**ANN and SVM based classification system for classifying Higgs Boson Decay against background noise and using genetic algorithm to select the optimal feature subset.**

Harsh Kumar Chanda Kumari

hkumar2@ph.iitr.ac.in ckumari@ph.iitr.ac.in

Enrolment No – 18122008 Enrolment No - 18122006

**Abstract**

The project presents a classification system for Higgs Boson decay based on Artificial Neural Networks and SVM. Higgs Boson dataset from Kaggle has been used for testing the model. The system has been made more accurate and time efficient by the use of Genetic Algorithm which selects optimal feature subset form total features and ultimately enhances the system’s classification accuracy. The overall accuracy achieved by the system is 98.31 percent which shows the potential of the system to be used for practical purposes.

1. **Main Objectives**

* Classification of Higgs Boson Decay using Artificial Neural Networks.
* Using Genetic Algorithm to find the optimal feature subset for classification with help of SVM.

1. **Status and Other Details**

* Completed
* Percentage Contribution of Members
* Harsh – 50%
* Chanda – 50%
* Total time spent on project – 3 weeks

1. **Major stumbling blocks**

* High computational time of software which reduced the speed of progress of the entire project.
* Absence of any published work to test the model’s performance.

1. **Introduction**

The ATLAS experiment and the CMS experiment recently claimed the discovery of the Higgs boson [1,2]. The discovery was acknowledged by the 2013 Nobel prize in physics given to Franc¸ois Englert and Peter Higgs. This particle was theorized almost 50 years ago to have the role of giving mass to other elementary particles.

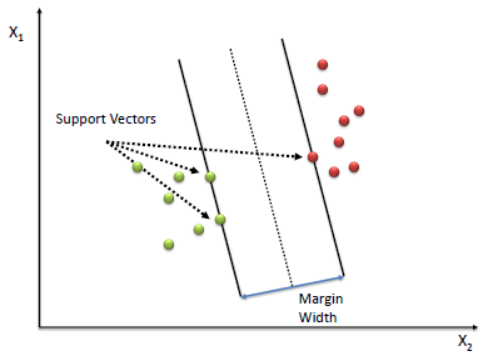
The Higgs boson has many different processes through which it can decay. When it decays, it produces other particles. In physics, a decay into specific particles is called a channel. The Higgs boson has been seen first in three distinct decay channels which are all boson pairs. One of the next important topics is to seek evidence on the decay into fermion pairs, namely tau-leptons or b-quarks, and to precisely measure their characteristics. The first evidence of the H to tau tau channel was recently reported by the ATLAS experiment [3]. The goal of this challenge is to try and improve the analysis. The entire technical background of this challenge can be found in [4].

In this project we present a classification model for Higgs Boson decay against background noise. The dataset was taken from Kaggle [5]. The dataset consists of 31 features in which 17 are primitive features and 14 are derived features. The dataset is divided into 2 classes namely (s and b). s represents signal and b represents background. In order to improve the computational efficiency and to avoid curse of dimensionality we use Genetic Algorithm supported with SVM to find the optimal feature subset which maximizes overall classification accuracy with reduced number of features. A genetic algorithm to select optimal feature subset for use with back propagation artificial neural networks has been described in [6]. We have used SVM in Genetic algorithm for feature selection as it is fast to train as compared to ANN. A genetic algorithm for feature selection as well as for optimization of SVM parameter has been proposed in [7]. The optimal features obtained are then used for training 8 different models of ANN with different values of hyperparameters and the model achieving the highest classification accuracy was selected.

The rest of the project goes as follows. Section 5 describes basic theory of SVM. Section 6 describes the theory of Artificial Neural Networks. Section 7 summarizes the employed Genetic Algorithm in this model. Section 8 discusses the experimental results and finally Section 9 concludes the project with some general remarks.

1. **Support Vector Machines**

The concept of Support Vector Machines (SVM) was originally introduced for binary classification problem. It finds a decision boundary or hyperplane such that it separates the two sets sets in such a way that the distance between the hyperplane and nearest point of each of the data sets (support vectors) is maximum [8].



The binary classification problem may be formulated as follows.

Given a training set withinput features andclassification output , of the form

Where and

Here N represents total number of samples and m represents total number of features.

In SVM method, optimal margin classification for linearly separable input patterns is achieved by finding a hyperplane in m dimensional space [9]. The hyperplane must linearly separate the two classes {+1,-1} on its either side. The equation of the decision surface is given by

Here ***w*** is the weight vector and **b** is the hyperplane bias. The equations corresponding to the two classes can be represented mathematically as:

for

for

The support vectors are the training data points for which

that is, the points for which the corresponding inequalities are binding.

The distance between hyperplanes is and the problem is of maximizing this distance or minimizing .

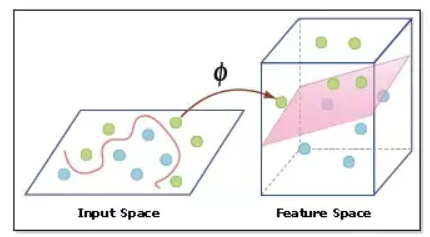
Thus, we can formulate the quadratic optimization problem:

Find and such that

is minimized,

subject to the constraint

The non-linear classification problem can also be solved by introducing concept of Soft Margin and mapping data to a High-Dimensional space by finding some kernel function Φ(x) for this purpose such that the problem changes into a linear classification problem (Figure 2).



Now the mathematical formulation becomes:

Min , such that

;

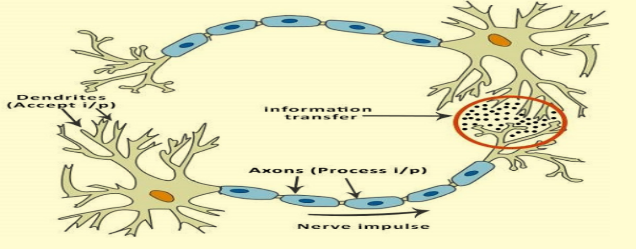
In soft margin problem, we introduce extra cost term to penalize for misclassified instances and those within the margin.

Though the concept of SVM was originally proposed for binary classification, various methods have been proposed to use SVM for multi-class problems also. “One against One” and “One against All” methods are among the most popular methods for multi-class classification problems [10]. The former involves constructing *pC2* binary classifiers, one for each pair of a total of *p* classes. The final class of the test point is determined by a pre-defined voting mechanism. In the “One against All” method, there is a binary classifier for each class to separate the members of that class from all other classes.

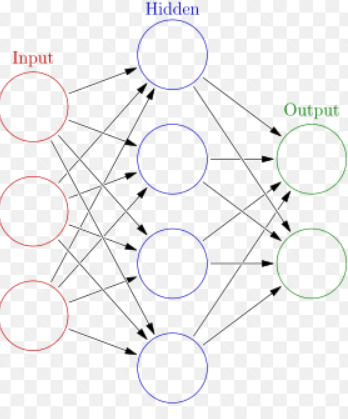
1. **Artificial Neural Networks**

Artificial Neural Networks are basically a computational model that is based on the structures and functions of biological neural networks.

Generally, the working of Human brain by making the right connections is the idea behind ANN. In brains the neurons are used to transmit messages and inputs are received from sensory organs.



Similarly, in ANN’s there are different layers consisting of various number of nodes. The first layer is the input layer and last layer is output layer. The remaining layers in between are hidden layers whose nodes contain specific information. All the nodes in certain layer are connected to all the nodes in the previous layer i.e the entire network is fully connected.



The ANN’s can be used to perform variety of takes form regression to binary classification and multi classification. The processes performed by ANN are:

* Training Process
* Feed Forward
* Calculation of Loss function
* Backpropagation to update weights.
* Testing process
* Feed forward for predicting values or classes.

The classification problem can be formulated as:

Given a set *S* with input features *Xi* and classification output *di* of the form:

Where and .

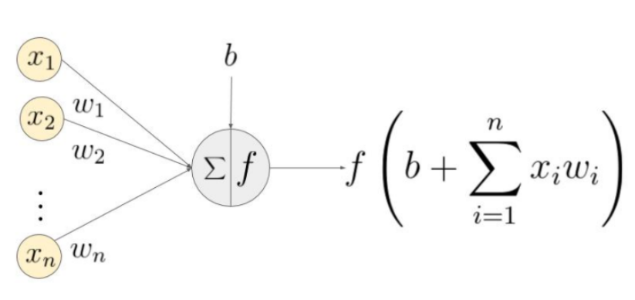
Here N represents total number of samples, m represents total number of features and p represents total number of classes.

It is to be noted that the total number of nodes in the input layer of ANN is equal to the total number of features and similarly total number of nodes in the output layer is equal to the total number of classes corresponding to each class.

*Feed forward:*

The feed forward has been explained in the paper [11].

Each connection between the nodes has a certain weight and bias.



Let be the first node of second layer

be the nodes of first layer for

where k is the total number of nodes on the first layer.

The value of is calculated as follows:

Where and are corresponding weights and bias of the connection between node of first layer and 1st node of second layer.

The value of generated is then passed to an activation function which determines whether the neuron should be fired or not [12]. It also introduces non-linearity in the network.

Some common activation functions used are:

Sigmoid function –

Relu function –

Hyper Tangent function –

Binary Step function –

The output obtained after passing through activation function is stored in the corresponding node and is used for calculation of node values of the further layers in the similar way as described above.

Since the output layer of ANN is used for determining classes, the activation function applied is such that it converts the values of all nodes in the final layer into probabilities which represents the confidence of ANN to classify a given input into certain class. The activation function most generally used for converting values of output nodes into probabilities is:

SoftMax activation function-

The node with the highest value of probability is taken as the class predicted by the ANN for a particular input.

*Calculation of Loss function:*

When modelling a classification problem, we are interested in mapping input variables to a class label and in order to improve the model’s performance we define a loss function which gives a scalar measure of the deviation of model’s output form the target values. Since we are dealing with classification problem here, the loss function used is:

Cross Entropy Loss function-

Where is the softmax output of the node of output layer.

Our goal is to minimise the loss function in order to improve model’s performance.

*Backpropagation:*

The backpropagation algorithm was originally introduced in the 1970s, but its importance wasn't fully appreciated until the famous paper [13]. That paper describes several neural networks where backpropagation works far faster than earlier approaches to learning, making it possible to use neural nets to solve problems which had previously been insoluble.

The goal of minimising the loss function in backpropagation is achieved by changing the weights and biases at each step in the direction of steepest descent of cost function.

Let be the derivative of the loss function w.r.t certain weight then the weight update is given by:

where is arbitrary learning rate.

The process of feed forward and backpropagation is repeated for certain number of times called as epochs, till the loss function has reduced by considerable amount.

After training the model, test inputs are fed and the class with maximum probability is taken to be the predicted value of model for a certain input.

1. **Genetic Algorithm**

Genetic Algorithms (GA) are a family of computational models inspired by evolution. These algorithms encode a potential solution to a specific problem on a simple chromosome like data structure and apply recombination operators to these structures so as to preserve critical information [14]. In this project we use integer coded genetic algorithm which selects top N features out of a total of M features. We first initiate a population of 40 chromosomes where each chromosome is a vector of size N consisting integers which represent a feature. The initiation is done randomly.

Since we are interested in selecting optimal feature subset for classification, we use the classification accuracy achieved by a feature subset through SVM as its fitness value. The population is subjected to *selection* process where the chromosomes are sorted according to the fitness value and the first half of them are taken into the new population. Now we generate 20 children chromosomes by subjecting the previous 40 chromosomes to crossover operation. In crossover operation two chromosomes are randomly selected called as parent chromosomes and a child chromosome which is a vector of size N is initiated. Now each gene of child is either from the first parent or the second parent which depends on the crossover probability which was selected as 0.5 after experimentation.

After crossover the children chromosomes are subjected to mutation which maintains diversity from one generation of population to next by randomly changing a gene with certain probability called as mutation rate which was selected as 0.2.

It can be clearly seen that after crossover and mutation there are chances of gene duplication which is not desired. Therefore, the gene duplicates are removed through duplicacy removal step in which the duplicates are removed by selecting a random integer form 1 to M such that there are no duplicates after this operation.

The initial half population are combined with the children chromosomes to generate population of original size and this step is repeated till there is saturation in the best classification accuracy of the population.

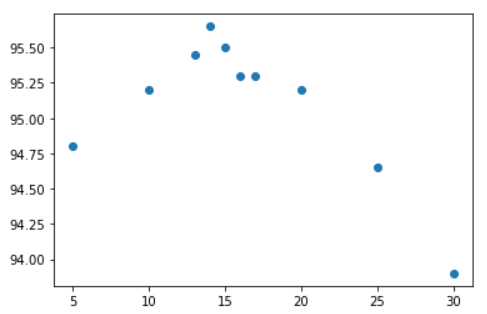
1. **Experimental results**
   1. **Dataset Used**

The dataset we used to test our model was Higgs Boson Machine Learning Challenge taken from Kaggle. There are a total of 31 features. The features prefixed with PRI (for PRImitives) are “raw” quantities about the bunch collision as measured by the detector, essentially the momenta of particles. Those prefixed with DER (for DERived) are quantities computed from the primitive features. The classes have 2 different integers 0 and 1. 0 represents background noise and 1 represents Higgs boson Decay. 7000 samples are used for training SVM and 3000 for testing for calculation of fitness value in Genetic Algorithm. 30000 samples with the best feature subset are used for training ANN and 20000 are used for testing.

* 1. **Experimental Results**

The Genetic algorithm was used to select the best N features out of 31 features and the corresponding features obtained by different values of N are summarized in table below:

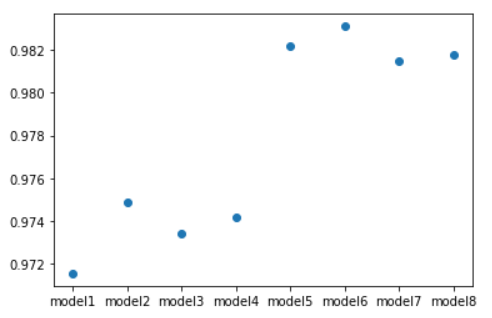
|  |  |
| --- | --- |
| N | Optimal features subset |
| 5 | [22, 16, 26, 2, 30] |
| 10 | [16, 26, 2, 14, 7, 5, 30, 1, 12, 3] |
| 13 | [26, 27, 30, 2, 1, 3, 6, 5, 4, 21, 7, 8, 9] |
| 14 | [2, 14, 21, 27, 20, 9, 8, 1, 30, 5, 12, 3, 7, 6] |
| 15 | [7, 22, 29, 27, 9, 30, 3, 1, 5, 2, 12, 4, 6, 8, 21] |
| 16 | [30, 5, 1, 6, 22, 28, 7, 2, 3, 4, 8, 9, 10, 19, 11, 21] |
| 17 | [6, 20, 1, 22, 2, 30, 16, 3, 4, 8, 24, 26, 5, 19, 15, 7, 12] |
| 20 | [9, 18, 1, 13, 26, 30, 2, 3, 4, 23, 5, 6, 21, 27, 12, 8, 7, 10, 11, 28] |
| 25 | [6, 28, 1, 14, 7, 23, 19, 17, 3, 30, 12, 10, 24, 5, 8, 29, 18, 4, 20, 27, 9, 25, 22, 21, 15] |
| 30 | [13, 5, 16, 11, 29, 20, 8, 15, 0, 6, 4, 30, 27, 12, 14, 19, 24, 25, 7, 23, 17, 2, 28, 3, 22, 1, 26, 21, 9, 18] |



The parameters of Genetic Algorithm are:

* Number of Generations:15
* Population Size: 40
* Crossover Probability: 0.5
* Mutation probability: 0.2

It can be seen that for N = 14 the SVM achieved best classification accuracy. The corresponding 14 features were then used for training 8 different ANN models which have different number of hidden layers and nodes in them. The classification accuracy obtained on test samples by the ANN models are shown in graph below.



The hyperparameters of the models used are:

* Optimizer: Adam
* Learning Rate: 0.001
* Dropout Probability: 0.1
* Number of Epochs: 20

The model 6 with the following description achieves the highest accuracy of 98.31% .

Model 6 –

* Input Layer – 14 nodes
* Hidden Layer 1 – 128 nodes
* Hidden Layer 2 – 256 nodes
* Hidden Layer 3 – 256 nodes
* Output Layer – 2 nodes

1. **Conclusion**

In this paper we designed a model for classification of Higgs boson decay against the background noise. The Genetic Algorithm (in which SVM was used to calculate classification accuracy for fitness of population) was used to find the optimal feature subset with reduced number of features to improve overall accuracy.

The optimal feature subset was then used to train various ANN models and we obtained 98.31% as the best classification accuracy against 93.9% when all the features were used for modelling.

**References**

1. G. Aad et al., “Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC”, Phys.Lett., vol. B716, pp. 1–29, 2012.
2. S. Chatrchyan et al., “Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC”, Phys.Lett., vol. B716, pp. 30–61, 2012.
3. The ATLAS Collaboration, “Evidence for higgs boson decays to tau+tau- final state with the atlas detector”, Tech. Rep. ATLAS-CONF-2013-108, November 2013, http://cds.cern.ch/record/1632191
4. Higgs Boson technical background : <https://higgsml.lal.in2p3.fr/files/2014/04/documentation_v1.8.pdf>
5. Kaggle: Higgs Boson Machine Learning Dataset:https://www.kaggle.com/c/higgs-boson
6. Yang J. and Honavar, V., ‘Feature Subset Selection using a Genetic Algorithm’, *IEEE Intelligent Systems*, March-April 1998, pp 44-49.
7. Huang, C.-L. and Wang, C.-J.,(2006),’A GA-based feature selection and parameters optimization for support vector machines’, *Expert Systems with applications*, Vol.-31,(2006), pp -231-240.
8. Pedroso, J. P., & Murata, N. (2000). ‘Optimisation on support vector machines.’ *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN'00)*, July 24-27, 2000, Como, Italy,Volume 6 (pp 399-404).
9. Begg, R. K., Palaniswami, M., & Owen, B. (May 2005). ‘Support Vector Machines for Automated Gait Classification’, *IEEE Transaction on Biomedical Engineering*, vol 52, No. 5, (pp. 828-838).
10. Hsu, C.W. and. Lin, C.J.,(2002), ‘A comparison of methods for multi-class support vector machines’, *IEEE Transactions on Neural Networks*, 2002, Vol. 13, No. 2., pp 415-425.
11. Dawai Dai, Weimin Tan and Hong Zhan, *Feedforward Artificial Neural Network Model From the Perspective of Network Flow*: https://arxiv.org/ftp/arxiv/papers/1704/1704.08068.pdf
12. Chigozie Enyinna Nwankpa, Winifred Ijomah, Anthony Gachagan, and Stephen Marshall, ‘*Activation functions: Comparision of trends in practice and research for deep learning*: https://arxiv.org/pdf/1811.03378.pdf
13. Learning representations by back-propagating errors : <https://www.nature.com/articles/323533a0>
14. Whitley, D. (1994), ‘A Genetic Algorithm Tutorial’. *Journal of Statistics and Computing*. Vol 4, pp: 65-85.