Inferring Signal Strength Parameters using Conditional Invertible Neural Networks at the CMS Experiment

Von der Fakultät für Mathematik, Informatik und Naturwissenschaften der RWTH Aachen University zur Erlangung des akademischen Grades eines Doktors der Naturwissenschaften genehmigte Dissertation

vorgelegt von

Mate Zoltan Farkas

PREVIOUS QUALIFICATION

aus HOMETOWN

Berichter: Universitätsprofessor Dr. Hans Martin Erdmann
Universitätsprofessor Dr. BAZ TEST

Datum der mündlichen Prüfung: XX. Month 2018

Diese Dissertation ist auf den Internetseiten
der Hochschulbibliothek online verfügbar.

Mate Zoltan Farkas
INFERRING SIGNAL STRENGTH PARAMETERS USING CONDITIONAL INVERTIBLE NEURAL NETWORKS AT THE CMS EXPERIMENT

Dissertation in Physik
Rheinisch-Westfälische Technische Hochschule Aachen
III. Physikalisches Institut A

Berichter: Universitätsprofessor Dr. Hans Martin Erdmann
Universitätsprofessor Dr. BAZ TEST

Contents

Page

1 Theoretical Foundations **3**
1.1 The Standard Model of Particle Physics . . . 3
1.2 The Higgs Mechanism and the Higgs Boson . . . 4
1.3 Gluon-Gluon Induced VH-Higgs Production . . . 5

2 Experimental Setup **9**
2.1 The Large Hadron Collider . . . 9
2.2 The Compact Muon Solenoid . . . 11
2.2.1 Silicon Tracking . . . 12
2.2.2 Electromagnetic Calorimeter . . . 12
2.2.3 Hadronic Calorimeter . . . 13
2.2.4 Superconducting Solenoid Magnet and the Return Yokes . . . 14
2.2.5 Muon Chambers . . . 14
2.2.6 Trigger System . . . 15
2.3 Physics Object Reconstruction . . . 16
2.3.1 Particle Flow . . . 16
2.3.2 Jet Clustering . . . 17
2.3.3 Flavour Tagging . . . 17
2.3.4 Muons . . . 18

3 Deep Learning and Conditional Invertible Neural Networks **21**
3.1 Foundations of Deep Learning and Neural Networks . . . 21
3.1.1 The Universal Approximation Theorem . . . 21
3.1.2 Fully-Connected Neural Networks . . . 21
3.1.3 Activation Functions . . . 22
3.1.4 Objective Functions . . . 23
3.1.5 Training and Loss Optimization . . . 24
3.2 Conditional Invertible Neural Networks . . . 25
3.2.1 General INN Architecture . . . 26
3.2.2 Normalizing Flows . . . 27
3.2.3 Affine Coupling Blocks . . . 27
3.2.4 The Kullback-Leibler Divergence as a Loss Function . . . 29
3.2.5 Posterior Inference . . . 30

4 Analysis- Strategy and Setup **31**
4.1 Analysis Strategy . . . 31
4.2 Analysis Setup . . . 33
4.3 Event Categorisation . . . 34
4.4 Multi-Process Classification and Weight Application . . . 34
4.5 Likelihood Fits . . . 36

5 Network Setup **39**
5.1 Training Dataset Generation . . . 39
5.1.1 Prior Selection . . . 39
5.1.2 Systematic Uncertainties . . . 40
5.1.3 Normalization and Shape-Changing Uncertainties . . . 41

5.1.4 Non-Linear Histogram Morphing 42
5.1.5 Systematic Uncertainty Representation in Training Data 43
5.2 Statistical Uncertainties 44
5.3 Network Architecture 45

6 Signal Modifier Parameter Inference with cINNs 47
6.1 Training Performance 47
6.2 Latent Space Distribution 47
6.3 Calibration Curves 48
6.4 Network Predictions 49
6.5 Posterior Inference 50

Bibliography 55

1 Theoretical Foundations

1.1 The Standard Model of Particle Physics

The Standard Model of Particle Physics (Standard Model or SM) is our best understanding of the fundamental interactions (strong, weak and the electromagnetic interaction) occuring in Nature (excluding gravity due to the inaccessibility of the scales it acts under laboratory conditions). It describes the dynamics of elementary particles and the processes they undergo upon production or decay.

Depending on the interactions they participate in, these particles can be sorted into hadrons and leptons; hadrons take part in the strong interaction and couple directly to its mediator particle, the gluon. Leptons on the other hand do not participate in it. Particles can be electrically charged or neutral. In the former case, they couple to photons and hence participate in the electromagnetic interaction. The Z and W bosons are the mediators of the weak interaction.

Regarding their spin, one distinguishes between two main categories: bosons (particles of integer spin) and fermions (particles with half-odd integer spin).

The SM Lagrangian can be written in a compact form as

$$\mathcal{L} = -\frac{1}{4}F_{\mu\nu}F^{\mu\nu} + i\bar{\psi}\not{D}\psi + \psi_i y_{ij}\psi_j\phi + h.c. + |D_\mu\phi|^2 - V(\phi) \tag{1.1}$$

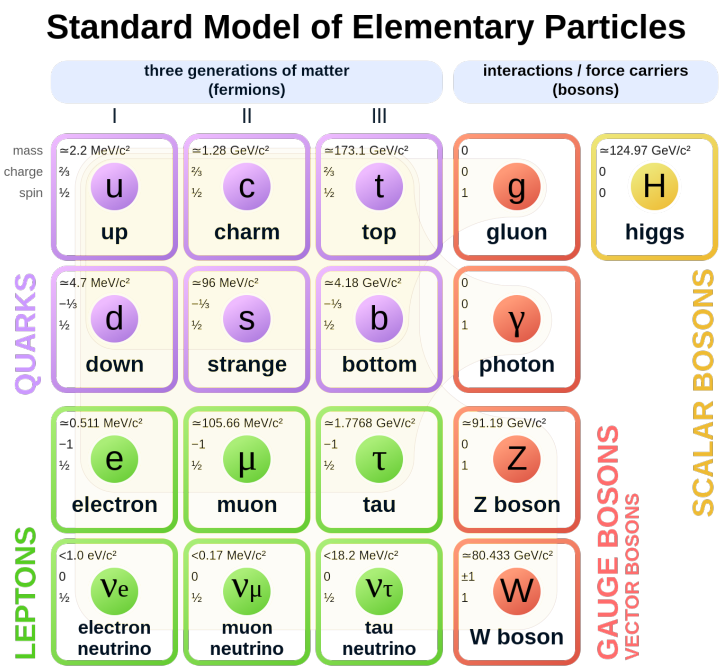

Figure 1.1: Standard model of elementary particles. Note the three generation of quarks and leptons, and their isospin doublet scheme. The four force carriers and the Higgs boson are listed on right. [1]

Despite its incredible predictivity (electron g-2), it is known that the Standard Model is an incomplete description of Nature. For this reason, several SM testing and BSM measurements are performed.

1.2 The Higgs Mechanism and the Higgs Boson

The Higgs term in the Lagragian from eq. 1.1 gives rise to the mass of the gauge bosons. The potential $V(\phi)$ is given by

$$V(\phi) = -\mu^2\phi^\dagger\phi + \frac{\lambda}{4}\left(\phi^\dagger\phi\right)^2$$

and is rotational symmetric and takes a "mexican hat"-like shape. The form of the potential in the complex plane is shown if fig. 1.2.

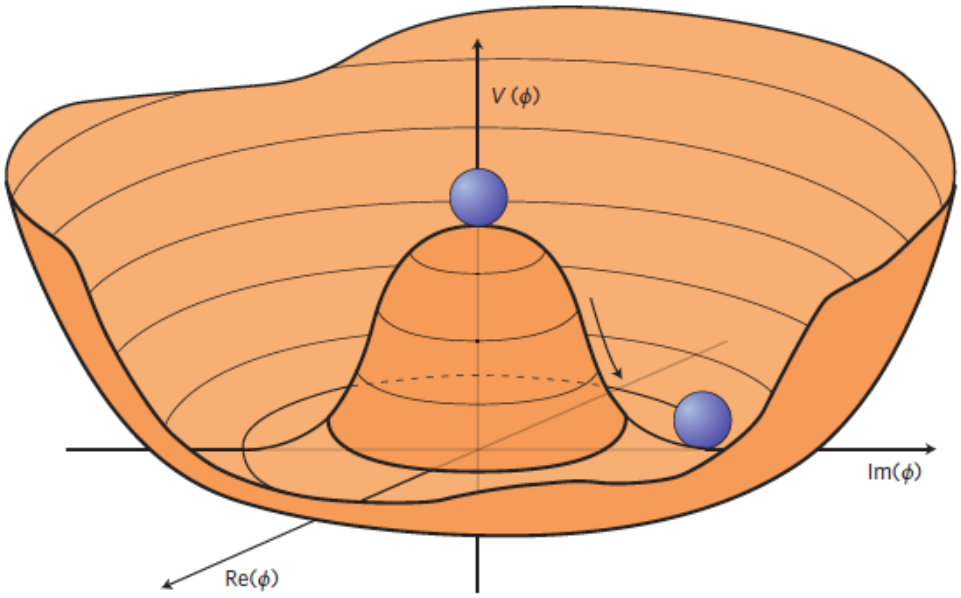

Figure 1.2: Shape of the Higgs potential ("mexican hat"). [2]

Due to the location of the potential minimum lying at $\phi \neq 0$ the field ϕ obtains a vacuum expectation value (spontaneous symmetry breaking). In addition to that, due to the covariant derivative D_μ, additional coupling terms of the gauge bosons to this field ϕ appear. It turns out, that by writing the field ϕ as

$$\phi = \frac{1}{\sqrt{2}}\begin{pmatrix} 0 \\ v + H \end{pmatrix}$$

where v is the location of the potential minimum (a constant) and H is real scalar field, the electroweak gauge bosons obtain a mass through the constant v

$$m_W^2 = \frac{g^2v^2}{4}, \quad m_Z^2 = \frac{(g'^2 + g^2)v^2}{4}$$

with the electroweak coupling constants g, and g'. Not only the gauge bosons obtain a mass thanks to the Higgs field though. Through the Yukawa coupling, fermions (and all massive particles at tree level) couple directly to the Higgs boson with a coupling dependent on their mass and spin. Additionally, the resulting Lagrangian density respects the physical symmetries

$$SU(3)_c \times SU(2)_L \times U(1)_Y$$

which would have broken in the absence of the field ϕ due to the mass terms of the electroweak gauge bosons. As symmetries in a Lagrangian imply the existence of charges, those of the Lagrangian $\mathcal{L}$ induce the colour charges, isospin charges and electric charges; the existence of the latter two is only visible through the electroweak unification.
Since its observation in 2012 by the ATLAS and CMS experiments at CERN [3] the properties of the Higgs boson have been extensively studied in detail. The SM Higgs a CP-even scalar of charge 0 with a measured mass of

$$m_H = 125.25 \pm 0.17\,\text{GeV}$$

The most common Higgs production mechanisms at high energy colliders are shown in fig. 1.3 [4]. These are gluon-gluon fusion (ggF), vector-boson fusion (VBF), Higgs-strahlung (VH), associated gauge boson production at one loop from a gluon-gluon interaction, associated top quark pair production ($t\bar{t}H$) and associated single top quark-quark production (tHq).

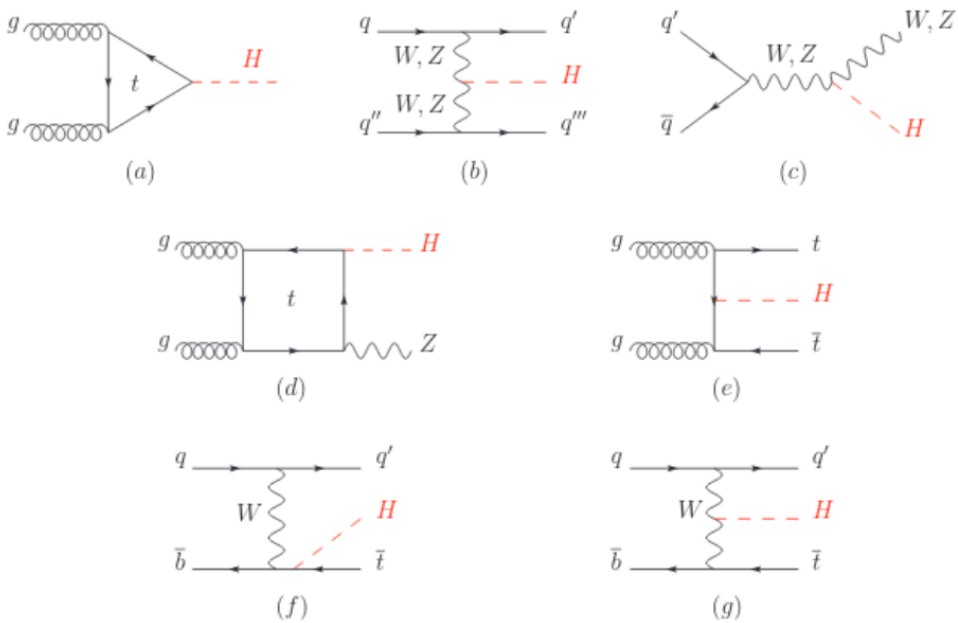

Figure 1.3: Main Higgs boson production processes in high energy hadron colliders [4]. From (a) to (g) these are: ggF (a), VBF (b), Higgs-strahlung (c), associated vector boson production from gg (d), $t\bar{t}H$ (e) and tHq $(f) - (g)$

The most common Higgs decay final state branching ratios for various Higgs masses are shown in fig. 1.4. Note that the main decay channels are swamped with background processes, which made direct detection first available in the $\gamma\gamma$ and $ZZ \to 4l$ final states. Also note that some of the final states are only available through higher-order diagrams as the Higgs couples only to massive particles directly.
In the following the gluon-gluon induced vector boson associated Higgs production ($gg \to VH$) with a focus on a BSM-sensitive physics observable will be discussed in detail.

1.3 Gluon-Gluon Induced VH-Higgs Production

The processes describing $gg \to VH$ are shown in fig. 1.5. As these processes contain loops, they are perfect candidates for SM testing precision measurements. As for the ZH final states the box

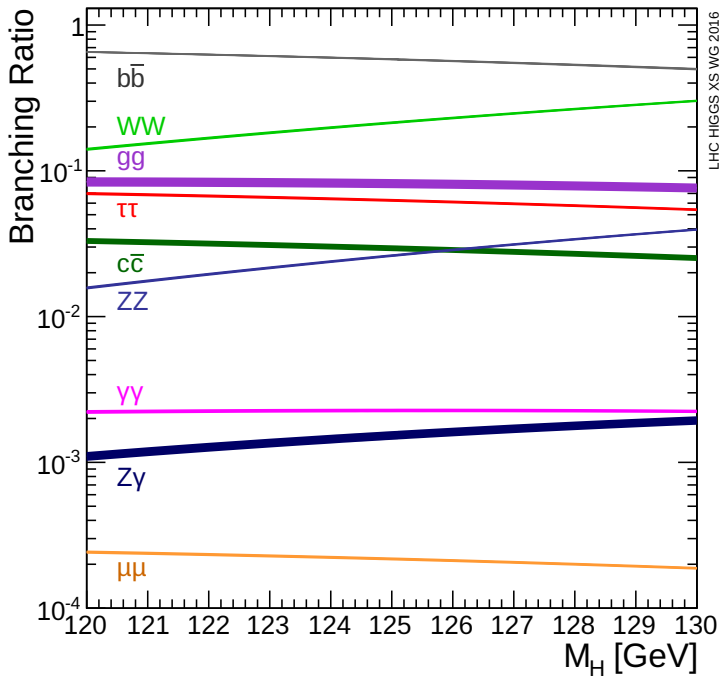

Figure 1.4: Most common Higgs decay channels expressed as final state branching ratios depending on the Higgs mass. The band widths represent the uncertainties of each channel. [5]

and the triangle diagrams interfere destructively, its contribution leads to enhanced sensitivity to BSM physics. This can be seen by the loop itself, to which BSM particles can contribute. Another advantage of the ZH final state is the top-quark exclusibe SM contribution which have characteristic thresholds in several kinematical distributions of the cross-section; well-selected cuts increase the number of gluon-initiated events [6].

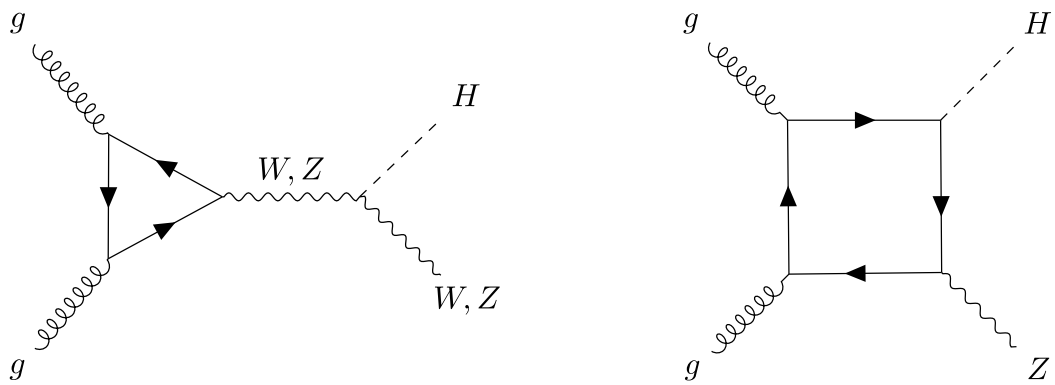

Figure 1.5: Feynman diagrams of the contributing $gg \to VH$ processes at leading order.

The total cross-section for the VH production can be written as

$$\sigma^{VH} = \sigma^{VH}_{\text{DY}} + \sigma^{VH}_{\text{non-DY}}$$

where "DY" stands for the contribution of the Drell-Yan-like production channel (the triangle diagram) and non-DY stands for that of the box diagram. Note that in the SM at leading order we can assume that

$$\sigma^{WH}_{\text{non-DY}} = 0.$$

It can be shown, that the ratio R^{ZH}_{DY} defined as

$$R^{ZH}_{\text{DY}} = \frac{\sigma^{ZH}}{\sigma^{ZH}_{\text{DY}}} = 1 + \frac{\sigma^{ZH}_{\text{non-DY}}}{\sigma^{ZH}_{\text{DY}}} \tag{1.2}$$

is a suitable observable for SM test measurements as it can be seen in fig. 1.6 if one considers the distribution in the invariant M_{ZH} mass.

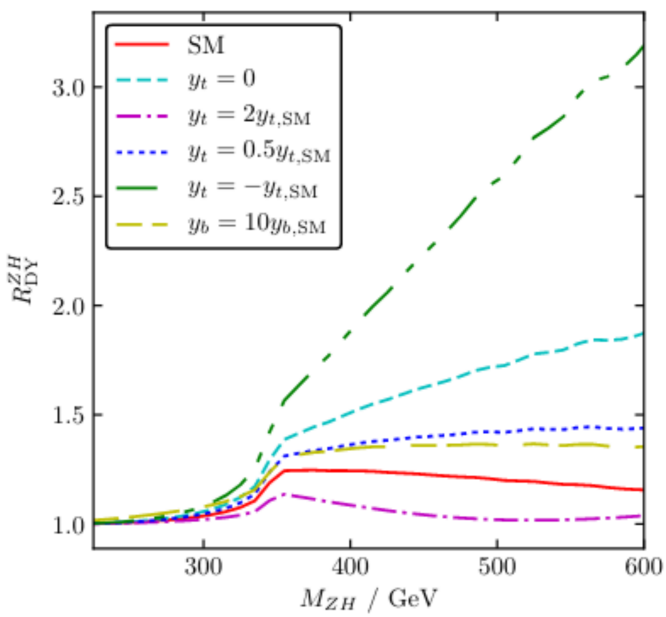

Figure 1.6: The dependence of the ratio R^{ZH}_{DY} on different BSM scenarios in the M_{ZH} distribution. Note that differing top- and bottom quark Yukawa couplings (y_t and y_b, respectively), the resulting distribution differs significantly.

However, an experiment/theory comparison using R^{ZH}_{DY} suffers from arising systematic uncertainties; fortunately it still can be use a basis for other BSM-senstive observables. For this purpose, the similarity between the Drell-Yan-like WH and ZH final states can be exploited. Defining expanded fraction in eq. 1.2 and writing

$$R^{ZW}_{R} = \frac{\sigma^{ZH}/\sigma^{WH}}{\sigma^{ZH}_{\text{DY}}/\sigma^{WH}} \tag{1.3}$$

results in an excellent observable candidate for two reasons. First, it is expected that measuring the nominator σ^{ZH}/σ^{WH} a number of systematic uncertainties cancel especially if the parameters of ZH and WH analyses are aligned as much as possible. Second, the denominator $\sigma^{ZH}_{\text{DY}}/\sigma^{WH}$ can be theoretically evaluated with high precision and is insensitive to BSM effects, as the only couplings (the electroweak couplings) in the related processes are experimentally heavily constrained.

2 Experimental Setup

In this chapter the experimental context of this thesis will be discussed.

2.1 The Large Hadron Collider

The Large Hadron Collider (LHC) is currently the most powerful particle accelerator in the world. Hosted at CERN in Geneva at the Swiss-French border and first put into operation on 10th September 2008, the LHC is designed for proton and heavy lead ion collisions. The machine has gone through several upgrades between the consecutive data-taking phases (Runs) called Long Shutdowns (LS). During these the proton beam energy has been gradually increased from 3.5 TeV to a recently – on the 5th July, 2022 to be precise – achieved energy of 6.8 TeV [7] resulting in a total centre-of-mass (CM) proton-proton collision energy of $\sqrt{s} = 13.6\,\text{TeV}$. Similarly, the beam intensity has seen an increase from 1.1×10^{11} protons per bunch (ppb) and ~200 bunches to a projected ~1.8×10^{11} ppb and ~2500 bunches [8, 9]. With a theoretical maximum CM energy of $\sqrt{s} = 14\,\text{TeV}$ and instantaneous luminosity of $L = 10 \times 10^{34}\,\text{cm}^{-2}\text{s}^{-1}$ it holds the record in these measures among concurring experiments.
As a result of consecutive accelerator upgrades, the collider complex has an impressive and complex pre-accelerator structure as shown in fig. 2.1. Consequently, the proton bunches first go through multiple preparation steps before they get injected into the 27 km tunnel of the LHC where the four main experiments (ALICE, ATLAS, CMS and LHCb) and their interaction points are located.

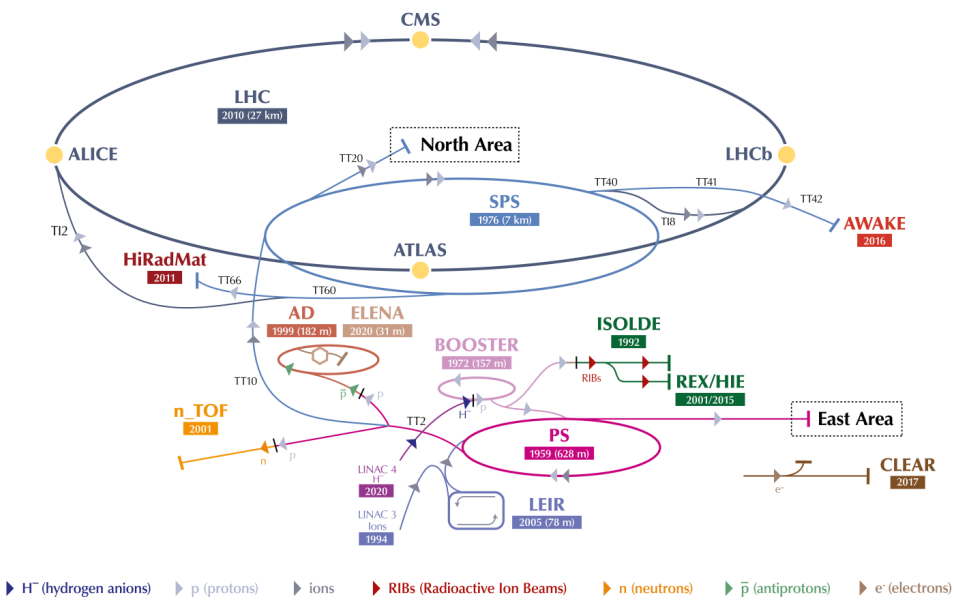

Figure 2.1: The accelerator complex of the LHC, their corresponding construction years and circumferences. Individual stages are shown in different colours, the particle types they accelerate are indicated as arrows. Note that the pre-acceleators do not serve the LHC ring exclusively and the diverging paths lead to other independent experiments. Old tunnels of previous experiments serve as pre-accelerators now in the LHC injection chain. [10]

The four main stages of pre-acceleration for protons are listed in tab. 2.1 below. As proton source, hydrogen is used. The protons are first accelerated in the form of H^- ions through the

86 metre long tunnel of the recently (2020) constructed Linear Accelerator 4 (Linac4). Stripped of their pair of electrons, the protons enter the Proton Synchrotron Booster (PSB) where they reach up to 2 GeV. In the next step of the injection chain, they enter the Proton Synchrotron (PS), historically the first synchrotron at CERN serving exclusively as a pre-accelerator now. Travelling through the 628 metres long ring and accelerated to 26 GeV, the particles are injected into the Super Proton Synchrotron (SPS), where they are awaiting injection into the LHC once they reach 450 GeV.

Accelerator	Peak Energy
Linear accelerator 4 (Linac4)	160 MeV
Proton Synchrotron Booster (PSB)	2 GeV
Proton Synchrotron (PS)	26 GeV
Super Proton Synchrotron (SPS)	450 GeV
Large Hadron Collider (LHC)	7 TeV

Table 2.1: The acceleration chain the protons undergo to reach their final energy of 7 TeV.

In the LHC the beams are circulating in opposing directions. They are kept on a circular trajectory using superconducting NbTi magnets operating at 1.9 K thanks to the superfluid helium bath at about 0.13 MPa [11]. In the tunnel itself there are eight interactions points (IP). ATLAS, ALICE, CMS and LHCb are located at IP1, IP2, IP5 and IP8, respectively. IP3 and IP7 are where the momentum and betatron collimators are located ensuring beam quality; the radiofrequency (RF) cavities are at IP4 increasing the bunch energy to 7 TeV. The beam dump is at IP6 where old bunches are deflected by the fast-pulsing "kicker" magnets and directed towards the carbon absorber at the end of their lifetime. [12]

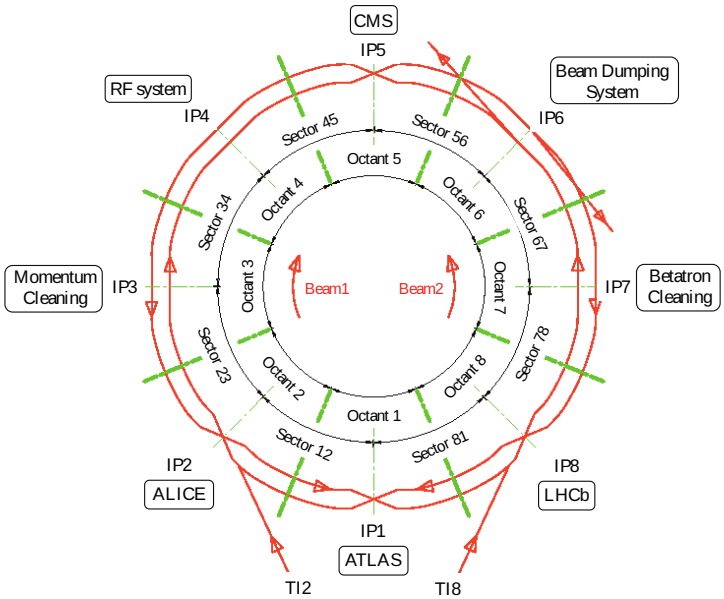

Figure 2.2: The location of the interaction points, the experiments, and the beam adjustment systems. The LHC ring is also divided into further sectors and octants for each IP. [13]

The four main detectors perform the particle collision measurements. LHCb specializes on flavour physics and measures b-quark decays focussing on the measurement and study of CP-violating processes. ALICE has been constructed to mainly study the quark-gluon plasma result-

ing from heavy ion collisions. ATLAS and CMS are sister experiments and are general purpose detectors. In the following, the latter will be described in detail only.

2.2 The Compact Muon Solenoid

The Compact Muon Solenoid (CMS) detector is a 14 000 tonnes heavy detector measuring 28.7 metres in lenght and 15.9 metres in diameters. It has been built to be general purpose detector, featuring additional muon chambers at its most outer side allowing the precise measurement of muon momenta. The detector is built around a solenoid magnet, producing a magnetic field of 4 T. CMS has several subdetector components which surround the beampipe in a layered, onion-like structure. CMS has rotational symmetry around the beam pipe along which the z-axis is usually defined. The positions of each components and some of their most important technical details are shown in fig. 2.3.

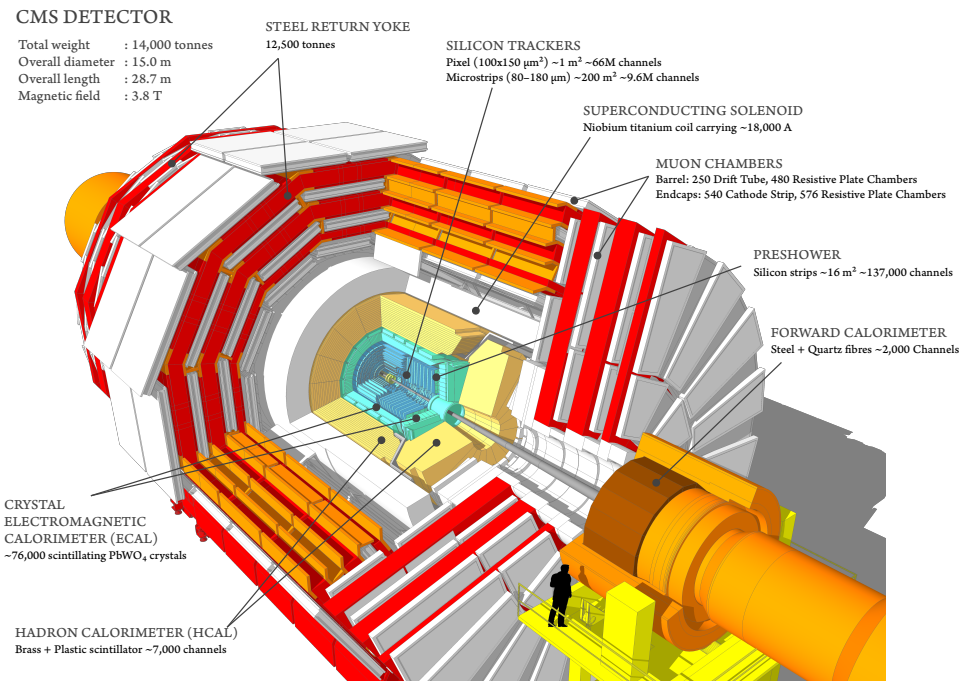

Figure 2.3: The inner structure of the CMS detector. Note the layered structure and rotational symmetry of the detector. Each layer performs an individual step in particle identification. [14]

The coordinate system has its origin at the interaction point, is right-handed, with the z-axis pointing towards the anti-clockwise beam direction. Hence, the xy plane lies orthogonal to the beam pipe. Instead of the polar angle directly the pseudorapidity

$$\eta = -\ln \tan \frac{\theta}{2}$$

is used, as it is an Lorentz-invariant quantity along the beam. $\eta = \pm\infty$ hence corresponds to the beampipe and $\eta = 0$ defines the xy (transversal) plane. Using the pseudorapidity, the differences in the $\eta - \phi$ plane can be given by

$$\Delta R = \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2}$$

With that, particle kinematics can be described in terms of their energy E, the transversal momentum p_T, η, ϕ and their mass m. In some cases, instead of the pseudorapidity, the rapidity

$$y = \frac{1}{2} \ln \frac{E + p_z}{E - p_z}$$

is used.
In the following, the main subdetector components, their location and role focussing on high p_T events will be briefly described.

2.2.1 Silicon Tracking

CMS has an all-silicon tracking system, which lies at the core of the detector close to the beam pipe. As the name suggests, the tracking system enables the reconstruction of the trajectories of the child particles passing through the detector. It thus plays a crucial role in particle identification as the charge of particles can be inferred from the track signature (or the lack thereof); apart from that, it also enables secondary vertex reconstruction of short-living primary particles. Particles passing through the silicon ionize the tracker, producing a signal.
Having a length of 5.4 metres and a diameter of 2.4 metres it covers the pseudorapidity range $|\eta| < 2.5$, the best tracking efficiency being in the barrel region $|\eta| < 0.9$ [15]. Closest to the interaction point lies the pixel silicon tracker. Made-up of four cylindrical layers at 3 cm, 7 cm, 11 cm and 16 cm from the beam pipe at disks at the end, it is segmented into 124 million pixels of 100 µm × 150 µm and kept at −20 °C during operation.
Outside of the silicon pixel detector lie the silicon strip detectors, which is divided into four parts: the inner barrel par (TIB), the inner disk part (TID), the outer barrel (TOB) and the outer endcap part (TEC). This part of the tracker contains about 10 million strip components. The complete layout of the strip detectors is shown in fig. 2.4.

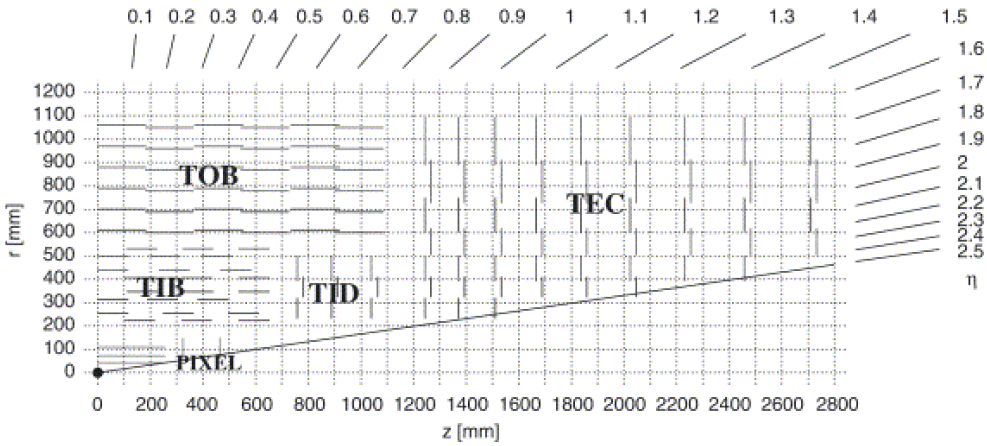

Figure 2.4: View of one quarter of the silicon strip detectors in the $r - z$ plane, indicating both the geometry and the pseudorapidity range of the individual parts. [16]

2.2.2 Electromagnetic Calorimeter

The CMS electromagnetic calorimeter (ECAL) measures the energy of electrons and photons by absorbing them completely and measuring the induced particle shower during that. It lies outside of the tracking system and consists of highly transparent and scintillating lead tungstate

($PbWO_4$) crystals. The ECAL consists of two components as well, and central barrel (EB) covering the region up to $|\eta| = 1.48$ and an endcap region (EE) extending this coverage to pseudorapidities of $|\eta| = 3.0$ [17]. The EB has 61 200 crystals, which are formed into modules weighing around 3 t and containing 1700 crystals, which are 23 cm (or $25.8X_0$ in terms of radiation length X_0) in length. The EC and the end of the barrel modules have almost 15 000 more crystals; the total volume of the crystals is 11 m^3 with a weight of 92 t. The layout of the calorimeter structure is shown in fig. 2.5.

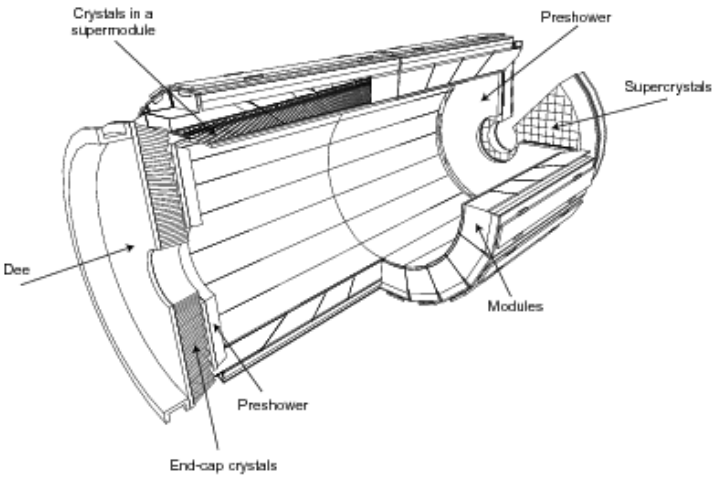

Figure 2.5: Schematic view of the CMS barrel and endcap calorimeters. Note the coverage around the tracker (not shown explicitly) and the preshower module, responsible for better seperation of photon hits in the endcap region. [18]

The energy resolution of the barrel calorimeters has tree terms: a stochastic, a noise and a constant term and was measured to be [18]

$$\frac{\sigma_E}{E} = \underbrace{\frac{2.8\%}{\sqrt{E\,[GeV]}}}_{\text{stochastic}} \oplus \underbrace{\frac{12\%}{E\,[GeV]}}_{\text{noise}} \oplus \underbrace{0.3\%}_{\text{constant}}$$

2.2.3 Hadronic Calorimeter

The CMS hadronic calorimeters (HCAL) is a sampling calorimeter. Similarly to the ECAL, its purpose is to measure the energy of hadrons by measuring the induced avalanche of secondary particles upon passing through the detector as they are not directly stopped in the ECAL. In the $|\eta| < 3$ region it consists of a brass scintillator calorimeter; in the forward region $3 < |\eta| < 5$ in has a iron quartz-fiber calorimeter [19]. It is organized into barrel (HB), outer barrel (HO), endcap (HE) and forward (HF) regions. The layout and the positioning of the respective components are shown in fig. 2.6. Combined with the ECAL, the HCAL measures jets with a resolution of [20]

$$\frac{\sigma_E}{E} = \frac{100\%}{\sqrt{E\,[GeV]}} \oplus 5\%$$

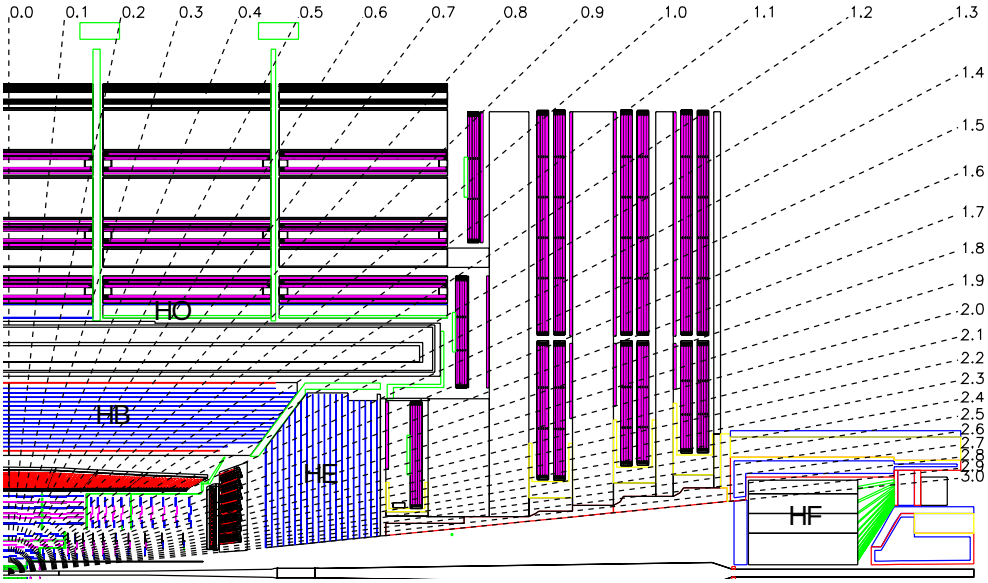

Figure 2.6: The CMS HCAL module setup. Note that the outer calorimeter lies outside of the solenoid (the white rectangle between HB and HO). [21]

2.2.4 Superconducting Solenoid Magnet and the Return Yokes

A core component of the detector is the solenoid magnet itself. Weighing 12 000 t and cooled down to $-268.5\,°$C it can produce a magnetic field of 4 T (3.8 T during the runs), it is the largest solenoid in the world. The solenoid magnet and the return yokes play a crucial role in particle identification and muon measurement. These high magnetic fields are achieved using NbTi coils and a current of 18 160 A. Outside of the solenoid magnet itself lie the muon chambers and the five return yokes, which are constructed from steel. The field strength thanks to the yokes there is around 2 T.

2.2.5 Muon Chambers

In the outermost layers of the detector, the muon chambers can be found. CMS specializes in muon measurements and these chambers are characteristic for the whole detector. As muons provide a clear signature (like in a $H \rightarrow ZZ \rightarrow \mu\mu\mu\mu$, which is referred to as the "Golden Channel"), their reconstruction is highly motivated. Due to muons having several orders of magnitude higher mass compared to electrons, they can pass the ECAL and they are not stopped by it. Hence, they do not deposit their complete energy there meaning they can escape all the inner detector components. For this reason, additional measurements on their momenta are needed.

The chambers have four subcomponents: the drift tubes (TB), the cathode strip chambers (CSCs), the resistive plate chambers (RPCs) and the gas electron multipliers (GEMs), all of which are gaseous detectors. In the region $|\eta| < 1.2$ the four layers of drift tubes have been installed at a radius of approximately 4 m, 4.9 m, 5.9 m and 7 m. The tubes consist of 250 chambers and looking from the transversal plane, they are grouped into 12 sectors, with each covering 30° azimuthal angle. These chambers are staggered in a way, such that a high p_T muon produced a the sector boundary crosses at least 3 of the four layers. In the endcap region up to $|\eta| < 2.4$, the CSCs are deployed. The CSCs are used both in the endcap and the barrel region. The newly

added GEMs have been installed in the region $1.6 < |\eta| < 2.2$ [22]. The exact position of the components are shown in fig. 2.7.

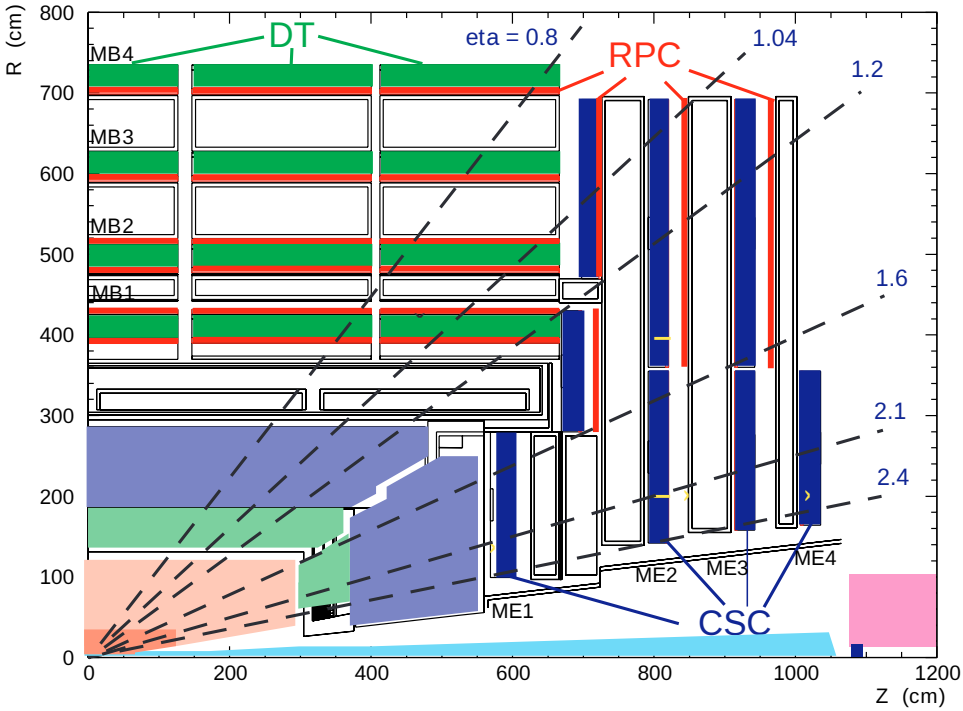

Figure 2.7: The location of the drift tubes (DT), the cathode strip chambers (CSCs) the resistive plate chambers (RPCs). The newly deployed GEMs are in the region $1.6 < |\eta| < 2.2$ (not shown here) [23].

2.2.6 Trigger System

With a collision frequency of 40 MHz with a new collision in every 25 ns it is both impossible and unnecessary to record every event in the detector; impossible because the complete registration of this enormous data-flow is technologically impossible and unnecessary as most of these events arise from soft collision processes. For this reason, a filtering (triggering) mechanism has been implemented in the detector.

The trigger system reduces the dataflow in two steps. The Level-1 (L1) trigger operates on a hardware level and reaches a decision in less than 1 µs. During the decision-making period, the high-resolution data is held the memory and increasingly complex algorithms are used to approach the quality of final reconstruction. These decisions involve information from the calorimeters and the muon systems and some correlation between them. It is based mostly on jet E_T and p_T thresholds and physics objects like photons, electrons and muons. The resulting data-stream from the L1 trigger is less than 100 kHz.

In the second step, the High-Level Triggers continue with the processing. They operate on a software level and have an output rate of 100 Hz which can be stored offline for further analysis. Instead of fully reconstructing events from every detector component, they are first only partially reconstructed and eventually discarded as soon as possible [23].

2.3 Physics Object Reconstruction

2.3.1 Particle Flow

Individual particles leave (up to the detector resolution) unique event signatures in each subdetector component, which is summarized in fig. 2.8. In order to perform a complete physics analysis, these event signatures have to be assigned to physically interpretable objects in order to obtain the physical final state and kinematics such as energy, momenta and charge.

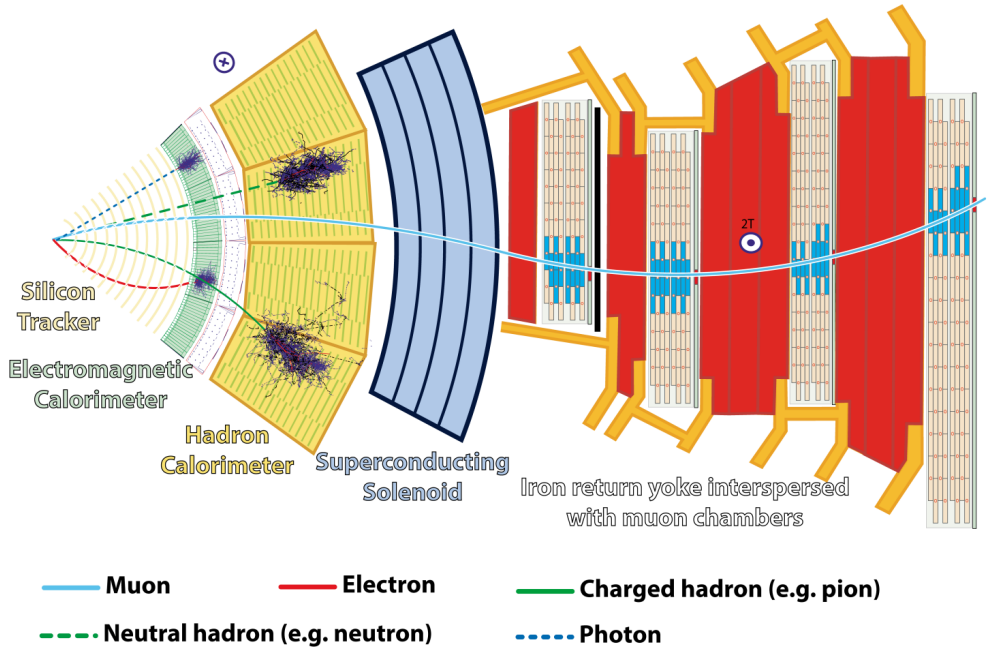

Figure 2.8: Transversal view of a slice of the CMS detector with several characteristic signatures drawn in different colours. Note that each detector slice functions as a filter for different particle kinds, i.e. the number of signatures decreases towards the outer layers. [24]

The reconstruction algorithm used at the CMS experiment is Particle Flow (PF) [25]. In order to obtain the physics objects, it first combines the particle tracker hits (both from the tracker and from the muon chambers) and clusters the energy deposits in the calorimeters. In order to keep the number of misreconstructed tracks minimal during track reconstruction several quality fit criteria are relaxed with each iteration. For the calorimeter clusters, an expectation-maximization algorithm based on a Gaussian-mixture model is used to reconstruct the clusters within the computed topological clusters around the pre-constructed cluster seeds.
Following that, the link algorithm is executed which connects the aforementioned PF elements from different subdetectors. At this point, it cannot be avoided that some particles are misidentified or wrongly reconstructed, especially in cases where a large missing transverse momentum p_T^{miss} is present in the event. As such events (usually originating from misreconstructed high p_T muons) may be wrongly selected by a large set of new physics searches, they undergo a post-processing algorithm in order to correct these effects.
Finally, the reconstructed particles are used to build the physics objects, namely jets, missing transverse momenta p_T^{miss}, muon, electrons and taus and other related quantities. Note that as a hadron collider, the exact centre-of-mass system is a priori unknown due to the interacting partons within the proton structure, hence the missing momenta can only be evaluated in the

transversal plane.

2.3.2 Jet Clustering

The hadronisation and fragmentation processes give rise to an avalanche of secondary particles in the detector called jets. Their reconstruction requires sophisticated tools to assign the detector hits to specific clusters. Such algorithms should ideally satisfy two requirements: they should be infrared and collinear safe, i.e. they should not be disturbed by soft emmisions (particles with low momenta) while collinear particles belonging to the same jet should be assigned correctly. CMS uses the anti-k_T algorithm, a generalization of the k_t and Cambridge/Aachen algorithm [26]. They are given by[1]

$$d_{ij} = \min\left(k_{ti}^{2p}, k_{tj}^{2p}\right)\frac{\Delta_{ij}^2}{R^2}$$
$$d_{iB} = k_{ti}^{2p}$$

where $\Delta_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2$ is the distance in the plane spanned by the rapidity y and the azimuthal angle ϕ between particles i and j, R is a radius parameter, k_{ti} is the transverse momentum of particle i and d_{iB} is the distance between the particle and the beam. Setting the parameter p to $p = -1$ results in the both infrared, collinear safe and fast anti-k_t algorithm.
Anti-k_t accumulates soft particles which are in a radius R of a hard particle if no other hard neighbours are present within $2R$ to the hard particle themselves in a perfectly conical jet. If there is another hard particle with $R < \Delta < 2R$, then two not necessarily conical jets will be returned; for $\Delta < R$, both particles will form a single jet. From here one can see the stability of the algorithm with respect to soft emmisions while maintaining collinear safety.
One can differentiate two kinds of jets depeding on the radius R. $R = 0.4$ or $R = 0.8$ results in AK4 (resolved) or AK8 (boosted[2]) jets, respectively. Nevertheless, the quark flavour which produces the jet cannot simply be obtained by jet clustering. Consequently, flavour identification (flavour tagging) has to be performed separately.

2.3.3 Flavour Tagging

Assigning flavour to the jet-inducing quarks is possible through several heavy-flavour jet discriminating variables, which can be reconstructed by deep neural networks [27].
For b quarks, the displacement of the secondary vertex due to the long enough lifetime of b hadrons which is at the order of 1.5 ps (which corresponds to displacements of a few mm to 1 cm) can be reconstructed. These displacements can be resolved with the tracker and can be characterized by the impact parameter (which is defined as the vector pointing from the primary vertex to the point of closest approach) of the child particles. Due to the heavier mass of the b and c quarks, the decay products have a larger transversal momentum with respect to the jet axis yielding larger impact parameter vectors compared to those of their light counterparts. In addition to that, in approximately 20% and 10% of the cases in the decay chain of b and c hadrons, respectively, a muon or an electron is present, which can be further used for heavy-flavour jet identification.

[1] Setting $p = 0$ yields the Cambridge/Aachen algorithm, while $p = 1$ one recovers the k_t algorithm.
[2] or, in "CMS slang": fat

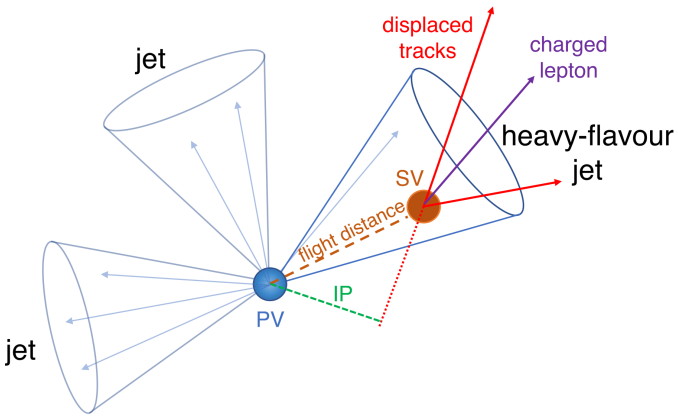

Figure 2.9: Schematic view of the displacement of the secondary vertex with respect to the primary vertex with the definition of the impact parameter indicated. Note the possible appearence of a soft lepton (shown in purple) in the jet cone characteristic for heavy-flavour jets.

2.3.4 Muons

Since muons can leave signatures in both the tracker and the muon system, they are reconstructed and categorized depending on the muon signal quality and their physics properties. For the track reconstruction, the three categories in increasing quality are: [28]

Standalone-muon tracks are constructed from the CSC, RPC and DT signatures using a Kalman-filter technique.

Tracker muon tracks are built "from the inside out" by loose matching tracker tracks (with $p_T > 0.5\,\text{GeV}$ and a total momentum of $p > 2.5\,\text{GeV}$) to DT and CSC segments. tracks automatically qualify if at least one muon segment matches the extrapolated track.

Global muon tracks are obtained "outside-in" by matching the standalone-muon tracks with the tracker tracks with the Kalman filter.

Note that these categories are not exclusive. Thanks to the high resolution and efficiency of muon track reconstruction, approximately 99% of the muons are reconstructed as either as a tracker track muon, global muon track or both.
Following the categorization by the tracks, the muon IDs are assigned [29]. Using the variables from the track fit, the following identification types are assigned:

Loose muon ID are for muons selected by the PF algorithm which is either a tracker or global muon. This selection ensures the identification of prompt muons originating from the primary vertex or from light and heavy flavour decays while minimizing the rate of charged hadrons misidentified as muons.

Medium muons are prompt muons and muons from heavy flavour decays which carry the loose ID with tracker track hits of more than 80% of the inner tracker layers.

Tight muon IDs suppress hadronic punch-through and muons from decay in flight. They are both tracker and global muons with at least pixel hit and hits from at least six inner layer tracker hits, fulfilling additional event geometric criteria.

Soft muons are low p_T muons with high purity tracker track of the same tracker requirements as tight muons.

High momentum muon IDs are for muons with $p_T > 200\,\text{GeV}$. They are both tracker and global muons with the same tracker hit and event geometric requirements as tight muons. In addition to that, they need at least one hit from the muon system for the global muon.

3 Deep Learning and Conditional Invertible Neural Networks

Deep Learning is a subdomain within Machine Learning focussing on the construction and training of deep neural networks. Such networks have been proven to be extremely successful in different, analytically unsolvable problems such as (but not restricted to) image classification tasks, regression tasks, image generation tasks, language translation and reinforcement learning.
In this chapter, a brief introduction to these networks and their properties with a focus on so-called normalizing flows and invertible neural networks is given.

3.1 Foundations of Deep Learning and Neural Networks

3.1.1 The Universal Approximation Theorem

The success of Neural Networks (NNs) lies in their potential of learning (almost) arbitrary models. This property of NNs can be expressed mathematically through the Universal Approximation Theorem which in words states a feed-forward network with linear output and at least one hidden layer with a finite number of nodes can approximate any real continuous function on a given closed and bounded subset to arbitrary precision [30].
However, the Universal Approximation Theorem does not state how the network should be constructed to achieve the desired precision. For this reason, the theorem has been proven for several network architectures; in case of ReLU-activated feed-forward networks the theorem can be written as the following [31]:

Theorem 1 *For any real and continuous function $f : [0,1]^{d_{in}} \to \mathbb{R}^{d_{out}}$ and every $\epsilon > 0$ there is a ReLU-network $\mathcal{N}$ with the same input and output dimension d_{in} and d_{out} and a minimum hidden layer width w_{min} for which*

$$\sup_{x \in [0,1]^{d_{in}}} \|f(x) - f_{\mathcal{N}}(x)\| \leq \epsilon$$

and

$$d_{in} + 1 \leq w_{min}(d_{in}, d_{out}) \leq d_{in} + d_{out}$$

This theorem does not state anything about the exact depth (number of layers) the network needs to have, its speed of convergence and the optimization process it needs to undergo to achieve this arbitrary approximation. On the other hand, it is reassuring to have a mathematical guarantee for convergence for a given feed-forward network structure. For this reason, empiric studies are usually performed to look for a locally optimal solution for a given task.
In the following, a brief overview of the most common NN components is going to be given with special focus on the ones used for the construction of the cINN.

3.1.2 Fully-Connected Neural Networks

The most elementary NNs perform affine transformations (consisting of a linear transformation represented by a weight matrix W and a shift b) on a given input x followed by the application of a non-linear function (called activation) g:

$$y = g(Wx + b)$$

Graphically, one such transformation is represented as a layer. For a fully-connected neural network, these transformations are called in succession, resulting in stacked layers of n nonlinearities as shown in fig. 3.1. The resulting mapping

$$f(x,\theta) = g^n \left\{ W^{n-1} g^{n-1} \left[... \left(W^2 g^1 (W^1 x + b^1) + b^2 \right) ... \right] + b^{n-1} \right\}$$

is a universal approximator with the free parameters θ (containing all the matrices W^i and the biases b^i) for the target output for y. Finding the optimal parameters θ is called training and the θ will be referred to as trainable parameters. The graphical representation of a neural network mapping is shown in fig. 3.1.

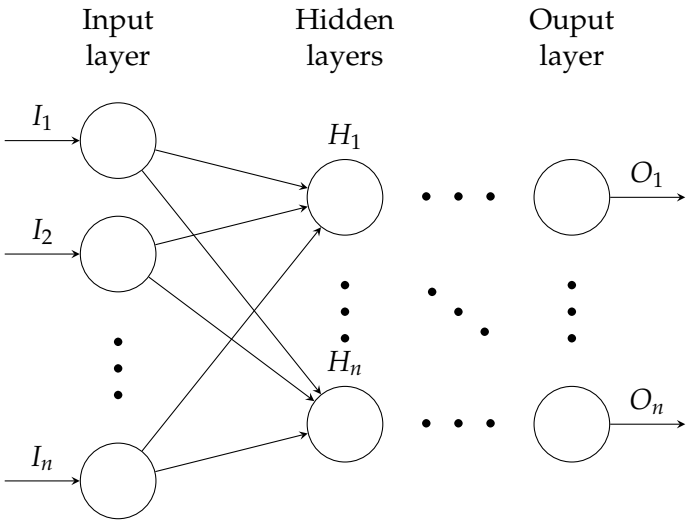

Figure 3.1: Schematic structure of a feed-forward NN. The nodes represent the variables for the hidden and output layers or the biases and the activations g applied to the intermediate outputs for the hidden layers. Arrows represent the matrix W.

Note that for the approximation to work arbitrary well, the structure of the network and training have to be correctly adjusted to the task to be solved.

3.1.3 Activation Functions

It is essential for convergence to select the right activation function g. Historically, several candidate activation functions such as sigmoid, tanh, the rectified linear unit (ReLU) and its variants such as SeLU and leaky-ReLU have emerged. In this work ReLU will be used exclusively. An advantage of ReLU is the lack of saturation range compared to the sigmoid or the tanh activation functions, where gradients above or below a given value of x become negligible, "paralysing" the training. Apart from that, ReLU has tractable derivatives and has proven to be an efficient activator empirically. It is defined as

$$\mathrm{ReLU}(x) = \max\{0, x\}$$

where the maximum is taken element-wise over the vector components of x. Note that its derivative is not continuous and makes a jump at the origin from 0 to 1, which unnecessarily renders parameters 0 during training. On the other hand, this activation function has no scale, making it an excellent candidate for general approximation tasks. A graph of the most commonly used activation functions is shown in fig. 3.2.

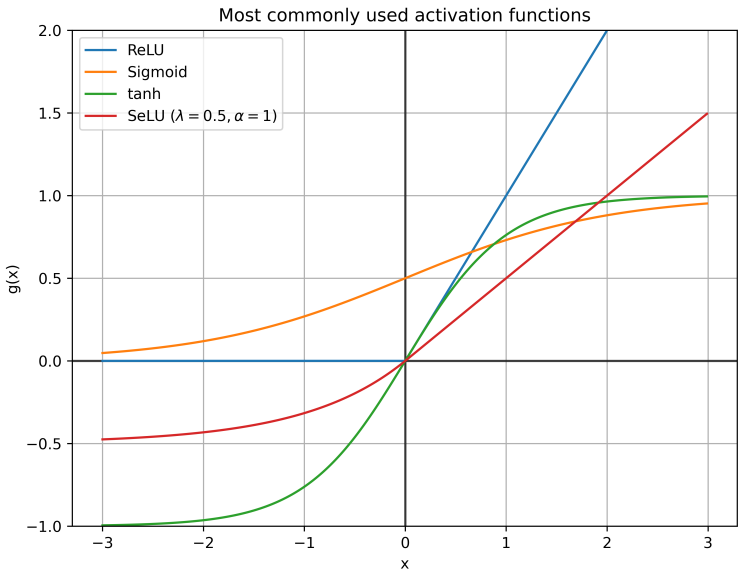

Figure 3.2: Most commonly used activation functions. Note the plateaus of the sigmoid and tanh functions and for all $x<0$ for ReLU. Setting the gradient zero in case of ReLU for $x<0$ renders several parameters inactive during training; on the other hand, large positive parameters do not get stuck due to vanishing gradients and the function is scaleless.

3.1.4 Objective Functions

Having established the network structure, a measure of fit quality remains to be chosen. Such measures are called objective functions (error or loss functions) and used to govern the training by setting the gradients of the trainable parameters θ. Several loss functions have been established for each NN-specific task to be solved. The mean-squared-error (MSE) has been found useful for regression tasks due to its resemblance to the least-squared-error obtained from likelihood fits of normal distributed data. The so-called cross-entropy error function has been found performant for classification tasks due to the relation of the loss to information theory and the accessible probabilistic interpretation of the outputs following sigmoid activation. Both of these loss functions need target labels to compare the network outputs to. Training procedures where fixed labels guide the parameter estimation are called supervised learning.

Distribution fitting requires the loss function to compare samples from a probability distribution function to a target distribution. Since the discrete labels are replaced with continuous target distributions and none of the datapoints has a well-defined target these training procedures belong to the class of unsupervised learning. As the name suggests, the neural network does not obtain any additional feature representation of the data necessary for the optimization procedure. In layman's terms the network has to "figure out on its own" which input features are more important for the given task.

In this work, distribution fitting will be performed which is reflected by the choice of the loss function. The Kullback-Leibler divergence (also called relative entropy) is a measure of resemblance of $p_{\mathrm{x}}(x;\theta)$ to $q_{\mathrm{x}}(x)$ both defined over the random variable x and belongs to the families of f-divegences defined as

$$D_f\left[q_{\mathrm{x}}(x)\ ||\ p_{\mathrm{x}}(x;\theta)\right] = \mathbb{E}_{p_{\mathrm{x}}(x;\theta)}\left[f\left(\frac{q_{\mathrm{x}}(x)}{p_{\mathrm{x}}(x;\theta)}\right)\right]$$

with the convex function $f(r) = r\log r$ [32]. In its expanded form it can be written as

$$\begin{aligned}\mathbb{KL}\left[q_{\mathrm{x}}(x)\ ||\ p_{\mathrm{x}}(x;\theta)\right] &= \mathbb{E}_{q_{\mathrm{x}}(x)}\left[\frac{\log q_{\mathrm{x}}(x)}{\log p_x(x;\theta)}\right] \\ &= \underbrace{-\mathbb{E}_{q_{\mathrm{x}}(x)}\left[\log p_{\mathrm{x}}(x;\theta)\right]}_{\mathcal{L}(\theta)} + \underbrace{\mathbb{E}_{q_{\mathrm{x}}(x)}\left[\log q_{\mathrm{x}}(x)\right]}_{const}\end{aligned} \tag{3.1}$$

and can be decomposed into a θ-parameter dependent and a θ-independent term[3]. The KL-divergence has a global minimum of value 0 where both distributions are identical. With this expression, one can approximate an unknown posterior distribution $p(x|y)$ using a model function $p_\theta(x|y)$ (both conditioned on a random variable y) by taking the parameter-dependent term $\mathcal{L}(\theta)$ as a basis for the loss function.
Objective functions "guide" only the training, but are not a measure about the quality of the predictions per se. It can occur in terms of the weights, that a given feature of one or several inputs enjoys a higher-than-optimal weight at the cost of generalization (overfitting and overtraining). For this reason, the entries of the weight matrix during gradient descent are penalized with an additional L^2 regularization term

$$L'(\theta) = L(\theta) + \alpha\,||\theta||_2^2 \tag{3.2}$$

where α is a fixed regularization parameter.
What is missing from a complete training is data the network is trained with respect to and the optimizer looking for an optimum of the loss function.

3.1.5 Training and Loss Optimization

The optimization is data-driven. The dataset is split up to two disjoint datasets: a training and a test set. For every iteration a number of samples from the training set are evaluated through the network from which the current network performance can be evaluated. Ideally, finding a local optimum of a one-dimensional function requires a closed form of the function itself with respect to its parameters. Unfortunately, this is usually not the case due to the nonlinear nature of the optimization problem. For this reason, the weight matrices are adjusted for every node through backpropagation: the gradients for each hyperparameter are calculated from the previous layer and the parameters are updated through gradient descent along the negative of the parameter gradient moving slowing to a local optimal value. For a simple case, it can be intuitively written with a positive learning rate η as

$$\theta^t = \theta^{t-1} - \eta\left[\frac{\partial\mathcal{L}}{\partial\theta}\right]^{t-1} \tag{3.3}$$

Note that the learning rate is a decisive parameter in the update process; while a too small value of η does not improve the loss significantly, a large η on the other hand cannot achieve the de-

[3]Note that the Kullback-Leibler divergence is not symmetric in its arguments and swapping q and p does not yield a similar decomposition due to the expectation value taken over the probability density function $p_{\mathrm{x}}(x;\theta)$. Hence a distinction between the forward and the backward Kullback-Leibler divergence has to be made. In to following, only the forward KL-divergence (introduced above) will be used.

sired convergence. To speed up the optimization process, this simple idea has been improved further upon and several competing optimizers have emerged. In this work, the quasi-standard Adam optimizer [33] has been used. It uses a sophisticated self-tuning learning rate optimization (taking the "momentum of convergence" into consideration) and several, empirically useful parameters to optimize the convergence. The algorithm can be summarized as the following:

$$\begin{cases} \theta^{t+1} & = \theta^t - \eta \cdot \frac{\hat{m}^t}{\sqrt{\hat{v}^t + \epsilon}} \\ g^t & = \nabla_\theta f^t(\theta^{t-1}) \\ m^t & = \beta_1 \cdot m^{t-1} + (1 - \beta_1) \cdot g^t \\ v^t & = \beta_2 \cdot v^{t-1} + (1 - \beta_2) \cdot g^t \\ \hat{m}^t & = \frac{m^t}{1-\beta_1^t} \\ \hat{v}^t & = \frac{v^t}{1-\beta_2^t} \end{cases}$$

Note that while Adam evaluates the next step using multiple measures in a sophisticated manner, in its core and using the right parameters the same gradient descent from eq. 3.3 can be found. It has been found experimentally that the parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$ are suitable choices for general tasks.

In order to further improve the optimization, the updates are evaluated for a fixed number of samples (batches). The main advantage of stochastic gradient descent is the consecutive updates towards a more general optimum and avoiding getting stuck at local optima due to strongly varying gradients evaluated from the dataset. Once the whole training dataset has been used up, an epoch has reached its end and the process starts anew with a shuffled training dataset. After several epochs, the procedure converges. In order to monitor the network performance and its generalization capabilities, the test set is used. For this, the loss can be evaluated for these "unseen" datasamples, which shows signs of anomalies during training. Should for example the test loss rise while the training loss decreases, one can be certain that overtraining has had occurred.

3.2 Conditional Invertible Neural Networks

Conditional Invertible Neural Networks have been found performant in machine learning tasks such as image colouring and guided image generation with style transfer [34], parton shower generations and detector effect simulation [35], stellar parameter determination [36] and cosmic ray source property inference [37].

Most neural network setups tackle the challenge of posterior inference where the probability distribution

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)}$$

in Bayes' Theorem cannot be evaluated analytically. Most of these models are constructed in a one-directional way and do not preserve probability during inference. For this reason, even for (potentially) invertible problems, the model inverse is ill-defined. To avoid this, the network architecture has to be adjusted to satisfy the requirements of invertibility.

3.2.1 General INN Architecture

To ensure invertibility the network input and output dimensions have to match to fulfil bijection. General invertible neural networks map the inputs x to their targets y while encoding some information regarding x in the latent variables z. These variables are assumed to follow a fixed distribution which is also represented in the loss function. Usually these distributions are chosen to be a multivariate normal distribution, as they are unbiased and easy to sample from, which is crucial for obtaining the posterior distributions $p(x|y)$. Indeed, it can be shown, that upon perfect convergence, sampling from the latent space and picking an y network inference yields the true posteriors $p(x|y)$ for any y [38].

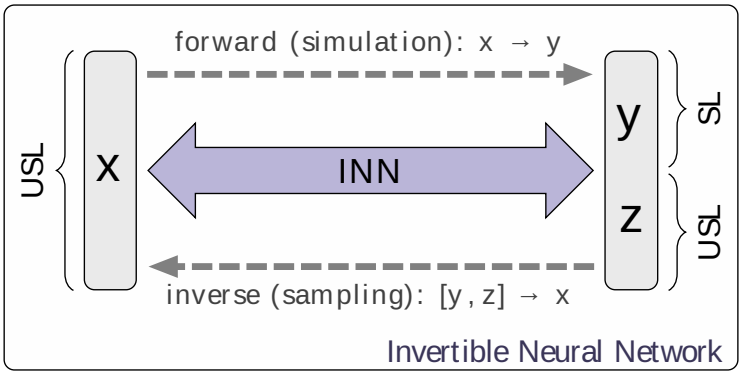

Figure 3.3: General INN structure. Uniquely to these structures[4], the network can be trained in both directions with a supervised loss (SL) term for the target labels y and an unsupervised loss term (USL) for the latents z and inputs x. [38]

A conceptual disadvantage arises from the conditions being included in the network outputs. Assuming the dimensionality of x and y are wide apart from each other zero-padding or several noise-extending techniques are necessary to construct the network potentially introducing unnecessary redundancies. This can be fixed by removing the conditions y from the output and by using them as an additional input for the network and by setting the dimension of the latent space distribution to match that of the inputs x. With this approach, the conditions can also be pre-processed by applying a – either pre- or jointly with the network trained – network (a conditioning or summary network) [39]. Such setups enable processing of highly correlated or feature-rich data at the cost of potential overfitting [36]. A sketch of the resulting network – the conditional invertible neural network (cINN) – structure is shown in fig. 3.4.

By constructing an invertible structure, the challenge of exact posterior inference remains with the added difficulty of mapping the inputs to the latent space distribution (LSD) in a meaningful (invertible and information conserving) way. To tackle this challenge, normalizing flows (NF) have to be applied.

[4]but not for cINNs

Figure 3.4: A sketch of the network setup with a summary network

3.2.2 Normalizing Flows

The main idea of NFs is to construct an invertible transformation T which maps a probability distribution p to a target probability distribution q in an invertible way. Hence, both T and its inverse T^{-1} have to be invertible and differentiable to preserve the probability flow. Such mappings are called diffeormophisms; note that a composition of diffeomorphisms

$$T' = T_1 \circ T_2 \circ ... \circ T_N$$

yields a diffeomorphism too.
In a historic sense a whitening transformation can be already viewed as a normalizing flow the target distribution being a (multivariate) normal distribution. In more general and complicated cases however, the transformation T cannot be easily constructed by hand. As the transformation is applied to samples of the distribution p, the flow has to normalize the distribution internally.
Assume we would like to express the probability $p(x)$ using the target distribution. For this, a straightforward coordinate transformation involving the Jacobian from the source to the target space can be used:

$$p(x) = p(z)\,|\det J_T| = p(z)\left|\frac{\partial x}{\partial z}\right|$$

The backward transformation can be written using the inverse of T and its determinant

$$p(z) = p(x)\,|\det J_{T^{-1}}| = p(x)\left|\frac{\partial z}{\partial x}\right|$$

Here we can directly observe the explicit reason why T has to be diffeomorphism. A graphical representation of a normalizing flow for two-dimensional distributions can be seen in fig. 3.5.

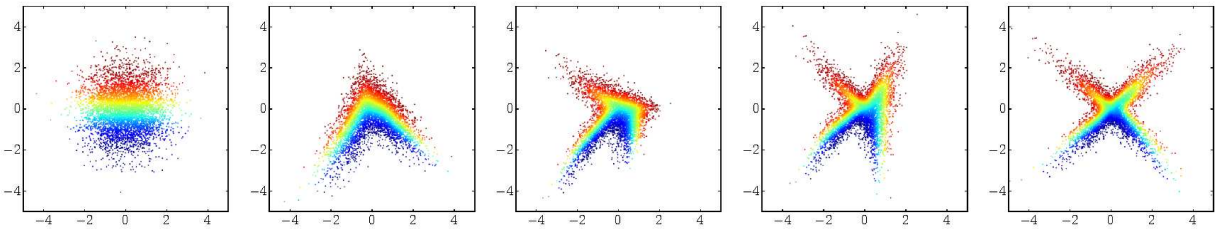

Figure 3.5: Example of a normalizing flow transforming a normal distribution to a star-shaped one in 4 steps. Note that the subsequent steps are also done with diffeomorphisms while the resulting transformation still remains a diffeomorphism. [32]

cINNs use neural networks to learn the mapping T. In the following, the construction of the diffeomorphic transformation will be discussed in detail.

3.2.3 Affine Coupling Blocks

To represent the transformation T Affine Coupling Blocks (ACB) can be utilized. In this work, exclusively the Generative Flow (GLOW) coupling block [40] will be used. The GLOW block follows a design similar to the RealNVP block from [41]. These blocks in practice need to fulfil at least two conditions: first, they have to invertible and second, the Jacobian of the mapping it

represents has to be numerically tractable, especially its determinant. The GLOW block fulfils both of these conditions. In order to do so, it first splits up its input $u = [u_1, u_2]$ in two equal parts, on which a scaling $s_i(\cdot)$ and a translation $t_i(\cdot)$ using arbitrary, non-invertible functions are applied. The resulting transformation reads

$$\begin{cases} v_1 = u_1 \odot \exp(s_2(u_2)) + t_2(u_2) \\ v_2 = u_2 \odot \exp(s_1(v_1)) + t_1(v_1) \end{cases} \tag{3.4}$$

where $\odot$ denotes the element-wise (Hadamard) product. Note that although the functions $s_i(\cdot)$ and $t_i(\cdot)$ are not necessary invertible themselves and can be approximated thanks to the Universal Approximation Theorem with neural networks, the transformation in eq. 3.4 is invertible, as u_1 and u_2 can be expressed with v_1 and v_2:

$$\begin{cases} u_1 = (v_2 - t_1(v_1)) \odot \exp(-s_1(v_1)) \\ u_2 = (v_1 - t_2(u_2)) \odot \exp(-s_2(u_2)) \end{cases}$$

At this point, the exponential functions may seem arbitrary. Looking at the Jacobian

$$\frac{\partial(v_1, v_2)}{\partial(u_1, u_2)} = \begin{pmatrix} \exp(s_2(u_2)) & \frac{\partial v_1}{\partial u_2} \\ 0 & \exp(s_1(v_1)) \end{pmatrix}$$

and its determinant

$$\left| \frac{\partial(v_1, v_2)}{\partial(u_1, u_2)} \right| = \exp\left(\sum_j s_2^j(u_2) + s_2^j(v_1) \right)$$

it can be seen that they keep the determinant is nonzero independently of the behaviour of the scaling transformations $s_i(\cdot)$. It becomes also clear from the determinant, that even for high-dimensional input and output vectors u, v the determinant is computationally cheap to evaluate.

Due to their mutual dependence on each other the expressions in eq. 3.4 are not evaluated directly in practice and the transformation is split up in two intermediate steps. First, a term with

$$f_1(u_1, u_2) = (v_1, u_2) = \begin{cases} u_1 \odot \exp(s_2(u_2)) + t_2(u_2) \\ u_2 \end{cases}$$

is evaluated followed by

$$f_2(v_1, u_2) = (v_1, v_2) = \begin{cases} v_1 \\ u_2 \odot \exp(s_1(v_1)) + t_1(v_1) \end{cases}$$

It is easy to check, that the composed transformation $f = f_2 \circ f_1$ has the same Jacobian (and hence the same determinant) as the one in eq. 3.4. This step-wise evaluation can be visualized in a flow chart, as shown in fig. 3.6.

Note that the exponential functions in eq. 3.4 can potentially produce extreme values which can lead to numerical instabilities during optimization. For this reason, additional soft-clamping is introduced, and an additional scale function with a fix clamping coefficient α is applied to the scale coefficients $s_i(\cdot)$:

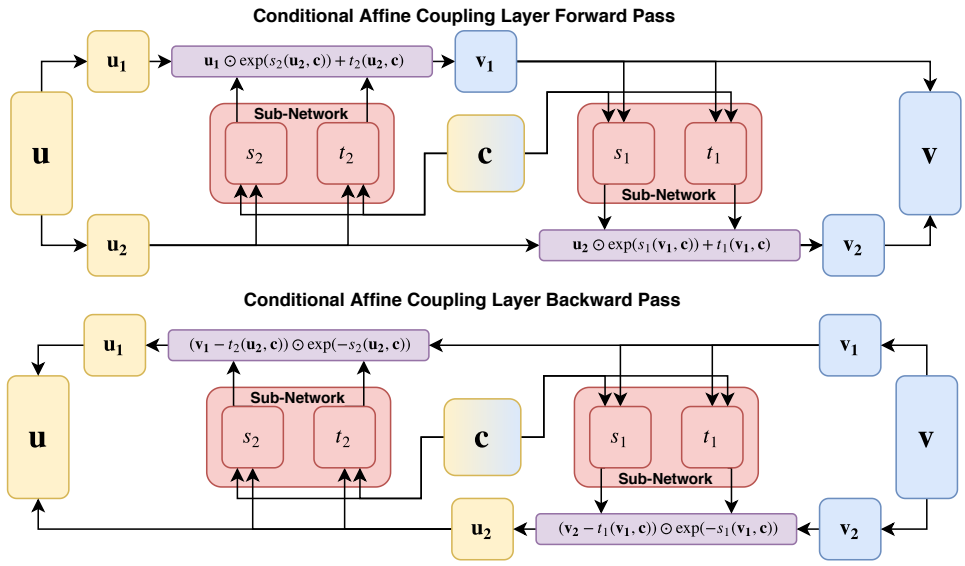

Figure 3.6: The forward and the backward pass of the GLOW block. Note that per construction, the sub-networks do not have to invertible themselves, and the block can be evaluated in both directions. The conditions node c is used as an additional input for each sub-network. [36] replace with Bister et al. sketch?

$$s' = \frac{2\alpha}{\pi} \arctan\left(\frac{s}{\alpha}\right)$$

For $\lim_{s \to \pm\infty} s' = \alpha$, while $s' \approx s$ if s is of the same order of magnitude as α. This naturally dampens the gradients from one block to the other meaning α has to be carefully chosen to avoid information loss between the consecutive blocks. For most analyses clamping coefficients of $\alpha \approx 1.9$ suffices.

With the construction of the normalizing flow with the GLOW block and the subnetworks representing the scaling $s_i(\cdot)$ and transitions $t_i(\cdot)$, the setting of the loss function using the Kullback-Leibler divergence remains to construct the whole cINN.

3.2.4 The Kullback-Leibler Divergence as a Loss Function

Our goal is to approximate the real posterior distribution of a physics parameter x under the observation y, i.e. we would like to write:

$$p(x|y) \approx p_\theta(x|y)$$

With the Kullback-Leibler divergence from 3.1, this can be formalized with $q(x) = p(x|y)$ and $p(x;\theta) = p_\theta(x|y)$ as

$$\mathbb{KL}\left[p(x|y)||p_\theta(x|y)\right] = -\mathbb{E}_{p(x|y)}\left[\log p_\theta(x|y)\right] + const.$$

This form of the KL-divergence can be understood in terms of likelihoods as finding the optimal parameters that maximizes the likelihood is asymptomatically equivalent to finding the distribution which has minimal distance to the real distribution in terms of the KL-divergence [42]. This expression can now be rewritten using the Jacobian and the known and fixed latent space

distribution as

$$\begin{aligned}\mathbb{KL}\left[p(x|y)||p_\theta(x|y)\right] &= -\mathbb{E}_{p(x|y)}\left(\log p(z)\left|\det\frac{\partial z}{\partial x}\right|\right) + const. \\ &= -\mathbb{E}_{p(x|y)}\left(\log p(z) + \log\left|\det\frac{\partial z}{\partial x}\right|\right) + const.\end{aligned}$$

where $z = f(x, y, \theta)$ is the output of the network in the latent space. At this point we can fix the distribution $p(z)$ to be a multivariate Gaussian of mean 0 and standard deviation 1. Writing

$$p(z) \propto \mathcal{N}(z; \mu = 0, \Sigma = \mathbb{1}) \propto \exp\left(-\frac{z^2}{2}\right)$$

the resulting loss function becomes

$$\mathcal{L}(\theta) = \frac{1}{M}\sum_{m=1}^{M}\frac{f(x_m;\theta)^2}{2} - \log\left|\det\frac{\partial f(x_m;\theta)}{\partial x}\right|$$

for all inputs x_m of batch m to train the network with respect to.

3.2.5 Posterior Inference

After training, the network should be capable of inferring the true posterior distribution. For this, a condition has to be fixed first. Then, one can select samples from the latent space distribution (as its shape has been fixed by the loss function) and invert the network to obtain the samples from the posterior distribution. At first glance there is no guarantee from the loss and from the architecture only that it is meaningful to map the datapoints to a more or less arbitrary function, from which independently drawn samples can be used for posterior inference. However, it can be shown that the latent output distribution is statistically independent on the training data at the global optimum, and the latent output distribution is the same for any fixed condition [39]. This motivates the rightfulness of the sampling from the latent space.

4 Analysis- Strategy and Setup

In this chapter, an overview of the analysis- strategy and setup will be given with a special focus on the physical observable used as the condition for the cINN inference.

4.1 Analysis Strategy

The analysis has been performed on Monte Carlo (MC) data exclusively. The phase space in consideration has been selected to contain the highest signal yield. Hence the $gg \to WH \to b\bar{b}l\nu_l$ and $gg \to ZH \to b\bar{b}l\bar{l}$ final states have been considered. The Feynman diagram of these processes are shown in fig. 4.1. Note that due to their weakly-interacting nature, neutrinos cannot be reconstructed in the detector. For this reason, former will be referred to as semi-leptonic and the latter as di-leptonic final state.

Figure 4.1: The analysed final states of the VH processes.

The events are categorised through a triggering system and are assigned to four channels. The list of the triggers for the di- and semileptonic channels are listed in tab. 4.1.

Channel	Trigger Name
ee	`Ele23_Ele12_CaloIdL_TrackIdL_IsoVL` `Ele23_Ele12_CaloIdL_TrackIdL_IsoVL_DZ`
e	`Ele32_WPTight_Gsf` `Ele32_WPTight_Gsf_L1DoubleEG`
$\mu\mu$	`Mu17_TrkIsoVVL_Mu8_TrkIsoVVL_DZ` `Mu17_TrkIsoVVL_Mu8_TrkIsoVVL_DZ_Mass3p8`
μ	`IsoMu24_eta2p1` `IsoMu27`

Table 4.1: The lepton triggers used in the analysis.

The naming in 4.1 indicates the phase space which the triggers cut off. In the di-leptonic channels the lepton with the higher energy (the leading lepton) is required to have $p_T > 23\,\text{GeV}$ and $p_T > 17\,\text{GeV}$ for ee and $\mu\mu$, respectively; the sub-leading (the "other") lepton is then required to have $p_T > 12\,\text{GeV}$ and $8\,\text{GeV}$ for ee and $\mu\mu$, respectively. In the semi-leptonic categories, only electrons with $p_T > 32\,\text{GeV}$ and isolated muons with $24\,\text{GeV}$ and $|\eta| < 2$ and isolated muons with $p_T > 27\,\text{GeV}$ are kept.
In addition to these selection criteria, both the muon tracks have to be isolated in the $\mu\mu$ channel. For the e channel, the tight working point has been selected as trigger efficiency.
The datasets used for the analysis have been simulated with `PYTHIA` [43] and `HERWIG++` [44]. These simulated events not only contain matrix element calcualtion samples, but also the CMS detectors simulations using `GEANT4` in order to represent the final observed distribtuions accurately [45, 46, 47]. The list of the signal sample datsets is shown in tab. 4.2. These processes are `ggZH` (for the box diagram process), `ZH_DY` (for the Drell-Yan-like ZH production process) and `WH` (for the WH Higgsstrahlung process).
For the background processes, one distinguishes between the following processes:

Dataset Name	N	σ/pb
`Dataset1`		

Table 4.2: The signal datasets

`dy`: Drell-Yan-like vector boson production with hadronic initial state radiation. This process has high cross section and yields the same final state (for radiated bottom jets) as the signal. In addition, it also has the same lepton signature. The process is shown in fig. 4.2.

Figure 4.2: The Feynman diagram

`t$\bar{t}$`: Production of top quark pairs $t\bar{t}$. These quarks decay via $t \to Wb$ and with the leptonic decay channel of the W, this has the same final state as the signal process. This has the second largest background contribution. The corresponding Feynman diagram is shown in fig. 4.3.

Figure 4.3: The Feynman diagram

`t$\bar{t}$V`: Associated vector boson production in the initial or final states with a $t\bar{t}$ pair. For the top quarks decaying into bottom quarks and the leptonically decaying vector bosons, this yields a final state similar to the signal final state.

`st`: Single top quark production. A hadronically decaying top with a QCD-induced bottom jet results in a semileptonic di-bottom state.

`VV`, `VVV` and `t$\bar{t}$VV`: di- or triboson production with (or without) associated $t\bar{t}$ production. If any of these vector bosons decay hadronically while the other(s) decay leptonically, this results in a semi- or dileptonic di-bottom final state.

`t$\bar{t}$VH`, `ggH` and `VBFH`: Higgs production channels.

`wjets`: leptonically decaying jet-associated W boson production.

`QCD`: reducible QCD background.

`rare`: rare processes such as four top production.

The list of the background samples from the MC simulations are all listed in tab. 4.3.

Sample Name	N	σ/pb
`Dataset2`		

Table 4.3: List of simulated background samples.

4.2 Analysis Setup

In the following the selected electron, muon and jet final state objects (reconstructed and identified by ParticleFlow) will be described in detail. As a brief overview, the analysis setup has been summarized in fig. X.Y

The muon selection criteria are listed in tab. 4.4. For the leading muons, a p_T-cut of 25 GeV has been required; qualifying subleading muons need to satisfy $p_T > 15\,\text{GeV}$. Furhtermore, muons ought to be identified as tight muons and have to be contained within the tracker, while their impact parameter (IP) geometry being also compatible with the primary vertex (PV). In addition, the IP significance `sip3d` (defined as the ratio between the uncertainty of the IP and the IP itself) is also restricted to guarantee that these muons originate from the PV. The `pfRelISO04_all` parameter cut ensures the isolation of the muons and hence their origin from the PV.

`tightId`
$p_T > 25\,\text{GeV}/15\,\text{GeV}$
$\lvert\eta\rvert \leq 2.4$
$d_{xy} \leq 0.05\,\text{cm}$
$d_z \leq 0.1\,\text{cm}$
`pfRelIso04_all` ≤ 0.15
`sip3d` ≤ 8

Table 4.4: Selection criteria for the muons. The first and the second values correspond to the cuts applied to the leading and sub-leading muons, respectively.

The electron selection criteria is listed in tab. 4.5. The electrons are required to be included in the tracker (through η) to ensure quality transversal momentum reconstruction. These electrons should also originate from the primary vertex through the impact parameter (IP) cuts d_i. Similarly to muons, the IP significance and isolation measures are also restricted. For quality reconstruction, the maximum number of lost hits in the tracker is also limited. As identification efficiency, the 80% working point (WP80, defined on a Drell-Yang sample) has been chosen.

`MVA WP80`
$p_T > 25\,\text{GeV}/15\,\text{GeV}$
$\lvert\eta\rvert \leq 2.5$
$d_{xy} \leq 0.05\,\text{cm}$
$d_z \leq 0.1\,\text{cm}$
`pfRelIso04_all` ≤ 0.15
sip3d ≤ 8
lost hits ≤ 1

Table 4.5: Selection criteria for the electrons.

For the jets in consideration, the selection criteria in tab. 4.6 are applied. For AK8 jets, the transversal momentum selection lies higher than those of the AK4 jets; both jets are required to carry the tight ID. In addition to that, they should be restricted to the region covered by the tracker. Based on the jet selection, the flavour tacking algorithm identifies the bottom jets. For the AK4-tagging, DeepFlavour and for AK8 jets, DeepDoubleX are used.

`tightId`
$p_T > 30\,\text{GeV}/70\,\text{GeV}$
$\lvert\eta\rvert < 2.4$

Table 4.6: Jet selection criteria. For the transversal cuts the first and the second values are applied to the AK4 and AK8 jets, respectively.

4.3 Event Categorisation

For better signal yield the selected events have been categorized depending on their final state configuration. In the network analysis, only channels with two b tagged AK4 jets with semi- or dileptonic final states have been chosen.
In the dileptonic configurations, the lepton charges have been required to be opposite, such that the total charge of selected objects is $Q_{tot} = 0$. In addition to that, the invariant mass defined by the dilepton system is required to be close to the Z mass and only events with maximum absolute deviation of 15 GeV are kept. For the semileptonic channels, such a selection is not necessary. Additional selection is performed with supplementary leading lepton transversal momentum p_T^l and missing trasverse energy $\not{E}_T$ cuts.

channel	$\mu\mu$	ee	μ	e
p_T^l	$> 30\,\text{GeV}$	$> 25\,\text{GeV}$	$> 25\,\text{GeV}$	$> 35\,\text{GeV}$
n_μ	2	0	1	0
n_e	0	2	0	1
Q_{tot}	0	0	–	–
$\not{E}_T$	–	–	$> 30\,\text{GeV}$	
m_{ll}	$\lvert m_{ll} - m_Z\rvert \leq 15\,\text{GeV}$		–	–

Table 4.7: Selected categories with the additional kinematic cuts they impose. Note that there is a channel for each trigger.

4.4 Multi-Process Classification and Weight Application

Following the definition of the channels, the contained final state objects are furthermore categorized using a DNN. The network assigns these final state objects to so-called subcategories for general channels. Hence, the input variables include properties of additional physics objects, such as AK8 jets. The physics objects kinematic variables, correlations among them and additional jet and particle identification values. These complete list of input variables is shown in tab. 4.8.
The DNN then assigns a probability (following sigmoid activation) to each final state object belonging to a process group. As the sensitivity for some rare background processes is low, these are grouped into a single process group; hence, for the 16 processes the analysis differentiates between there are only 13 process groups the DNN sorts the physics final state objects into. The 13 process groups are listed in tab. 4.9.
The events carry a generator weight, as it would be computationally infeasible to simulate all processes proportionally to their occurrence; rather they are scaled with the experiment luminosity and process cross section. At the same time, it is desirable to reduce the uncertainties due to the finite size of Monte Carlo samples. The resulting weights written as

Object	Variable
Leptons (2)	E, p_x, p_y, p_z, q, PDG ID
AK4 jets (4)	E, p_x, p_y, p_z, b-tag value
AK8 jets (2)	E, p_x, p_y, p_z, b-tag value
Correlated objects	$m_{ll}, \Delta\phi_{jj}, \Delta R_{ll}$
	$m_{jj}, p_{T,jj}, \Delta\phi_{jj}, \Delta R_{jj}$
	$\min(\Delta R_{l1j1}, \Delta R_{l1j2})$
	$\min(\Delta R_{l2j1}, \Delta R_{l2j2}$
	$m_{lljj}, N_{AK4j}, p_T^{miss}$

Table 4.8: The 54 inputs of the classifier DNN with object multiplicity indicated in brackets. Note that the network has been trained as a general final state classifier, so additional kinematic variables irrelevant for the dilepton and single lepton channel analysis are also used as inputs.

$$w_i = \frac{\mathcal{L}\sigma w_i^{\text{gen}}}{\sum_i w_i^{\text{gen}}}$$

with luminosity $\mathcal{L}$ and cross section σ and are applied to the DNN scores after classification for each process i.

Process group	Included Processes
ggZH	ggZH
ZH_DY	ZH_DY
WH	WH
dy	dy
tt	$t\bar{t}$
st	st
ggH	ggH
VBFH	VBFH
wjets	wjets
QCD	QCD
merged_ttVX	$t\bar{t}V, t\bar{t}VV, t\bar{t}VH$
merged_multiboson	VV, VVV

Table 4.9: The output nodes of the DNN corresponding to a single process group each. Note that some process groups contain multiple rare processes which are difficult to resolve.

An example output of a such classification is shown for an *ee* category and ZH_DY process group in fig. 4.4. Note that there are $13 \times 4 = 52$ plots for the 13 process groups and 4 categories in total. On this particular plot, the strong Drell-Yan and $t\bar{t}$ background is clearly visible, while a large portion of ZH_DY signal events are classified correctly.

Unfortunately, no single process can be seen in a measured dataset. What is possible though, is to fit the data to the distribution of the DNN scores of every category. A such histogram for a single category is shown in fig. 4.5; for all four categories covered by this study, the resulting histogram in shown in fig. 4.6. In case of a measured dataset, only the resulting bin heights would be visible on which an analytical likelihood fit can be performed extracting the contribution of each process to the bins.

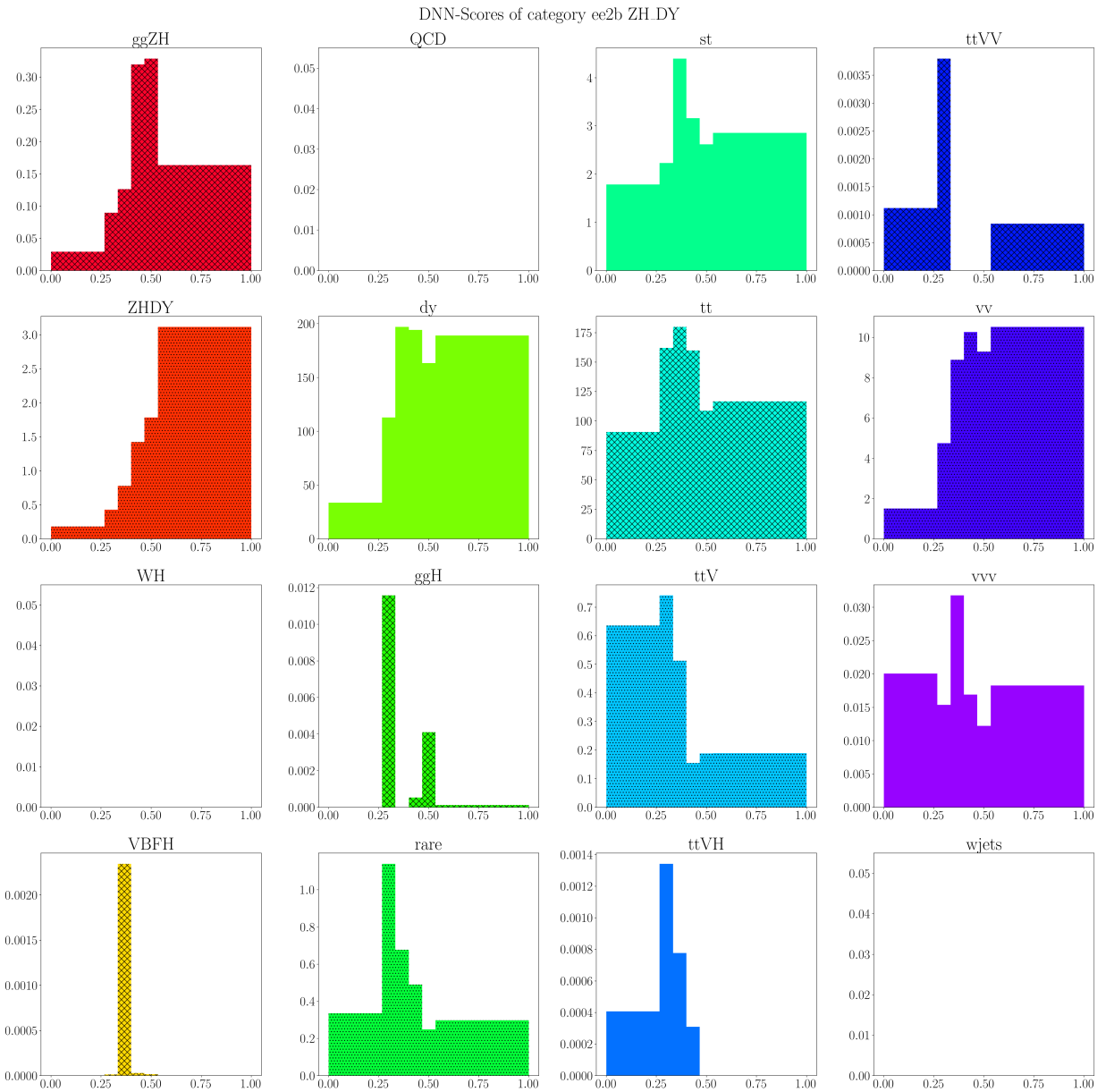

Figure 4.4: The DNN score distribution for the categorised final state objects for the `ZH_DY` subcategory. Note the the different norming of the y axis and the high share of $t\bar{t}$, `dy` and `ZH_DY` processes. Due to the effective categorization none of the `WH`, `QCD` and `wjets` events landed in this category. This is one of the total 52 subcategories.

4.5 Likelihood Fits

Using these histograms, the maximum likelihood fits can be performed to obtain the signal strength parameters. For that, the likelihood function for a signal+background hypothesis defined as

$$\mathcal{L}(N|\{\mu_i\},\{\theta_i\}) = \prod_i \frac{(\mu_i s_i(\theta) + b_i(\theta))^N}{N!} e^{-\mu_i s_i(\theta)+b_i(\theta)} \prod_j p(\tilde{\theta}_j|\theta_j)$$

with Poisson-distributed binwise signal and background contributions s_i and b_i depending on

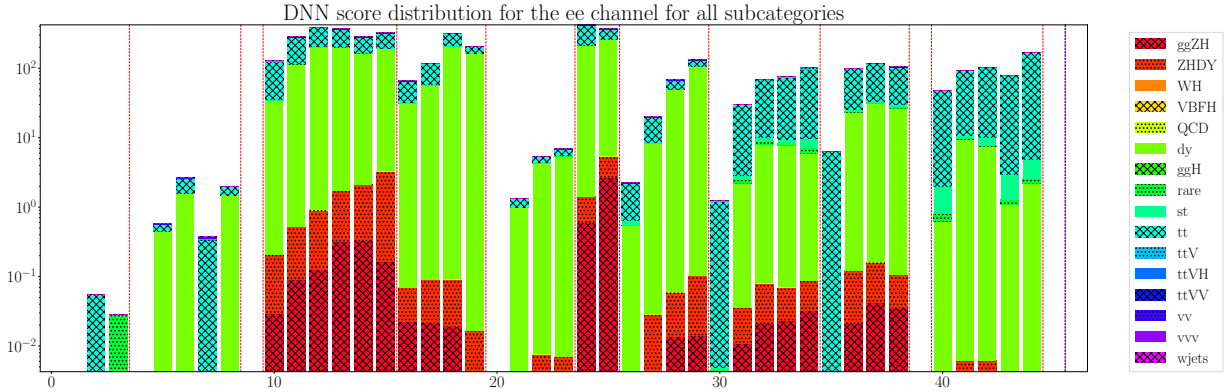

Figure 4.5: The DNN score histogram distribution for the *ee* category and each 13 subcategory (separated in red). On the x-axis, only the bin index is shown. The fourth subcategory from the left corresponds to the above-mentioned `ZH_DY` subcategory. This is one of the four categories.

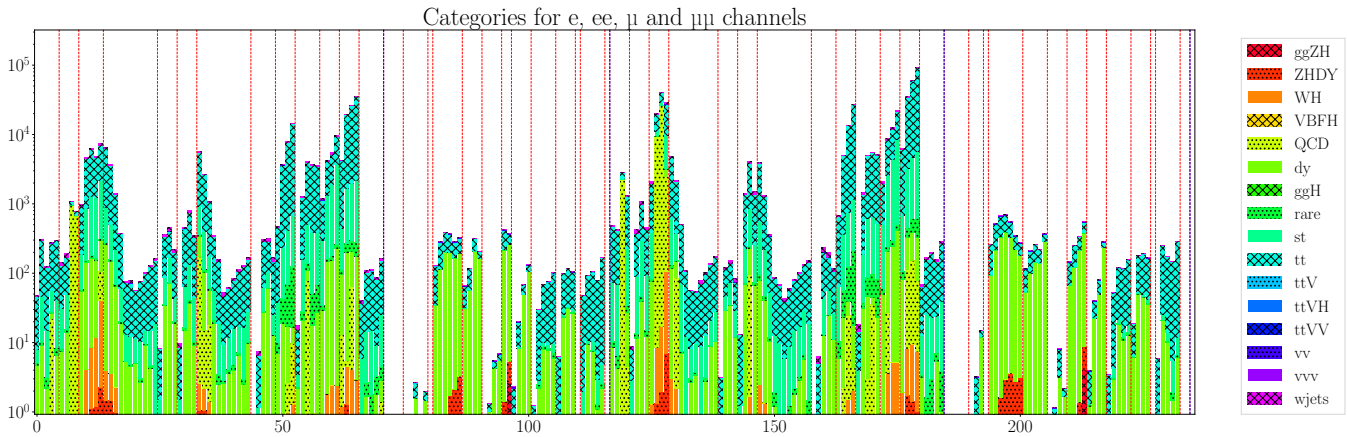

Figure 4.6: The distribution of the DNN scores for each subcategory (separated in red) in the semi- and dileptonic categories in consideration (separated in blue). Note the high share of Drell-Yan, $t\bar{t}$ and QCD background events in each bin and the absence of WH events in the dileptonic bins. The *ee* category is the second one from the left.

the uncertainty-modelling nuissance parameter θ following the probability distribution $p(\tilde{\theta}|\theta)$. Note that this fit assumes only changes in the background processes within the uncertainties, but not in their proportion. In other words, there is a single signal strength modifier parameter for each signal process exclusively.
In this thesis, these signal strength modifier parameters will inferred using a conditional invertible neural network and will be compared to their maximum likelihood counterpart.

5 Network Setup

Let's briefly summarize the content of the previous chapters. In chapter 1 the Standard Model of Particle Phyiscs has been presented. It is a remarkable theory explaining the dynamics of elementary particles; however, due to its incompleteness, precision measurements are necessary. The extraction of the $ggZH$ process poses as a suitable candidate for such measurements at the CMS experiment, described in chapter 2. In order to achieve that, new analysis techniques based on deep learning (chapter 3) are being explored. The exact analysis strategy and -setup has been described in chapter 4.
In this chapter the cINN setup which will be used for inferring the signal strength modifiers μ_i will be discussed. In the first half, the generation of the training and test datasets with a detailed focus on the analysis specifics, such as interpolation between histograms (morphing) and the representation of the statistic and systematic uncertainties is going to be elaborated.
In the second half of this chapter, the final cINN architecture is going to be presented. The inference results and the interpretation of the network outputs will be discussed in the next chapter.

5.1 Training Dataset Generation

The conditions shown in fig 4.6 in chapter 4 represent the expectation on the measured dataset. Unfortunately, the exact share of each signal process in each bin in the measured data is a priori unknown (which would mean the complete physics is known to the very last detail). For this purpose the different scenarios of a signal process having more or less shares in a bin have to be first artificially represented in the dataset in order to obtain a cINN which is capable of inferring the share of each signal process – or in other words, to obtain the signal modifier parameters μ_i. For this, both signal and background processes have to be scaled to achieve the desired sensitivity towards different scenarios.

5.1.1 Prior Selection

In order to do so, a set of signal modifier parameters are drawn from a pool of expected parameter distribution (the prior). Then, each process in the histogram is scaled with the corresponding parameter and the bins will summed up meaning (similar to the expected dataset) the information about the exact contribution of each process to each bin is lost thereafter.
The prior distribution for the processes have been selected as the following. For the signal processes `ggZH`, `ZHDY` and `WH` the μ_i were sampled from a gamma distribution

$$\Gamma(x; k, \theta) = \frac{x^{k-1} e^{-x/\theta}}{\theta^k \Gamma(k)}$$

with shape parameter k, scale θ and $\Gamma(k)$ being the gamma function. An advantage of this distribution as prior is the positiveness of the samples $\{x_i\}$ from the distribution as negative signal modifier parameters carry no physical meaning; apart from that, the distribution has a long tail for increasing values of x which allow a more fined selection in a given range while maintaining some generalization property. The parameters were set to be $k = 1.5$ and $\theta = 7$; with this specific selection, one obtains a prior which is sensitive in the low signal yield region $\mu \lesssim 10$ while maintaining network sensitivity for values $\mu \lesssim 100$.

For the background processes, it is expected that the MC simulations follow the expectation with good accuracy. Hence, one expects $\mu \approx 1$. For this reason, the prior has been chosen to follow a lognormal distribution

$$\mathcal{N}(x;\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma x} \exp\left(-\frac{(\ln x - \mu)^2}{2\sigma^2}\right)$$

with mean $\mu = 0.05$ and $\sigma = 0.25$, lognormal distributions being positive themselves.

In order the estimate the arising uncertainty from the luminosity, the obtained conditions (after scaling and uncertainty variation) are then scaled with an additional luminosity factor. These have been drawn from a lognormal distribution with $\mu = 0$ and $\sigma = 0.02$. All distributions are shown in fig. 5.1.

With these priors, the cINN has a total of 17 inputs for 3 signal, 13 background modifier parameters and the luminosity nuisance parameter.

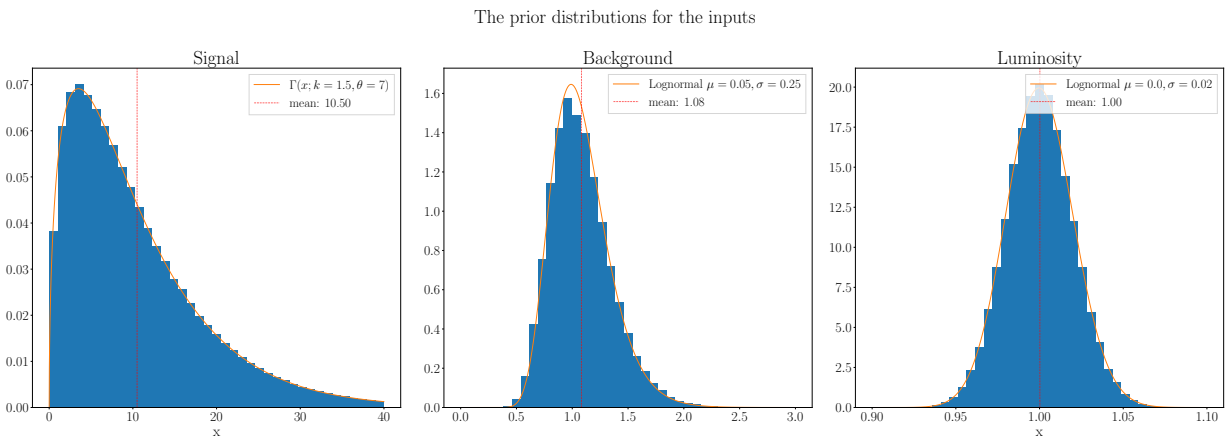

Figure 5.1: Prior distributions for the signal and background processes. Samples are shown in blue; the mean of each distribution is shown with the red dashed line. The analytic shape of each distribution is shown in orange.

It does not suffice to scale the processes with their respective parameters exclusively, as systematic and statistic effects change the overall shape of the resulting histogram. For this reason, the nuisance parameter effects have to be mapped onto the training and test dataset in order to avoid that the network learns features not present in a general measured dataset.

5.1.2 Systematic Uncertainties

The systematic uncertainties require an interpolation method as the shifts in the histograms are computationally expensive to obtain. In other words, as only the effect of the 1σ up- and down-shifts of the uncertainties (templates) are given, the task is to represent the in-between-cases (and obviously, those on the outside of the given range) in the histrograms. For this purpose, a non-linear histogram morphing method has been implemented.

In the following, an overview on the treated uncertainties and the morphing algorithm will be given with a discussion on the implementation and on its effects on the histograms following thereafter.

5.1.3 Normalization and Shape-Changing Uncertainties

There are 25 nuisance parameter which have been taken into account. These can be grouped into normalization and shape-changing uncertainties.

Normalization uncertainties do not affect the shape of the histograms. Rather, they have an equal effect on each bin (hence the name "normalization"). To this group belong the following: the luminosity uncertainty and the parton density function uncertainity (which arise from the unknown interaction initial state due to the nontrivial proton structure).

The shape-changing uncertainties influence all histogram bins differently and incorporate most uncertainties in this work. The 23 shape-changing and the parton density function normalization uncertainties have been listed in tab. 5.1; interpolation among these systematic effects given the 1σ shifted histograms have been performed via histogram morphing, which will be discussed in the next section. In order to study the network's nuisance parameter expressive power, the luminosity uncertainty has been treated as an input.

Uncertainty name	Description
`PDFSet`	Parton density function uncertainty
`PSWeight_FSR` `PSWeight_ISR`	Parton shower mismodelling uncertainties for final and initial state radiation

`ScaleWeight_Envelope`
`ScaleWeight_Fact`
`ScaleWeight_Mixed`
`ScaleWeight_Renorm`
	Jet energy scale uncertainties

`btagWeight_lf`
`btagWeight_cferr1`
`btagWeight_cferr2`
`btagWeight_hf`
`btagWeight_hfstats1`
`btagWeight_hfstats2`
`btagWeight_lfstats1`
`btagWeight_lfstats2`
	b-tag score weight correction uncertainties

`l1_ecal_prefiring`	2017 ECAL miss-calibration correction uncertainty

`electron_reco`
`electron_id`
`muon_id`
`muon_iso`
	Lepton identification, reconstruction and isolation uncertainties

`trigger_ee_sf`
`trigger_mu_sf`
`trigger_mumu_sf`
	Trigger efficiency uncertainties

`pileup`	Pileup uncertainty

Table 5.1: The morphed one normalization (`PDFSet`) and the other 23 shape-changing uncertainties with a brief description.

5.1.4 Non-Linear Histogram Morphing

Non-linear histogram morphing has been implemented on the basis of [48]. For the interpolation a nominal distribution and – in this case – 24 other distribution templates representing the 1σ and -1σ shifts are given. The nominal template $f(x|m)$ with morphing parameter m can then be approximated around $m = 0$ with the other templates in a simple 1D case using

$$f(x|m) \approx \sum_j \frac{\partial^{(j)} f(x|m)}{\partial m^{(j)}}\bigg|_{m=0} \frac{m^j}{j!} = \sum_j f'_j(x|m=0)m^j$$

where the derivatives and the factorials have been absorbed in f'_j.
For T uncorrelated uncertainties (hence for T morphing parameters m) with morphing parameter values ± 1 there are $2T + 1$ sampled distributions in total (counting the nominal distribution, defined as $m = 0$). An intuitive representation of the "morphing parameter space" with the indication of the available templates is shown in fig. 5.2.

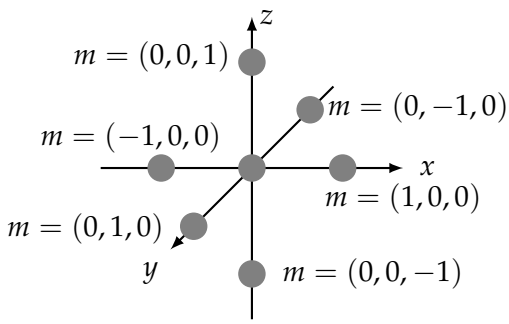

Figure 5.2: The morphing "phase space" showing the available distributions for $T = 3$ uncertainties of $\sigma = \pm 1$. The available templates in the phase space are shown in grey. Note that the nominal template with $m = (0, 0, 0)$ lies in the origin and there are no "mixed" templates (such as $m = (1, 1, 0)$) available. In the setting described above, one has $T = 24$, hence there are $24 \times 2 + 1 = 49$ templates in total.

For this multidimensional case where there are no "mixed templates" as the above uncertainties are considered to be uncorrelated. Hence, the mixed derivative terms from the above expansion are zero and the expression can be written for a given $\mathbf{m}_i$[5] as:

$$f(x|\mathbf{m}_i) = \underbrace{f(x|0)}_{f_0} + \sum_{j=1}^{T} \underbrace{\frac{\partial f(x|\mathbf{m})}{\partial m_j}\bigg|_{\mathbf{m}=0}}_{f'_j} (m_i)_j + \sum_{j=1}^{T} \underbrace{\frac{1}{2!}\frac{\partial^2 f(x|\mathbf{m})}{\partial m_j^2}\bigg|_{\mathbf{m}=0}}_{f'_{jj}} (m_i)_j^2 + \mathcal{O}\left((m_i)_j^3\right) \tag{5.1}$$

where $(m_i)_j$ denotes the j-th component of the i-th morphing parameter vector. The aim is to find an expression for the unknown derivatives as only the histogram $f(x|\mathbf{m})$ (but not its derivative) is given. If one knew the derivatives, one could easily select a general $\mathbf{m}_i$ and insert in into eq. 5.1, obtaining the morphed distribution.
In order to express the derivatives write all given templates in a vectorized form, and make use of the matrix form

[5] $\mathbf{m}_i = (0, 0, ..., 0, \pm 1, 0, ..., 0)$ for the i-th template with $\sigma = \pm 1$

$$\begin{pmatrix} f\Big(x|(0, & 0,0,..., & 0)\Big) \\ f\Big(x|(0, & 1,0,..., & 0)\Big) \\ f\Big(x|(0, & -1,0,..., & 0)\Big) \\ & \vdots & \\ f\Big(x|(0, & 0,0,..., & 1\Big) \\ f\Big(x|(0, & 0,0,..., & -1\Big) \end{pmatrix} = \underbrace{\begin{pmatrix} 1 & 0 & 0 & 0 & & 0 & 0 \\ 1 & 1 & 1 & 0 & \cdots & 0 & 0 \\ 1 & -1 & 1 & 0 & & 0 & 0 \\ & & \vdots & & \ddots & \vdots & \\ 1 & 0 & 0 & 0 & & 1 & 1 \\ 1 & 0 & 0 & 0 & \cdots & -1 & 1 \end{pmatrix}}_{M} \begin{pmatrix} f_0 \\ f'_1 \\ f'_{11} \\ \vdots \\ f'_T \\ f'_{TT} \end{pmatrix}$$

With an invertible coefficient matrix $M \in \mathbb{R}^{49\times 49}$, encoding the position of the templates in the morphing phase space. Inverting M[6]

$$M^{-1} = \begin{pmatrix} 1 & 0 & 0 & 0 & & 0 & 0 \\ 0 & 0.5 & -0.5 & 0 & \cdots & 0 & 0 \\ -1 & 0.5 & 0.5 & 0 & & 0 & 0 \\ & \vdots & & & \ddots & & \vdots \\ 0 & 0 & 0 & 0 & & 0.5 & -0.5 \\ -1 & 0 & 0 & 0 & \cdots & 0.5 & 0.5 \end{pmatrix}$$

this collapses into

$$M^{-1}\mathbf{f} = \mathbf{f}'$$

and eq. 5.1 can be then written for a general $\mathbf{m}$ as

$$f(x|\mathbf{m}) = (1, m_1, m_1^2, ..., m_T, m_T^2)M^{-1}\mathbf{f} \tag{5.2}$$

Eq. 5.2 is the final shape of the morphing algorithm implemented for the representation of the effect of the 24 nuisance parameters in the dataset. The results of applying morphing for some selected nuisance parameters are discussed in the next chapter.

5.1.5 Systematic Uncertainty Representation in Training Data

The effect of the morphing algorithm from above has been studied by looking at some arbitrary histograms explicitly. Below an example for one normalisation and two shape-changing uncertainties for both morphing parameters $m = -0.5$ and $m = 0.5$ is given.
For the parton desitity function uncertainty, negative values of m decrease the event content of each bin, while $m > 0$ yields more events compared to the nominal histogram. As a normalisation uncertainty, it keeps the shape of the distribution. This behaviour can be seen in fig. 5.3.
As a shape-changing uncertainty, `pileup` influences all bins separately. Due to this behaviour, the content of a given bin for a process might increase in one bin but decrease in another bin for the same morphing parameter. The morphed histogram for the same morphing parameters are shown in fig. 5.4.

[6]Note that thanks to the block matrix structure the inversion is computationally feasible. In addition, the inverse has to be evaluated only once.

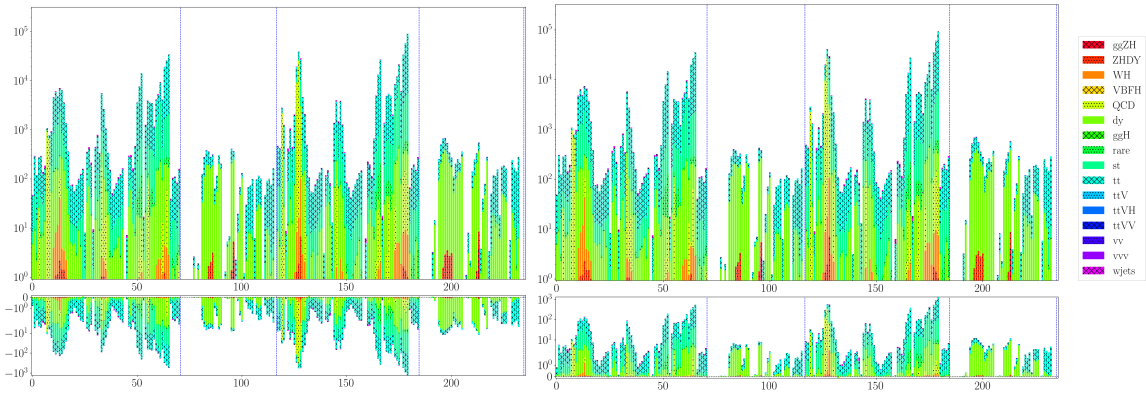

Figure 5.3: Morphed histograms for $m = -0.5$ (left) and $m = +0.5$ (right) for `PDFSet` for the e, ee, μ and $\mu\mu$ categories separated with the blue dashed line. Below the difference of the morphed histogram to the nominal histogram ($m = 0$) is shown. Note that each bin has been influenced uniformly, keeping the overall shape of the conditions.

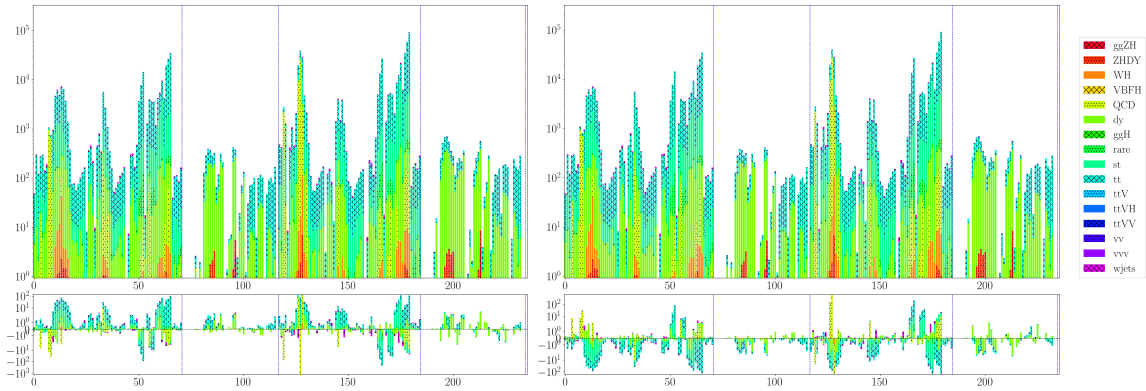

Figure 5.4: Morphed histograms for $m = -0.5$ (left) and $m = +0.5$ (right) for `pileup`. Below the difference of the morphed histogram to the nominal histogram is shown. Note that effect is different for each process and in each bin as expected for a shape changing uncertainty.

As the second shape-chaning uncertainty to showcase, the dilepton trigger efficiency effects have been chosen. Due to their focus on given $\mu\mu$ or ee final states, these uncertainties influence single categories only. The effect of `trigger_mumu_sf` and `trigger_ee_sf` are shown in fig. 5.5.

5.2 Statistical Uncertainties

There are two sources of statistical uncertainties on Monte Carlo level. First, there is a Poisson fluctuation due to the generator producing a finite amount of samples for each process. As the number of samples is large, one can assume that these fluctuations follow a normal distribution with $\mu = N$ and $\sigma = \sqrt{N}$. Second, it is expected that the data in the DNN-score histograms follow a Poisson distribution themselves.

The generator-related statistical effects have been applied following morphing, after which eventually arising negative values (due to the variation being performed through a normal distribution) have been clamped to zero. Then, the signal strength modifiers drawn from the prior distributions have been applied to each process and the resulting bins have been summed up, losing

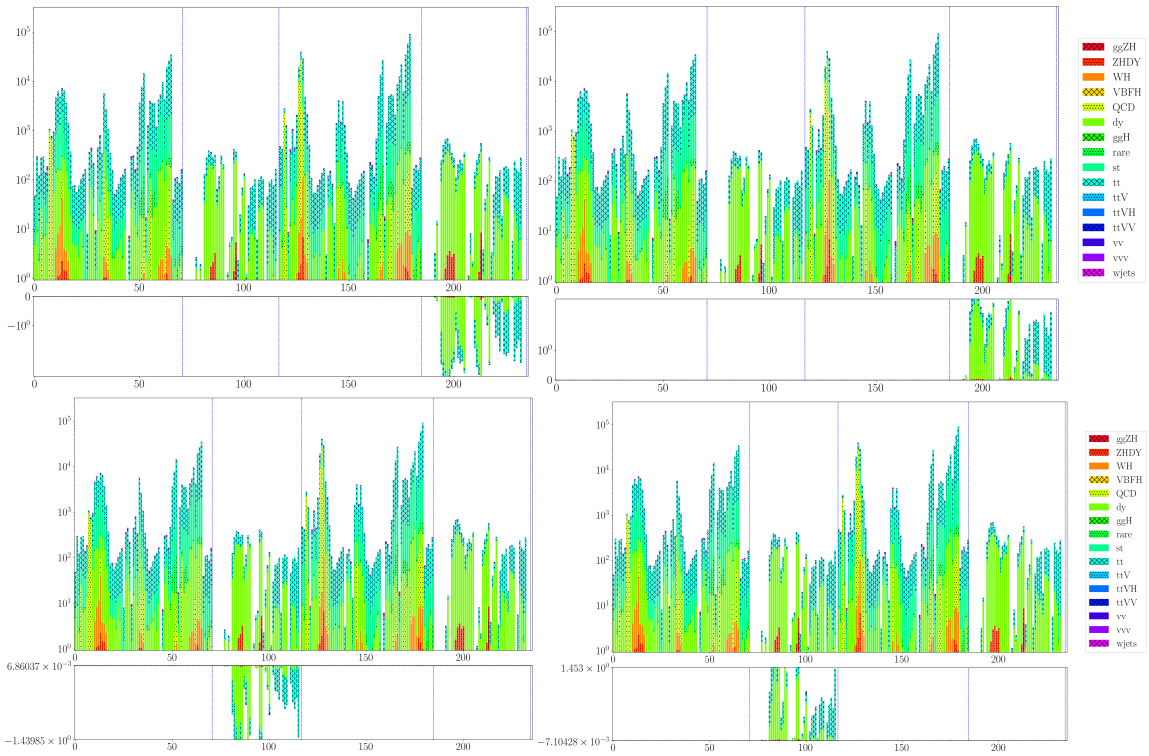

Figure 5.5: Morphed histograms for $m = -0.5$ (left) and $m = +0.5$ (right) for `trigger_mumu_sf` (above) and `trigger_ee_sf` (below). The four categories from left to right are: e, ee, μ and $\mu\mu$. Note the exclusive action of the efficiency uncertainties on a single category.

information on the exact bin content (just like in the case of real data). Last, the resulting histogram bin heights have been independently varied following a Poisson distribution to model statistical effects in the data and the application of the normalising luminosity uncertainty has been performed. The resulting histograms have been saved and used as training or test dataset. graph showing when to apply what effect?

5.3 Network Architecture

In the following, the cINN network architectures for the inference will be elaborated.
The network inputs are three signal and 13 background modifier parameter for each process and an additional normalizing nuisance parameter for the luminosity uncertainties. In total, the network has 17 inputs.
Regarding the conditions, it had a dimension of 235 for the four considered categories (e, ee, μ and $\mu\mu$) and all of their subcategories.
The implementation has been done in `Python` using `PyTorch` [49]; the network has been constructed using the "Framework for Easily Invertible Architectures" (`FrEIA`) framework [50]. The cINN uses 12 GLOW blocks. After each block, an additional permutation layer is introduced. These layers perform arbitrary (but fixed) permutations among the outputs. Note that this transformation has per construction a determinant of ± 1 making no change in the volume for the resulting normalizing flow. On the other hand, it does help removing any correlations between neighbouring input values, making training more performant.

The GLOW networks contain subnetworks with 128 Nodes, of 4 layers. The activation function for each layer was chosen to be ReLU.
For the summary network studies, a shallow network has been constructed, as more complex setups tended to overtrain in the considered cases. The network had a layer of 300 nodes between the DNN scores and the condition node; the output dimensionality has been chosen to be 100.
Both networks have been trained for 11000 epochs on the same dataset containing 1.5 million generated samples. For training, a batchsize of 500 has been chosen. For the test set, 150 000 samples have been generated.
The network inputs have been normalized to mean 0 and standard deviation 1 via

$$\mu_i^N = \frac{\mu_i - \langle \mu_i \rangle}{\sigma_{\mu_i}}$$

For the conditions, as they stretch over 5 orders of magnitude, their logarithm has been normalized:

$$y^N = \frac{\log(y+1) - \langle \log(y+1) \rangle}{\sigma_{\log(y+1)}}$$

As a learning rate scheduler, an empirically well-functioning cosine scheduler has been used. This scheduler starts off with a learning rate of $\eta = 10^{-3}$ and reduces it to $\eta = 10^{-5}$ in steps in each epoch. The evolution of the learning rate is shown in fig. 5.6. As an optimizer, Adam with parameters $\beta_1 = 0.9$ and $\beta_2 = 0.999$ has been chosen. In order to penalize large weights, a normalizing parameter of $\alpha = 10^{-5}$ has been chosen; for the clamping parameters, a value of 2 has been selected.
All tests have been performed three times (3 runs) with differently initialized models and the model with the lowest test loss has been studied further.

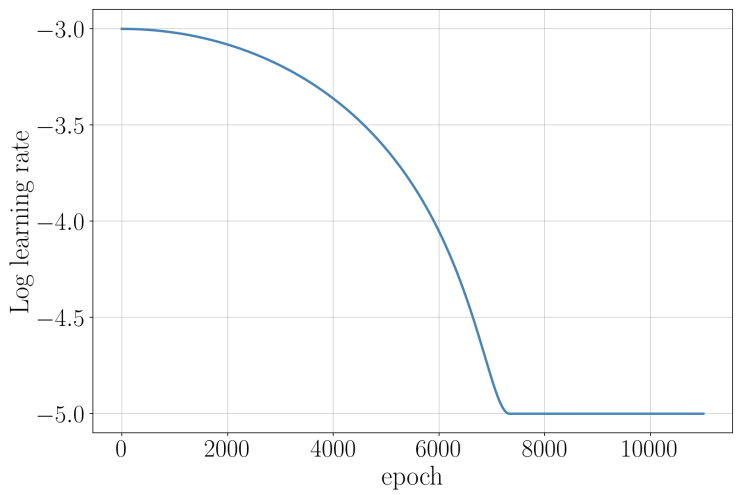

Figure 5.6: The evolution of the learning rate with each epoch on a logarithmic scale.

6 Signal Modifier Parameter Inference with cINNs

In this chapter, the performance of the network will be characterized in detail. Two network setups will be considered: one without and one with a summary network reducing the representation of the conditions. These networks will be referred to cINN and SN-cINN (for summary network-extended cINN). It will be shown that the setup without a summary network results in superior performance for which an explanation will be given.

6.1 Training Performance

The evolution of the loss functions are shown in fig. 6.1. The location of the models with the lowest validation loss is shown with the red dashed line; the corresponding loss values are -5.16 for the cINN and -4.96 for the SN-cINN. It becomes clear at first glance the SN-cINN models suffer from overtraining – no network from the three independent runs with different initializations have converged during training. This effect has also been observed in [36]. The structure of the summary network has been chosen in a way that this network had the lowest validation loss; models with a deeper summary network or less output nodes tend to produce worse results with a higher loss. Fortunately, the predictions are of the same quality if the summary network architecture is suitably chosen.

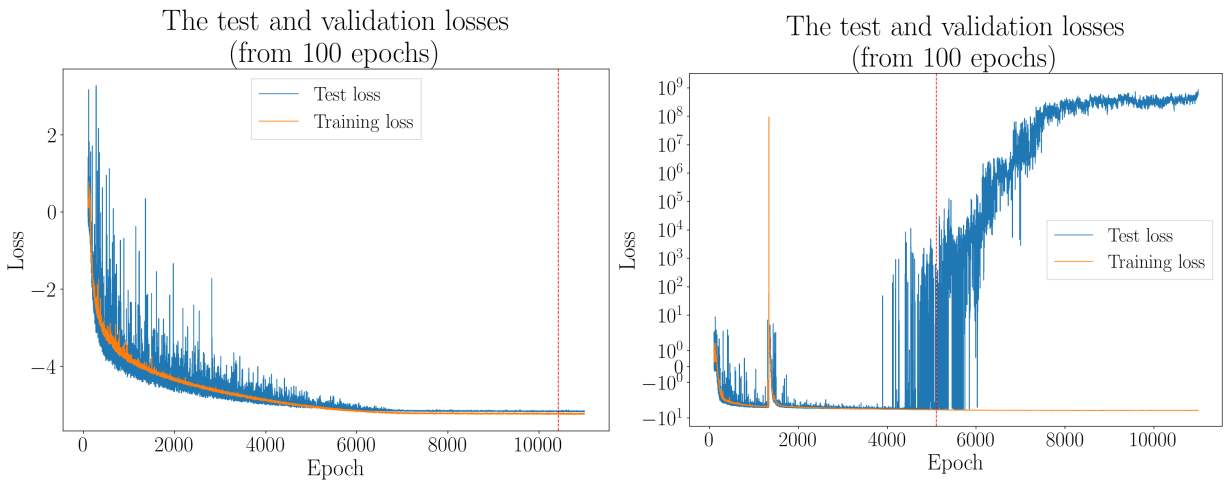

Figure 6.1: The loss values for each network from the 100th epoch (initial losses cut off as they disturb the graph) for the cINN and for the SN-cINN. Models with the lowest validation loss are shown with the red dashed line. Note the overtraining for the summary network (right)

From the losses it becomes clear that the condition dimensionality cannot be significantly reduced as each input bin carries the same value of information as these are uncorrelated among themselves. For this reason, the reduction of the conditions necessarily result in performances of lower quality.

6.2 Latent Space Distribution

At the end of chapter 3, the loss function has been constructed in a way that the input variables should be mapped to a multivariate normal distribution with $\mu = 0$ and $\Sigma = 1$. Note that thanks to the Universal Approximation Theorem, the diffeomorphic mappings in the normalizing flow represented by the GLOW coupling blocks can be approximated arbitrary well

for a well-constructed network architecture and well-chosen optimization procedure. Since loss functions per se do not carry any information about the quality of the resulting mapping, the shape of the network output – the latent space distribution – serves as a suitable candidate to check whether there are any issues with the setup.
The latent space distribution for both networks are shown if fig. 6.2 for the models with the lowest test loss. The distribution of the datapoints are shown in the blue histograms while a normal function with mean 0 and standard deviaiton 1 is shown in orange.

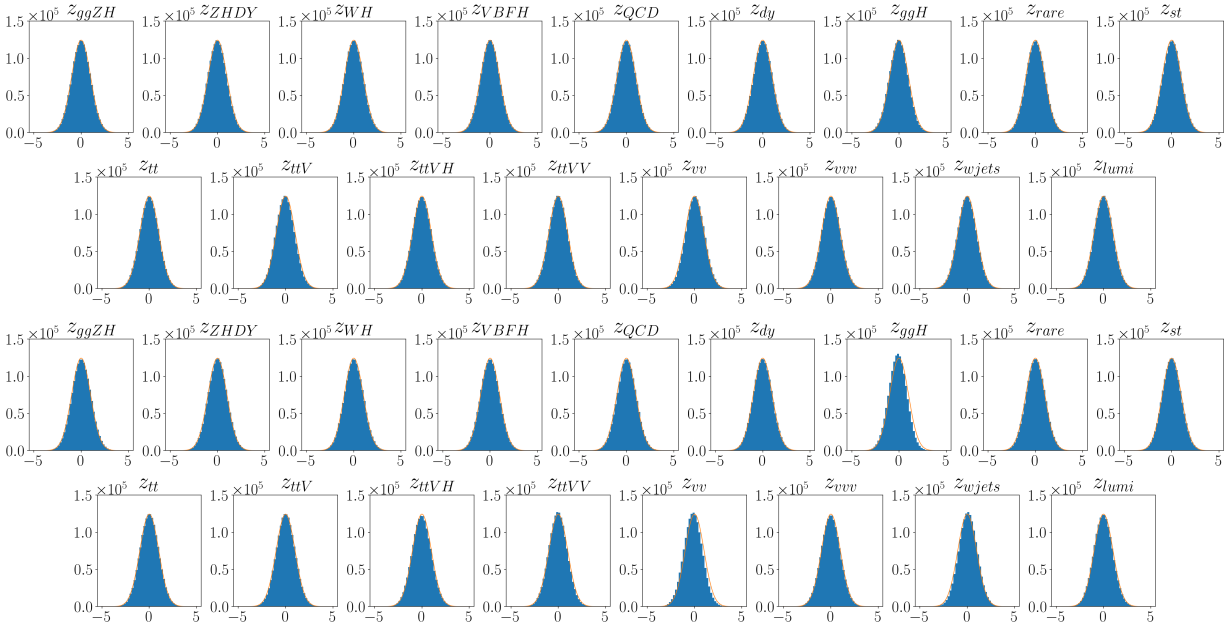

Figure 6.2: Latent space distribution for the cINN without (top) and with (bottom) a summary network.Need to be updated!

All of the above distributions in blue are well-described by the orange curve for both networks. The network thus manages to map the input distributions to $\mathcal{N}(z; 0, 1)$ well enough. The question now remains how well the cINN is capable of producing the posteriors.

6.3 Calibration Curves

A measure of evaluating the quality of the produced posteriors are the (median) calibration errors. The calibration error for a given quantile (i.e. $q \in [0, 1]$) is defined for all histograms N as

$$e_{cal}(q) = \frac{N_{in}^q}{N} - q$$

where N_{in}^q is the number of histograms containing the true value within their q quantile. For ideal posteriors, this measure is 0 for all quantiles q; any deviations from that signals the presence of biases and of hidden correlations within the network. Naturally, no network can reproduce the ideal results, as the global optimum can only be approximated (albeit arbitrarily well). For all quantiles, the ratio of histograms containing the true value in their q quantile as a function of q is shown in fig. 6.3.
A natural measure to describe how strongly these biases are present in the network is the meadian calibration error, defined as

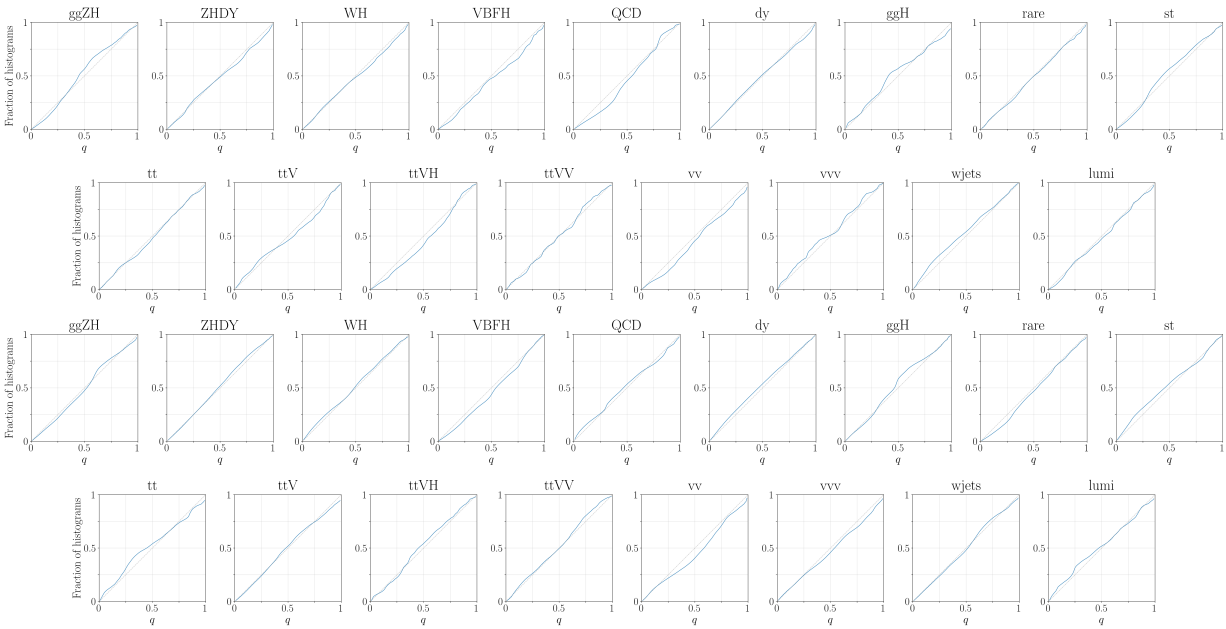

Figure 6.3: Need to be updated! The calibration curves for the "pure" cINN and summary network-extended cINN. Note the closeness to the perfect calibration curve shown in the gray dashed line.

$$e_{cal}^{med} = \underset{q}{\text{med}} \left| \frac{N_{in}^{q}}{N} - q \right|$$

describing the median absolute distance of the histogram ratios to the ideal case (where this ratio equals q). These values for both networks have been listed in tab. 6.1. As it can be seen from the table all of these values are $e_{cal}^{med} \lesssim \mathcal{O}(0.04)$. With such small mean deviations and calibration curves being close to the perfect calibration line, the resulting network model has no significant inherent biases (due to wrong initialization or other nontrivial effects).

6.4 Network Predictions

The predictions for the complete test dataset are shown in fig. 6.4. For each two dimensional histogram, the x axis shows the predicted values for a parameter on the y axis. In these plots several structures are clearly visible, which can be categorised into three groups for both networks:

Unrecognised processes are those, where the network has clear difficulties obtaining the parameter with which the processes had been scaled. These are `VBFH`, `ggH`, `ttVH`, `ttVV`, `vvv` all of which are rare background processes. Due to the network not being able to differentiate between the different scalings, all of these are fitted close to the mean of the prior distribution, resulting in the thin vertical distribution in fig. 6.4. looking at the DNN score histograms

Weakly recognised processes are `rare`, `ttV`, `vv`. For these processes, the network can already resolve sevaral parameters close to the mean, from where the sensitivity starts to develop during training. Parameters further away from the mean are still fit the average value, resulting in an overall lense-like shape.

Process	e_{cal}^{med}	
	cINN	SN-cINN
ggZH	0.0156	0.0258
ZHDY	0.0299	0.0103
WH	0.0148	0.0176
VBFH	0.0256	0.0428
QCD	0.0147	0.0285
dy	0.0133	0.0323
ggH	0.0395	0.0183
rare	0.0399	0.0261
st	0.0229	0.0323
tt	0.0101	0.0300
ttV	0.0212	0.0144
ttVH	0.0159	0.0227
ttVV	0.0403	0.0146
vv	0.0333	0.0441
vvv	0.0326	0.0331
wjets	0.0070	0.0127
lumi	0.0156	0.0248

Table 6.1: The median calibration errors for the cINN and the summary network-extended (SN-cINN).

Recongnised processes are all signal processes ggZH, ZHDY and WH and the major background sources QCD, dy, st, tt, wjets. For these processes, most scaling parameters predictions lie close to the real value, hence distribution scatter around the "perfect prediction diagonal".

Note the unique decrease in sensitivity for the signal processes ggZH, ZHDY and WH for $\mu \lesssim 10$ resulting in a tail-like structure stretching towards the mean. Uniquely to these processes, the network cannot resolve them as long as the signal strength modifier parameters lie below a certain threshold value for some configurations. Since their contribution (compared to the other significant background processes) is already low, the added statistical and systematic effects (the Poisson bin-variation being the most significant one, as it modifies all bins with $\sqrt{N}$) renders these statistically undetectable.
Note that the calibration curves show no signs of inherent biases in the resulting model, hence the fact that this effect is due to some training relics can be excluded. It is expected then, that the posterior distributions are broader in this range covering the true value, which will be discussed in the next section.
In general, this resolution effect only be can further tackled by a more refined event selection and/or category creation and similar improved analysis techniques.

6.5 Posterior Inference

Following the sampling from the target Gaussian function in the latent space, the posteriors can be inferred for the physics parameters for both networks. These are shown in fig. 6.5 for the nominal histogram where all expectations are $\mu = 1$. [7] The obtained posterior distributions

[7] Can be understood as inference on an Asimov datasets.

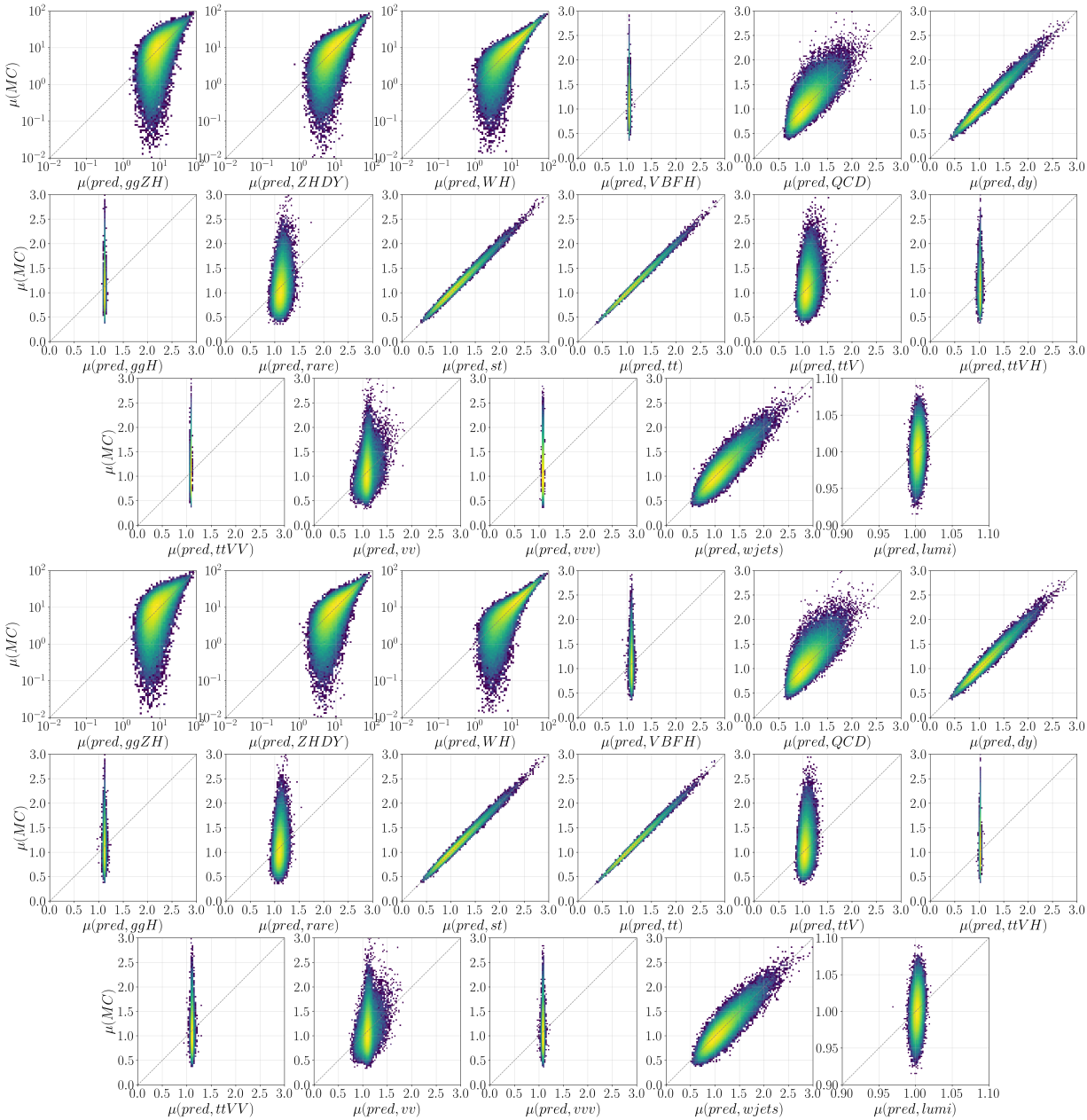

Figure 6.4: The predicted signal scale parameters vs. the simulated "real" Monte Carlo values. Note the three main

(normalized to 1) are shown in blue with their means (the predictions) drawn in black. The true values are shown with the red line. The edges of the 68% quantiles are drawn in black; these can be interpreted as the uncertainty on the scaling parameters. The original posterior distributions for the signal modifier parameters and the nuisance parameters are shown in orange.

Both networks succeed in the reconstruction of the nominal parameters with differing precision. It appears at first glance that the some recongnized background processes (`dy`, `st`, `tt`) have an order of magnitude narrower posterior distribution. From the posterior estimation, such behaviour is expected as the calibration curves show no signs of biases, meaning the posteriors should all be scattered around the true values with their widths covering it. This implies the

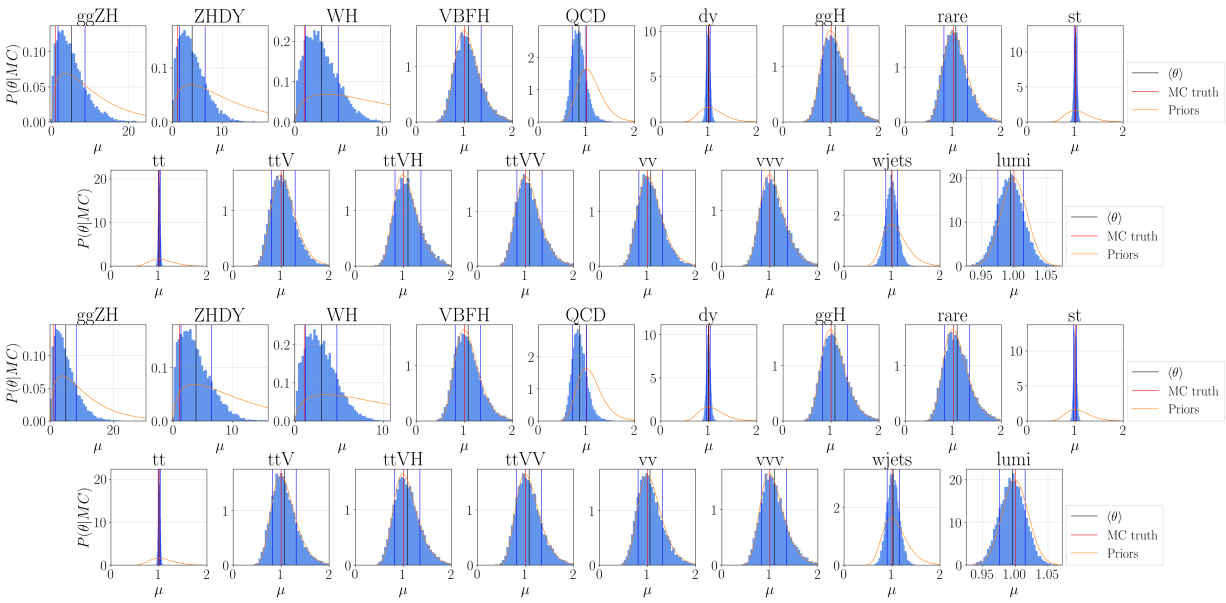

Figure 6.5: The posterior distributions for each process.

network sensitivity can only be characterized by the width of the posterior distributions for well-behaving (single-peaked) posteriors; in other words, enhanced sensitivity implies narrower posteriors.

All three signal processes are overestimated with the true values lying closely outside of the 68% quantiles. All background scaling parameters (except for `QCD`, which is underestimated and lies outside of the 1σ range) lie close to their nominal values.

For weakly recognized and unrecognized processes, an interesting property of cINNs can be observed. Taking a glimpse at the blue posteriors and the orange analytic prior distributions for any of these processes (`VBHF`, `ggH`, `rare`, `ttV`, `ttVH`, `ttVV`, `vv`, `vvv`) only minimal changes in the former can be observed. For the cINN, these processes are statistically independent of the posteriors (and therefore, redundant in a way), so the normalizing flow mapping the priors to the Gaussian distributions does not need to decorrelate them from the other observables (which, in reality, should be done, but their contribution appears negligible at the current sensitivity), as they – from the network's perspective – have no influence on the conditions whatsoever.

Mathematically speaking, if some physics observations y are independent of a phyics parameter x, it means that by Bayes' Theorem the posteriors can be rewritten using the priors $p(x)$ as

$$p(x|y) = \frac{p(y|x)p(x)}{p(y)} = \frac{\cancel{p(y)}}{\cancel{p(y)}}p(x)$$

Which means the posteriors and the priors coincide. Note that the independence should be mutual, so the probability of obtaining x given y does not change either, directly yielding $p(x|y) = p(x)$. Hence, the blue histograms coincide with the orange priors.

This observation shows the "naked" presence of the normalizing flow itself. The NF maps the datapoints from the prior distributions to a normal distribution via a diffeomorphism while the network is trained to optimize the modelled posterior distributions with respect to the unknown real distribution. From the results above, it becomes less surprising to observe that the network is capable of reproducing the priors for parameters which are independent of the conditions —

from a single condition and samples from the latent space distribution alone.
The numerical results for the inference have been summarized in tab. 6.2.

Processes	μ		
	cINN	SN-cINN	`CMSCombine`
μ_{ggZH}	$5.10^{+3.57}_{-3.50}$	$4.86^{+3.38}_{-3.33}$	combine
μ_{ZHDY}	$3.90^{+2.62}_{-2.57}$	$3.76^{+2.57}_{-2.52}$	
μ_{WH}	$3.02^{+1.92}_{-1.90}$	$2.93^{+1.80}_{-1.80}$	
μ_{VBFH}	$1.08^{+0.27}_{-0.27}$	$1.08^{+0.26}_{-0.26}$	–
μ_{QCD}	$0.84^{+0.14}_{-0.14}$	$0.86^{+0.15}_{-0.15}$	–
μ_{dy}	$0.99^{+0.04}_{-0.04}$	$0.99^{+0.04}_{-0.04}$	–
μ_{ggH}	$1.09^{+0.27}_{-0.27}$	$1.08^{+0.26}_{-0.26}$	–
μ_{rare}	$1.06^{+0.25}_{-0.25}$	$1.08^{+0.26}_{-0.26}$	–
μ_{st}	$1.00^{+0.03}_{-0.03}$	$1.00^{+0.03}_{-0.03}$	–
μ_{tt}	$1.00^{+0.02}_{-0.02}$	$1.00^{+0.02}_{-0.02}$	–
μ_{ttV}	$1.05^{+0.24}_{-0.24}$	$1.07^{+0.25}_{-0.25}$	–
μ_{ttVH}	$1.09^{+0.27}_{-0.27}$	$1.08^{+0.27}_{-0.27}$	–
μ_{ttVV}	$1.08^{+0.26}_{-0.27}$	$1.07^{+0.26}_{-0.26}$	–
μ_{vv}	$1.06^{+0.25}_{-0.25}$	$1.06^{+0.25}_{-0.25}$	–
μ_{vvv}	$1.09^{+0.27}_{-0.27}$	$1.09^{+0.27}_{-0.27}$	–
μ_{wjets}	$0.99^{+0.12}_{-0.12}$	$1.02^{+0.13}_{-0.13}$	–
μ_{lumi}	$1.00^{+0.02}_{-0.02}$	$1.00^{+0.02}_{-0.02}$	–

Table 6.2: The predicted values of the signal strength modifier and of the nuisance parameter for the cINN and the summary-network extended network (SN-cINN). Note the similarity of the output of both networks.

Bibliography

[1] Wikipedia contributors, "Higgs mechanism — Wikipedia, the free encyclopedia", 2022. https://en.wikipedia.org/w/index.php?title=Higgs_mechanism&oldid=1101993746. [Online; accessed 1-September-2022].

[2] J. Ellis, M. K. Gaillard, and D. V. Nanopoulos, "A Historical Profile of the Higgs Boson. An Updated Historical Profile of the Higgs Boson", (2016) , arXiv:1504.07217. https://cds.cern.ch/record/2012465. 22 pages, 5 figures, update of arXiv:1201.6045, to be published in the volume 'The Standard Theory of Particle Physics', edited by Luciano Maiani and Gigi Rolandi, pp. 255-274.

[3] S. C. et al., "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC", *Physics Letters B* **716** no. 1, (Sep, 2012) 30–61. https://doi.org/10.1016%2Fj.physletb.2012.08.021.

[4] **Particle Data Group** Collaboration, R. L. Workman and Others, "Review of Particle Physics", *PTEP* **2022** (2022) 083C01.

[5] CERN, "Cern yellow reports: Monographs, vol 2 (2017): Handbook of lhc higgs cross sections: 4. deciphering the nature of the higgs sector", 2017. https://e-publishing.cern.ch/index.php/CYRM/issue/view/32.

[6] R. V. Harlander, J. Klappert, C. Pandini, and A. Papaefstathiou, "Exploiting the WH/ZH symmetry in the search for new physics", *The European Physical Journal C* **78** no. 9, (Sep, 2018) . https://doi.org/10.1140%2Fepjc%2Fs10052-018-6234-x.

[7] A. Alici, M. Bomben, I. Dawson, and J. Sonneveld, "The LHC machine and experiments", (2021) . http://cds.cern.ch/record/2773265.

[8] S. Fartoukh, S. Kostoglou, M. Solfaroli Camillocci, G. Arduini, H. Bartosik, C. Bracco, K. Brodzinski, R. Bruce, X. Buffat, M. Calviani, F. Cerutti, I. Efthymiopoulos, B. Goddard, G. Iadarola, N. Karastathis, A. Lechner, E. Metral, N. Mounet, F.-X. Nuiry, P. S. Papadopoulou, Y. Papaphilippou, B. Petersen, T. H. B. Persson, S. Redaelli, G. Rumolo, B. Salvant, G. Sterbini, H. Timko, R. Tomas Garcia, and J. Wenninger, "LHC Configuration and Operational Scenario for Run 3", tech. rep., CERN, Geneva, Nov, 2021. https://cds.cern.ch/record/2790409.

[9] N. Karastathis, M. Barnes, H. Bartosik, K. Brodzinski, X. Buffat, F. Cerutti, S. Fartoukh, B. Goddard, G. Iadarola, S. L. Naour, A. Lechner, J. M. Heredia, A. Mereghetti, E. Metral, D. Missiaen, N. Mounet, F. X. Nuiry, S. Papadopoulou, Y. Papaphilippou, B. Petersen, G. Rumolo, B. Salvant, C. Schwick, M. S. Camillocci, G. Sterbini, H. Timko, R. T. Garcia, J. Uythoven, and J. Wenninger, "LHC Run 3 Configuration Working Group Report", (2019) . http://cds.cern.ch/record/2750302.

[10] E. Mobs, "The CERN accelerator complex in 2019. Complexe des accélérateurs du CERN en 2019", (Jul, 2019) . https://cds.cern.ch/record/2684277. General Photo.

[11] O. S. Brüning, P. Collier, P. Lebrun, S. Myers, R. Ostojic, J. Poole, and P. Proudlock, *LHC Design Report*. CERN Yellow Reports: Monographs. CERN, Geneva, 2004. http://cds.cern.ch/record/782076.

[12] L. Evans and P. Bryant, "LHC machine", *Journal of Instrumentation* **3** no. 08, (Aug, 2008) S08001–S08001. https://doi.org/10.1088/1748-0221/3/08/s08001.

[13] C. Bracco, "Commissioning Scenarios and Tests for the LHC Collimation System", 2009. http://cds.cern.ch/record/1174254. Presented on 29 Jan 2009.

[14] **CMS Collaboration** Collaboration, T. Sakuma, "Cutaway diagrams of CMS detector", (May, 2019) . http://cds.cern.ch/record/2665537.

[15] V. Veszpremi, "Operation and performance of the CMS tracker", *Journal of Instrumentation* **9** no. 03, (Mar, 2014) C03005–C03005. https://doi.org/10.1088%2F1748-0221%2F9%2F03%2Fc03005.

[16] P. Azzurri, "The CMS Silicon Strip Tracker", *J. Phys.: Conf. Ser.* **41** (Dec, 2005) 127–134. 8 p. http://cds.cern.ch/record/914891. 8 pages, 8 figures, talk given at XIX EPS NPDC Conference on New Trends in Nuclear Physics Applications and Technology, September 5-9, 2005 Pavia, Italy Subj-class: Instrumentation and Detectors.

[17] C. Biino, "The CMS electromagnetic calorimeter: overview, lessons learned during run 1 and future projections", *Journal of Physics: Conference Series* **587** (Feb, 2015) 012001. https://doi.org/10.1088/1742-6596/587/1/012001.

[18] **CMS Collaboration** Collaboration, C. et al., "Energy calibration and resolution of the CMS electromagnetic calorimeter in pp collisions at $\sqrt{s}$ = 7 TeV", *JINST* **8** (Jun, 2013) P09009. 51 p, arXiv:1306.2016. https://cds.cern.ch/record/1554142. Comments: Submitted to JINST.

[19] K. Cankoçak, P. de Barbaro, D. Vishnevskiy, Y. Onel, and C.-H. Collaboration, "CMS HCAL installation and commissioning", *Journal of Physics: Conference Series* **160** (Apr, 2009) 012055. https://doi.org/10.1088/1742-6596/160/1/012055.

[20] López, Silvia Goy, "Cms detector performance", *EPJ Web Conf.* **182** (2018) 02076. https://doi.org/10.1051/epjconf/201818202076.

[21] **CMS Collaboration** Collaboration, C. et al., "The CMS experiment at the CERN LHC. The Compact Muon Solenoid experiment", *JINST* **3** (2008) S08004. 361 p. https://cds.cern.ch/record/1129810. Also published by CERN Geneva in 2010.

[22] A. Colaleo, A. Safonov, A. Sharma, and M. Tytgat, "CMS Technical Design Report for the Muon Endcap GEM Upgrade", tech. rep., Jun, 2015. https://cds.cern.ch/record/2021453.

[23] **CMS Collaboration** Collaboration, B. et al., *CMS Physics: Technical Design Report Volume 1: Detector Performance and Software*. Technical design report. CMS. CERN, Geneva, 2006. https://cds.cern.ch/record/922757. There is an error on cover due to a technical problem for some items.

[24] D. Barney, "CMS Detector Slice", Jan, 2016. http://cds.cern.ch/record/2120661. CMS Collection.

[25] A. S. et al., "Particle-flow reconstruction and global event description with the CMS detector", *Journal of Instrumentation* **12** no. 10, (Oct, 2017) P10003–P10003. https://doi.org/10.1088%2F1748-0221%2F12%2F10%2Fp10003.

[26] M. Cacciari, G. P. Salam, and G. Soyez, "The anti-kt jet clustering algorithm", *Journal of High Energy Physics* **2008** no. 04, (Apr, 2008) 063–063. https://doi.org/10.1088%2F1126-6708%2F2008%2F04%2F063.

[27] A. S. et al., "Identification of heavy-flavour jets with the CMS detector in pp collisions at 13 TeV", *Journal of Instrumentation* **13** no. 05, (May, 2018) P05011–P05011. https://doi.org/10.1088%2F1748-0221%2F13%2F05%2Fp05011.

[28] C. Collaboration, "Performance of CMS muon reconstruction in cosmic-ray events", *Journal of Instrumentation* **5** no. 03, (Mar, 2010) T03022–T03022. https://doi.org/10.1088/1748-0221/5/03/t03022.

[29] A. S. et al., "Performance of the cms muon detector and muon reconstruction with proton-proton collisions at $\sqrt{s}$ = 13 tev", *Journal of Instrumentation* **13** no. 06, (Jun, 2018) P06015–P06015. https://doi.org/10.1088%2F1748-0221%2F13%2F06%2Fp06015.

[30] M. Erdmann, J. Glombitza, G. Kasieczka, and U. Klemradt, *Deep Learning for Physics Research*. WORLD SCIENTIFIC, 2021. https://www.worldscientific.com/doi/pdf/10.1142/12294. https://www.worldscientific.com/doi/abs/10.1142/12294.

[31] B. Hanin and M. Sellke, "Approximating continuous functions by relu nets of minimal width", 2017. https://arxiv.org/abs/1710.11278.

[32] G. Papamakarios, E. Nalisnick, D. J. Rezende, S. Mohamed, and B. Lakshminarayanan, "Normalizing flows for probabilistic modeling and inference", (2019) . https://arxiv.org/abs/1912.02762.

[33] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", 2014. https://arxiv.org/abs/1412.6980.

[34] L. Ardizzone, C. Lüth, J. Kruse, C. Rother, and U. Köthe, "Guided image generation with conditional invertible neural networks", *CoRR* **abs/1907.02392** (2019) , 1907.02392. http://arxiv.org/abs/1907.02392.

[35] M. Bellagente, A. Butter, G. Kasieczka, T. Plehn, A. Rousselot, R. Winterhalder, L. Ardizzone, and U. Köthe, "Invertible networks or partons to detector and back again", *SciPost Physics* **9** no. 5, (Nov, 2020) . https://doi.org/10.21468%2Fscipostphys.9.5.074.

[36] V. F. Ksoll, L. Ardizzone, R. Klessen, U. Koethe, E. Sabbi, M. Robberto, D. Gouliermis, C. Rother, P. Zeidler, and M. Gennaro, "Stellar parameter determination from photometry using invertible neural networks", *Monthly Notices of the Royal Astronomical Society* **499** no. 4, (Nov, 2020) 5447–5485. https://doi.org/10.1093%2Fmnras%2Fstaa2931.

[37] T. Bister, M. Erdmann, U. Köthe, and J. Schulte, "Inference of cosmic-ray source properties by conditional invertible neural networks", *The European Physical Journal C* **82** no. 2, (Feb, 2022) . https://doi.org/10.1140%2Fepjc%2Fs10052-022-10138-x.

[38] L. Ardizzone, J. Kruse, S. Wirkert, D. Rahner, E. W. Pellegrini, R. S. Klessen, L. Maier-Hein, C. Rother, and U. Köthe, "Analyzing inverse problems with invertible neural networks", 2018. https://arxiv.org/abs/1808.04730.

[39] S. T. Radev, U. K. Mertens, A. Voss, L. Ardizzone, and U. Köthe, "Bayesflow: Learning complex stochastic models with invertible neural networks", 2020. https://arxiv.org/abs/2003.06281.

[40] D. P. Kingma and P. Dhariwal, "Glow: Generative flow with invertible 1x1 convolutions", 2018. https://arxiv.org/abs/1807.03039.

[41] L. Dinh, J. Sohl-Dickstein, and S. Bengio, "Density estimation using real nvp", 2016. https://arxiv.org/abs/1605.08803.

[42] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag, Berlin, Heidelberg, 2006.

[43] T. Sjöstrand, S. Mrenna, and P. Skands, "A brief introduction to PYTHIA 8.1", *Computer Physics Communications* **178** no. 11, (Jun, 2008) 852–867. https://doi.org/10.1016%2Fj.cpc.2008.01.036.

[44] S. Gieseke, D. Grellscheid, K. Hamilton, A. Papaefstathiou, S. Plätzer, P. Richardson, C. A. Röhr, P. Ruzicka, A. Siódmok, L. Suter, and D. Winn, "Herwig++ 2.5 release note", 2011. https://arxiv.org/abs/1102.1672.

[45] S. A. et al., "Geant4—a simulation toolkit", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **506** no. 3, (2003) 250–303. https://www.sciencedirect.com/science/article/pii/S0168900203013688.

[46] A. et al., "Geant4 developments and applications", *IEEE Transactions on Nuclear Science* **53** no. 1, (2006) 270–278.

[47] J. A. et al., "Recent developments in geant4", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **835** (2016) 186–225. https://www.sciencedirect.com/science/article/pii/S0168900216306957.

[48] M. Baak, S. Gadatsch, R. Harrington, and W. Verkerke, "Interpolation between multi-dimensional histograms using a new non-linear moment morphing method", *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* **771** (Jan, 2015) 39–48. https://doi.org/10.1016%2Fj.nima.2014.10.033.

[49] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library", in *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, eds., pp. 8024–8035. Curran Associates, Inc., 2019. http://papers.neurips.cc/paper/9015-pytorch-an-imperative-style-high-performance-deep-learning-library.pdf.

[50] L. Ardizzone, T. Bungert, F. Draxler, U. Köthe, J. Kruse, R. Schmier, and P. Sorrenson, "Framework for Easily Invertible Architectures (FrEIA)", 2018-2022. https://github.com/VLL-HD/FrEIA.

Selbständigkeitserklärung

Hiermit versichere ich an Eides statt, dass ich diese Arbeit einschließlich evtl. beigefügter Abbildungen, Zeichnungen u.Ä.m. selbstständig angefertigt und keine anderen als die angegebenen Hilfsmittel und Quellen benutzt habe. Alle Stellen, die dem Wortlaut oder dem Sinn nach anderen Werken entnommen sind, habe ich in jedem einzelnen Fall unter genauer Angabe der Quelle deutlich als Entlehnung kenntlich gemacht.

Aachen, den XX. MONTH 2018

Mate Zoltan Farkas

Danksagung

Thx.