GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN IN PUBLICA COMMODA SEIT 1737

Fakultät für Physik

Bachelor's Thesis

Untergrundmodellierung durch Normalisierende Flüsse in $t\bar{t}H(b\bar{b})$ mit Ein-Lepton Endzuständen bei $\sqrt{s} = 13$ TeV in ATLAS

Background modelling through Normalising Flows in $t\bar{t}H(b\bar{b})$ at One-Lepton Final States at $\sqrt{s} = 13$ TeV in ATLAS

prepared by

Paul Wollenhaupt

from Witzenhausen

at the II. Physikalischen Institut

Thesis number:	BSc-2023/08
Thesis period:	3rd April 2023 until 4th October 2023
First referee:	Prof. Dr. Arnulf Quadt
Second referee:	Prof. Dr. Max Wardetzky

Zusammenfassung

Diskrepanzen zwischen Daten- und Simulationsverteilungen werden auf Regionen mit hoher Jet- und b-Tag-Multiplizität extrapoliert, indem ein normalisierender Fluss als Korrektur nur auf die $t\bar{t}H$ Hintergrundprozessdaten angewendet wird. Dadurch wird die systematische Fehlmodellierung der Skalarsumme der transversalen Impulse der hadronischen Jets und die Ausgabe eines $t\bar{t}H$ klassifiziererenden, tiefen neuronalen Netzes in Regionen, die während des Trainings nicht gesehen wurden, wirksam reduziert. Es werden verschiedene Methoden für das Training des normalisierenden Fluss untersucht. Die endgültige Methode bringt außer der statistischen Unsicherheit der simulierten Stichproben keine signifikanten systematischen Unsicherheiten mit sich und übertrifft eine maschinenlernfreie Basismethode.

Stichworte : Datengesteuerte Hintergrundabschätzung, $t\bar{t}H$, Normalisierende Flüsse

Abstract

Discrepancies between data and simulation distributions are extrapolated to regions of high jet and b-tag multiplicity, by applying a normalising flow as correction to the $t\bar{t}H$ background samples only. This effectively reduced the systematic mismodelling of the scalar sum of the hadronic jet transverse momentum and the score of a deep neural network $t\bar{t}H$ classifier in regions not seen during training. Different methods for training the normalising flow are investigated. The final method does not introduce significant systematic uncertainties other than the statistical uncertainty of the simulated samples and outperforms a machine learning free baseline.

Keywords Data-driven background estimation, $t\bar{t}H$, Normalising flows

Contents

1 Introduction

From the profound reflections of ancient Greek philosophers who pondered the fundamental nature of matter and postulated the existence of indivisible particles known as atoms, a journey through the ages has led humankind into the fascinating realm of particle physics. The story of the search for the indivisible building blocks of the universe is a testament to humankind's insatiable curiosity and relentless quest to understand the fundamental nature of the universe.

Throughout history, the concept of the atom has evolved, with influential figures such as John Dalton in the 19th century proposing the idea of atoms as distinct, indivisible entities that combine to form compounds [1, 2]. However, it wasn't until the turn of the 20th century that the true nature of the atom began to be revealed. In 1897, J.J. Thomson's groundbreaking experiments with cathode rays led to the discovery of the electron, a tiny negatively charged particle that makes up the shell of the atom. This revelation paved the way for the development of the atomic model. Ernest Rutherford's famous gold foil experiment in 1909 provided a further insight [3]. It showed that atoms are mostly empty space with a small, dense nucleus at their core. Rutherford's model introduced the concept of protons in the nucleus, while the subsequent discovery of neutrons by James Chadwick completed the trinity of subatomic particles [4].

As scientific exploration continued, it became increasingly clear that even these subatomic particles, once thought to be the ultimate building blocks of matter, were not really fundamental. In the mid-20th century, the advent of high-energy particle accelerators allowed scientists to probe deeper into the subatomic realm. They discovered a plethora of new particles, such as mesons and baryons [5–8], which challenged the simplicity of the atomic structure. The Standard Model of particle physics, developed in the second half of the 20th century, provided a comprehensive framework for describing the interactions and properties of particles thought to be fundamental [9–26].

1.1 The Standard Model of Particle Physics

The Standard Model (SM) is currently the most successful theory in particle physics [9–26], having made several successful predictions, including the discovery of the Higgs boson by the ATLAS and CMS collaborations in 2012 [27, 28] and for example the precision measurement of the anomalous magnetic dipole moment of the muon [29]. It predicts elementary particles in two categories: 12 fermions (f) and the 5 bosons, the photon, gluon, the $W^{\pm}$ and Z bosons, and the Higgs bosons. Fermions comprise the building blocks of matter, while bosons mediate the fundamental forces. There are three generations of fermions, with each generation being more massive than the one before it. All of them have corresponding antiparticles ($\bar{f}$) with opposite charge and are further categorized into quarks (q) and leptons (ℓ). See Table 1.1 for an overview.

Quarks are further categorised according to their electric charge. The up (u), charm (c) and top quarks (t), from lightest to heaviest, have an electric charge of $+2/3e$ and are called up-type quarks. Conversely, down-type quarks have an electric charge of $-1/2e$ and make up the down, strange and bottom quarks. In addition, each type of quark has three versions with different colour charges: red, green and blue. The corresponding antiquarks ($\bar{q}$) have the corresponding anticolours: antired, antigreen and antiblue. Leptons come in two varieties: electrically neutral or with electric charge $-1e$, where e stands for the elementary electric charge. The second and third generation counterparts of the electron are the muon and the tau. Each charged lepton corresponds to a type of neutrino, which has no electric charge and rarely interact with matter. Any pair of fermions from the same generation with a property called '*left-handedness*" form a weak isospin doublet. '*Right-handed*" fermions do not pair up, and form singlets instead.

In SM, interactions between any particles are mediated by bosons. The photon (γ) is the force carrier of the electromagnetic force, which governs the interaction of electrically charged particles. It is massless and its own antiparticle. The strong force is mediated by gluons (g), which bind quarks together to form colour-neutral hadrons. There are eight distinct gluon types for every possible combination of a quark color with a different anti-quark color. The Z and $W^{\pm}$ bosons are involved in the weak force, which is the only force that may change the flavour of quarks. The Z boson is its own antiparticle and therefore electrically neutral, while the $W^{\pm}$ bosons have electric charge $\pm 1e$. The weak force primarily acts on left handed fermions. The Higgs boson is responsible for giving mass to the other bosons via the Higgs mechanism and to the fermions via the Yukawa coupling [10–12]. It is noteworthy that the field underlying the Higgs boson is the only one with a positive vacuum expectation value.

Type	El. charge	1st Gen.	2nd Gen.	3rd Gen.
Leptons	$-1e$	electron e	muon μ	tau τ
	$0e$	electron neutrino ν_e	muon neutrino ν_μ	tau neutrino ν_τ
Quarks	$+2/3e$	up u	charm c	top t
	$-1/3e$	down d	strange s	bottom b

Table 1.1: Fermions of the Standard Model organised by generation and electric charge.

1.2 The ATLAS Detector at the Large Hadron Collider

To test the properties of and discover the existence of the heavier particles within the Standard Model, experiments at high energy levels are needed. Particle accelerators are therefore essential. The Large Hadron Collider (LHC), a large circular synchrotron accelerator at CERN (the European Organisation for Nuclear Research) in Geneva, plays a crucial role in these investigations [30]. The LHC accelerates protons to almost the speed of light and then collides them in the detectors of international collaborations, allowing scientists to observe and analyse the results of these collisions. The ATLAS detector is one of the two general-purpose detectors in the LHC [31]. It is made up of several different layers and components, each serving a specific purpose in particle detection and analysis. Some parts are illustrated in Fig. 1.1.

Inner Detector

The Inner Detector (ID) is the innermost part of the ATLAS detector, closest to the collision point, and is responsible for the precise tracking and identification of charged particles [32]. It consists of three sub-detectors. The Pixel Detector is the innermost component of the ID and is known for its exceptional spatial resolution. It is made up of several layers of pixel sensors, which are incredibly fine detectors that can pinpoint exactly where charged particles pass through, allowing scientists to reconstruct the paths of the particles with micrometre accuracy. This information is vital for identifying the types of particles being produced. For example, B hadrons, which contain a bottom quark, have relatively long lifetimes compared to other particles produced in the collisions. As a result, B hadrons can travel a macroscopic distance before decaying. The high spatial resolution of the pixel detector makes it possible to identify the B hadrons by detecting the secondary vertices they produce when they decay. This identification is called "*b-tagging*".

Located just outside the pixel detector, the Semiconductor Tracker (SCT) extends the

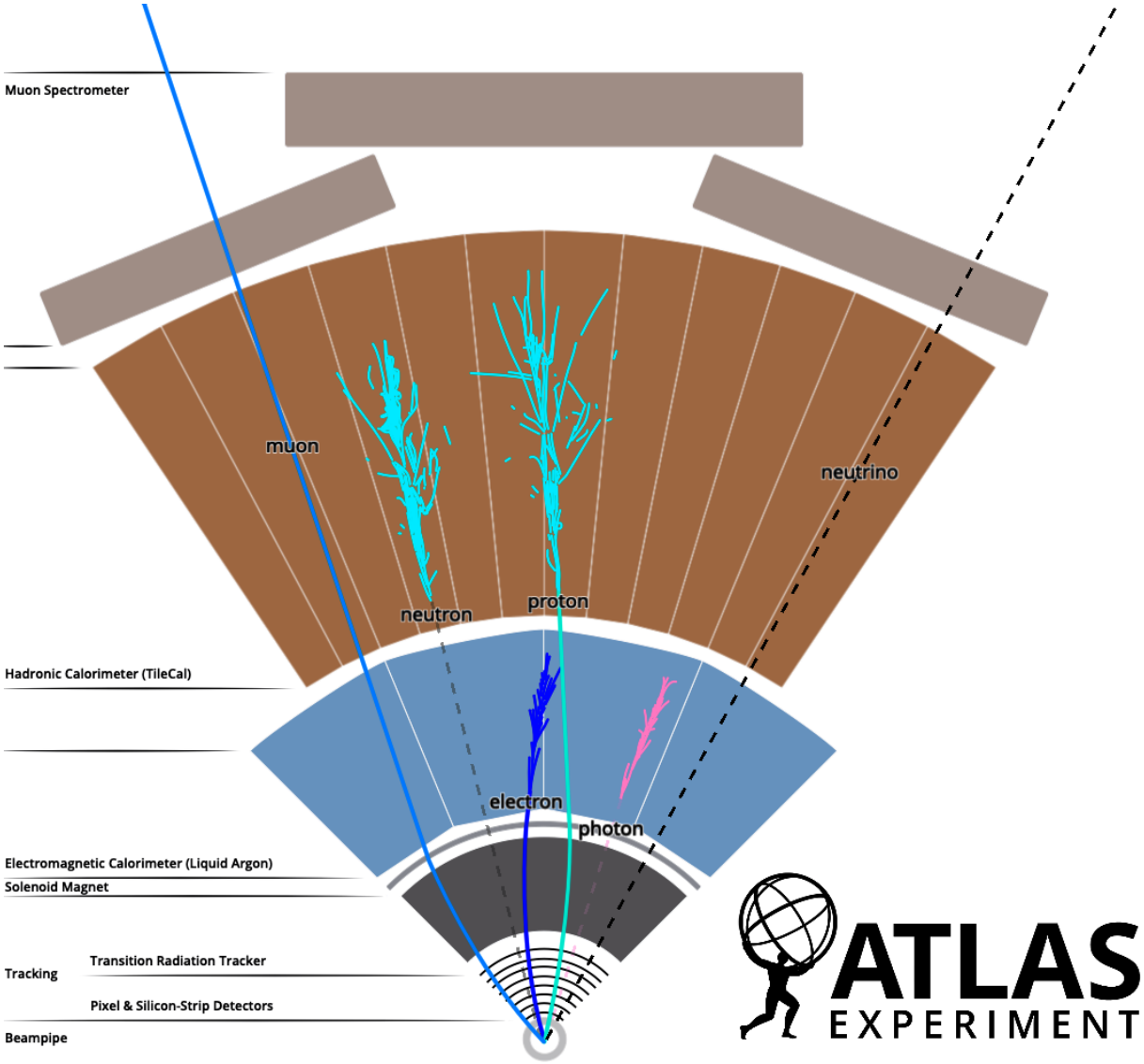

Figure 1.1: Schematic slice of the ATLAS detector, and visualisations of different particles (© CERN). From the inside out: The inner detector in dark grey, then the electromagnetic calorimeter in light blue, after that the hadronic calorimeter in brown and finally the muon spectrometer in beige. The dark blue line stopping in the ECAL shows a electron starting a electromagnetic jet. The photon in pink, also stops in the ECAL. Protons and neutrons (shown in light teal) are both stopped in the HCAL, with the proton path bending slighly becaus of its electric charge. Muons are only bent slightly and go through the whole detector. Neutrinos, shown in the black dashed line, do not typically interact with the detector.

tracking capabilities of the inner detector further outwards. It uses silicon strip sensors arranged in a barrel-like structure and endcap discs. As charged particles pass through these sensors, they leave traces of electrical signals that can be used to reconstruct the particle's trajectory. The SCT provides additional information on particle momentum and charge, complementing the data from the Pixel Detector. The momentum can be deduced because the paths of the charged particles are bent by a magnetic field passing through the detector, and the radius of curvature is proportional to the momentum.

The Transition Radiation Tracker (TRT) is the outermost component of the Inner Detector and uses a different tracking technology. It consists of straw tubes filled with a gas mixture. As charged particles pass through these tubes, they ionise the gas, producing electrical signals that can be detected. The TRT is particularly useful for identifying electrons and distinguishing them from charged pions.

Electomagnetic Calorimeter

The next detector component is the Electromagnetic Calorimeter (ECAL) [33]. Its primary function is to accurately measure the energy of electrons and photons by detecting the electromagnetic showers they produce when interacting with the lead absorber material. When an energetic electron or photon enters the ECAL, it undergoes multiple scatterings, creating a cascade of secondary particles. These secondary particles ionise the liquid argon, producing electrical signals that are detected by sensitive electrodes inside the ECAL. The ECAL helps distinguish electrons and photons from charged hadrons, which produce less localised clusters of energy deposits. The high energy resolution of the ECAL played a crucial role in the discovery of the Higgs boson, where it helped to identify rare events in which Higgs bosons decayed into pairs of photons [27, 28].

Hadronic Calorimeter

The Hadronic Calorimeter (HCAL) is designed to measure the energy of hadrons and is located outside and around the ECAL [34]. The HCAL is constructed similarly to the ECAL, with alternating layers of dense absorber material and active detector material. The dense absorber material is typically made of steel or brass, while the active detector material is scintillating plastic or tiles that emit flashes of light when charged particles pass through them.

The energy of the hadrons is measured by stopping and absorbing them in the dense absorber material. As the hadrons interact with the absorber material, they produce secondary particles which deposit their energy in the active detector material as they pass

through. The light flashes produced in the active medium are then detected and converted into electrical signals. One of the key applications of the HCAL is the reconstruction of hadronic jets, which are cones of particles produced when a high-energy quark or gluon creates a cascade of hadrons in a process called hadronisation. The HCAL makes it possible to measure the total energy of these jets, which helps to study processes involving strong interactions. For the analysis of jets energy deposits are clustered by for example the anti-k_t algorithm [35]. Using particle flow algorithms the energy deposits of reconstructed particles can be subtracted [36].

Other parts of the ATLAS detector include the muon spectrometer [37], which measures the trajectories and momenta of muons, the electronics and trigger system [38], which process and select collision events for further analysis, and detectors for forward physics [39], which specialise in measuring particles produced at small angles relative to the beamline.

1.3 The $t\bar{t}H$ Process

The Yukawa coupling between the Higgs boson and the fermions is proportional to the mass of the fermions [16]. Since the top quark is the heaviest known fermion, studying the Higgs-top coupling is naturally a good test for the SM. This could be done by studying the decay of a Higgs boson into a top-antitop pair. However, this decay mode is kinematically very unlikely, since a top quark is already heavier than the Higgs boson. The interaction vertex that produces a Higgs boson from a top-antitop quark pair, the time-reversed version of the decay mentioned above, is therefore a better candidate for measuring the Higgs-top coupling. This interaction typically occurs when a Higgs boson is produced in association with a top quark pair, a process called $t\bar{t}H$. This process was observed in 2018 by the ATLAS and CMS collaborations [40, 41]. A possible Feynman diagram for this process is shown on the left of Fig. 1.2. The $t\bar{t} \to H$ interaction is circled in blue. The top quarks are produced by gluons coming from the protons accelerated at the LHC.

The most important background for the $t\bar{t}H$ production with the Higgs boson decaying into a bottom quark-antiquark pair is the $t\bar{t}b\bar{b}$ process, since it shares possible final states and is kinematically very similar [42]. A Feynman diagram of the $t\bar{t}b\bar{b}$ process, similar to the one for the $t\bar{t}H$ process, is shown in Fig. 1.2 on the right. From the Feynman diagrams it can be deduced that for both processes, if only one W boson decays leptonically, at least 6 jets are expected, of which at least 4 are b-tagged, since 6 quarks are in the final state, of which 4 are defined as bottom flavour. More quarks can be produced by the hadronic decay of the other W boson or by the radiation of a gluon by the strong force,

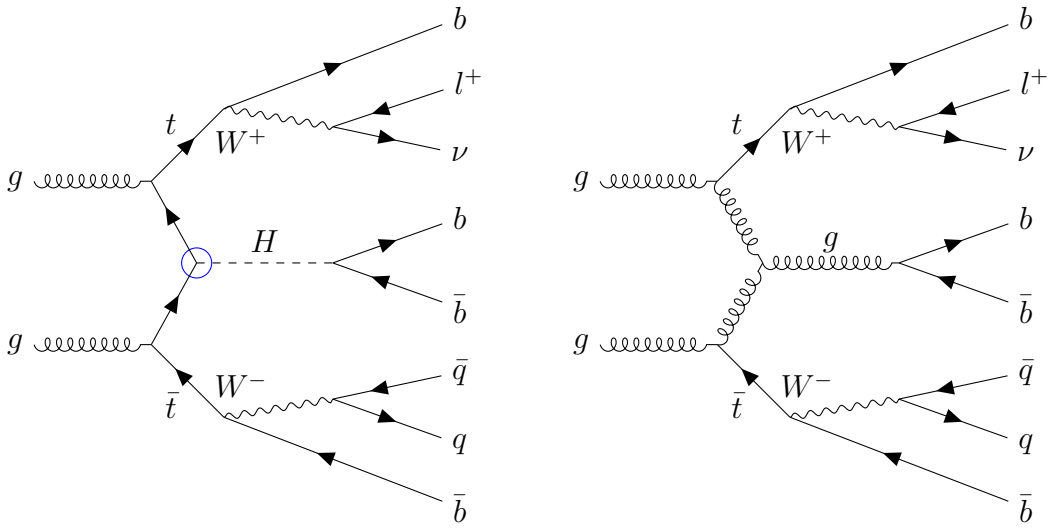

Figure 1.2: A possible Feynman diagram for the $t\bar{t}H$ process is shown on the right. The top quarks are predicted to decay with almost 100% probability into a W boson and a bottom quark [43]. The W bosons can decay into a quark-antiquark pair or into a charged lepton and its corresponding neutrino. Both decays are shown. The Higgs boson is most likely to decay into a bottom quark-antiquark pair, with a probability of about 58% [44]. On the right is a Feynman diagram for the $t\bar{t}b\bar{b}$ process, the main background. It can share the exact final states. Here, the bottom quark-antiquark pair is produced by the strong force rather than by the decay of the Higgs boson.

which then decays into quarks. Other important background processes are the associated production of a top quark-antiquark pair with only one b-jet, called $t\bar{t}b$, or a $t\bar{t}b\bar{b}$ event with very collinear b jets that cannot be resolved as two distinct b-jets, called $t\bar{t}B$. The production of $t\bar{t}$ associated with a jet of lighter flavour, i.e. jets originating from a charm quark, called $t\bar{t}c$, or from lighter quarks, called $t\bar{t}+\text{light}$, are also a source of background events.

2 Machine Learning

In an era marked by an unprecedented influx of data across diverse scientific disciplines, the need for innovative tools and techniques to extract meaningful insights has never been more pressing. This chapter delves into the realm of *Machine learning* (ML), a powerful and versatile approach that holds immense potential for accelerating scientific discovery.

ML algorithms circumvent the problem, that designing algorithms by hand is not feasible for a large class of tasks, by explicitly searching for functions that solve a given task. This process is often called "*learning*". ML is particularly good at tasks where the performance of a candidate can be evaluated over large amounts of data, for example by comparing the output of the current candidate to a known solution [45].

2.1 Neural Networks

Artificial Neural Networks (NNs) are a way of specifying the space of functions to be considered by a learning algorithm. A NN architecture converts a vector of parameters $\vec{\theta} \in \mathbb{R}^{|\vec{\theta}|}$ into a possible solution, i.e. vector valued functions for some $n, m \in \mathbb{N}$:

$$f : \mathbb{R}^{|\vec{\theta}|} \to (\mathbb{R}^n \to \mathbb{R}^m), \quad \vec{\theta} \mapsto f_{\vec{\theta}}. \tag{2.1}$$

This makes it possible to represent the class of considered functions by their parameters $\vec{\theta}$ and to reformulate the learning problem as a search for suitable parameters $\vec{\theta}^*$.

The simplest NN architecture is the perceptron [46], which takes the form a simple linear function $f_{\vec{w}}(\vec{x}) = \vec{w}^\intercal \vec{x}$. Stacking several perceptrons yields the multilayer perceptron (MLP):

$$f_{\left(\vec{W}^{(1)},\ldots,\vec{W}^{(k)}\right)}(x) = \vec{W}^{(k)}(\sigma(\vec{W}^{(k-1)}(\ldots\sigma(\vec{W}^{(1)}(x))))), \tag{2.2}$$

where $\vec{W}^{(i)}$ denotes the ith layer's weight matrix and σ is a non-linear activation function that is applied element-wise [47]. The activation function allows a sufficiently wide MLP to approximate any continuous function [48]. Commonly used activation functions are shown in Figure 2.1.

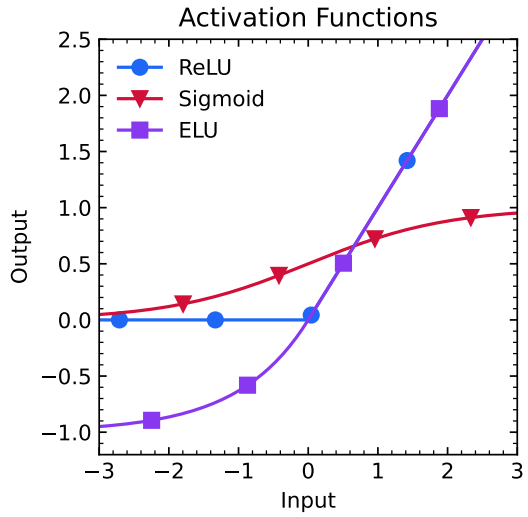

Figure 2.1: Common activation functions for neural networks: The most common activation function is the Rectified Linear Unit, which is shown in blue with circle markers. The sigmoid function, which was used during the initial phase of machine learning with neural networks, is shown in red with triangle markers. The Exponential Linear Unit looks similar to the ReLU function, but has a continous derivative and a negative asymptote. It is shown in pruple with square markers.

2.2 Optimisation

The goal of a machine learning procedure is usually stated by defining a cost function, also called loss function $\mathcal{L} : \mathbb{R}^{|\vec{\theta}|} \to \mathbb{R}$, where a lower loss indicates a better solution. Loss functions for NNs are usually required to be differentiable because most neural networks are trained by variants of gradient descent, which is an iterative optimisation algorithm, that uses linear approximations to update the current parameter estimate $\vec{\theta}_t$ in the direction of the steepest descent of the loss function:

$$\vec{\theta}_{t+1} = \vec{\theta}_t - \eta \cdot \nabla_{\vec{\theta}_t} \mathcal{L}(\vec{\theta}_t), \tag{2.3}$$

where $\eta \in \mathbb{R}^+$ is the update scale, also called the "*learning rate*". Adam is the most widely used variant of gradient descent for training neural networks [49], mainly due to its practical trade-off between memory between memory usage, convergence speed and stability. It achieves this by keeping a running average of the first and second moments of the gradient, which are then used to update the parameters with the additional information from previous iterations.

2.3 Generalisation

In most practical applications, what would be the ideal/true loss function cannot be calculated, but only an approximation of it, called the training loss. This is usually due to the fact that the ideal loss function is defined as an expectation over the data distribution, which is approximated by a finite set of training data points. A model

whose training loss is close to the true loss is said to generalise well. If the training loss is much smaller than the true loss, the model is said to overfit. Since the true loss may not be accessible, the model is usually evaluated on a statistically independent set of data points, called the validation set.

In many cases, a simple relationship between validation loss and model complexity, called the bias-variance trade-off, can be observed. First, the validation loss decreases with increasing model complexity because the model is able to fit the training data better. Therefore, model capacity is sometimes deliberately reduced by adding a regularisation term to the loss function that penalises large parameter values. If the model has less capacity than optimal, it is said to be underfitting [45]. The trade-off is illustrated in Figure 2.2.

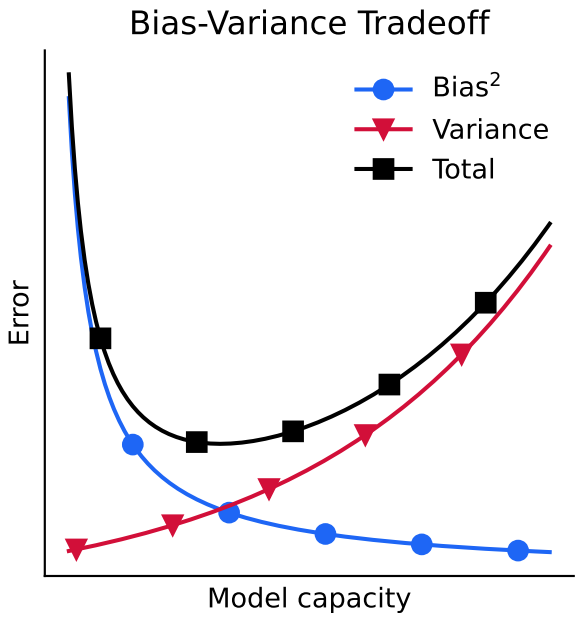

Figure 2.2: Illustration of the bias-variance trade-off, the model complexity increases from left to right, resulting in the bias squared (blue circles) decreasing and the variance (red triangles) increasing. The total error (black squares) has a unique minimum in the center.

A good illustration of this principle is the case of training a function $\hat{f}$ to approximate observations of a true function f with added zero mean, σ^2 variance noise ε, the expected mean squared error can be decomposed into three terms: the squared bias, the variance and the irreducible error σ^2:

$$\begin{aligned} \mathrm{E}\left[\mathcal{L}(x)\right] &= \mathrm{E}\left[\left(f(x) + \varepsilon - \hat{f}(x)\right)^2\right] && (2.4)\\ &= \mathrm{E}\left[\hat{f}(x) - f(x)\right]^2 + \mathrm{E}\left[\left(\mathrm{E}\left[\hat{f}(x)\right] - \hat{f}(x)\right)^2\right] + \mathrm{E}\left[\varepsilon^2\right] && (2.5)\\ &= \mathrm{Bias}\left[\hat{f}(x)\right]^2 + \mathrm{Var}\left[\hat{f}(x)\right] + \sigma^2, && (2.6) \end{aligned}$$

for some fixed x. A full derivation of Eq. 2.5 is done in Ref. [50]. The bias can be inter-

preted as the error due to the model's inability to represent the true function, while the variance is the error due to the model's sensitivity to the training data's noise. However, there is some recent evidence against this simple relationship between model complexity and validation performance [51].

2.4 Generative Models

As Richard Feynman aptly stated: "What I cannot create, I do not understand.". Generative machine learning embodies this principle by applying ML techniques to the task of generating new data. The typical goal of generative modelling is to infer a probability distribution from a set of independent and identically distributed (i.i.d.) samples. This can be done by explicitly modelling the distribution, or implicitly by searching for a function that takes in a random variable of a known distribution and outputs a sample from the desired distribution.

2.4.1 Maximum Mean Discrepancy Optimisation

Maximum mean discrepancy optimisation is an implicit generative modelling technique [52]. It involves minimising a distance between the target and the model distribution based on i.i.d. samples from both of them. The distance function is called the *maximum mean discrepancy* (MMD) [53]. The MMD is defined for probability measures p, q on a measureable space $\mathcal{X}$, with a characteristic kernel function $k : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ as

$$\mathrm{MMD}_k^2(p, q) = \mathrm{E}_{x,x' \sim p}\left[k(x, x')\right] + \mathrm{E}_{y,y' \sim q}\left[k(y, y')\right] - 2 \cdot \mathrm{E}_{x \sim p, y \sim q}\left[k(x, y)\right]. \tag{2.7}$$

For training, the MMD^2 is used as the loss function, where p is the target distribution and q is the model distribution. Note that the MMD requires $\mathcal{O}(n^2)$ operations to compute, where n is the number of samples from each distribution.

2.4.2 Normalising Flows

Normalising flows search for a differentiable, bijective function $f_{\vec{\theta}}$, that transforms a random variable z with a known probability distribution p_z into a random variable x with a desired probability distribution p_x. The change of variables formula allows the calculation of the probability density function p of $f_{\vec{\theta}}(z)$ [54]:

$$p(f(z)) = p(z) / \left|\det(\mathrm{J}_z\, f_{\vec{\theta}}(z))\right|, \tag{2.8}$$

where $\mathrm{J}_z\, f_{\vec{\theta}}(z)$ denotes the Jacobian of $f_{\vec{\theta}}$ at z. An example is shown in Figure 2.3. This makes it possible to calculate the likelihood of a given sample and therefore also to maximise the log likelihood of the model distribution:

$$\mathcal{L}(\vec{\theta}) = \mathrm{E}_{z\sim p_z}\left[\log p(f_{\vec{\theta}}(z))\right]. \tag{2.9}$$

Although computing the determinant of the Jacobian is computationally infeasible for larger unstructured matrices. Therefore, f is usually decomposed into a series of invertible transformations f_i with tractable Jacobians: $f_{\vec{\theta}} = f_n \circ \ldots \circ f_1$, where the Jacobian is either diagonal or triangular. The resulting model distribution can be calculated by applying the variable change formula repeatedly:

$$\log p(f_{\vec{\theta}}(z)) = \log p(z) - \sum_{i=1}^{n} \log\left|\det(\mathrm{J}_z\, f_i(z))\right|. \tag{2.10}$$

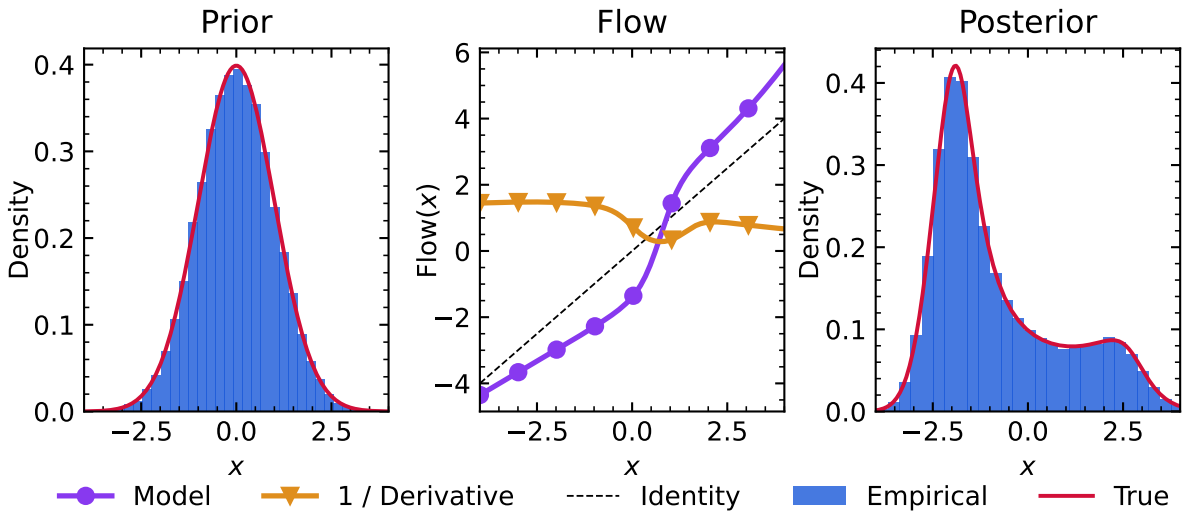

Figure 2.3: Example of a randomly initialised sigmoidial flow with a 1D standard normal distribution as prior [55]. The posterior density is calculated using the change of variables formula.

2.4.3 Autoregressive Models

Autoregressive models are a class of generative models that model a joint distribution by decomposing it into a product of conditional probabilities [56]:

$$p(x_1, \ldots, x_n) = \prod_{i=1}^{n} p(x_i \mid x_1, \ldots, x_{i-1}). \tag{2.11}$$

Normalising flows can be used to model the conditional probabilities $p(x_i \mid x_1, \ldots, x_{i-1})$. This needs each individual flow $f_i(x_i \mid x_1, \ldots, x_{i-1})$ to be invertible only in x_i, which significantly reduces the complexity of the architecture, allows for parallelisation of training and makes the Jacobian triangular or block triangular, depending on the whether x_i is a scalar or a vector [57].

The generative pre-trained transformer, widely used in natural language processing, is another example of an autoregressive model [58] that benefits significantly from parallelising the training process. The generation of new samples from the model distribution is done by sampling from the first distribution $p(x_1)$ and then successively from the conditional distributions $p(x_i \mid x_1, \ldots, x_{i-1})$ for $i = 2, \ldots, n$.

2.4.4 Generative Adversarial Networks

Generative Adversarial Networks (GANs) are a class of implicit generative models that are optimised by simulating a zero-sum, two-player game between a generator and a discriminator [59]. The goal of the discriminator is to discriminate between samples from the target distribution and samples from the generator distribution, while the goal of the generator is to generate samples that the discriminator does not flag as false. Gradient based updates are applied alternately to the discriminator and the generator, while keeping the parameters of the other network fixed. It should be noted that the training of GANs is often unstable and the resulting model often does not capture all aspects of the target distribution [60].

3 Data-Driven Background Estimation

Data-driven background estimation methods use experimental data in subsets of the phase space called *control regions* (CR) to predict the contribution of background events in disjoint *signal regions* (SR). This is particularly useful in cases where the background is not accurately modelled by simulations, which can occur due to approximations that must be made in light of computational constraints (e.g. in Ref. [61]). Data-driven background estimation has been successfully applied to many different high-energy particle physics analyses, including the successful searches for the top quark [62] and the Higgs boson [27, 28].

3.1 Background Yield Estimation

The simplest method for estimating the number of background events in a given region is the ABCD method. Three orthogonal CRs (B, C, D) and one disjoint SR (A) are defined by two binary variables x and y. If the variables are independent and the number of signal events in the CRs is negligible, the number of background events in the SR can be estimated by

$$N_{\mathrm{SR}} = \frac{N_{\mathrm{C}}}{N_{\mathrm{B}}} \cdot N_{\mathrm{D}} \,, \tag{3.1}$$

The statistical uncertainty of this estimate can be calculated using the standard error propagation formula. If at least one of the variables is defined by a cut on a continuous observable, assumed without loss of generality to be x, the independence assumption can be tested by introducing another cut on the same observable through the region D. The ABCD method is then used with the newly introduced cut parallel to x and the other variable y to predict the number of events in the region adjacent to the SR, also called the validation region VR. This prediction is compared with the number of events observed in (VR). If the prediction matches the observation, the independence assumption can be

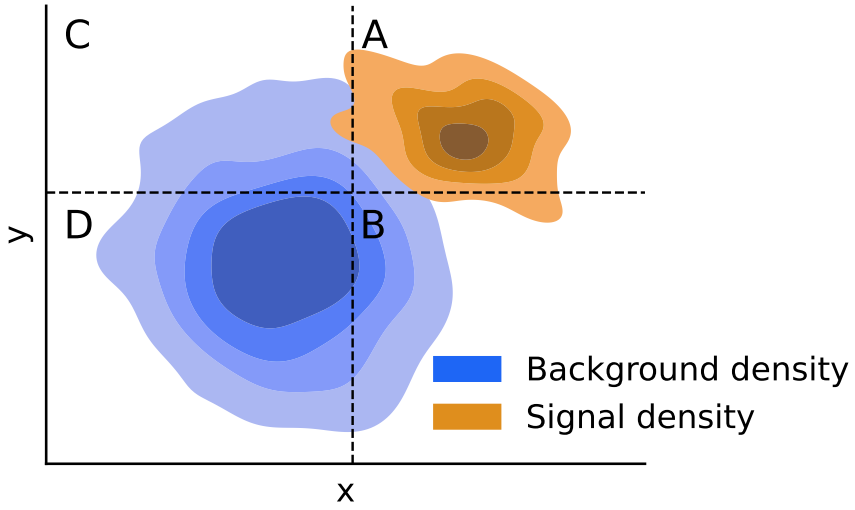

Figure 3.1: A simple example dataset with two continuous variables and possible region definitions for the ABCD method. The CRs are dominated by background events, while the signal events are concentrated in the SR. Furthermore, the background events are uncorrelated in x and y.

considered valid. An ideal example for the ABCD method is shown in Fig. 3.1.

3.2 Background Shape Estimation

Background shape estimation techniques extrapolate the distribution of the background process in the CRs into the SR. Extrapolation of the distribution may be advantageous if the simulation of the background process has large systematic uncertainties or is biased in some way. This task is orthogonal to yield estimation as the distributions are normalised to unity and the yield prediction can be applied by scaling the distribution by the predicted distribution. Since the goal is to predict a distribution, a generative model is a natural choice for this task.

The *ABCDnn* method uses autoregressive normalising flows conditioned on each region to predict the shape of the background distribution in the SR [63]. First, similar to the ABCD method, the phase space is divided into rows and columns along two observables, where each bin is assigned to be a CR, VR or SR. Ideally, the background is well modelled in the CRs. If there is systematic mismodelling in the SRs, this should be reflected in the VRs. This method can also be used if the genertion of samples in the CRs is strongly computationally favoured to the generation of samples in the SRs. Then the VRs should be chosen such that their distributions closely match those of the SRs, but it is still

possible to generate enough samples in them for an accurate evaluation of the trained model.

An autoregressive normalisation flow is fitted to the observables whose shape is to be predicted. The region information is given as a conditional input to the flow. The row and column coordinate of the bin are encoded as a one-hot vector and concatenated, so that the flow can extrapolate to regions that were not part of the fit, but share either the same row or column with at least one of the CRs. The resulting model may be validated by checking whether the agreement between the data and the simulation in the VRs improves when the background samples are replaced by the predicted distributions in the VRs.

A version of the ABCDnn method can be found in Ref. [64], where it was used to extrapolate the shapes of $t\bar{t}$ + QCD events in the high jet multiplicity regime for the measurement of the $t\bar{t}t\bar{t}$ process at CMS. This was done by constructing a flow that matches the simulation samples to the data, changing only the background samples. The data and samples are quantised so that the data-driven background distribution can be constructed by subtracting the signal samples from the data according to the simulation ratio of signal to background events:

$$p_{\text{data-driven}}((b_i, b_{i+1}]) = \frac{N_{\text{samples}}}{N_{\text{background}}} \cdot p_{\text{data}}((b_i, b_{i+1}]) - p_{\text{signal}}((b_i, b_{i+1}]) \,, \tag{3.2}$$

where $(b_i, b_{i+1}]$ denotes bin i with b_i being the lower and b_{i+1} the upper bin edge. The quantisation is necessary to ensure that $p_{\text{data-driven}}$ is non-negative. Note that this method assumes that both the signal-to-background ratio and the signal shape are modelled correctly. The quantisation into bins also puts a strong limit on the number of dimensions, since the number of bins scales exponentially in the number of dimensions and if there are too many bins, the number of events decreases and therefore the likelihood of statistical fluctuations creating bins, that would contain negative probability mass increases.

In Ref. [65] GANs are used to simulate background events in a SR by replacing an object in an event from a CR with a generated object that would have been misidentified and therefore placed in the SR. This shows that, in principle, any generative modelling technique can be used for background shape estimation. However, normalising flows are expected to be advantageous as they are comparatively easy to train [60].

4 Shape Estimation for $t\bar{t}H$ Background

This work applies data-driven background shape estimation to the measurement of the $t\bar{t}H$ production process, i.e. distributions of observables for top-antitop processes with associated jets are modelled. Two observables with systematic differences between simulation and data were chosen: First, the sum of hadronic jet p_{T}, called H_{T}, where p_{T} describes the transverse momentum of a jet measured in the plane orthogonal to the beam axis. This variable is known to be systematically mismodelled [66], because it involves approximations in the production of the top quark pair production [61]. The jet radius setting of the anti-k_t jet reconstruction algorithm is set to $r = 0.4$ [35]. Only jets with $p_{\mathsf{T}} \geq 25$ GeV and absolute pseudorapidity $|\eta| = |-\ln(\tan(\theta/2))| \leq 2.5$, where θ is the angle between the 3D momentum vector of the particle and the beam axis, were retained. Second, the score classifier trained to discriminate events of the $t\bar{t}H$ process from all other events, using a transformer network trained for the on-going Run 2 legacy analysis also providing the input samples [1]. Similar systematic simulation data discrepancies were found in this variable. Ratio plots of the data versus simulation are included in the figures 4.12 and 4.8.

To simplify the analysis and allow for later comparison with other background estimation techniques, the semileptonic decay channel was chosen. This means only events with exactly one reconstructed lepton are considered. Furthermore, events with jet multiplicity less than 4 are omitted, since they are dominated by $t\bar{t}c$ and $t\bar{t}$ + light events, which may not be representative of the most relevant background processes, where a top-antitop quark pair are accompanied by one or more bottom quarks, i.e. $t\bar{t}b$, $t\bar{t}B$ and $t\bar{t}b\bar{b}$, producing bottom flavour jets.

[1] The development of this classifier and production of the simulated samples were not the focus of the author

4.1 Region Definitions and Preprocessing

Regions are defined by the jet multiplicity and the number of b-tagged jets using the DL1r algorithm at a working 70% efficient [67], similar to Ref. [64]. The maximum number of b-tags is chose to be 4, the same as the minimum number of jets, such that the defined form a rectangle, see Fig. 4.1 for region definitions among other information. The number of jet multiplicity bins is limited to 4 because regions with 8 or more jets would contain too few events. To still be able to use events with more than 7 jets or more than 4 b-tags, the region definitions at the edge are made inclusive, i.e. events with 7 or more jets, or in the case of b-tags 4 or more, are included. The four regions with the highest jet and b-jet multiplicity are chosen as SRs, since they are most sensitive to the $t\bar{t}H$ process. The two regions with 2 b-tags, one fewer, than the two of the SRs, and 6 or greater than 7 jets are chose as VRs, such that at least two CRs belong to each row and each column. The event yields and $t\bar{t}H$ significance ratios of all regions are shown in Fig. 4.1.

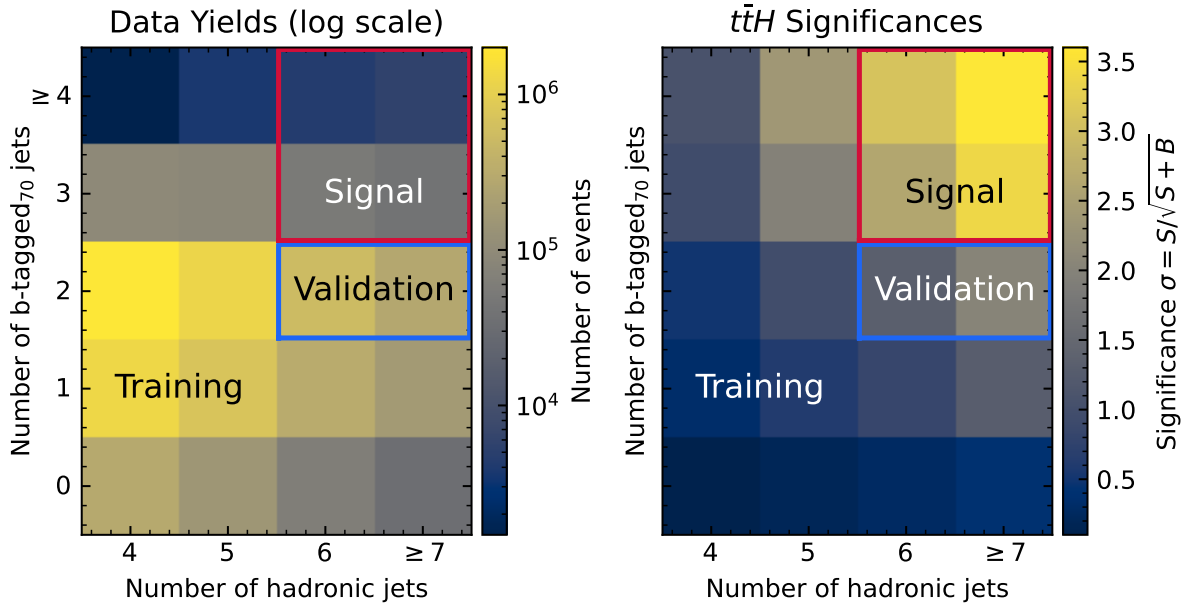

Figure 4.1: The number of data events in each of the regions is shown on the left, note the log-scale. The significance ratio of the $t\bar{t}H$ signal process computed on simulation data for each region is shown on the right. Note, that the signal regions have the highest significance ratio. Statistical uncertainties are indicated by the hatched band around the histogram.

Since both the H_{T} and the DNN score distributions have long tails and are non-negative, instead of modelling them directly the natural logarithm is applied to them. The raw distributions are shown in Fig. 4.2. As is common with machine learning algorithms,

both features are separately normalised to zero mean and unit variance. The means and variances are calculated from the simulated background. Normalisation can help to make better use of finite numerical precision and make hyperparameter choices more transferable between features.

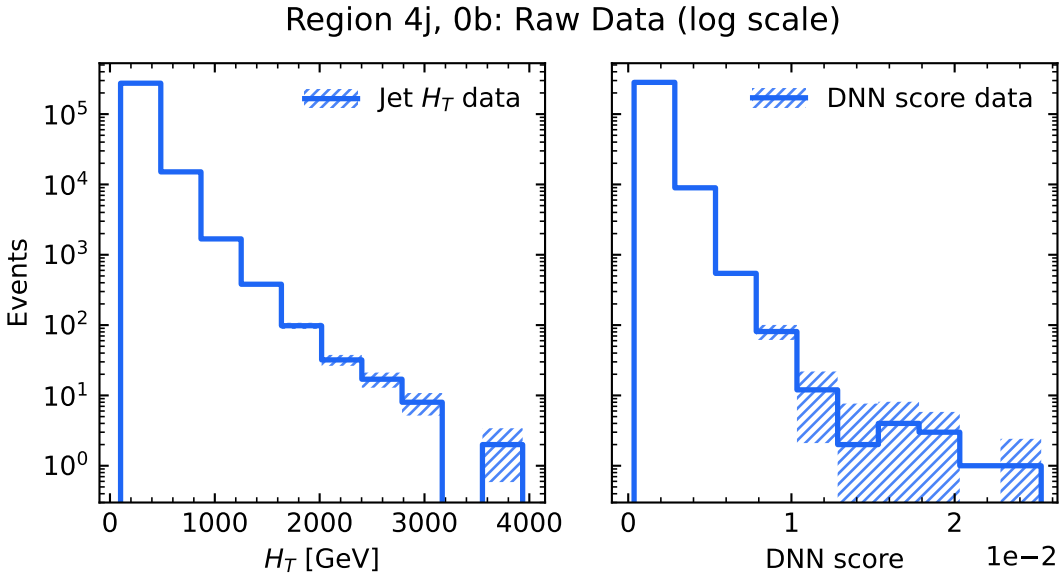

Figure 4.2: Histograms of the H_{T} (left) and the DNN $t\bar{t}H$ classifier score (right) are shown with a logarithmic y-scale. Most of the events are concentrated to the far left of the value range, while few events are observed with significantly larger H_{T} values or DNN scores.

4.2 ABCDnn Shape Estimation

The standard ABCDnn method was applied to a similar setup as described above[2] [63]. Since the goal is to predict the background in the VRs and SRs means and variances are not available in these regions. Therefore the mean and standard deviations were calculated for the background samples in all CRs together, but independently for each feature variable and then the same normalisation was applied to each region. For the normalising flow a standard normal prior was used, because the normalised features seemed already close enough to a standard normally distributed. The density of the standard normal can be efficiently calculated and differentiated in closed form, therefore maximum likelihood training is the best available option. The maximum likelihood objective is directly weighted by the simulation event weights.

[2] A dataset containing events with jet multiplicities ≥ 5

The data points in the CRs were randomly assigned to either the training set or the validation set, with a split of 80% and 20%, to allow generalisation to be tested on an independently sampled set with exactly the same underlying distribution as the model was trained on. The loss curves in Fig. 4.3 show that without regularisation, the model generalises to the distribution of CRs but not to the VRs, as the overall loss on the validation set decreases but the loss on the VRs increases over the course of training. By adding a l_2 penalty to the training process, the model generalises even better to the training distributions, as the training set and validation set loss almost perfectly match. Furthermore, the model generalises to the VRs, but still does not perform as well on them, since the VRs loss is significantly higher than the other two losses. Replacing the background samples with the predicted background distributions in the VRs does not improve the simulation-data agreement, which is not very surprising since systematic mismodelling is already present in the CRs to a similar extent as in the other regions: Even with the perfect extrapolation algorithm, biased extrapolation is expected when extrapolating from biased data points.

Therefore, it makes more sense to directly train a model to remove the mismodelling in the CRs by fitting the simulated background distribution to the data-driven background (i.e. data minus simulated signal). The following sections investigate three different approaches for this. Each of these methods must be able to solve the problem of translating between two distributions that are only indirectly accessible via i.i.d. random samples with weights. It is worth noting that a significant proportion of the weights in one of the sample sets are negative.

4.3 Maximum Likelihood Training with Kernel Density Estimation

The probability density function of neither the simulated background nor the data-driven background is known, but for maximum likelihood training at least one of them must be known. This can be mitigated by approximating the true density of the data minus simulated signal using kernel density estimation. The requirement to handle (negatively) weighted samples can be solved by weighting the kernel around each data point with the weight associated with that data point. The bandwidth should be chosen high enough so that the resulting density estimate is non-negative. This allows calculating region wise means and variances from the background samples.

This approach was not able to closely match the ground truth correction in the CRs for any of the hyperparameter choices tested. This is likely due to the fact that the true

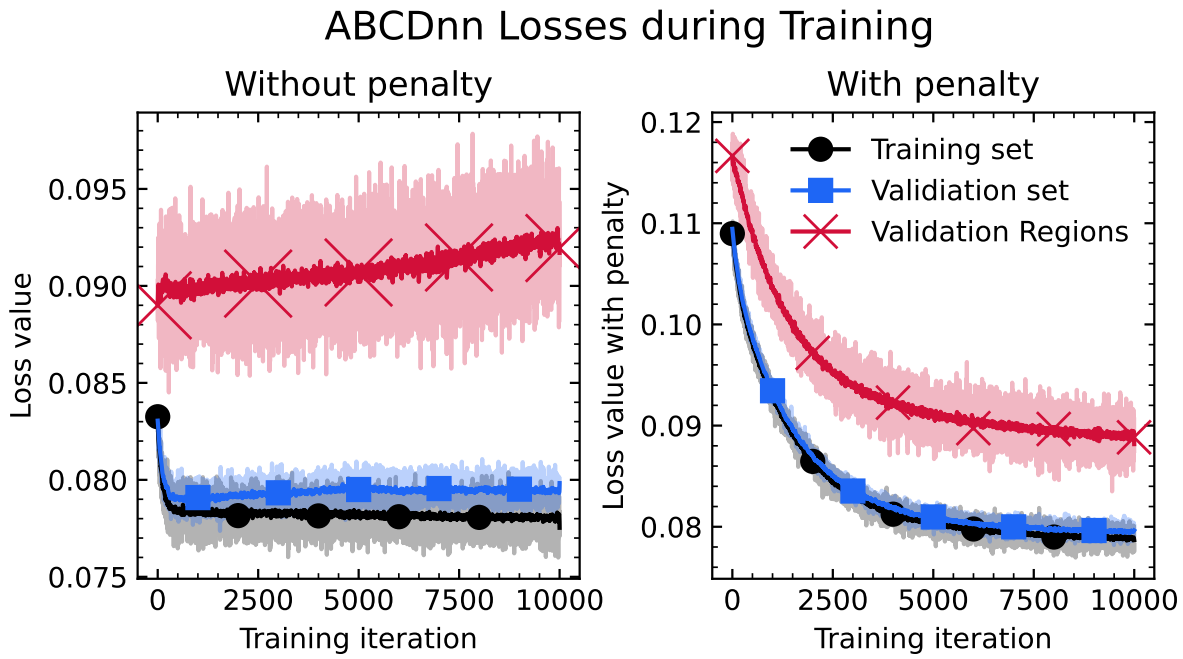

Figure 4.3: Losses on the training set, validation set and VRs over the course of training are shown. For the setup on the right, no weight penalty was used. The model on the left was trained with a l_2 penalty of strength 10^{-5}. For ease of comparison, both training runs were run for 10000 iterations, although the loss of the CRs in the validation set for the unpenalised model on the left starts to increase slightly after about 1000 iterations.

correction is very close to the identity function, as testing this approach on toy examples, where a simple shift is added to one of the distributions, resulting in a ground truth correction further away from the identity function, yielded positive results. Using the maximum mean discrepancy training approach from Ref. [64] gives similar results for any choice of hyperparameters, including kernel for the MMD, binning and batch size up to 1024 instances. It is likely that both of these approaches fail in the case where only small corrections are needed, because even using relatively large batches the distribution(s) are not described well enough for the loss to be informative. This hypothesis is supported by the fact of the large fluctuations in the loss curves during training, which are shown in Fig. 2.1).

4.4 Cumulative Distribution Function Matching

This section proposes a way to improve the informativeness of the loss by making more effective use of the datasets for the background samples and the data minus the simu-

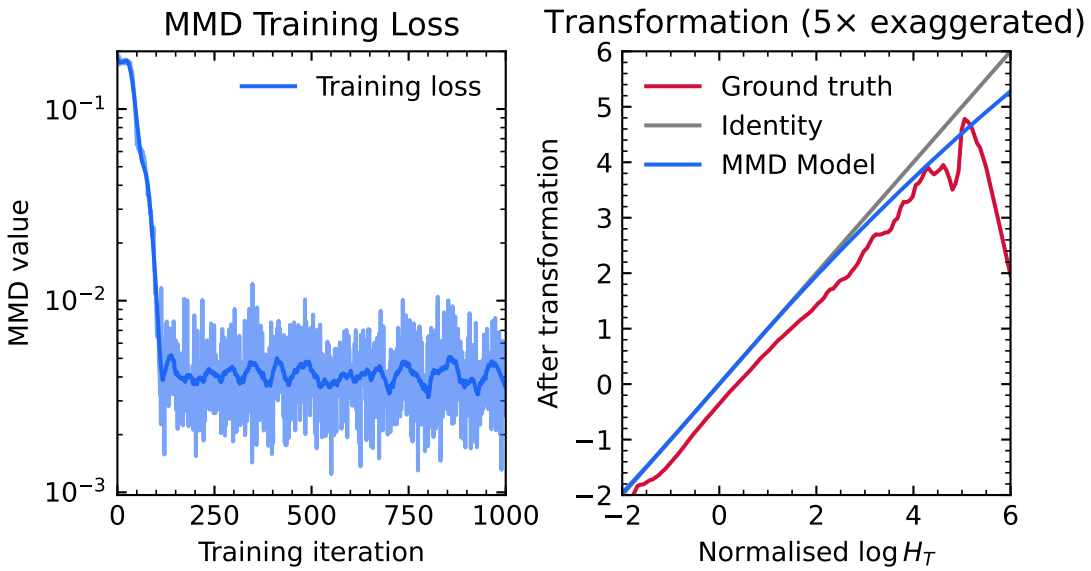

Figure 4.4: The training loss during the binned MMD training using KDE for the data minus simulated signal distribution is shown on the right. The decrease in loss is initially large, but then small compared to the random fluctuations. The difference between the ground truth transformation and the model to the identity function is exaggerated by a factor of 5 (left). The ground truth transformation is a monotone function.

lated signal distribution, through a pre-processing step that approximates the underlying *cumulative distribution function* (CDF). This approximation can then be used directly in the loss function

$$\mathcal{L}_{\text{CDF}}(x, \theta) := \left(\hat{F}_{\text{data-signal}}(f_\theta(x)) - \hat{F}_{\text{background}}(x)\right)^2 , \tag{4.1}$$

where $\hat{F}_{\text{name}}$ denotes the approximations to the CDFs and f_θ denotes the model. If this loss is zero and $\hat{F} = F$ is continuous, then by the probability integral transform $\hat{F}_{\text{background}}(x)$ with $x \sim p_{\text{background}}$ is distributed according to a standard uniform, and since $\mathcal{L}_{\text{CDF}}(x, \theta) = 0 \; \forall x \in \mathbb{R} \Rightarrow \hat{F}_{\text{data-signal}}(f_\theta(x)) = \hat{F}_{\text{background}}(x)$ so is $\hat{F}_{\text{data-signal}}(f_\theta(x))$. If $\hat{F} = F$ is continuous, its proper inverse F^{-1} exists and applying it to $\hat{F}_{\text{data-signal}}(f_\theta(x))$, a standard uniform distributed variable, yields $f_\theta(x)$, which is then distributed according to $\hat{F}_{\text{data-signal}}$ by the inverse sampling transform. This is discussed more explicitly in section 4.5.

Before discussing the construction of the CDF approximation, some definitions are needed: Consider the case of a real-valued observable x modelled as a random variable. Simulated data points and their associated weights $(x, w) \in \mathbb{R}^2$ come from a joint prob-

ability measure space $(\mathbb{R}^2, \mu)$ which satisfies the conditions of non-negativity and finite yield:

$$\int_{A\times\mathbb{R}} w \, \mathrm{d}\mu(x,w) \geq 0 \quad \forall A \in \mathfrak{B}(\mathbb{R}) \tag{4.2}$$

$$\wedge \int_{\mathbb{R}^2} w \, \mathrm{d}\mu(x,w) =: \mathrm{yield}(\mu) < \infty \,. \tag{4.3}$$

In the case of measured data points, μ can be decomposed into $p \otimes \delta_{\mathrm{yield}}$. The background sample distribution should satisfy these conditions, since predicting a negative yield in any region of phase space would indicate an error in the simulation, and predicting an infinite yield would be unphysical. The data minus the simulated signal distribution satisfies these conditions if the predicted number of signal events does not exceed the measured data at any point. Formally, with μ_{signal} and μ_{signal} normalised according to Eq. 3.2, this gives

$$\int_{A\times\mathbb{R}} w \, \mathrm{d}\mu_{\mathrm{signal}}(x,w) \leq \int_{A\times\mathbb{R}} w \, \mathrm{d}\mu_{\mathrm{data}}(x,w) \quad \forall A \in \mathfrak{B}(\mathbb{R}) \,. \tag{4.4}$$

The normalised weighted distribution μ' is defined as the push-forward measure of mu by dividing the weights by the yield

$$\mu'(A) := \mu(\{(x, w \cdot \mathrm{yield}(\mu) : (x,w) \in A\}) \tag{4.5}$$

$$\Rightarrow \int_A w \, \mathrm{d}\mu'(x,w) = \frac{1}{\mathrm{yield}(\mu)} \cdot \int_A w \, \mathrm{d}\mu(x,w) \,, \tag{4.6}$$

for any $A \in \mathfrak{B}(\mathbb{R}^2)$. Then $\int w \, \mathrm{d}\mu' = 1$. The CDF $F : \mathbb{R} \to [0,1]$ of the intended distribution of the observable is then given by

$$F(t) := \int_{\mathbb{R}^2} w \cdot 1_{x\leq t} \, \mathrm{d}\mu'(x,w) \,, \tag{4.7}$$

which is monotone because of the condition 4.2 and satisfies $\lim_{t\to-\infty} F(t) = 0 \wedge \lim_{t\to\infty} F(t) = 1$ because of the Eq. 4.3. Although the empirical estimate $F_n : \mathbb{R} \to [0,1]$

$$F_n(t) := \frac{1}{\sum_{i=1}^n w_i} \sum_{i=1}^n w_i \cdot 1_{x\leq t} \,, \tag{4.8}$$

from i.i.d. samples $(x_1, w_1), (x_2, w_2), \ldots (x_n, w_n) \sim \mu$ with $\sum_{i=1}^n w_i > 0$ for some $n \in \mathbb{N}$ is not guaranteed to be monotone because of possible negative weights, but has the same bounds $\lim_{t\to-\infty} F_n(t) = 0 \wedge \lim_{t\to\infty} F_n(t) = 1$. Using F_n as an approximation for the CDF in the loss does not work because its derivative is either 0 or undefined. This

can be mitigated by using linear interpolation and using the one-sided derivative at the interpolation points, which were previously discontinuity points.

However, when optimising this loss with a CDF approximation that is not monotone, a gradient based algorithm may get stuck at a local extreme value, because the value of $F_n(f_\theta(x))$ should be higher, but $f_\theta(x)$ is a local maximum of F_n, and vice versa for a lower desired value and a minimum of F_n. So an approximation that is guaranteed to be monotone is needed. This is done by constructing the highest monotone function that is lower at all points than the pointwise constant empirical approximation F_n, called F_n^+, and the lowest monotone function that is greater than F_n, called F_n^-:

$$F_n^+(t) = \min\{f(t) : f \text{ monotone} \wedge f(x) \geq F_n(x)\ \forall x \in \mathbb{R}\} \tag{4.9}$$

$$= \max_{x \leq t} \min_{x' \leq x} F_n(x') \tag{4.10}$$

$$F_n^-(t) = \max\{f(t) : f \text{ monotone} \wedge f(x) \leq F_n(x)\ \forall x \in \mathbb{R}\} \tag{4.11}$$

$$= \min_{x \geq t} \max_{x' \geq x} F_n(x')\,. \tag{4.12}$$

The minima and maxima exist because the image of F_n is finite. The boundary conditions of the CDFs are applied to the envelope functions F_n^+ and F_n^- by shifting and scaling them, and The final approximation of the CDF is then simply defined as the midpoint between these two approximations $\hat{F}(t) := (F_n^{+\prime} + F_n^{-\prime})/2$ and is therefore also guaranteed to be monotonic. The prime indicates the applied boundary conditions. The construction of $\hat{F}$ is illustrated in Fig. 4.5. For training in more than one dimension, the conditional CDF can be estimated by integrating the KDE, although the use of the KDE imposes a tight upper bound on the number of dimensions.

4.4.1 Results

A conditional sigmoidial flow is trained on the CRs [55], using Eq. 4.1 as the loss function with an additional l_2 weight penalty term. The hyperparameters are slightly tuned using the loss in the VRs. The final model with the optimised hyperparameters is trained on the CRs and VRs together. All regions are given the same weight during training, regardless of their number of samples or their yield. This procedure was done independently for the H_T and the DNN score, as training a joint correction did not improve the results in the VRs and therefore no improvement was expected in the SRs. The loss curves for the three different region types during training on the CRs only of the model with optimised hyperparameters are shown in Fig. 4.6. The model seems to generalise well to both the VRs and the CRs.

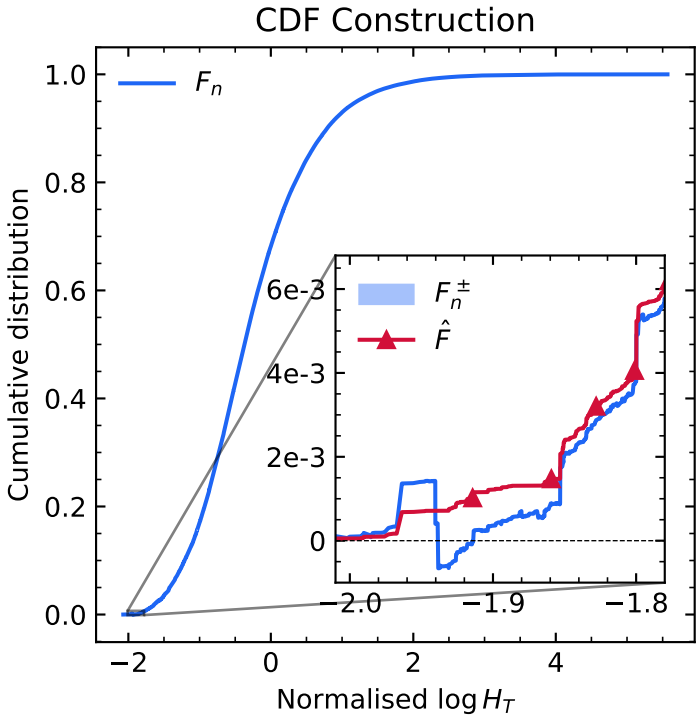

Figure 4.5: Construction of the monotone CDF approximation from data points with weights, some of which may be negative. The shaded band is given by the two maximally close monotone functions F_n^+ from above and F_n^- from below. The influence of negative weights is shown in the inset plot. As the blue line, i.e. the empirical CDF, decreases, the red line, i.e. the monotone CDF approximation, simply stops increasing around this point.

Extrapolation results for the final model in the SRs are shown in Fig. 4.12 for the H_T and Fig. 4.8 for the DNN score. The quality of the extrapolation can be evaluated by comparing the discrepancy between the data minus the simulated signal distribution and the background samples before and after applying the model to the background samples. The Hellinger distance of the histograms was chosen as a measure of this discrepancy, as opposed to the χ^2 statistic, because it does not depend on the number of samples in the bins. The distance was also chosen to be binned, rather than using the underlying structure of the metrically scaled variables, as would be the case with the earthmover distance or the Kolmogorov-Smirnov statistic, because this structure is unlikely to be used for further analysis of the signal process that this background estimation allows. For example, a binned likelihood fit could be performed to measure the cross section of the signal process [68]. The formula for the Hellinger distance between two histograms P and

Q with the same bin definitions and bin probabilities p_i and q_i for $i = 1, \ldots, n$ is given by

$$H(P,Q) = \frac{1}{\sqrt{2}} \sqrt{\sum_{i=1}^{n} (\sqrt{p_i} - \sqrt{q_i})^2}. \tag{4.13}$$

The ratios of the model improvements of the H_T shapes are ≥ 2 for all SRs except the 6 jets 4 b-tags region, where the model background distribution fits the data minus signal distribution worse than the simulated distribution with a ratio of 0.75. It is noteworthy that this region has the lowest yield of all SRs and also the worst extrapolation performance for the DNN score with a ratio of 1.13. A possible explanation for this could be that there are relatively few events in the CRs with 4 b-tags, which is the row corresponding to this region. Furthermore, the range of the fraction of jets with b-tags in this row is relatively large, as it reaches one in the 4 jets 4 b-tags region, which is also the region with the fewest events. However, this hypothesis does not explain why the H_T model performs so well in the 7 jets 4 b-tags region. The remaining DNN score extrapolations do not improve as much as the H_T, but are still significant with values of $\{1.18, 1.29, 1.4\}$.

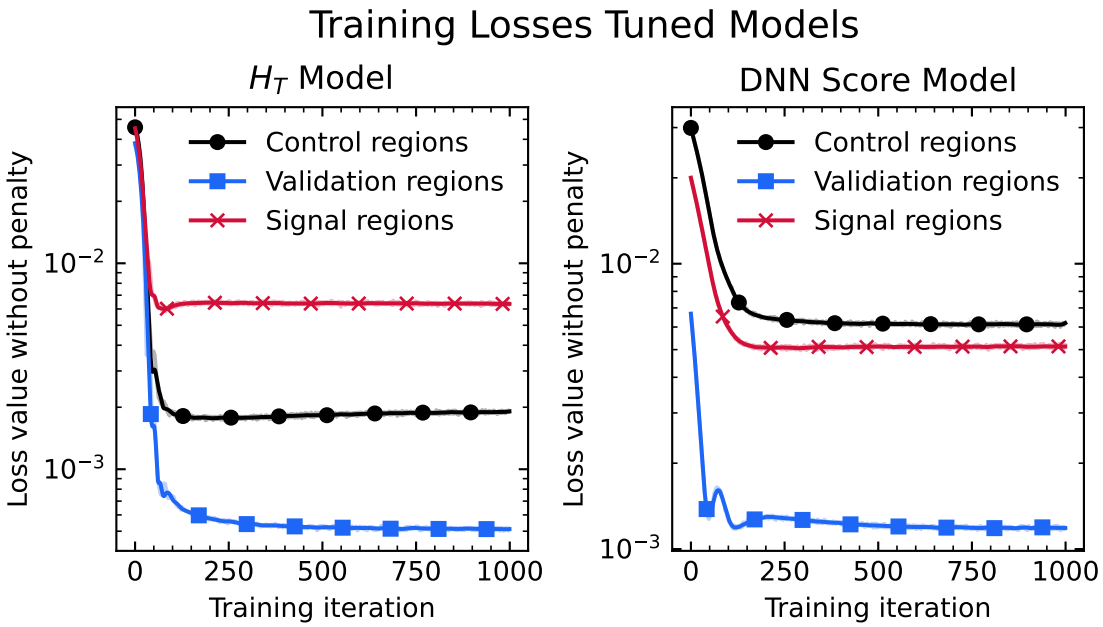

Figure 4.6: Training the tuned models only on the CRs using the CDF loss results in the loss curves above. The losses of each region type are normalised by the number of regions, so that the loss values can be compared between regions. The penalty is not included in the loss values shown. The loss decreases rapidly and then converges for all region types, although the VRs take longer to reach diminishing returns.

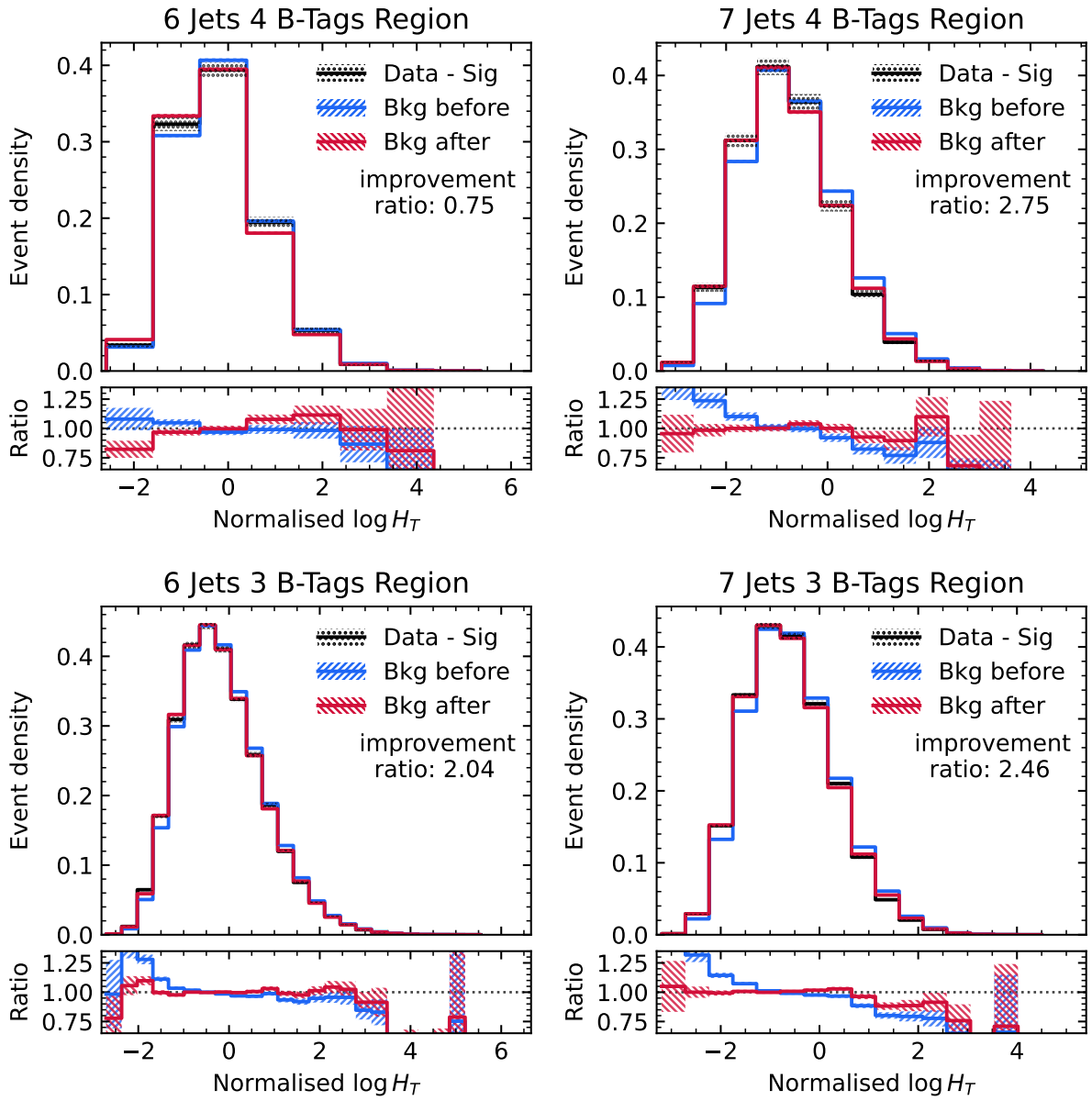

Figure 4.7: Extrapolation results for the H_T in the SRs. The black histogram is the data minus the simulated signal distribution, the blue histogram is the background samples and the red histogram is the extrapolated background samples generated by applying the learned model to the simulated background samples in the same region. All histograms are normalised to area one and the number of bins depends on the number of data events. The hatched band around the extrapolated histogram includes all the systematic uncertainties of the method, the MC statistical uncertainties, but not the systematic uncertainties of the simulated samples themselves. The improvement ratio shown in the legend is the ratio of how much the Hellinger distance between the data Mius signal histogram and the background histogram is reduced by extrapolating the background.

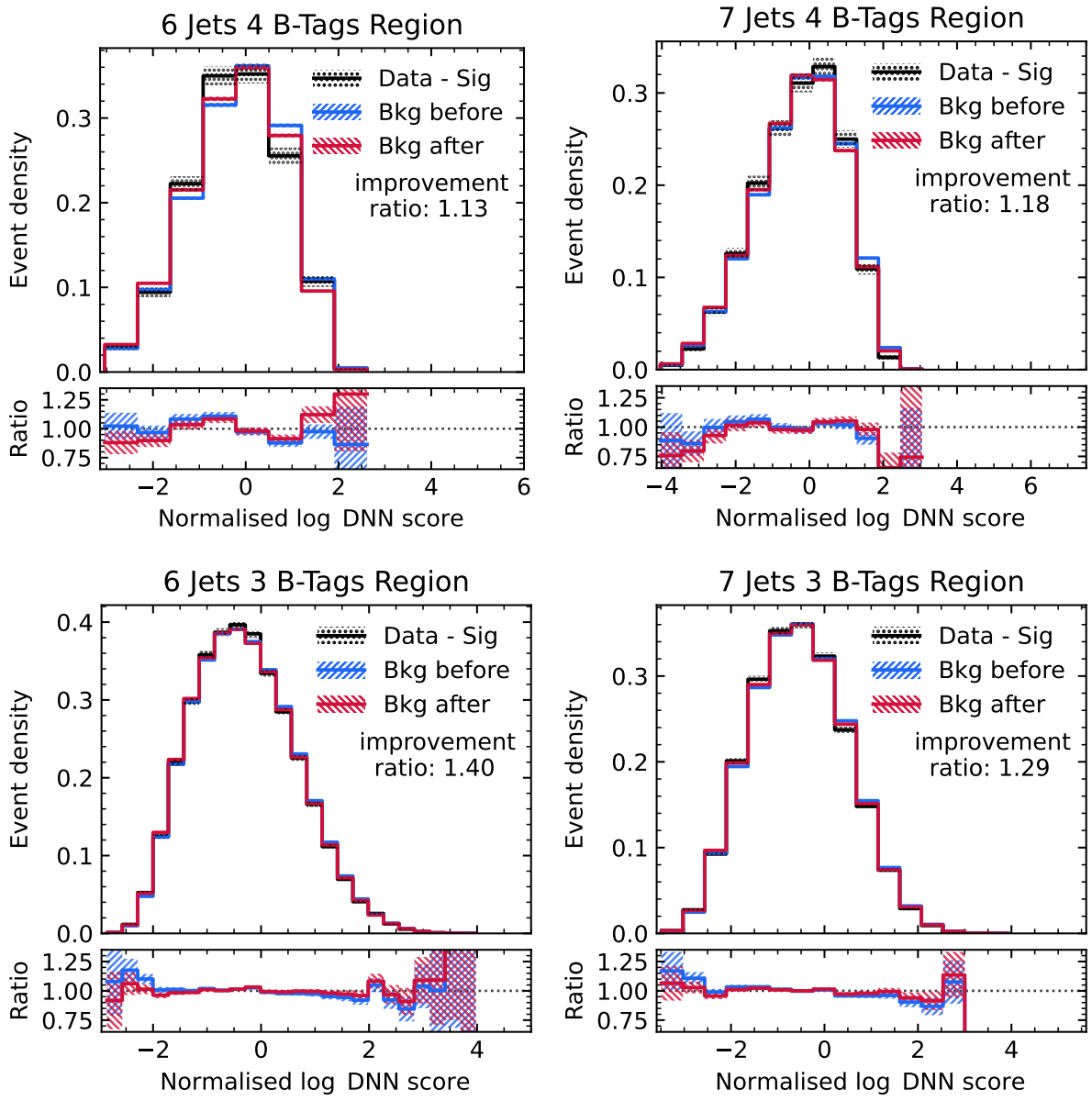

Figure 4.8: Extrapolation results for the DNN score in the SRs. Plot details are the same as in Fig. 4.12. All extrapolations improve the agreement between data and simulation, as all improvement ratios are greater than 1. It is noteworthy that the overall discrepancies are smaller than for H_{T}, so there may be less room for improvement.

4.4.2 Systematic Uncertainties

Well-calibrated predictions can only be made if good estimates of uncertainties can be calculated. This includes the uncertainties introduced by the methods themselves. To estimate the systematic uncertainties of the results presented, different choices of hyperparameters are considered. For each hyperparameter, two models are trained, one with the hyperparameter at a reasonable upper bound and one at a lower bound, with all other hyperparameters remaining the same, i.e. at their nominal values. The systematic uncertainty for each hyperparameter is then given by the range of the prediction. The uncertainties are assumed to be independent, so that a quadratic addition can be used to obtain the total uncertainty. What constitutes reasonable bounds for a hyperparameter is a difficult and, in the general case, unresolved question, as it may strongly depend on the given problem. In this case, it is worth noting that many hyperparameter configurations can already be disqualified because it is clear that they do not allow the model to generalise to the VRs. For all variations of the hyperparameters, the loss in the VRs was checked not to diverge. All approximately continuous hyperparameters such as network width, l_2 penalty strength, batch size, learning rate and number of optimisation steps were halved for the lower bound and doubled for the upper bound. For the activation function, sigmoid, relu, elu and leaky relu were tried, but the model using the sigmoid activation function was largely ineffective on the VRs and was therefore not considered in the uncertainty calculation. The number of layers in the encoder and decoder was varied from 1 in the encoder and decoder each to 3 in the encoder and 2 in the decoder. The random initialisation of the neural network and the random shuffling of the dataset are also considered as a source of uncertainty, since the training process is otherwise completely deterministic. The full results of the systematic uncertainty calculation are shown in Fig. 4.9 for H_T and Fig. 4.10 for the DNN score. It can be seen that the statistical uncertainties of the number of simulated samples largely dominate all other sources of systematic uncertainties. The most sensitive hyperparameters are the penalty strength, width and depth, which are all closely related to the model capacity. The choice of activation function, number of training iterations, network initialisation and data shuffling are almost negligible sources of uncertainty. The addition of an encoding for the fraction of b-tagged jets was also included in the systematic uncertainties, but turned out to be negligible.

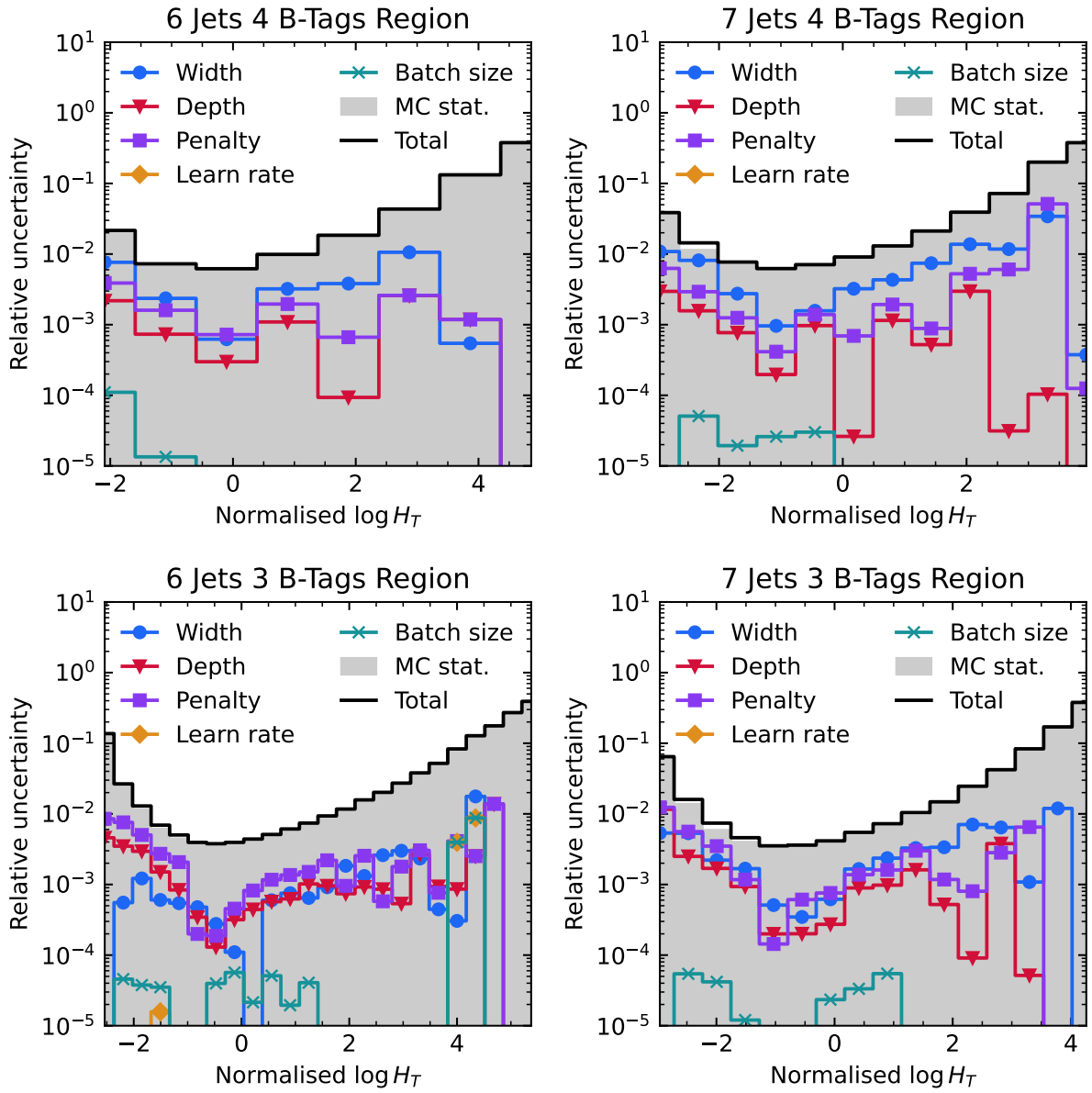

Figure 4.9: The individual relative systematic uncertainties for the H_T extrapolation in the SRs. Note the log scale. The uncertainties due to initialisation, activation function and number of optimisation steps are not shown because they do not exceed 10^{-5} in any bin, but are included in the calculation of the total systematic uncertainty. The total uncertainty is everywhere dominated by the systematic uncertainty from the number of simulated samples.

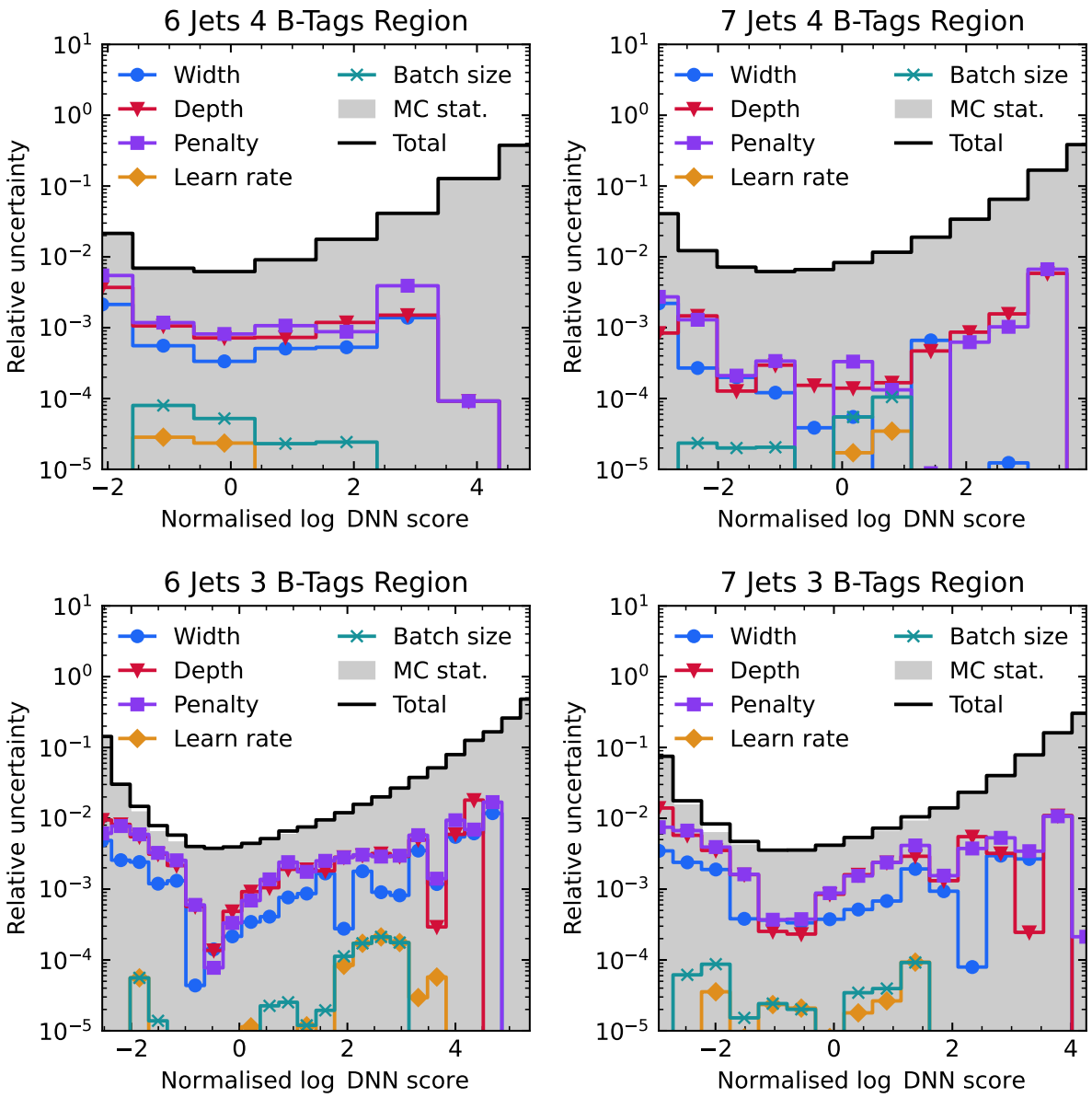

Figure 4.10: The individual relative systematic uncertainties for the DNN score extrapolation in the SRs. Plot details are the same as in Fig. 4.9. The results are also very similar, although the neural network width seems to be a bit less important.

4.5 Machine Learning Free Shape Correction

A ML-free alternative to the above approach for one-dimensional data is discussed in this section. This method will be used as a baseline to test whether ML is needed for the task of extrapolating the H_T and DNN score. To do this, the monotone CDF approximation for the background sample distribution and the data minus signal distribution from above is used to construct functions that directly transform one into the other in each region:

First, the $\hat{F}_\text{background}$ functions are applied to the background samples in each of the CRs to transform them to be approximately uniformly distributed. Doing this with a known CDF is called a probability integral transform. Second, the right inverse $\hat{F}^{-1}_\text{data-signal}$ of the CDF approximations for the data minus the simulated signal distributions are constructed in each CR. This is possible because $\hat{F}$ is guaranteed to be monotone. Applying $\hat{F}^{-1}_\text{data-signal}$ to the approximately standard uniformly distributed $\hat{F}_\text{background}(x)$, where x is a background sample, yields samples that are close in distribution to the data minus signal distribution. For known quantile functions, this part is also known as inverse transform sampling. By combining these functions, which transform the background samples in the CRs to have a shape similar to the data minus simulated signal distribution in the same CR, the shapes of the background samples in the VRs and SRs can be corrected. A combination similar to the ABCD method is most natural here: All transformations of CRs with either the same row or column number are averaged. Applying the transformations from the CRs and VRs to the SRs for the H_T results in worse improvement ratios on average, although the ML-based approach only performs better in one region. A similar effect can be observed for the DNN score, where the ML-free approach significantly decreases the data versus simulation agreement in the SRs with 7 jets, but performs slightly better than the normalising flow in the SR with 6 jets and 4 b-tags.

The overall tendency for the ML-based approach to extrapolate better is likely due to the fact that the ML-based approach can use data points from CRs that are not in the same row or column. In addition, the normalisation flow with, trained with a weight penalty, introduces a bias towards smoother transformations, which could be helpful for generalisation. As the method using normalised flows does not appear to increase uncertainties significantly, the only reason for using the ML-free approach is that it may be simpler and therefore quicker and easier to implement.

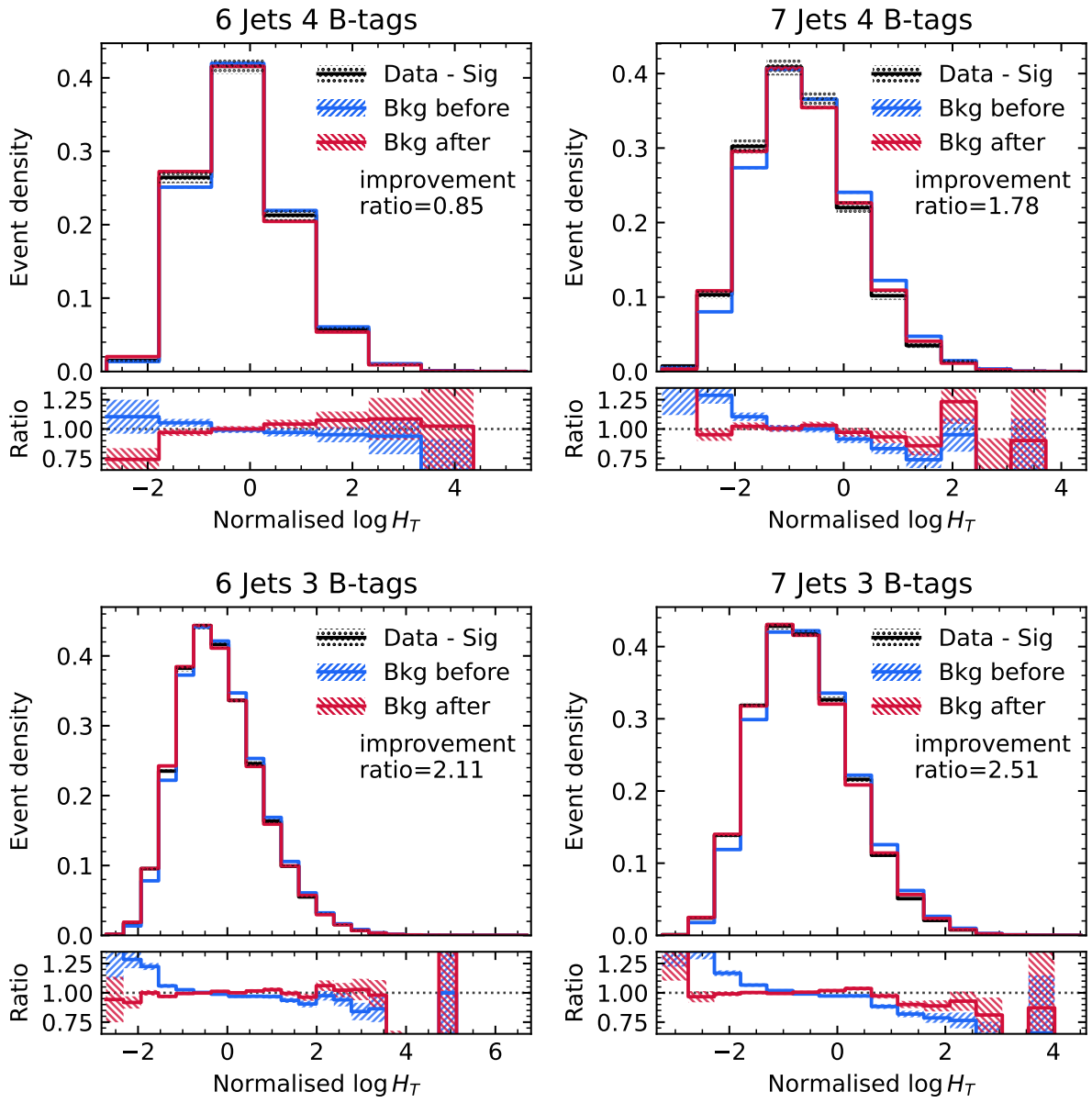

Figure 4.11: ML free extrapolation results for the H_{T} in the SRs. Plot details are the same as in Fig. 4.12. The data versus simulation agreement is improved by the applied transformation in all SRs except the 6 jets 4 b-tags region. This is similar to the ML-based results.

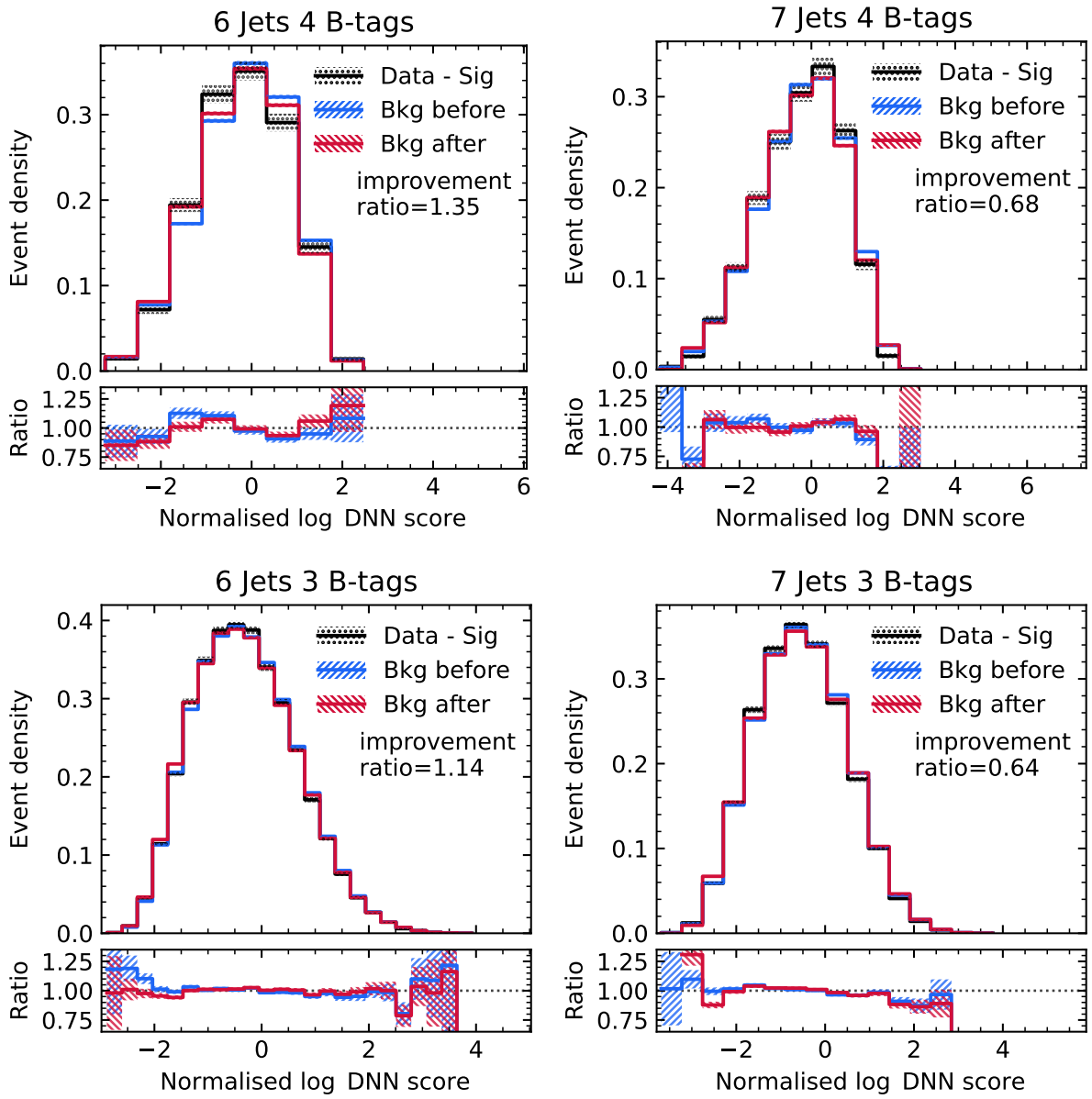

Figure 4.12: ML free extrapolation results for the DNN score in the SRs. Plot details are the same as in Fig. 4.12. The data versus simulation agreement only increases in the regions with 6 jets. This is not the case not eh ML-based approach, where the agreement of the DNN score increases in all regions.

5 Conclusion

Data-driven background shape estimation has been successfully applied to reduce the systematic data versus simulation discrepancy of the scalar sum of the hadronic jet transverse momentum H_T and the score of a deep neural network $t\bar{t}H$ classifier distribution for events with more than 6 jets and more than 3 b-tags. For this, only the background samples were transformed with a normalising flow. Training a normalisation flow to correct for small discrepancies between datasets with possible negatively weighted samples was the main difficulty. Several approaches were investigated. Constructing a monotone approximation to the cumulative distribution function of the data minus the simulated signal and the simulated background distributions for the loss function proved successful. The discrepancy between data and simulation was reduced by $\{0.75, 2.75, 2.04, 2.46\}$ times in different regions for the H_T and $\{1.13, 1.18, 1.40, 1.29\}$ for the DNN score, which outperforms the machine learning free baseline. Systematic uncertainties were quantified by variations of the hyperparameters and found to be dominated by the statistical uncertainties of the simulation samples.

5.1 Limitations

The main limitation of the method is that it does not appear to scale to correct more than one variable at a time. Although in Ref. [64] MMD-based training was successfully applied to two variables. Furthermore, it is unclear whether or in what situations a reduction in loss on the VRs is a robust predictor of generalisation to the SRs. This is likely to be highly dependent on the definition of the regions, so guidelines on how to select regions could be of great benefit. In addition, it is not clear to what extent the method is able to improve our ability to distinguish theories with good predictions from those with poor predictions. Assuming that the model is able to perfectly extrapolate the systematic differences for samples from very different simulations, this would make it possible to make accurate predictions, but not necessarily to learn humanly understandable laws of particle physics, since the predictions of neural networks are usually difficult to interpret.

5.2 Outlook

The requirement to translate only part of one distribution, i.e. the background sample, to match another distribution, where both distributions are specified only by data points, possibly with negative weights, precludes most standard generative modelling techniques. It may be possible to adapt methods from the unpaired domain translation literature to work with negative weights, which could address the poor performance on higher dimensional data. A concrete possibility would be to combine the simple method of Ref. [69] for domain translation with a diffusion based decoder [70]. The inherent composability of diffusion models can then be used to model the data minus the simulated signal distribution [71]. A more direct approach to the task of correcting only the background samples would be to condition the generator of a CycleGAN on whether a given sample is from a signal or background process and add a term that minimises corrections to the signal events [72].

To find recommendations for region definitions, systematic tests on a wider range of datasets are needed. The full hadronic decay of $t\bar{t}H(bb)$ would be one good candidate for this, as it provides additional jets and therefore more possible regions, similar to Ref. [64]. Finally, the sensitivity of the neural network training to the number of data points could be tested experimentally and added to the systematic uncertainties.

Bibliography

[1] J. Dalton, *Foundations of the Atomic Theory*, William F. Clay, Edinburgh (1893)

[2] J. Dalton, *Foundations of the Molecular Theory*, William F. Clay, Edinburgh (1893)

[3] E. Rutherford, *The Scattering of α and β Particles by Matter and the Structure of the Atom*, Philosophical Magazine **21** (1911)

[4] J. Chadwick, *Existence of a Neutron*, Proceedings of the Royal Society A **136(830)**, 692 (1932)

[5] H. Yukawa, *On the Interaction of Elementary Particles*, Proc. Phys.-Math. Soc. Jpn. **17(48)** (1935), URL `URL_OF_PDF`

[6] C. Lattes, G. Occhialini, H. Muirhead, C. Powell, *Processes involving charged mesons*, Nature **159**, 694 (1947)

[7] M. Gell-Mann, *A Schematic of Baryons and Mesons*, Physics Letters **8(3)**, 214 (1964)

[8] H. L. Anderson, E. Fermi, E. A. Long, D. E. Nagle, *Total cross-sections of positive pions in hydrogen*, Physical Review **85(5)**, 936 (1952)

[9] F. Englert, R. Brout, *Broken Symmetry and the Mass of Gauge Vector Mesons*, Phys. Rev. Lett. **13**, 321 (1964)

[10] P. W. Higgs, *Broken Symmetries and the Masses of Gauge Bosons*, Phys. Rev. Lett. **13**, 508 (1964)

[11] G. S. Guralnik, et al., *Global Conservation Laws and Massless Particles*, Phys. Rev. Lett. **13**, 585 (1964)

[12] P. W. Higgs, *Spontaneous Symmetry Breakdown without Massless Bosons*, Phys. Rev. **145**, 1156 (1966)

[13] T. W. B. Kibble, *Symmetry Breaking in Non-Abelian Gauge Theories*, Phys. Rev. **155**, 1554 (1967)

[14] P. W. Higgs, *Broken Symmetries, Massless Particles and Gauge Fields*, Phys. Lett. **12**, 132 (1964)

[15] S. L. Glashow, *Partial Symmetries of Weak Interactions*, Nucl. Phys. **22**, 579 (1961)

[16] S. Weinberg, *A Model of Leptons*, Phys. Rev. Lett. **19**, 1264 (1967)

[17] A. Salam, *Weak and Electromagnetic Interactions*, Almqvist & Wiksell, Stockholm (1968)

[18] S. L. Glashow, et al., *Weak Interactions with Lepton-Hadron Symmetry*, Phys. Rev. D **2**, 1285 (1970)

[19] H. Georgi, et al., *Unified Weak and Electromagnetic Interactions without Neutral Currents*, Phys. Rev. Lett. **28**, 1494 (1972)

[20] H. D. Politzer, *Reliable Perturbative Results for Strong Interactions*, Phys. Rev. Lett. **30**, 1346 (1973)

[21] H. D. Politzer, *Asymptotic Freedom: An Approach to Strong Interactions*, Phys. Rept. **14**, 129 (1974)

[22] D. J. Gross, et al., *Asymptotically Free Gauge Theories*, Phys. Rev. D **8**, 3633 (1973)

[23] S. Weinberg, *The Making of the Standard Model*, Eur. Phys. J. C **34**, 5 (2004)

[24] G. 'tHooft, *Renormalization of Massless Yang-Mills Fields*, Nucl. Phys. B **33(1)**, 173 (1971)

[25] G. 't Hooft, et al., *Regularization And Renormalization Of Gauge Fields*, Nucl. Phys. B **44**, 189 (1972)

[26] G. 't Hooft, et al., *Combinatorics of Gauge Fields*, Nucl. Phys. B **50**, 318 (1972)

[27] T. A. 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(1)**, 1 (2012), URL `https://doi.org/10.1016%2Fj.physletb.2012.08.020`

[28] T. C. Collaboration, *Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC*, Physics Letters B **716(1)**, 30 (2012), URL `https://doi.org/10.1016%2Fj.physletb.2012.08.021`

[29] T. Albahri, A. Anastasi, K. Badgley, et al., *Magnetic-field measurement and analysis for the Muon g- 2 Experiment at Fermilab*, Physical Review A **103(4)**, 042208 (2021)

[30] L. Evans, *The large hadron collider*, New Journal of Physics **9(9)**, 335 (2007)

[31] T. A. Collaboration, *The ATLAS Experiment at the CERN Large Hadron Collider*, Journal of Instrumentation **3(08)**, S08003 (2008), URL https://dx.doi.org/10.1088/1748-0221/3/08/S08003

[32] *ATLAS inner detector: Technical Design Report, 1*, Technical design report. ATLAS, CERN, Geneva (1997), URL https://cds.cern.ch/record/331063

[33] *ATLAS liquid argon calorimeter: Technical design report* (1996)

[34] *ATLAS tile calorimeter: Technical Design Report*, Technical design report. ATLAS, CERN, Geneva (1996), URL https://cds.cern.ch/record/331062

[35] M. Cacciari, G. P. Salam, G. Soyez, *The anti-kt jet clustering algorithm*, Journal of High Energy Physics **2008(04)**, 063 (2008), URL https://dx.doi.org/10.1088/1126-6708/2008/04/063

[36] T. A. Collaboration, *Jet reconstruction and performance using particle flow with the ATLAS Detector*, The European Physical Journal C **77(7)** (2017), URL https://doi.org/10.1140/epjc/s10052-017-5031-2

[37] *ATLAS muon spectrometer: Technical Design Report*, Technical design report. ATLAS, CERN, Geneva (1997), URL https://cds.cern.ch/record/331068

[38] *Technical Design Report for the Phase-II Upgrade of the ATLAS TDAQ System*, Technical report, CERN, Geneva (2017), URL https://cds.cern.ch/record/2285584

[39] L. Adamczyk, E. BanaÅ, A. Brandt, et al., *Technical Design Report for the ATLAS Forward Proton Detector*, Technical report (2015), URL https://cds.cern.ch/record/2017378

[40] A. Collaboration, et al., *Observation of Higgs boson production in association with a top quark pair at the LHC with the ATLAS detector*, arXiv preprint arXiv:1806.00425 (2018)

[41] T. C. collaboration, *Observation of $t\bar{t}H$ Production*, Physical Review Letters **120(23)** (2018), URL https://doi.org/10.1103/physrevlett.120.231801

[42] T. Plehn, G. P. Salam, M. Spannowsky, *Fat Jets for a Light Higgs Boson*, Physical Review Letters **104(11)** (2010), URL https://doi.org/10.1103%2Fphysrevlett.104.111801

[43] P. A. Zyla, P. D. Group, *QUARKS*, Progress of Theoretical and Experimental Physics **2020**, 083C01 (2020), URL https://pdg.lbl.gov/2020/reviews/rpp2020-rev-quarks.pdf

[44] B. Mellado, A. M. Cooper-Sarkar, M. o. Kramer, *Handbook of LHC Higgs Cross Sections: 2. Differential Distributions* (2012), URL http://cds.cern.ch/record/1416519

[45] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer, 1 edition (2007), URL https://www.microsoft.com/en-us/research/uploads/prod/2006/01/Bishop-Pattern-Recognition-and-Machine-Learning-2006.pdf

[46] F. Rosenblatt, *The perceptron: A probabilistic model for information storage and organization in the brain.*, Psychological Review **65(6)**, 386 (1958), URL https://doi.org/10.1037/h0042519

[47] A. G. Ivakhnenko, V. G. Lapa, *Cybernetic Predicting Devices*, CCM Information Corporation (1965), URL https://gwern.net/doc/ai/1966-ivakhnenko.pdf

[48] K. Hornik, M. Stinchcombe, H. White, *Multilayer feedforward networks are universal approximators*, Neural networks **2(5)**, 359 (1989)

[49] D. P. Kingma, J. Ba, *Adam: A Method for Stochastic Optimization* (2017), 1412.6980

[50] T. Hastie, R. Tibshirani, J. Friedman, *The Elements of Statistical Learning*, Springer New York (2009), URL https://doi.org/10.1007/978-0-387-84858-7

[51] P. Nakkiran, G. Kaplun, Y. Bansal, T. Yang, B. Barak, I. Sutskever, *Deep Double Descent: Where Bigger Models and More Data Hurt* (2019), 1912.02292

[52] G. K. Dziugaite, D. M. Roy, Z. Ghahramani, *Training generative neural networks via Maximum Mean Discrepancy optimization* (2015), 1505.03906

[53] A. Gretton, K. M. Borgwardt, M. J. Rasch, B. Schölkopf, A. Smola, *A Kernel Two-Sample Test*, Journal of Machine Learning Research **13(25)**, 723 (2012), URL http://jmlr.org/papers/v13/gretton12a.html

[54] D. J. Rezende, S. Mohamed, *Variational Inference with Normalizing Flows* (2016), 1505.05770

[55] C.-W. Huang, D. Krueger, A. Lacoste, A. Courville, *Neural Autoregressive Flows* (2018), 1804.00779

[56] M. Germain, K. Gregor, I. Murray, H. Larochelle, *MADE: Masked Autoencoder for Distribution Estimation* (2015), 1502.03509

[57] G. Papamakarios, T. Pavlakou, I. Murray, *Masked Autoregressive Flow for Density Estimation* (2018), 1705.07057

[58] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *Improving language understanding by generative pre-training* (2018)

[59] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, Y. Bengio, *Generative Adversarial Networks* (2014), 1406.2661

[60] N. Kodali, J. Abernethy, J. Hays, Z. Kira, *On Convergence and Stability of GANs* (2017), 1705.07215

[61] M. Czakon, D. Heymes, A. Mitov, *High-Precision Differential Predictions for Top-Quark Pairs at the LHC*, Physical Review Letters **116(8)** (2016), URL https://doi.org/10.1103%2Fphysrevlett.116.082003

[62] T. C. Collaboration (CDF Collaboration), *Observation of Top Quark Production in $\overline{p}p$ Collisions with the Collider Detector at Fermilab*, Phys. Rev. Lett. **74**, 2626 (1995), URL https://link.aps.org/doi/10.1103/PhysRevLett.74.2626

[63] S. Choi, J. Lim, H. Oh, *Data-driven Estimation of Background Distribution through Neural Autoregressive Flows* (2020), 2008.03636

[64] T. C. Collaboration, *Evidence for four-top quark production in proton-proton collisions at $\sqrt{s} = 13$ TeV*, arXiv preprint arXiv:2303.03864 (2023)

[65] V. Lohezic, M. O. Sahin, F. Couderc, J. Malcles, *Data driven background estimation in HEP using generative adversarial networks*, The European Physical Journal C **83(3)** (2023), URL https://doi.org/10.1140%2Fepjc%2Fs10052-023-11347-8

[66] T. A. Collaboration, *Measurements of top-quark pair differential and double-differential cross-sections in the ℓ+jets channel with pp collisions at $\sqrt{s}$=13 TeV using the ATLAS detector*, The European Physical Journal C **79(12)** (2019), URL https://doi.org/10.1140%2Fepjc%2Fs10052-019-7525-6

[67] T. A. Collaboration, *ATLAS flavour-tagging algorithms for the LHC Run 2 pp collision dataset*, Eur. Phys. J. C **83**, 681 (2023)

[68] R. Barlow, C. Beeston, *Fitting using finite Monte Carlo samples*, Computer Physics Communications **77(2)**, 219 (1993)

[69] L. Hui, X. Li, J. Chen, H. He, C. gong, J. Yang, *Unsupervised Multi-Domain Image Translation with Domain-Specific Encoders/Decoders* (2017), 1712.02050

[70] K. Preechakul, N. Chatthee, S. Wizadwongsa, S. Suwajanakorn, *Diffusion Autoencoders: Toward a Meaningful and Decodable Representation* (2022), 2111.15640

[71] J. Ho, T. Salimans, *Classifier-Free Diffusion Guidance* (2022), 2207.12598

[72] J.-Y. Zhu, T. Park, P. Isola, A. A. Efros, *Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks* (2020), 1703.10593

Acknowledgement

I would like to thank Prof. Dr. Arnulf Quadt for the opportunity to work on this interesting topic, as well as for the opportunity to learn more about deep learning in physics in Dortmund and to visit and give a talk at ATLAS-D in Bonn, both of which were very exciting and rewarding experiences. Next, I would like to thank Steffen Korn for all the fruitful discussions and his exceptional willingness to help. Furthermore my thanks goes out to Chris Scheulen for the production of the simulated samples and his patience with all my questions, and Elizaveta Shabaline for her recommendations on the literature about the mismodelling of top quarks. I would also like to acknowledge the IT support of Andreas Kirchhoff and Theresa Reisch when I had problems accessing the samples on the server and the latex template for this bachelor thesis. Moreover, I grateful for the time Melissa Quinnan took to answer my questions about her analysis. Additionally, I would like to thank all the people at the top meetings for making me feel welcome and the people at ATLAS-D and the workshop in Bonn for making my time there really enjoyable. Finally, I would like to express my appreciation for Prof. Dr. Max Wardetzky making himself available as my second reference, which is not a matter of course.

P.S. Gratitude to the unseen silicon assistant(s) that made the mundane more manageable. Modern times bring modern allies.

Erklärung nach §13(9) der Prüfungsordnung für den Bachelor-Studiengang Physik und den Master-Studiengang Physik an der Universität Göttingen:
Hiermit erkläre ich, dass ich diese Abschlussarbeit selbständig verfasst habe, keine anderen als die angegebenen Quellen und Hilfsmittel benutzt habe und alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten Schriften entnommen wurden, als solche kenntlich gemacht habe.
Darüberhinaus erkläre ich, dass diese Abschlussarbeit nicht, auch nicht auszugsweise, im Rahmen einer nichtbestandenen Prüfung an dieser oder einer anderen Hochschule eingereicht wurde.

Göttingen, den 1. November 2024

(Paul Wollenhaupt)