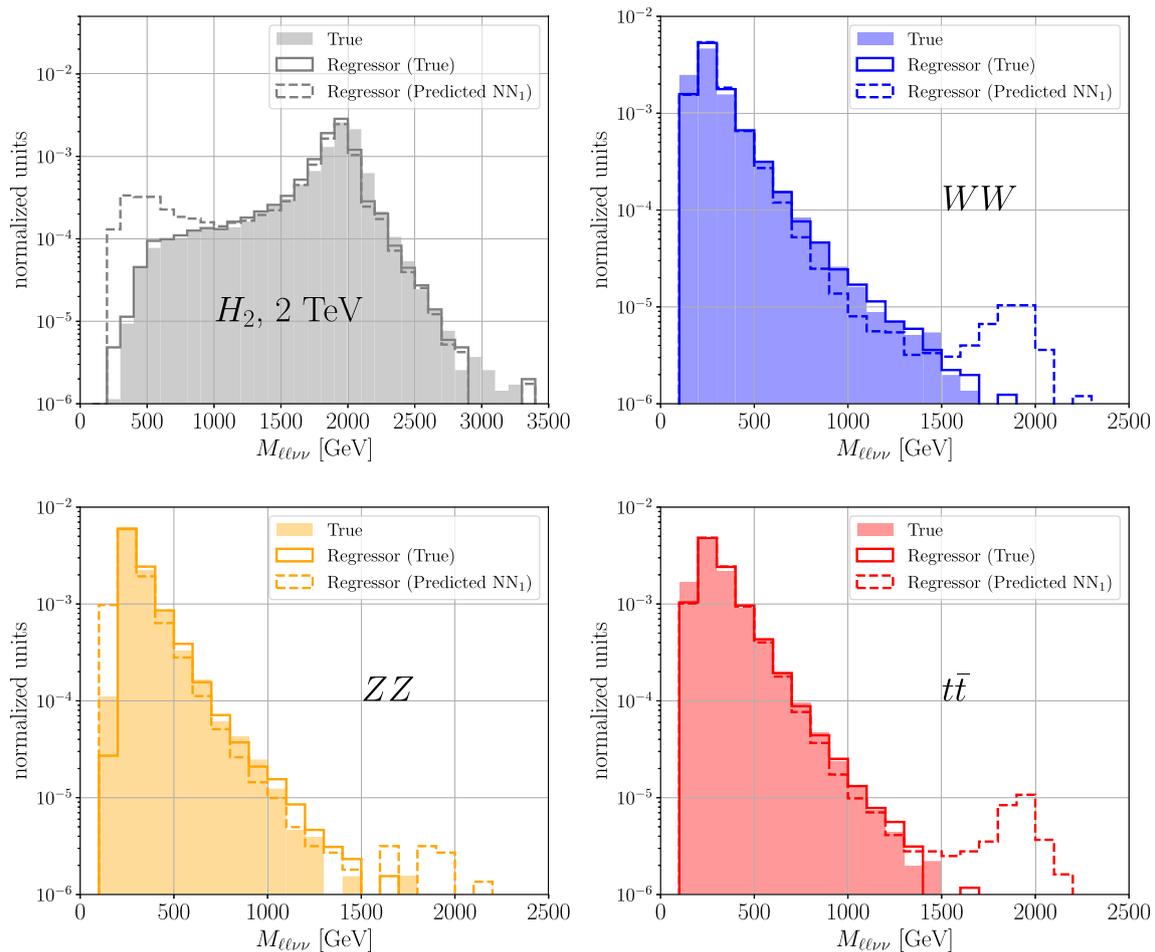


Fig. 6 The true (shaded areas), regressed from true samples (solid lines), and regressed from samples identified with NN1 (dashed lines) $M_{\ell\ell\nu\nu}$ distributions for the 2 TeV Higgs (upper left), WW (upper right), $ZZ(\gamma^*)$ (lower left) and $t\bar{t}$ background (lower right)

lated over the proper support dataset of the k NN algorithm. In other words, the nearest neighbors regressor does not generalize from one class to another.

If one presents instances never seen by the regressor, the lack of necessary correlations will result in meaningless outputs. For example, in the Higgs rest frame, the sum of the charged leptons energy and the energy of the neutrino equals the Higgs mass, $E_{\ell\ell}^* + E_{\nu\nu}^* = m_H$. In this case, the regressor can only learn the simple relation $E_{\nu\nu}^* = m_H - E_{\ell\ell}^*$ to recover the missing information from the observed one if it is trained on signal events with known m_H .

In Fig. 6, we show the predicted $M_{\ell\ell\nu\nu}$ mass of the events classified by the neural network model for a 2 TeV Higgs. Again, the results for other masses and total widths are nearly the same. We note clear contamination by signal events in the tail of the distributions for WW and $t\bar{t}$ events. This is expected, once 1.7 and 1.4% of signal events are classified as WW and $t\bar{t}$ events, respectively. Only 0.19% of H_2 events are classified as ZZ events, though and that’s why we do not observe a clear peak in the tail of the ZZ distribution. By

its turn, 4.1 and 3.7% of WW and $t\bar{t}$ sample, respectively, is mistagged as a signal event, populating the low mass bins of the H_2 distribution above the true distribution. In practice, if one is interested in identifying Higgs bosons, requiring the score to be greater than 0.5 or larger is effective to mitigate the contamination of background distributions permitting a reliable estimate of backgrounds in the resonance region. However, the signal contamination is not affected much, yet, once the signal distribution contamination occurs for low mass bins, the resonance region estimate is also reliable.

A way around these contaminations in order to improve the confidence in the mass estimates is presented in the next section.

3.2 Post-regression classification and the k NNNN algorithm

How can we get rid of the mistagged contamination in backgrounds and signal distributions? In Ref. [46], an ensemble of classifiers was used to boost the classification accuracy of

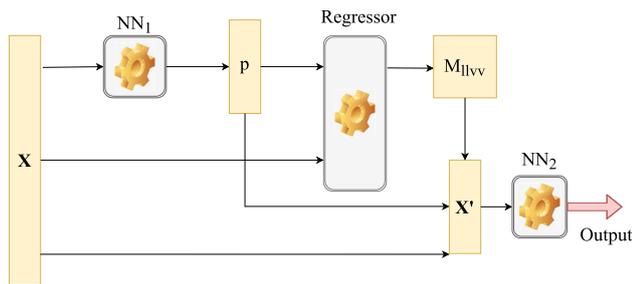


Fig. 7 Flow chart of the combined classification/regression algorithm with stacking – k NNNN for short. All kinematic variables described in Sect. 3 are passed to the Regressor except Higgsness

Higgs boson events with a performance almost as good as deep neural networks [47]. We used the same idea to boost the performance of our classifier by stacking another neural network model on the top of the first classifier described in the previous section. For a good review of ensemble methods, see [48].

We show a flowchart of our proposed algorithm from beginning to end in Fig. 7. The original dataset comprising the kinematic features, \mathbf{X} , described in Sect. 3, plus the Higgsness variable is first split into many subsets to train/validate the classifiers and the regressor. Two subsets are used to train the first classifier, depicted as NN_1 in Fig. 7, and the k NN Regressor. In this scheme, the Regressor is fed by kinematic features, but Higgsness, and also with the output scores, \mathbf{p} , provided by the NN_1 to decide what support set should be used to calculate $M_{\ell\ell\nu\nu}$ of a given event. After this stage, the algorithm has thus produced two important pieces of information, which are appended to \mathbf{X} : the scores vector, \mathbf{p} and $M_{\ell\ell\nu\nu}$, resulting in a new data representation, \mathbf{X}' . This new representation is then used to train a second neural network, NN_2 . Because of the combination of a k NN regressor with Neural Network classifiers, we call it k NNNN algorithm. Note that the output of NN_2 is the final output of the algorithm, the output of k NNNN itself.

In Fig. 8, we display the confusion matrix and the score outputs of the NN_2 classifier. The separation of the classes is improved after the second classification. To confirm that improvement, we calculate the overall accuracy and the score asymmetry defined as

$$\frac{N(\text{score} > 0.5) - N(\text{score} < 0.5)}{N(\text{score} > 0.5) + N(\text{score} < 0.5)}, \quad (5)$$

where N is the number of events of the class.

We also compute the positive and negative likelihood ratios, as defined in Ref. [49]

$$LR_+ = \frac{\text{Sensitivity}}{1 - \text{Specificity}} = \frac{\text{Sensitivity}}{\text{False Positive Rate}}, \quad (6)$$

$$LR_- = \frac{1 - \text{Sensitivity}}{\text{Specificity}} = \frac{\text{False Negative Rate}}{\text{Specificity}}. \quad (7)$$

These two metrics are aimed to measure how effective a classifier is in predicting the classes in a binary problem. Sensitivity, the ratio between the number of events correctly classified as positives and the total number of events classified as positives, measures how good the classifier is in identifying the positive class, our H_2 events. Specificity, by its turn, is the ratio between the number of events correctly classified as negatives and the total number of events classified as negatives, our backgrounds. In order to apply these metrics, we gather all background events into a single negative class. Analogously to sensitivity, specificity measures how competent the classifier is in correctly identifying negative instances.

For the signals, LR_+ summarizes how many times more likely signals are correctly predicted to be signals than backgrounds are wrongly predicted to be a signal. On the other hand, LR_- summarizes how many times less likely signals are wrongly predicted to be backgrounds than backgrounds events are correctly predicted to be a background. A better classifier must therefore maximize LR_+ and minimize LR_- . In the comparison of two classifiers, let's say, NN_1 and NN_2 , if $LR_+(NN_2) > LR_+(NN_1)$ and $LR_-(NN_2) < LR_-(NN_1)$, then NN_2 is better than NN_1 in the confirmation of both positives and negatives. When the inequality of the first condition still holds but the second flips, then NN_2 is better than NN_1 in the confirmation of positive class but worse for the negative class. At the same time, if the inequality of the first condition flips but the second still holds, then NN_2 is worse than NN_1 in the confirmation of positive class but better for the negative class.

In Table 2, we display the accuracy, the asymmetry, the positive and negative likelihood ratios, and the area under curve (AUC) for all the three Higgs boson masses investigated in our work. All metrics indicate an overall improvement of NN_2 over NN_1 , but the gain in performance is more pronounced in the 1 TeV case. Lighter masses present attributes less discernible than the backgrounds, so profit more from an ensemble of classifiers that use more distinctive features like the classification scores and the $M_{\ell\ell\nu\nu}$ mass.

The improvement is more significant for the signals and the WW background compared to $ZZ(\gamma^*)$ and $t\bar{t}$ events. This can be further confirmed by looking at the Fig. 9, the difference between the confusion matrices of the NN_1 and NN_2 classifiers. First of all, we want the diagonal of Fig. 9 to be all positive, which means that NN_2 increases the true positive rate compared to NN_1 . At the same time, negative non-diagonal entries mean less misclassification among classes. Overall, taking into account the results for the three Higgs masses, we see a clear improvement of NN_2 compared to NN_1 . Except for the $ZZ(\gamma^*)$ class in the 1 and 1.5 TeV cases, all diagonal entries are positive, with a major improvement of WW classification. Moreover, the 1 TeV signal class benefits more from NN_2 than the heavier masses. This is a good

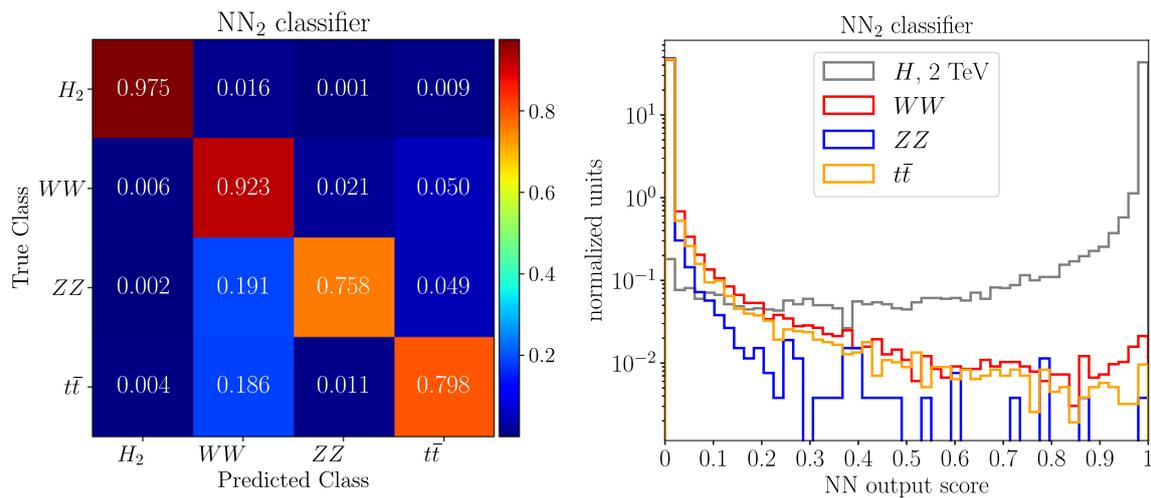


Fig. 8 Confusion matrix of the classification and the output scores of the second neural network at the left and right panels, respectively, for a 2 TeV Higgs boson and its main SM backgrounds

Table 2 Comparison of metrics performance of models trained (NN₁ and NN₂) to identify Higgs boson of all the three masses considering in this work

Metric	1 TeV		1.5 TeV		2 TeV	
	NN ₁	NN ₂	NN ₁	NN ₂	NN ₁	NN ₂
Accuracy	81.1%	87%	81.3%	87.6%	80.6%	87.5%
Asymmetry, H_2	0.821	0.933	0.887	0.944	0.917	0.941
Asymmetry, WW	-0.903	-0.978	-0.933	-0.988	-0.939	-0.991
Asymmetry, ZZ	-0.992	-0.996	-0.994	-0.997	-0.994	-0.999
Asymmetry, $t\bar{t}$	-0.916	-0.984	-0.934	-0.992	-0.947	-0.994
LR_+	28.225	74.371	45.299	86.010	61.330	82.360
LR_-	0.603	0.581	0.589	0.578	0.582	0.578
$AUC(H_2 \times WW)$	0.9813	0.9977	0.9915	0.9988	0.9940	0.9991
$AUC(H_2 \times ZZ)$	0.9965	0.9994	0.9975	0.9996	0.9975	0.9996
$AUC(H_2 \times t\bar{t})$	0.9822	0.9984	0.9929	0.9992	0.9950	0.9993

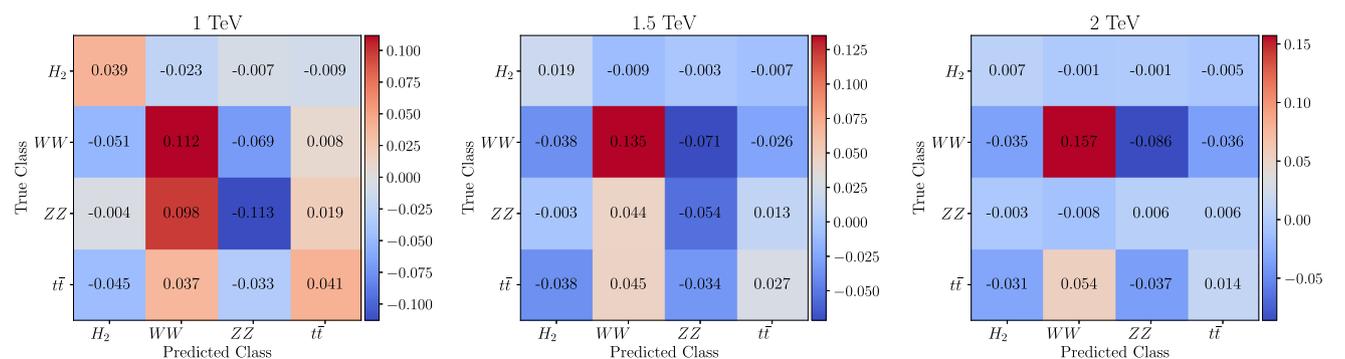


Fig. 9 Confusion matrices differences (NN₂ – NN₁) for models trained to separate Higgs bosons of mass 1 TeV (left), 1.5 TeV (center), and 2 TeV(right)

feature of $kNNNN$; it helps in the more difficult cases for the signals. Concerning the non-diagonal entries, we observe a clear trend – the $ZZ(\gamma^*)$ class is more accurately identified

by models whose task is to separate heavier Higgs signals. In contrast, the other classes are less confused among themselves by NN₂. On the other hand, the more accurate WW and

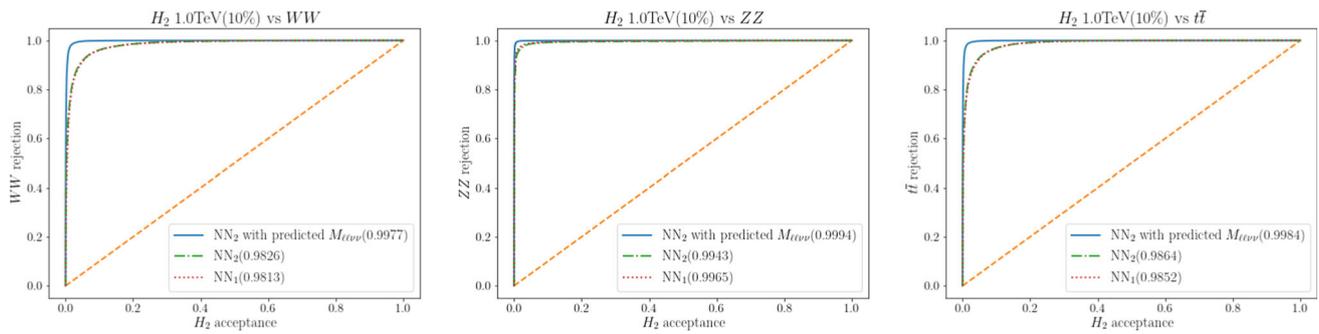


Fig. 10 The ROC curves of a 1 TeV Higgs (with $\Gamma_H/m_H = 0.1$) signal against the background classes, WW , ZZ , and $t\bar{t}$. In legends, in parenthesis, we show the AUC corresponding the first classifier, NN_1 , and the second classifier, NN_2 , with and without including the predicted mass

$ZZ(\gamma^*)$ classification comes at the cost of a slight increase in mistagging of $t\bar{t}$ events as WW .

The statistical significance of the signal hypothesis depends roughly on the fraction of background events which are rejected, $r_B = 1 - \varepsilon_B$ at a fixed signal acceptance, ε_S , $\varepsilon_S/\sqrt{1 - r_B}$. In Fig. 10, we display the Receiver Operator Characteristic (ROC) curves of a 1 TeV Higgs and $\Gamma_H/m_H = 0.1$ against the background classes. We see that, for a fixed H_2 acceptance, the backgrounds rejections increase with the NN_2 classification compared to NN_1 . The rejection is larger when we include the regressed $M_{\ell\ell\nu\nu}$ mass in the data representation. The area under curve (AUC), which summarizes the ROC curve, are shown at last row of Table 2 for all the three Higgs masses. AUC increases from NN_1 to NN_2 in all cases, and for other Higgs masses and total widths as well.

After the second classification, using the class scores of NN_1 and the predicted $M_{\ell\ell\nu\nu}$ mass of the Regressor, the second neural network NN_2 now provides more accurate predictions to inform the Regressor which support set to use for the regression task. As an outcome, the contaminations from other classes get reduced, and the prediction of $M_{\ell\ell\nu\nu}$ improves. We show the predicted $\ell\ell\nu\nu$ invariant mass after the second classification in Fig. 11.

4 Improvement of the signal significance

Now that we have established a working algorithm to predict the $M_{\ell\ell\nu\nu}$ mass, we want to investigate whether it is helpful to boost the statistical signal significance when employing a machine learning classifier. The signal significance is computed according to

$$N_\sigma = \frac{\varepsilon_{cut}^{(S)} \times N_S}{\sqrt{\sum_i \varepsilon_{cut}^{(i)} \times N_{B_i} + (\varepsilon_B \times \sum_i \varepsilon_{cut}^{(i)} \times N_{B_i})^2}}, \quad (8)$$

where N_S and N_{B_i} , $i = WW, ZZ, t\bar{t}$ denote the number of signal and backgrounds events, respectively; $\varepsilon_{cut}^{(S)}$ and

$\varepsilon_{cut}^{(i)}$, $i = WW, ZZ, t\bar{t}$ denote the signal and backgrounds cut efficiencies (both on kinematic variables and score outputs), respectively; finally, ε_B represents a systematic uncertainty in the backgrounds rates assuming, for simplicity, a common uncertainty for all background sources.

The production cross sections of WW , $ZZ(\gamma^*)$, and $t\bar{t}$, at leading order are given by 102.8, 14.15 and 674.1 picobarns, respectively. The branching ratios for $W \rightarrow \ell\nu$, $Z \rightarrow \ell^+\ell^-$, $Z \rightarrow \nu\bar{\nu}$, and $t \rightarrow bW^-$ are taken to be 10.68%, 3.37%, 20%, and 100%, respectively. Assuming the basic cuts of Eq. (2), and an integrated luminosity of 500 fb^{-1} , we estimate 2.35×10^6 , 1.91×10^5 and 1.53×10^7 events, amounting to around 1.8×10^7 background events at the 13 TeV LHC. Including NLO QCD corrections, these numbers should increase by a few tens of percent. We fix the number of signals events at 1000 for all masses for illustration purposes. The actual signal production cross section depends on the specific model of new Higgs bosons.

As discussed previously, we are interested in showing the increase in the signal significance that our proposed algorithm is expected to produce by including the predicted $M_{\ell\ell\nu\nu}$ mass in the data representation. The significance gain is defined as

$$\text{Significance Gain} = \frac{N_\sigma(\text{kNNNN})}{N_\sigma(\text{NN}_1)}, \quad (9)$$

where $N_\sigma(\text{kNNNN})$ is the statistical significance after using NN_2 for classification with or without the inclusion of the predicted $M_{\ell\ell\nu\nu}$ mass as a feature. Moreover, we also wish to check if the predicted masses cause an underestimation or overestimation of the statistical significance compared to what we could get if we knew the true $M_{\ell\ell\nu\nu}$ distribution.

In Fig. 12, we show, at the left panels, the statistical significance, assuming a $\varepsilon_B = 10\%$ systematic uncertainty in the backgrounds rates, for a new Higgs boson of 1, 1.5, and 2 TeV mass, from top to bottom rows, respectively. To raise the significance, we cut on the classifiers' signal score output represented in the plots' horizontal axis. The 1st NN and 2nd NN lines depict the significance of NN_1 and NN_2 , respec-

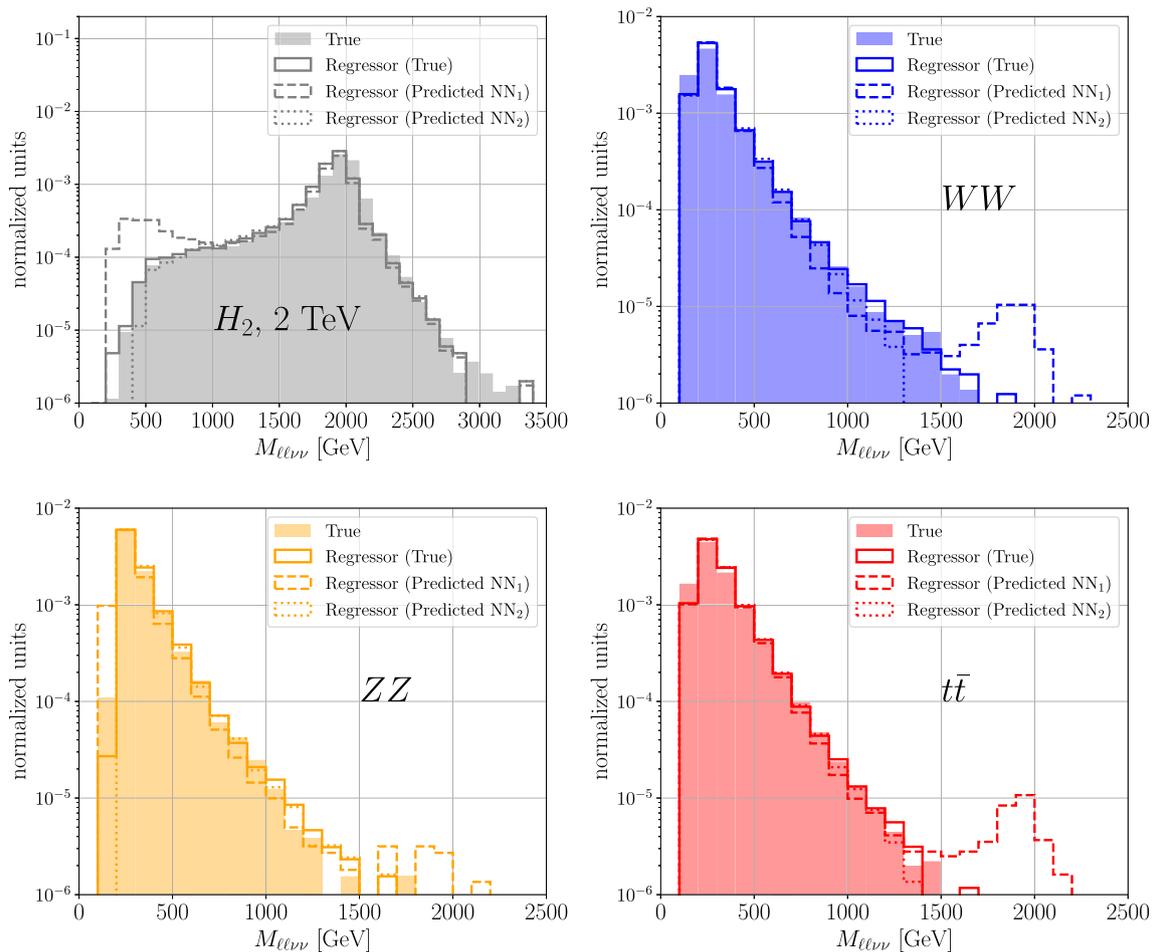


Fig. 11 The true (shaded areas), regressed from true samples (solid lines), regressed from samples identified with NN1 [NN₂] (dashed lines) [dotted lines] $M_{\ell\ell\nu\nu}$ distributions for the 2 TeV Higgs (upper left), WW (upper right), $ZZ(\gamma^*)$ (lower left) and $t\bar{t}$ background (lower right)

tively, without including the $M_{\ell\ell\nu\nu}$ prediction. Even without reconstructing the resonance, the stacking of the neural networks boosts the significance, as expected. The statistical significance is much enhanced, including the predicted mass, as shown in the top lines in all the left panels. As we see from the dashed lines, the agreement with what should be expected using the true masses in the data representation is good. The agreement is better for lower masses, while a more pronounced overestimation is observed in the 2 TeV case. An insufficient number of simulated background samples might cause that effect. Yet, the quality of the resonance reconstruction enables us to employ the method to select the signal events better.

At the right panels of Fig. 12, we show the significance gain relative to the first neural network classifier, NN₁. While not including the predicted $M_{\ell\ell\nu\nu}$ mass leads to gains around 2, including them boosts the gains to up to 6, 8 and 10 for 1, 1.5 and 2 TeV masses, respectively. As noted in the left panels, there is a more pronounced overestimation of around 20 to 25%, depending on the cut score, for 2 TeV

Higgs bosons. Similar gains were observed when we varied the Higgs bosons widths down to $\Gamma_H/m_H = 1\%$. The train/test/validation dataset was randomly split five times to assess the robustness of these results, and tiny variations were observed in this cross-validation.

The importance of the reconstructed $\ell\ell\nu\nu$ mass with kNNNN is confirmed from a feature importance analysis using Shapley values from the SHAP [50] package. In Fig. 13, we display the feature importance hierarchy in the case of a 1 TeV Higgs, but similar conclusions hold for the other masses. The most useful features for the second neural network meta-classifier appear at the left and their importance decreases towards the right end. As we see, the regressed mass is the most useful one when we included it in the data representation followed by the p_T of the harder lepton, and the NN₁ scores for the WW and H_2 classes. It is also noticeable the low ranking of the Higgsness variable which might be due the multiclass nature of the classification algorithms or a strong correlation with the other variables.

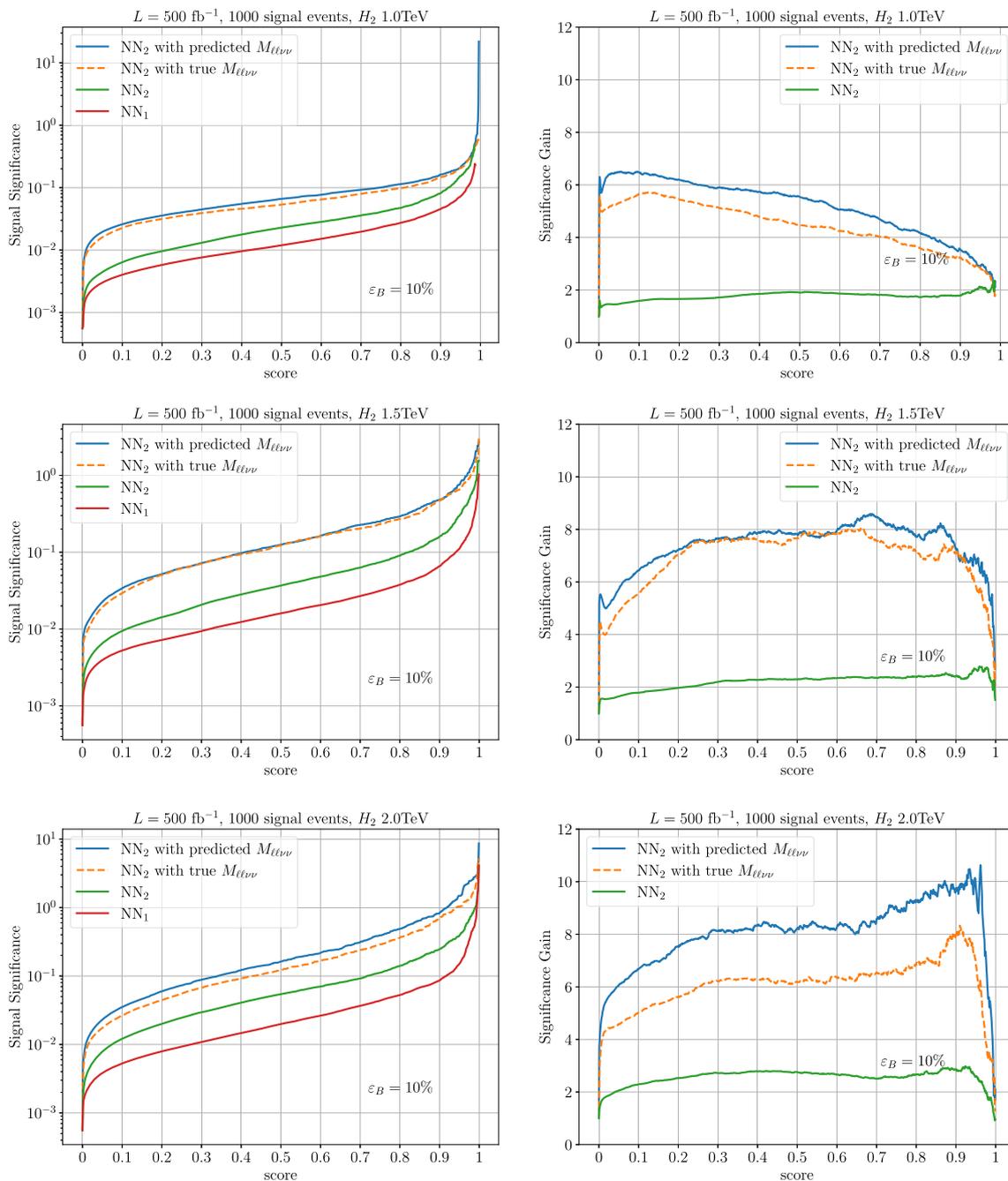


Fig. 12 The statistical significance of the k NNNN algorithm as a function of the cut on the output score (left panels) and the gain in the significance, Eq. (9), compared to the NN_1 classifier (right panels) for a 1

(upper row), 1.5 (center row) and 2 TeV (lower row) Higgs mass. The systematic uncertainty is fixed at 10%

5 Conclusions and prospects

As the search for new physics intensifies following the LHC program schedule, new ways to identify particles that hide information through invisible decays are surely welcome. In this work, we designed an algorithm capable of reconstructing the mass of a new heavy Higgs boson decaying to

$W^+W^- \rightarrow \ell^+\ell^-\nu\bar{\nu}\ell'$ and its main SM backgrounds using a simple but adequately tuned nearest neighbors algorithm. The algorithm assumes the previous knowledge of the event classes and the Higgs boson mass; therefore, it is useful for post-discovery studies, for example, an analysis that requires a selection of on-mass shell Higgs bosons.

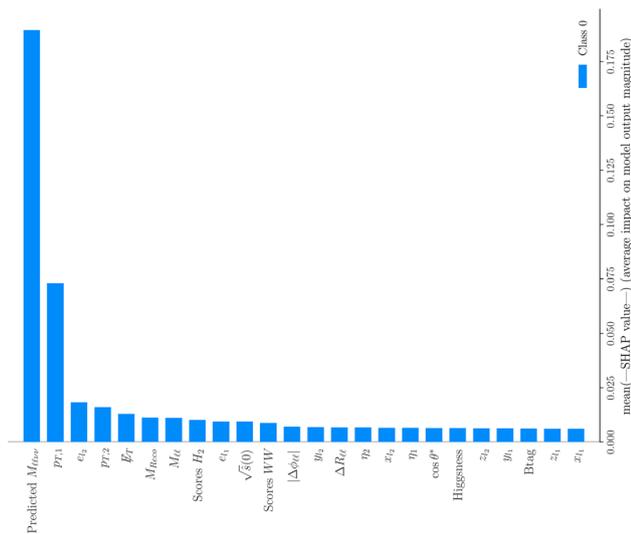


Fig. 13 The feature importance plot. Predicted $M_{\ell\ell\nu\nu}$ becomes the most important feature for classification when it is included in the data representation

More importantly, including the predicted kNN $M_{\ell\ell\nu\nu}$ mass as an attribute for a neural network classification improves the accuracy, the true and false positives/negatives rates, the likelihood of true class classifications when compared to a neural network that does not have a clue about the masses. We computed the ROC curves and confirmed that the second classifier increases AUC compared to the first one, especially when the predicted mass is included in the data representation, improving the backgrounds rejection for a fixed signal acceptance. A feature importance analysis corroborates the role played by the regressed $M_{\ell\ell\nu\nu}$ for the meta-classifier NN_2 . The gain in the statistical significance is the ultimate test for the proposed algorithm. We found a gain factor in significance up to a factor of 10 for a 2 TeV Higgs boson mass. For lighter masses, of 1 and 1.5 TeV, the gains are less pronounced but also high, up to 6 and 8, respectively, depending on the cut placed on the signal class score. We checked that the predicted mass is reliable and robust as a new feature for classification by comparing our results against classifiers trained with the true $M_{\ell\ell\nu\nu}$ masses. Not only the invariant mass distributions agree but also the final statistical significance agree within a few tens of percent, at most.

The $kNNNN$ algorithm can be applied to other observable variables as well. For example, the scattering angle of the W bosons can be obtained in the fully leptonic channel beside the charged leptons angles. The masses of particles in different topologies can also be obtained. For example, we guess that sparticles' mass distributions from decay chains of various lengths might be recovered after their determination with other methods.

The next step in this kind of investigation is to relax the previous knowledge of the mass parameters and weaken the level of supervision when training the classifiers and regressors. Outlier detection and other unsupervised techniques can be readily used to dismiss previous knowledge of the signal class, yet, using kNN for regression requires the knowledge of mass parameters. A completely weakly supervised regression algorithm that assumes just the knowledge of the background classes is challenging once it involves generalization across classes with essential information loss. We are currently investigating deep neural networks and variational autoencoders for regression algorithms trained on a single background class but still assuming previous knowledge of signal mass parameters. These results will be presented elsewhere.

Acknowledgements A. Alves is supported by grants from the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) (307317/2021-8) and Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) (2021/01089-1).

Data Availability Statement This manuscript has no associated data or the data will not be deposited. [Authors' comment: The data consist of simulated events, which can be reproduced from the simulation details provided in the manuscript.]

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Funded by SCOAP³. SCOAP³ supports the goals of the International Year of Basic Sciences for Sustainable Development.

References

1. M. Della Negra, P. Jenni, T.S. Virdee, *Annu. Rev. Nucl. Part. Sci.* **68**, 183 (2018). <https://doi.org/10.1146/annurev-nucl-101917-021038>
2. A.M. Sirunyan et al. (CMS), *JHEP* **03**, 034 (2020). [arXiv:1912.01594](https://arxiv.org/abs/1912.01594) [hep-ex]
3. S. Banerjee, B. Batell, M. Spannowsky, *Phys. Rev. D* **95**, 035009 (2017)
4. A. Alves, T. Ghosh, F.S. Queiroz, *Phys. Rev. D* **100**, 036012 (2019). [arXiv:1905.03271](https://arxiv.org/abs/1905.03271) [hep-ph]
5. A.M. Sirunyan et al. (CMS), *Phys. Lett. B* **790**, 140 (2019). [arXiv:1806.05264](https://arxiv.org/abs/1806.05264) [hep-ex]
6. G. Aad et al. (ATLAS), *Eur. Phys. J. C* **80**, 123 (2020). [arXiv:1908.08215](https://arxiv.org/abs/1908.08215) [hep-ex]
7. M. Aaboud et al. (ATLAS), *Phys. Lett. B* **798**, 134942 (2019). [arXiv:1904.12679](https://arxiv.org/abs/1904.12679) [hep-ex]

8. V. Khachatryan et al. (CMS), Phys. Lett. B **698**, 21 (2011). [arXiv:1012.5945](#) [hep-ex]
9. G. Aad et al. (CMS, ATLAS), JHEP **08**, 051 (2020). [arXiv:2005.03799](#) [hep-ex]
10. L. Sonnenschein, Phys. Rev. D **73**, 054015 (2006). [Erratum: Phys. Rev. D **78**, 079902 (2008)]. [arXiv:hep-ph/0603011](#)
11. A.M. Sirunyan et al. (CMS), JHEP **04**, 060 (2018). [arXiv:1708.07638](#) [hep-ex]
12. C.G. Lester, D.J. Summers, Phys. Lett. B **463**, 99 (1999). [arXiv:hep-ph/9906349](#)
13. V. Barger, P. Huang, Phys. Rev. D **84**, 093001 (2011). [arXiv:1107.4131](#) [hep-ph]
14. A.J. Barr, C.G. Lester, J. Phys. G **37**, 123001 (2010). [arXiv:1004.2732](#) [hep-ph]
15. P. Konar, K. Kong, K.T. Matchev, M. Park, JHEP **06**, 041 (2011). [arXiv:1006.0653](#) [hep-ph]
16. S. Kawabata, Y. Shimizu, Y. Sumino, H. Yokoya, JHEP **08**, 129 (2013). [arXiv:1305.6150](#) [hep-ph]
17. W.S. Cho, K. Choi, Y.G. Kim, C.B. Park, Phys. Rev. D **79**, 031701 (2009). [arXiv:0810.4853](#) [hep-ph]
18. A.J. Barr, B. Gripaios, C.G. Lester, JHEP **02**, 014 (2008). [arXiv:0711.4008](#) [hep-ph]
19. P. Konar, A.K. Swain, Phys. Rev. D **93**, 015021 (2016). [arXiv:1509.00298](#) [hep-ph]
20. C.B. Park, JHEP **07**, 089 (2020). [arXiv:2005.12297](#) [hep-ph]
21. K. Choi, S. Choi, J.S. Lee, C.B. Park, Phys. Rev. D **80**, 073010 (2009). [arXiv:0908.0079](#) [hep-ph]
22. K.T. Matchev, P. Shyamsundar, JHEP **04**, 027 (2020). [arXiv:1911.01913](#) [hep-ph]
23. J.-C. Yang, J.-H. Chen, Y.-C. Guo, JHEP **21**, 085 (2020). [arXiv:2107.13624](#) [hep-ph]
24. J. Li, C. Zhang, R. Zhang, Phys. Rev. D **105**, 016005 (2022). [arXiv:2109.09924](#) [hep-ph]
25. M. Arratia, D. Britzger, O. Long, B. Nachman, Nucl. Instrum. Methods A **1025**, 166164 (2022). [arXiv:2110.05505](#) [hep-ex]
26. D.F. Rentería-Estrada, R.J. Hernández-Pinto, G.F. R. Sborlini, P. Zurita, (2021). [arXiv:2112.05043](#) [hep-ph]
27. S.B. Menary, D.D. Price, Mach. Learn. Sci. Tech. **3**, 015021 (2022). [arXiv:2108.11481](#) [physics.data-an]
28. P. Flach, Machine learning: the art and science of algorithms that make sense of data (Cambridge University Press, Cambridge, 2012)
29. T. Hastie, R. Tibshirani, J. Friedman, *The elements of statistical learning, Springer series in statistics* (Springer New York Inc., New York, 2001)
30. P. Mehta, M. Bukov, C.-H. Wang, A.G. Day, C. Richardson, C.K. Fisher, D.J. Schwab, Phys. Rep. **810**, 1–124 (2019)
31. M. Grossi, J. Novak, B. Kersevan, D. Rebutti, Eur. Phys. J. C **80**, 1144 (2020). [arXiv:2008.05316](#) [hep-ph]
32. Q. Zhao, E. Adeli, N. Honnorat, T. Leng, K.M. Pohl, in *International conference on medical image computing and computer-assisted intervention* (Springer, 2019), pp. 823–831
33. P. Donmez, G. Lebanon, K. Balasubramanian, J. Mach. Learn. Res. **11**, 1323 (2010)
34. N.S. Altman, Am. Stat. **46**, 175 (1992)
35. M.P. Deisenroth, A.A. Faisal, C.S. Ong, *Mathematics for machine learning* (Cambridge University Press, Cambridge, 2020)
36. J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H.S. Shao, T. Stelzer, P. Torrielli, M. Zaro, JHEP **07**, 079 (2014). [arXiv:1405.0301](#) [hep-ph]
37. T. Sjostrand, S. Mrenna, P.Z. Skands, Comput. Phys. Commun. **178**, 852 (2008). [arXiv:0710.3820](#) [hep-ph]
38. J. de Favereau, C. Delaere, P. Demin, A. Giammanco, V. Lemaître, A. Mertens, M. Selvaggi (DELPHES 3), JHEP **02**, 057 (2014). [arXiv:1307.6346](#) [hep-ex]
39. A.J. Barr, JHEP **02**, 042 (2006). [arXiv:hep-ph/0511115](#)
40. J.H. Kim, K. Kong, K.T. Matchev, M. Park, Phys. Rev. Lett. **122**, 091801 (2019). [arXiv:1807.11498](#) [hep-ph]
41. P. Virtanen, R. Gommers, T.E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S.J. van der Walt, M. Brett, J. Wilson, K.J. Millman, N. Mayorov, A.R.J. Nelson, E. Jones, R. Kern, E. Larson, C.J. Carey, Í Polat, Y. Feng, E.W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E.A. Quintero, C.R. Harris, A.M. Archibald, A.H. Ribeiro, F. Pedregosa, P. van Mulbregt, SciPy 1.0 Contributors, Nat. Methods **17**, 261 (2020)
42. I. Goodfellow, Y. Bengio, A. Courville, Deep learning (MIT Press, 2016). <http://www.deeplearningbook.org>
43. F. Chollet et al., Keras (2015)
44. M.A. et al, TensorFlow: large-scale machine learning on heterogeneous systems (2015). Software available from tensorflow.org
45. J. Bergstra, Hyperopt: distributed asynchronous hyper-parameter optimization
46. A. Alves, JINST **12**, T05005 (2017). [arXiv:1612.07725](#) [hep-ph]
47. P. Baldi, P. Sadowski, D. Whiteson, Nat. Commun. **5**, 4308 (2014). [arXiv:1402.4735](#) [hep-ph]
48. Y. Ren, L. Zhang, P. Suganthan, IEEE Comput. Intell. Mag. **11**, 41 (2016)
49. N. Japkowicz, M. Shah, *Evaluating learning algorithms: a classification perspective* (Cambridge University Press, Cambridge, 2011)
50. S.M. Lundberg, S.-I. Lee, in *Advances in neural information processing systems* ed. by I. Guyon, U.V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, R. Garnett, vol. 30 (Curran Associates, Inc., 2017), pp. 4765–4774