Comparison of Multivariate Techniques in the Wt Single Top-Quark Production Channel at ATLAS

Nicolas Erasmus Boeing

Bachelorarbeit in Physik
angefertigt im Physikalischen Institut

vorgelegt der
Mathematisch-Naturwissenschaftlichen Fakultät
der
Rheinischen Friedrich-Wilhelms-Universität
Bonn

Aug 2017

Ich versichere, dass ich diese Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt sowie die Zitate kenntlich gemacht habe.

Bonn, ...
Datum Unterschrift

1. Gutachter: Prof. Dr. Ian C. Brock
2. Gutachter: Dr. Eckhardt v. Törne

Acknowledgements

First and foremost I would like to thank Prof. Brock for giving me the opportunity of working on such a contemporary and exciting topic that provided me with this unique experience.

A big thank you also to all members of the group for welcoming me so nicely, you were always ready to help and without your company it would not have been nearly as enjoyable as it was. Special thanks to my tutor Rui for taking so much time out of your day to show me the ropes and answer my many questions.

Lastly, I want to thank my parents for always supporting me on this journey.

Contents

CHAPTER 1

Introduction

Over time particle colliders have become more and more powerful, giving rise to increasingly complex processes that need to be studied in high energy physics. To achieve this, more sophisticated techniques are needed, resulting in machine learning techniques becoming more and more relevant over the past years. The goal of this thesis is to study different machine learning techniques that can be used to separate the desired signal process from unwanted background processes that have similar signatures. Here, the single top quark Wt channel with both W bosons decaying leptonically is used.

Chapter 2 establishes the physics needed in this thesis, summarizing the Standard Model and how it is tested with the ATLAS detector at the Large Hadron Collider, as well as giving some insight into the process measured in this analysis, top quark decay in the Wt dilepton channel. Chapter 3 introduces multivariate analysis as well as the specific methods used in this thesis. The results of these techniques applied on the Wt dilepton channel is presented in Chapter 4 using various performance metrics, and outlining the optimisation and variable selection process for each model. Chapter 5 compares the results of the different models and discusses them. Finally, Chapter 6 gives a brief summary of the results.

CHAPTER 2

Theoretical and Experimental Background

2.1 The Standard Model

The basis of modern particle physics is the Standard Model that combines our knowledge into the most well-tested physics theory in history. Developed in the 1970s, it describes the 12 elementary particles (and 12 antiparticles) and the forces between them, carried by 4 gauge bosons, as well as the Higgs boson, which gives mass to all particles[1].

2.1.1 Elementary particles

The elementary particles can be organised into two groups, quarks and leptons, which can further be split into three generations of pairs, as seen in Fig. 2.1. The quark group consists of up and down, charm and strange and top and bottom quarks with charges 2/3 and $-1/3$, respectively. Each quark is further divided into three possible "colours", red, green and blue, the charge of the strong force. Leptons are composed of electron, muon and tau, each with charge -1, and their respective neutrinos, which don't have any electrical charge. All fundamental particles have spin 1/2 [2]. Only particles of the first group exist in stable configurations in everyday life. Up and down quarks make up protons and neutrons that form the atomic nuclei, surrounded by electrons. So far all other elementary particles have only been observed in cosmic radiation after interacting with the earth's atmosphere and in particle colliders.

2.1.2 Fundamental forces and their gauge bosons

Three forces are described by the Standard Model: electromagnetic, strong and weak forces. The electromagnetic force governs almost all everyday phenomena as it is the force acting between atoms. Its force mediator is the photon, a massless particle with charge 0 and spin 1, which interacts with all elementary particles that have electric charge.

The second fundamental force is the strong force, which holds protons and neutrons together in atomic nuclei. It is, as its name suggests, very strong, but acts only at very short distances. Its charge is the colour charge, carried only by quarks and its gauge boson, the gluon, which comes in 8 different colour variants. Similar to the photon it is massless and has spin 1, however due to its colour charge it can interact with itself. Additionally, colour charge is not visible to the outside, quarks can only exist in bound states inside of colour-neutral composite particles, such as qqq or $q\bar{q}$.

The weak force can typically be seen in nuclear β-decay. Its mediators are the $W^{\pm}$ and Z^0 bosons with masses of 80 GeV and 91 GeV and electric charges of ± 1 and 0, respectively and a spin of 1. The weak bosons couple to all elementary particles, including neutrinos which are otherwise not detectable. At high

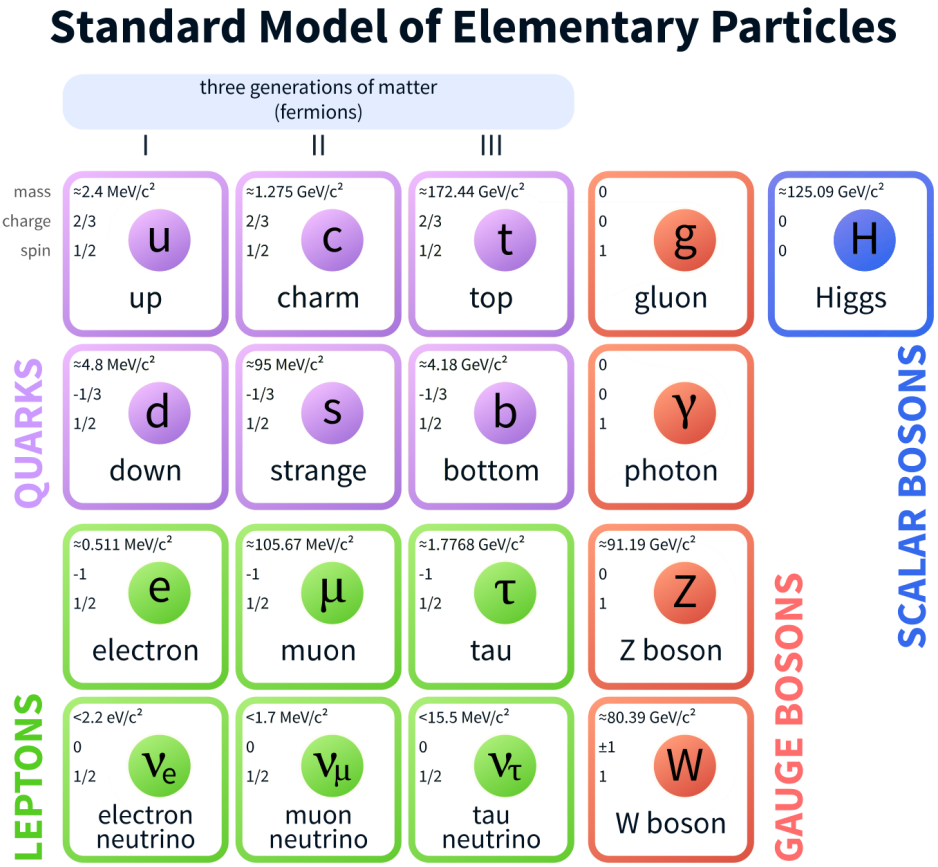


Figure 2.1: Elementary particles as described by the Standard Model [3].

energies electromagnetic and weak forces combine into the electroweak interaction with the coupling given only by the electric charge e which predicts all effects that can be seen and is a broken symmetry at low energies.

The fourth fundamental force, gravity, is not described by the Standard Model. It is incredibly weak, about 10^{36} times weaker than the electromagnetic force, and well described by Einstein's General Relativity. To this day it has not been possible to mathematically combine these two theories. Luckily however, gravity is weak enough to be disregarded in particle physics experiments. The hypothesised force carrier of gravity is the "graviton".

2.1.3 The Higgs boson

The Higgs boson is an excitation of the Higgs field, which gives particles their mass due to spontaneous symmetry breaking. It was first observed in July 2012 at the LHC and has a mass of (125.09 ± 0.24) GeV [4], almost 50 years after being theorised by Peter Higgs in 1964 [5].

2.2 The Large Hadron Collider and the ATLAS detector

2.2.1 The Large Hadron Collider

To observe and study the elementary particles they must first be created. This is achieved by colliding long-lived particles at high energies and studying the debris. The largest and most powerful particle collider in the world is the **L**arge **H**adron **C**ollider (LHC) located at CERN near Geneva, Switzerland. It is a 27 km long ring, built on average 100 m below the ground. It smashes together protons at energies up to 13 TeV, thereby allowing us to do precision measurements of the known particles and search for new high mass particles [6]. A total of 4 major experiments exist at the LHC: ATLAS and CMS, which are general purpose detectors [7, 8], ALICE for researching quark-gluon plasma [9] and LHCb, which specialises in b-physics [10].

2.2.2 The ATLAS detector

ATLAS (**A** **T**oroidal **L**HC **A**pparatu**S**) is one of the general purpose particle detectors at the LHC, designed to measure as many physics processes as possible, including Higgs boson decays and searches for physics beyond the Standard Model [11]. ATLAS, at 44 m length and 25 m height the largest collider-detector ever constructed [12], consists of 3 major parts, the inner detector, the electromagnetic and hadronic calorimeter and the muon spectrometer. The overall structure is shown in Fig. 2.2.

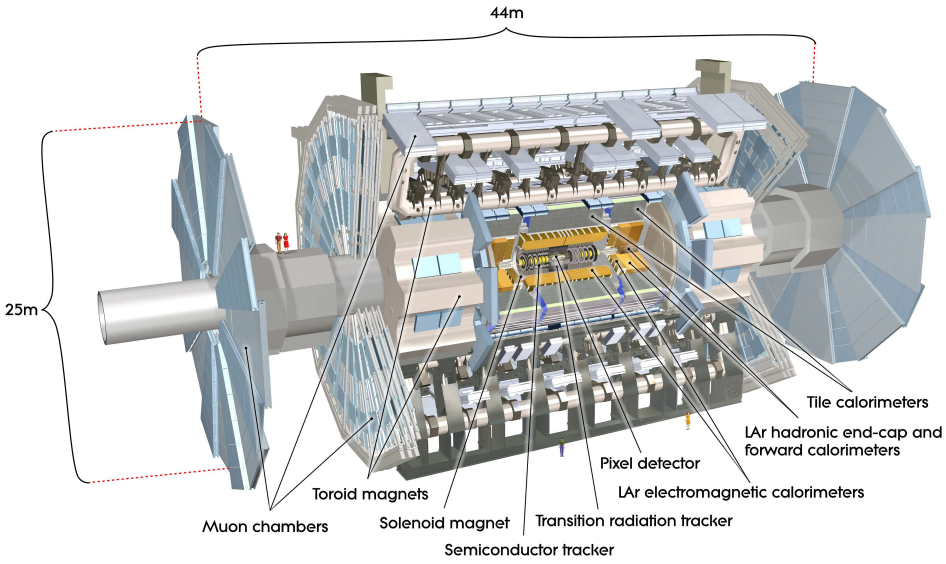


Figure 2.2: Sketch of the ATLAS detector [13].

Coordinate system. The interaction point describes the origin of the coordinate system used by ATLAS, the beam pipe defines the z-axis with the x-y-plane transverse to it. The positive x-axis points towards the center of the LHC ring and the y-axis points upwards. The azimuthal angle ϕ is measured around the beam axis while the polar angle θ is the angle from the beam axis. From these quantities additional variables can be defined, such as the pseudorapidity $\eta = -\ln\tan(\theta/2)$ that measures the angle of a particle away from the beam line, and the tranverse momentum $p_{\mathrm{T}} = \sqrt{x^2 + y^2}$. [13].

Magnet system. Strong permanent magnets are required to bend the tracks of particles in the trackers. A thin superconducting solenoids surrounds the inner detector cavity, providing a field strength of 2 T. Additionally, three superconducting toroid magnets surround the entire detector. They produce a field of 0.5 and 1 T for the muon tracker in the central and end-cap regions, respectively.

Inner detector. The Inner Detector (ID) has a radius of 1.15 m and a length of 7 m, surrounded by a solenoid magnetic field of 2 T. It tracks outgoing particles from the interaction point, measuring their R–Φ and z coordinates precisely in a range of $|\eta| < 2.5$. This is achieved by using high precision silicon pixel detectors in the innermost part, surrounded by silicone microstrip trackers (SCT). On the outside of the ID the Transition Radiation Tracker (TRT), a straw gaseous detector, is used for a more accurate momentum measurement of the particles.

Calorimeters. The calorimeters consist of electromagnetic and hadronic calorimeters arranged to cover a large range of $|\eta| < 4.9$. The electromagnetic detectors comprise of the lead-liquid argon (LAr) electromagnetic barrel and end-cap for precision measurements of electrons and photons. Hadronic showers are measured by the tile barrel and LAr hadronic end-cap ensuring good jet reconstruction and $E_{\mathrm{T}}^{\mathrm{miss}}$[1] measurement. Particles with high pseudorapidity $|\eta| > 3.2$ are measured by the LAr Forward Calorimeter (FCal), whose multiple modules detect both electromagnetic and hadronic showers.

Muon spectrometer. The muon spectrometer forms the outer layer of the ATLAS detector. It is designed to measure muons that pass though the calorimeters. Monitored Drift Tubes (MDT) precisely measure the particle tracks over a large η-range, complemented by Cathode Strip Chambers (CSC) at $|\eta| > 2$. Part of the muon spectrometer is also a trigger system for pseudorapidities of $|\eta| < 2.4$, consisting of Resistive Plate Chambers (RPC) in the barrel as well as Thin Gap Chambers (TGC) in the end caps. It allows bunch-crossing identification, precise p_{T} thresholds and complements the precision-tracking chambers in measuring muon coordinates.

Trigger system. The ATLAS detector produces many terabytes of raw data every second that has to be reduced for analysis. This is done with a multi-level trigger system. The first trigger (L1) uses limited information to make a decision within 25 µs, heavily reducing the amount of data. Its search includes high tranverse momentum and missing energy. L1 also defines Regions-of-Interest (RoI), regions in which potentially interesting interactions occur. The second trigger (L2) uses all available data within the RoI to further reduce the event rate. It has an average processing time of 40 ms. Finally, offline analysis is used to pick any events that are permanently stored. It uses about 4 s of processing time per event and reduces the data rate to 320 MB/s.

2.3 Top quark physics

2.3.1 The top quark

The top quark, with a mass of $(173.1 \pm 0.6)\,\mathrm{GeV}/c^2$, is the heaviest of all known elementary particles [4]. After being indirectly searched at LEP it was finally directly discovered in 1995 by the independent experiments CDF [14] and DØ [15]. As an "up-type" quark it has a charge of 2/3 and a spin of 1/2. Due to its extremely short mean lifetime of $\tau = 5 \times 10^{-25}$ s, less than the hadronisation time, it is the only quark that decays before forming bound states, allowing it to be studied directly at the LHC.

[1] missing transverse energy, required by the laws of energy and momentum conservation but not observed in the detector

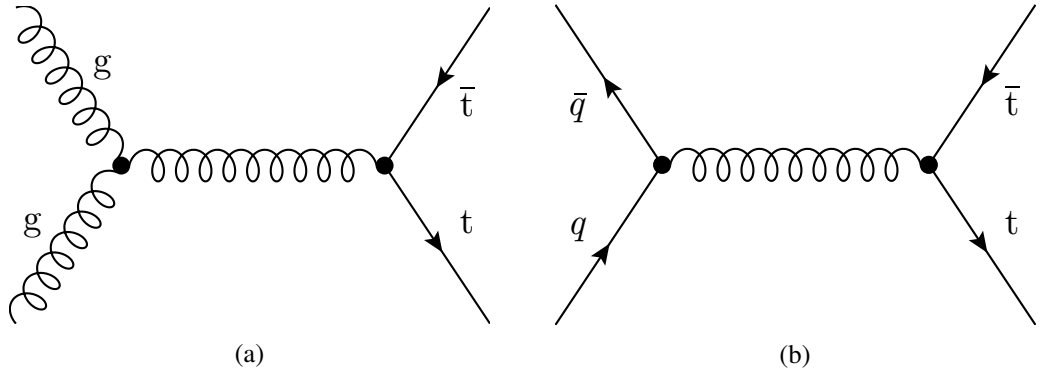


Figure 2.3: Feynman diagrams for leading order $t\bar{t}$ production: (a) gluon fusion, (b) $q\bar{q}$ annihilation.

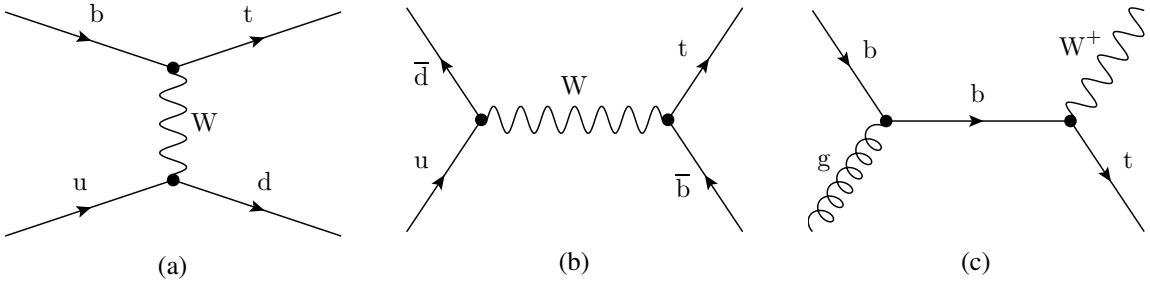


Figure 2.4: Example diagrams for leading order single top production: (a) t-channel, (b) s-channel, (c) Wt-channel.

2.3.2 Top-quark production at the LHC

There are two possibilities for producing top quarks at the LHC. Top quark pair production refers to the creation of a top anti-top pair via strong interaction while single top-quark production is, as the name suggests, the production of a single top quark via electroweak interaction [16].

Top-quark pair production. The leading source of top quarks in hadron colliders is top-quark pair ($t\bar{t}$) production. The two processes are illustrated in Fig. 2.3. At the LHC at $\sqrt{s} = 14\,\text{TeV}$ gluon fusion ($gg \to t\bar{t}$) makes up about 90%, while quark-antiquark annihilation ($q\bar{q} \to t\bar{t}$) is responsible for most of the rest [4].

Single top-quark production. Electroweak single top-quark production can also be observed at the LHC. Three mechanisms, shown in Fig. 2.4, contribute to it. The t-channel is the dominant one with a cross section of $\sigma = 217.0^{+9.0}_{-7.7}\,\text{pb}$ at $\sqrt{s} = 13\,\text{TeV}$ [17], where a sea b-quark exchanges a W boson with another typically light quark to change flavor to a top quark. In the s-channel an up-type quark and down-type antiquark annihilate to a W boson, which then turns into a top and antibottom quark. Is has the lowest cross section of the three mechanisms at $\sigma = 10.3^{+0.4}_{-0.4}\,\text{pb}$. The production mechanism studied in this thesis is the Wt-channel, in which a sea b-quark emits an on-shell W boson after (or at the same time as) being struck by a gluon, thereby turning into a down-type quark, which in almost all cases is a b-quark due to d- and s-quarks being highly suppressed by the CKM matrix. Its cross section is $\sigma = 71.7^{+3.4}_{-3.4}\,\text{pb}$.

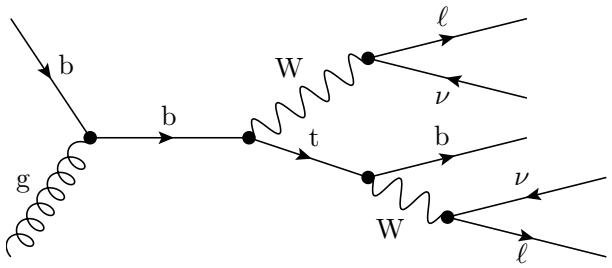


Figure 2.5: Example of a leading order Feynman diagram for the Wt dilepton channel including decays.

2.3.3 The Wt dilepton channel

As mentioned, in the Wt-channel the final state consists of a W boson and a top quark. The top quark will with a branching fraction of almost 100 % decay into a W boson and a b quark at leading order ($t \to Wb$). The b quark can be measured as a hadronic jet, which, due to the b quark's long lifetime, can be tagged [18]. A W boson can decay hadronically, into a $q\bar{q}$ pair of different flavour or leptonically into a lepton and its anti-neutrino, resulting in three possible channels for Wt-production: all-hadronic (both decay hadronically), lepton+jet (one decays hadronically, the other leptonically) and dilepton (both decay leptonically). In this thesis, only the Wt dilepton decay is considered (Fig. 2.5).

2.3.4 Background considerations

There are many different Standard Model processes with similar signatures that mix with the desired signal and have to be filtered out as best as possible. The most important background process is $t\bar{t}$ production, where both top quarks decay into a b quark and a W boson. Other background processes are "QCD multi-jets", denoting any process where no top quark or weak boson is produced, "W+jets" and "Z+jets", production of a single W or Z boson alongside jets, "diboson", production of a pair of weak bosons (WW, WZ or ZZ), and single top quark s- and t-channel production, none of which are included in this thesis.

CHAPTER 3

Multivariate Analysis in High Energy Physics

Separating a desired signal from background is a very important part of high energy physics analyses. Many processes have similar footprints in the detector and as there is no single discriminating variable to separate them, multivariate analysis (MVA) techniques are a powerful tool. MVA refers to any analysis that combines many variables into a single discriminator. In this thesis, three methods are compared: boosted decision trees (BDT), NeuroBayes (NB) and deep neural network (DNN).

3.1 Boosted Decision Trees

A decision tree is a binary tree structure, as shown in Fig. 3.1. Starting from a root node, the variable separating signal and background the best is picked. Then a value is chosen that is used to split the sample. Events above and below that value are separated into two new nodes. This process is then repeated for the newly created nodes until a certain condition is met. These conditions, such as the amount of times a sample can be split or the miminum amount of events that a node must contain, are subject to optimisation. Afterwards, resulting nodes are designated as signal or background depending on the majority of event in each node, creating a signal-enriched and a background-enriched sample [19]. Currently, BDT is used as the MVA technique for this type of analysis.

To improve the classification strength of decision trees, boosting and bagging can be used. *Boosting* refers to generating many different decision trees, forming a forest, which are combined into a single classifier. Each instance draws from the same training sample, but keeps a weight to denote how much it influences the overall classifier [20]. *Bagging* means that each tree uses a different training set sampled from the overall data. In this analysis both boosting and bagging are used.

3.2 Artifical Neural Network

The most important components of the human brain are neurons, specialised cells that are combined into a vast network through connections called synapses and which are able to receive, process and transmit information in electric and chemical form, allowing the brain to far outperform any CPU in pattern recognition and giving it the ability to learn.

The goal of artifical neural networks (ANN) is to model these interconnected neurons giving computers some of this pattern recognition and learning ability, while simplifying it enough to allow modern computers to run them in a reasonable timeframe. ANNs consist of nodes, representing the neuron cells, that are arranged in layers. According to the McCulloch-Pitts perceptron model a node accepts inputs

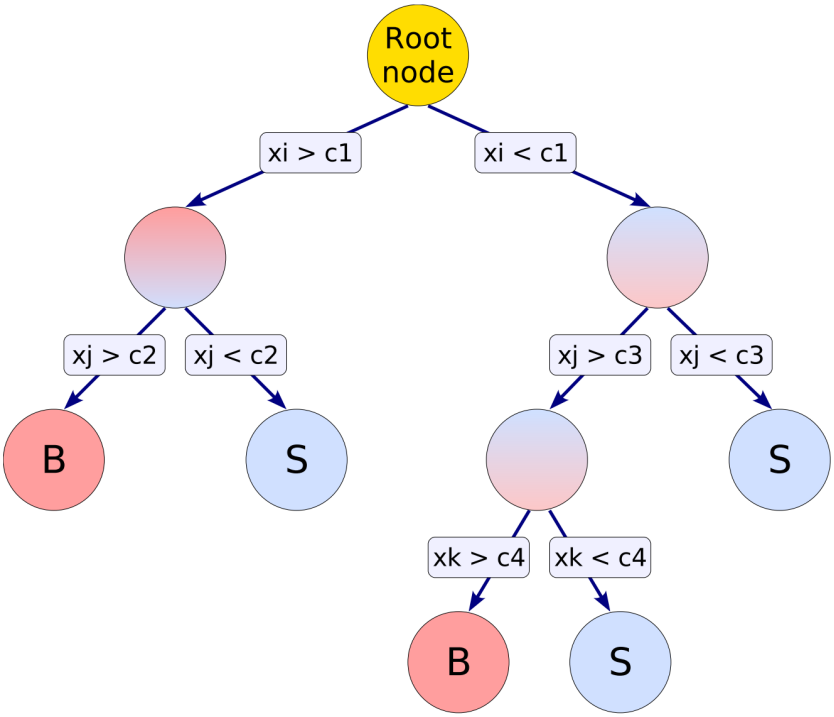


Figure 3.1: Illustration of a single decision tree [19].

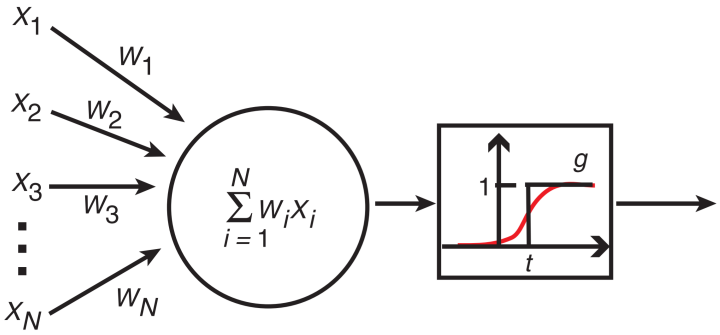


Figure 3.2: Illustration of a perceptron [22].

$X_1 \dots X_N$ each with a weight $W_1 \dots W_N$ and generates an output defined by an activation function, shown in Fig. 3.2. Classically, a step function is used, outputting 1 if a threshold t is reached, but other functions are also possible, such as sigmoid. [21]. The weights for each connection are determined in the learning process, called backpropagation, first they are set to small random numbers, then for each training input the output is compared to the desired output and an error is calculated. As more data is fed to the network, weights are adjusted with the goal of minimising the error. [22]. A major issue with NNs is overtraining. There may be too many parameters to be learned compared to the amount of training data, causing the network to mimick statistical fluctuations and reducing its performance for any data that is not part of the training set. This issue can be reduced by different methods, such as regularisation or Bayesian statistics.

The most common type of NN is the feed-forward neural network (FFNN), illustrated in Fig. 3.3, in which each node is connected directly to every other node in the next layer, with any layer that is not

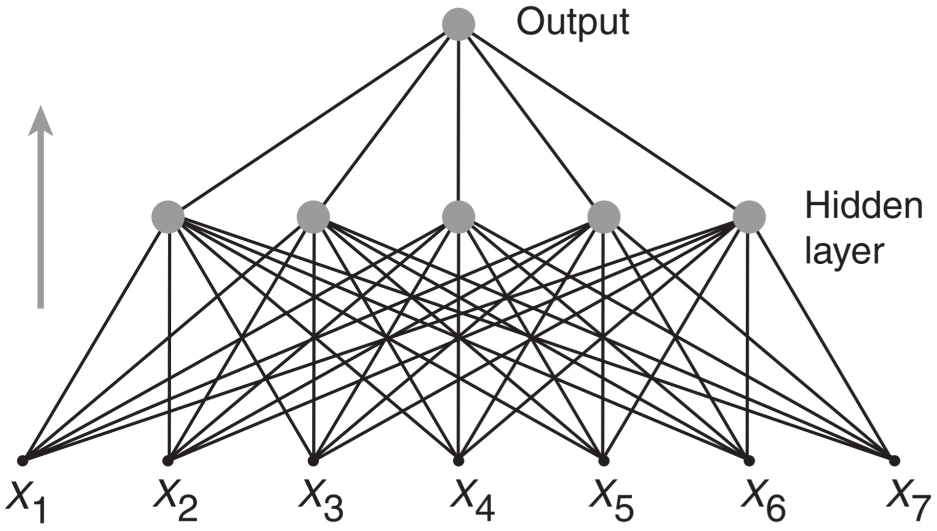


Figure 3.3: A two-layer feed-forward neural network. The input layer does not count as a layer as it does not perform any computation [22].

input or output being called a “hidden layer”.

3.2.1 NeuroBayes®

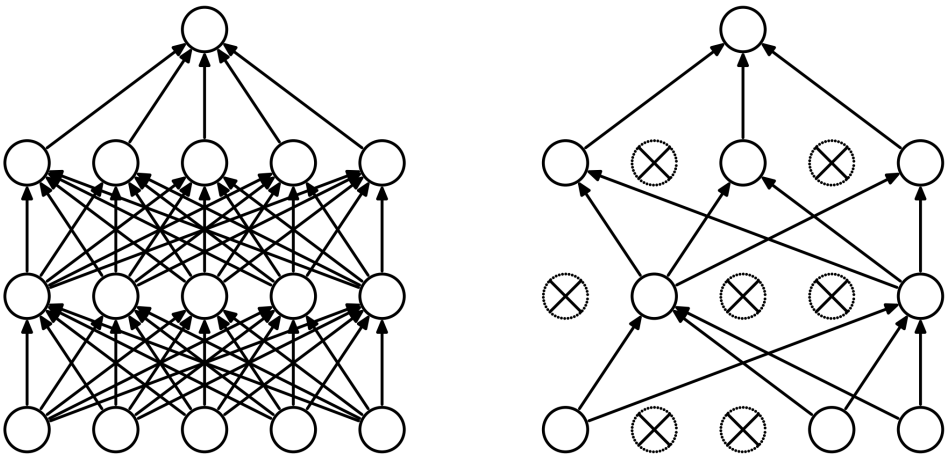
NeuroBayes is a multivariate analysis tool maintained by phi-t GmbH combining a single-layer FFNN with bayesian statistics. It was originally created for high energy physics and is still very commonly used in many analyses, however its use has also expanded into industries such as web, retail and financial companies. NeuroBayes takes all input variables and runs them through a preprocessing algorithm, choosing the best variables. Those are then used to train the single-hidden layer NN using the NeuroBayes-Teacher and evaluate it with training data. The weights are stored to be used for the analysis of real data with the NeuroBayes-Expert [23, 24].

3.2.2 Deep neural network with Keras

NeuroBayes only supports a single hidden node layer, so it can be classified as a “shallow” NN, however deep neural nets (DNNs) with more hidden layers have the potential to recognise more complex patterns and provide improved results compared to shallow ones. DNNs also have disadvantages, notably that they require more computing power (compared to a similarly sized shallow net). Most importantly however, they are very difficult to train such that they have only become viable in the last decade.

Keras [25] is an open-source neural network library, it functions as a wrapper around the popular machine learning libraries TensorFlow [26] and Theano [27]. The layout of the neural network is not fixed as is the case with NeuroBayes, instead the user can build whatever layout is best for the desired task in a modular way, choosing from a large number of possible layer types, such as the “classic” feed-forward layer, but also more advanced types like convolutional and pooling layers that are often used in image recognition tasks.

In this analysis a FFNN with two hidden layers will be used. Additionally, dropout is applied after each hidden layer, a type of regularisation that helps to prevent overfitting of the model by randomly dropping nodes from a layer, thereby allowing the use of a much higher amount of hidden nodes in the



(a) Standard Neural Net (b) After applying dropout.

Figure 3.4: Illustration of the deep neural network used in this analysis, on the left without dropout, on the right after applying dropout [28].

model. An illustration of the net used in this analysis including dropout is shown in Fig. 3.4 [28].

CHAPTER 4

MVA for the Wt Dilepton Channel

4.1 Analysis setup

Events are selected according to the decay products of the Wt dilepton channel. They must contain exactly two leptons of opposite charge ℓ_1 and ℓ_2 as well as one or two jets (j_1, j_2) that may be tagged as originating from a b quark. Wt events are separated from $t\bar{t}$ background with MVA techniques using two signal regions, 1j1b for overall one jet which is b-tagged and 2j1b with a b-tagged jet and an additional jet due to radiative processes. The 2j2b region is dominated by $t\bar{t}$ background and used to constrain the $t\bar{t}$ cross section.

This leads to the following variables, where $o_1, \ldots, o_n$ stands for one of the objects $\ell_1, \ell_2, j_1, j_2, E_{\mathrm{T}}^{\mathrm{miss}}$. [29]:

- $p_{\mathrm{T}}^{\mathrm{sys}}(o_1, \ldots, o_n)$, the magnitude of the vector sum of tranverse momenta;
- ΣE_{T}, the scalar sum of tranverse energies of objects contributing to the missing transverse energy $E_{\mathrm{T}}^{\mathrm{miss}}$;
- H_{T}, the scalar sum of tranverse momenta;
- $\sigma(p_{\mathrm{T}}^{\mathrm{sys}}(o_1, \ldots, o_n))$, the ratio of $p_{\mathrm{T}}^{\mathrm{sys}}(o_1, \ldots, o_n)$ to $\sqrt{H_{\mathrm{T}}(o_1, \ldots, o_n) + \Sigma E_{\mathrm{T}}(o_1, \ldots, o_n)}$;
- $\eta(o_1, \ldots, o_n)$, pseudorapidity of the system (see 2.2.2);
- $m(o_1, \ldots, o_n)$, the invariant mass of the system;
- $m_{\mathrm{T}}(o_1, \ldots, o_n)$, the tranverse mass of the system;
- $E/m(o_1, \ldots, o_n)$, the ratio of energy to invariant mass of the system.

Additionally, some variables comparing two systems of objects s_1 and s_2 are considered:

- $\Delta p_{\mathrm{T}}(s_1, s_2)$, the p_{T} difference between the systems;
- $\Delta R(s_1, s_2)$, the distance between the systems in $\phi - \eta$ space;
- $C(s_1, s_2)$, the centrality, defined as the ratio of the scalar sum of the p_{T} of the systems and the sum of their energies.

Variable selection is done separately for each method. However care is being taken to keep the final amount of variables similar and reasonably low as otherwise a proper comparison becomes unfair. The Wt and $t\bar{t}$ data used to train the methods is simulated using Monte-Carlo (MC) techniques.

4.2 Performance metrics

To retain the ability to compare different MVA techniques, the same performance metrics have to be used to score them. This requirement leads to two metrics that will be used here: Separation and ROC integral.

4.2.1 Separation

Separation describes how much signal and background distributions are separated from each other. It is defined as the following integral:

$$\langle S^2 \rangle = \frac{1}{2} \int \frac{(\hat{y}_S(y) - \hat{y}_B(y))^2}{\hat{y}_S(y) + \hat{y}_B(y)} \, \mathrm{d}\, y$$

where $\hat{y}_S$ and $\hat{y}_B$ are the signal and background distribution functions of y, respectively, with y being the classifier. Identical distributions have a separation of 0 while no overlap is a separation of 1 [19].

4.2.2 ROC integral

A receiver operating characteristic (ROC) curve describes the relation between the probability to correctly discard a background event (background rejection) and the probability to correctly predict a signal event (signal efficiency). The integral under the ROC curve can be used to quantify MVA performance, an integral of 0.5 would describe random distributions while 1 would be perfect efficiency [30].

4.3 Boosted Decision Trees

4.3.1 Implementation

The implementation of the BDT used here is part of the TMVA package [19], a toolkit for multivariate analysis that is part of ROOT, a powerful data analysis framework widely used in high energy physics [31]. Some simple C++ scripts interfacing ROOT and TMVA are enough to do the training, analysis and visualisation.

4.3.2 Variable selection

First all variables are ranked by their separation power $\langle S^2 \rangle$. Then all highly correlated variables (>70% linear correlation for the 1j1b region, >80% for other regions) are cut out from the list, keeping the variable with better separation power. To get the final variable list, training is done multiple times, adding a single variable to the list each time and including it in the final list if separation power gained is at least 1 % compared to the maximum achieved separation or 5 % compared to the separation power after the previous variables (1.2 % and 6 percent, respectively for 1j1b). This is done multiple times and the list that achieves the best result is picked. The final variable lists per region can be found in Table 4.1. Correlation matrices for the final variable lists are shown in figures B.1, B.2 and B.3, each for signal and background distributions.

4.3.3 Optimisation process

Training is optimised by maximising the achieved separation power while avoiding overtraining. A kolmogorov-smirnov (KS) test comparing test and training distributions is used to measure overtraining.

1j1b		2j1b		2j2b	
4.5	$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1)$	2.1	$\Delta p_{\mathrm{T}}(\ell_1\ell_2, j_2)$	4.0	$m(\ell_1 j_2)$
2.7	$\Delta p_{\mathrm{T}}(\ell_1\ell_2 j_1, E_{\mathrm{T}}^{\mathrm{miss}})$	1.5	$m(\ell_1 j_2)$	4.0	$\Delta p_{\mathrm{T}}(\ell_1\ell_2, j_2)$
2.7	$\Delta p_{\mathrm{T}}(\ell_1\ell_2, E_{\mathrm{T}}^{\mathrm{miss}} j_1)$	1.5	$\Delta R(\ell_1\ell_2, E_{\mathrm{T}}^{\mathrm{miss}} j_1 j_2)$	3.1	$p_{\mathrm{T}}^{\mathrm{sys}}(j_1 j_2)$
2.4	ΣE_{T}	1.4	$\Delta R(\ell_1\ell_2, j_1 j_2)$	2.6	$\Delta R(\ell_1\ell_2, j_1 j_2)$
1.8	$\Delta\phi(\ell_1\ell_2 j_1, E_{\mathrm{T}}^{\mathrm{miss}})$	1.1	$p_{\mathrm{T}}^{\mathrm{sys}}(E_{\mathrm{T}}^{\mathrm{miss}} j_1 j_2)$	2.5	$p_{\mathrm{T}}(j_2)$
1.3	$\eta(\ell_1\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1)$	1.0	$p_{\mathrm{T}}^{\mathrm{sys}}(j_1 j_2)$	2.1	$m(\ell_1\ell_2 j_1 j_2)$
1.1	$m(\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1)$	0.9	$p_{\mathrm{T}}(j_2)$	1.7	$\Delta p_{\mathrm{T}}(\ell_2, j_2)$
1.1	$\Delta p_{\mathrm{T}}(\ell_1\ell_2, E_{\mathrm{T}}^{\mathrm{miss}})$	0.8	$m(\ell_1 j_1 j_2)$	1.5	$m(\ell_1\ell_2 j_2)$
0.9	$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1 E_{\mathrm{T}}^{\mathrm{miss}} j_1)$	0.8	$\Delta\phi(\ell_1\ell_2, E_{\mathrm{T}}^{\mathrm{miss}} j_1 j_2)$	1.3	$m(\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1 j_2)$
0.8	$C(\ell_1\ell_2)$	0.6	$\sigma p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1 j_2)$	0.9	$\Delta R(\ell_2, j_2)$
0.7	$\Delta p_{\mathrm{T}}(\ell_1\ell_2, j_1)$	0.5	$\Delta R(\ell_1, j_2)$	0.5	$m(\ell_2 j_2)$
0.2	$m(\ell_2 j_1)$	0.4	$m(\ell_1\ell_2 j_2)$		
0.1	$m(\ell_1 j_1)$	0.4	$E/M(\ell_1\ell_2 j_1 j_2)$		
		0.3	$E/M(j_1 j_2)$		
		0.1	$m(\ell_2 j_2)$		

Table 4.1: Final variables chosen for BDT training by region. Corresponding separation power is shown to the left of each variable.

Region	nTrees	MaxDepth	Shrinkage	nCuts	MinNodeSize	BaggedSampleFraction
1j1b	1030	2	0.05	75	0.5%	0.35
2j1b	340	3	0.09	95	0.5%	0.45
2j2b	320	2	0.15	80	1.5%	0.15

Table 4.2: Best parameters found for BDT training per region.

Its value for signal and background is required to be >0.1 for a training run to be deemed not overtrained.

The following BDT parameters are optimised:

- **nTrees:** total amount of decision trees used in the forest.
- **MaxDepth:** maximum depth of a single decision tree.
- **Shrinkage:** learning rate of the BDT algorithm.
- **nCuts:** total number of steps during each node cut optimisation.
- **MinNodeSize:** minimum fraction of events required in a final node.
- **BaggedSampleFraction:** for bagging; fraction of events used in each tree.

Each parameter is optimised by repeating the training using different values. The best one that is not overtrained is picked. Due to their large impact on the final result the amount of trees and maximum depth are optimised first, subsequent parameters are then optimised one by one. The plots received in the optimisation process can be found in section B.2. The final selection of parameters can be seen in Table 4.2.

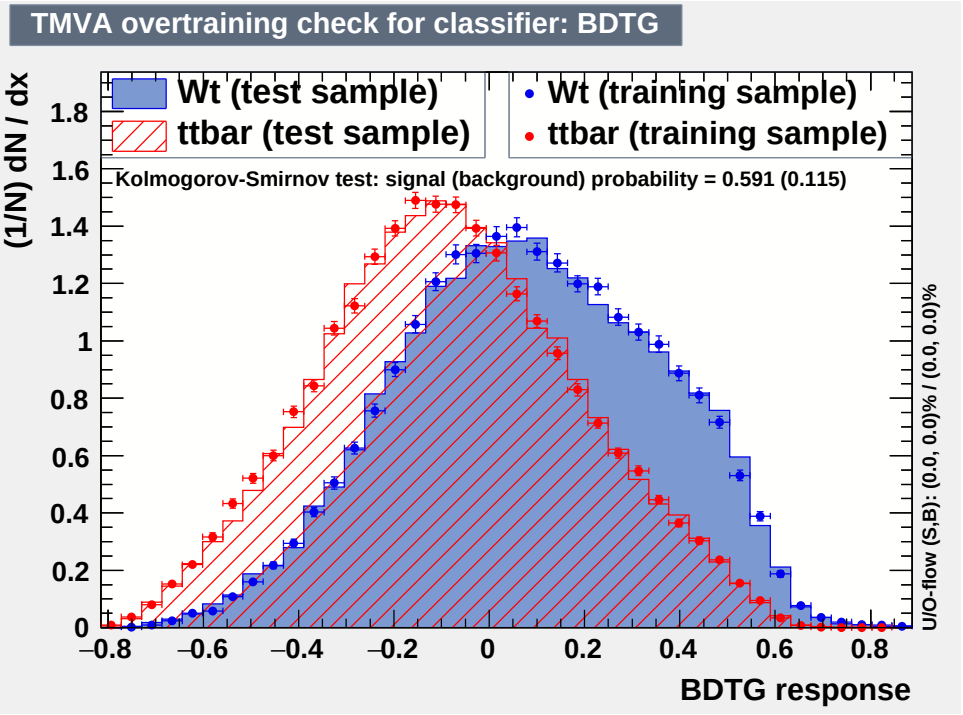


Figure 4.1: Probability distributions with overtraining check for BDT in 1j1b region

Region	Separation	ROC Integral
1j1b	0.0831	0.6626
2j1b	0.0852	0.6653
2j2b	0.1625	0.7249

Table 4.3: Performance achieved in BDT training.

4.3.4 Results

The achieved distributions, including KS test, are shown in Fig. 4.1 for region 1j1b. Distributions in the 2j1b and 2j2b regions can be seen in section A. The values for separation power and ROC integral are shown in Table 4.3. As expected, the signal regions yield similar results due to similar ratio of signal to background, while the validation region that is dominated by $t\bar{t}$ has stronger separation and a larger ROC integral.

4.4 NeuroBayes®

4.4.1 Implementation

Only a simple C++ script loading user options and variables is required to interface the NeuroBayes library. NeuroBayes is used in classification mode, with a single output node, and one node per variable with an additional bias node as input. A single hidden layer with an adjustable amount of nodes is used.

1j1b		2j1b		2j2b	
80.6	$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1)$	45.1	$m(\ell_1 j_2)$	43.2	$m(\ell_1 j_2)$
30.0	ΣE_{T}	42.4	$p_{\mathrm{T}}(j_2)$	41.7	$p_{\mathrm{T}}(j_2)$
35.5	$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2)$	35.7	$\Delta R(\ell_1\ell_2, j_1 j_2)$	36.0	$p_{\mathrm{T}}^{\mathrm{sys}}(j_1 j_2)$
29.2	$C(\ell_1\ell_2)$	32.4	$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1 j_2)$	26.8	$\Delta R(\ell_1\ell_2, j_1 j_2)$
27.2	$m_{\mathrm{T}}(j_1 E_{\mathrm{T}}^{\mathrm{miss}})$	32.0	$p_{\mathrm{T}}^{\mathrm{sys}}(j_1 j_2)$	13.8	$C(\ell_2 j_1 j_2)$
14.2	$\Delta R(\ell_1, j_1)$	16.3	$E/M(j_1 j_2)$	16.8	$y(\ell_1\ell_2 j_1 j_2)$
12.1	$m(\ell_1 j_1)$	21.6	$y(\ell_1\ell_2 j_1 j_2)$	10.4	$\Delta p_{\mathrm{T}}(\ell_1, j_2)$
10.0	$\eta(\ell_1\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1)$	17.2	$\Delta p_{\mathrm{T}}(\ell_1, j_2)$	14.3	$m(\ell_1\ell_2 j_2)$
10.6	$E/M(\ell_2 j_1)$	16.5	$m(\ell_1\ell_2 j_2)$	17.2	$m(\ell_2 j_2)$
9.7	$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2 j_1)$	15.4	$C(\ell_2 j_1 j_2)$	6.7	$\Delta p_{\mathrm{T}}(E_{\mathrm{T}}^{\mathrm{miss}}, j_2)$
7.7	$y(\ell_2)$	12.6	$H(j_1 j_2)$	7.6	$m(\ell_1\ell_2 j_1 j_2)$
7.1	$m(j_1)$	11.1	$m(\ell_2 j_2)$	7.2	$H_{\mathrm{T}}(j_1 j_2)$
7.3	$E(\ell_1\ell_2 j_1)$			9.1	$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1 E_{\mathrm{T}}^{\mathrm{miss}} j_2)$

Table 4.4: Final variables chosen for NeuroBayes training by region. Additional significance of each variable is shown on the left side.

4.4.2 Variable selection

All available variables are fed to NeuroBayes, the preprocessor then transforms them to a Gaussian distribution after which they are ranked by additional statistical significance according to the following procedure:

The correlation matrix is calculated, then one variable at a time is removed and correlation recomputed. After this is done for all variables, they are sorted by the loss of correlation caused by them being discarded and the one causing the smallest loss is removed from the set. The statistical significance of each variable is the loss of correlation they caused when being removed, multiplied by the square root of the sample size $\sqrt{n}$. Lastly the variables are decorrelated and if pairs of highly correlated variables are found the less significant one is removed from the list. Afterwards, a sigma cut can be defined such that all variables below this cut a discarded from training. The highest possible cut within NeuroBayes is 4.5σ, which picks up to 32 variables for training. This is too much for fair comparison, so cuts are manually tightened to 7σ, 8σ and 6.5σ for 1j1b, 2j1b and 2j2b regions, respectively to receive an amount of variables similar to BDT. This decreases separation power by up to 9 % and ROC integral by up to 1 % in the signal regions. The final lists of variables for each region including the significance are shown in Table 4.4. Many variables already chosen for the BDT training can be found, such as $p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1)$, ΣE_{T} and $C(\ell_1\ell_2)$ in the 1j1b region. However several are different, note for example the complete lack of Δp_{T} in the 1j1b region, which were included several times in the BDT list.

4.4.3 Optimisation process

Again, optimisation is done by maximizing the separation and the ROC integral. Overtraining is judged by checking network purity, the amount of events that pass at a certain cut compared to what would be expected from the network output. This relation should have a linear dependency. The following parameters are available:

- **NODE2:** amount of hidden nodes in the network.
- **REG:** regularisation option (see section C)

Region	Hidden Nodes	Regularisation
1j1b	15	REG
2j1b	8	ALL
2j2b	27	ALL

Table 4.5: Number of hidden nodes and regularisation option used for each region in NeuroBayes training.

- **ITER:** number of training iterations, kept at 100.
- **LOSS:** type of loss function, it is recommended to keep this at "Entropy".
- **SHAPE:** shape treatment, kept at "off".
- **RTRAIN:** ratio of events used for training, kept at 0.5.
- **EPOCH:** number of events after which to update weights, kept at 10.
- **SPEED:** multiplicative factor to increase learning speed, kept at 1.0.
- **MAXLEARN:** multiplicative factor to limit global learning speed, kept at 0.001.

Of these parameters only the amount of hidden nodes and regularisation have a large enough impact to be considered for optimisation. Training is done by varying the amount of hidden nodes between 2 and 30 with each regularisation option. The best training that is not overfitted is picked. As can be seen in Fig. 4.2 the performance of the neural network is very stable, changes in ROC integral when varying parameters are small. The final set of parameters is shown in table 4.5

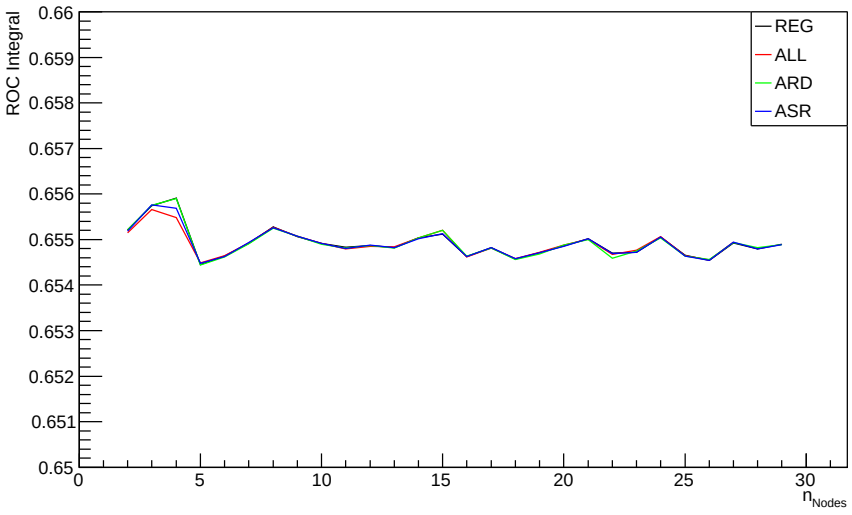


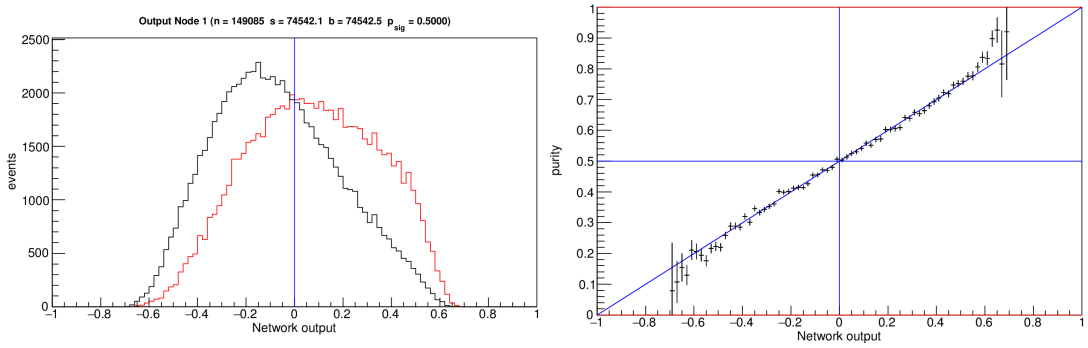
Figure 4.2: Impact of different parameters on the ROC integral in NeuroBayes for 1j1b region. For more regions see section B.4.

4.4.4 Results

Results for separation power and ROC curve integral can be found in Table 4.6. The probability distribution including overtraining for the 1j1b are shown in Fig. 4.3. The network is not overtrained as

Region	Separation	ROC Integral
1j1b	0.0745	0.6552
2j1b	0.0665	0.6463
2j2b	0.1432	0.7127

Table 4.6: NeuroBayes training performance for all regions.



(a) Distribution

(b) Purity

Figure 4.3: Distribution and network purity plots of NeuroBayes training in 1j1b region.

long as the purity stays close to the diagonal. Distributions for the 2j1b and 2j2b regions can be seen in figures A.3 and A.4. Again, the two signal regions produce similar results while the validation region is separated significantly stronger.

4.5 Deep neural network with Keras

4.5.1 Implementation

Training is done using the PyKeras method within pyMVA. pyMVA is a set of plugins for TMVA [19], allowing for python-based methods to be trained within the TMVA framework. The PyKeras method uses the Keras library for training, but allows the user to interface it in the same way as any other TMVA method. This includes reading data from and writing results to root files and generating plots. This way, work is reduced to a writing simple python script that both generates the model used by Keras and trains the network. Interpretation of results is done using another script making use of PyROOT [32], a python module that allows interfacing ROOT with python code.

4.5.2 Issues with training consistency

Several initial states in Keras, such as initial weights and dropped nodes, are determined randomly for every training, causing the results to vary slightly even when using identical settings. The best method to deal with this would be using k-fold cross-validation, a technique where the sample is split up into k subsamples and then trained on multiple times while varying which subsamples are used for training and testing. However, this wasn't available at the time of this thesis, so as a workaround, the results are averaged over multiple runs.

1j1b	2j1b	2j2b
$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1)$	$\Delta p_{\mathrm{T}}(\ell_1\ell_2, j_2)$	$m(\ell_1 j_2)$
ΣE_{T}	$m(\ell_1 j_2)$	$p_{\mathrm{T}}(j_2)$
$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2)$	$\Delta R(\ell_1\ell_2, E_{\mathrm{T}}^{\mathrm{miss}} j_1 j_2)$	$p_{\mathrm{T}}^{\mathrm{sys}}(j_1 j_2)$
$C(\ell_1\ell_2)$	$\Delta R(\ell_1\ell_2, j_1 j_2)$	$\Delta R(\ell_1\ell_2, j_1 j_2)$
$m_{\mathrm{T}}(j_1 E_{\mathrm{T}}^{\mathrm{miss}})$	$p_{\mathrm{T}}^{\mathrm{sys}}(E_{\mathrm{T}}^{\mathrm{miss}} j_1 j_2)$	$C(\ell_2 j_1 j_2)$
$\Delta R(\ell_1, j_1)$	$p_{\mathrm{T}}^{\mathrm{sys}}(j_1 j_2)$	$y(\ell_1\ell_2 j_1 j_2)$
$m(\ell_1 j_1)$	$p_{\mathrm{T}}(j_2)$	$\Delta p_{\mathrm{T}}(\ell_1, j_2)$
$\eta(\ell_1\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1)$	$m(\ell_1 j_1 j_2)$	$m(\ell_1\ell_2 j_2)$
$E/M(\ell_2 j_1)$	$\Delta\phi(\ell_1\ell_2, E_{\mathrm{T}}^{\mathrm{miss}} j_1 j_2)$	$m(\ell_2 j_2)$
$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2 j_1)$	$\sigma p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1\ell_2 E_{\mathrm{T}}^{\mathrm{miss}} j_1 j_2)$	$\Delta p_{\mathrm{T}}(E_{\mathrm{T}}^{\mathrm{miss}}, j_2)$
$y(\ell_2)$	$\Delta R(\ell_1, j_2)$	$m(\ell_1\ell_2 j_1 j_2)$
$m(j_1)$	$m(\ell_1\ell_2 j_2)$	$H_{\mathrm{T}}(j_1 j_2)$
	$E/M(\ell_1\ell_2 j_1 j_2)$	$p_{\mathrm{T}}^{\mathrm{sys}}(\ell_1 E_{\mathrm{T}}^{\mathrm{miss}} j_2)$
	$E/M(j_1 j_2)$	
	$m(\ell_2 j_2)$	

Table 4.7: Final variables chosen for Keras training

4.5.3 Variable selection

There is no standard way of selecting variables as there is with BDT and NeuroBayes and the process used in section 4.3.2 is not feasible in light of the immense computing power required, so the lists obtained previously and some minor variations are tested. The variables producing the best results are listed in 4.7. In the 1j1b region the NB list with the least significant variable removed is used, the 2j1b and 2j2b regions just use the BDT and NB lists, respectively. For each run half of the sample is used for training and the other half for testing.

4.5.4 Optimisation process

Even with the relatively simple three-layer FFNN used in this analysis many options are available, the most important ones are:

- **Nodes:** amount of hidden nodes (per hidden layer).
- **Dropout:** rate of dropout (per hidden layer).
- **NumEpochs:** number of epochs to train the model.
- **Activation:** activation function to be used (per hidden layer), default "relu" for both hidden layers.
- **Init:** initialiser to be used (per layer), default value is "glorot_normal".
- **Loss:** loss function to be used in the model.
- **Batch size:** number of events after which to update weights.

In order to get a decent approximation of actual performance while keeping computing time reasonable every training is repeated 10 times per parameter value and the results are averaged. For simplicity, any parameter that can be set per layer is kept constant across each layer. Overtraining is evaluated by

Region	Nodes	Dropout	NumEpochs	Loss	Batch size
1j1b	96	0.35	35	categorical_crossentropy	64
2j1b	140	0.45	60	categorical_crossentropy	32
2j2b	160	0.3	60	binary_crossentropy	16

Table 4.8: Best parameters found for Keras training per region.

Region	Separation	ROC Integral	Overtraining
1j1b	0.0811	0.6628	99.44 %
2j1b	0.0665	0.6463	99.07 %
2j2b	0.1997	0.7527	99.70 %

Table 4.9: Keras training performance for all regions.

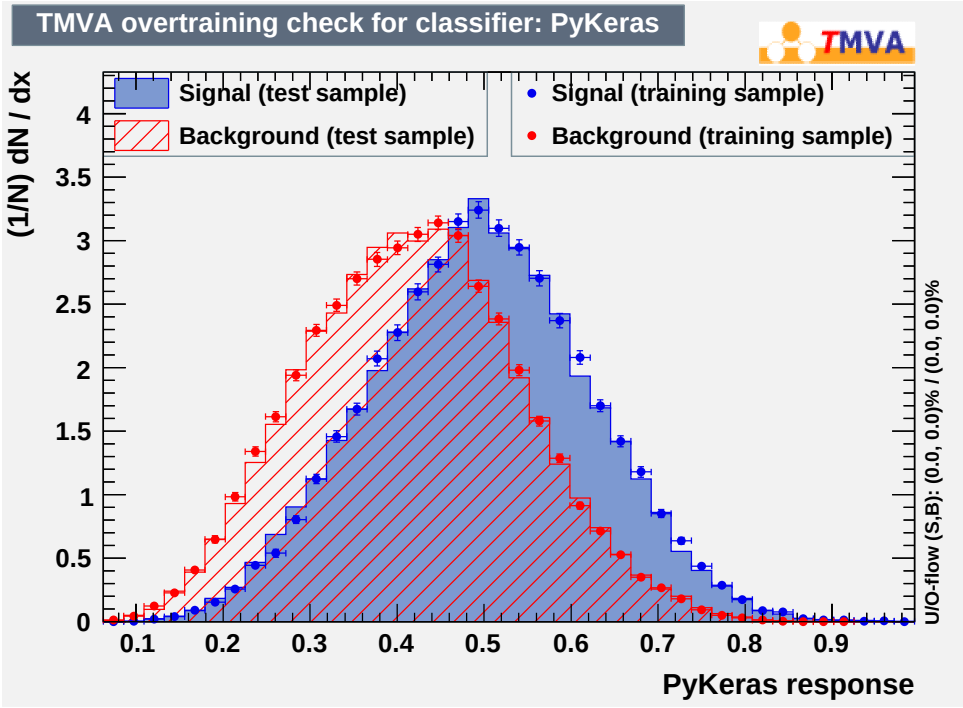


Figure 4.4: Achiveved distribution with Keras in the 1j1b region

comparing ROC curves for training and test samples, they have to agree to 99% at 30% background rejection in order for the run to be deemed not overfitted. The parameters yielding the best results are shown in Table 4.8

4.5.5 Results

Values of ROC integral and separation achieved are shown in Table 4.9. Distribution and ROC curve for each region can be seen in Fig. 4.4, A.5 and A.6. Note the distribution in the 2j2b region, it has an unusual shape, so these result will need further investigation. Again, the 2j2b region shows better results compared to the signal regions. Interestingly however, separation is noticeably stronger in the 1j1b region compared to the 2j1b region.

Due to the issues with variable selection and optimisation mentioned earlier, it is recommended to validate these results using k-fold cross-validation.

CHAPTER 5

Comparison of MVA Techniques

A comparison of the ROC curves obtained in the techniques tested can be found in figures 5.1, 5.2 and 5.3 for regions 1j1b, 2j1b and 2j2b, respectively. Separation and ROC integral values are shown in Table 5.1.

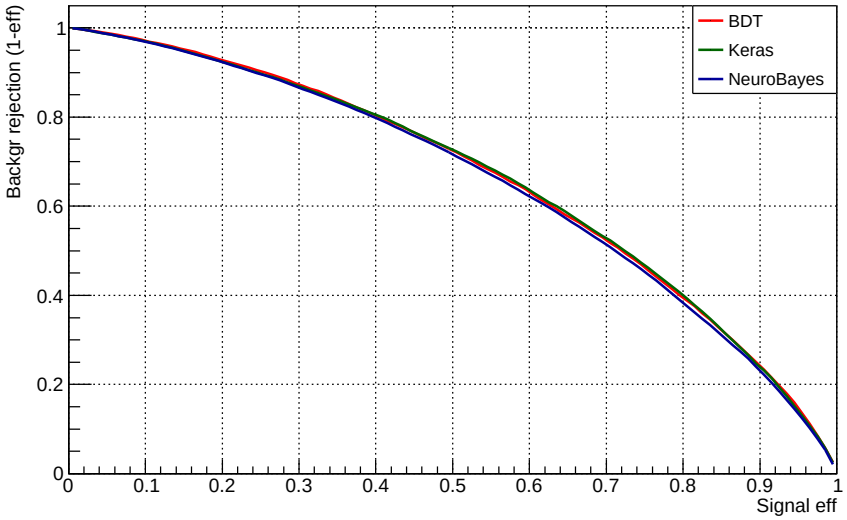


Figure 5.1: Comparison of ROC curves of BDT, NB and Keras methods in the 1j1b region.

Based on the obtained results, NeuroBayes appears to be the weakest technique. A possible explanation might be a different response to statistics, however since it cannot gain ground in the 2j2b region, this cannot be confirmed. While it is certainly the most straightforward to use, very stable and light on

Region	Separation			ROC Integral		
	BDT	NB	Keras	BDT	NB	Keras
1j1b	0.0831	0.0745	0.0811	0.6626	0.6552	0.6628
2j1b	0.0852	0.0665	0.0859	0.6653	0.6463	0.6664
2j2b	0.1625	0.1432	0.1997	0.7249	0.7127	0.7527

Table 5.1: Separation and ROC integral achieved for every method in each region.

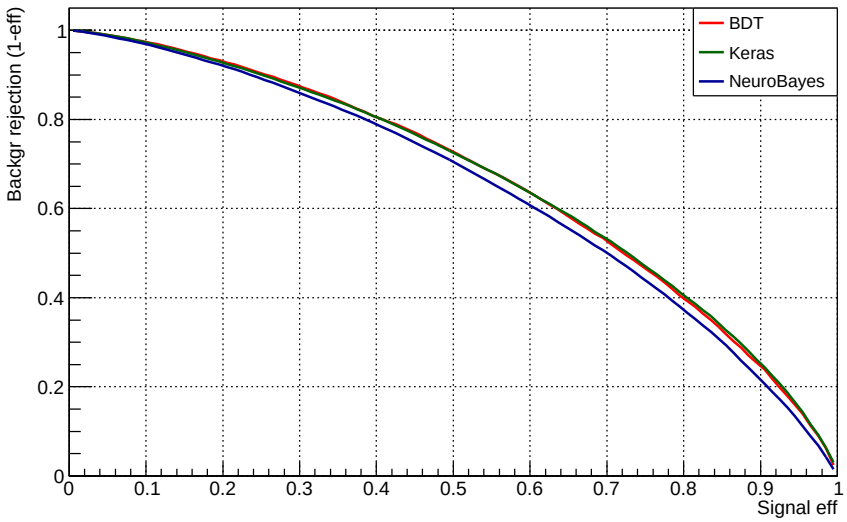


Figure 5.2: Comparison of ROC curves of BDT, NB and Keras methods in the 2j1b region.

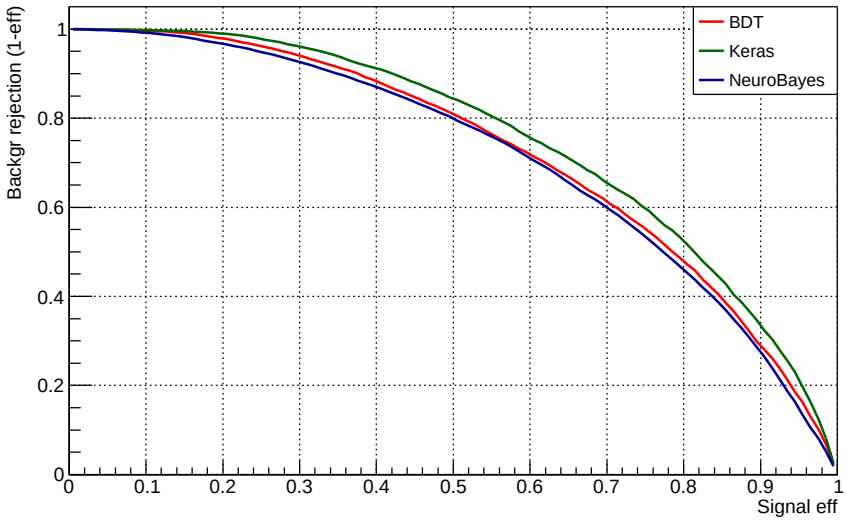


Figure 5.3: Comparison of ROC curves of BDT, NB and Keras methods in the 2j2b region.

computing resources, its use is not recommended in classifying *Wt* dilepton data.

BDT shows solid results, being neck and neck with Keras in the 1j1b and 2j1b signal regions, but weaker in the 2j2b validation region. However since it is much less complex compared to Keras and due to the validation region being less important than the two signal regions, it is still the best method to use.

Results with the deep neural network using Keras are very promising. It is a strong contender in the signal regions despite rough parameter optimisation and limited variable selection, refinement of which could further improve performance. Currently it does however have some issues. The training inconsistency causes massively increased computing resource requirements, lending the variable selection technique used in BDT infeasible and limiting optimisation abilities, while also making it more difficult to make a definitive statement about final performance. A possible solution for these issues is k-fold cross-validation, as described in section 4.5.2, so further exploration of Keras is encouraged.

CHAPTER 6

Conclusion

BDT, NeuroBayes and a deep neural network with Keras were used to separate signal and background in the *Wt* dilepton channel at ATLAS. For each technique the best variables were chosen and any parameters optimised. Performance was judged using separation power and the integral below the ROC curve. BDT showed noticeably higher performance than NB, however Keras has the potential to yield ever better results if the performance achieved in this comparison can be validated and further improved.

Bibliography

[1] *The Standard Model*, (2012), URL: http://cds.cern.ch/record/1997201 (cit. on p. 3).

[2] D. H. Perkins, *Introduction to High Eenergy Physics*, 4th ed., Addison-Wesley, 2000 (cit. on p. 3).

[3] MissMJ, *Standard Model of Elementary Particles*, [Online; accessed June 2017], 2006, URL: https://commons.wikimedia.org/wiki/File:Standard_Model_of_Elementary_Particles.svg (cit. on p. 4).

[4] C. Patrignani et al., *Review of Particle Physics*, Chin. Phys. **C40** (2016) 100001 (cit. on pp. 4, 6, 7).

[5] P. W. Higgs, *Broken Symmetries and the Masses of Gauge Bosons*, Phys. Rev. Lett. **13** (16 1964) 508 (cit. on p. 4).

[6] *The Large Hadron Collider*, (2014), URL: http://cds.cern.ch/record/1998498 (cit. on p. 5).

[7] *About the ATLAS experiment*, URL: https://atlas.cern/discover/about (cit. on p. 5).

[8] *CMS*, (2012), URL: http://cds.cern.ch/record/1997263 (cit. on p. 5).

[9] *ALICE*, (2012), URL: http://cds.cern.ch/record/1997265 (cit. on p. 5).

[10] *LHCb*, (2012), URL: http://cds.cern.ch/record/1997262 (cit. on p. 5).

[11] ATLAS Collaboration, ed., *ATLAS detector and physics performance: Technical Design Report, 1*, Technical Design Report ATLAS, 1999, URL: https://cds.cern.ch/record/391176 (cit. on p. 5).

[12] "LHC Guide", 2017, URL: http://cds.cern.ch/record/2255762 (cit. on p. 5).

[13] G. Aad et al., *The ATLAS Experiment at the CERN Large Hadron Collider*, JINST **3** (2008) S08003 (cit. on p. 5).

[14] F. Abe et al., *Observation of Top Quark Production in pp Collisions with the CDF Detector at Fermilab*, Phys. Rev. Lett. **74** (1995) 2626, arXiv: hep-ex/9503002 [hep-ex] (cit. on p. 6).

[15] S. Abachi et al., *Observation of the top quark*, Phys. Rev. Lett. **74** (1995) 2632, arXiv: hep-ex/9503003 [hep-ex] (cit. on p. 6).

[16] T. Loddenkötter, *Implementation of a kinematic fit of single top-quark production in association with a W boson and its application in a neural-network-based analysis in ATLAS*, BONN-IR-2012-06, PhD Thesis: University of Bonn, 2012, URL: http://hss.ulb.uni-bonn.de/diss_online (cit. on p. 7).

[17] M. Aliev et al., *HATHOR: HAdronic Top and Heavy quarks crOss section calculatoR*, Comput. Phys. Commun. **182** (2011) 1034, arXiv: 1007.1327 [hep-ph] (cit. on p. 7).

[18] S. Chatrchyan et al., *Identification of b-quark jets with the CMS experiment*, JINST **8** (2013) P04013, arXiv: 1211.4462 [hep-ex] (cit. on p. 8).

[19] A. Hoecker et al., *TMVA: Toolkit for Multivariate Data Analysis*, PoS **ACAT** (2007) 040, arXiv: physics/0703039 (cit. on pp. 9, 10, 14, 19).

[20] J. R. Quinlan, "Bagging, Boosting, and C4.S", *Proceedings of the Thirteenth National Conference on Artificial Intelligence - Volume 1*, AAAI'96, AAAI Press, 1996 725, ISBN: 0-262-51091-X, URL: http://dl.acm.org/citation.cfm?id=1892875.1892983 (cit. on p. 9).

[21] N. Siddique and H. Adeli, *Computational Intelligence: Synergies of Fuzzy Logic, Neural Networks and Evolutionary Computing*, John Wiley & Sons, 2013 (cit. on p. 10).

[22] A. Krogh, *What are artifical networks?*, Nat. Biotechnol. **26** (2008) 195 (cit. on pp. 10, 11).

[23] M. Feindt, *The NeuroBayes User's Guide*, (2010) (cit. on pp. 11, 45).

[24] M. Feindt and U. Kerzel, *The NeuroBayes Neural Network package*, URL: http://www-ekp.physik.uni-karlsruhe.de/~feindt/acat05-neurobayes (cit. on p. 11).

[25] *Keras*, URL: https://keras.io/ (cit. on p. 11).

[26] Google, A. Schumacher et al., *TensorFlow*, URL: https://www.tensorflow.org/ (cit. on p. 11).

[27] Theano Development Team, *Theano*, URL: http://deeplearning.net/software/theano/ (cit. on p. 11).

[28] N. Srivastava et al., *Dropout: A Simple Way to Prevent Neural Networks from Overfitting*, JMLR **14** (2014) 1929 (cit. on p. 12).

[29] I. Brock et al., *Measurement of the cross-section of the production of a W boson in association with a top quark with ATLAS at* $\sqrt{s} = 13\,\text{TeV}$, 2016 (cit. on p. 13).

[30] www.mathworks.com, *Detector Performance Analysis Using ROC Curves*, retrieved July 2017, URL: http://de.mathworks.com/help/phased/examples/detector-performance-analysis-using-roc-curves.html (cit. on p. 14).

[31] R. Brun and F. Rademakers, "ROOT - An Object Oriented Data Analysis Framework", vol. 389, AIHENP'96, 1996 81, URL: http://root.cern.ch/ (cit. on p. 14).

[32] W. Lavrijsen et al., *PyROOT*, URL: https://root.cern.ch/pyroot (cit. on p. 19).

APPENDIX A

Distributions for 2j1b and 2j2b Regions

The resulting distributions along with the corresponding overtraining check are shown here for regions 2j1b and 2j2b.

A.1 BDT distributions

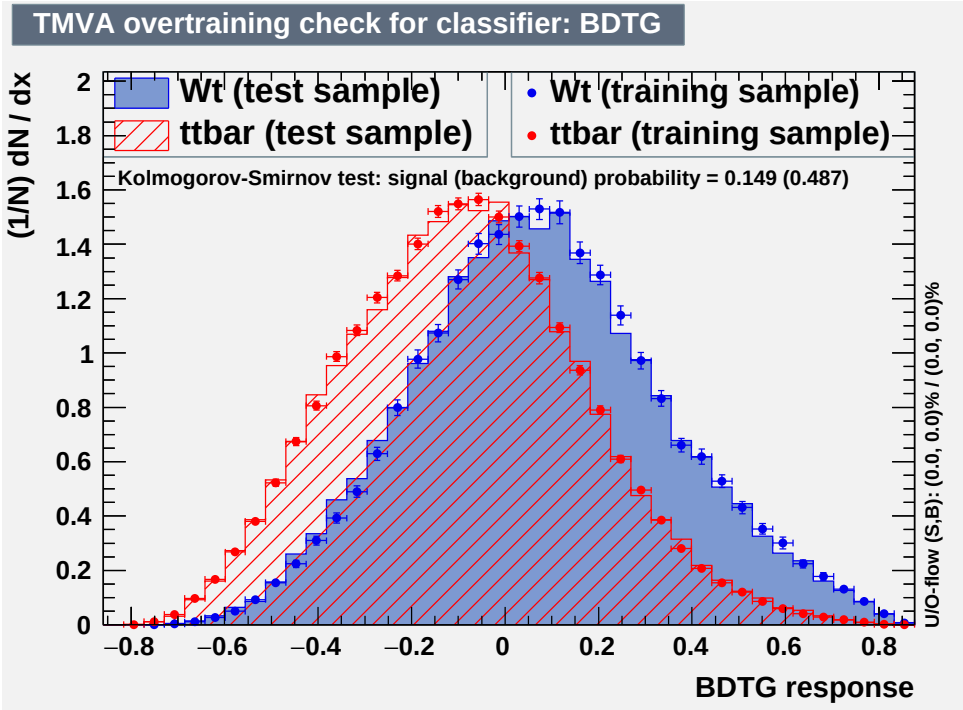


Figure A.1: Probability distributions with overtraining check for BDT in 2j1b region

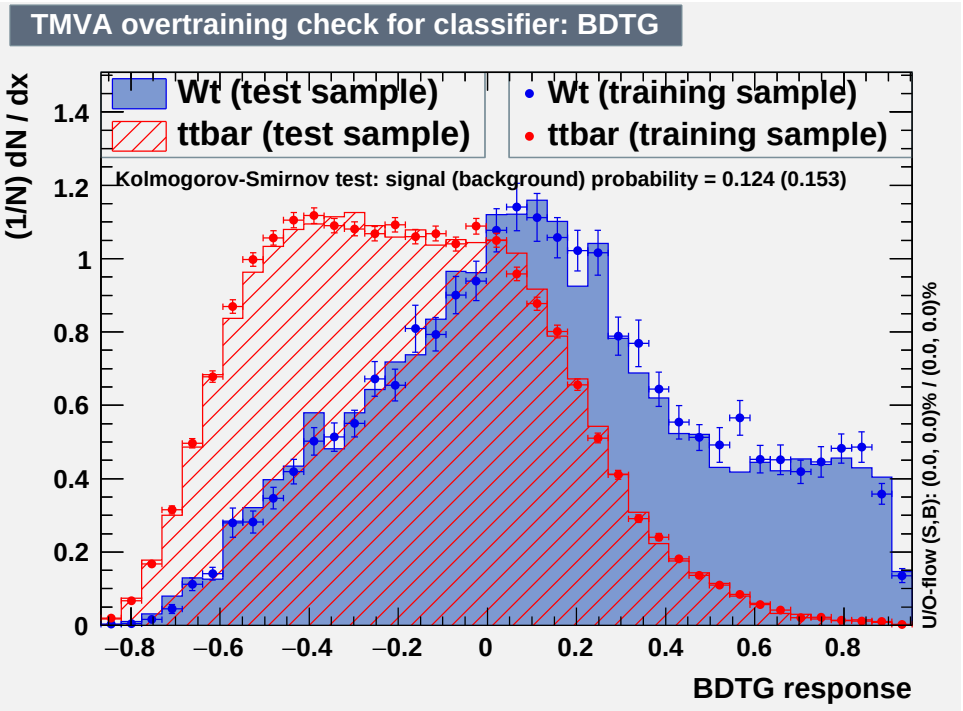
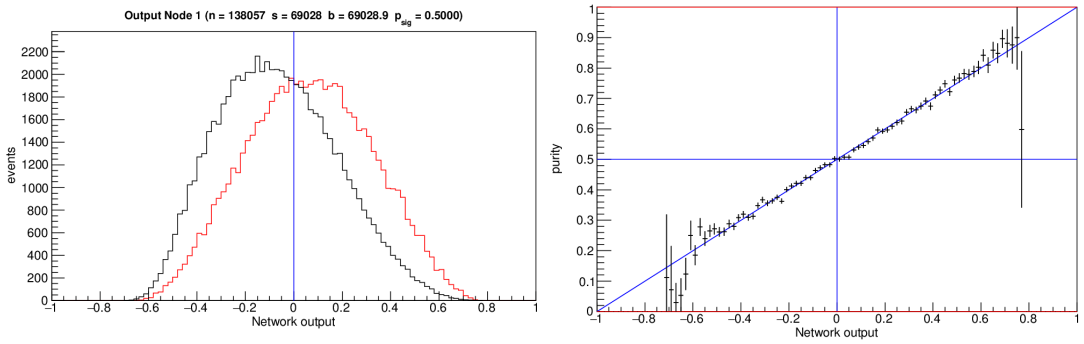


Figure A.2: Probability distributions with overtraining check for BDT in 2j2b region

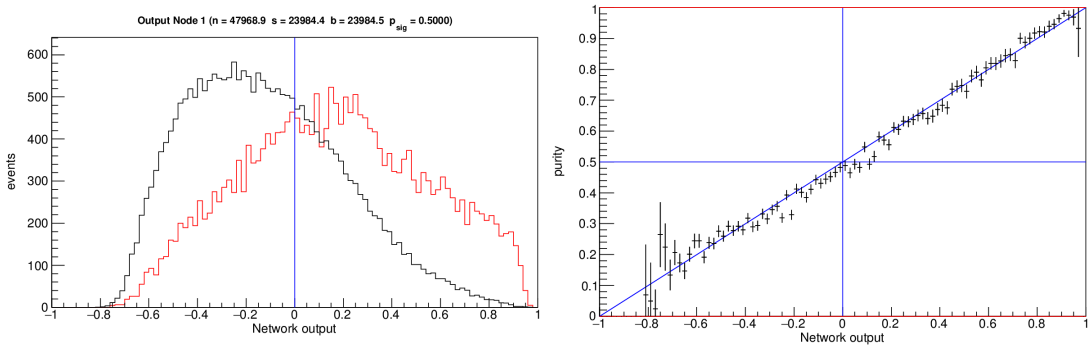
A.2 NeuroBayes distributions



(a) Distribution

(b) Purity

Figure A.3: Distribution and network purity plots of NeuroBayes training in 2j1b region. The red curve shows signal distribution, the back curve background distribution.



(a) Distribution

(b) Purity

Figure A.4: Distribution and network purity plots of NeuroBayes training in 2j2b region. The red curve shows signal distribution, the back curve background distribution.

A.3 Keras distributions

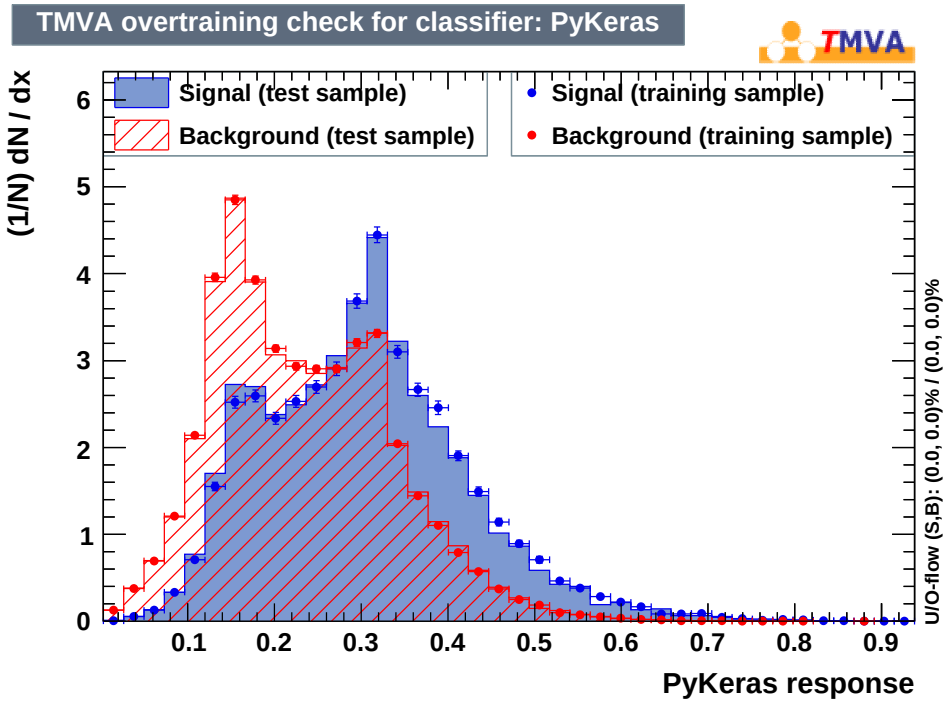


Figure A.5: Achiveved distribution with Keras in the 2j1b region.

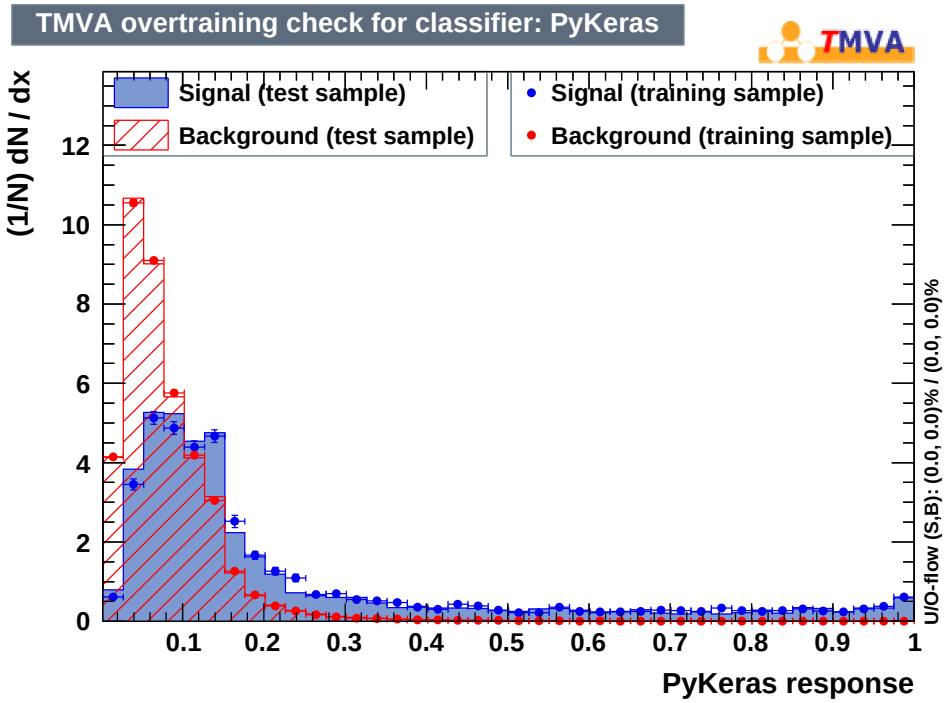


Figure A.6: Achiveved distribution with Keras in the 2j2b region.

APPENDIX B

Additional Plots

Correlation matrices for the BDT and NB techniques are shown here. Correlation for Keras variables is not shown as the lists are very similar, see B.9, excluding the last variable, for 1j1b, B.2 for 2j1b and B.11 for 2j2b regions. Additionally, optimisation plots for all parameters are shown for BDT, and plots for optimisation of hidden nodes and regularisation are shown for NB.

B.1 BDT correlation matrices

Correlation cut was 70 % for the 1j1b regions and 80 %. Several variables are still correlated strongly in the 2j1b and 2j2b regions, such as, expectedly, $m(\ell_1\ell_2 j_2)$ and $m(\ell_1\ell_2 j_1 j_2)$ in the 2j2b region. However, none improve training performance upon being removed

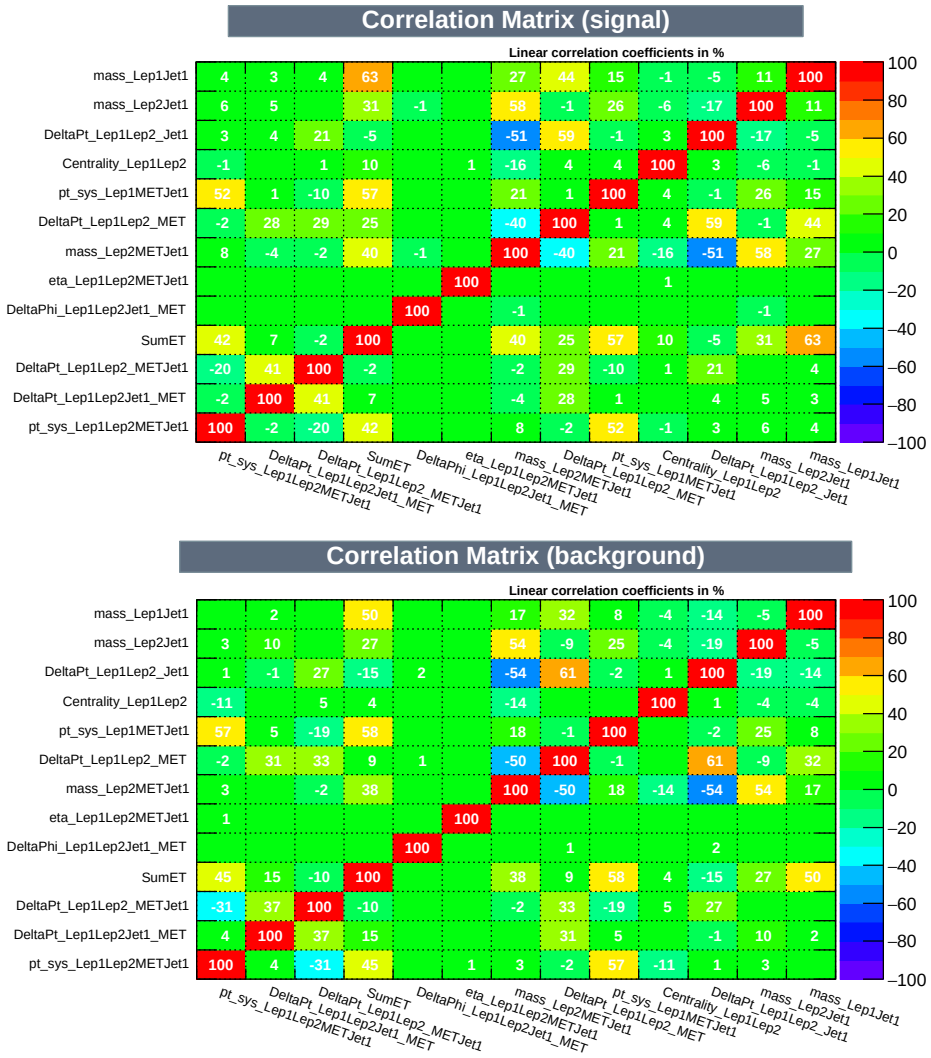


Figure B.1: BDT Correlation matrices for region 1j1b.

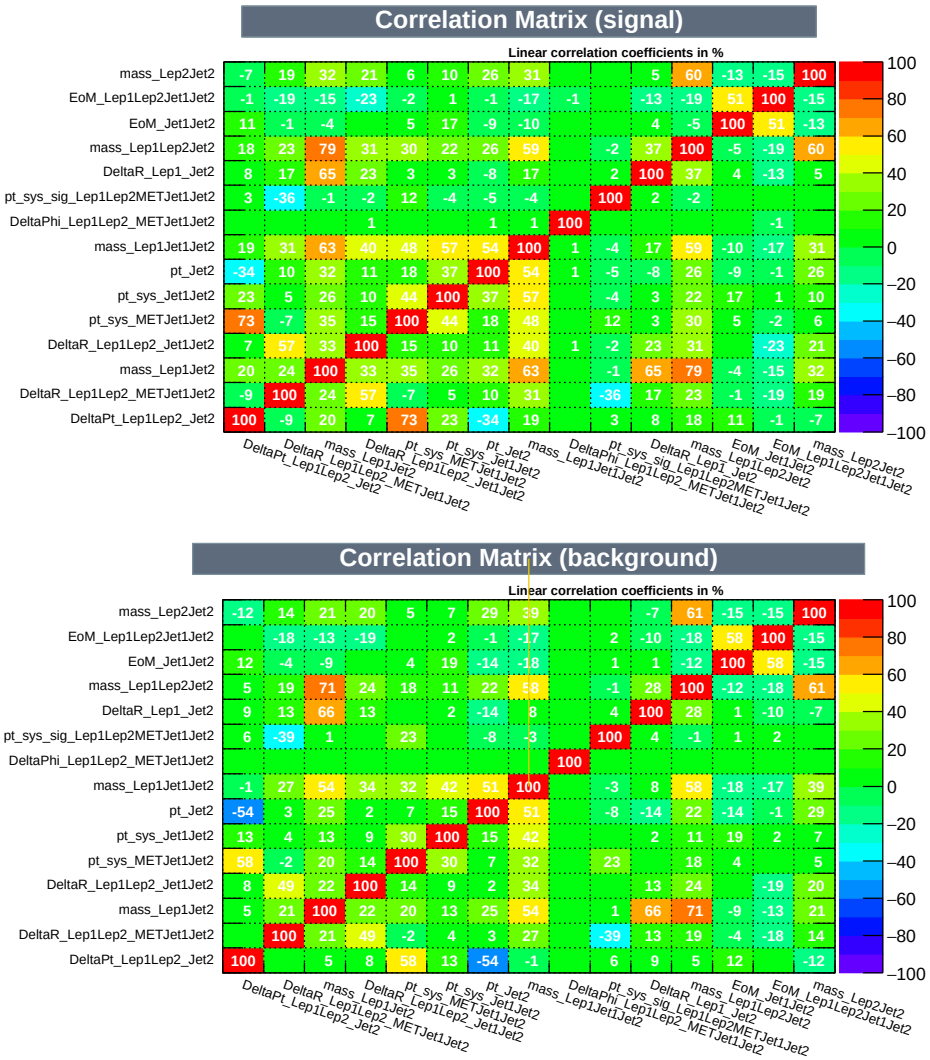


Figure B.2: BDT Correlation matrices for region 2j1b.

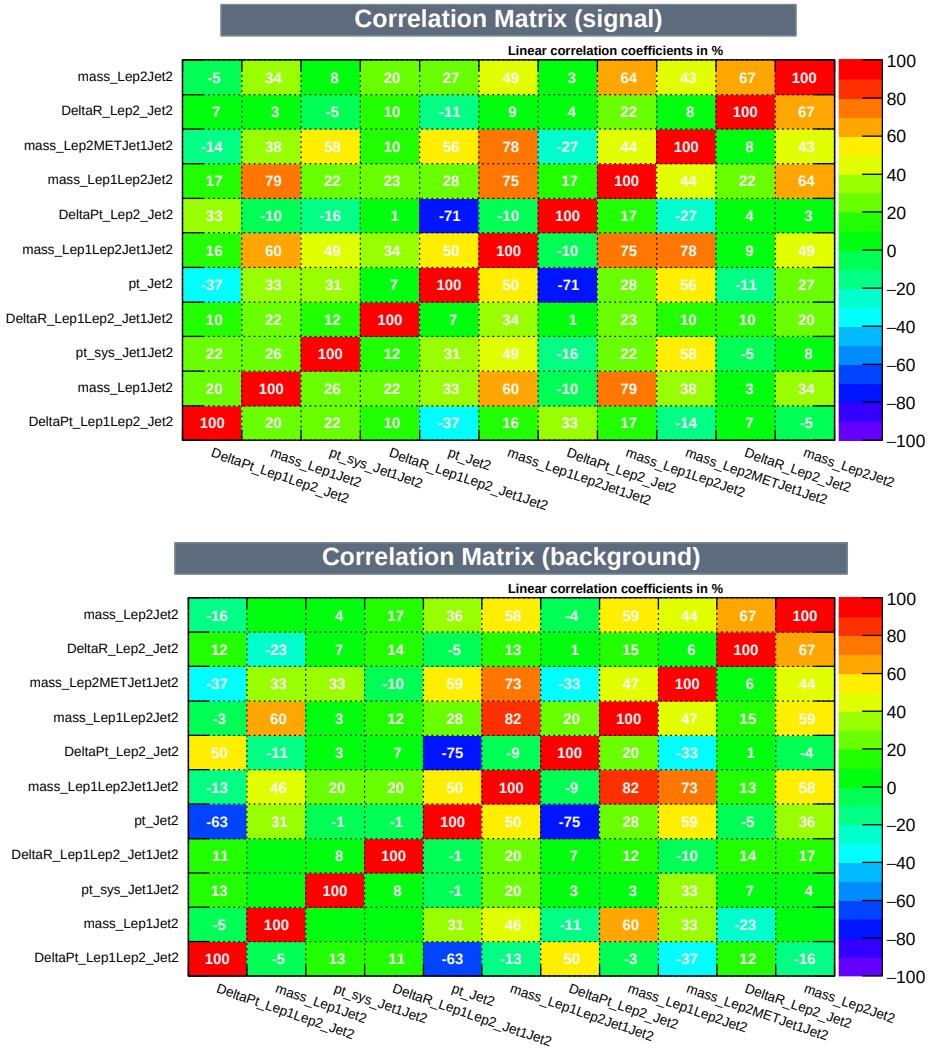
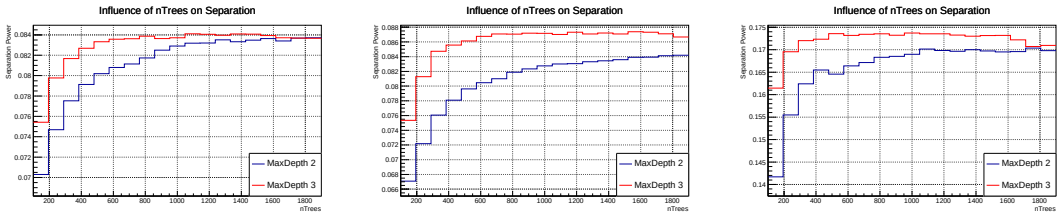


Figure B.3: BDT Correlation matrices for region 2j2b.

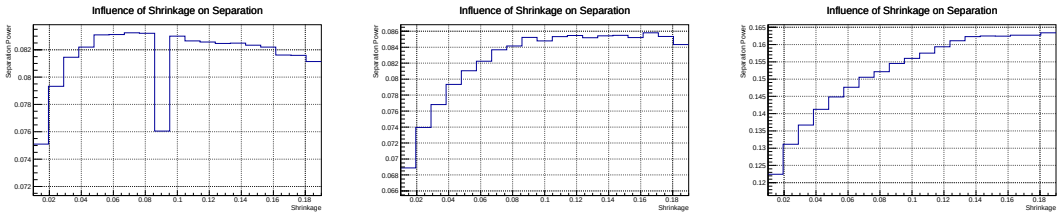
B.2 BDT optimisation plots

The way separation reacts to parameter adjustment in BDT is shown here. The amount of trees in the forest (nTrees) is optimised with multiple MaxDepth settings, as they have a very strong influence on each other. Note that the highest separation value does not necessarily set the best parameter for training as overtraining checks exclude some options.



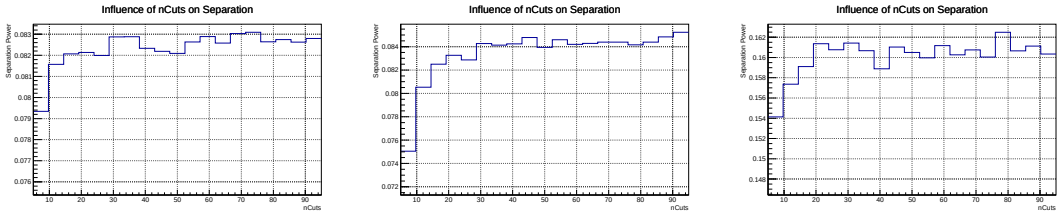
(a) 1j1b (b) 2j1b (c) 2j2b

Figure B.4: Influence of amount of trees and maximum depth on separation power in BDT.



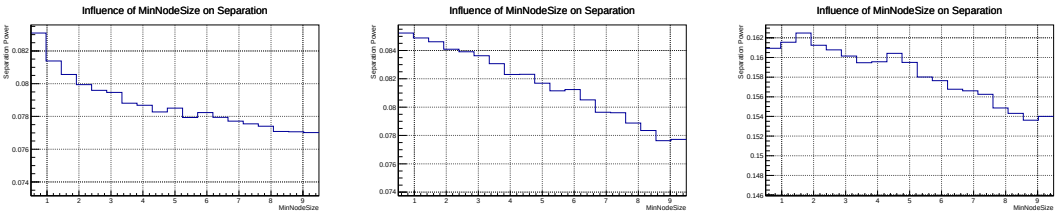
(a) 1j1b (b) 2j1b (c) 2j2b

Figure B.5: Influence of learning rate on separation power in BDT.



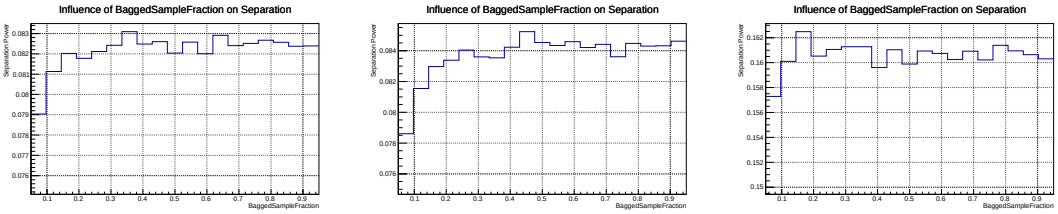
(a) 1j1b (b) 2j1b (c) 2j2b

Figure B.6: Influence of amount of cuts on separation power in BDT.



(a) 1j1b (b) 2j1b (c) 2j2b

Figure B.7: Influence of minimum node size on separation power in BDT.

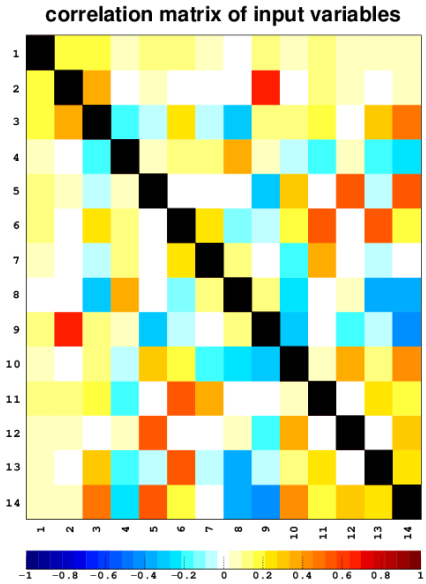


(a) 1j1b (b) 2j1b (c) 2j2b

Figure B.8: Influence of bagged sample fraction on separation power in BDT.

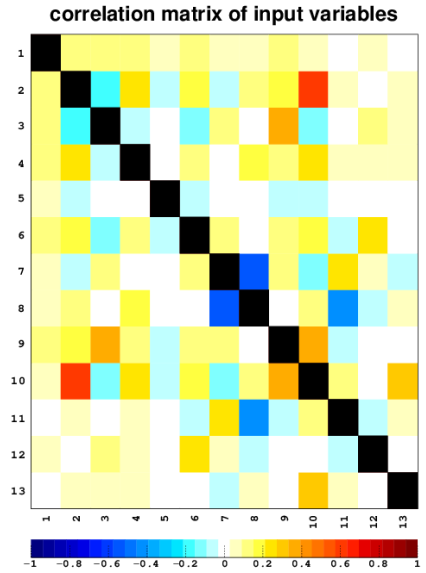
B.3 NeuroBayes correlation matrices

Several variables still show correlation in all regions, however, none improve training performance upon removal.



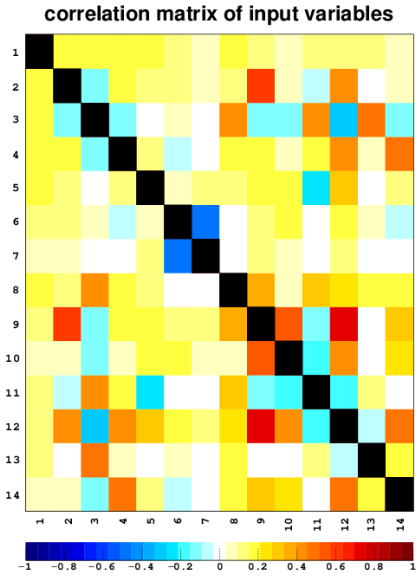
No.	Variable
1	Bias Node
2	$p_T^{\text{sys}}(\ell_1\ell_2 E_T^{\text{miss}} j_1)$
§	ΣE_T
4	$p_T^{\text{sys}}(\ell_1\ell_2)$
5	$C(\ell_1\ell_2)$
6	$m_T(j_1 E_T^{\text{miss}})$
7	$\Delta R(\ell_1, j_1)$
8	$m(\ell_1 j_1)$
9	$\eta(\ell_1\ell_2 E_T^{\text{miss}} j_1)$
10	$E/M(\ell_2 j_1)$
11	$p_T^{\text{sys}}(\ell_1\ell_2 j_1)$
12	$y(\ell_2)$
13	$m(j_1)$
14	$E(\ell_1\ell_2 j_1)$

Figure B.9: NeuroBayes correlation matrix and variables for region 1j1b.



No.	Variable
1	Bias Node
2	$m(\ell_1 j_2)$
3	$p_T(j_2)$
4	$\Delta R(\ell_1\ell_2, j_1 j_2)$
5	$p_T^{\text{sys}}(\ell_1\ell_2 E_T^{\text{miss}} j_1 j_2)$
6	$p_T^{\text{sys}}(j_1 j_2)$
7	$E/M(j_1 j_2)$
8	$y(\ell_1\ell_2 j_1 j_2)$
9	$\Delta p_T(\ell_1, j_2)$
10	$m(\ell_1\ell_2 j_2)$
11	$C(\ell_2 j_1 j_2)$
12	$H(j_1 j_2)$
13	$m(\ell_2 j_2)$

Figure B.10: NeuroBayes correlation matrix and variables for region 2j1b.



No.	Variable
1	Bias Node
2	$m(\ell_1 j_2)$
3	$p_T(j_2)$
4	$p_T^{\text{sys}}(j_1 j_2)$
5	$\Delta R(\ell_1\ell_2, j_1 j_2)$
6	$C(\ell_2 j_1 j_2)$
7	$y(\ell_1\ell_2 j_1 j_2)$
8	$\Delta p_T(\ell_1, j_2)$
9	$m(\ell_1\ell_2 j_2)$
10	$m(\ell_2 j_2)$
11	$\Delta p_T(E_T^{\text{miss}}, j_2)$
12	$m(\ell_1\ell_2 j_1 j_2)$
13	$H_{\text{T}}s(j_1 j_2)$
14	$p_T^{\text{sys}}(\ell_1 E_T^{\text{miss}} j_2)$

Figure B.11: NeuroBayes correlation matrix and variables for region 2j2b.

B.4 NeuroBayes optimisation plots

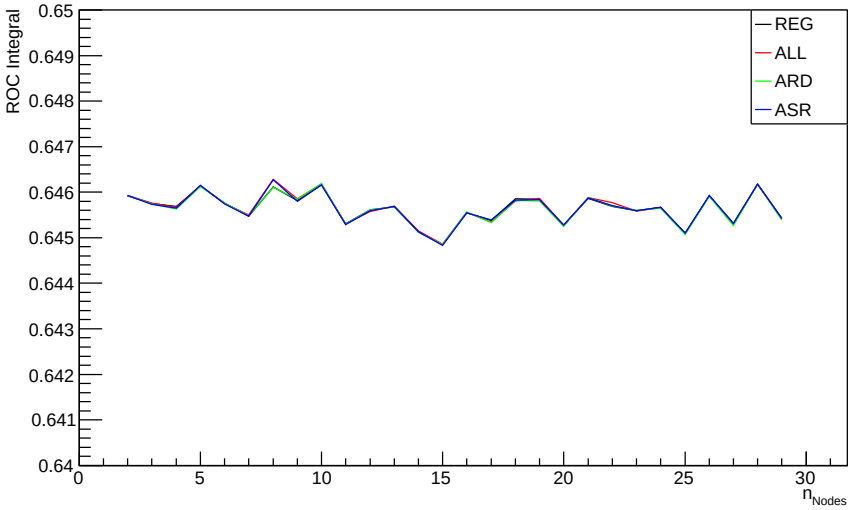


Figure B.12: Impact of optimisation on ROC integral for 2j1b region in NeuroBayes.

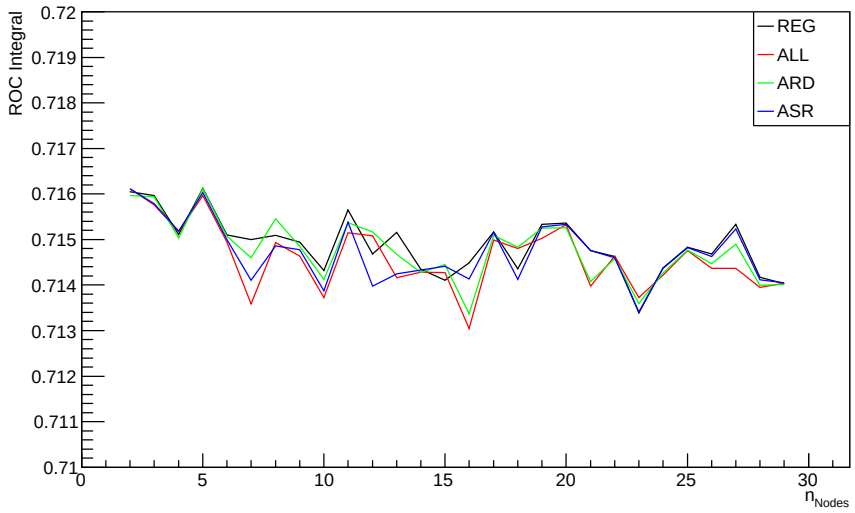


Figure B.13: Impact of optimisation on ROC integral for 2j2b region in NeuroBayes.

APPENDIX C

Regularisation Option in NeuroBayes

The following regularisation options are available in NeuroBayes:

- **OFF:** no regularisation.
- **REG:** Bayesian regularisation, weights are divided into classes: one for the bias node, one for all other input nodes and one for all hidden nodes.
- **ARD:** "Automatic Relevance Detection": all input nodes receive individual regularisation constants.
- **ASR:** "Automatic Shape Regularisation": all output nodes receive individual regularisation constants.
- **ALL:** both ARD and ASR are applied.

Taken from [23].

List of Figures

List of Tables