Application of Machine Learning Techniques to Separate Resonance Particles From Background Processes

9th Semester Presentation

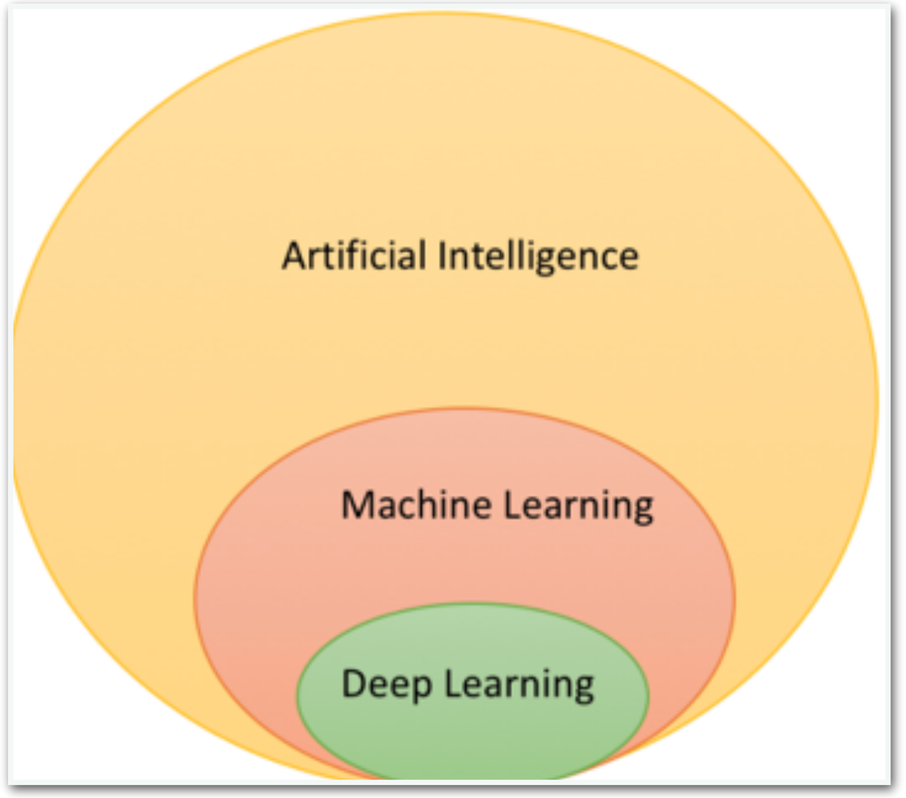
Supervised by: - Dr. Prolay Kr. Mal

23/11/21

Outline

- What is Machine Learning?
- What is DNN?
- How DNN work?
- Variables used in training and testing
- Correlation Plots
- Results and outputs from the DNN
- Conclusion

Machine Learning



Machine learning(ML) is part of artificial intelligence.

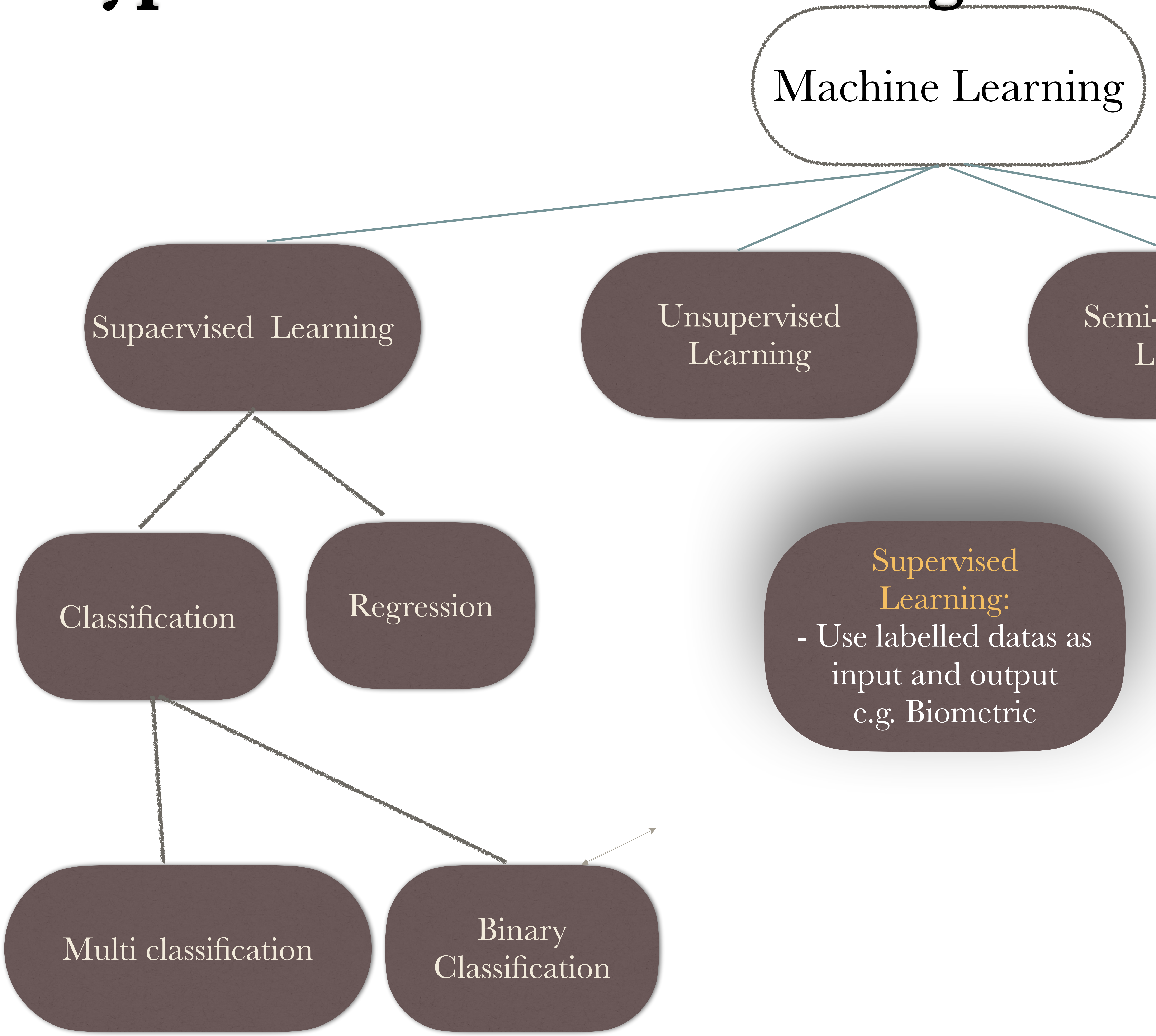
ML allows softwares to become accurate without being explicitly programmed

Example of ML:-

1. siri, Alexa: virtual personal assistance
2. All social media(use GNN)
3. Email classification as spam or not spam

source: Edureka

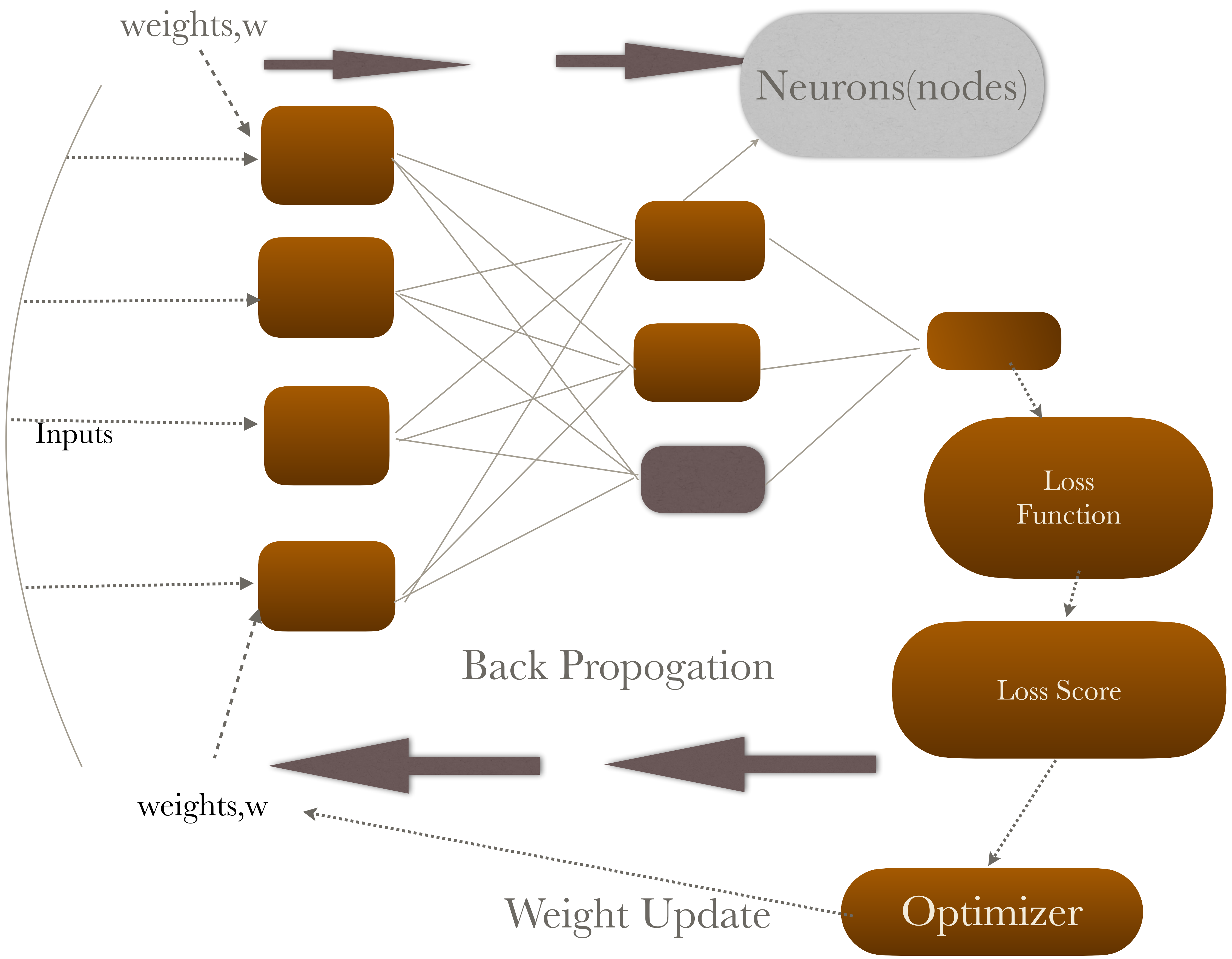
Types of machine Learning

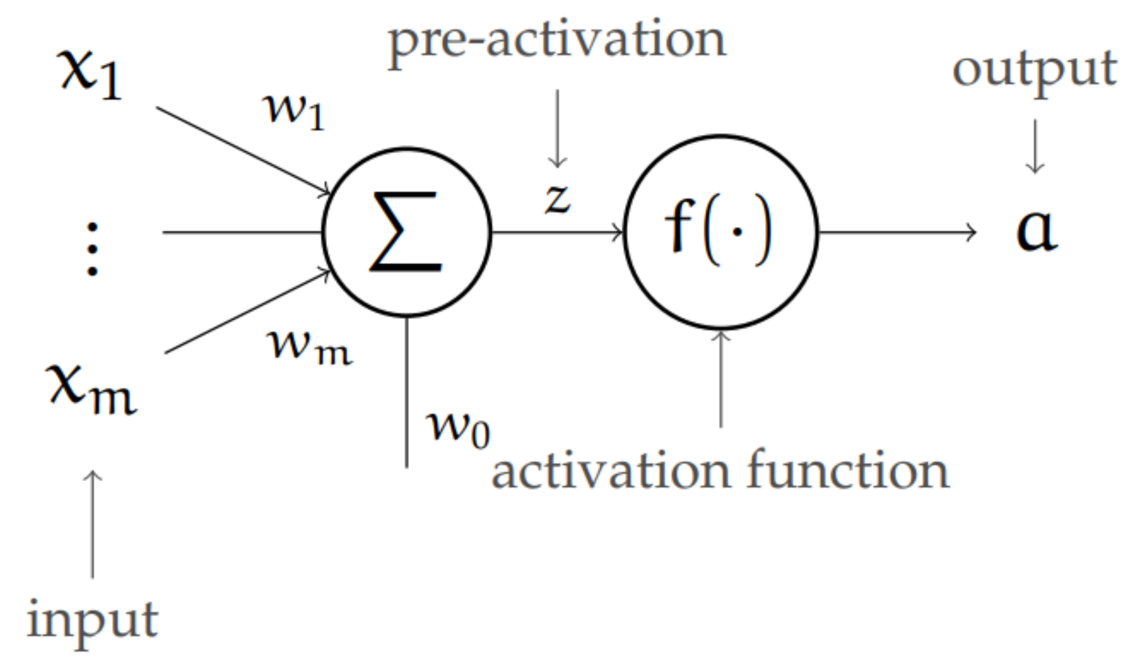


Every ML algorithm divided into:
Training and Testing phases

Deep Neural Networks

It can be conceived as structure of human brain. Where neurons can be compared to nodes and synapses.





$$a = f(z) = f\left(\sum_{j=1}^{j=m} x_j w_j + w_0\right) = f(w^T x + w_0)$$

The output depends on the type of activation function used.

Low level framework High level framework

Keras

Single layer network is also known as perceptron.

- Deep means hidden layer >2.

DNN: Few activation function

Three types of Activation Functions:-

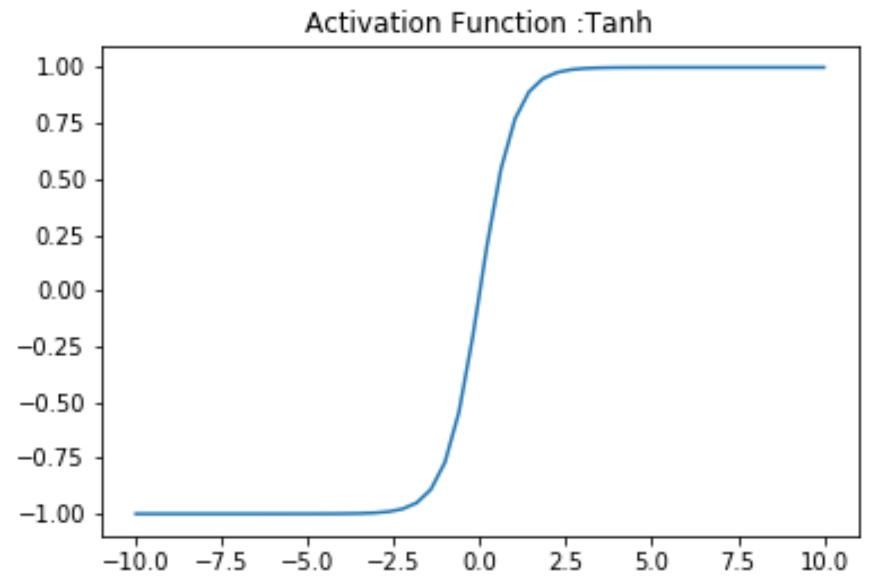
Binary activation Function

$$f(x) = \begin{cases} 0 for x < 0 \\ 1 for x \geq 0 \end{cases}$$

Linear Activation Function

$$f(x) = x$$

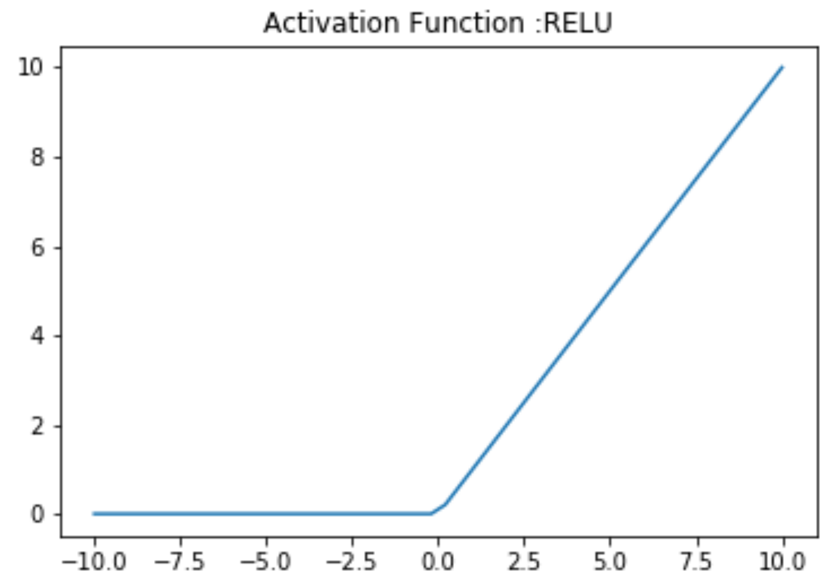
Non-linear Activation functions



tanh

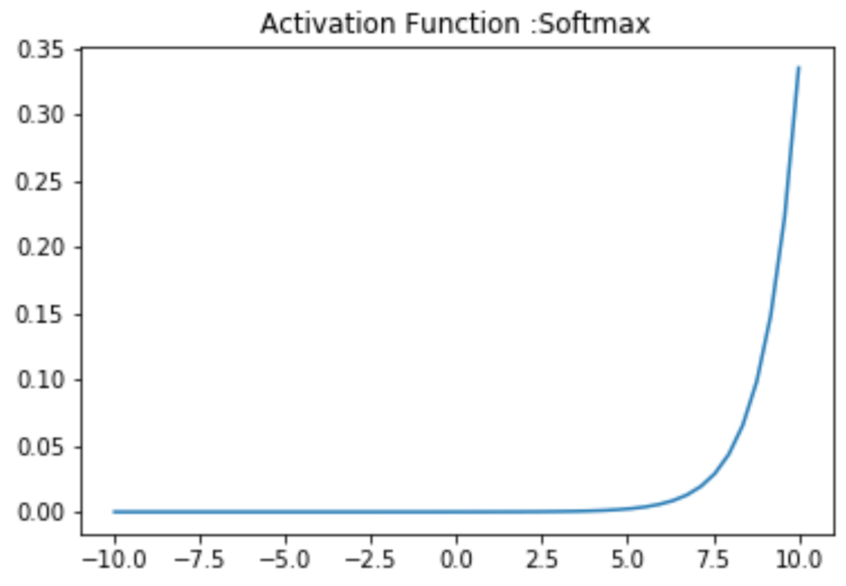
$$g(z) = \frac{1 - e^{-2z}}{1 + e^{-2z}}$$

$$g'(z) = 1 - g(z)^2$$



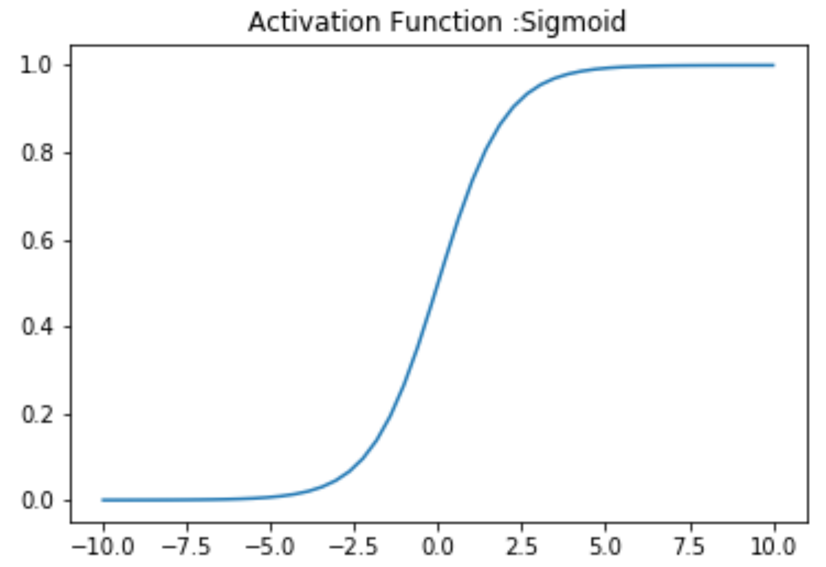
ReLU

$$g(z) = max(0,z)$$



Softmax

$$g(z) = \frac{exp(z_i)}{\sum_j exp(z_j)}$$



Sigmoid

$$g(z) = \frac{1}{1 + e^{-z}}$$

Loss Function

- Loss is calculated for the previously used function as:

$$\mathcal{L}(\underbrace{f(x^{(i)}; \boldsymbol{W})}_{\text{Predicted}}, \underbrace{y^{(i)}}_{\text{Actual}})$$

- For any model, the main goal is to minimize the loss
- Also known as cost function, objective function.

$$L(f(x^i; W), y) = Loss(A^L, y)$$

$$A^L = f^L(z^L) and z^L = W^{L^T} A^{L-1}$$

Back Propogation

- To minimize the loss function, we apply chain rule w.r.t the weights in the final layer:

$$\frac{\partial loss}{\partial W^L} = \underbrace{\frac{\partial loss}{\partial A^L}}_{\text{depends on loss function}} \cdot \underbrace{\frac{\partial A^L}{\partial Z^L}}_{f^{L\prime}} \cdot \underbrace{\frac{\partial Z^L}{\partial W^L}}_{A^{L-1}}$$

- If we continue to apply chain rule we can write the loss as,

$$\frac{\partial loss}{\partial Z^l} = \frac{\partial A^l}{\partial Z^l} \cdot W^{l+1} \cdot \frac{\partial A^{l+1}}{\partial Z^{l+1}} \cdot \ldots W^{L-1} \cdot \frac{\partial A^{L-1}}{\partial Z^{L-1}} \cdot W^L \cdot \frac{\partial A^L}{\partial Z^L} \cdot \frac{\partial loss}{A^L}$$

Regulization

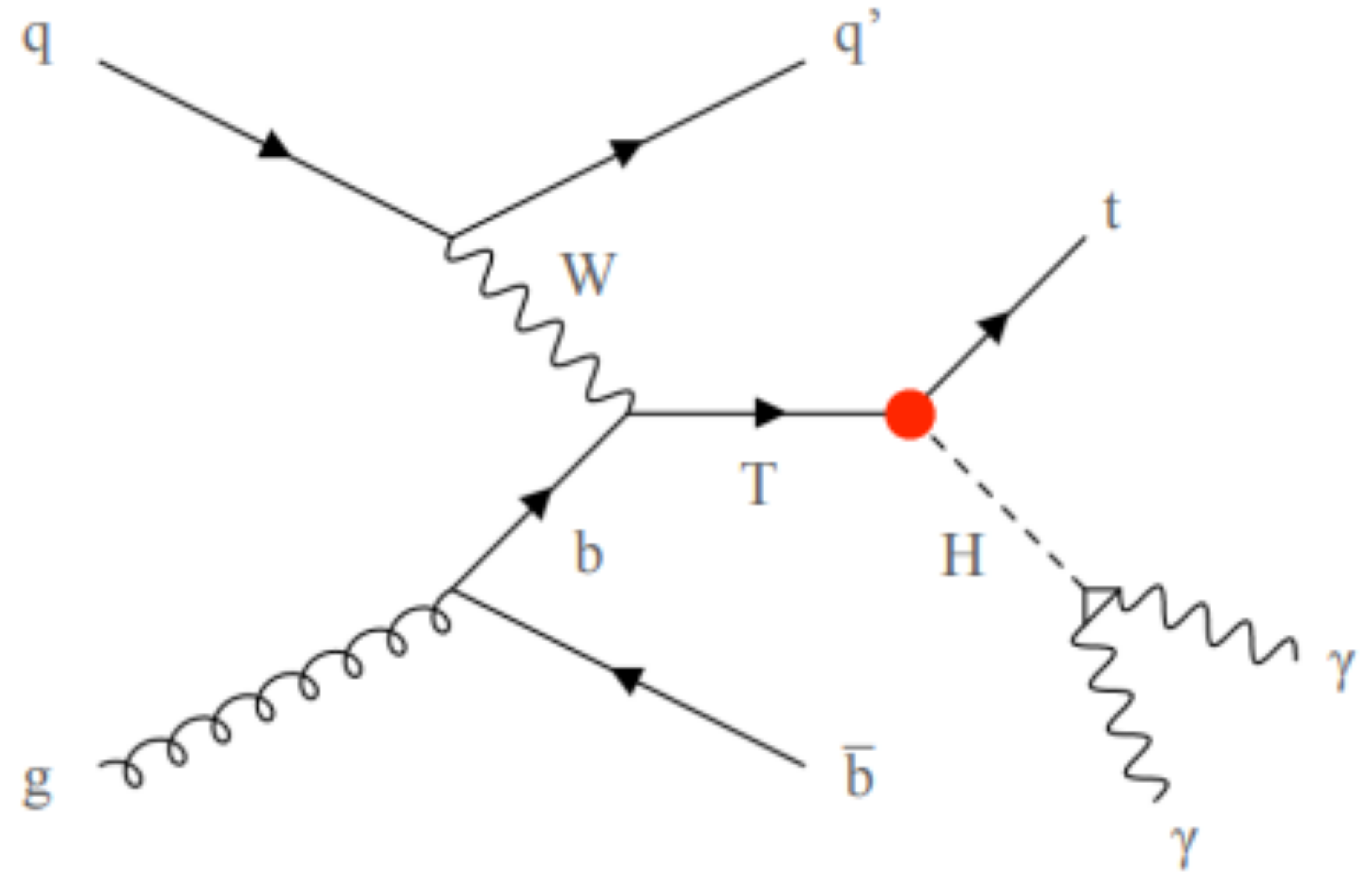
Resonance Particle(signal)

Resonance Particle

- Particles with very short life time.
- Lifetime is in order of 10^(-23).
-

- Signal corresponds to the features, processes, or events, in which we are interested.
- The resonance particles T' is used as Signal.
- T' is hypothetical spin-1/2 coloured particles with electric charge of +2e/3.
- T' is a vector like quark(VLQs) with two production modes.

1. Pair production through strong interaction
2. Single production through electroweak interaction



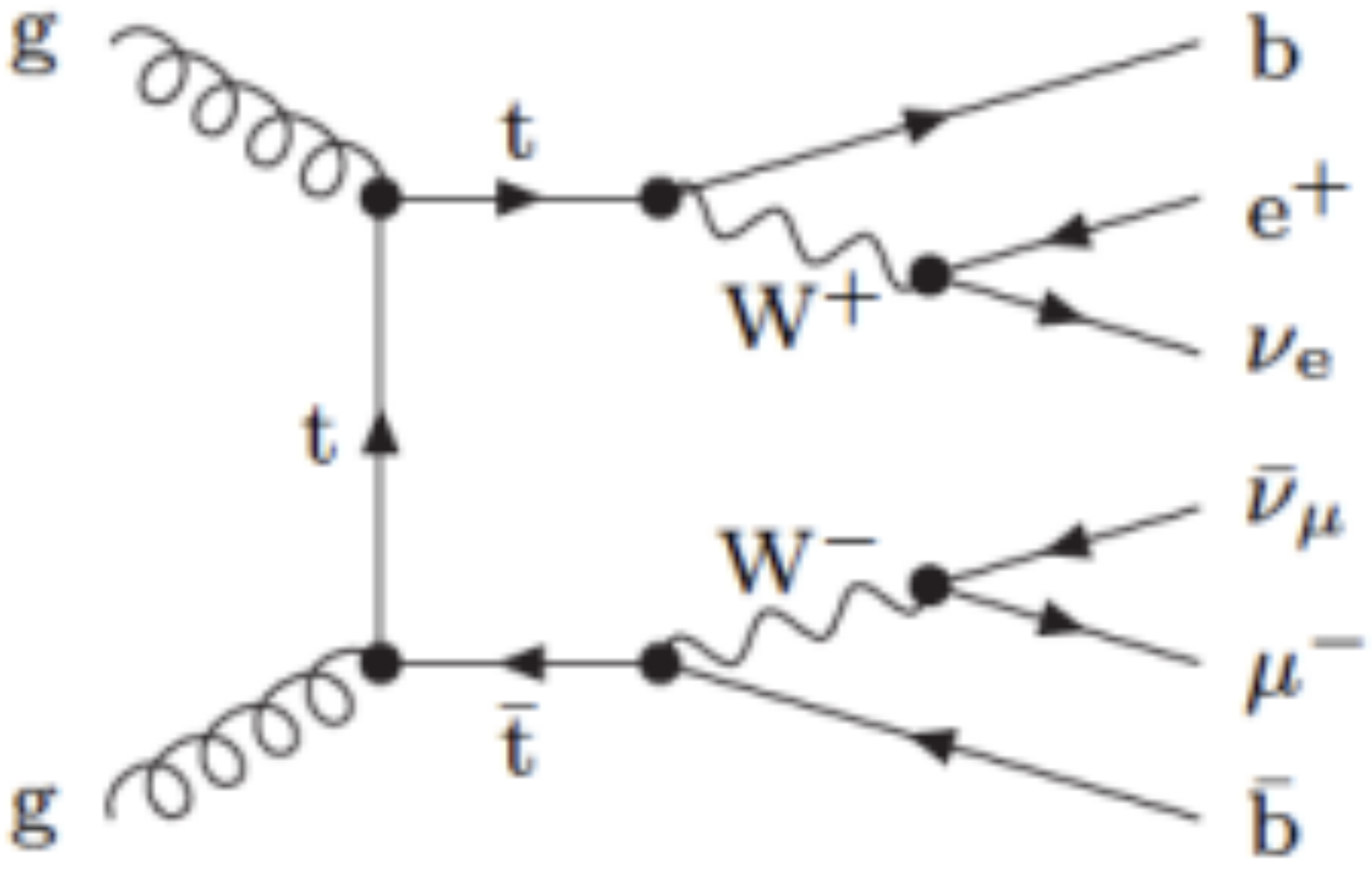
Tprime

T' quarks could couple to bW, tZ, or tH, resulting in corresponfing T' quark decays.

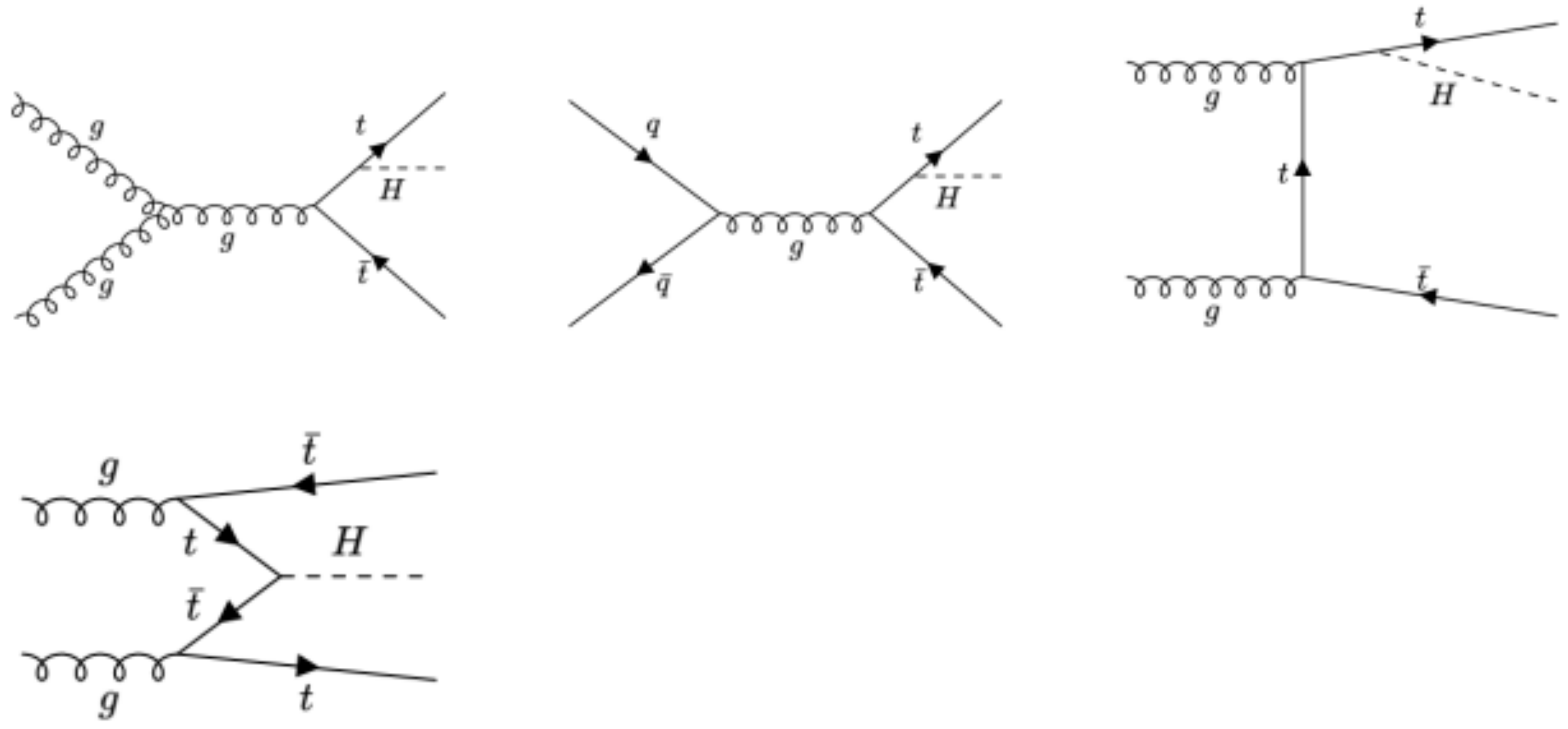
Backgrounds

- These backgrounds are the events that may seem like signal but we are not interested in.
- T', ttH, and tHq have been used as the backgrounds.

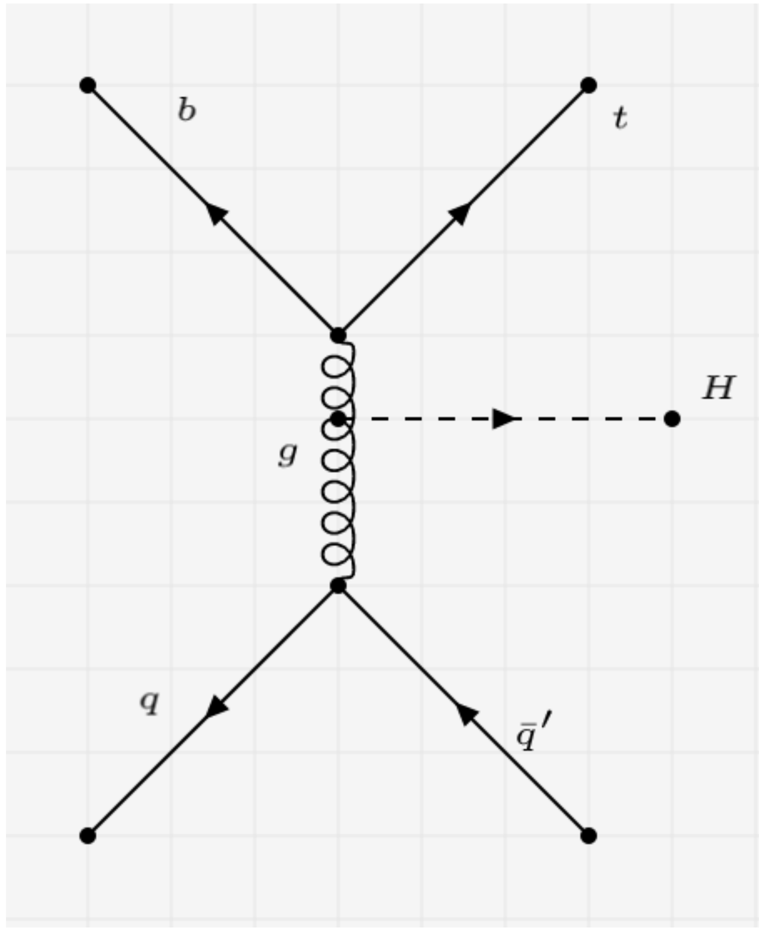
ttgg



ttH



tHq



Input Data Variables used for model training

There were total 29 input variables were used from data files:

- $dipho_{P_T}$
- $dipho_{\phi}$
- $dipho_{\eta}$
- $dipho_{lead\eta}$
- $dipho_{lead\phi}$
- $dipho_{sublead\eta}$
- $bjet1_{\eta}$
- $bjet2_{\eta}$
- $bjet2_{\phi}$
- $bjet3_{\phi}$

- $dipho_{e}$
- $dipho_{mass}$
- $bjet1_{\eta}$
- $bjest2_{\eta}$
- $jet1_{P_T}$
- $Jet2_{P_T}$
- n_{jets}
- $jet1_{e}$
- $jet2_{e}$
- $jet3_{e}$

How to Read Data from Root File in python?

ROOT File

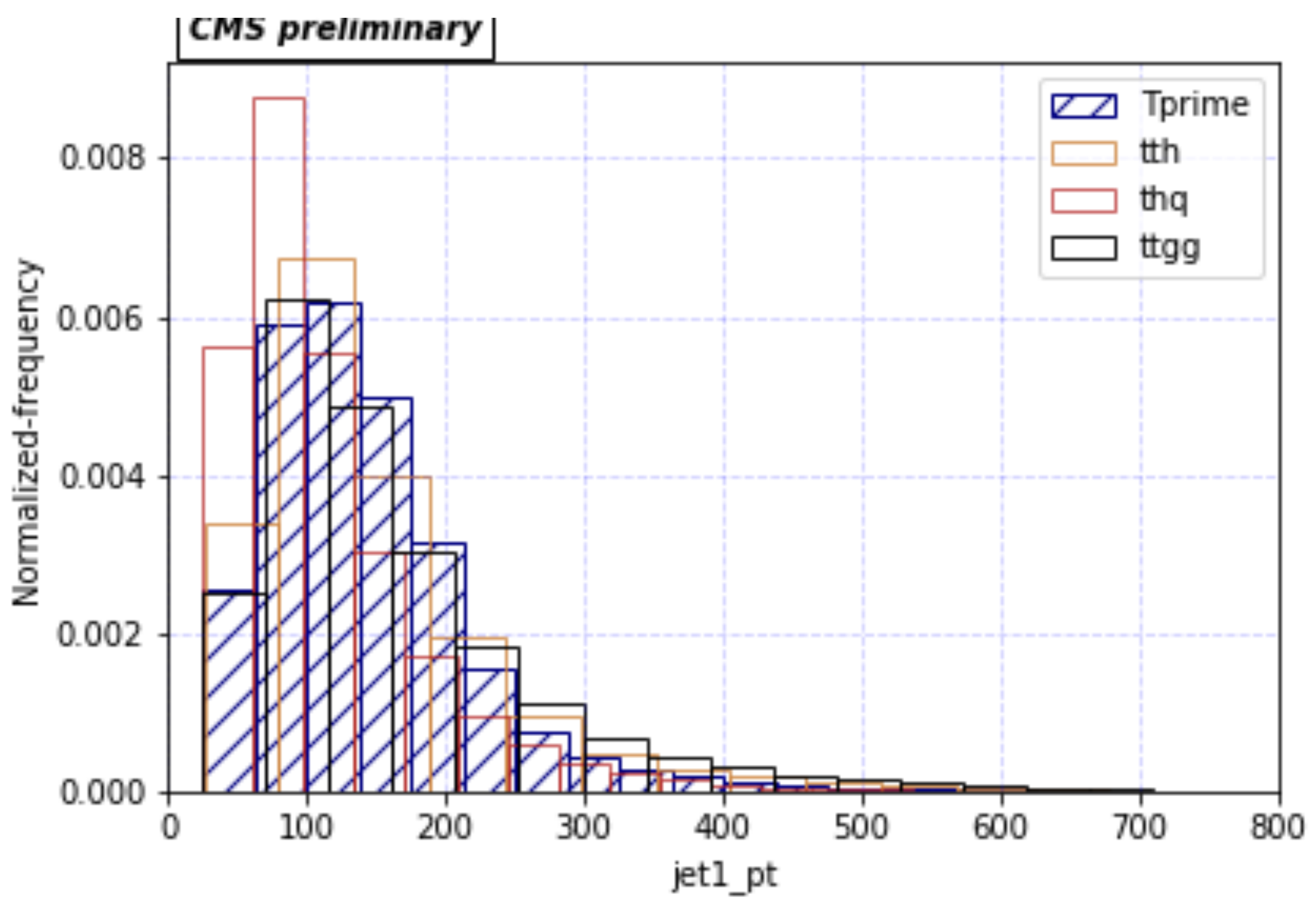
trees

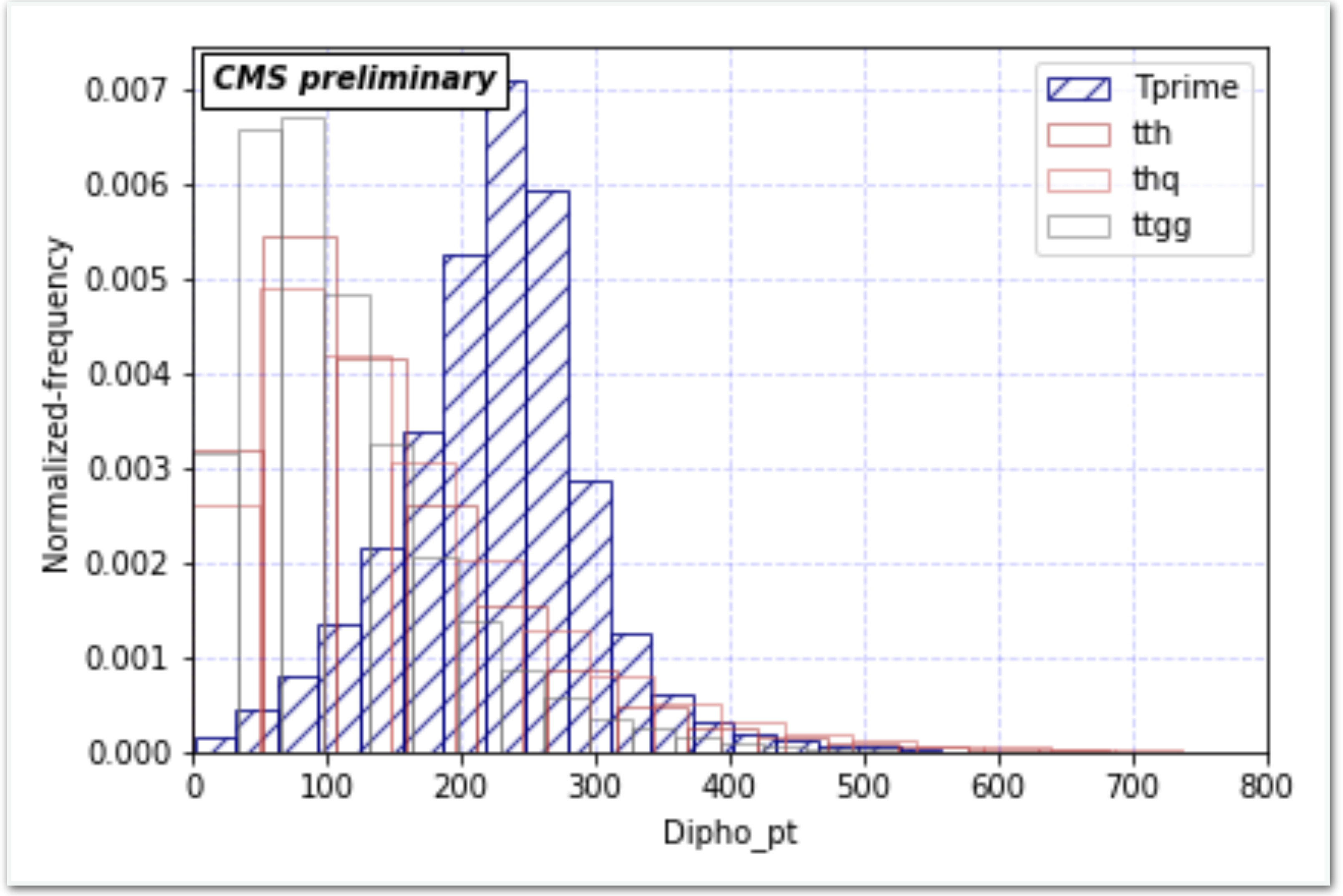
events

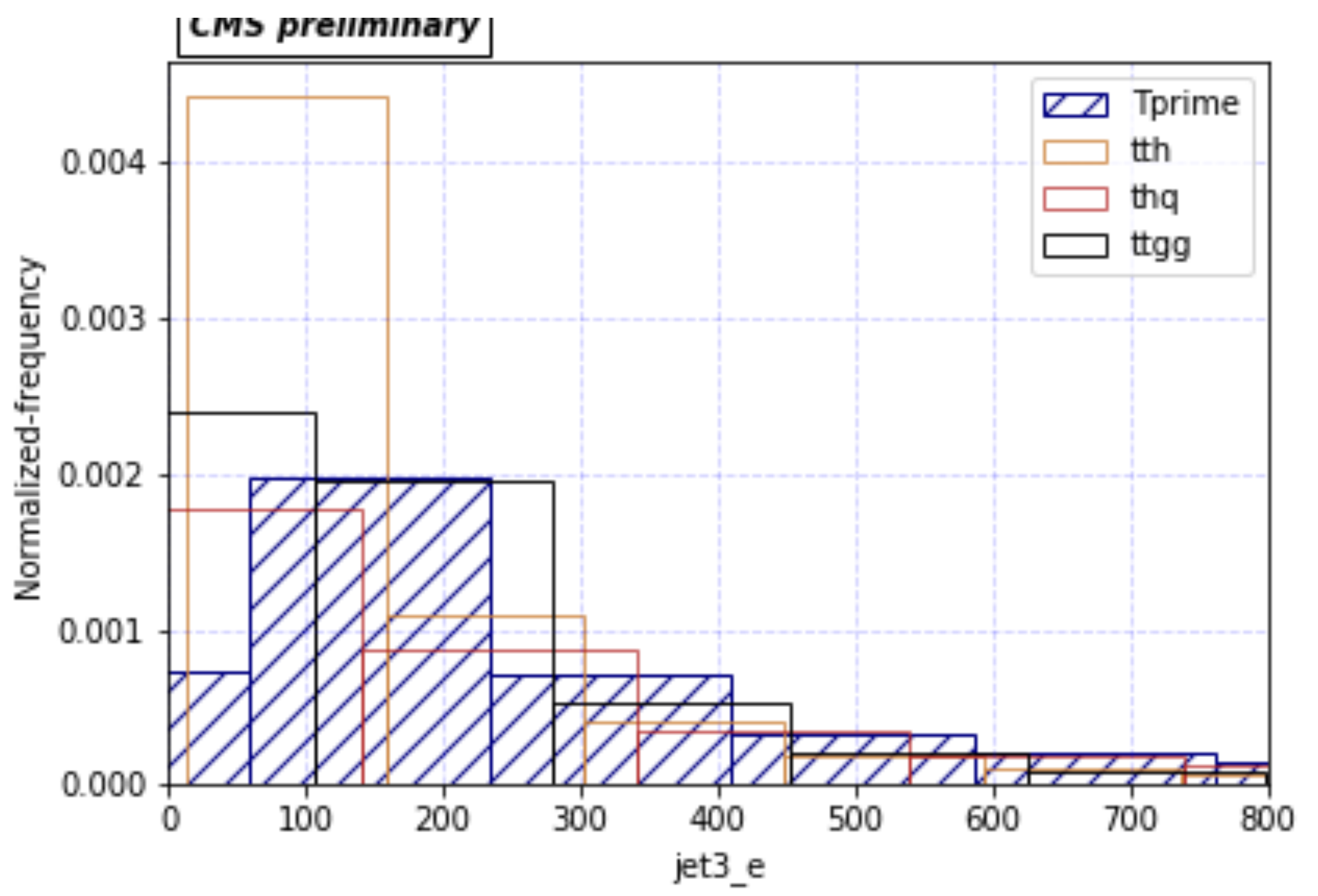
Variables

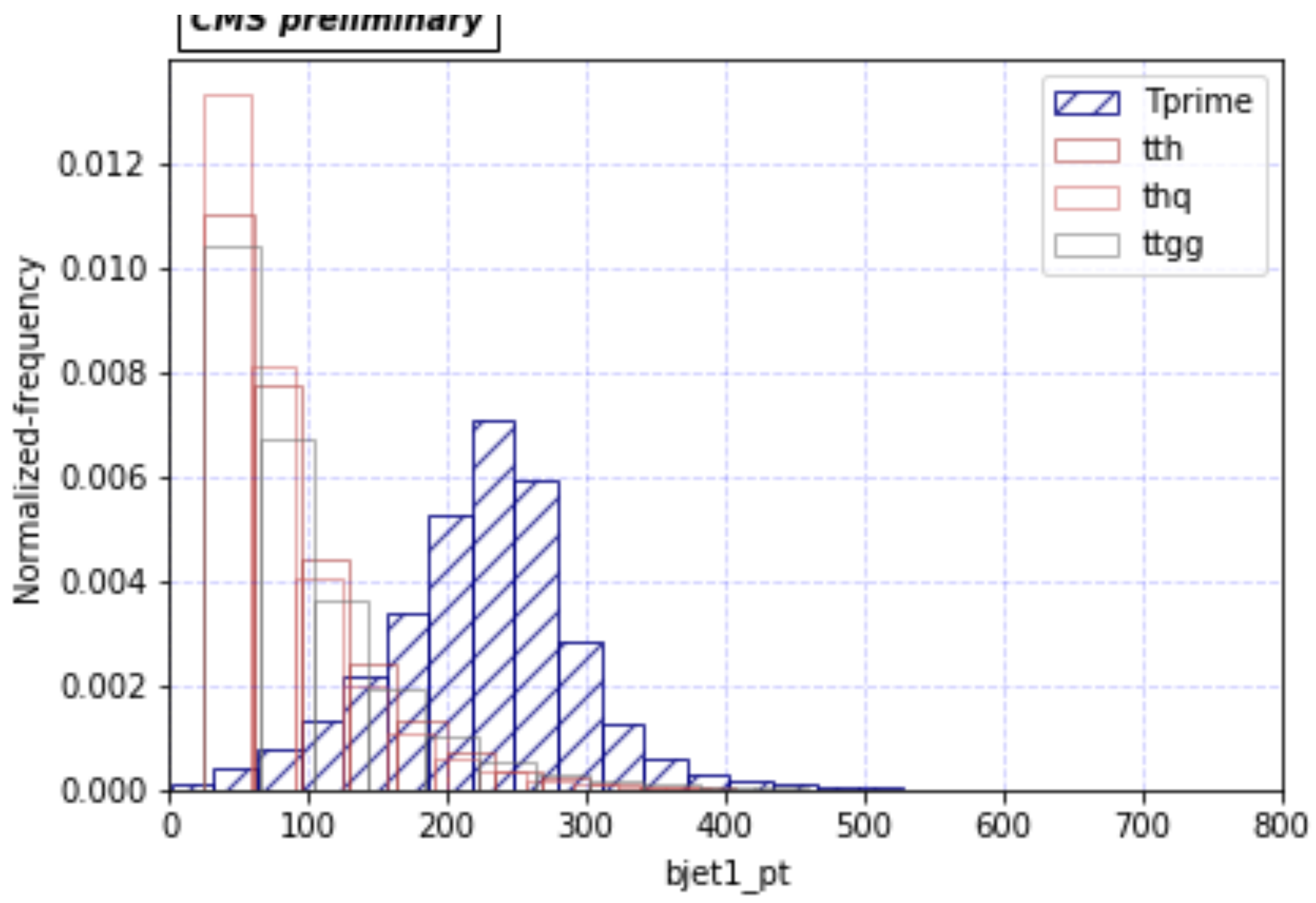
Use of root2numpy

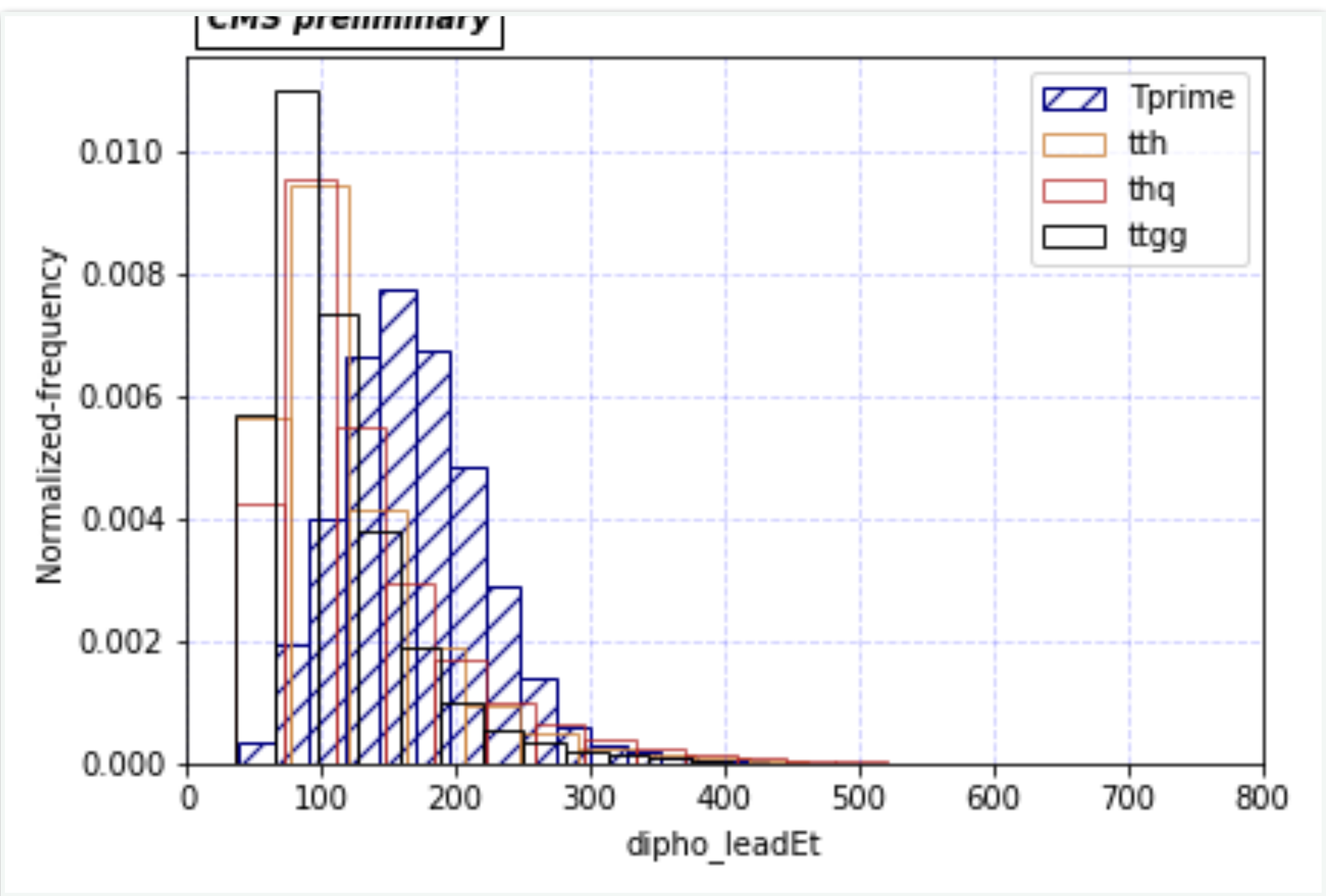
plot of input variables

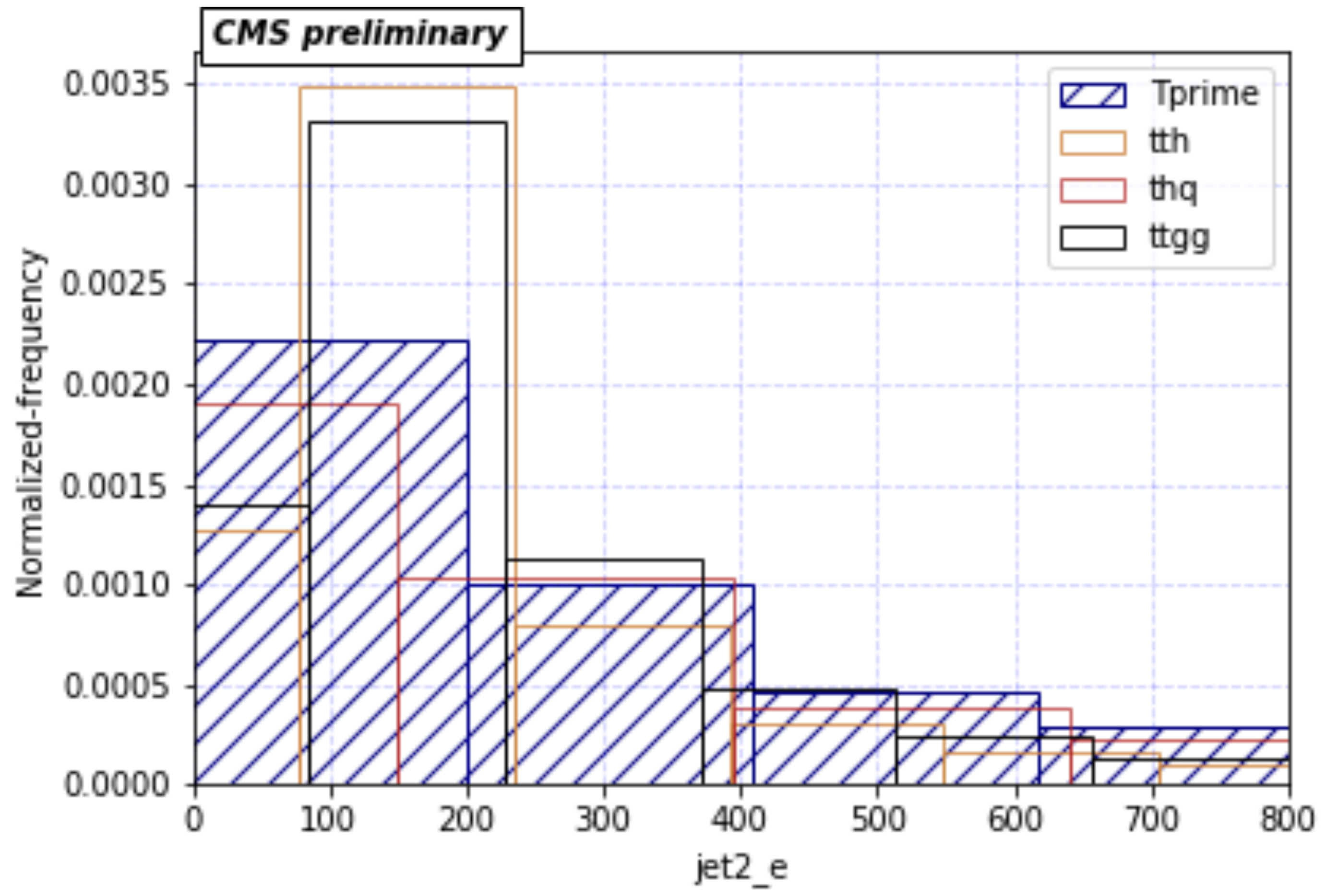




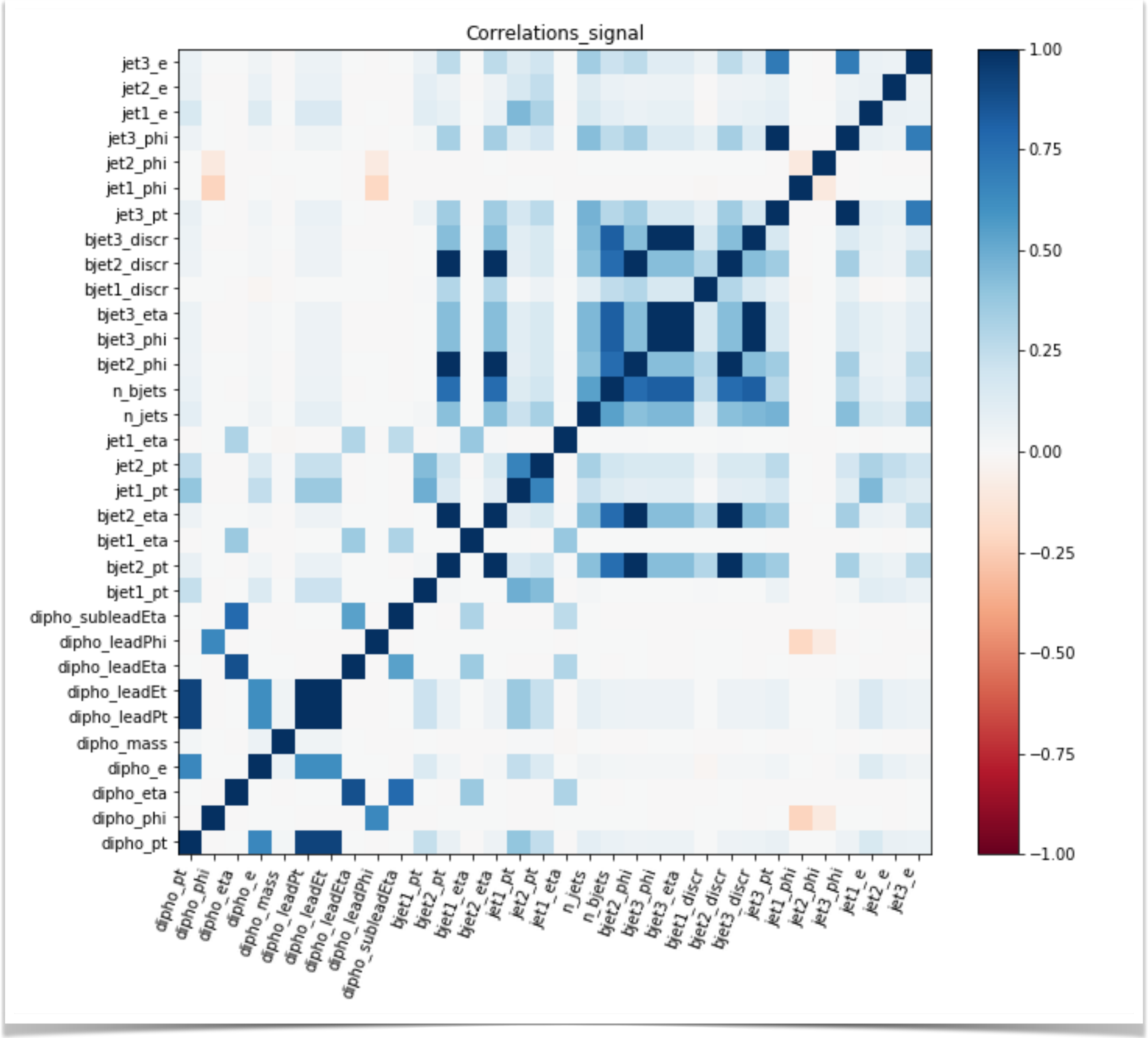








Correlation Plot



correlation coefficient formula

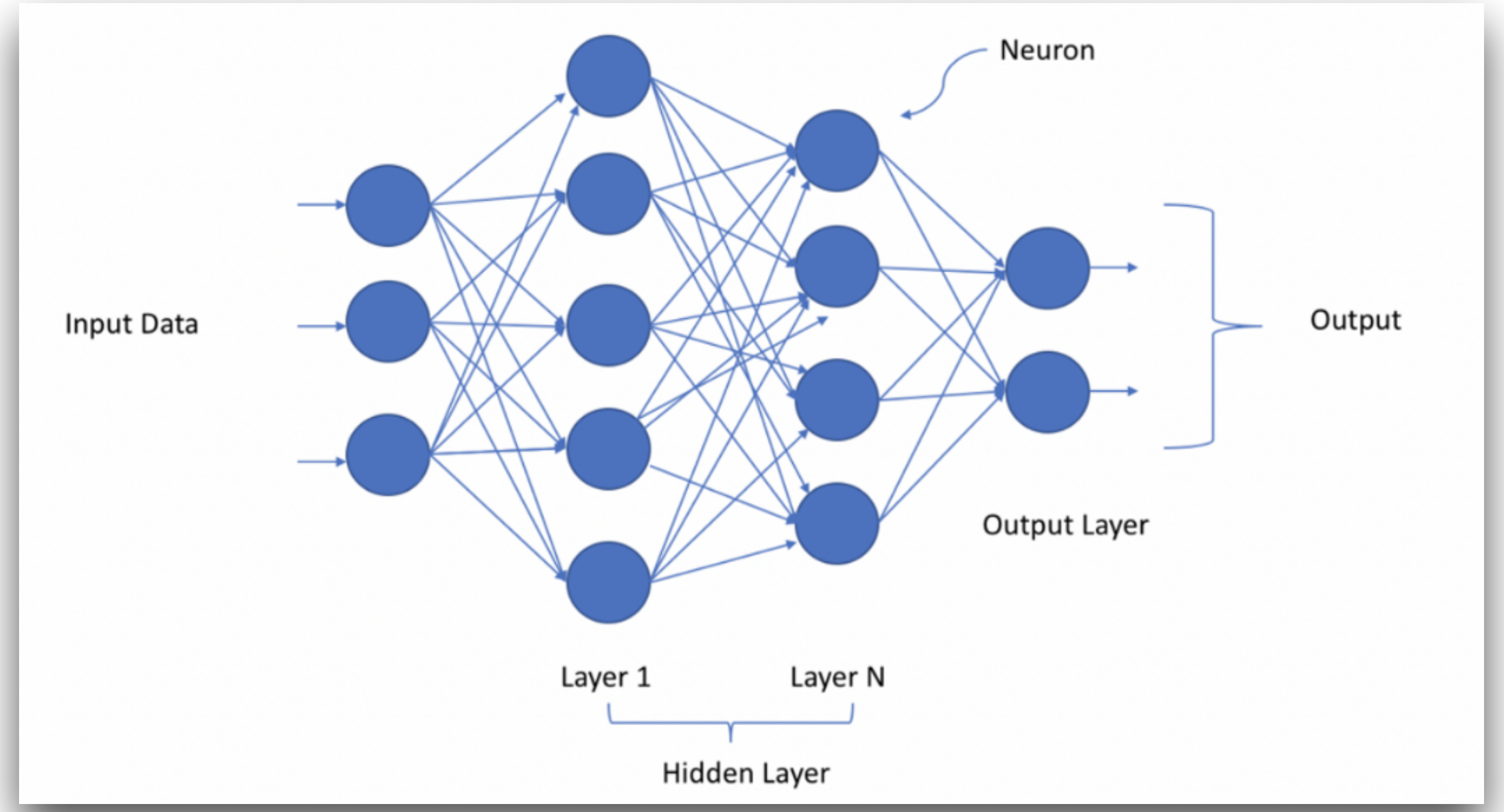
$$r = \frac{n(\sum xy) - \sum x \sum y}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}}$$

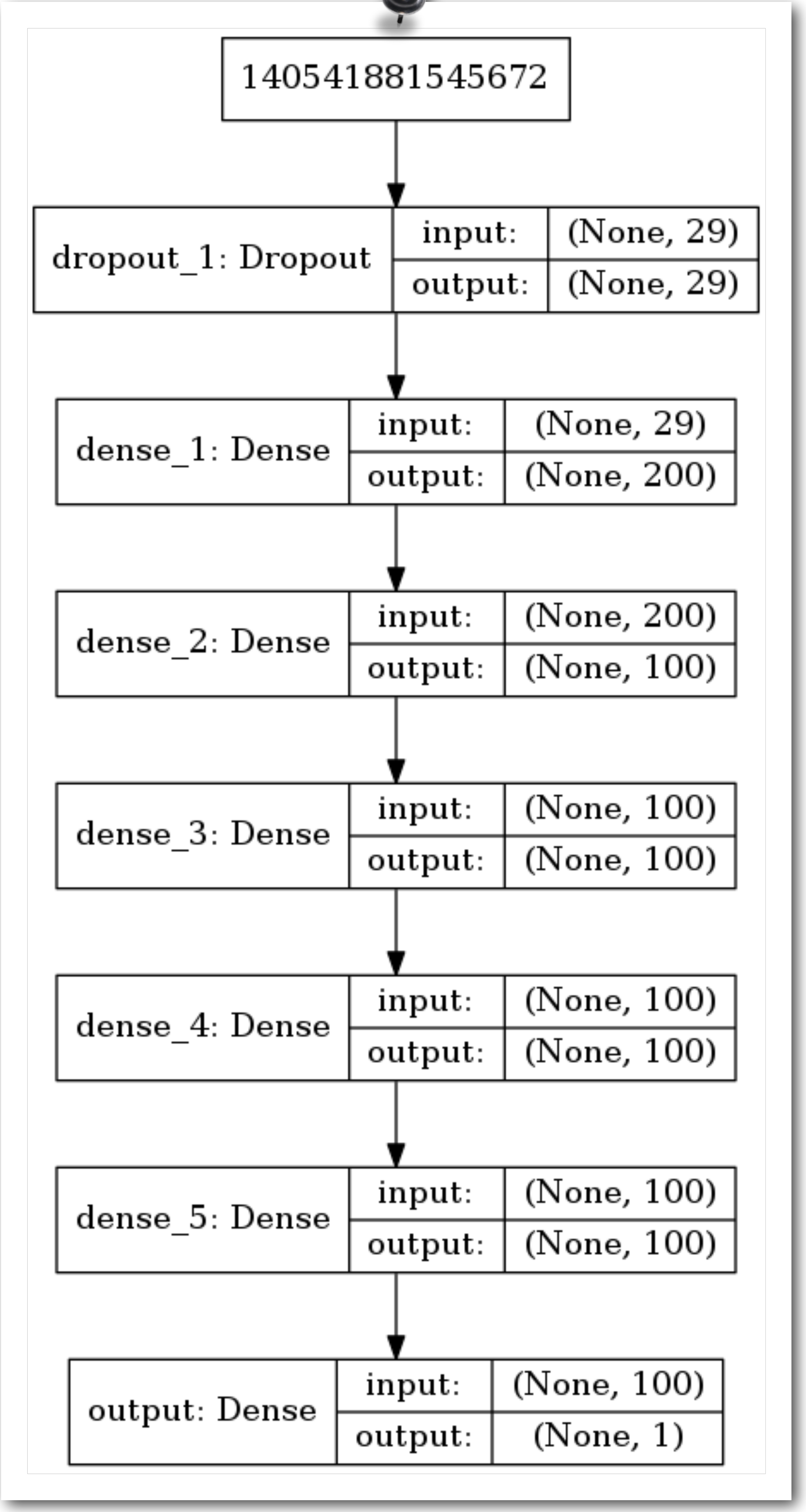
- Represents strength of linear dependence
- Two types of correlation
- Positive correlation: Both variable moving in same direction
- +1 is perfect positive correlation -1 Shows perfect negative correlation.

Deep Neural Network(DNN)

<u>Model Summary</u>
Total number of input variable :- 29
Initial weight =5
Total number of hidden layer :- 5
Loss function:- "binary_crossentropy"
Optimizer:- ADAM
Batch size:- 900
No. of epochs:-100
Ntrainsignal:-
Ntrainbackground:
Ntestsignal:-
Ntestbackground:-

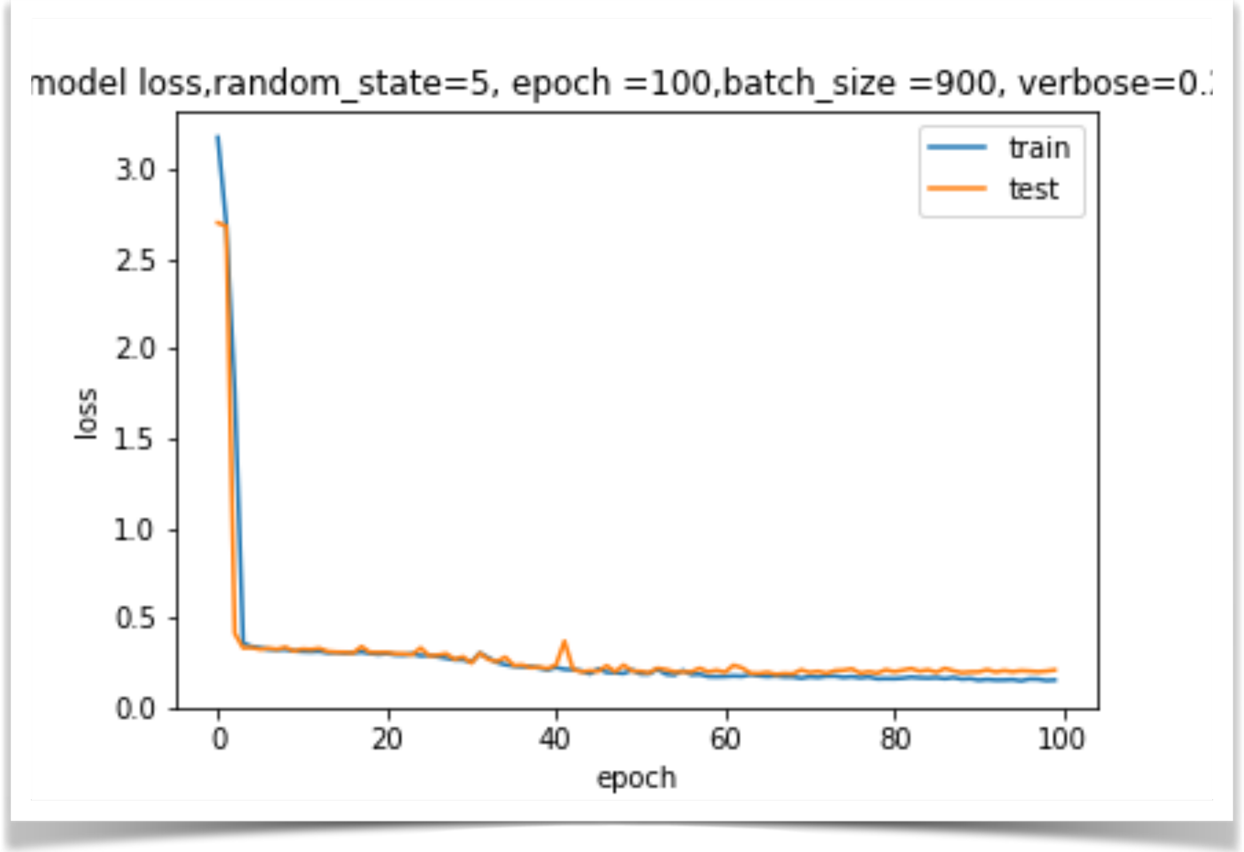
- whole dataset divided into the ratio of 80:20.

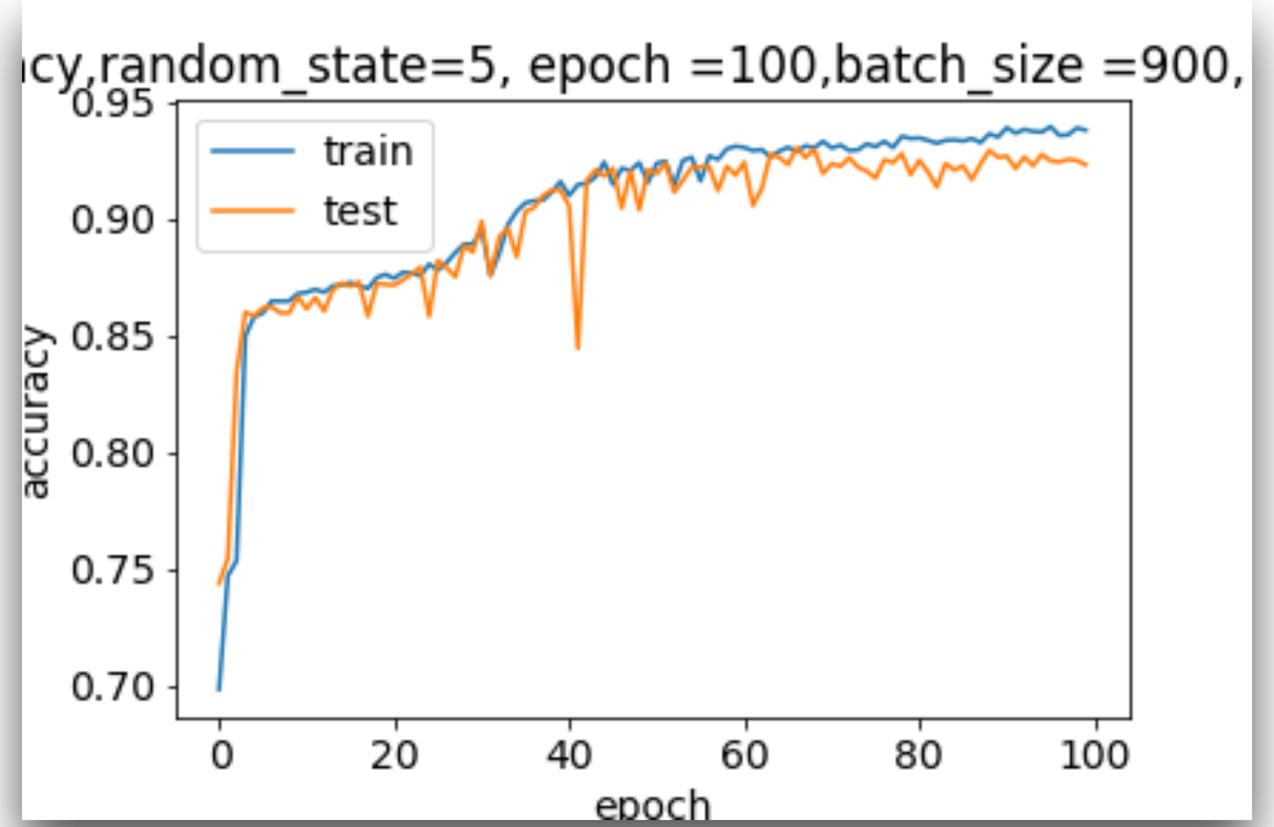




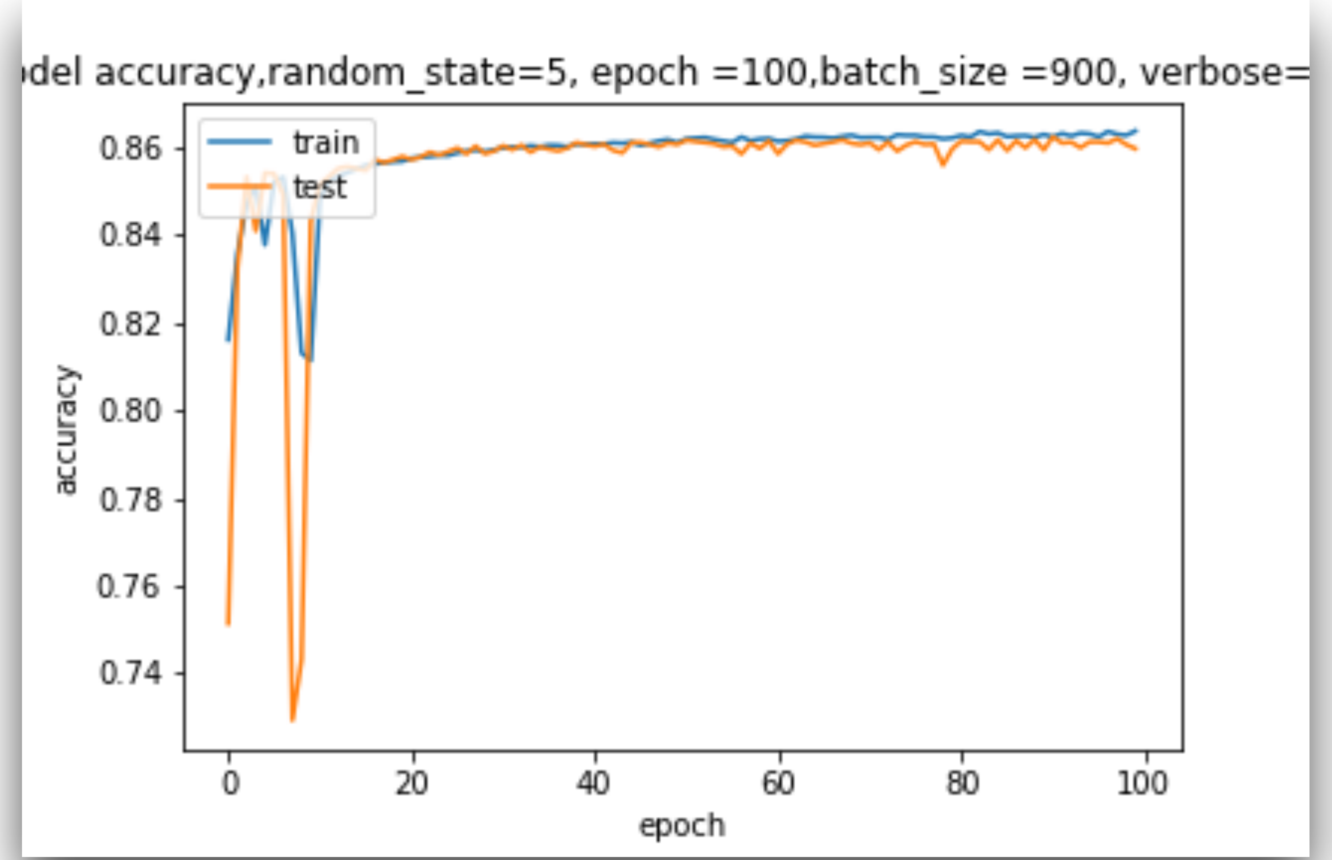
Output (DNN)
Binary Classification

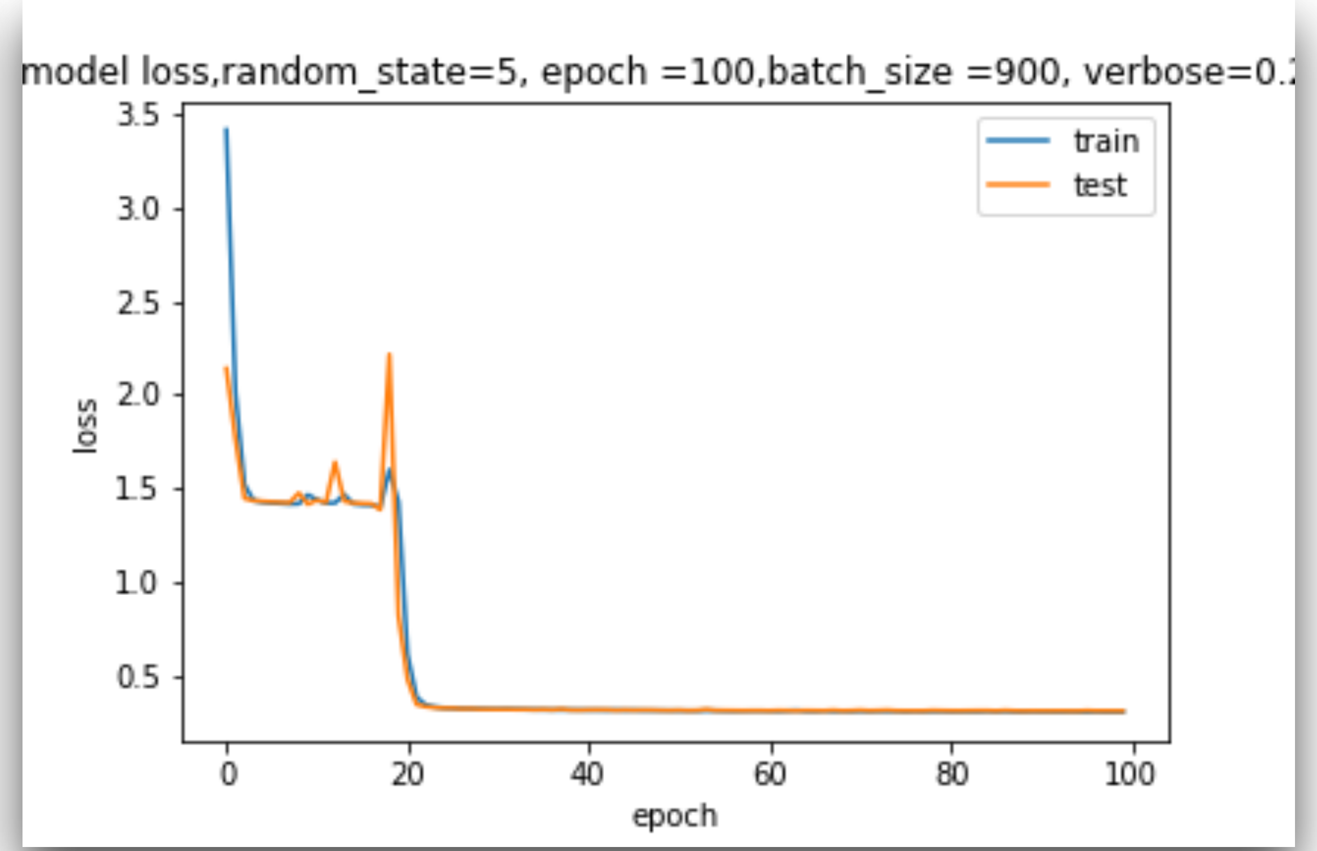
Signal	Background	Training Accuracy	Testing Accuracy
T'	ttgg	93.30%	92.06%
T'	ttgg & tth	89.84	89.07
T'	ttgg& tth &thq	86.36	86.10





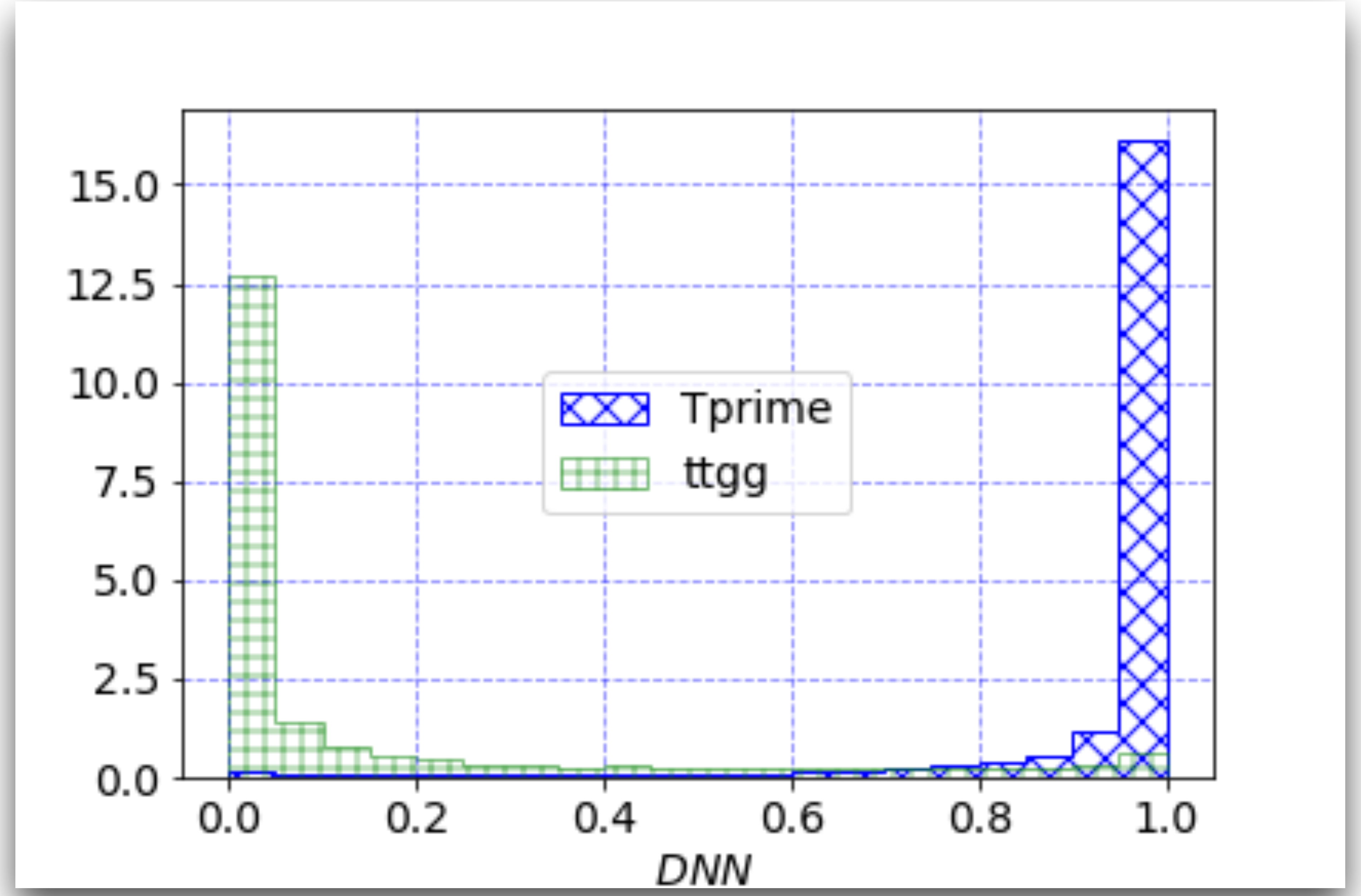
Model output when Tprime used as Signal and ttgg used as the background





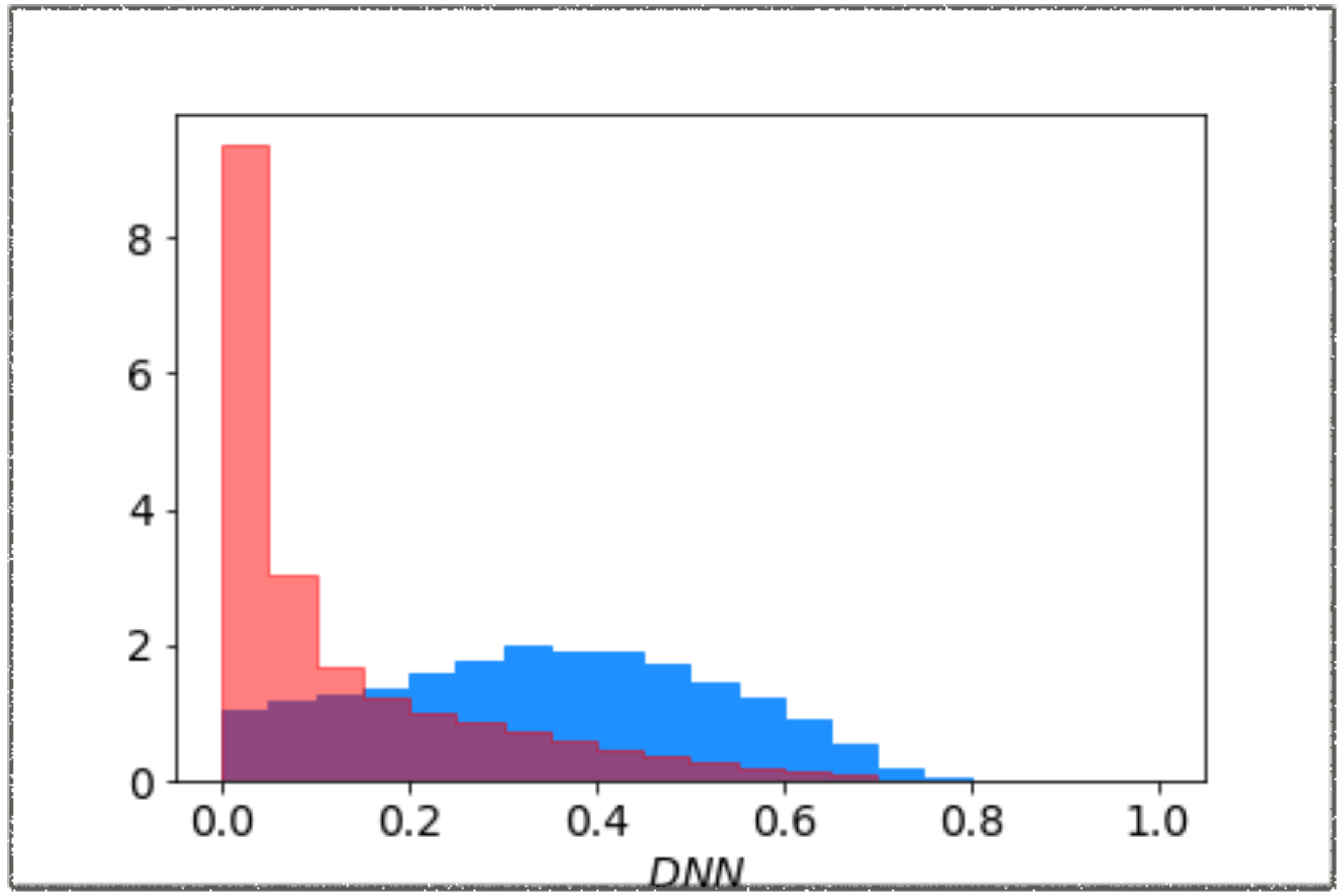
T' as signal & ttgg, ttH, and tHq as background

Output(DNN)
Binary Classification

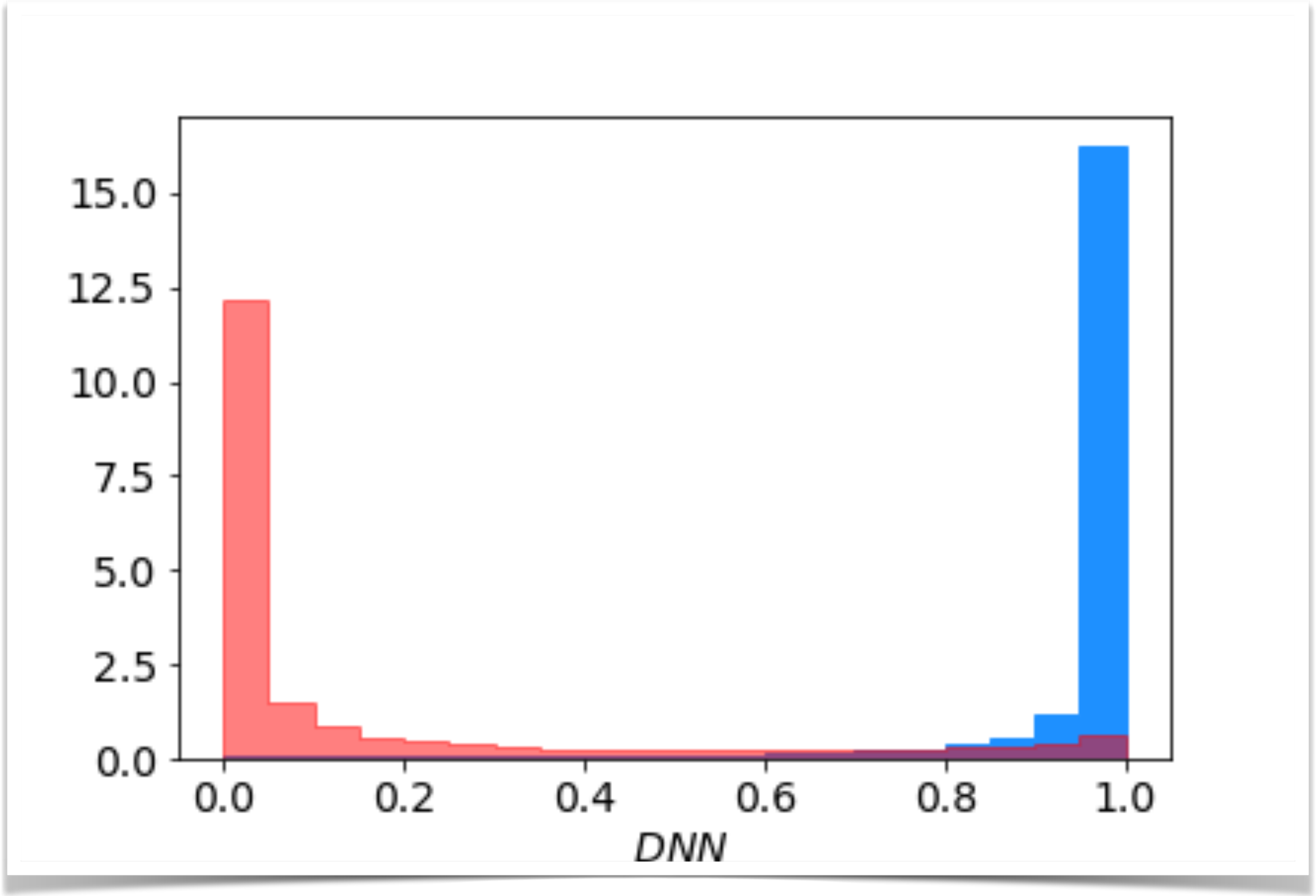


Tprime as signal and ttgg as the background

Tprime as signal and ttgg, tth, and thq as background

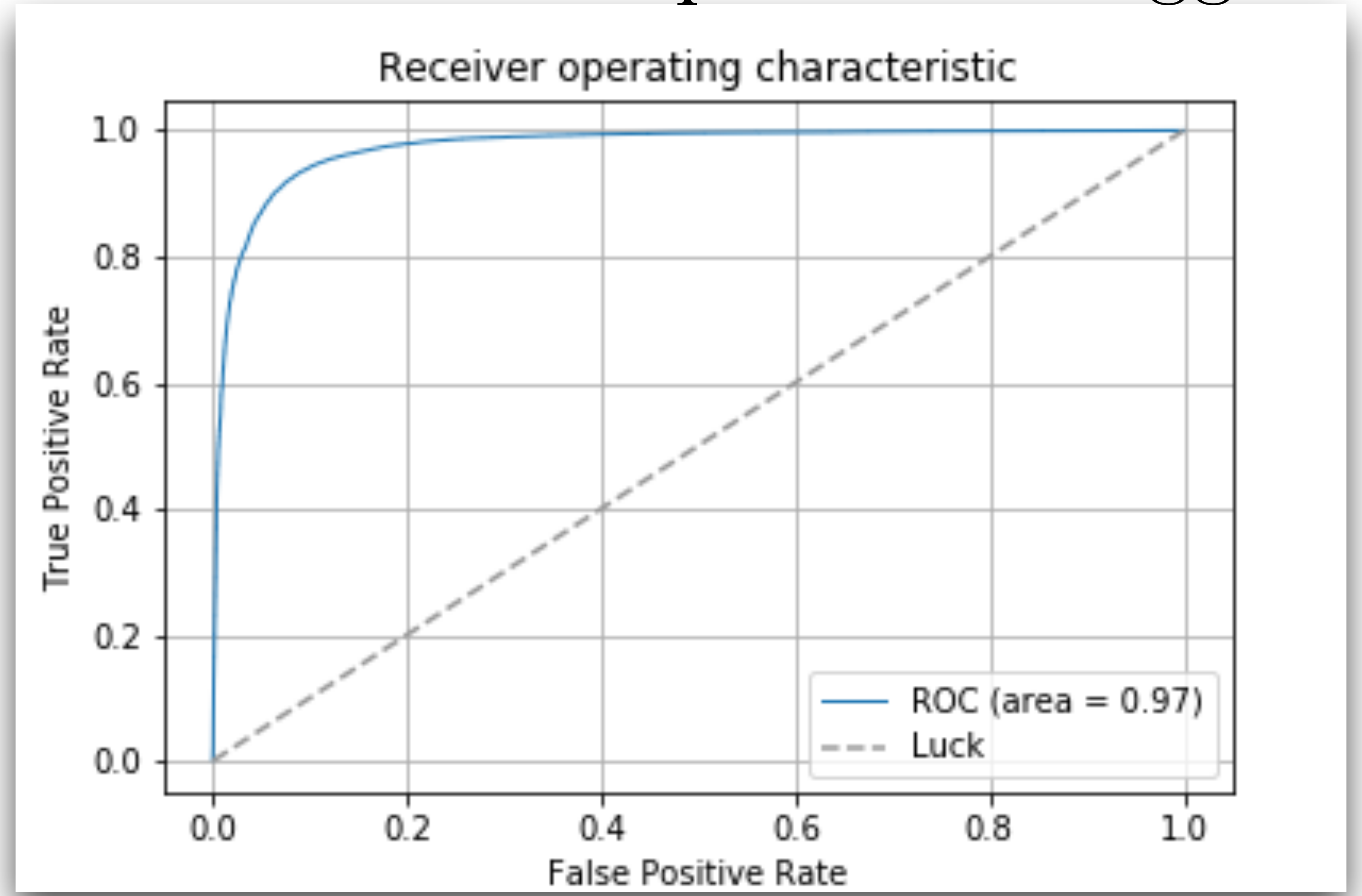


ttgg and ttH as the background

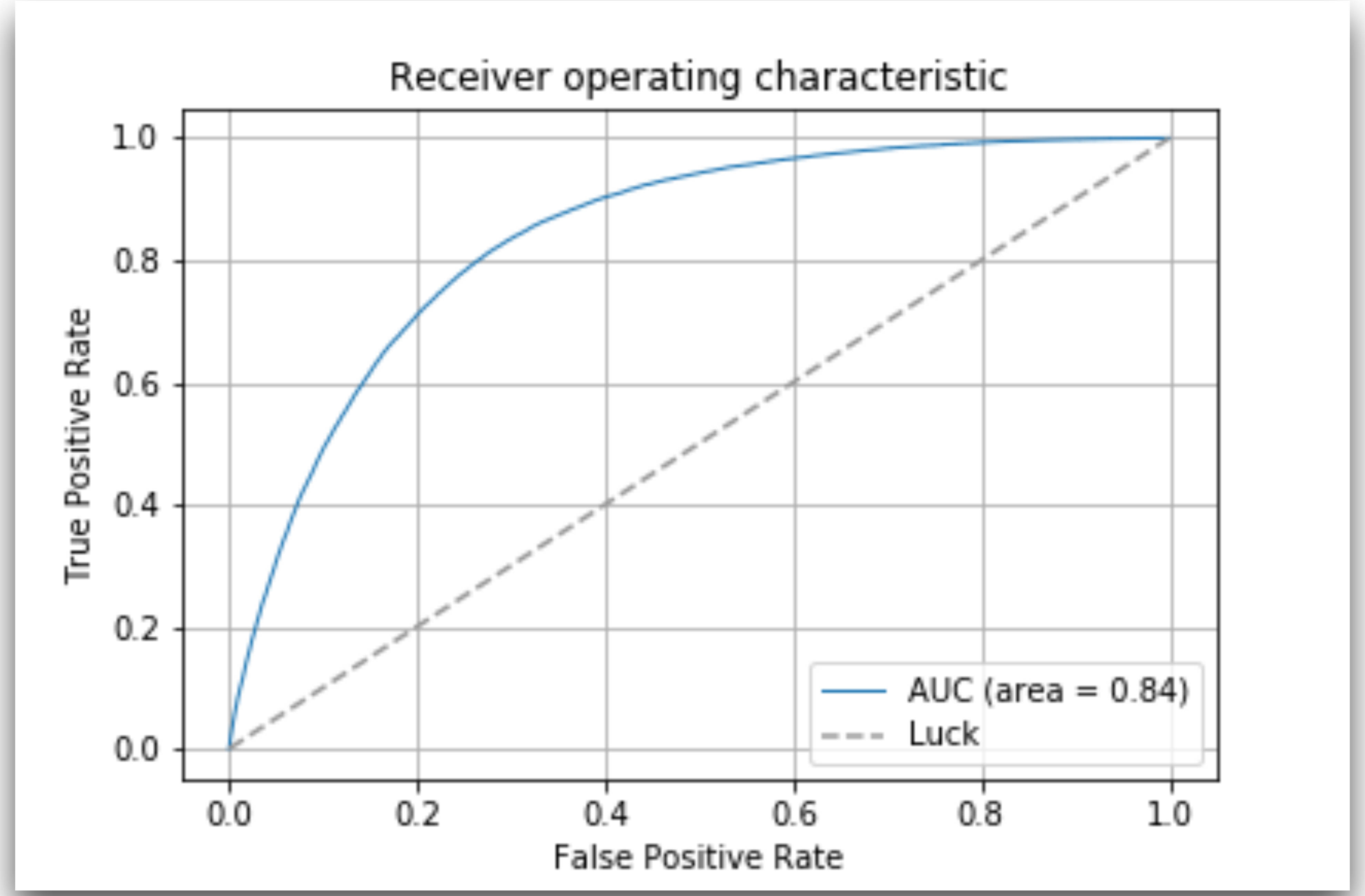


Output(DNN)

ROC Curve for Tprime and ttgg



ROC for Tprime as signal and ttH, tHq, and ttgg as background



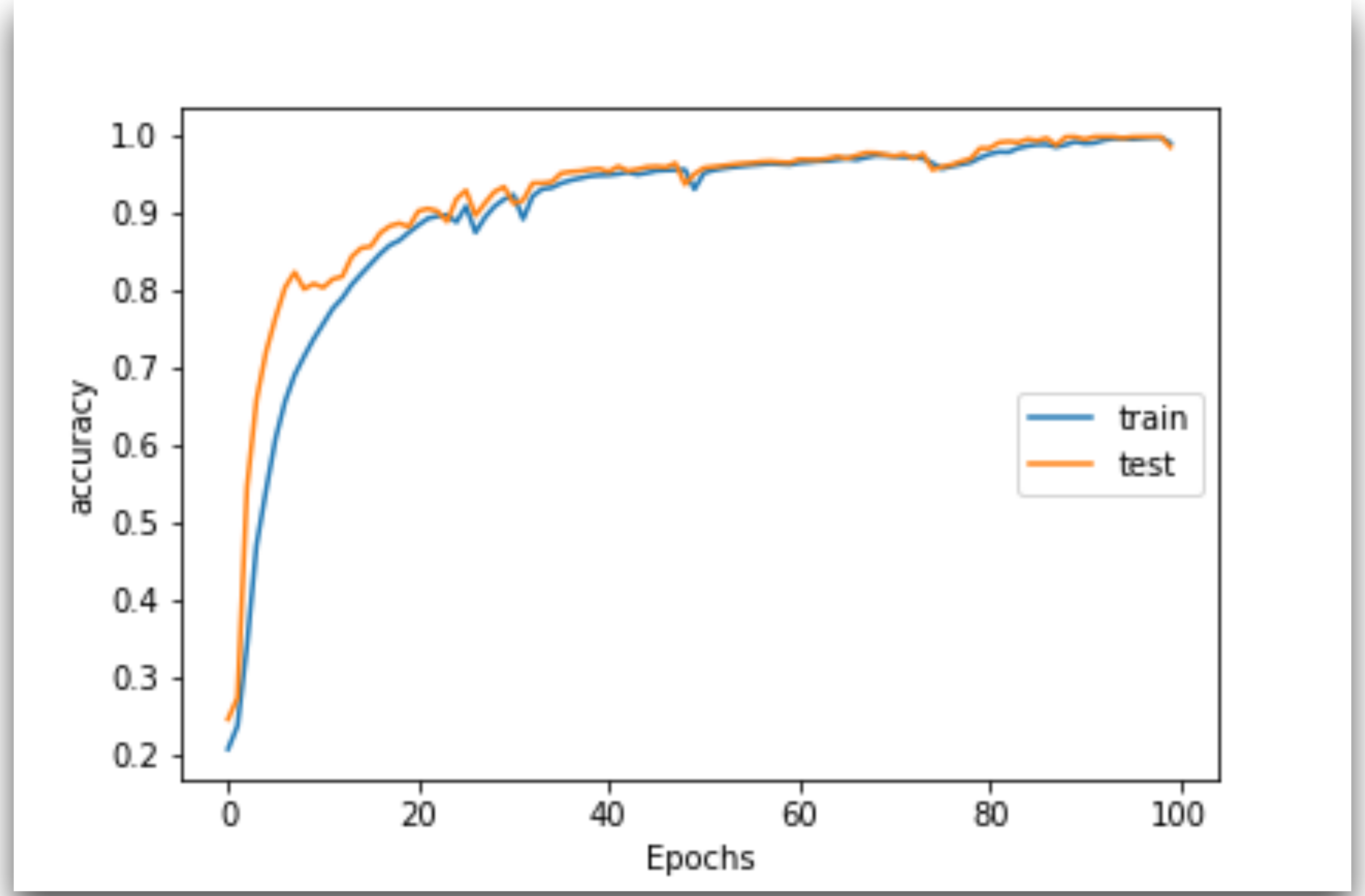
Receiver Operating Characteristic Curve(ROC Curve)
True positive rate = True positive/(True positive + false Negative)

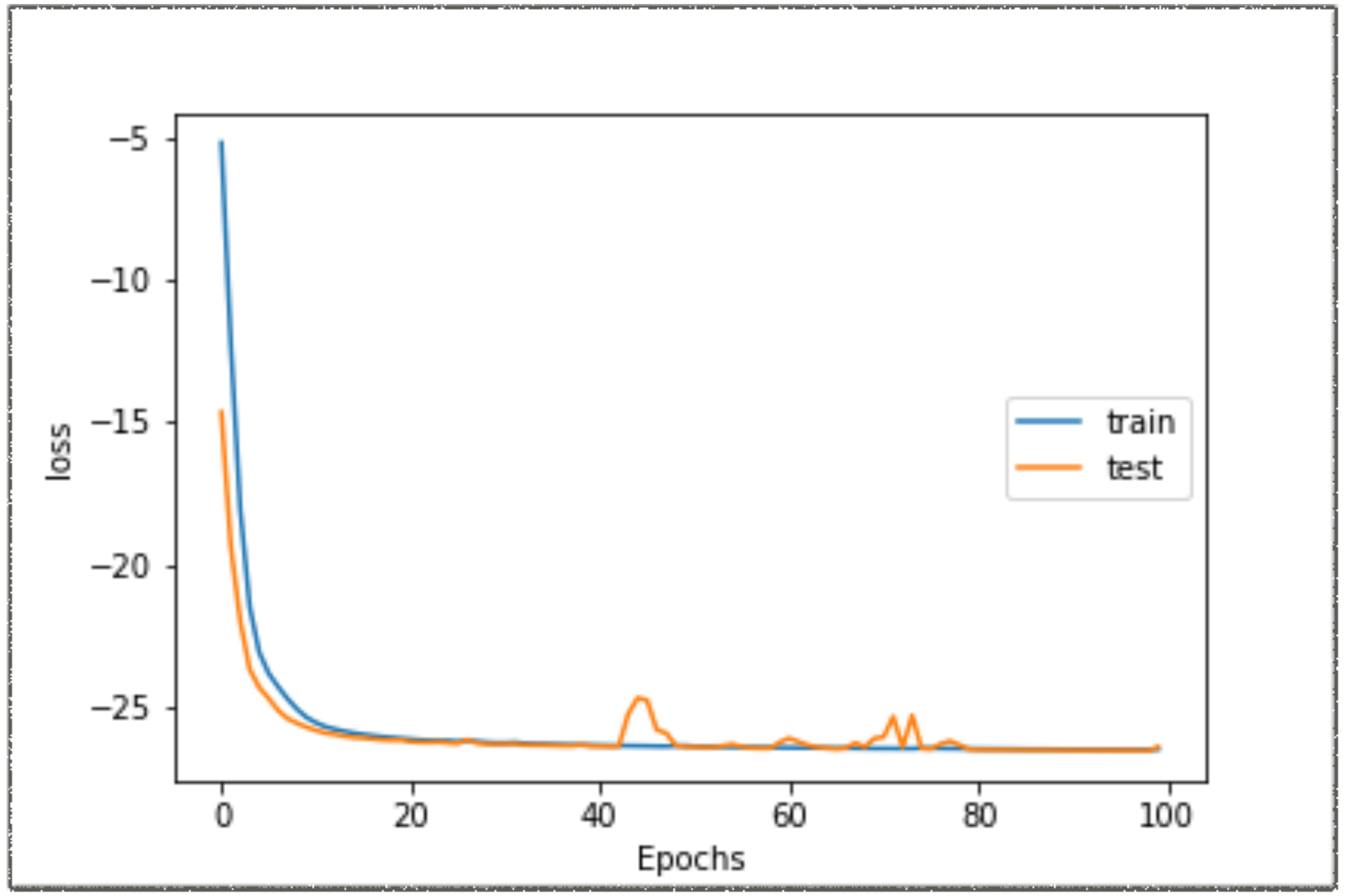
DNN Model
Multiclass

Model Summary:
Total Input:- 29
Activation function(dense): 'ReLU'
Activation function(output):- 'softmax'
Optimizer:-'ADAM'
loss function:- 'Categorical_crossentropy'
metrics = 'accuracy'
No. of Epochs: -100

Output(DNN)
Multi classification

- With the earlier binary classification, Output was not good as Multi classification

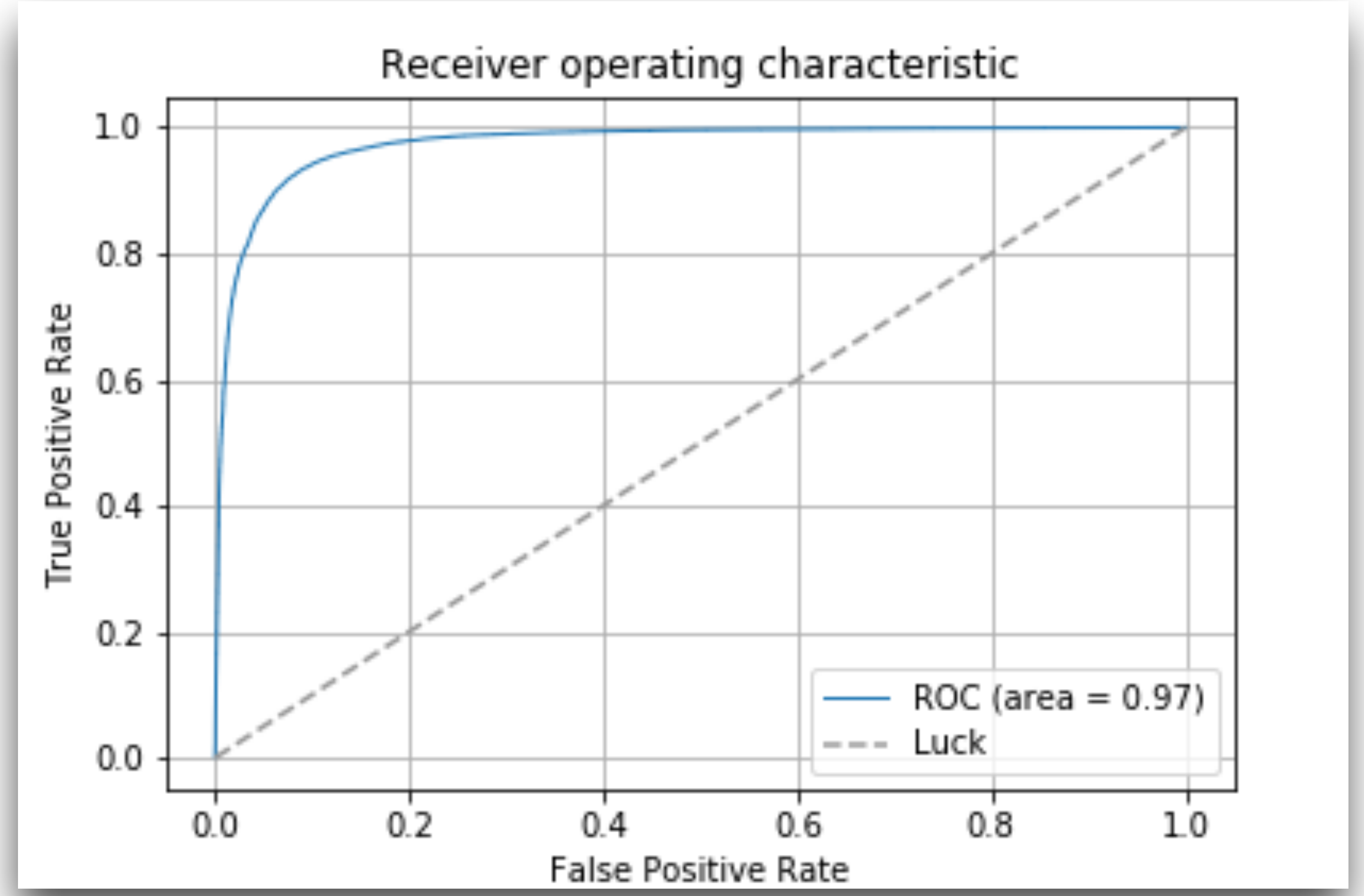




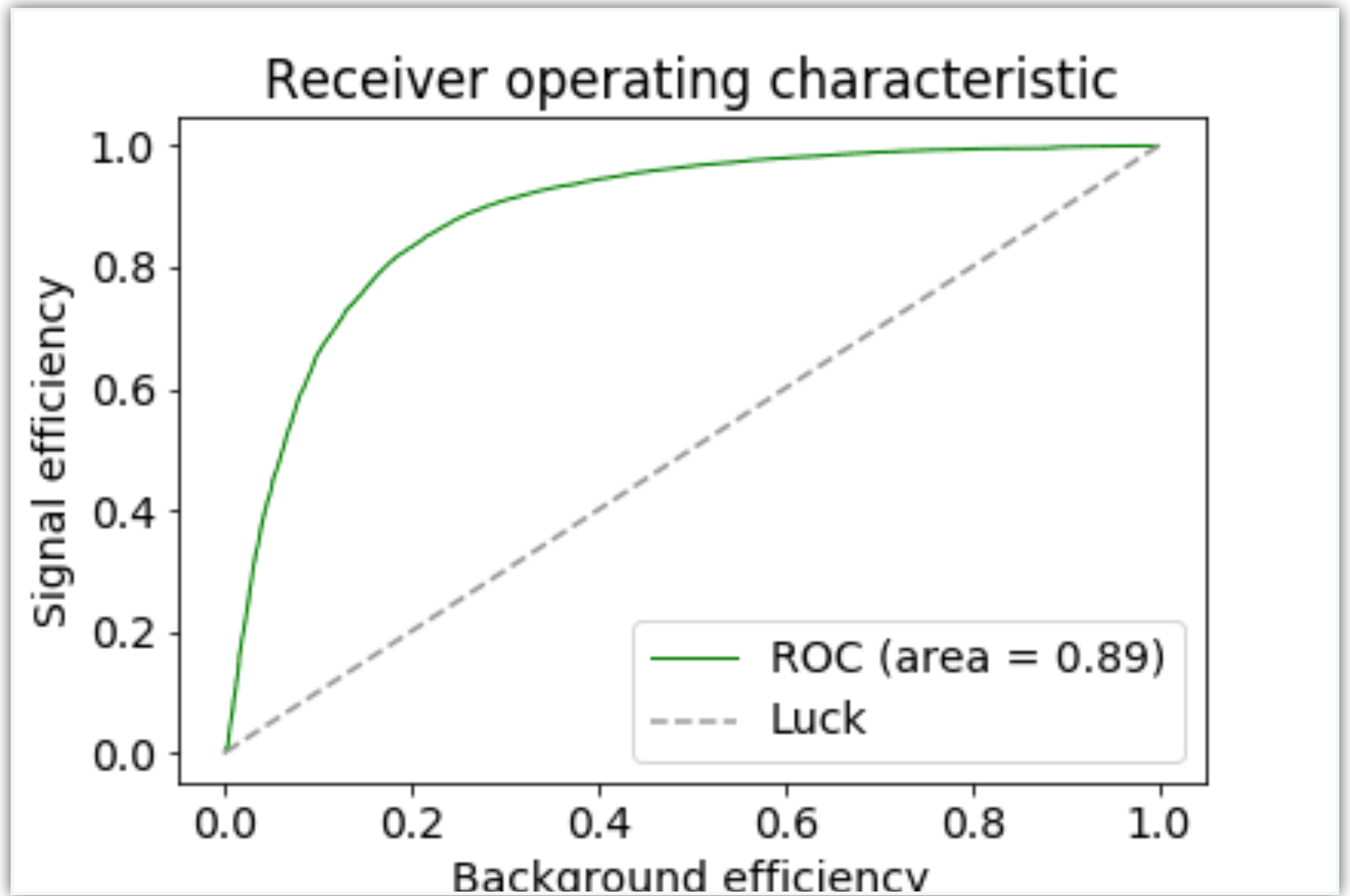
Training Accuracy:- 98.57%

Testing Accuracy :- 98.58%

Comparison of Output from DNN and BDT



ROC for DNN



ROC for BDT

- ROC for DNN are better compared to BDT
- This data comparison done over same sample.
- BDT are simple to implement.

Boosted Decision Tree

- Consists of leaves
- Classify the data into classes and then predict the output after training.
- It train on sequence of data.

Thank You

Summary

We saw how, using,

- Working of DNN model
- T' as signal and few backgrounds, we can use DNN to separate them a signal and background.

Correlation Coefficient