Higgs Bosons Detection With Deep Learning Techniques

By

Jia Yoong Chong

100120425

Final Project PHY 407

Dec 20 , 2017

Physics background

Large Hadron Collider collides bunches of protons every 50 nanoseconds within each of its four experiments. Each bunch crossing results in a random number of proton-proton collisions (with a Poisson expectation between 10 and 35, depending on the LHC conditions) called events.

Two colliding protons produce a small "explosion" in which part of the kinetic energy of the protons is converted into new particles. Most of the resulting particles are very unstable and decay quickly into a cascade of lighter particles. The ATLAS detector (Figure 1.0) measures three properties of these surviving particles (the so-called final state): the type of the particle (electron, photon, muon, etc.), its energy, and the 3D direction of the particle. From these quantities, the properties of the decayed parent particle is inferred, and the inference chain is continued until reaching the heaviest primary particles.

An online trigger system discards the vast majority of bunch collisions, which contain uninteresting events. The trigger is a three-stage cascade classifier which decreases the event rate from 20000000 to about 400 per second. The selected 400 events are saved on disk, producing about one billion events and three petabytes of raw data per year.

Each event contains about ten particles of interest in the final state, which are re- constructed from hundreds of low-level signals. The different types of particles or pseudo particles of interest for the Challenge are electrons, muons, hadronic taus, jets, and missing transverse energy, which are explained below. Electrons, muons, and taus are the three leptons7 from the standard model. Electrons and muons live long enough to reach the detector, so their properties (energy and direction) can be measured directly. Taus, on the other hand, decay almost immediately after their creation into either an electron and two neutrinos, a muon and two neutrinos, or a collimated bunch of charged particles and a neutrino. The bunch of hadrons can be identified as a pseudo particle called the Hadronic tau. Jets are pseudo particles rather than real particles; they originate from a high energy quark or gluon, and they appear in the detector as a collimated energy deposit associated with charged tracks. The measured momenta of all the particles of the event is the primary features provided for this project. Refer to reference for all the features.

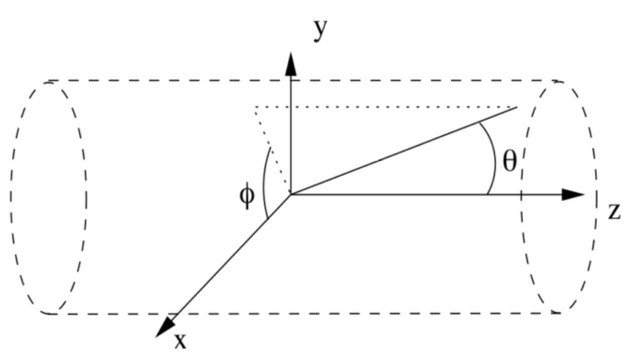

Figure 1.0 shows ATLAS detector and its reference frame

Computational background

Dropout neural network

- Neural network is an interconnected system of processing nodes inspired from biological neurons. The computational ability of the individual processing nodes is stored in interneuron strength, known as weights, obtained by learning process from a dataset.
- Dropout is a technique in neural networks to prevent the model from over-fitting. It can be accomplished by randomly dropping neural networks and their connections during the training process.

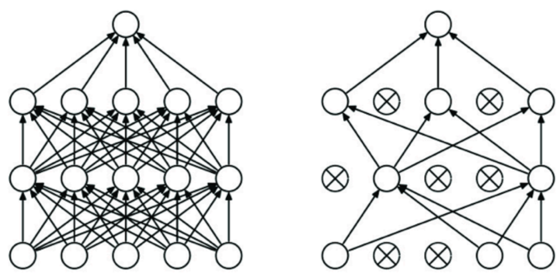

Figure 2.0 shows the standard neural network (left) and neurons being dropped out throughout training process (right).

Statistical Model

- The Approximate Median Significance (AMS) is an objective function used to determine a region in the feature space where one expects an enhanced number of signal events. This value will follow a Poisson distribution with a mean of s + b, where s and b are the mean numbers of events from signal and background processes, respectively. If n is found much greater than b, then the background-only hypothesis is rejected.

$$AMS = \sqrt{2((S + B + 10) \ln\left(1 + \frac{S}{B}\right) - S)} \qquad (?)$$

where S (signal) and B (background) are expected number of signal and background events selected by the classifier, respectively. Derivation of equation (2) can be found in reference. The AMS score is then computed for both training set and validation set as a metric to evaluate performance of the model.

Deep learning Terminologies

- **Training set** is a data set to build a model in order to determine its parameters. It can be alternatively thought as seen data by the model. Labelled as training.csv

- **Test set** is a data set to measure the performance of the model by holding the parameters constant. It can be alternatively thought as unseen data. Labelled as test.csv

- **Validation set** is a data set used to tune the model.

- **Over-fitting** refers to a model that models the training data too well that corresponds too closely to a particular set of data. This causes the model to predict future observations poorly given another training set.

- **Cross validation** is a process to prevent over-fitting of the model. Usually data is randomly divided k-fold into equal-sized subset. The model is then trained on k -1 folds, in which 1 fold is left for testing. The process is repeated for shuffling the training set and test set. Average performance of k test sets is then computed.

- **Activation function** of a neuron defines the output of that neuron given a set of input vectors. It is defined by
$$f(x) = \max(0, x)$$

- **Soft-max function** takes a set of input vectors and transforms it into vectors of real number in range (0,1) which eventually sums up to 1. It is often used in the final layer of neural-network based classifier. Tensorflow's tf.nn.softmax() computes the soft-max activation.

- **Cross entropy** is a function that defines the distance between the output model and original model. It is alternatively understood as mean squared error.
$$H = -\sum p(x)\log(p(x))$$

- **Cost function** is a function intended to minimize, in this case, soft-max cross entropy.

Dataset description:

The two CSV files can be downloaded from from Kaggle Machine Learning challenge website. The link provided is https://www.kaggle.com/c/higgs-boson/data

Simulated events using the official ATLAS full detector simulator is provided instead of the real data collected from the particle detector. The simulator yields simulated events with properties that mimic the statistical properties of real events of the signal type as well as several important backgrounds.

Events generated in the collider are preprocessed and represented as a feature vector. The goal is to classify events as signal (that is, an event of interest, in our case a H to tau tau decay) or background (an event produced by already known processes). More precisely, the classifier is used as a selection method, which defines a (not necessarily connected) signal-rich region in the feature space.

Signal sample contains events in which Higgs bosons (fixed mass 125GeV) were produced. Background sample contains events corresponding to other known process which can produce events with at least one electron or muon and a hadronic tau, mimicking the signal. In this project, only three background processes were retained.

1.) Decay of the Z boson (91.2 GeV) in two taus. This decay produces events with a topology very similar to that produced by decay of a Higgs .

2.) Events with a pair of top quarks, which can have lepton and Hadronic tau among the decay products.

3.) Third set involves decay of W boson, where one electron or muon and a Hadronic tau can appear simultaneously only through imperfections of of particle identification procedure.

In the data set. there are event ID, 30 vectors of input features representing the parameters and measurements taken from collider, and {s.b}/signal background label and weighting factor for each event ID.

The sum of signal and background weighting factors are expected occurrences for each event. The values can be computed by taking conditional densities of the input features and dividing it by instrumental densities.

Objective

This project is categorized as supervised learning problems where the data given is a set of training examples with the associated "correct answers" – test/validation set. The implemented deep learning model learns to predict the correct answer based on the training set.

In this project, this would be learning to predict whether an event ID is either a signal or background noise given a huge dataset, each of which is labeled as "signal" or "background".

Objective of the project is to train a binary classifier based on the training set in order to maximize AMS on a held-out test data set using dropout neural networks.

Additional libraries such as Tensorflow and Scikit-learn are used.

Algorithm

- Import useful libraries from matplotlib, tensorflow, scikit-learn and numpy
- Loading data from CSV files into numpy array.
- Normalize the dataset such that its mean is zero and standard deviation .
- Perform cross validation on data set 2-fold stratified cross validation with random shuffling.
- Define dropout neural network function with three hidden layers.
- Set up 26 x 600 x 600 x 600 x 2 dropout neural network.
- Define cost function, training optimizer and probability prediction optimizer.
- Implement the deep learning model and make prediction on the outputs probability.
- Train the classifier to distinguish signal and background by feeding training data.
- Compute the AMS score for training set and validation set.
- Plot the histogram of signal and background distribution

Results

Plots

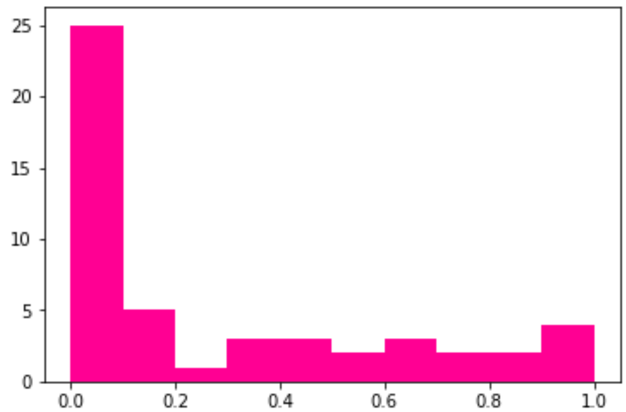

Figure 3.0 shows the Higgs Boson Signal Distribution with x-axis as probability output and y-axis as signal count (in thousandth).

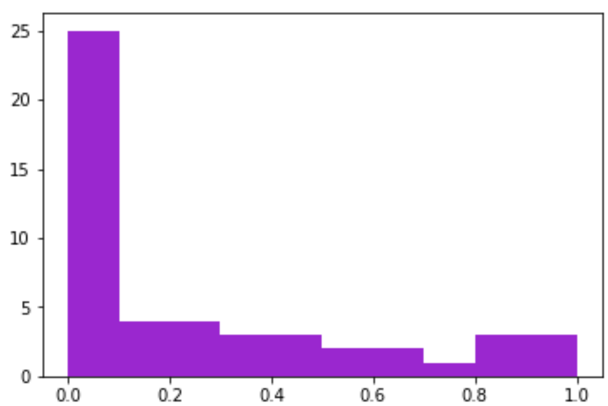

Figure 4.0 shows the Higgs Boson Background Distribution with x-axis as probability output and y-axis as background count in (thousandth).

Commentaries

Figure 3.0 & 4.0 shows that approximately 25000 event IDs that have low (close to zero) probabilities of being signal and background noise respectively.

It is important to notice there is a probability ranging from 0.8 to 1.0 with more background noise than signals. Background noises are more evident than signal in the low probabilities region (0 to 0.4),

The accuracy of the deep learning model in classifying signal and background noise is high. This can be seen through the evident differences between signal distribution and background distribution. In this case, it is important to note that the occurrence of high probability signal events is greatly overshadowed by high probability background events.

For feature scaling purposes, the input vectors were normalized to have zero mean and unit variance, equivalently standard deviation of 1. This deep learning model has three hidden layers each with 600 neurons and two softmax neurons outputs. Rectified Linear Unit (ReLUs) activation function is employed for much faster training due to the large neural network.

Input is classified as signal if output of softmax signal neuron was greater than 0.55. If the output probability of event is greater than 0.8, event is then classified as signal. In setting up the dropout neural network, the probability of neurons in hidden layer being dropped is set to 0.5.

The final AMS score for training set and validation set is 2.314 and 2.557.

Discussion

Performance & discussion

The time taken to run the whole program takes around 10 hours. Hence due to time constraint, only one tenth of the dataset is used.

If the whole data set has been used instead, the model could learn better and its expected final AMS score would be well above 3. The learning rate of the model is adjusted suitably to 0.02 in order to prevent poor learning and over-training.

Dimensionality of the input vectors could be reduced to further improve performance and significantly reduce the time taken to train the model. Such reduction requires an application of Principal Component Analysis.

In this project, the cross validation is only made 5-fold. This means data set is divided into 5 equal-sized subsets and is trained on 4-fold. To avoid the possibility of over-fitting, k could be made 10.

The deep learning model employed in this project consist of three hidden layers with 600 neurons and two output layers of dimension 26 x 600 x 600 x 600 x 2. Adding extra number of neurons and hidden layers is likely to allow the deep learning to pick up more patterns from the input data, resulting in better AMS score.

The physical significance of input measurements would allow feature extractions, constructions and selections. For example, 4 input vectors were removed manually to prevent over-training. However, if certain input vectors could be necessarily and selectively removed, this would allow us to create a simpler model from less complicated features.

Conclusion

This project demonstrates the application of dropout neural networks to statistically model the signal and background events generated by ATLAS simulator. Deep learning shows that there are more background noises than signals detected. For time consideration, training set was made to one-tenth the size of original dataset. Hence it is still possible that the model is over-trained. The model computes an average AMS score of 2.2. (Winner of Higgs Boson machine learning challenge has AMS score of 3.8).

Reference

[1] Claire Adam-Bourdarios, Glen Cowan, Cecile Germain, Isabelle Guyon, Balázs, Kégl, Facid Rousseau. Learning to discover: the Higgs boson machine learning challenge, 21 July 2014.

[2] Nitish Srivistva, Geoffrey Hinton, Alex Krizhevsky, Illya Sutskever Ruslan Salakhutdinov . Dropout: A Simple Way to Prevent Neural Networks from Overfitting. Journal of Machine Learning Research 2014, pages 1929-1958.

[3] Claire Adam-Bourdarios, Glen Cowan, Cecile Germain, Isabelle Guyon, Balázs, Kégl, Facid Rousseau. The Higgs boson machine learning challenge. 2015. JMLR: Workshop and Conference proceedings 42:19-55, 2015.

[4] Aurélion Geron. Hands-On Machine Learning with Scikit-Learn and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent System. O'Reilly Media. April 9, 2017.