Hadronic Tau MVA

Christian Kirfel
on behalf of the Hadronic Tau analysis teams

26th of March

UNIVERSITÄT BONN

Machine learning methods

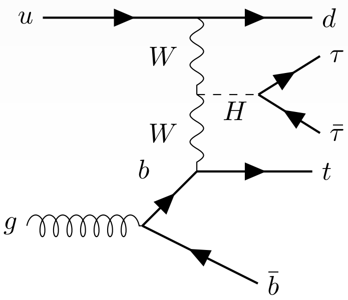
- Developing a neural network for the lepditau and dileptau channel
- Testing a BDT for the dileptau channel

1. Preselection of channel and region
2. Implementation of the MVA methods
3. Preliminary results
4. Future plans

Training specifications for the lepditau channel

- Training a deep neural network using tensorflow.Keras
- Coarse optimisation using an evolutionary neural network
- Fine optimisation doing a grid search
- Optimised hyperparameters:
  - Number of nodes
  - Number of layers
  - Dropout percentage
  - Activation function
  - Weight initialisation
  - Optimiser
- Signal: tHq
- Background: Diboson, Z+jets, ttbar, W+jets, tW, tZq
- Using absolute weights

General ML selection



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

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

Features

- Preliminary selection of variables
- Diffent choices have high impact $\rightarrow$ more investigation needed

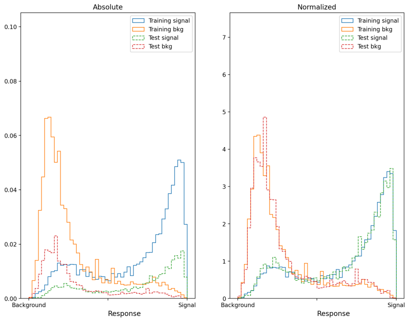
eta_jf	forward jet eta
pt_jf	forward jet transverse momentum
mass_jf	forward jet mass
phi_jf	forward jet phi
eta_b	b-jet eta
pt_b	b-jet transverse momentum
phi_b	b-jet phi
HvisMass	mass of LorentzV sum of hadronic taus
m_met	Missing energy
Reco_w_mass_2	Reconstructed mass of the W case 1
Reco_w_mass_1	Reconstructed mass of the W case 2

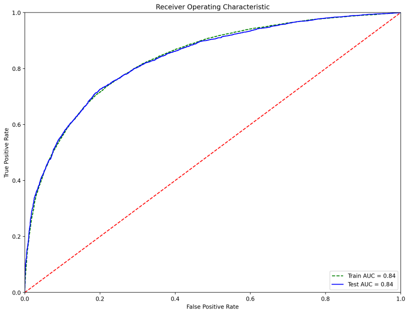
deltaRTau	Delta R of the hadronic taus
deltaPhiTau	Delta phi of the hadronic taus
HvisPt	pt of LorentzV sum of hadronic taus
HvisEta	eta of LorentzV sum of hadronic taus
TvisMass	mass of reconstructed top
TvisPt	pt of visible top
TvisEta	eta of visible top
M_b_jf	Mass of LorentV sum of b and jf
HT	Sum of transverse energies
lep_Top_pt	Light lepton pt
lep_Top_eta	Light lepton eta

A first optimisation

Hyperparameter	Setting
Nodes	240
Layers	3
Dropout	0.85
Batchnormalisation	On
Activation	elu
Output activation	sigmoid
Batch size	1000
Optimisation	Adam
Weight Initialisation	Lecun Normalisation

Results





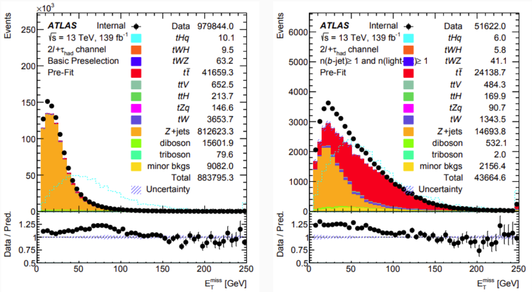
- Really good separation for this early stage of Optimisation
- Slight problems in training and validation agreement shape. Could be a statistical problem.

Future steps for the neural network analysis

- Full optimisation of the neural network
- Testing feature ranking and decorrelation
- Investigating the impact of negative weights
- Properly separating smaller backgrounds
- Cosmetic work on the response and output

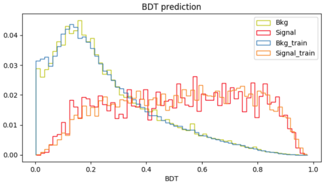
BDT Preselection

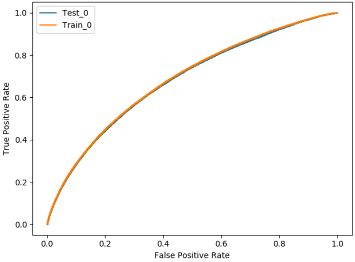
- Channel: 2 tight leptons (e/mu) and 1 (hadronic) tau
- Region: n-jets $\geq$ 1 and b-jets $\geq$ 1
- Significantly decreasing the dominant Z+jets background
- Discrepency due to missing tau fake differentiation



BDT early results

- Using XGBoost library
- Preliminary: Hyperparameters are not optimised
- Visible separation, good training and test agreement





Plan for BDT development

- BDT hyperparameter tuning
- Including BDT score in the trees
- Cut on the BDT score
- Create a BDT for the CR of the main background
- Study variables related to SS and OS: Usually, SS and OS have different background contributions.
- Get the distributions for forward and central jets separately

Conclusion and future steps

- Both neural network and BDT show very good separation for an early stage of optimisation
- For the lepditau channel the separation is outstanding but the stability is bad. This means the agreement between training and validation is often bad or has regions that are not understood.
- In both cases the variables have a significant impact on the result. Different sets of features need to be explored to find better separations.
- A decorrelation and ranking would be desireable
- The handling of negative weights is an ongoing problem that needs to be investigated to see if a conclusion can be reached for MVA techniques.

Evolutionary neural networks

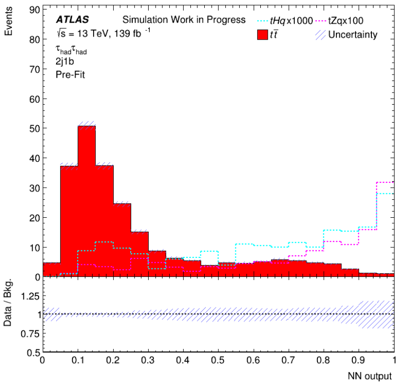
- Starting with a set of random configurations
- Evaluate the results of the first generation and generate a new generation based on
- Repeat until a good configuration is reached
- Advantages:
  - Decrease user bias for hyperparameter choice
  - Optimised to run on worker nodes
  - Quick discarding of bad configurations
  - User friendly for unexperienced students

Inner product

$$a^\mu = \begin{pmatrix} p_T \cosh(\eta) \\ p_T \cos(\phi) \\ p_T \sin(\phi) \\ p_T \sinh(\eta) \end{pmatrix}$$

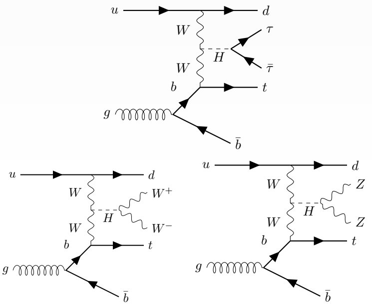
$$\begin{aligned} \langle A|B\rangle = A_\mu B^\mu \\ = p_{T,A}p_{T,B}(\cosh(\eta_A)\cosh(\eta_B) - \cos(\phi_A)\cos(\phi_B) \\ - \sin(\phi_A)\sin(\phi_B) - \sinh(\eta_A)\sinh(\eta_B)) \\ = p_{T,A}p_{T,B}\left(\cosh(\eta_A - \eta_B) - \cos(\phi_A - \phi_B)\right) \end{aligned}$$

Background processes in neural network training



- tHq against other processes
- ttbar dominates the training
- tZq misclassified as signal
- Possible approaches:
  - multiple networks
  - multiple targets
  - reweighting samples

General ML selection



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

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

1. Understand impact of negative weights on variable shape
2. Investigate variable shapes for different treatments
3. Understand impact of negative weights in ML algorithms
4. Test different approaches for negative weight handling and investigate the model stability.