Search for BSM physics at the LHC with machine learning

Khalil Pierre
Department of Physics, University of Bristol

(Dated: May 2021)

In the Standard Model the Higgs boson can decay to invisible particle with a branching ratio of 0.1%. This could be enhanced if the Higgs decays into new particles such as DM. So far an upper bound of 11% has been placed on the invisible Higgs branching ratio by ATLAS. Measurements of a enhanced branching ratio would be strong evidence of BSM physics. In this project, the ability of different artificial neural networks to distinguish simulated invisible Higgs events from the SM background at CMS was assessed. Three different network models were built: a FFN, a RNN, and a combined neural network. Initially, work was focused on the ttH channel, the combined neural network performed the best at distinguishing ttH→inv events from its associated background, receiving an AOC score of 0.801 ± 0.002. Other production channels were considered, with the combined network performing the best at distinguishing WH→inv events, with an AOC score of 0.869 ± 0.005. Finally, a multi-classifier combined neural network was built and trained on ttH and WH events. The multi classifier received a micro averaged AUC score of 0.87 ± 0.01, showing a good level of discernment.

CONTENTS

INTRODUCTION

The standard model (SM) is currently the most precisely tested theory of particle physics. However, it is an incomplete theory. There are many observed phenomenon that the standard model cannot explain, these include the baryon asymmetry of the universe, the SM currently does not have a description of gravity and the SM has no dark matter (DM) candidate that is consistent with astronomical observations. These deficiencies plus more suggest the existence of physics beyond the standard model (BSM).

One possible method to probe BSM physics is via measurements of the invisible Higgs decay branching ratio Br($H \rightarrow inv$). The Higgs boson is believed to be a possible portal between the dark sector and the SM sector via its decay into DM particles. Strong cosmological evidence supports the existence of DM [1] and so all extensions to the SM contain DM candidates. Supersymetric models propose the lightest supersymetric particle as a DM candidate [2, 3], extra dimensional models suggests graviscalars [4] and in fourth generation quark models a massive neutrino takes the role of DM [5]. If the Higgs decays into one of these DM particles then the Higg's decay will be 'invisible' to the detectors at the LHC as DM interacts very weekly via non-gravitational forces.

In the SM the Higgs can only decay invisibly via two Z bosons which then decay into four neutrinos $H \rightarrow ZZ^* \rightarrow 4\nu$. The branching ratio of the SM invisible Higgs decay is predicted to be 0.1% [6–9]. Observations of an invisible Higgs branching ratio larger then the SM prediction would be strong evidence for the existence of BSM physics. CMS and ATLAS are two general purpose detectors at the LHC searching for invisible Higgs decays. Measurements by ATLAS have put an upper bound of 11% on Br($H \rightarrow inv$) [10] and measurements at CMS have put an upper bound of 19% on Br($H \rightarrow inv$) [11] both with a 95% confidence level. The high luminosity LHC which is expected to come online later this decade is predicted to be able to refine the observed upper limit of Br($H \rightarrow inv$) to 2.5% [9]. So whilst the theoretical limit cannot be probed as of yet, work is on going to refine measurements of the Br($H \to inv$) and there is still a lot of room for possible BSM physics.

Invisible Higgs decays can be observed via the radiation patterns they produce. The most common method for detecting invisible Higgs decays involves using event kinematic and selection thresholds to reduce SM background events. Recently machine learning has been deployed to improve measurements of Br($H \to inv$). A recent paper "Invisible Higgs search through vector boson fusion: a deep learning approach" found that using machine learning, the upper bound of the invisible Higgs branching ratio could be improved by a factor of 3 compared to previous attempts on the same data set. The team used low-level calorimeter data at CMS which was then passed through different neural networks and an upper bound of Br($H \to inv$) was calculated.

In this project high level event data and object data was used to train 3 different types of networks, a feed forward network (FFN) a recurrent network (RNN) and a combined network (FFN + RNN). All the networks used in this project were built using the TensorFlow Keras libary. The networks were trained on simulated data with ground truth labels to test the abilities of the different networks to classify invisible Higgs decay events.

BACKGROUND THEORY

Higgs production at the LHC

There are four main Higgs production modes at the LHC: gluon-gluon fusion (ggF), vector boson fusion (VBF), associated production with a gauge boson (VH) and associated production with a pair of top quarks (ttH) [6–9, 11–14]. The leading order Feynman diagrams for these production modes can be seen in Figure 1 and the cross section for each production mode at the LHC can be seen in Table I. Whilst the decay products of an invisible Higgs decay cannot be detected, the by products of a production event can. Jets are the most common way the by products are observed. Jets are the results of the Hadronization of unbound colour charge [8, 15]. The properties of these jets; the mass, the angle between jets, the number of jets and more, are determined by the specific event and are predicted by the SM. Invisible Higgs decay events will also produce a large missing transverse momentum P_T^{miss} as the decay products will carry the Higgs mass beyond the detector. The radiation patterns of Higgs production events can therefor be used to identify possible invisible Higgs decay events [6, 8]. For instance, if you wanted to observe invisible Higgs decay via the VBF production mode, one way to do so would be to look for events with 2 or more jets as the quarks the Higgs is produced in conjunction with would form jets. However things are more complicated as other SM events will have many of the same properties as an invisible Higgs events. The most common method to distinguish invisible Higgs events from background events is to use stringent selection threshold to suppress the SM background. The most basic selection threshold is requiring a large missing transverse energy (MET) to isolate the signal region.

	Production cross section (in pb) for m_H = 125GeV					
$\sqrt{s}$ TeV	ggF	VBF	WH	ZH	ttH	total
13	$48.6^{+5\%}_{-5\%}$	$3.78^{+2\%}_{-2\%}$	$1.37^{+2\%}_{-2\%}$	$0.88^{+5\%}_{-5\%}$	$0.50^{+9\%}_{-13\%}$	55.1
14	$54.7^{+5\%}_{-5\%}$	$4.28^{+2\%}_{-2\%}$	$1.51^{+2\%}_{-2\%}$	$0.996^{+5\%}_{-5\%}$	$0.60^{+9\%}_{-13\%}$	62.1

TABLE I: The cross sections for the main Higgs production modes at the LHC [12]

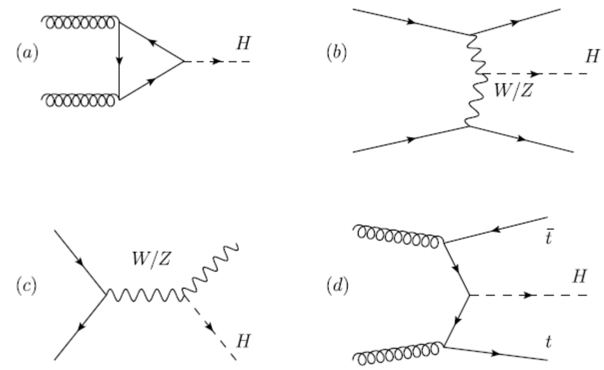


FIG. 1: The different production modes of the Higgs boson at the LHC. The production modes are ordered from highest to lowest cross section a) gluon-gluon fusion, b) vector boson fusion, c) associated production with a gauge boson, d) associated production with a pair of top quarks.

The dominant background processes for ggF, VH, and VBF are due to Z($\to v\bar{v}$) + jets and W($\to l\bar{v}$) + jets where the lepton escapes detection [6, 8, 14]. There are also contributions from other QCD process which can be reduced significantly by requiring a large missing transverse energy [6]. Whilst ggF has the highest cross section it predominately produces a Higgs in isolation [12] as shown by the leading order Feynman diagram, as shown in Figure 1a. However, higher order ggF production does result in final states with monojets and dijets which, can be used to identify invisible Higgs events [8, 11]. Whilst VBF production is the subdominant production mode at the LHC, it has a few properties that generally make it the most sensitive channel for studying invisible Higgs decay events [12]. Namely VBF to invisible events are characterised by two leading jets in the forward and backward direction with a wide separation in rapidity. This allows for a central jet veto which can effectively reduce the SM background from QCD events [6, 8]. The VH production mode has a smaller cross-section than both the VBF and ggF. However, the presence of a W or Z boson allows for a larger variety of final states which can be tagged to produce a relatively low background [6, 12]. In this study only hadronic VH final states were considered. A possible extension may look at leptonic VH final states. The ttH channel is the least studied channel because of its low cross section. Recent studies have looked at constraining Br($H \rightarrow inv$) using the ttH channel; a study in 2019 was able to place an upper bound of 46% on Br($H \rightarrow inv$) with a 95% confidence level [7]. This study focused on the ttH channel initially. The dominant SM background comes from other top quark pair production processes. Top quark decays can be categorised into three channels: di-leptonic $t\bar{t} \rightarrow 2bjets{+}2(lv)$, semi-leptonic $t\bar{t} \rightarrow 2bjet + (lv) + 2jets$ and hadronic $t\bar{t} \rightarrow 4jets + 2bjets$. All three were considered in this project.

DATA

Simulated samples

The signal and background processes where simulated using Monte Carlo generators. MadGraph was used to generate the Z+jets, W+jets, and QCD background; aMC@NLO was used to generate the ttH background processes; and PowHEG was used to generate the signal processes. The typical process for generating the sample data is as follows. Firstly the underlying processes are generated using the leading order or next to leading order Feynman diagrams to produce the final states. Next, any final state quarks or gluons are hadrizonized to produce realistic jets. Geant4 was used to simulate the interactions of the outgoing particles with the CMS detector material. The results of other events are overlaid on top of each other to simulate the other particles that would be present in the detector which is called pile up mixing. The detector electronics are simulated to produce realistic read out data. The readout data is then passed through the particle flow algorithm which is the algorithm used at CMS to reconstruct collision events. Finally, the data is reduced to the relevant variables for the analysis. The variables can be broken into two categories; event variables which contain global information, and jet variables which contains information about each individual jet within an event. For a full description of the variables used see appendix C. Whilst the number of jet variables is the same for every jet, the number of jets in each event varies and so the jet data for each event has a variable length. It is this fact which motivates the use of RNNs this will be discussed further.

The following pre-selection cuts were applied to isolate the signal region:

- MET $>$ 200 MeV
- Missing transverse hadronic energy (MHT) $>$ 200 MeV
- MHT $\div$ MET < 1.2
- The highest value of P_t for any jet is at least 80 GeV
- The two jets with the highest P_t have at least 10% of their hadronic energy coming from charged particles.
- The two jets with the highest P_t have at least 80% of their hadronic energy coming from neutral particles.
- All jets pass the clean jet requirement (they can be distinguished)
- The azimuthal angle between MET and MHT is less than 0.5 radians.
- There are no electrons, muons, or photons in the event.

Data prepossessing

Before the data could be fed into the neural networks it had to be pre-processed. For the purpose of clarity this section will discuss the pre-processing of the ttH data for a binary classifier, though the description of the data pre-processing is general. The combined background for the ttH data will be referred to as tt¯ and consists of di-leptonic, semi leptonic, and hadronic top quark pair decays.

Each event type in the data set was generated a random number of times. Because of this weights are required, firstly to retain physicality, and secondly to make sure that any network which is trained on the data does not over-fit to one class (ttH or tt¯). Figure 2 shows how the data changes as the different weights are applied. The first figure on the far left shows the unweighted data. The second figure in the middle shows the shape of the data when the first set of weights are applied. These weights, known as the cross section weights, comes from the generation of the simulated data. When applied to the data set, the cross section weights tell us the expected number of events for 1 pb^{-1} integrated luminosity at the LHC. This puts the different events into the correct physical proportions. However, as can be seen the expected number of background events far outweighs the expected number of signal events. If the input data is heavily weighted towards one class, in this case tt¯ the network will over fit to the dominant class. To overcome this the cross-section weights of each event were normalised by class type. The effect of the normalisation can be seen in the far right plot in Figure 2. The cross-section weights of the ttH events were summed and each ttH events corresponding cross section weight was divided by the summed cross section weight. This was repeated for the tt¯ events with the summation now including all events in the tt¯ background. Written out this is:

$$\tilde{w_i} = \frac{w_i}{\sum_{j=0}^{N} w_j}, \qquad 1$$

where $\tilde{w_i}$ is the normalised cross-section weight, w_i is the unnormalised cross-section weight and N is the total number of events in a class. When the weights are fed into the neural network, it will give extra significance to the ttH events when it calculates its update, overcoming the class imbalance.

The reason this weighting method is used, rather than simply taking the class weights as the ratio of the different classes is because, despite the physical ratio between signal and background events not being preserved across the classes within the background class, which is a composite class, the physical ratio of each event is preserved by the weights. This is important as the network will put more significance on learning to distinguish the dominant background event type(s) within the signal region rather than the more rare background event types.

Finally the event and jet data variables have to be normalised between zero and one. To do this the sklearn min max scalar function was used. This is standard practice and is done in order to help the network train efficiently.

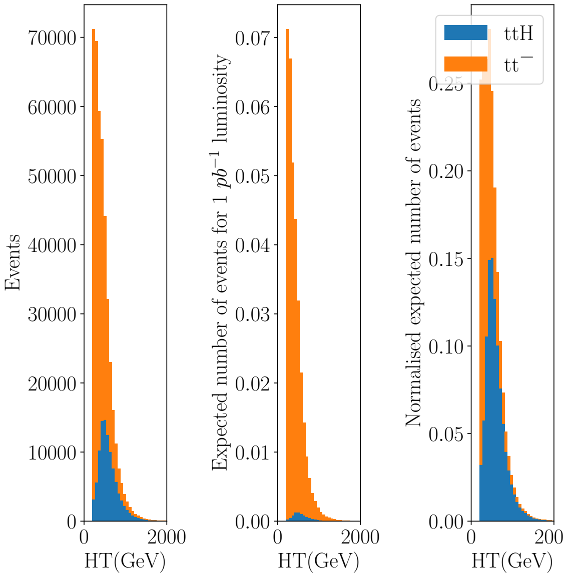


FIG. 2: Stacked histograms of the HT event data as the different weights are applied. From left to right; no weight, cross section weight and normalised cross section weight.

ARTIFICIAL NEURAL NETWORKS

Artificial Neural Networks (ANN) are a subset of machine learning methods. They are loosely inspired by neurons in animal brains [14]. Broadly speaking, there are three kinds of machine learning algorithms, supervised learning, unsupervised learning and reinforcement learning.

Supervised learning - These methods use labeled data to 'learn' the relationship between input variables so that the output can as closely approximate the true label as possible. Supervised learning has been used in this project.

Unsupervised learning - These methods are applied to unlabeled data, the algorithms try to identify structures within a data set. A common unsupervised learning method is K Means Clustering.

Reinforcement learning - To complete a task the machine is given rewards for certain actions. the machine will perform actions that maximise the reward. Penalties can also be introduced if necessary.

The two types of ANN that were looked at in this study are FFNs and RNNs.

Feed forward neural network

The basic structure of a FNN is shown in Figure 3. ANNs generally consist of multiple layers of neurons where the neurons in each layer are connected by a series of vertices. These vertices represent weights. FFNs are differentiated from other ANNs by the fact that information in a FFN is only carried forward. The value of each neuron is determined by the value of the neurons in the previous layer which are then modulated by the weights [16]. It is the goal of the network to update the weights and other network parameters so that the network approximates some function. The 'learning' process is governed by trying to minimise some objective function $J(\theta)$ which is parameterised by the network parameters. The network seeks to find the global minimum of the objective function. The most common method to do this is gradient descent. After a training sample is passed through the network, the network will calculate the gradient of the objective function w.r.t. the parameter of the network. The network will then step in the opposite direction to the gradient vector in the parameter space. By doing this iteratively after each training sample (or sample set), the network will reach a local or ideally a global minimum [17]. The development of neural network optimisation algorithms has become more sophisticated. In this study TensorFlow's Adam optimisation algorithm was used. Adam follows the same principle as the basic gradient descent algorithm but the learning rate (step size) varies. This is to avoid the optimisation algorithm from overshooting the minima and avoid getting stuck in saddle points [17].

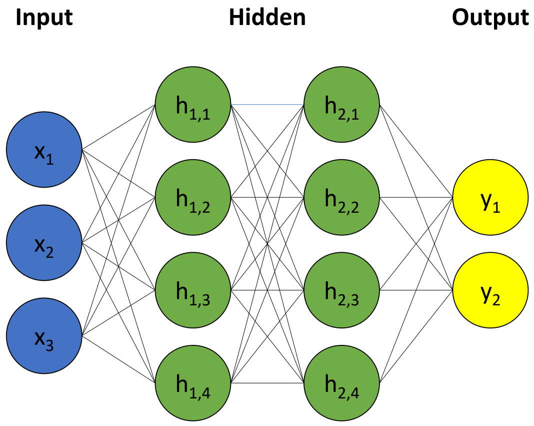


FIG. 3: Schematic of a simple feed forward neural network with two fully connected hidden layers.

The inputs into each neuron are not just modulated by the weights but are also 'squished' by some activation function.

The activation function maps the input to a point normally between 0 and 1 or -1 and 1. This is done for two reasons, firstly so the inputs don't blow up as they propagate through the network (especially a problem for networks with many neurons) and secondly so the network can learn complicated non-linear relationships between the input variables [16]. It can be shown that for a non linear activation function a network can be a universal function approximator [18]. In this study two activation functions were used; a sigmoid activation function and a rectified linear unit (ReLu). Sigmoid activation functions are usually placed in the output layer and are used for predicting probability based outputs. ReLu activation functions are often used within the hidden layers of a neural network. ReLu activation functions are the most widely used activation functions since their conception in 2010. They offer greater generalisability than the sigmoid activation function and train more efficiently [16].

Recurrent neural networks

The second type of network that was used in this project was a RNN, as shown in Figure 4. RNNs have a memory of the previous state of the network. This memory is passed into the next iteration of the network making recurrent networks perfect for sequential data where the order matters, such as time series data and natural language data. Recurrent networks are also optimal for dealing with variable length data such as the jet data in this study. Each jet within an event can be passed through the network sequentially (the data was ordered by jet P_t from highest to lowest) with each instance imparting information onto the network. After the final jet for an event is passed through the network, the network can make a prediction. The specific type of recurrent cell which was used was a long short term memory (LSTM) cell which specialise in retaining information over multiple inputs, with a slowly vanishing gradient unlike standard RNNs [19].

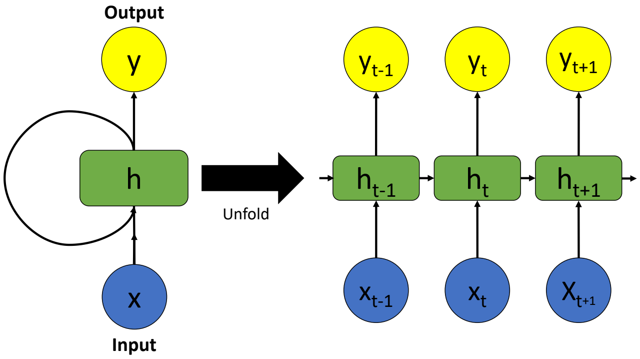


FIG. 4: Schematic of a recurrent neural network, the network is unfolded to show that the state of the network is passed between recurrent cells allowing the network to infer structure within the order of data.

Note it is not the case that a FFN could not be trained on the jet data. An input layer that was the size of the maximum number of jets in a single event multiplied by the number of jet variables would need to be constructed. The jet data for each event would then be fed in, ordered again by jet P_t with the missing jet data being made up by zero inputs. This would however create a very large input layer and would be very inefficient in terms of memory usage and training time. Though it might not make that much difference to the effectiveness of the network as the network would probably learn pretty quickly that the leading order jets are the most important for determining the event type.

Hyperparameter tuning

A hyperparameter is a model parameter whose value is used to control the training process. Hyperparameters have to be set before training. Because of this, there is an element of trial and error involved in optimizing networks. To optimise the networks in this study the hyperparameters were varied, the set of parameters which optimised the accuracy of each network were used. The accuracy of a classifier is given simply by:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}, \qquad 2$$

where TP is number of true positives, FP is the number of false positives, TN is the number of true negatives and FN is the number of false negative predictions. Figure 5 shows an example of how the different hyperparameters were assessed. The number of neurons in layer 1 were varied for the FFN. For each variation 50 networks were trained and the accuracy score recorded. Box plots were then made to show the distribution of the accuracy scores. In cases where it was unclear which network variation performs the best, as in figure 5, other factors were taken into consideration. In this case, 32 neurons were chosen, as the more neurons within each layer the longer the training time. Schematics of all the networks that were built for binary classification were created using Netron. The schematics are shown in figures 6, 7, and 8. For multi classification the number of output layers has to be varied from one to the number of classes that need to be distinguished.

RESULTS

Once the different network models had been optimized, they were assessed and their ability to distinguish between the different Higgs production modes and their dominant backgrounds were compared. The University of Bristol's high performance computing cluster, Blue Pebble, was used to train multiple networks. For the binary classification problem, where the models had to distinguish between ttH (signal) and its dominant backgrounds, 1000 FFNs, 500 RNNs

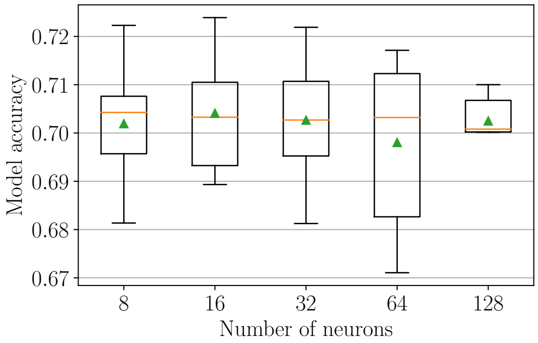


FIG. 5: Example of hyperparameter tuning for the number of neurons in layer 1 for the FFN.

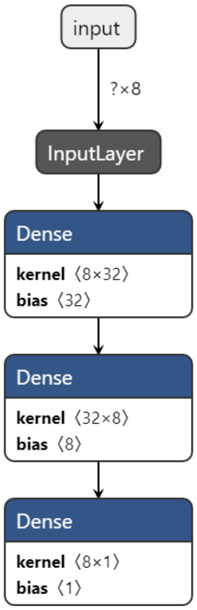


FIG. 6: Architecture of FFN used. The FFN consists of two hidden layers; layer one has 32 neurons, layer two has 8 neurons. Both hidden layers use a ReLU activation function and the final output layer has 1 neuron with a sigmoid activation function.

and 500 combined networks were built and trained. The networks were trained on 408,972 simulated events, with 99,305 of those being signal events and the remaining 309,667 being background events. The FFN was trained on the event data and the RNN was trained on the jet data, the combined network used both. Eighty percent of the data was used for training and twenty percent as a validation set. The data was shuffled differently each time the networks were trained; whilst the overall data set remained the same throughout, the data in the training and test set changed for each network. The average training curves for each of the different models is shown in figure 9. The networks generally train within two or three epochs which is shown by the training curves plateauing after about two epochs. The lack of divergence between the training and test curves implies that the networks are not over fitting.

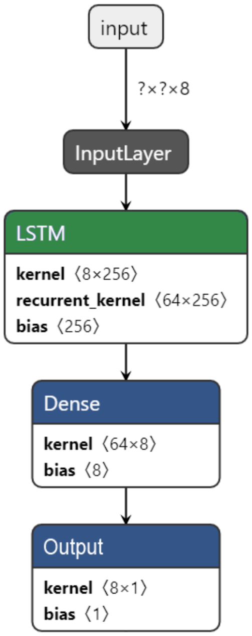


FIG. 7: Architecture of RNN used. The RNN consists of 2 hidden layers; layer one has 8 LSTM cells with a sigmoid activation function, layer two has a ReLu activation function and the final output layer has 1 neuron with a sigmoid activation function.

For a binary classifier a discriminator plot can be produced by plotting a histogram of the label prediction values for every event in the test set, see figure 10. The discriminator plot shows that the FFN has tried to separate the signal events from the background events, with the background distribution peaking at around 0.2, and the signal distribution peaking at around 0.8. However, there is a lot of overlap between the event and background distribution. To produce the results shown above and below, a discriminator threshold of 0.5 was used (unless specifically mentioned otherwise) were everything above 0.5 was classified as signal and everything below 0.5 was classified as background.

The simplest comparison between the different network models can be made by comparing the accuracy score of the different models when applied to the test data as seen in figure 11. According to the accuracy, the FFN and the combined network outperform the RNN. The combined network shows no discernible difference between itself and the FFN. However, accuracy can be an oblique and somewhat flawed metric [20]. It does not tell us how well the networks preform at distinguishing each individual class (signal and background) and for an unbalanced data set, like the one used in this project, it can give a false sense of predictive capability. If for instance there was a 1000 to 1 split between two classes, A and B, then by simply predicting A every time a network would receive an

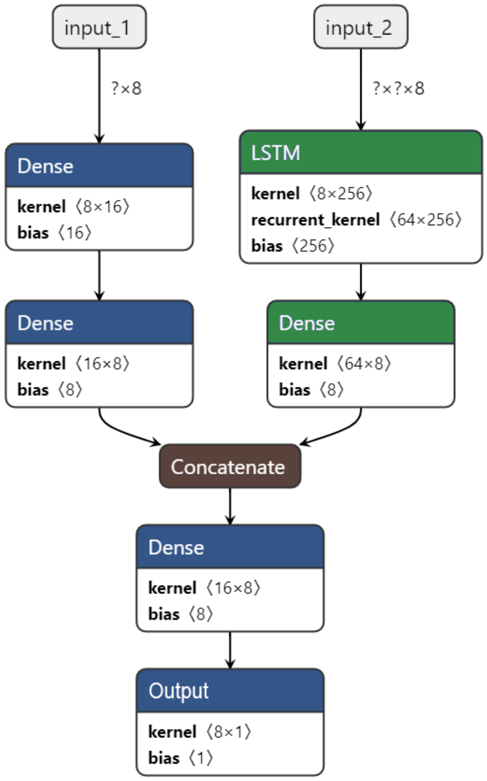


FIG. 8: Architecture of combined neural network used; the output of the FFN and RNN are combined and fed into a penultimate layer with 8 neurons and a ReLu activation function. The output layer consists of a single neuron with a sigmoid activation function.

accuracy score of 99.9% despite it having no actual predictive capability. For the ttH and tt¯ data set, if the networks were to predict every event as a background event they would score an accuracy of 76% because of the class imbalance. This false accuracy score is uncomfortably close to the accuracy scores recorded above.

For a more resolved view of the behaviour of the different models and to make sure that the models are not simply predicting a single class every time, confusion matrices can be made, as shown in figure 12. Values along the major diagonal of the confusion matrix show correct predictions and the values of the off diagonal show the error for each class. The confusion matrices shown in figure 12 were made by averaging the results for every network type.

The confusion matrices show that the networks can distinguish between ttH signal and SM background events to a reasonable degree. All the different models seem to perform worse at identifying background events compared to signal events. This could suggest that the weights are not quite right. However, the sample weights were summed and added to one for each class which implies the correct weighting. More likely, the network has a harder time identifying one of the background event types within the background composite class. If one of the background event types is harder to distinguish from ttH events, this could bring down the overall TN

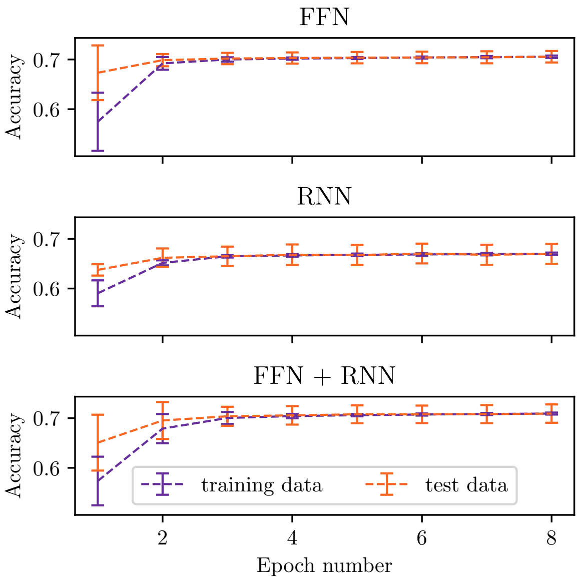


FIG. 9: The average training curve of the different network models for ttH binary classification. The average maximum test accuracy achieved by the networks after training was 0.71 ± 0.01 for the FFN, 0.67 ± 0.02 for the RNN and 0.71 ± 0.02 for the combined network.

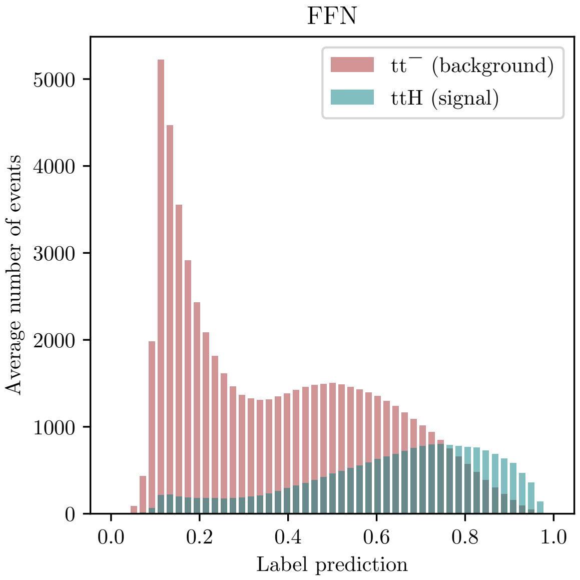


FIG. 10: Histogram of the FFN label predictions for each event in the test set for ttH binary classification. The histogram was produced by taking the average of the bin values for each of the FFNs that were trained.

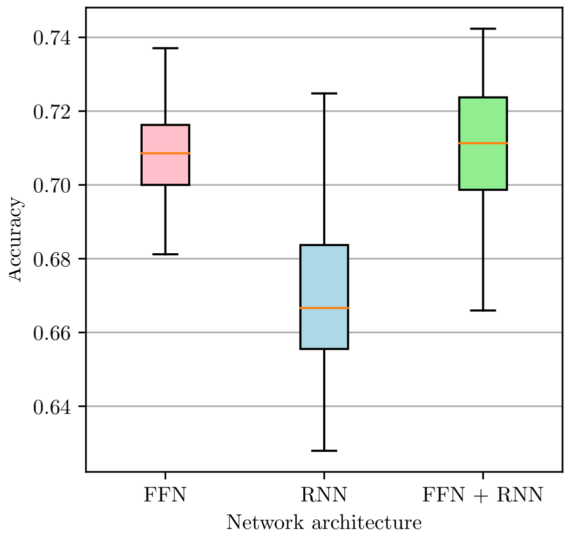


FIG. 11: Accuracy score of the different networks models for ttH binary classification. The box plots were produced by training each network model multiple times on the tth signal and background. The mean accuracy scores are FFN = 0.71 ± 0.01, RNN = 0.67 ± 0.02, and the combined network = 0.71 ± 0.02.

score. This hypothesis is given extra credence by the fact that there seems to be a second smaller peak in figure 10 at around 0.5. Multiple peaks in a discriminator plot can suggest multiple underlying structures i.e. different event types. A peak at 0.5 suggest that one of the background events is harder to distinguished from signal events. There is a similar shape for the other network models discriminator plots.

The RNN seems to be better at predicting signal events than the FFN and combined network, however, all the TP scores are within one standard deviation of each other. The performance of the RNN does not seem to transfer to the combined network which once again seems to perform more or less identically to the FFN. This seems to suggest that the combined network is putting a lot of weight on the FFN results and learning to 'ignore' the RNN results.

Receiver operator characteristic (ROC) curves are a useful way of showing the behaviour of a classifier as the discriminator threshold is varied. The area under a ROC curve (AUC) also provides a useful and commonly used metric for evaluating a classifiers performance. The AUC score has the advantage that it is not affected by skewed data and the AUC score is independent of the discriminator threshold [20]. To produce a ROC curve, the discriminator threshold is varied and the true positive rate (TPR) and false positive rate (FPR) are calculated at every discriminator position. TPR and FPR are calculated using:

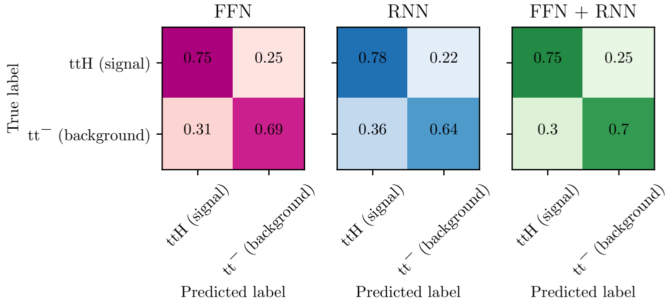


FIG. 12: Average confusion matrices of the different network models for ttH binary classification. The standard deviation on the different confusion matrix values are 0.02 for the FFN, 0.04 for the RNN, and 0.03 for the combined network.

$$TPR = \frac{TP}{TP + FN}, \tag{3}$$

$$FPR = \frac{FP}{FP + TN}. \tag{4}$$

ROC curves capture the trade off between TPR and FPR. For certain classification problems, it may be beneficial to tune the discriminator threshold to boost or reduce the TPR. For example for a corona-virus test you would want a high TPR even if that meant you got more false positives as false negatives can result in the virus spreading and worse public health outcomes[20].

Figure 13 shows the Average ROC curves for the different network models. The average ROC curves were produced by randomly sampling points along the ROC curves of multiple networks. The average of the sampled data sets were taken to produce the average ROC curve for each network type. The mean AUC score and standard deviation for each model type was taken as the AUC score and its error for each model type.

To understand a ROC curve some important points are needed. The first point (0,0) reflects the strategy of never issuing a positive classification. The opposite strategy of always predicting a positive classification is given by the point (1,1). Finally the point (0,1) represent perfect classification. The line y=x on a ROC curve represents random classification. A classifier whose ROC curve falls upon the line y=x has no predictive capability [20]. The ROC curve in figure 13, shows that the networks have discriminating power with the AUC scores showing a reasonable level of discrimination between the signal and background classes. The AUC score parameterize the overlap between the signal and background probability distributions for the different networks. For a network

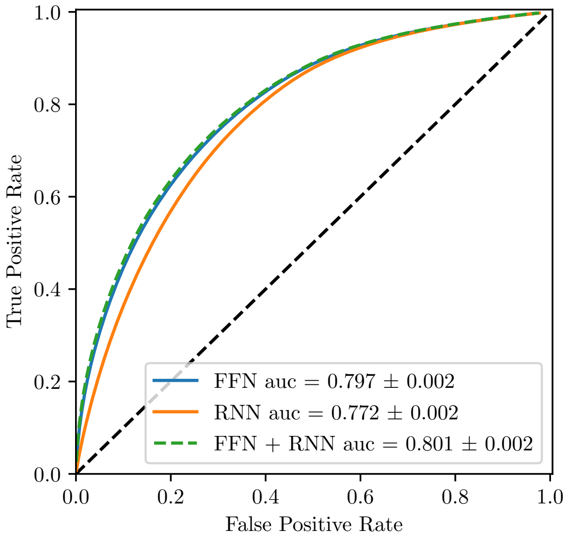


FIG. 13: Average ROC curves of the different network models for ttH binary classification.

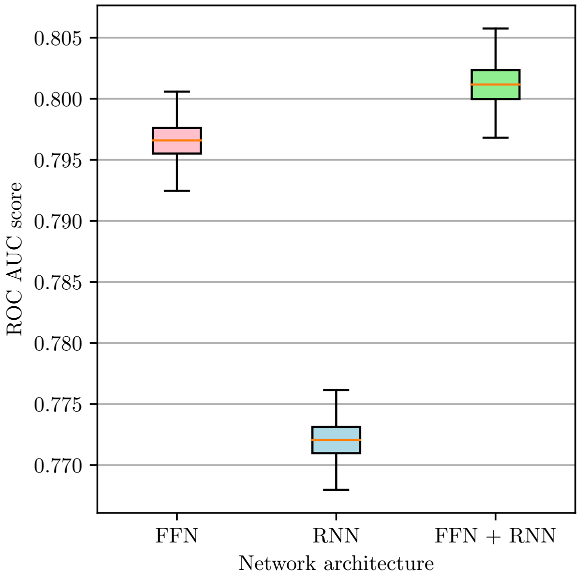


FIG. 14: Distribution of AOC scores for the different network models for ttH binary classification. The mean and standard deviation of the different models is shown in Figure 13.

with perfect predictive capabilities, the AUC score would be 1 and there would be no over-lap between the signal and background probability distributions. A network with no predictive capability would have an AUC score of 0.5 and the signal and background probability distribution would overlap completely [20]. The results shown in Figure 14 suggest that the combined network is better at distinguishing between signal and background events with over 50% of the combined networks performing better than the next best network the FFN.

When trying to detect invisible Higgs signal statistical significance, S, is important. Significance is a quantification of the probability of a random fluctuation causing an observed signal for some expected background [21]. An approximation of the statistical significance is:

$$S = \sqrt{2N_o \ln\left(\frac{s}{b}\right) - 2s}, \qquad 5$$

where N_o is the observed number of events, b is the expected number of background events, and s is the number of signal events (number of events above expected background) [21]. To understand the sensitivity of the FFN to a invisible Higgs signal at CMS, an estimation for the significance can be made assuming the cross section used to generate the invisible Higgs events is correct. To do this the cross-section weights of the events above the discriminator threshold are summed for the signal and background events. This gives the expected number of events for $1pb^{-1}$ integrated luminosity at CMS. During run 2, CMS collected $136fb^{-1}$ of integrated luminosity, and so to simulate the significance at CMS during run 2 the expected number of events was multiplied by 136000. Using a discriminator threshold of 0.5 and averaging over multiple networks the FFN was found to have an estimated significance of 3.26 ± 0.04 at CMS during run 2. Again, this is based on the cross-section that was used to generate the invisible Higgs events. The actual cross section of invisible Higgs events is not known.

One way of increasing the significance of the invisible Higgs signal is by tuning the discriminator threshold to further suppress the background. Figure **??** shows the significance score as the discriminator threshold is varied for the FFN. An optimal discriminator threshold for the FFN was found to be 0.66 ± 0.02 when the average discriminator threshold was taken. This results in a boosted significance of 3.39 ± 0.03 a 4% improvement.

Further improvements to the significance can be made by carefully considering the search channel. In the initial analysis the ttH channel was used. Further analysis of the other Higgs production modes was conducted. The networks shown in Figures 6, 7, and 8 were trained on the simulated ggF, VBF, WH, and ZH invisible Higgs events. Background from Z($\rightarrow v\bar{v}$) + jets, W($\rightarrow l\bar{v}$) + jets and QCD events were used in the analysis. The different network models were trained 100 times each on the different production modes over 8 epochs. Again, a validation set containing 20% of the data was used. The mean AUC score of the different channels are shown in Table II, further results can be seen in appendix D. The results show that the VH production modes were the most distinguishable from the background. ggF failed the training criteria for all but the FFN and even then the results were poor. The VBF channel also failed the training criteria for the RNN. The reason for this is probably due to the small number of VBF events in the training. There were only about

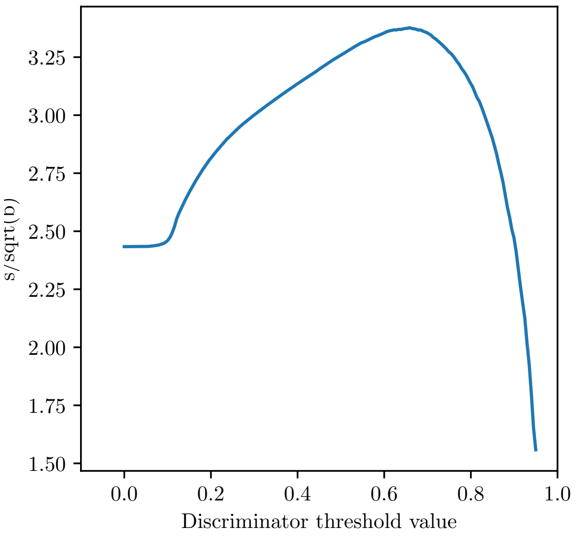


FIG. 15: Plot of the average significance score of the FFN for ttH binary classification as the discriminator threshold is varied.

	FFN	RNN	Combined
ggF	0.597 ± 0.002	N/A	N/A
VBF	0.78 ± 0.02	N/A	0.80 ± 0.01
WH	0.86 ± 0.02	0.85 ± 0.01	0.869 ± 0.005
ZH	0.857 ± 0.006	0.80 ± 0.01	0.85 ± 0.01
ttH	0.797 ± 0.002	0.772 ± 0.002	0.801 ± 0.002

TABLE II: Mean AUC scores of different network models when trained on different Higgs production modes.

4000 events in the VBF data set. Even with the weighting it is not clear that the RNN could train on such a limited data-set. The VBF results may be improved if a larger training data-set was used. Further tuning of the networks for the specific production modes may increase the AUC scores of the networks. Individual tuning could not be conducted due to time constraints.

To determine the most sensitive channel for detecting an invisible Higgs event using the FFN, a predicted significance and boosted significance was calculated using the method described above for each production mode. The results are summarised in Table III. It was found that the WH channel was the most sensitive channel, with a boosted significance score of 3.64 ± 0.06.

Multiple production channels can also be considered to improve the significance score. By changing the output layer from one neuron to multiple neurons, multiple classifications can be made by the same network. Sixty multi-classifier combined neural networks were built and trained on the simulated data for the ttH and WH production channels, as these were determined to be the most sensitive using the previously outlined analysis. The different background processes were split into a ttH background and a WH background, denoted by tt¯ and W¯ respectively. A schematic of the multi classifier network architecture, as well as a the multi classifier training curve, can be seen in appendix E. The confusion matrix for the multi-classifier combined network is shown in figure 16. The multi-classifier performs worse at identifying ttH and WH invisible Higgs events than the binary classifier, but performs better at identifying tt¯ events. The worse performance at detecting ttH and WH events is expected as there are now more processes for the network to discriminate against. However, the better performance at classifying tt¯ events is unexpected, the reason for the better performance is unknown. The network performs worst at identifying W¯ events. This could be because the W¯ background is a composite of the Z(→ $v\bar{v}$) + jets, W(→ $l\bar{v}$) + jets, and QCD backgrounds. The network could have learnt to discriminate based on MET, which would be an effective strategy for distinguishing the QCD background [6], the largest background by weighting, but would result in a poor performance when distinguishing the other backgrounds. However, this does not seem to be a problem for the binary classifier, so further investigation is required. To test the above hypothesis the QCD events could be separated into their own background class.

	Significance	Optimal discriminator	Boosted significance
VBF	1.8 ± 0.3	0.7 ± 0.1	2.4 ± 0.2
WH	3.5 ± 0.1	0.61 ± 0.05	3.64 ± 0.06
ZH	2.07 ± 0.08	0.7 ± 0.1	2.2 ± 0.1
ttH	3.26 ± 0.04	0.66 ± 0.02	3.39 ± 0.03

TABLE III: Mean significance of different production modes when a discriminator threshold of 0.5 is used compared to significance score when optimal discriminator threshold is used. All the results above are created using the FFN.

The ROC curves for the multi classifier is shown in Figure 17. The ROC curves for each individual class are generated by treating each class as a binary classification problem, with the classifier either predicting correctly or incorrectly for said class. The micro average is calculated by aggregating the contributions of all classes to compute the average metric. The AUC scores for the various classes shows a high level of discrimination. It would be interesting to estimate the significance for the multi classifier neural network however, due to time constraints this could not be done. The multi classifier was tuned but not extensively because of the time it takes to train. Better tuning as well as distinguishing QCD as a separate background could improve the multi classifiers performance further.

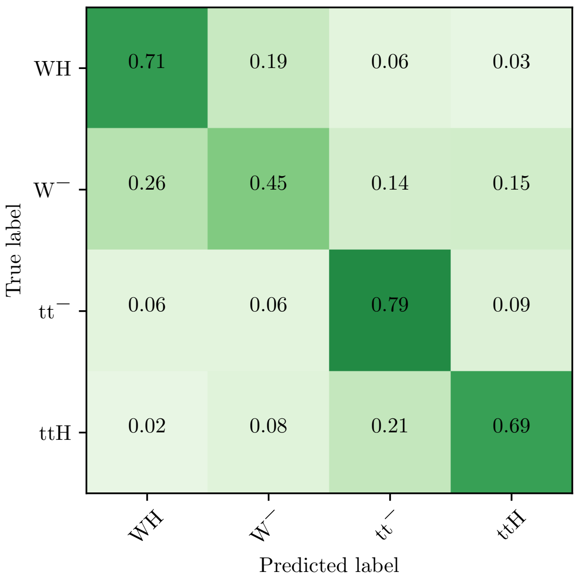


FIG. 16: Confusion matrix for the multi-classifier combined neural network

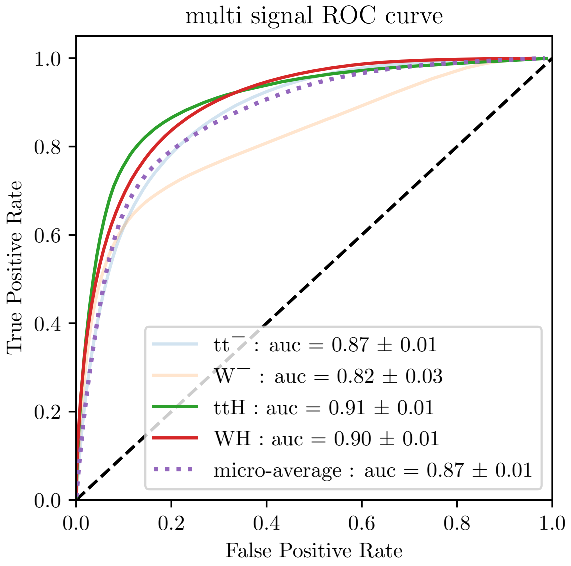


FIG. 17: ROC curves for the multi-classifier combined neural network.

FUTURE DIRECTION

Throughout the report, suggestions for extensions to this project have been mentioned. In this section those extension mentioned previously and more will be summarised. The first and most basic extension would be to apply the networks to an experimental data set and to work out an upper-bound of Br($H \to inv$). To do this, further analysis of the different model's uncertainty as well as the model's response to systematic errors, would have to be conducted. It would be interesting to see how the neural networks compare to the selection threshold method, and whether applying the analysis outlined in this report would improve previous upper-bounds on the Br($H \to inv$) if applied to the same data set. An estimate of how the neural networks compare to the selection thresholds could be made by removing the pre-selection cuts and tailoring specific cuts to the different production channels to see how the two methods compare when using the simulated data.

As previously mentioned, the analysis of the VH channels only considered hadronic final states; there was a lepton veto in the pre-selection cuts. It may be beneficial to remove this pre-selection threshold as the presence of leptons may help the networks to distinguish the VH events to a greater degree. This may help improve the multi-classifier combined neural network, as well as categorizing QCD events as its own background class. To further improve the multi-classifier, as well as the binary classifiers for the different production channels, further tuning is required. A logical next step for the multi-classifier is to include the other Higgs production channels as their own classes. This was attempted but the results were poor. Again due to how long it took to train, the proper tuning to build a network capable of classifying the full Higgs production space could not be built.

The results from the combined neural network were not as distinct from the FFN as was hoped. Exploring other network architectures, as well as other network models, would be a reasonable extension to this project. In this project ReLu and sigmoid activation functions were used. The ability of a network can be improved by tailoring the activation functions to the specific problem [16]. A greater consideration of the behaviour of invisible Higgs events may reveal a better activation function for classification. Other network models that could be explored are graph neural networks and transformer neural networks (TNN). TNNs would solve one of the major limiting factors in this project; the time taken to train the RNN and the combined neural network [22].

In the future development of new accelerators would be able to refine measurements of Br($H \to inv$) to an unprecedented degree. Measurements of the Br($H \to inv$) at the LHC have a theoretical limit of 2.5% because of pile up caused by the collision of other partons within a proton-proton collision. The proposed International Linear Collider (ILC) which would collide electrons with positrons would have a clean collision environment. Current estimates put the theoretical limit for measurements of the Br($H \to inv$) at the ILC at 0.23% [9].

CONCLUSION

This project looked at whether neural networks trained on high level event variables could be used to distinguish invisible Higgs event from their SM background. Initially, work was focused on the ttH channel. Three different binary classifiers were built and trained multiple times on simulated ttH data. A FFN, a RNN, and a combined neural network were built. The networks had a mean accuracy of 0.71 ± 0.01, 0.67 ± 0.02, and 0.71 ± 0.02 respectively when tested on a validation set. The networks showed a reasonable level of discrimination with the different models receiving an AUC score of 0.797 ± 0.002, 0.772 ± 0.002, and 0.801 ± 0.002 respectively.

An estimation for the significance of an invisible Higgs signal that would be detected at CMS using the FFN and simulated data was made. It was predicted that the FFN would detect a invisible Higgs signal with a significance of 3.26 ± 0.04. Further tuning of the discriminator threshold boosted the significance by 4% to 3.39 ± 0.03. These calculations were made using the assumption that the cross-section used to generate the invisible Higgs events was correct. This is not the significance that would be expected at CMS. This exercise was done to tune the FFN and find the optimal discriminator threshold for the FFN; this was found to be 0.66 ± 0.02.

Further analysis considered the other Higgs production modes. It was found that networks performed the best at distinguishing the VH channels. For the WH channel a AUC score of 0.86 ± 0.02 was recorded for the FFN, 0.85 ± 0.01 for the RNN, and 0.869 ± 0.005 for the combined network. An optimal discriminator threshold for the FFN was found to be 0.61 ± 0.05, which boosted the significance of the invisible Higgs signal by 4%. For the ZH channel a AUC score of 0.857 ± 0.006 was recorded for the FNN, 0.80 ± 0.01 for the RNN and 0.85 ± 0.01 for the combined network. A optimal discriminator threshold of 0.7 ± 0.1 was observed, this boosted the significance of an invisible Higgs signal by 6%.

Finally, a multi-classifier combined neural network was built. Despite insufficient Hyperparameter tuning, the multi-classifier combined neural network received a micro-average AUC score of 0.87 ± 0.01, showing a good level of discrimination between the signal and background classes.

Further analysis is required to determine whether the methods used in this project can replace the tried and tested selection threshold method. The pre-selection cuts on the simulated data should be removed and tailored selection thresholds should be used to determine the significance of an invisible Higgs signal.

Appendices

APPENDIX A

Transverse momentum - Transverse momentum is the momentum transverse to the beam line. The initial momentum of partons involved in a proton-proton collision is unknown as the partons that make up a proton share the protons momentum. The initial momentum transverse to the beam is known however, it is zero and so the total transverse momentum of a proton-proton collision must add to zero. This makes the transverse momentum a useful quantity.

Transverse Energy - The transverse energy is defined as the negative vectorial sum of the transverse momentum.

Leading jets - Jets are ordered by their transverse momentum, leading jets refers to jets with the greatest transverse momentum.

Rapidity - Rapidity is defined as

$$y = \frac{1}{2} \ln \left(\frac{E + P_z c}{E - P_z c} \right)$$

where z is taken as the axis along the beam line, E is the energy of a particle, P_z is the momentum of the particle in the z direction, and c is the speed of light. Rapidity is a useful quantity in particle accelerators. A particle traveling perpendicular to the beam axis will will have a small rapidity. A particle traveling in the forward direction along or close to along the beam axis would have a large positive rapidity. A particle traveling in the backwards direction along or close to along the beam axis would have a large negative rapidity [23]. The forward and backwards direction are a matter of definition.

APPENDIX B - THE STANDARD MODEL

SM is a Quantum Field Theory (QFT). In QFT the evolution of a system is described by the Lagrangian density which encodes the physics of the system. The Lagrangian density is a function of the fields that exist within a system and their derivatives. For a single scalar field ψ the Lagrangian density will take the form $L(\psi, \nabla\psi, \frac{\partial\psi}{\partial t}, x, t)$ [24].

If QFT is the theoretical framework that the SM is built on, then observational evidence is used to construct the SM Lagrangian. To start spinor fields are introduced to describe the fermions. Perturbations in the spinor fields give rise to the quarks and leptons. Then, by demanding that the SM Lagrangian is invariant under the local gauge transformations SU(3), SU(2) and U(1), the gauge fields are recovered. The gauge fields are the force fields of the SM and describe the fundamental forces (excluding gravity). They allow the fermions to interact with each other. The interactions are mediated by the gauge bosons which are perturbations in the gauge fields. The gauge bosons are the gluon, the $W^{1,2,3}$ bosons and the B boson. These are massless [24]. However, there is a problem; the boson are predicted to be massless but the W and Z bosons have mass. The Higgs field and electroweak symmetry breaking are needed to explain the mass of the W and Z bosons [24].

APPENDIX C - FEATURE VARIABLES

Cone type algorithms are used to group the products of hadrionzation together into a single jet object. Clean jets refer to the output of the grouping algorithm in the case where jets are distinguishable. Cone type algorithms work by selecting the most energetic particle as a seed and drawing a cone of radius r, every particle within the radius is considered part of the jet [15]. In this project alternative angles to the standard η (polar angle) and ϕ (azimuthal angle) variables were used. For more information see [25].

Event Variables:

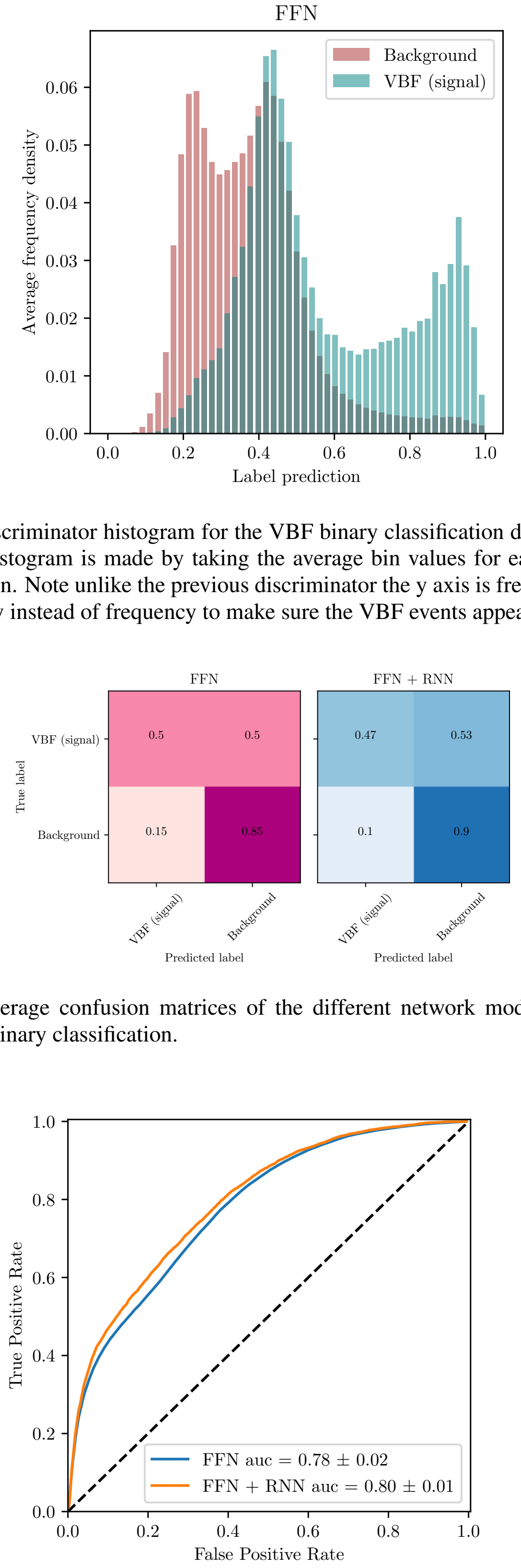
- DiJet mass - Invariant mass of two leading order jets
- HT - Scalar sum of clean jet P_t
- MHT - Magnitude of vector sum of clean jet P_t
- MinChi - Angular variable see [25]
- MinOmegaHat - Angular variable see [25]
- MinOmegaTilde - Angular variable see [25]
- nMediumBJet - Number of jets identified as coming from b quarks
- ncleanedJet - Number of clean jets in the event

Jet Variables:

- Clean Jet btag - DeepB tagging algorithm's discriminator output, how likely is the jet to have originated from a b quark.
- Clean Jet eta - Polar angle of the jet
- Clean Jet mass - Mass of all jets
- Clean Jet phi - Azimuthal angle of the jets
- Clean jet pt - Transverse momentum of all clean jets
- Clean Jet area - Area of jet when projected onto the eta-phi plane

APPENDIX D - BINARY-CLASSIFIER

VBF

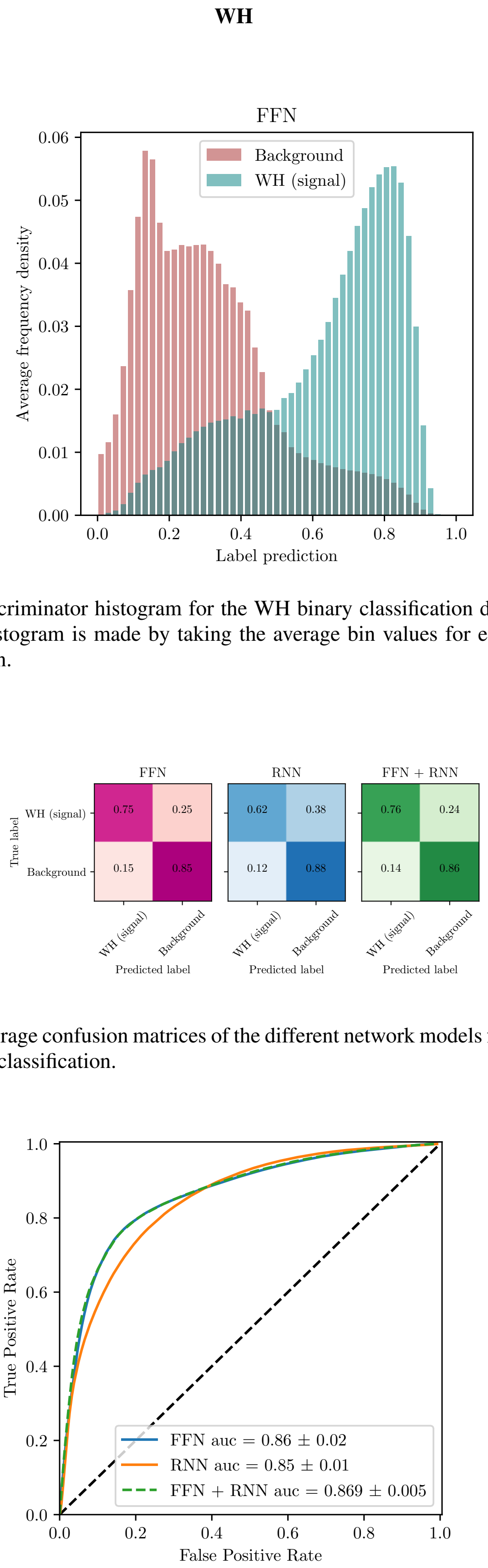


(a) Discriminator histogram for the VBF binary classification data-set. The histogram is made by taking the average bin values for each bin position. Note unlike the previous discriminator the y axis is frequency density instead of frequency to make sure the VBF events appear.

(b) Average confusion matrices of the different network models for VBF binary classification.

(c) Average ROC curves of the different network models for VBF binary classification.

FIG. 18: Results from VBF analysis

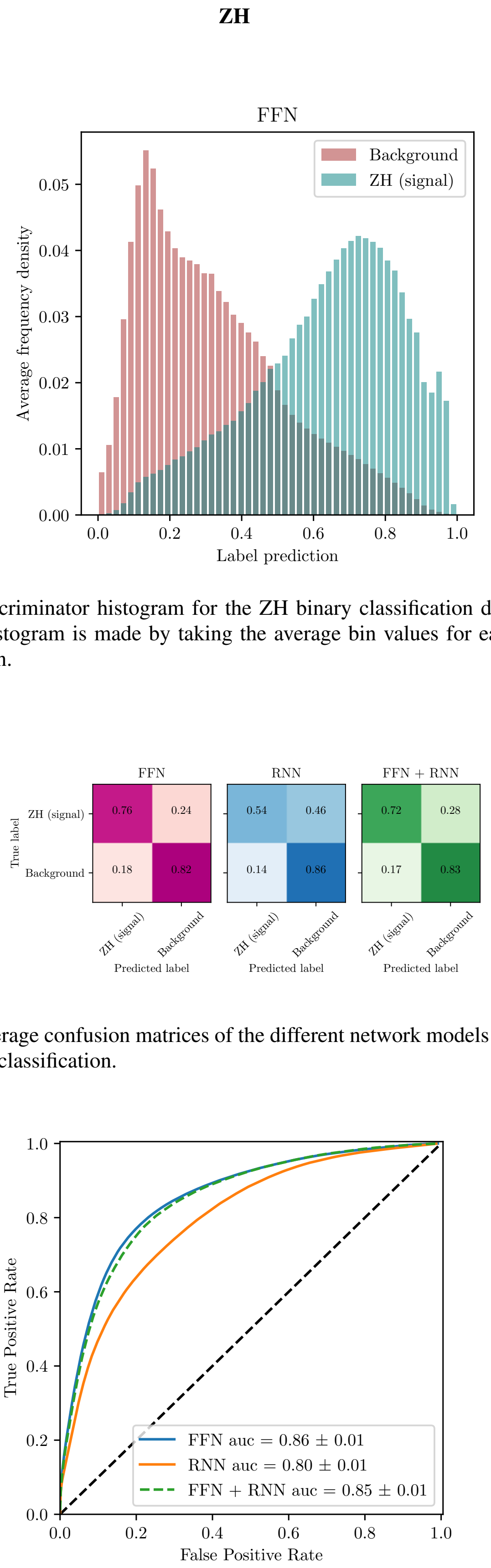


(a) Discriminator histogram for the WH binary classification data-set. The histogram is made by taking the average bin values for each bin position.

(b) Average confusion matrices of the different network models for WH binary classification.

(c) Average ROC curves of the different network models for Wh binary classification.

FIG. 19: Results from WH analysis



(a) Discriminator histogram for the ZH binary classification data-set. The histogram is made by taking the average bin values for each bin position.

(b) Average confusion matrices of the different network models for ZH binary classification.

(c) Average ROC curves of the different network models for ZH binary classification.

FIG. 20: Results from ZH analysis

APPENDIX E - MULTI-CLASSIFIER

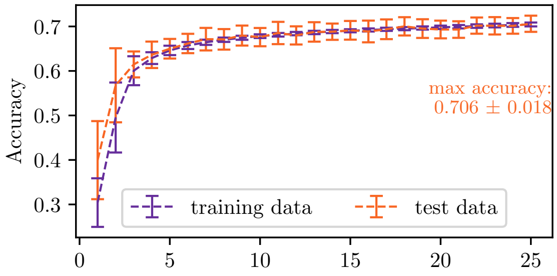


FIG. 21: Training curve of multi classifier combined neural network. The multi classifier was not just larger in terms of neurons but also took more epochs to train than the binary classifier.

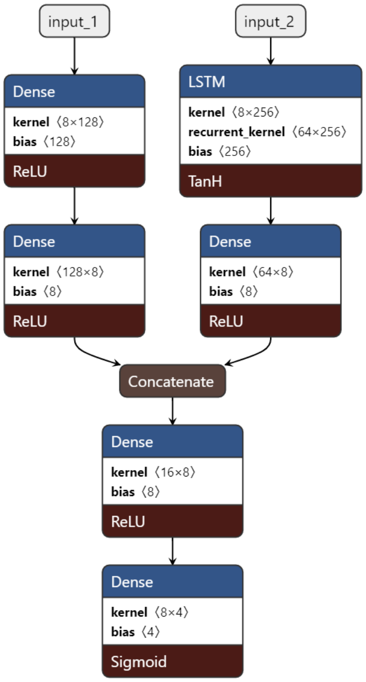


FIG. 22: Architecture of multi-classifier neural network used in this project.

[1] K. Freese, “Review of Observational Evidence for Dark Matter in the Universe and in upcoming searches for Dark Stars,” *EAS Publ. Ser.*, vol. 36, pp. 113–126, 2009.

[2] S. P. Martin, “A Supersymmetry primer,” *Adv. Ser. Direct. High Energy Phys.*, vol. 21, pp. 1–153, 2010.

[3] G. Bélanger, F. Boudjema, A. Cottrant, R. Godbole, and A. Semenov, “The mssm invisible higgs in the light of dark matter and g-2,” *Physics Letters B*, vol. 519, no. 1, pp. 93 – 102, 2001.

[4] G. F. Giudice, R. Rattazzi, and J. D. Wells, “Graviscalars from higher dimensional metrics and curvature Higgs mixing,” *Nucl. Phys. B*, vol. 595, pp. 250–276, 2001.

[5] K. Belotsky, D. Fargion, M. Khlopov, R. Konoplich, and K. Shibaev, “Invisible Higgs boson decay into massive neutrinos of fourth generation,” *Phys. Rev. D*, vol. 68, p. 054027, 2003.

[6] D. Ghosh, R. Godbole, M. Guchait, K. Mohan, and D. Sengupta, “Looking for an Invisible Higgs Signal at the LHC,” *Phys. Lett. B*, vol. 725, pp. 344–351, 2013.

[7] “First constraints on invisible Higgs boson decays using $t\bar{t}H$ production at $\sqrt{s} = 13$ TeV,” 2019.

[8] V. S. Ngairangbam, A. Bhardwaj, P. Konar, and A. K. Nayak, “Invisible Higgs search through Vector Boson Fusion: A deep learning approach,” *Eur. Phys. J. C*, vol. 80, no. 11, p. 1055, 2020.

[9] A. Steinhebel, J. Brau, and C. Potter, “H→invisible at the ILC with SiD,” in *International Workshop on Future Linear Colliders*, 4 2021.

[10] “Combination of searches for invisible Higgs boson decays with the ATLAS experiment,” 10 2020.

[11] A. M. Sirunyan *et al.*, “Search for invisible decays of a Higgs boson produced through vector boson fusion in proton-proton collisions at $\sqrt{s} = 13$ TeV,” *Phys. Lett. B*, vol. 793, pp. 520–551, 2019.

[12] M. Tanabashi, K. Hagiwara, and et al, “Review of particle physics,” *Physical Review D*, vol. 98, 8 2018.

[13] G. Aad, T. Abajyan, and et al, “Observation of a new particle in the search for the standard model higgs boson with the atlas detector at the lhc,” *Physics Letters B*, vol. 716, no. 1, pp. 1 – 29, 2012.

[14] “Search for invisible Higgs boson decays with vector boson fusion signatures with the ATLAS detector using an integrated luminosity of 139 fb^{-1},” 4 2020.

[15] M. H. Seymour, “Jets in QCD,” *AIP Conf. Proc.*, vol. 357, pp. 568–587, 1996.

[16] C. Nwankpa, W. Ijomah, A. Gachagan, and S. Marshall, “Activation functions: Comparison of trends in practice and research for deep learning,” 12 2020.

[17] S. Ruder, “An overview of gradient descent optimization algorithms,” 9 2016.

[18] G. Cybenko, “Approximation by superpositions of a sigmoidal function,” *Mathematics of Control, Signals, and Systems*, vol. 2, pp. 303–314, 1989.

[19] R. M. Schmidt, “Recurrent neural networks (rnns): A gentle introduction and overview,” *CoRR*, vol. abs/1912.05911, 2019.

[20] T. Fawcett, “An introduction to roc analysis,” *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861–874, 2006. ROC Analysis in Pattern Recognition.

[21] D. Griffiths, *Introduction to elementary particles*. 2008.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention is all you need,” 2017.

[23] C. Y. Wong, *Introduction to high-energy heavy ion collisions*. 1995.

[24] S. F. Novaes, “Standard model: An Introduction,” in *10th Jorge Andre Swieca Summer School: Particle and Fields*, 1 1999.

[25] T. Sakuma, H. Flaecher, and D. Smith, “Alternative angular variables for suppression of QCD multijet events in new physics searches with missing transverse momentum at the LHC,” 3 2018.