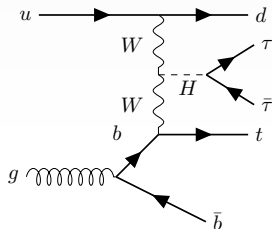# Neural network development in the analysis of single top-quark production in association with a Higgs boson and light-quark at ATLAS

Christian Kirfel

15th of March 2020

tHq ditau hadhad channel selection

- n-jets: 2 (b-jets: 1)
- b-jet WP: 70 DL1r
- nLeptons & nTaus: 1e/μ 2τ_{had}
- $E_{\text{T,miss}}$: no cut (to 800 GeV)

- jets:
  - $p_T > 35\,\text{GeV}$
  - $|\eta| < 4.5$
  - EMPFlow
- electrons:
  - $p_T > 20\,\text{GeV}$ leading 27 GeV
  - $|\eta| < 2.5$ not in 1.37 - 1.52
  - WP: LooseAndBLayerLH ;
    isolation: no requirement
- muons:
  - $p_T > 20\,\text{GeV}$ leading 27 GeV
  - $0.01 < |\eta| < 2.5$
  - WP: Loose ; isolation: no requirement
- taus:
  - $p_T > 20\,\text{GeV}$ leading 27 GeV
  - $|\eta| < 2.5$ not in 1.37 - 1.52
  - WP: RNNLoose
  - ASG recommended OLR (τ_{had} remove jets)

Challenges

Negative weights in Monte Carlo

- Negative weights arise from ... source missing
- In neural network training negative weights lead to an unwanted behaviour

Different sizes of background processes

- Dominating background can diminish the significance of secondary backgrounds in training
- Especially very signal-like backgrounds are likely to be mislabeled

Accelerating network optimisation

- Exploration of new features is only possible in an optimised network
- An evolutionary optimisation approach minimises the work effort

Background processes in neural network training
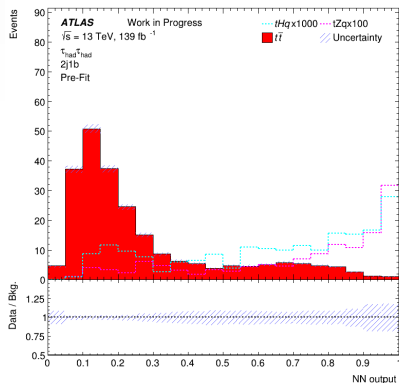
- Generally expected shape for signal and background response
- Due to the dominating size of $t\bar{t}$ it defines the training
- For example tZq gets classified as signal
- The network has to be trained to take care of smaller signal-like samples
- Approaches: multiple networks, multiple targets, reweighting of the samples

4

Impact of negative weights on the training
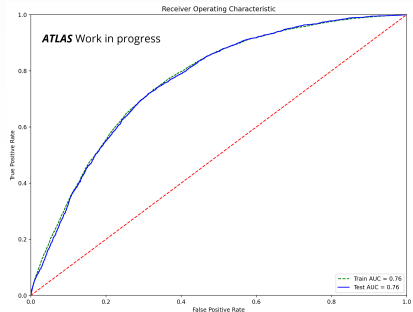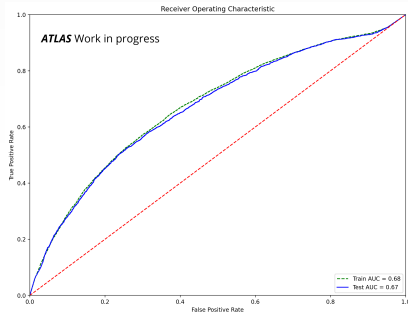
- About 35% of events have negative weights
- Negative weights break the networks training
- Possible ways of handling the weights is to use the absolutes or just use positive weights

Problems in network optimisation

Obstacles of optimisation processes

- Grid searches are both tedious and resource intensive
- Knowledge gained in one problem is rarely universal
- Experienced users can develop a bias towards certain hyperparamters

Applications of evolutionary optimisation

- A neural network should be developed in parallel to an ongoing analysis.
- New additions need a new optimisation.

Training specifications (Redundant?)

- Training a deep neural network

- Coarse optimisation using an evolutionary neural network

- Fine optimisation doing a grid search

- Optimised hyperparameters:
  - Number of nodes
  - Number of layers
  - Dropout percentage

- Signal: tHq

- Background: $t\bar{t}$

- Using absolute weights

Evolutionary optimisation of neural networks

- Combination of a grid searches with a survival of the fittest setup

- Start with a number of random hyperparamters

- Evaluate the set of hyperparameters

- Create a a new set of networks based on the previous best

- Add recombination and variation to avoid local minima and bias

Example of an evolutionary training

- tHq as signal against t$\bar{\text{t}}$ as background
- Using basic kinematic variables
- Region: 2j1b
- Weights: absolute
- Testing: nodes, layers, dropout
- Fixed hyperparameters:
  - Optimizer: Adam
  - Activation: relu, sigmoid
  - Batchsize: 1000
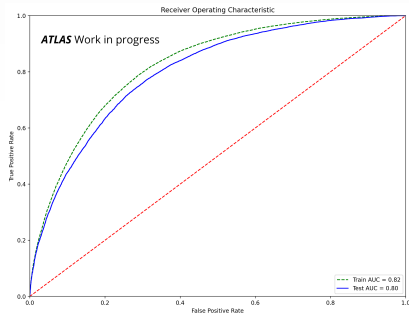 - Epochs per generation: 25

Initial parameters

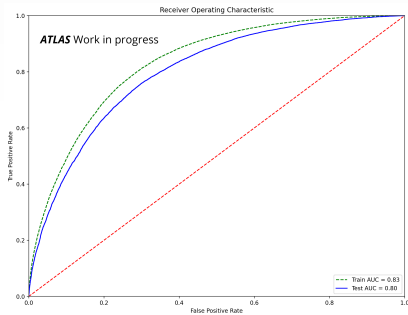- Layers: $1 - 10$
- Nodes: $1 - 100$
- Dropout: $0 - 1$

Final parameters

- Layers: 4 ± 2
- Nodes: 67 ± 33
- Dropout: 0.4 ± 0.3

Comparing ROC to a grid search

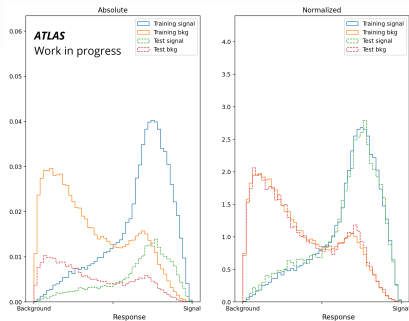

Grid search
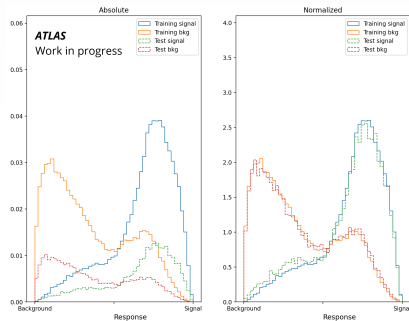

Evolutionary search

Comparing response to a grid search

Grid search   Evolutionary search

Conclusions

- Using neural networks for signal to background separation in the tHq channel is an interesting analysis that offers many challenges.

- Negative weights resulting from Monte Carlo generators cannot easily be handled by machine learning algorithms.

- Different sizes of background samples result in smaller backgrounds getting ignored or even classified as signal

- Evolutionary optimisation of neural networks is a nice approach to reoptimise a network automatically in an ongoing analysis.