Use of Deep Learning techniques and Kinematic Fit with constraints in the search for the $H \to c\bar{c}$ channel at CMS

von:

Guillermo Rocamora Pérez

Masterarbeit in Physik

vorgelegt der

Fakultät für Mathematik, Informatik und Naturwissenschaften
der RWTH Aachen

im Oktober 2020

angefertigt im

Physics Institute III A

bei

Prof. Dr. Alexander Schmidt

Name of 1th examiner: Prof. Dr. Alexander Schmidt
Name of 2nd examiner: Prof. Dr. Martin Erdmann

Abstract

Two different analysis techniques to help improve the sensitivity of the direct search of the standard model Higgs boson decaying into a pair of charm quarks are presented. To reduce the amount of background in a direct $H \to c\bar{c}$ observation, the search is focused on events where the Higgs boson is produced in association with a Z or W boson, also called vector bosons. A total of five distinct final states are considered: $ZH \to \nu\nu c\bar{c}$, $WH \to e\nu c\bar{c}$, $WH \to \mu\nu c\bar{c}$, $ZH \to eec\bar{c}$, $ZH \to \mu\mu c\bar{c}$, divided into three different channels with 0, 1 or 2 charged leptons from the vector boson decay.

A deep neural network is trained to perform a classification task to discriminate signal from background events. Its results are compared to the current machine learning method implemented in the analysis, a boosted decision tree with gradient boost. In each channel of the analysis different kinematic variables are used as the input, and different models are considered and tested for each channel. The area under the ROC curve shows a marginal improvement in each channel when a deep neural network is employed compared against a boosted decision tree.

A least square kinematic fit is implemented in the $ZH \to l^+l^-c\bar{c}$ channel in order to achieve a better $m_{c\bar{c}}$ resolution. The low resolution in the c-jets momentum reconstruction can be improved by forcing some physical constraints on them. Conservation of momentum in the transverse plane and the assumption that the invariant mass of the two leptons should be equal to the mass of the Z boson are the constraints exploited in this method. Up to a 30% improvement in the mass resolution of the reconstructed Higgs boson was achieved by using this technique.

In loving memory of my father

Contents

Chapter 1: Introduction

The Standard Model of particle physics represents humanity's best understanding of the universe at a subatomic scale. It is the unifying framework that incorporates three of the four fundamental forces in nature (weak, electromagnetic and strong forces) along with their force carrying particles: the massless photons and gluons, and massive Z and W bosons. By the year 2000, all 24 fundamental fermions (12 particles and their associated antiparticles) and all gauge bosons, had been found. However, the Standard Model predicted the existence of one last boson: the Higgs Boson.

It wouldn't be until 2012 when the last component of the Standard Model was found. Physicists from the Large Hadron Collider at CERN announced the discovery of a new boson. It was observed for its decay into two photons or two Z bosons by the ATLAS and CMS experiments [1, 2]. Later observations confirmed this new particle to be the Higgs boson, proposed almost 50 years before its discovery. It has been observed to also decay into WW [3], $\tau\tau$ [4], and to both third-generation quarks $t\bar{t}$ [5, 6] and $b\bar{b}$ [7, 8].

As my Master's Project, I joined a team researching the Higgs boson coupling to second-generation quarks ($c\bar{c}$) to upgrade and propose new tools or techniques to improve their current analysis. My two main subjects of study were: Deep Neural Networks (DNN) and Kinematic Fitting.

Due to the quantum nature of both the collisions in a particular experiment and the interaction between the products of the collision and the detector, the resulting observations are fundamentally probabilistic. This probabilistic nature makes a classical approach to classification tasks challenging. Machine learning provides the simplicity and generalization required to achieve the desired accuracy in a suitable amount of time [9]. A boosted decision tree is the chosen machine learning method implemented by the analysis. As part of my research, a deep neural network was tested since, in theory, it had the ability to discover high level abstract features of the data.

The hadronization of quarks produced in a pp collision at LHC makes their reconstruction difficult, leading to a worse resolution of the Higgs mass, determined by the invariant mass of the two c-jets. This is were Kinematic Fitting could be useful. As defined by Paul Avery [10]: "Kinematic fitting is a mathematical procedure in which one uses the physical laws governing a particle interaction or decay to improve the measurements of the process." The physical law exploited is the conservation of momentum, which we will use in our advantage to correct the c-jet momentum measurements and thus, improve the Higgs boson mass resolution. The conservation of momentum is only applicable in the transverse plane with respect to the particle beam.

This dissertation will focus in the search for $H \to c\bar{c}$ at LHC, where the Higgs bosons are produced in association with a vector boson, W or Z.

Chapter 2: Technical Introduction

2.1 LHC

The Large Hadron Collider (LHC) is currently the world's largest particle collider [11]. It lies in a 26.7 km long tunnel buried up to 170 m underground located at CERN, near Genova, Switzerland. Their first collision was achieved in 2010 with an energy of 3.5 TeV per beam, for a total collision energy of $\sqrt{s} = 7$ TeV. In 2015, the LHC received a mayor upgrade to achieve 6.5 TeV per beam. The upgrades lead to a proton-proton (pp) collision with a combined energy of $\sqrt{s} = 13$ TeV.

The pp collisions take place in 4 points along the LHC, where the main experiments are located: CMS, ATLAS, ALICE and LHCb. The main purpose of the LHC was to test predictions from different theories of particle physics, including the search for the Higgs boson and different particles predicted by supersymmetric theories.

2.2 The CMS detector

The Compact Muon Solenoid (CMS) is a general-purpose detector at LHC. It sits at one of the four collision points, designed to observe and investigate any new physics phenomena that the LHC might reveal. The CMS consists in different layers on top of each other, each one fulfilling different functions [12]:

- **The Magnet:** The central feature of the CMS is a superconducting solenoid capable of generating a magnetic field up to 4 Teslas with the purpose of bending the charged particles originated in the collision, which allows to measure their charge and momentum. Within the solenoid, the tracker and the calorimeters are placed.
- **The Tracker:** A silicon pixel and strip tracker, which contains 65 millions pixels, allows to obtain the path of particles with extreme accuracy. Being the closest to the interaction point, it is the most relevant part to reconstruct the short-lived particles.
- **The Calorimeters:** Two different calorimeters are located inside the solenoid magnet:
  - **Electromagnetic Calorimeter (ECAL):** Designed to measure the energies of electrons and photons, consists in nearly 80000 crystals of lead tungstate that scintillate when electrons and photons pass through it. With the help of photodetectors, the light produced by these crystals is measured to obtain the energy of the particles.

– **Hadronic Calorimeter (HCAL):** It is a sampling calorimeter that measures the energy of hadrons and that provides indirect measurements of the presence of non-interacting, uncharged particles such as neutrinos.

- **The Muon Detector:** Detecting muons is one of the core features of CMS. Since they are not stopped by any of the CMS's calorimeters, special chambers to detect them are placed in the outer layer of the detector. With 1400 muons chambers, the detector is able to reconstruct the path and momentum of muons by tracking their position through the multiple layers (it consist of 4 layers). The momentum can be reconstructed because of the powerful magnet, capable of bending these high-energy muons.

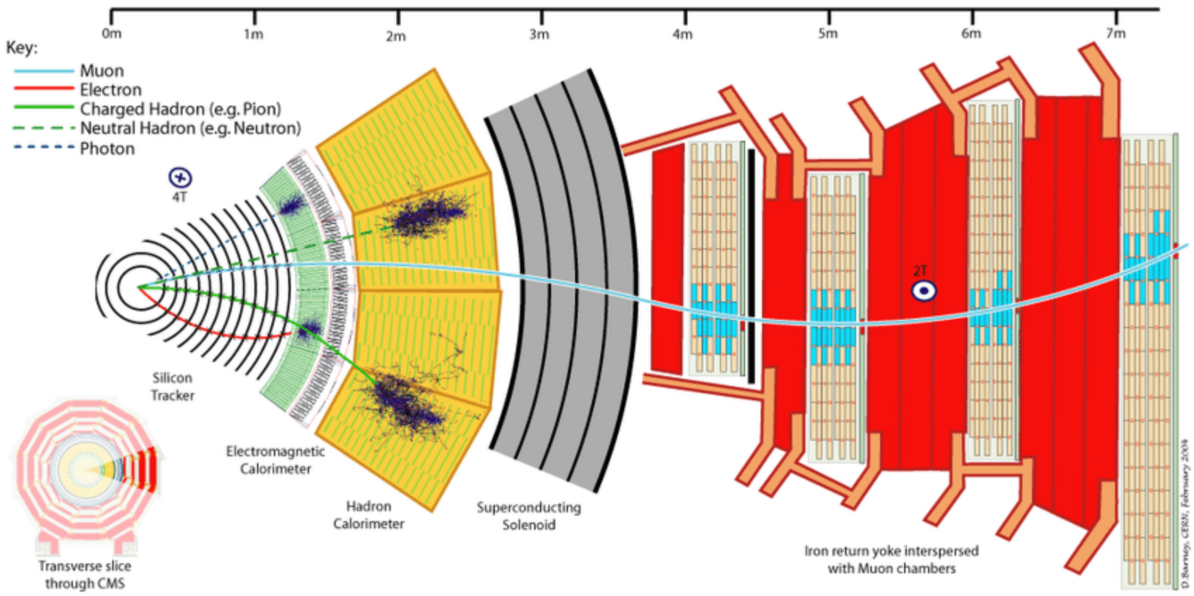


Figure 2.1: Cross-sectional view of the CMS detector [13].

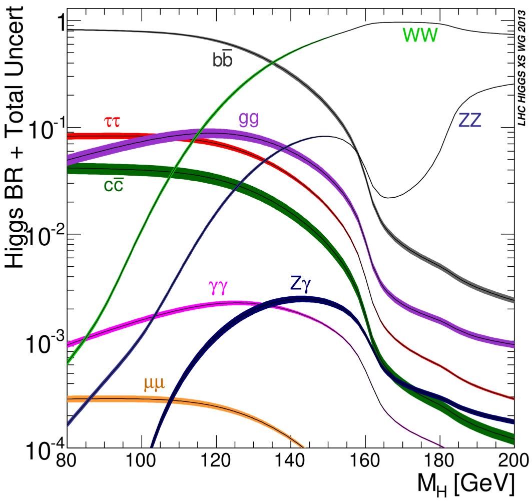
The expected collision frequency of 40 MHz requires a two-tiered trigger [14], composed of custom hardware processors, that selects events at a rate of 100 kHz. After this trigger, a second level, known as high-level trigger (HLT), reduces the event rate to around 1 KHz by running event reconstruction in ordinary processing servers.

The coordinate system adopted by CMS is centered at the nominal collision point. The y-axis points vertically upwards and the x-axis points towards the center of the LHC. The z-axis points along the beam direction. In the x-y-plane, the azimutal angle ϕ is measured from the x-axis and the radial coordinate is denoted by r. The polar angle θ is measured from the z-axis and the pseudorapidity is defined as $\eta = -ln tan(\theta/2)$. The transverse components p_T and E_T are computed from the x and y components.

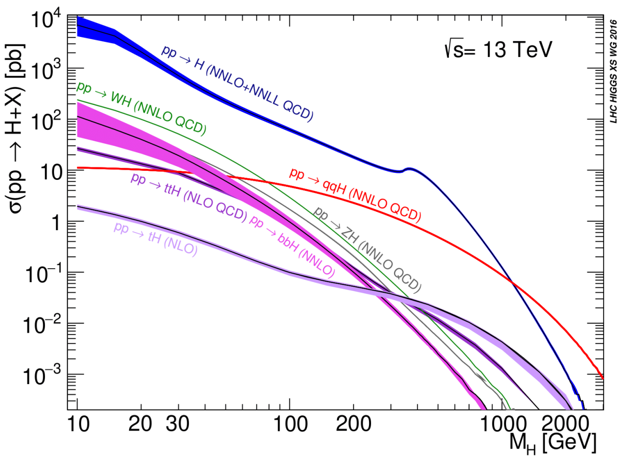
2.3 Search for the Higgs Boson

The Higgs mechanism is an essential piece of the Standard Model that explains the property of mass for gauge bosons. Before its theorization, W and Z bosons lack an explanation for their mass. The Higgs field is an auxiliary field to accomplish that, and it also gives mass to the charged fermions via Yukawa interactions. The Higgs boson is a consequence of this theory. Since it is supposed to give mass to every massive particle in the Standard Model, it couples to all of them. This coupling should be tested experimentally.

As of today, the Higgs boson, with a measured mass of 125 GeV, has been observed to decay into $\gamma\gamma$, ZZ, WW, $\tau\tau$, $t\bar{t}$ and, most recently, $b\bar{b}$. As seen in figure 2.2a, at a Higgs boson mass of 125 GeV, the highest branching ratio correspond to the $H \rightarrow b\bar{b}$ channel. Even though it is the most common decay produced, the overwhelming background from multijet production made necessary to consider processes that exhibits features not present in QCD. The most effective is the production of the Higgs boson in association with a vector boson Z or W, which later decays into $Z \rightarrow \nu\nu$, $W \rightarrow l\nu$ and $Z \rightarrow ll$ (where l stands for an electron or a muon and ν stands for a neutrino). As shown in figure 2.2b, the cross section for the Higgs boson production with a vector boson is quite low, and other background besides QCD arise from top quark or vector boson production.



(a) SM Higgs boson branching ratios



(b) SM Higgs boson cross-section at $\sqrt{s} = 14$ TeV

Figure 2.2: Standard Model decay branching ratios (a) and production cross section (b) of the Higgs boson [15].

2.4 Relevance of $H \to c\bar{c}$

Detecting the Higgs boson decay to two charm quarks is a much more challenging task than to two bottom quarks. To reduce the QCD background, the search also focuses in Higgs boson production with a vector boson and the same decay channels for it. However, the main difference between $H \to b\bar{b}$ and $H \to c\bar{c}$ is the branching ratio: for a $M_H = 125$ GeV, 58% of Higgs bosons decay to $b\bar{b}$ and only 2.8% decay to $c\bar{c}$ [16].

This branching ratio difference translates into an important background in the search for $H \to c\bar{c}$, where $H \to b\bar{b}$ events need to be discriminated. Machine learning techniques, as explained in section 5.1.1, are used for this tasks. Jet flavour identification also plays an important factor. The main incentive to search for $H \to c\bar{c}$ events is to experimentally prove the Higgs boson coupling to second-generation fermions.

Chapter 3: Description of the VHcc Analysis

The team I collaborated with works in the first search by the CMS Collaboration for the standard model Higgs boson decaying into two charm quarks [17]. The search exploits the Higgs boson production associated with a W or Z boson to reduce the QDC background, using a data set of *pp* collisions collected by the CMS experiment in LHC, corresponding to an integrated luminosity of 35.9 fb^{-1} and a center of mass energy of 13 TeV. In the paper, two strategies are used to better exploit the topology of the Higgs boson decay: resolved-jet topology, by targeting lower vector boson transverse momentum one can reconstruct the Higgs boson via two resolved jets originated from the two charm quarks; merged-jet topology, in which the two c-jets merge into a single jet because the vector boson has higher transverse momentum.

My contributions will help the resolved-jet topology strategy by applying a kinematic fit algorithm to improve the reconstructed Higgs boson mass resolution. Currently, a BDT is used to discriminate signal events from background events, but a deep neural network was tested to improve efficiency vs background rejection.

3.1 Event reconstruction

At CMS, the particle-flow (PF) algorithm [18] seeks to reconstruct and identify the individual particles with information obtained from the detector. The list of PF candidate physics objects include charged or neutral hadrons, electrons, muons and photons.

Electrons and photons reconstruction primarily concerns the ECAL, but electrons use additional information provided by the tracker. Muons are also reconstructed with data from the tracker, but the muon system provides the required information for an optimal reconstruction.

Jets consist mainly of hadrons originated from the hadronization of quarks and gluons. They are reconstructed using the anti-k_T algorithm [19], with a radius parameter of $R = 0.4$. One of the most important tasks for this analysis is to identify which particle originated the jet: gluon, b, c or light quarks. Because of their similarity to light-flavor or gluons jets, tagging c-jets is more complicated than tagging b-jets [20]. Due to the lifetime of c quarks being shorter than that of the b quarks, the flight distance lie between the distributions for b and light-flavour jets. One has to find a balance between a high efficiency and a high purity tagger.

The particles that do not interact with the detector material, like neutrinos, can be detected with the missing transverse momentum $\vec{p}_T^{miss}$(or missing transverse energy E_T^{miss}) [21]. It is defined such as it balances the vector sum of all the PF candidates in an event, later corrected with jet energy corrections:

$$\vec{p}_{T,PF}^{miss} = -\sum_{i=1}^{N_{particles}} \vec{p}_{T,i} - \sum_{j=1}^{N_{PFjets}} (\vec{p}_{T,j}^{corr} - \vec{p}_{T,j}) \tag{3.1}$$

Neutrinos are reconstructed using the missing transverse energy, E_T^{miss}. All particles that decay before they reach the detector are reconstructed only by their decay products. Their energy-momentum vector can be reconstructed as the sum of the energy-momentum vectors of their decay products.

3.1.1 c-tagging

The identification of c-jets is achieved with the DeepJet algorithm [22]. The output of the network can be interpreted as the probabilities, p, for a given jet to have originated from either b, c or light-flavour jets, $p(b)$, $p(c)$, $p(udsg)$. By combining the outputs, it is possible to define two discriminators for c-tagging. The discriminant between c-jets and light-flavour quarks or gluons is defined in equation 3.2, and the discriminant between c-jets and b-jets in equation 3.3. The two jets with the highest score of $CvsL$ are chosen to reconstruct the Higgs boson candidate (H_{cand}).

$$CvsL = \frac{p(c)}{p(c) + p(udsg)} \tag{3.2}$$

$$CvsB = \frac{p(c)}{p(c) + p(b)} \tag{3.3}$$

3.2 Monte Carlo simulated events

The analysis is performed on simulated events called Monte Carlo (MC) samples. They try to recreate all the events happening at the detector so that the analysis on data and MC samples are identical. `GEANT4` [23] is used to recreate the measurement process as the CMS would do. The simulations are only approximations of usually leading order (LO) or next to leading order (NLO).

To produce the Monte Carlo simulated events, different event generators were employed[1]:

- `POWHEG` v2: it produces quark induced ZH and WH at NLO with the help of Multi-scale improved NLO (MiNLO). It also generates gluon-induced ZH processes at LO, $tW-$ at NLO, $t\bar{t}$ at NLO and $t-$channels at NLO.

[1]References to every generator and sample can be found in section 3 of [17]

- MADGRAPH5_AMC@NLO v2.4.2: this generator produces V+jets processes at NLO for two additional partons and at LO with four additional partons. The same generator was used to produce diboson background events (WW, WZ and ZZ) at NLO with up to two additional partons and QCD multijet events at LO. It is also used to generate $s-$channel single top processes.
- PYTHIA V8.230: with the CUETP8M1 underlying event tune, it generates parton showering and hadronization, including the $H \to c\bar{c}$ decay.

The parton distribution functions (PDF) used to generate all samples are the NNLO NNPDF3.1 set. The production cross section for the different processes are scaled to match the luminosity of the collected data.

3.3 Framework

The analysis framework used in this dissertation is based on ROOT [24]. The Toolkit for Multivariate Analysis (TMVA) is a ROOT-integrated framework that facilitates the processing, parallel evaluation and application of multivariate classification techniques, designed for machine learning applications [25]. A TMVA analysis mainly consist in training and application. In this dissertation, TMVA was used to train a boosted decision tree (BDT) and to research deep neural networks (DNNs) to discriminate signal vs background events.

PyMVA [26] is an interface for third-party MVA tools based on Python. It allows for external libraries, like Keras, to be easily accesible and integrated in the TMVA workflow. Keras [27] is a deep learning API written in Python, capable of running on top of TensorFlow [28]. The DNN models implemented in this dissertation were developed with Keras, and implemented in TMVA with PyKeras and PyMVA, to allow an easy comparison between the BDT and DNN results.

3.4 Event selection

Three different channels are defined by the charged-lepton multiplicity of the vector boson decay: "0L" channel as referring to the $ZH \to \nu\nu c\bar{c}$ signal process; "1L" channel as referring to the $WH \to l\nu c\bar{c}$ signal process; "2L" channel as referring to the $ZH \to l^+l^-c\bar{c}$ signal process. Only electrons and muons are considered as possible lepton decays.

- **0L channel:** only events with a p_T^{miss} threshold of 170 GeV pass the trigger (or 110 GeV with an additional missing hadronic transverse energy of 110 GeV). Events with an identified isolated lepton are rejected.

- **1L channel:** they require an isolated lepton with a p_T above 30 GeV for electrons and 25 GeV for muons. The W boson is reconstructed as the vectorial sum of the lepton momentum and the $\vec{p}_T^{miss}$. It is required that $\Delta\phi(p_T^{miss}, l) < 2.0$ so that it is compatible with the leptonic decay of a Lorentz-boosted W boson.

- **2L channel:** events are obtained by requiring a pair of leptons to be the same flavour, to have different electric charge, and to pass the p_T threshold of 23 and 20 GeV for electrons, and 20 GeV for muons. The Z boson is reconstructed as the vector sum of the two leptons, and only events where the invariant mass of the candidates is compatible with the Z boson mass ($75 < m_Z < 105$ GeV).

There are also some extra cuts designed to suppress background processes as much as possible while still retaining signal events:

- **QCD multijet background:** it can be greatly reduced by requiring a high-p_T vector boson, that the $\vec{p}_T^{miss}$is not aligned with any jet in the event, and that the azimuthal angular separation $\Delta\phi(\vec{p}_{T,trk}^{miss}, \vec{p}_T^{miss}) < 0.5$, where $\vec{p}_{T,trk}^{miss}$ is calculated only from charged particles.

- **$t\bar{t}$ background:** only relevant in 0L and 1L channels, where it is suppressed by rejecting events where the additional small-R jet multiplicity ($N_{small-R}^{aj}$) is greater than 1.

- **V+jets background:** by requiring the dijet invariant mass to satisfy $50 < m_{jj} <$ 200 GeV, this background contribution is reduced. This cut is applied as a result of the known mass of the Higgs boson.

- **$H \rightarrow b\bar{b}$ background:** because of its kinematic similarities to $H \rightarrow c\bar{c}$ events, the dedicated jet flavour taggers have to be exploited to reduce it.

Chapter 4: Kinematic Fit in the 2L channel

4.1 Introduction

The aim of the following study is to improve the transverse momentum resolution of the c-jets, bringing them closer to the momentum of the true c-jets. As a consequence, an improvement in the resolution of the invariant mass of the reconstructed Higgs boson is expected. Given that in the 2L channel ($ZH \rightarrow l^+l^-c\bar{c}$) there is no E_T^{miss} on truth level, and thus the transverse plane momentum should be conserved, it is the ideal channel to perform the kinematic fit. It is possible to apply a kinematic fit analysis to other channels where there are unmeasured parameters, like the momentum of neutrinos, but more research would be necessary.

In the 2L channel, we could also take advantage of the fact that both leptons decay from a Z boson which has a well known mass of $m_Z = 91.2$ GeV and a decay width of $\Gamma = 2.5$ GeV [29]. This constraint is not expected to have a great impact in the $m_{c\bar{c}}$ distribution, but combined with momentum conservation, should provide improved results.

4.1.1 Kinematic Fit with constraints

By forcing well defined kinematic hypotheses about some event, one can change the measured values within their errors to better fulfill a series of constraints. These constraints can be expressed as physical laws, such as momentum conservation or the known mass of a certain particle.

The kinematic fit is a least square method that uses and modifies kinematic variables. The constraints are chosen so that the true values of these variables can fulfill them. If we want to solve a problem with y_i as n measured uncorrelated parameters (with $\bar{y}$ as the true parameter value) and f_k constraints that should be fulfilled:

$$f_k(\bar{y}_1, \bar{y}_2, ..., \bar{y}_n) = 0 \qquad for \quad k = 1, 2, ..., m \tag{4.1}$$

It's important to notice that, in our case, we don't have any unmeasured parameters like the momentum of a neutrino. In general, the measured values $\bar{y}$ will not fulfill the constraints. One has to calculate a correction in such a way that the likelihood is minimized. By using the definition of Lagrange Multipliers, we can determine a local extrema of a non-linear function of many variables:

$$L(\vec{y}, \vec{\lambda}) = S(\vec{y}) + 2\sum_{k=1}^{m} \lambda_k f_k(\vec{y}) \qquad S(\vec{y}) = \sum_{i=1}^{n} \frac{(y_i' - y_i)^2}{\sigma_i^2} = \Delta\vec{y}^T V^{-1} \Delta\vec{y} \tag{4.2}$$

In equation 4.2, λ_k represents the Lagrange Multipliers and σ_i the standard deviation of the measured value y_i. Because of the non-linearity of the problem, the solution has to be found iteratively, where the problem has to be linearized in every iteration:

$$f_k(\vec{y}') \approx f(\vec{y}^*) + \sum_{i=1}^{n} \frac{\partial f_k}{\partial y_i}(y_i' - y_i^*) \approx 0 \tag{4.3}$$

with $\vec{y}^*$ as the values of the parameter after the last iteration and $\vec{y}'$ as the value at the current iteration. This linearization allows to calculate the value of $\vec{y}'$ in every step until the convergence criteria is fulfilled. Usually, this condition is that the χ^2 changes less than a given value ε_S and that the constraints are also smaller than a given value ε_F:

$$\frac{S(n-1) - S(n)}{ndf} < \varepsilon_S \qquad and \qquad F_{(n)} = \sum_{k=1}^{m} \left| f_k^{(n)}(\vec{y}) \right| < \varepsilon_F \tag{4.4}$$

The number of degrees of freedom (ndf) denotes the difference between the constraints and the amount of unmeasured quantities. In this particular implementation, there is a total of three constraints, momentum conservation in x and y axis and a known two lepton invariant mass, with no unmeasured particles. There is also the possibility to protect the fit from overshooting by comparing the value of the constraints in every iteration with its previous value. If $F_{(n)} > F_{(n-1)}$, the corrections are divided by two and then compared again to the previous value. We repeat this process until the new constraints value is smaller or after a fixed amount of times. A more detailed mathematical explanation to the solution of equation 4.3 is explored in appendix A.

The kinematic fit implementation uses the *KinFitter* software package for ROOT [30]. One important task during my project was to rewrite the kinematic fit implementation in the analysis from python to C++.

4.1.2 Pull Histograms

One important tool to analyse the behavior of the kinematic fit are the pull histograms. Thanks to the central limit theorem, we can apply the idea that for a measured parameter y with an uncertainty of σ distributed around a true value μ, we can define the pull as:

$$g = \frac{y - \mu}{\sigma} \tag{4.5}$$

which is by definition standard distributed with mean zero and unity width [31]. Two separate definitions will be used to describe different aspects:

$$g_m = \frac{y_f - y_m}{\sqrt{\sigma_m^2 - \sigma_f^2}} \tag{4.6}$$

$$g_{tr} = \frac{y_f - y_{tr}}{\sigma_f} \tag{4.7}$$

where y_m and σ_m are the measured parameter and its uncertainty; y_f and σ_f are the fitted parameter and its error after the fit; and y_{tr} represents the true value of the parameter.

The measured pull histograms (g_m), provide information even without data about the true value of the parameter. The width of the distribution is related to the change of the parameter expected by the errors: a broader width means a greater change, and vice versa. Badly chosen constraints or errors chosen wrong could explain this behavior. A shifted peak or asymmetrical tails can be caused by systematic error or wrong constraints.

The truth pull histograms (g_{tr}) measure how different is the fitted parameter to its true value. The width is related to how big or small the errors are chosen, a broader (smaller) width represents error too low (high). An asymmetry in the distribution means that the fitted parameter is systematically shifted from its true value.

4.2 Constraints

The first constraint exploited in this study is that the transverse plane momentum, divided into x-axis (eq. 4.8) and y-axis (eq. 4.9), should be conserved. Some neutrinos might originate from semileptonic c (or b) decays, but it only adds to the already high uncertainty of the jets.

$$p_x^{l_1} + p_x^{l_2} + p_x^{c_1} + p_x^{c_2} + p_x^{r} = 0 \tag{4.8}$$

$$p_y^{l_1} + p_y^{l_2} + p_y^{c_1} + p_y^{c_2} + p_y^{r} = 0 \tag{4.9}$$

In the constraints, r denotes the so called *Remnant Vector*. It is a necessary constructed momentum because the initial and final state radiation can carry some transverse momentum. Here, the Remnant Vector is defined as the vectorial sum of all non-tagged jets that pass a pileup test [32] with $p_T > 20$ GeV (eq. 4.10). A more detailed analysis on the different possibilities for the Remnant Vector definition can be found in [33].

$$r_{jets} = \sum_{j \neq c} j \tag{4.10}$$

The other constraint is that the invariant mass of the two leptons should be equal to the mass of the Z boson. However, the *KinFitter* package allows to add this mass constraint as a gaussian distribution, and not only as a delta distribution.

$$(p_{l_1} + p_{l_2})^2 - m_z^2 = 0 \qquad m_z = 91.2 \text{ GeV and } \Gamma = 2.5 \text{ GeV} \tag{4.11}$$

For our particular implementation, a total of five "particles" will be fed to the fitter program: the two c-jets, the two leptons and the Remnant Vector (or recoil particle). Both c-jets and leptons use polar coordinates (p_T, η and ϕ), while the Remnant Vector is defined in cartesian coordinates (since its p_T value is by design close to zero). One of the drawbacks of the previous implementation is that the mass of the c-jets and leptons is not considered during the fit. The mass implementation will be studied in section 4.4.1.

Not all feeded parameters are affected by the constraints. Only p_T and ϕ of the c-jets are modified to satisfy equations 4.8 and 4.9, so η^c should not be affected. The p_z^r parameter of the Remnant Vector is also not affected by the constraints. For leptons, all their kinematic parameters are modified to satisfy equation 4.11.

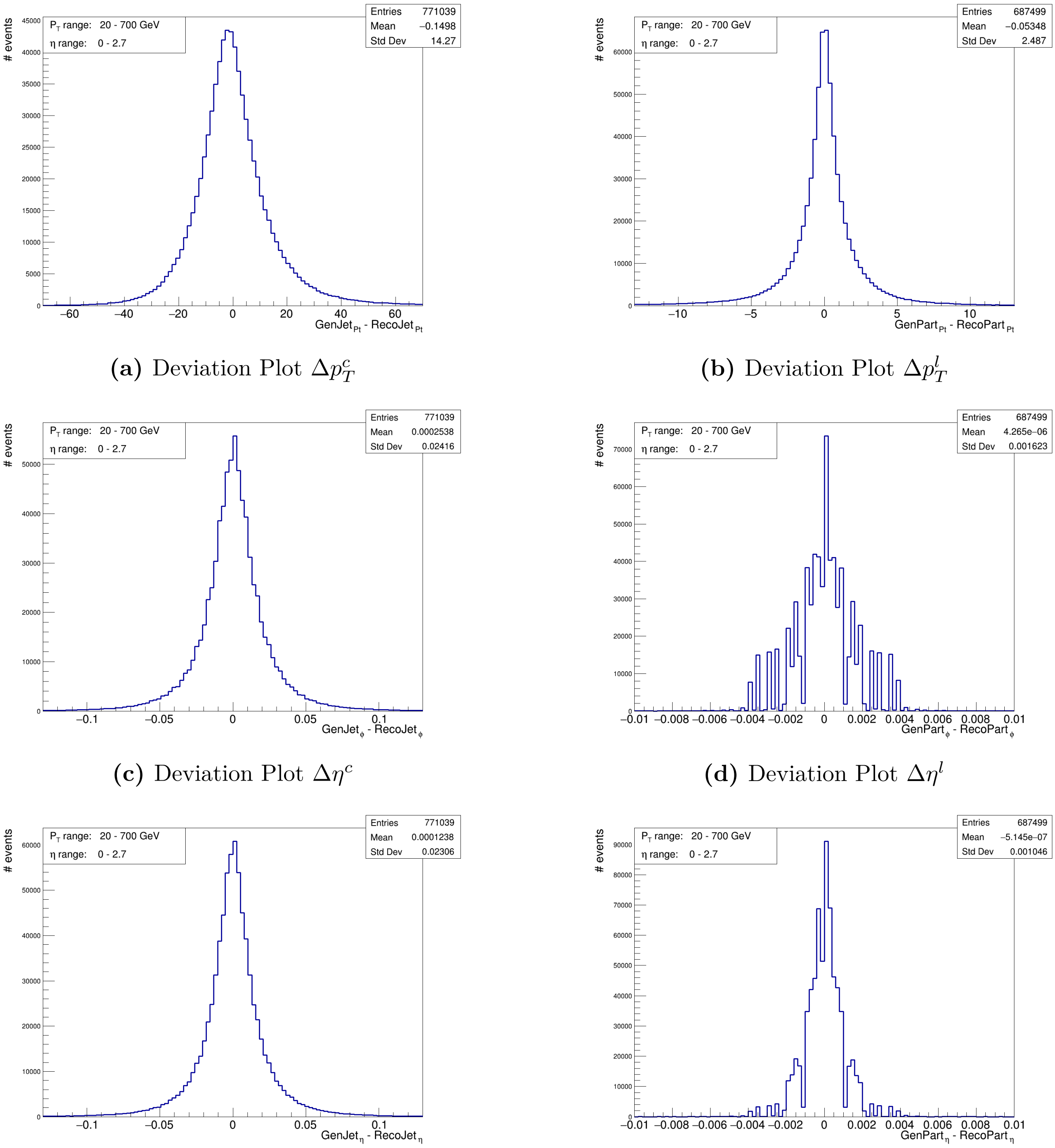
4.3 Error Parametrization

One of the most relevant aspects of the kinematic fit implementation is a correct error estimation σ_m of the kinematic variables. The fit is highly dependant on the right choice of errors. To obtain the deviation values empirically, one can use truth information contained in the Monte Carlo samples. Each reconstructed object needs to have an associated generated true object.

The leptons are associated to truth leptons decayed from the Z boson. When choosing a truth object for the c-jets, one can use the associated c-quarks decaying from the Higgs boson before any final state radiation or an associated generator level jet. A study of whether to use c-quarks or generator level jets concluded that using generator jets as the associated object provides better and more consistent results. Since no truth information is available to obtain the Remnant Vector's standard deviation, jet resolutions from the *JME Group* [34] were used.

Once the reconstructed object (*RecoJet* or *RecoPart*) is associated to a truth object (*GenJet* or *GenPart*), their difference is fed to a histogram. The RMS value of these histograms is used as the standard deviation of the corresponding kinematic parameter. These histograms are not perfectly gaussian, so the RMS is not a perfect representation of the standard deviation. However, it is a good enough estimation, since small changes in σ_m have a small impact on the final fit result.

An example of how the error is calculated can be seen in figure 4.1. The plots presented are not the ones used in the analysis because the deviation changes for different regions of the phase space. The histograms are split in different regions of p_T and η for each momentum vector. More regions account for better resolution in the σ_m parametrization, but can also lead to not sufficient data. A trade off has to be found between obtaining sufficient regions to better parametrize and having enough events per region so that the RMS value is a good representation of the standard deviation of the distribution.



(a) Deviation Plot Δp_T^c

(b) Deviation Plot Δp_T^l

(c) Deviation Plot $\Delta \eta^c$

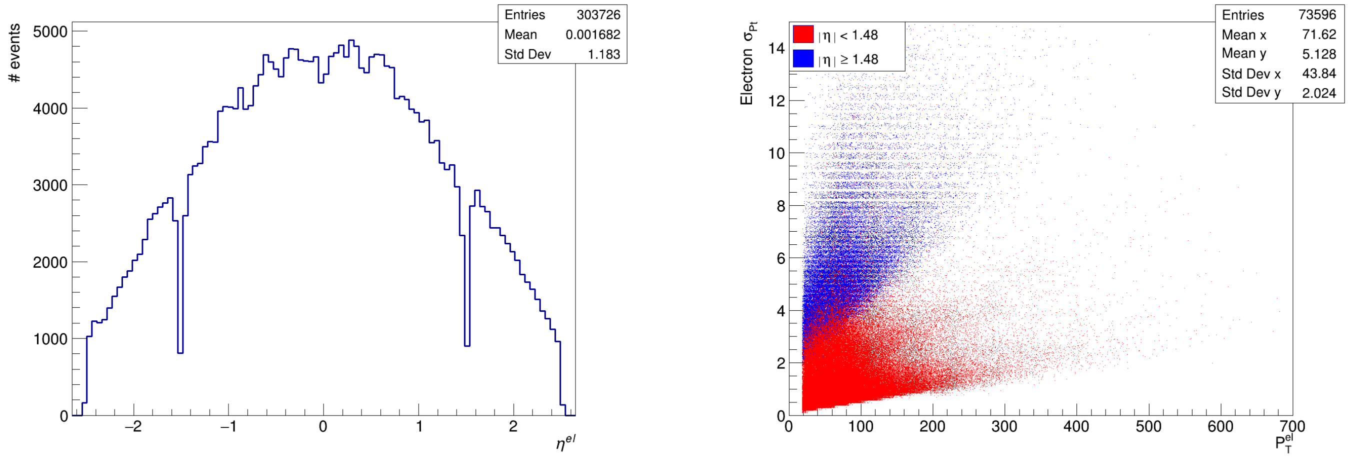
(d) Deviation Plot $\Delta \eta^l$

(e) Deviation Plot $\Delta \phi^c$

(f) Deviation Plot $\Delta \phi^l$

Figure 4.1: Deviation plots showcasing how σ_m for every kinematic parameter is calculated. The shifted peak in Δp_T^c is caused by the p_T cuts and corrections applied. Spikes in the $\Delta \eta^l$ and $\Delta \phi^l$ distributions are most likely caused by floating-point errors.

The p_T regions are chosen arbitrarily with the idea of keeping an approximately similar number of events per segment. The η regions are chosen differently for jets and leptons, where for jets they are chosen arbitrarily, and for leptons they are chosen after a discontinuity in the η distribution of the measured electrons.



(a) Electron η distribution

(b) Electron p_T vs Electron σ_{Pt}

Figure 4.2: Figure (a) shows the discontinuity in the η distribution for electrons. In figure (b), we use an already existing σ_{Pt} distribution[2]to show that when the previous η cut is applied, two distinct regions appear.

Once the RMS values for the different parameters and ranges are obtained, one has to parametrize them so that a function can obtain the standard deviation per each event. These parametrization are done with respect to the particle's p_T (as a continuous function), in the different η ranges. The RMS values calculated and the parametrization functions can be seen in appendix B.

4.4 Implementation

Once the parametrization functions are obtained, one can directly translate a particle's p_T and η into a known σ_m. Pull tests will help determine whether the chosen parametrization is correct or not. During the analysis, each event that fulfills the conditions required[3], will execute the kinematic fit by sending all the parameters and their uncertainties to the *KinFitter* package algorithms. The parameters sent are after final state radiation (gluons irradiated by the c-quark before hadronization) recovery.

As mentioned in section 4.1.1, the kinematic fit algorithm also needs some configuration parameters: ε_S, ε_F and the maximum number of iterations. These values are chosen so that most of the events converge in a reasonable amount of time, but an extended study was not required.

[2]In the samples provided to run the code to calculate the standard deviations, there already exist branches with the uncertainties from both electrons and muons p_T for each event. It was decided not to use these already existing uncertainties as it is more appropriate for the analysis to obtain all standard deviations from a similar method.

[3]These conditions are that the event has two resolved jets and that a Z boson decays into either two electrons or two muons.

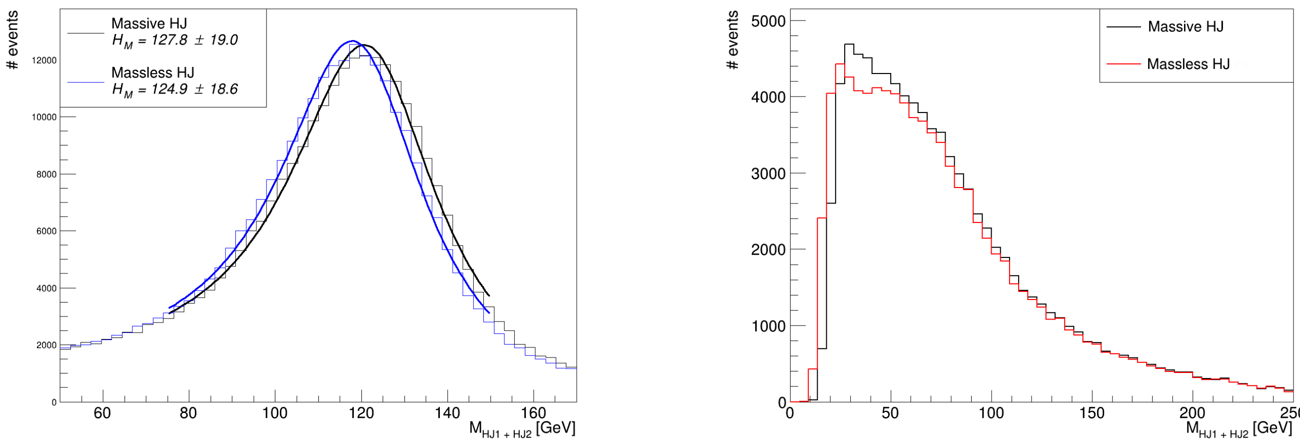
Hyperparameter	Value
ε_S	5×10^{-3}
ε_F	1×10^{-3}
Max Number Iter.	100

Table 1: Hyperparameters chosen for the kinematic fit algorithm.

4.4.1 Mass in the ***KinFitter*** package

The *KinFitter* package used in this study lacks the ability to use the mass of the particles involved. Since only three parameters are fed into the algorithm, there is not enough information to completely reconstruct the four-momentum of the particles. There are two options studied here: one possibility is to ignore the mass of the object (this was the previous implementation); the other is to keep the mass of the object constant.

If the Higgs jets mass is chosen to be zero, a drift towards smaller masses of the reconstructed m_H is observed. Figure 4.3 shows the effect in both the $ZH(c\bar{c})$ signal sample and in the Drell-Yan background sample.



(a) $ZH(c\bar{c})$ signal sample **(b)** Drell-Yan background sample

Figure 4.3: Effect of the mass of Higgs jets in $m_{HJ1+HJ2}$.

The option presented in this study is a combination of both. Leptons will have their masses set to zero, since their rest mass is basically zero compared to other components in the analysis, while c-jets will have their masses constant, but not during the whole process. The algorithm will execute the kinematic fit to massless particles, and only to the output results the mass they previously had will be added.

4.5 Pull Tests

As mentioned in section 4.1.2, pull histograms are a good tool to understand the effects of the kinematic fit. By analyzing how close to a gaussian both g_m (eq. 4.6) and g_{tr} (eq. 4.7) are, we can infer whether the errors were chosen wrong or not.

The results of both types of pull tests were analyzed in p_T ranges to better understand which parameters needed a better error parametrization and where. Plots can be seen in appendix C. To facilitate the presentation of results, no p_T ranges are shown.

4.5.1 Measured pull histograms

The measured pull histograms g_m (figure C.1) are not close to a normal distribution. For c-jets, the width of the distributions shows that the parameters are changed more than expected by their errors. In the case of leptons, both p_T and ϕ shows a more normal-like distribution. However, the lepton's η parameter does seem to have some systematic problem causing a lot of events to not change their value (explaining the peak at zero).

4.5.2 Truth pull histograms

The truth pull histograms g_{tr} (figure C.2) show different results. The distributions for c-jets are much closer to a normal distribution than in g_m, except for the p_T parameter which is at the same standard deviation value. For the leptons, some granularity appears in the η and ϕ parameters that might be cause of floating-point errors. Even with that, results shows a clear gaussian-like distribution, even in the p_T parameter.

The overall shape of the pull distributions (both g_{tr} and g_m) shows the presence of systematic uncertainties or wrong error choices in some parameters. It was tested to use a gaussian fit instead of the RMS value to obtain the standard deviation (as explained in 4.3), but it had no impact in the results or in the pull histograms. According to [31], *"It may happen that the pull distribution is approximately Gaussian, but its width is not 1. Assuming that this is understood to be an effect of the non asymptotic nature of the problem [...] one may want to correct the quoted uncertainties by multiplying them by the width of the pull distribution"*. This approach was also tested, and while it did improve the shape of the pull histograms, the kinematic fit algorithm worsens the Higgs boson mass resolution. Several different methods to obtain the errors were tested, including the use of c-quarks as the associated truth object or using generator level information with and without neutrinos, but none provided pull tests results as good as the ones presented.

4.6 Results

After implementing the corresponding error parametrization and the *KinFitter* package algorithms into the currently existing analysis tools, results when applied to $ZH(c\bar{c})$ samples are presented in different signal regions:

- **Zee Low VPt:** figure 4.4a. Events where the Z boson decays into electrons and $50 < V_{Pt} < 150$ GeV. Results show an improvement in mass resolution of 14%.
- **Zmm Low VPt:** figure 4.4b. Events where the Z boson decays into muons and $50 < V_{Pt} < 150$ GeV. A 11% improvement in mass resolution is observed in this signal region.
- **Zee High VPt:** figure 4.4c. Events where the Z boson decays into electrons and $V_{Pt} > 150$ GeV. The kinematic fit provides up to a 31% improvement in the mass resolution.
- **Zmm High VPt:** figure 4.4d. Events where the Z boson decays into muons and $V_{Pt} > 150$ GeV. As with the other High VPt region, the mass resolution improves by 31%.

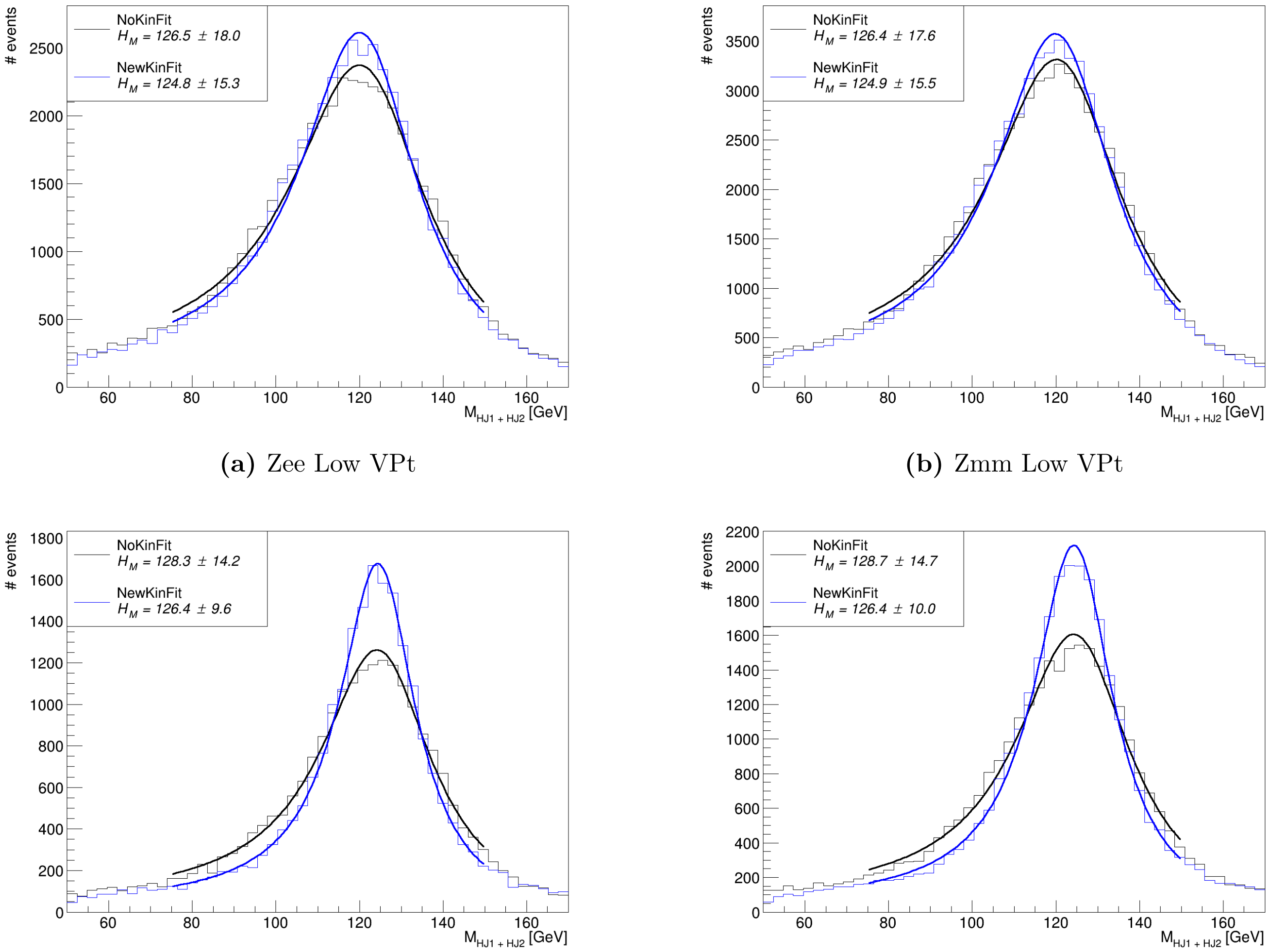
These results use the mass implementation mentioned in 4.4.1, where the mass of the c-jets is chosen constant. Results show a great improvement in every signal region, more noticeable in the High VPt region.

4.6.1 Updating the implementation

The kinematic fit algorithm was already implemented in the $H \to b\bar{b}$ analysis with a Python script executed separately from the rest of the analysis. One of my tasks was to update this implementation for $H \to c\bar{c}$ analysis, so that the algorithm is executed per event. I also rewrote the code to C++, to take advantage of the full potential of the ROOT framework. One important consequence of the C++ implementation is that the algorithm is executed much more efficiently both in resources and time needed (up to a 70% time reduction).

As in section 4.5, pull tests were applied to the previous kinematic fit implementation, and their results showed that errors needed to be updated. The hyperparameters from section 4.4 were also updated to improve the convergence rate and to achieve more precise results.

Thanks to these changes, the new implementation obtains a 5% improvement in the Low VPt region and a 9% improvement in the High VPt region over the previous implementation, while using less resources and time.



(a) Zee Low VPt

(b) Zmm Low VPt

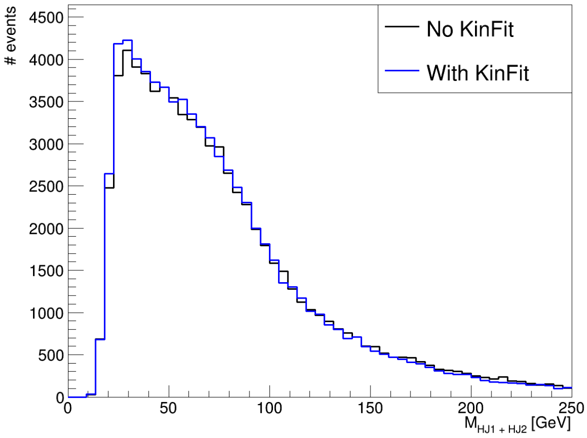
(c) Zee High VPt

(d) Zmm High VPt

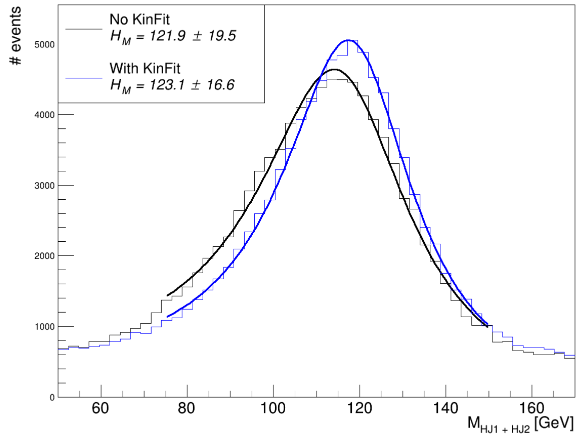
Figure 4.4: Reconstructed $m_{c\bar{c}}$ before and after the implementation of kinematic fit.

4.6.2 Effect on background Samples

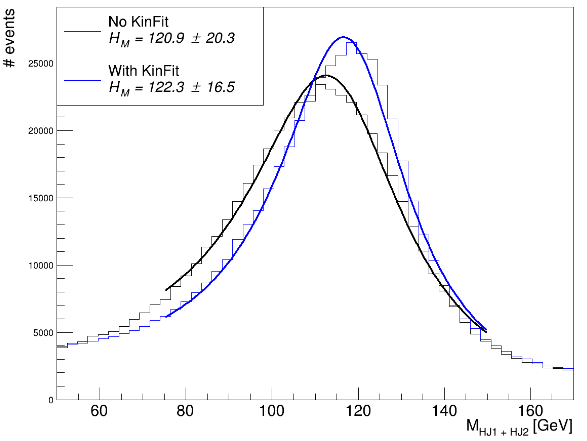
The kinematic fit was optimized for its application in the $ZH \rightarrow l^+l^-c\bar{c}$ signal sample. In figure 4.5, the effect of the kinematic fit applied to several background samples is observed. The effect on the Drell-Yan samples (figure 4.5a) is quite relevant, since no mass sculpting is observed. For the other samples, the kinematic fit improves the reconstructed $m_{HJ1,HJ2}$ resolution. In figure 4.5d the check in the diboson channel ($ZZ \rightarrow l^+l^-q\bar{q}$) also shows improvement and a peak at around m_Z as it was expected, since $m_H = 125$ GeV is not used in the fit itself.



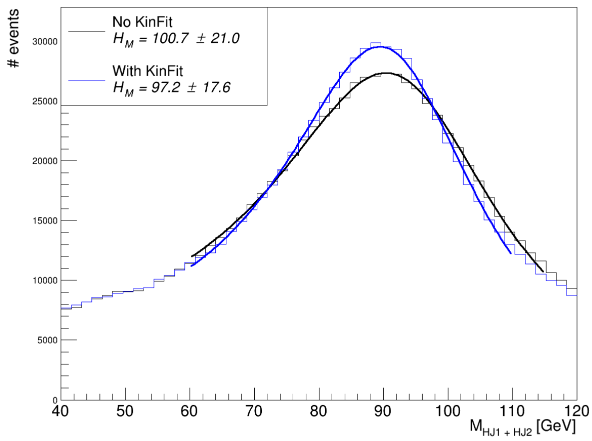
(a) Drell-Yan



(b) $ggZH(b\bar{b})$



(c) $ZH(b\bar{b})$



(d) ZZ to 2 leptons + 2 quarks

Figure 4.5: Effects of the kinematic fit algorithm applied to different background samples.

Chapter 5: Deep neural network application in $VH(c\bar{c})$ analysis

5.1 Introduction

Signal separation from background is an important task in every high energy physics (HEP) research. Highly specialized machine learning algorithms are created to execute these tasks [35]. In $H \to c\bar{c}$ analysis, the Higgs boson decay into a pair of c quarks signal has to be extracted over a huge background.

The study of how a deep neural network can perform this classification task is presented in this dissertation, and it will be compared with the current machine learning method employed in the analysis: Boosted decision tree with gradient boost (BDTG).

5.1.1 Boosted Decision Tree

A decision tree is a tool that uses tree shaped models with the objective to predict the value of a target variable starting with other input variables. An example can be seen in figure 5.1. They are structured in nodes and leaves, the former represents the features being tested and the later represents the cuts that apply to these features and that split the path. The split is chosen to maximize information gain or minimize entropy. The process continues until some convergence criteria is met or until the maximum tree depth has been reached.

Boosted decision tree (BDT) is a tree in which we combine many weak trees into a strong classifier. The performance of the previous tree is taken to optimize the next one. Boosting stabilizes the response of the decision trees which can considerably enhance the performance.

BDT is implemented with TMVA [25] as a machine learning algorithm able to predict an outcome for a measurement. In the analysis, a BDT is trained to perform a discrimination between signal and background events.

5.1.2 Deep Neural Network

Deep neural networks are a class of machine learning techniques that attempt to learn in multiple levels of representation to model a complex relationship among data [36]. A visual representation of a DNN can be seen in figure 5.2. It consists of several hidden layers with a large number of neurons in each. In this study, low-level features from

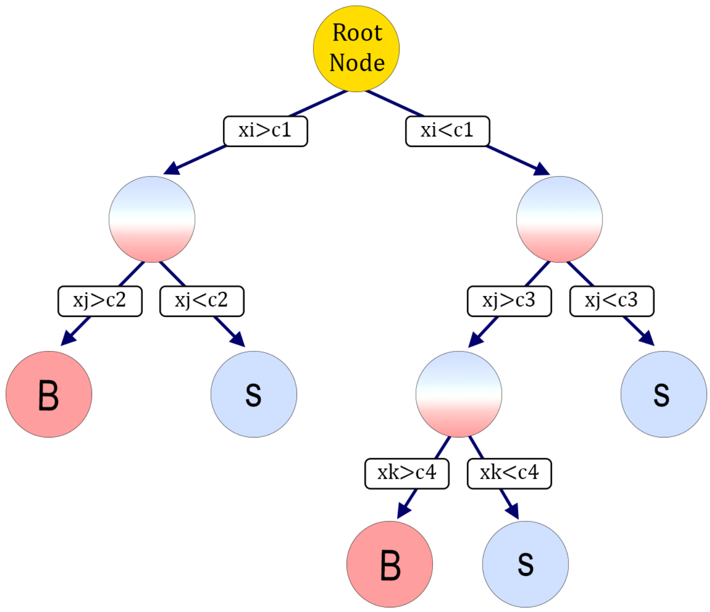


Figure 5.1: Visual representation of a decision tree.

basic kinematic quantities along with c-tagging discriminants are the inputs from which complex relations are exploited to classify events.

It is proven that shallow neural networks (consisting of one hidden layer) are capable of approximate any function [37]. However, they usually require a high number of neurons and fail to obtain useful functions from high-dimensional data sets. In a deep neural network, the layers can be interpreted as building blocks of a hierarchical representation of the data. In image recognition objects for example, the first layers learn low-level features, such as edges or corners, while deeper layers learn more complex features, like the shape of an object, to finally be able to classify them.

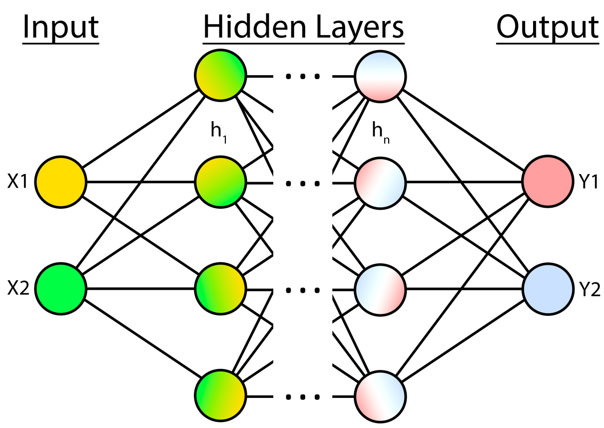


Figure 5.2: Visual representation of a deep neural network.

The DNN is integrated in the analysis with TMVA, which allows for models created with Keras to be trained and tested.

5.1.2.1 Hyperparameters of the Network

In any machine learning algorithm, the hyperparameters are parameters that must be initialized before training a model. Opposite to usual "model parameters" (like weights or biases) which usually are properties of training data that will be modified to "learn", hyperparameters (like learning rate or activation function) govern the entire training process as they control the behaviour of the training algorithm.

One of the most important requirements in machine learning research is to find the optimal hyperparameters for the task. Several techniques were considered to study the different hyperparameter possibilities:

- **Grid search:** it consist of manually testing each possible combination of hyperparameters. For example, if we would like to test four different learning rates $L_R = \{1, 0.1, 0.001, 0.0001\}$ and three different activation functions $Act.Func = \{``relu", ``elu", ``tanh"\}$, a total of twelve combinations would be tested.
- **Random search:** if we have several hyperparameters to test, the grid search method quickly becomes unfeasible. A random search instead tries random combinations of the possible hyperparameters, which will outperform the grid search method, especially when only a small number of hyperparameter affects the performance. A comparison between these two methods can be seen in figure 5.3.
- **Bayesian optimization:** it is a general strategy to optimize any "black-box" function, where the interest lays in finding the minimum of a function on some bounded set. What makes it different to other procedures is that it exploits previous results and uses all available information from them [38].

5.1.3 The classifier response

The output of both the BDT and the DNN is called "classifier response". It represents how certain is the algorithm of the event being either background or signal. The current BDTG distributions of the four channels are shown in figure 5.4.

5.1.3.1 ROC curves

A Receiver Operating Characteristic (ROC) curve is a graphical representation of the response of a binary classifier as the discrimination threshold is changed. It is one of the

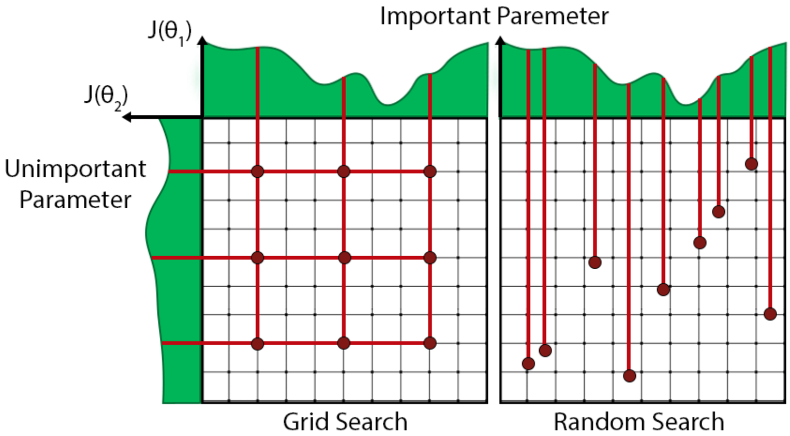


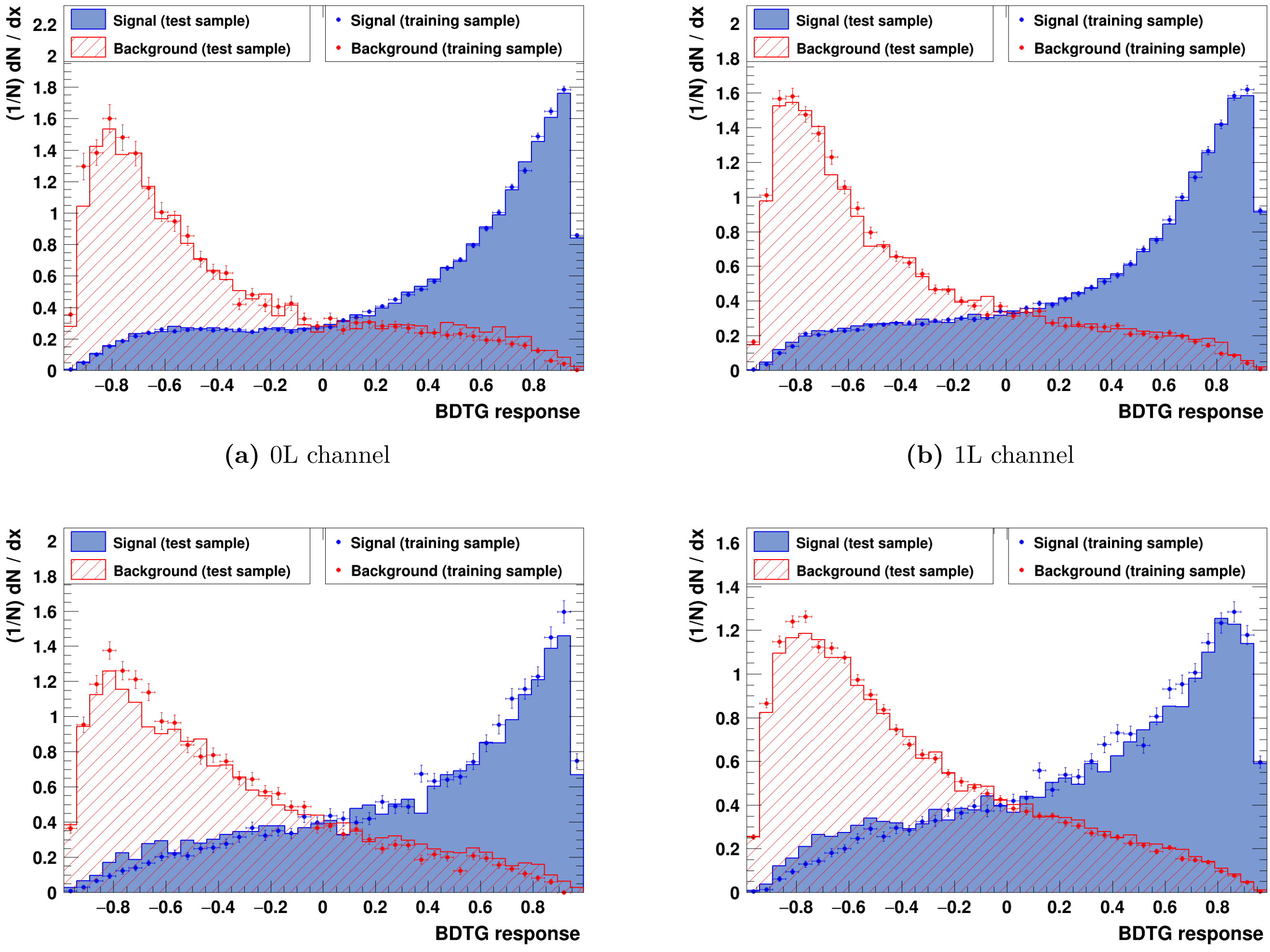
Figure 5.3: Visual representation of grid search and random search in hyperparameter tunning. It shows how a random search could provide better results as they explore more values for each hyperparameter. It is especially useful when there are unimportant hyperparameters that do not have a great impact in the performance of the algorithm, which is our case.

most important evaluation metrics to check the model's performance.

From a classifier response as the shown in figure 5.5a, one has to decide a cut-off point (or criterion value) to discriminate between the two populations. Because of this cut, we can find:

- **True positive (TP):** fraction of signal events that are correctly identified as signal events after the cut (signal events on the right of the cut-off point).
- **True negative (TN):** fraction of background events successfully identified as background (background events on the left of the cut-off point).
- **False positive (FP):** fraction of background events that are mistakenly classified as signal (background events on the right of the cut-off point).
- **False negative (FN):** fraction of signal events incorrectly classified as background events (signal events on the left of the cut-off point).

A trade off has to be found in order to maximize TP events and reduce FP events. The ROC curve is created by plotting signal efficiency, or true positive rate



(a) 0L channel

(b) 1L channel

(c) 2LH channel

(d) 2LL channel

Figure 5.4: BDTG response for the different channels. In blue, the results for signal sample and in red, results for background sample. To check for overfitting, the training sample is also displayed with the dots. If test and training sample follow the same distribution, overfitting is unlikely.

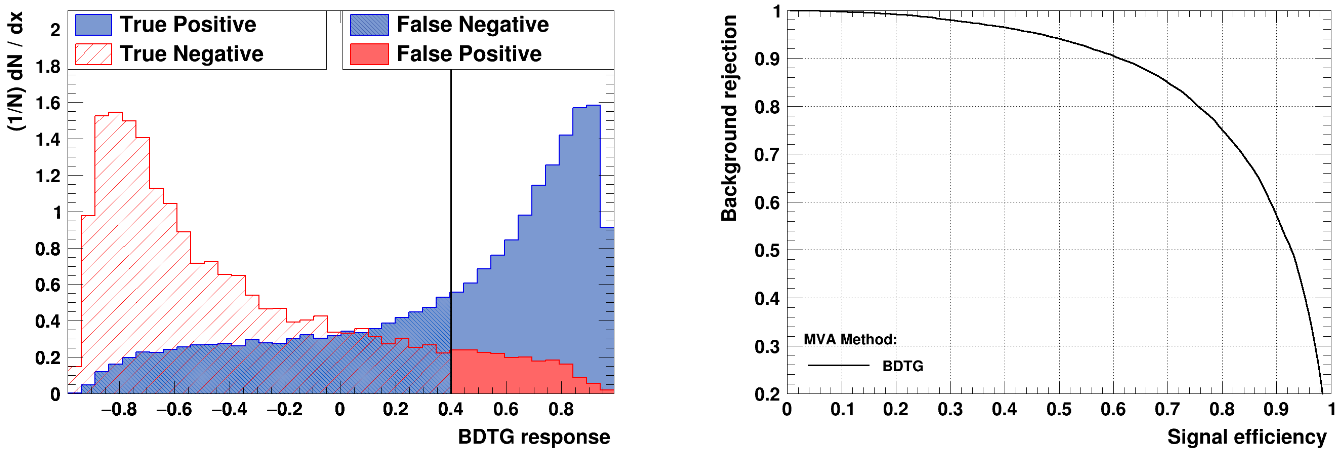
(TPR), against the background rejection, or true negative rate (TNR).

$$TPR = \frac{TP}{TP + FN} \tag{5.1}$$

$$FPR = \frac{FP}{FP + TN} \tag{5.2}$$

$$TNR = 1 - FPR \tag{5.3}$$

As a result, the plot in figure 5.5b is created.



(a) 1L channel classifier response with arbitrary cut-off

(b) 1L channel ROC curve

Figure 5.5: ROC curve definition.

5.1.3.2 AUC - ROC

The area under the ROC curve (AUC) is a performance measurement for classification problems. It translates into how good a model is in distinguishing between classes. It is calculated from the area under figure 5.5b. Throughout this dissertation, it will be the main parameter to compare different discrimination models.

5.2 Kinematic variables

The kinematic variables employed in the BDTG training, different for each vector boson decay channel, are shown in table 2. The DNN training was performed with the same variables as input parameters in order to better compare both methods.

The background samples employed in the training are also dependant of the channel. Some common to all channels are diboson (WW, WZ and ZZ) processes, Drell-Yan and W+jets. In the 2L channel, $ggZH \to l^+l^-b\bar{b}$ and $ZH \to l^+l^-b\bar{b}$ processes are added, while $Z(\nu\nu)$+jets, $ggZH \to \nu\nu b\bar{b}$ and $ZH \to \nu\nu b\bar{b}$ samples are added to the 0L channel. Both single top and double top processes would normally also be considered, specially for 0L and 1L channels were they are one of the main background. However, due to some Data-MC disagreement with the samples at the time of the study, they were not included.

Variables	Description	0L	1L	2L
$m(H_{cand})$	H_{cand} mass	✓	✓	✓
$p_T(H_{cand})$	H_{cand} transverse momentum	✓	✓	✓
$p_T(V)$	Vector boson transverse momentum	✓	✓	✓
$m(V)$	Vector boson mass	–	–	✓
$m_T(V)$	Vector boson transverse mass	–	✓	–
p_T^{miss}	Missing transverse momentum	–	✓	–
$p_T(V)/p_T(H_{cand})$	Ratio between vector boson ad H_{cand} transverse momenta	✓	✓	✓
$CvsL_1$	$CvsL$ tagger value of the leading $CvsL$ jet	✓	✓	✓
$CvsB_1$	$CvsB$ tagger value of the leading $CvsB$ jet	✓	✓	✓
$CvsL_2$	$CvsL$ tagger value of the subleading $CvsL$ jet	✓	✓	✓
$CvsB_2$	$CvsB$ tagger value of the subleading $CvsB$ jet	✓	✓	✓
p_{T,j_1}	p_T of the leading $CvsL$ jet	✓	✓	✓
p_{T,j_2}	p_T of the subleading $CvsL$ jet	✓	✓	✓
$\Delta\phi(V, H_{cand})$	Azimutal angle between vector boson and H_{cand}	✓	✓	✓
$\Delta R(j_1, j_2)$	ΔR between leading and subleading $CvsL$ jets	–	✓	✓
$\Delta\phi(j_1, j_2)$	Azimutal angles between leading and subleading $CvsL$ jets	✓	✓	–
$\Delta\eta(j_1, j_2)$	Difference in pseudorapidity between leading and subleading $CvsL$ jets	✓	✓	✓
$\Delta\phi(l_1, l_2)$	Azimutal angles between leading and subleading p_T leptons	–	–	✓
$\Delta\eta(l_1, l_2)$	Difference in pseudorapidity between leading and subleading p_T leptons	–	–	✓
$\Delta\phi(l_1, j_1)$	Azimutal angle between leading p_T lepton and leading $CvsL$ jet	–	✓	–
$\Delta\phi(l_2, j_1)$	Azimutal angle between subleading p_T lepton and leading $CvsL$ jet	–	–	✓
$\Delta\phi(l_2, j_2)$	Azimutal angle between subleading p_T lepton and subleading $CvsL$ jet	–	–	✓
$\Delta\phi(l_1, p_T^{miss})$	Azimutal angle between leading p_T lepton and missing transverse momentum	–	✓	–

Table 2: Variables employed by the BDTG algorithm to train in each channel. The 2L channel is trained in two separate channels for the low VPt and high VPt cases.

5.3 Network architecture and Hyperparameter tunning

The chosen network architecture for this classification task is a fully connected network, like the one shown in figure 5.2. Unlike with the BDTG training, the DNN inputs are re-scaled into a Gaussian shape with mean zero and unity width, as it was tested to increase the training performance over other distributions. The chosen loss function is categorical cross-entropy with stochastic gradient descent as the optimizer. The activation function for the output layer is softmax, so the output values can be interpreted as probabilities.

Given the quantity of hyperparameters that are going to be tested in this research, the chosen method is random search. Bayesian optimization could have provided faster results, but its implementation in the TMVA framework was not possible.

- **Number of epochs:** {20, 21, ..., 200}. The full dataset should be passed to the algorithm multiple times, and the number of epochs determine the number of times it is passed.
- **Batch size:** {16, 32, 64, 128, 256}. The dataset is divided into batches that are sent to the neural network. The batch size hyperparameter determines how big these batches are.
- **Learning rate:** {0.1, 0.01, 10^{-3}, 10^{-4}, 10^{-5}}. It controls how much to change the

model each time it is updated.

- **Decay rate:** {"Yes", "No"}. Instead of using a fixed learning rate during the training, a scheduler can be used to vary its value as the training progress. If "decay rate" is applied, the learning rate decreases linearly with a factor dependant on the original learning rate and the total number of epochs. It is used to reduce overtraining.

- **Activation function:** {"tanh", "relu", "elu", "sigmoid"}. It defines the output of the nodes given a set of inputs. The different activation functions provide different responses to the input data.

- **Number of hidden layers:** {1, 2, ..., 7}. It determines the depth of the neural network. It represents the intermediate layers between the input and output layers of the network.

- **Neurons per layer:** {20, 21, ..., 250}. The number of neurons (or nodes) each layer has.

- **Dropout rate:** {0.0, 0.1, ..., 0.5}. To prevent overfitting, the network can randomly ignore some neurons. It makes the training process noisy, which forces nodes to take more or less responsibility. The dropout rate is the probability of at neuron to be ignored.

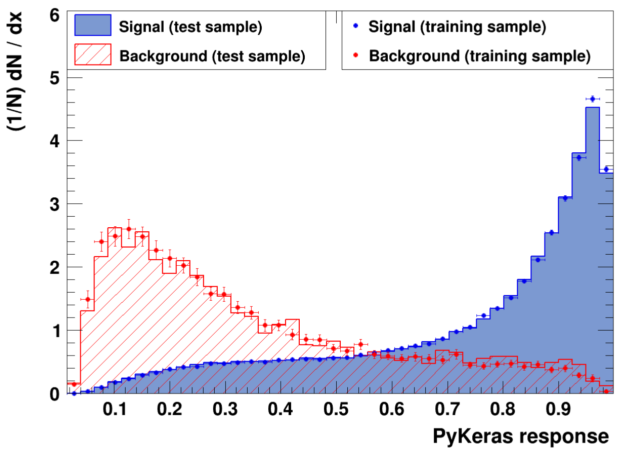
- **L^2 regularization:** $\{0.01, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}\}$. As with the dropout rate, the L^2 regularization is used to prevent overfitting by forcing a decay in the weights that do not contribute to the reduction of the objective function [39].

- **Momentum:** {0.5, 0.6, ..., 0.9}. To avoid local minima in the objective function, the momentum can increase the size of the steps. It adds a fraction of the previous weight update to the current one.

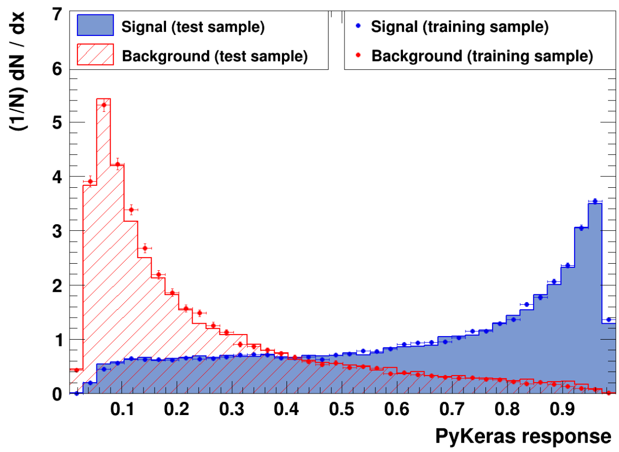
The hyperparameter tunning was performed by generating random combinations of the possibilities for a total of 1500 models for each channel. The area under the ROC curve is the parameter being compared by every combination tested.

5.4 Results

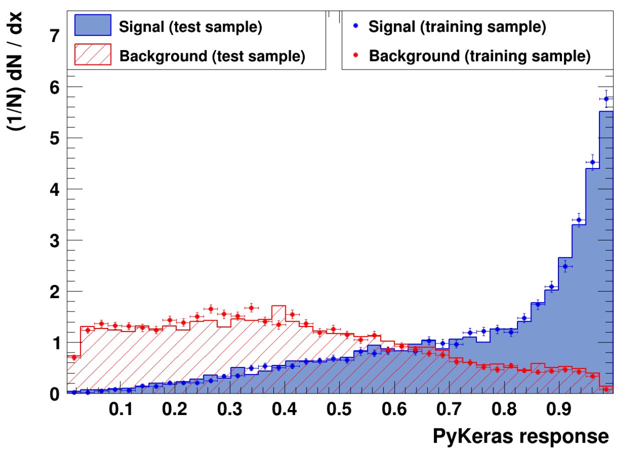
The hyperparameter tunning outputs the area under the curve along with several other analytics. The classifier responses of the best performing model for each channel are presented in figure 5.6. The overall shape in all channels resembles the ones previously seen for the BDTG method, with the exception of the two lepton Low VPt channel, where the background response is different. Since the parameter being optimized is the area under the curve, no optimization was seek in the distributions shape.



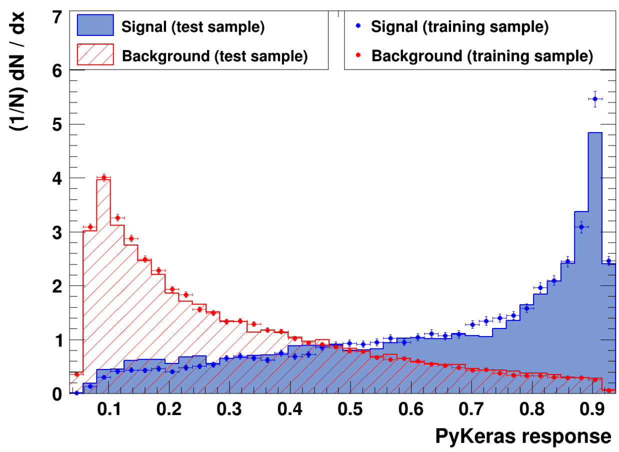
(a) 0L channel



(b) 1L channel



(c) 2LH channel



(d) 2LL channel

Figure 5.6: DNN response for the different channels. The signal sample is displayed in blue and the background in red. Overtraining is checked with the colored dots, which represent the DNN response to the training data. How similar they are is a sign of whether there is overfitting.

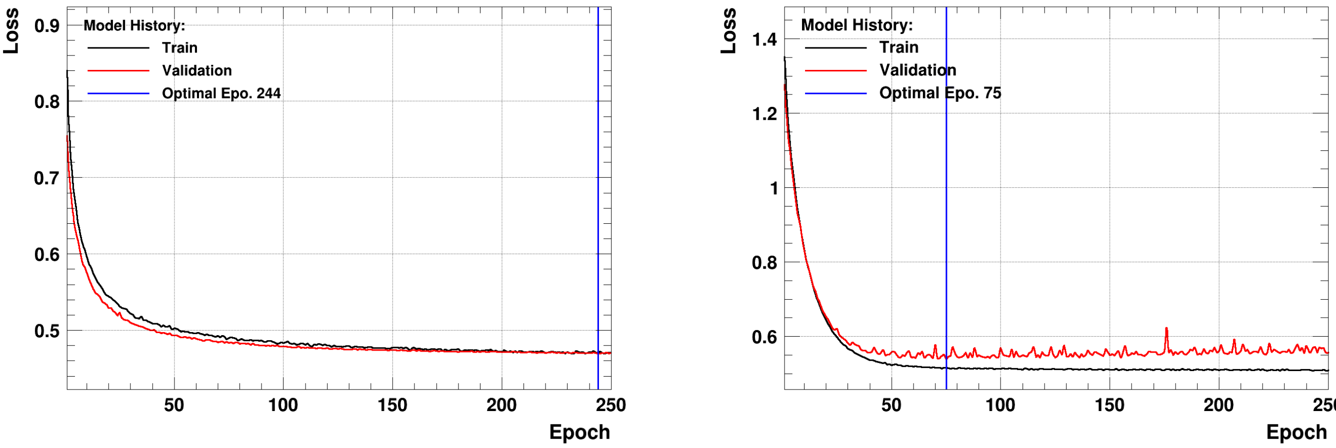
The hyperparameters that performed the best for each channel can be seen in table 3. There are a lot of similarities between the 0L, 1L and 2LL channels, leaving the 2LH channel with a different trend in hyperparameters. This might explain the different shape for the classifier response. Several analyses of the results showed that some hyperparameters had more relevance in the network efficiency, for example, decay rate or number of hidden layers had much less impact than activation function, dropout rate or learning rate.

In general, results show that no overtraining is present in any channel. Other tools to test for overtraining was to track the model's metrics (accuracy and loss) across epochs, which can be seen in figure 5.7. The objective function is used to validate a

Hyperparameter	0L	1L	2LH	2LL
Number of epochs	85	104	179	61
Batch size	256	128	16	256
Learning rate	0.01	0.01	0.0001	0.01
Decay rate	Yes	Yes	No	No
Activation function	relu	relu	elu	relu
Number of hidden layers	1	2	4	6
Neurons per layer	240	202	242	122
Dropout rate	0.3	0.3	0.1	0.1
L^2 regularization	0.001	0.001	0.000001	0.001
Momentum	0.7	0.8	0.6	0.5

Table 3: Best performing hyperparameters for each channel.

candidate solution for the algorithm. The calculated value is called loss and, during the training, the loss function aims to minimize it. When overfitting is present, the validation loss starts to increase even thought the loss value might still be decreasing. Figure 5.7a shows that the smallest validation loss achieved was near the end, so more epochs could improve the model's performance. On the other hand, figure 5.7b shows that for most of the training no improvement was achieved, and the number of epochs could have been reduced without loss in performance.



(a) 0L channel

(b) 2LL channel

Figure 5.7: Tracked loss (loss function applied to training data) and validation loss (loss function applied to validation data) during the training.

The final area under the curve results are shown in table 4, with minor improvements with respect to BDTG in every channel.

Sample	AUC - BDTG	AUC - PyKeras	Improvement
0 Lepton channel	0.853	**0.859**	+0.7%
1 Lepton channel	0.857	**0.863**	+0.7%
2 Lepton High channel	0.829	**0.841**	+1.4%
2 Lepton Low channel	0.825	**0.832**	+0.8%

Table 4: Area under the ROC curve results for BDTG and PyKeras.

Chapter 6:
Conclusions

In this dissertation, new and updated techniques were applied to improve the search for the standard model Higgs boson decaying to a pair of charm quarks. A least square kinematic fit was applied to the $ZH \rightarrow l^+l^-c\bar{c}$ decay channel achieving successful results. Using the transverse momentum conservation and the invariant mass of the two leptons as constraints, up to a 30% improvement in the mass resolution of the reconstructed Higgs boson was achieved. The application of the kinematic fit over background showed that it was not affected in an undesirable way. No mass sculpting appeared and the mass resolution of reconstructed particles also improved.

In order to better discriminate signal from background events, a deep neural network was trained to replace the current boosted decision tree method implemented. A hyperparameter search showed that better results are possible, but are marginally small compared to the efforts required to obtain them. The networks require more time both to run and to execute for an improvement of around 1%. These results may imply that there is no more complex correlations of features to extract from the input samples, and that a boosted decision tree can achieve the required level of complexity. Different network architectures or more input events could help to increase the final efficiency, but more research is needed.

Chapter A: KinFit explanation

To better explain the process to obtain the corrections applied in the kinematic fit, we will consider the case in which we have p unmeasured parameters $\vec{a}$. Equations 4.1 and 4.2 become:

$$f_k(\bar{a}_1, \bar{a}_2, ..., \bar{a}_p; \bar{y}_1, \bar{y}_2, ..., \bar{y}_n) = 0 \qquad for \quad k = 1, 2, ..., m \tag{A.1}$$

$$L(\vec{y}, \vec{a}, \vec{\lambda}) = S(\vec{y}) + 2\sum_{k=1}^{m} \lambda_k f_k(\vec{a}, \vec{y}) \tag{A.2}$$

Equation 4.3 should also include a term for the unmeasured parameters. This equation can be written in vector notation as:

$$\vec{f}^* + \mathcal{A}(\Delta\vec{a} - \Delta\vec{a}^*) + \mathcal{B}(\Delta\vec{y} - \Delta\vec{y}^*) \approx 0 \tag{A.3}$$

$$\mathcal{A} = \frac{\partial\vec{f}}{\partial\vec{a}} \qquad \mathcal{B} = \frac{\partial\vec{f}}{\partial\vec{y}} \tag{A.4}$$

With the introduction of the vector $\vec{c}$, which only depends on quantities from the previous iteration, the expression can be further reduced to:

$$\mathcal{A}\Delta\vec{a} + \mathcal{B}\Delta\vec{y} - \vec{c} = 0 \qquad \text{and} \qquad \vec{c} = \mathcal{A}\Delta\vec{a}^* + \mathcal{B}\Delta\vec{y}^* - \vec{f}^* \tag{A.5}$$

The likelihood can now be rewritten like:

$$L = \Delta\vec{y}^T\mathcal{V}^{-1}\Delta\vec{y} + 2\lambda^T(\mathcal{A}\Delta\vec{a} + \mathcal{B}\Delta\vec{y} - \vec{c}) \tag{A.6}$$

Equation A.6 is then differentiate with respect to $\vec{a}$, $\vec{y}$ and $\vec{\lambda}$ to get the conditions for an extremum:

$$\mathcal{V}^{-1}\Delta\vec{y} + \mathcal{B}^T\Delta\vec{\lambda} = 0 \tag{A.7}$$

$$\mathcal{A}^T\vec{\lambda} = 0 \tag{A.8}$$

$$\mathcal{B}\Delta\vec{y} + \mathcal{A}\Delta\vec{a} = c \tag{A.9}$$

Which can be written in matrix form:

$$\begin{pmatrix} \mathcal{V}^{-1} & 0 & \mathcal{B}^T \\ 0 & 0 & \mathcal{A}^T \\ \mathcal{B} & \mathcal{A} & 0 \end{pmatrix} \begin{pmatrix} \Delta\vec{y} \\ \Delta\vec{a} \\ \lambda \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ c \end{pmatrix} \tag{A.10}$$

The corrections can then be calculated by solving the system of equations. The solutions are:

$$\Delta\vec{y} = (\mathcal{V}\mathcal{B}^T\mathcal{V}_\mathcal{B} - \mathcal{V}\mathcal{B}^T\mathcal{V}_\mathcal{B}\mathcal{A}\mathcal{V}_\mathcal{A}{}^{-1}\mathcal{A}^T\mathcal{V}_\mathcal{B})\vec{c} \tag{A.11}$$

$$\Delta\vec{a} = (\mathcal{V}_\mathcal{A}-1\mathcal{A}^T\mathcal{V}_\mathcal{B})\vec{c} \tag{A.12}$$

$$\vec{\lambda} = (-\mathcal{V}_\mathcal{B} + \mathcal{V}_\mathcal{B}\mathcal{A}\mathcal{V}_\mathcal{A}{}^{-1}\mathcal{A}^T\mathcal{V}_\mathcal{B})\vec{c} \tag{A.13}$$

with:

$$\mathcal{V}_\mathcal{B} = (\mathcal{B}\mathcal{V}\mathcal{B}^T)^{-1} \qquad \mathcal{V}_\mathcal{A} = (\mathcal{A}^T\mathcal{V}_\mathcal{B}\mathcal{A}) \tag{A.14}$$

The new parameter is then $\vec{y}' = \vec{y} + \Delta\vec{y}$. The variance of the fitted parameter is calculated with error propagation from the covariance matrix. All this procedure is already implemented in the *KinFitter* package.

Chapter B: Standard Deviations

p_T \ $\lvert\eta\rvert$	< 1.4	$1.4 - 2.7$
$20 - 35$	6.1	7.5
$35 - 50$	8.6	10.0
$50 - 80$	11	13
$80 - 180$	17	19
$180 - 700$	25	25

(a) σ_m for p_T^c in GeV

p_T \ $\lvert\eta\rvert$	< 1.4	$1.4 - 2.7$
$20 - 35$	0.034	0.041
$35 - 50$	0.026	0.032
$50 - 80$	0.019	0.025
$80 - 180$	0.014	0.018
$180 - 700$	0.01	0.013

(b) σ_m for η^c

p_T \ $\lvert\eta\rvert$	< 1.4	$1.4 - 2.7$
$20 - 35$	0.036	0.039
$35 - 50$	0.028	0.031
$50 - 80$	0.022	0.025
$80 - 180$	0.015	0.018
$180 - 700$	0.011	0.012

(c) σ_m for ϕ^c

p_T \ $\lvert\eta\rvert$	< 1.48	$1.48 - 2.7$
$20 - 35$	1.8	2.3
$35 - 50$	2.5	3.1
$50 - 80$	2.8	3.7
$80 - 180$	3.4	5
$180 - 700$	5.2	7.8

(d) σ_m for p_T^{el} in GeV

p_T \ $\lvert\eta\rvert$	< 1.48	$1.48 - 2.7$
$20 - 35$	0.43	0.76
$35 - 50$	0.70	1.3
$50 - 80$	1.2	2.2
$80 - 180$	2.7	4.1
$180 - 700$	5.8	6.7

(e) σ_m for p_T^{mu} in GeV

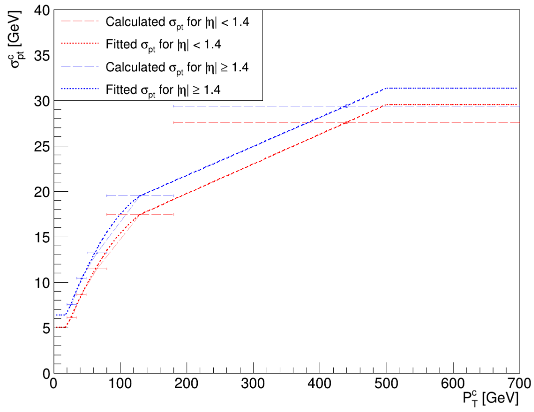
p_T \ $\lvert\eta\rvert$	< 1.48	$1.48 - 2.7$
$20 - 35$	0.00074	0.0018
$35 - 50$	0.00073	0.0017
$50 - 80$	0.00073	0.0017
$80 - 180$	0.0074	0.0016
$180 - 700$	0.00073	0.0016

(f) σ_m for η^l

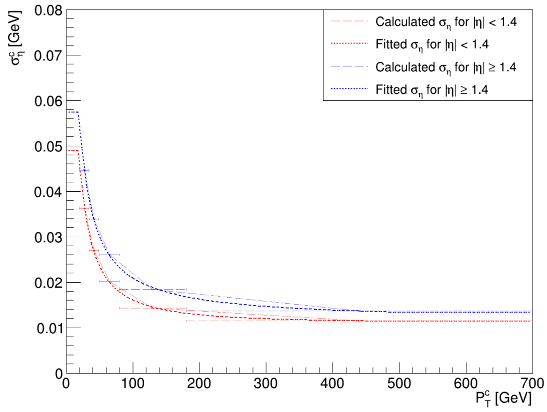
p_T \ $\lvert\eta\rvert$	< 1.48	$1.48 - 2.7$
$20 - 35$	0.0016	0.0018
$35 - 50$	0.0016	0.0018
$50 - 80$	0.0016	0.0017
$80 - 180$	0.0016	0.0017
$180 - 700$	0.0015	0.0016

(g) σ_m for ϕ^l

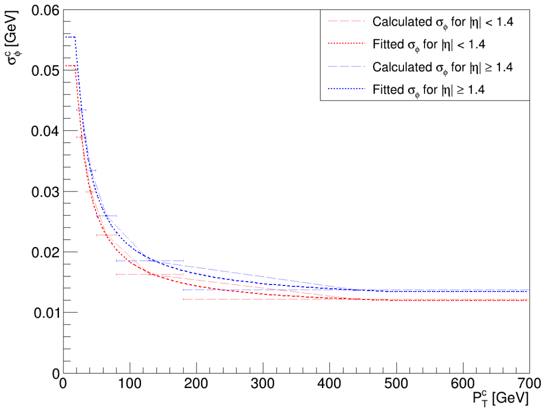
Table 5: Standard Deviations calculated for jets and leptons in different p_T (in GeV) and η regions.



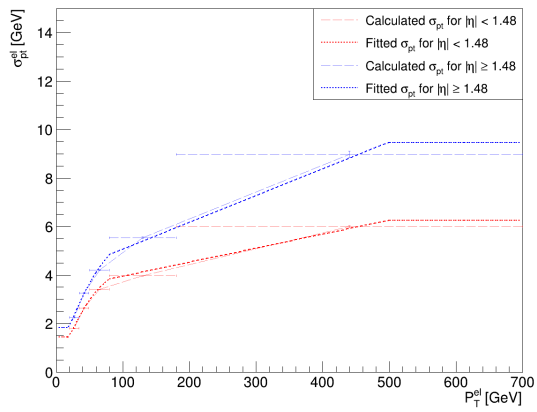
(a) σ_{Pt} parametrization for c-jets



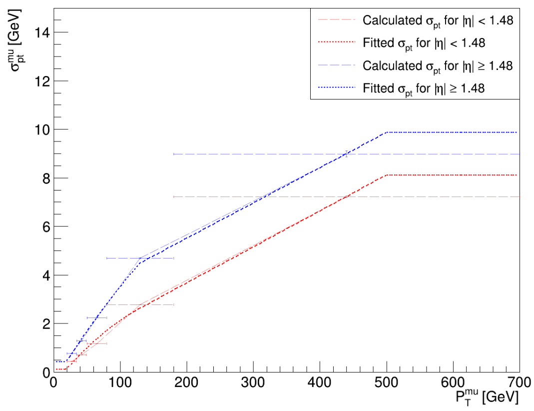
(b) σ_{η} parametrization for c-jets



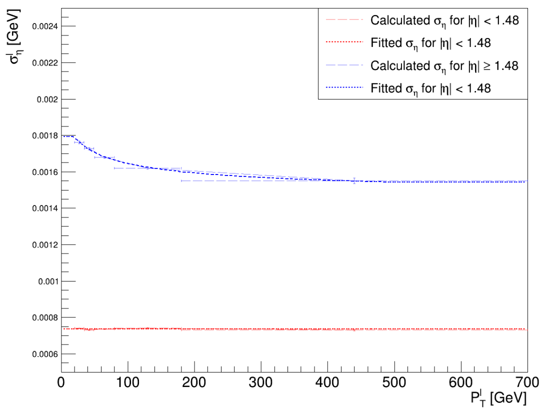
(c) σ_{ϕ} parametrization for c-jets



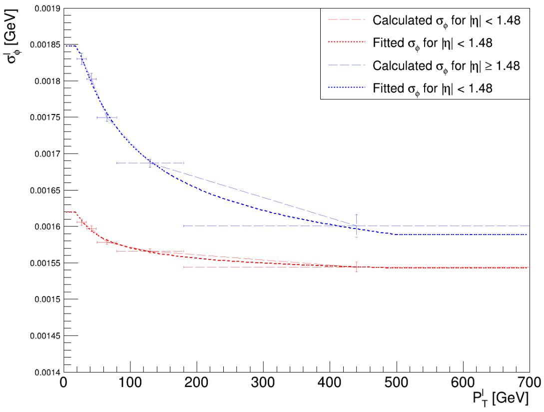
(d) σ_{Pt} parametrization for electrons



(e) σ_{Pt} parametrization for muons



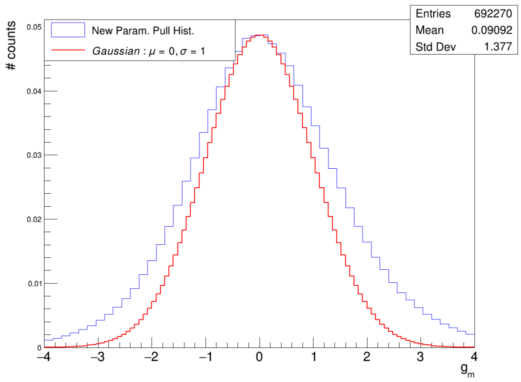
(f) σ_{η} parametrization for leptons



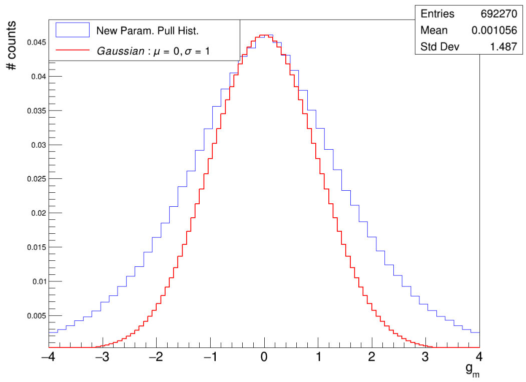
(g) σ_ϕ parametrization for leptons

Figure B.1: The σ_m parametrization that will be used in the analysis. The parametrization is calculated with respect to p_T, but in different η regions.

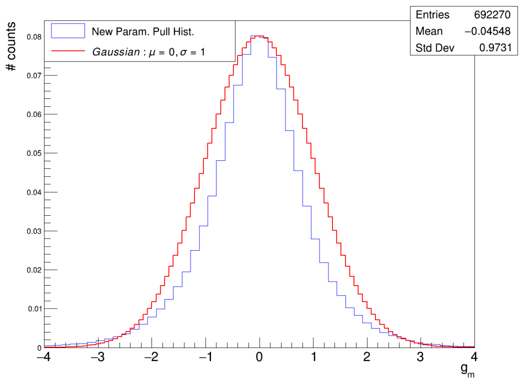
Chapter C:
Pull Histograms



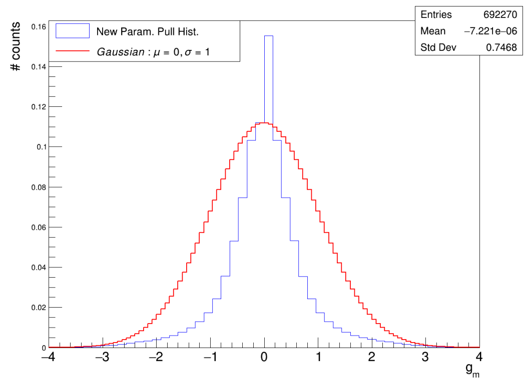
(a) g_m for p_T^c



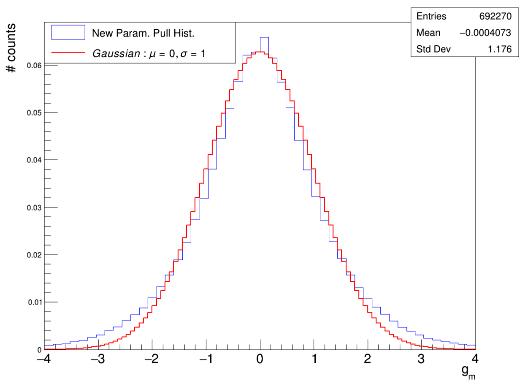
(b) g_m for ϕ^c



(c) g_m for p_T^l

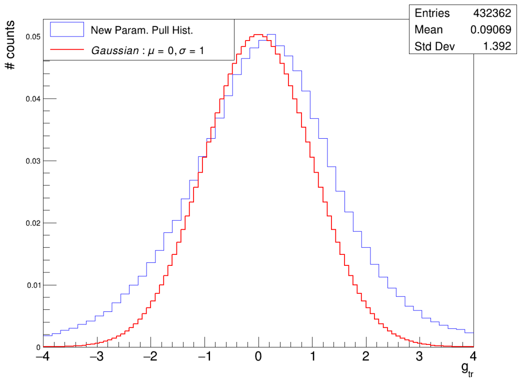


(d) g_m for η^l

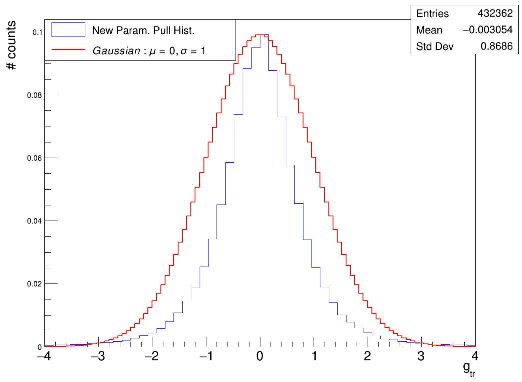


(e) g_m for ϕ^l

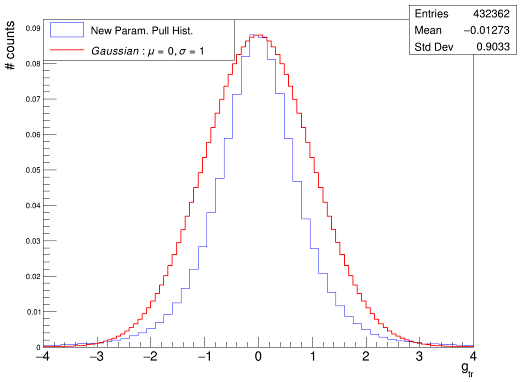
Figure C.1: Measured pull histograms (g_m) for the different parameters after the kinematic fit.



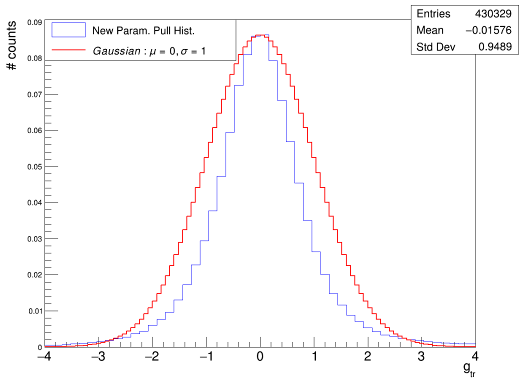
(a) g_{tr} for p_T^c



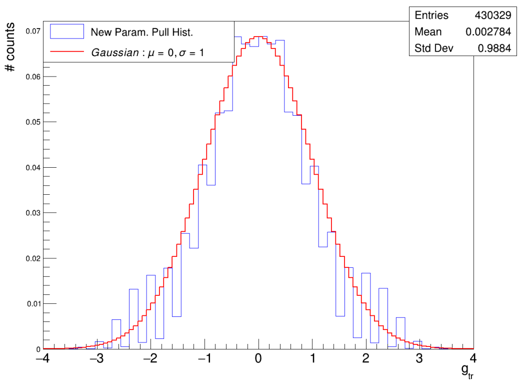
(b) g_{tr} for η^c



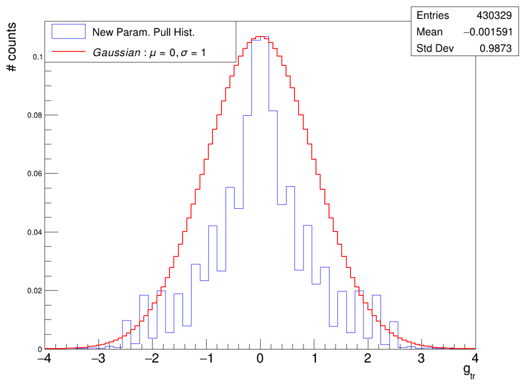
(c) g_{tr} for ϕ^c



(d) g_{tr} for p_T^l



(e) g_{tr} for η^l



(f) g_{tr} for ϕ^l

Figure C.2: Truth pull histograms (g_{tr}) for the different parameters after the kinematic fit.

References

[1] ATLAS collaboration, *"Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC"*, *Physics Letters B* **716** (2012) 1–29 `doi:10.1016/j.physletb.2012.08.020`.

[2] CMS collaboration, *"Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC"*, *Physics Letters B* **716** (2012) 30–61 `doi:10.1016/j.physletb.2012.08.021`.

[3] ATLAS collaboration, *"Observation and measurement of Higgs boson decays to WW^* with the ATLAS detector"*, *Physical Review D* **92** (2014) `doi:10.1103/PhysRevD.92.012006`.

[4] CMS collaboration, *"Observation of the Higgs boson decay to a pair of tau leptons with the CMS detector"*, *Physics Letters B* **779** (2018) 283–316 `doi:10.1016/j.physletb.2018.02.004`.

[5] CMS collaboration, *"Observation of $t\bar{t}H$ production"*, *Physical Review Letters* **120** (2018) `doi:10.1103/physrevlett.120.231801`.

[6] ATLAS collaboration, *"Observation of Higgs boson production in association with a top quark pair at the LHC with the ATLAS detector"*, *Physics Letters B* **784** (2018) 173 `doi:https://doi.org/10.1016/j.physletb.2018.07.035`.

[7] ATLAS collaboration, *"Observation of $H \to b\bar{b}$ decays and VH production with the ATLAS detector"*, *Physics Letters B* **786** (2018) 59–86 `doi:10.1016/j.physletb.2018.09.013`.

[8] CMS collaboration, *"Observation of Higgs Boson decay to bottom quarks"*, *Physical Review Letters* **121** (2018) `doi:10.1103/physrevlett.121.121801`.

[9] D. Guest, K. Cranmer and D. Whiteson, *"Deep Learning and its application to LHC Physics"*, *Annual Review of Nuclear and Particle Science* **68** (2018) 161–181 `doi:10.1146/annurev-nucl-101917-021019`.

[10] P. Avery, *"Vertexing and Kinematic Fitting Basic Theory"*, 1998. `url:http://www.phys.ufl.edu/~avery/fitting/kinfit_talk1.pdf`.

[11] LHC collaboration, *The Large Hadron Collider*. `url:https://home.cern/science/accelerators/large-hadron-collider`.

[12] CMS collaboration, *"The CMS experiment at the CERN LHC. The Compact Muon Solenoid experiment"*, *JINST* **3** (2008) S08004. 361 p `doi:10.1088/1748-0221/3/08/S08004`.

[13] S. Dris, *"Performance of the CMS Tracker Optical Links and Future Upgrade Using Bandwidth Efficient Digital Modulation"*, `1004.5574`.

[14] CMS collaboration, *"The CMS trigger system"*, *Journal of Instrumentation* **12** (2017) P01020–P01020 `doi:10.1088/1748-0221/12/01/p01020`.

[15] LHC collaboration, *LHC Higgs Cross Section Working Group*. `url:https://twiki.cern.ch/twiki/bin/view/LHCPhysics/LHCHXSWG`.

[16] LHC collaboration, *LHC Higgs Cross Section Working Group*. `url:https://twiki.cern.ch/twiki/bin/view/LHCPhysics/CERNYellowReportPageBR`.

[17] CMS collaboration, *"A search for the standard model Higgs boson decaying to charm quarks"*, *Journal of High Energy Physics* **2020** (2020) `doi:10.1007/jhep03(2020)131`.

[18] CMS collaboration, *"Particle-flow reconstruction and global event description with the CMS detector"*, *JINST* **12** (2017) P10003 `doi:10.1088/1748-0221/12/10/p10003`.

[19] M. Cacciari, G.P. Salam and G. Soyez, *"The anti-k_t jet clustering algorithm"*, *Journal of High Energy Physics* **2008** (2008) 063–063 `doi:10.1088/1126-6708/2008/04/063`.

[20] CMS collaboration, *"Identification of heavy-flavour jets with the CMS detector in pp collisions at 13 TeV"*, *Journal of Instrumentation* **13** (2018) P05011–P05011 `doi:10.1088/1748-0221/13/05/p05011`.

[21] CMS collaboration, *"Performance of the CMS missing transverse momentum reconstruction in pp data at $\sqrt{s} = 8$ TeV"*, *JINST* **10** (2015) P02006 `doi:10.1088/1748-0221/10/02/P02006`.

[22] E. Bols, J. Kieseler, M. Verzetti, M. Stoye and A. Stakia, *"Jet Flavour Classification Using DeepJet"*, 2020.

[23] ATLAS Collaboration collaboration, *"Search for the Decay of the Higgs Boson to Charm Quarks with the ATLAS Experiment"*, *Phys. Rev. Lett.* **120** (2018) 211802 `doi:10.1103/PhysRevLett.120.211802`.

[24] R. Brun and F. Rademakers, *"ROOT: An object oriented data analysis framework"*, *Nucl. Instrum. Meth. A* **389** (1997) 81 `doi:10.1016/S0168-9002(97)00048-X`.

[25] A. Hoecker, P. Speckmayer, J. Stelzer, J. Therhaag, E. von Toerne and H. Voss, *"TMVA: Toolkit for Multivariate Data Analysis"*, *PoS* **ACAT** (2007) 040.

[26] O. Zapata, S. Gleyzer and L. Moneta, *Python TMVA*, 2017. `url:http://oproject.org/pages/PyMVA.html`.

[27] F. Chollet et al., *Keras*, 2015. `url:https://keras.io`.

[28] M. Abadi, A. Agarwal, P. Barham, E. Brevdo, Z. Chen, C. Citro et al., *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*, 2015. url:https://www.tensorflow.org/.

[29] Particle Data Group collaboration, *"Review of Particle Physics"*, *Phys. Rev. D* **98** (2018) 030001 doi:10.1103/PhysRevD.98.030001.

[30] J. D'Hondt, S. Lowette, O.L. Buchmüller, S. Cucciarelli, F.-P. Schilling, M. Spiropulu et al., *"Fitting of Event Topologies with External Kinematic Constraints in CMS"*, Tech. Rep. CMS-NOTE-2006-023, CERN, Geneva (Jan, 2006).

[31] L.L. Luc Demortier, *"Everything you always wanted to know about pulls"*, *CDF/ANAL/PUBLIC/5776* (2008) .

[32] CMS collaboration, *Jet identification in high pile-up environment.* url:https://twiki.cern.ch/twiki/bin/viewauth/CMS/PileupJetID.

[33] H. Ritter, *"Improvement of the Higgs mass resolution in the $ZH \to l^+l^-b\bar{b}$ channel using a Kinematic Fit with constraints"*, Master's thesis, Bonn U., 2014.

[34] JME Group collaboration, *Jet Energy Resolution.* url:https://twiki.cern.ch/twiki/bin/viewauth/CMS/JetResolution.

[35] D. Bourilkov, *"Machine and deep learning applications in particle physics"*, *International Journal of Modern Physics A* **34** (2019) 1930019 doi:10.1142/s0217751x19300199.

[36] L. Deng and D. Yu, *"Deep Learning: Methods and Applications"*, Tech. Rep. MSR-TR-2014-21, Microsoft (May, 2014).

[37] K. Hornik, M. Stinchcombe and H. White, *"Multilayer feedforward networks are universal approximators"*, *Neural Networks* **2** (1989) 359 doi:https://doi.org/10.1016/0893-6080(89)90020-8.

[38] J. Snoek, H. Larochelle and R.P. Adams, *"Practical Bayesian Optimization of Machine Learning Algorithms"*, in *Advances in Neural Information Processing Systems 25*, F. Pereira, C.J.C. Burges, L. Bottou and K.Q. Weinberger, eds., pp. 2951–2959, Curran Associates, Inc. (2012), url:http://papers.nips.cc/paper/4522-practical-bayesian-optimization-of-machine-learning-algorithms.pdf.

[39] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*, MIT Press (2016).