

CMS-EXPERIMENT
JET REKONSTRUKTION
DATEN SIMULATION ÜEBEREINSTIMMUNG
DOMAIN ADAPTATION

David Walter

Fakultät für Physik
Karlsruher Institut für Technologie (KIT)

MASTERARBEIT

Referent: Dr. Matthias Ulrich Mozer
Korreferent: Prof. Dr. Günter Quast
Institut für Experimentelle Teilchenphysik

03. September 2018

Akzeptiert vom ersten Referenten der Masterarbeit.
Karlsruhe, den 03. September 2017

.....
(Dr. Matthias Ulrich Mozer)

Ich versichere wahrheitsgemäß, die Arbeit selbstständig angefertigt, alle benutzten Hilfsmittel vollständig und genau angegeben und alles kenntlich gemacht zu haben, was aus Arbeiten anderer unverändert oder mit Abänderungen entnommen wurde.

Karlsruhe, den 03. September 2017

.....
(David Walter)

Deutsche Zusammenfassung

DOMAIN ADAPTATION STUDIES IN DEEP NEURAL
NETWORKS FOR HEAVY FLAVOR TAGGING ALGORITHMS
AT THE CMS EXPERIMENT

David Walter

Department of Physics
Karlsruhe Institute of Technology (KIT)

MASTER THESIS

Referee: Dr. Matthias Ulrich Mozer
Co-Referee: Prof. Dr. Günter Quast
Institut für Experimentelle Teilchenphysik

September 03, 2018

Introduction

Contents

1	The Standard Model	5
1.1	Overview	5
1.2	Electroweak Interaction	6
1.3	Quantum Chromodynamics	8
2	The Compact Muon Solenoid Experiment at the Large Hadron Collider	11
2.1	The Large Hadron Collider	11
2.2	The Compact Muon Solenoid Detector	13
3	Simulation and Reconstuction of Events	19
3.1	Event Simulation	19
3.1.1	Proton-Proton Scattering	19
3.1.2	Simulation Tools	22
3.2	Event Reconstruction	23
3.2.1	Tracks and Vertices	23
3.2.2	Particle Reconstruction	24
3.2.3	Jet Reconstruction	25
3.2.4	Lepton Isolation	26
3.2.5	Reconstruction of Neutrinos and Leptonically Decaying W Bosons	27
3.3	Differences between Data and Simulation	28
3.3.1	Jet Energy Corrections	28
3.3.2	Pileup Reweighting	28
3.3.3	Trigger and Lepton Efficiencies	29
4	Artificial Neural Networks	31
4.1	Multivariate Classification	31
4.2	Perceptron	32
4.3	Multilayer Perceptron	32
4.3.1	Fully Connected Feed-Forward Networks	33
4.3.2	Training	34
4.3.3	Regularization	36
4.4	Domain Adaptation in Deep Neural Networks	37
4.4.1	Moments Method	38

4.4.2	Domain-Adversarial Method	39
5	Heavy Flavor Jet Tagging	41
5.1	Heavy Flavor Jet Properties	41
5.1.1	Secondary Vertex Reconstruction	42
5.2	The DeepCSV Algorithm	42
5.2.1	Categories	43
5.2.2	Input Features	43
5.3	The DeepFlavour Algorithm	43
5.3.1	Architecture	44
5.3.2	Categories	46
5.3.3	Input Features	46
5.4	Performance of b Tagging Algorithms	46
6	Event Selection	49
6.1	Top Quark Decay	49
6.2	Particle Identification	51
6.2.1	Muon Selection	51
6.2.2	Electron Selection	51
6.2.3	Jet Selection	52
6.3	Top Quark Pair - Semileptonic Selection	52
6.3.1	Datasets	53
6.3.2	Data-to-Simulation Comparison	54
6.4	Top Quark Pair - Dileptonic Selection	57
6.4.1	Datasets	57
6.4.2	Data-to-Simulation Comparison	58
7	Domain Adaptation Studies	61
7.1	Domain Adaptation on the DeepFlavour Algorithm	61
7.1.1	•	61
7.2	Domain Adaptation on the DeepCSV Algorithm	61
7.3	Two Tag Efficiencies	61

1 The Standard Model

The standard model of particle physics (SM) describes all known elementary particles and three of the four known fundamental forces. These forces are the electromagnetic interaction, the weak interaction and the strong interaction. Gravity is not included yet, but is negligible on a microscopic scale. The SM predicted many particles that were found later in experiments, for example the top quark in 1995 [1], the tau neutrino in 2000 [2] or the Higgs boson in 2012 [3, 4]. It also provides precise predictions about the properties of elementary particles like the Landé g -factor [5]. Despite the great success of the SM, hints exist that the particles and their interactions are described by a more fundamental mechanism. For example, many free parameters, like the masses of particles, cannot be predicted by the SM and have to be determined by measurements. Furthermore, dark matter and dark energy are not described by the SM. Therefore, to find evidences for new physics, it is important to measure the particle properties predicted by the SM with best possible precision.

This chapter gives a brief summary of the SM with focus on the electroweak interaction and quantum chromodynamics. A more detailed description of the SM can be found in references [6–9]. Natural units with $\hbar = c = 1$ are used to simplify the formulas quoted in this chapter.

1.1 Overview

The SM is a quantum field theory that describes the particles as fields and that includes quantum mechanics as well as special relativity. It is a gauge theory in which underlying interactions are described by the $U(1)_Y \otimes SU(2)_L \otimes SU(3)_C$ symmetry group.

Special relativity requires a Lorentz invariant description of the SM. The spin of a particle defines its representation of the Lorentz group and has a direct influence on its behavior under Lorentz transformation. With respect to the spin quantum number, the particles in the SM can be divided into two groups, fermions with half integer spin and bosons with integer spin. Fermions are described as spinor fields and are the constituents of all microscopic matter, for example atoms. They exist in three generations, where only the first generation of particles is stable and does not decay over time. Fermions can be further distinguished into quarks and leptons. Quarks are the constituents of protons and neutrons, they interact through all forces and can be dis-

Table 1.1: The fermions of the standard model. All quarks and leptons of the SM and their electroweak charges are listed. The fermions occur with negative chirality (left handed) in weak isospin doublets or with positive chirality (right handed) in weak isospin singlets. Under the assumption that the neutrino mass is zero, there is no right handed neutrino in the SM. For each fermion, an antifermion exists with opposite charge and chirality.

Fermions	Generation			Weak hyper-charge Y_W	3 rd comp. of isospin I_3	Electric charge Q
	1	2	3			
Quarks	$\begin{pmatrix} u \\ d \end{pmatrix}_L$	$\begin{pmatrix} c \\ s \end{pmatrix}_L$	$\begin{pmatrix} t \\ b \end{pmatrix}_L$	$+1/3$	$+1/2$	$+2/3$
	u_R	c_R	t_R	$+1/3$	$-1/2$	$-1/3$
	d_R	s_R	b_R	$+4/3$	0	$+2/3$
				$-2/3$	0	$-1/3$
Leptons	$\begin{pmatrix} \nu_e \\ e \end{pmatrix}_L$	$\begin{pmatrix} \nu_\mu \\ \mu \end{pmatrix}_L$	$\begin{pmatrix} \nu_\tau \\ \tau \end{pmatrix}_L$	-1	$+1/2$	0
	e_R	μ_R	τ_R	-1	$-1/2$	-1
				-2	0	-1

tinguished by their electric charge. Quarks with an electric charge of $+2/3e$ are called up-type quarks, while down-type quarks carry an electric charge of $-1/3e$. The charged leptons, the electron, the muon and the tauon can interact via the electromagnetic and weak force. The neutral leptons, the neutrinos, can only interact via the weak force. A summary of the fermions is given in table 1.1. The second group of the SM consists of bosons. Bosons with spin 1, described by vector fields, are the mediator particles for the three fundamental forces of the SM: the photon (γ) for the electromagnetic interaction, the W^\pm and Z bosons for the weak interaction and the gluons (g) for the strong interaction. The Higgs boson is the only particle of the SM with spin 0 and is described by a scalar field. It interacts with all massive particles. A summary of the bosons is shown in table 1.2.

1.2 Electroweak Interaction

The electromagnetic interaction and the weak interaction are described in a unified electroweak theory by the $U(1)_Y \otimes SU(2)_L$ group. By demanding the theory to be locally invariant under $U(1)_Y$ transformations, there must exist a massless vector boson (B), and a conserved quantity, the weak hypercharge Y_W . To obtain a local $SU(2)_L$ symmetry, three additional massless vector bosons (W_0 , W_1 and W_2) are needed; the conserved quantity of this symmetry is the third component of the weak isospin I_3 . The electric

Table 1.2: Summary of the bosons of the SM. The vector bosons with spin 1, the photon, the W^\pm and Z bosons and the gluons are the mediator particles of the three fundamental forces that are described by the SM. The Higgs boson with spin 0 is not mediating a force, but it couples to the mass of the particles, masses are taken from [10]

Boson	Force	Couples to	Mass (GeV)
photon (γ)	electromagnetic	electric charge	0
W^\pm	weak	weak charge	80.385
Z			91.188
8 gluons (g)	strong	color charge	0
Higgs (H)	-	mass	125.09

charge

$$Q = \frac{Y_W}{2} + I_3 \quad , \quad (1.1)$$

is related with these conserved quantities and is hence conserved. The Higgs boson originates from a spontaneous symmetry breaking of the local $U(1)_Y \otimes SU(2)_L$ symmetry and leads to a non-diagonal mass matrix for the four vector bosons. This mass matrix can be diagonalized to find the mass eigenstates

$$\begin{aligned} \gamma &= \cos(\theta_W)B + \sin(\theta_W)W_0 \quad , \\ Z &= -\sin(\theta_W)B + \cos(\theta_W)W_0 \quad , \\ W^\pm &= \frac{1}{\sqrt{2}}(W_1 \mp iW_2) \quad , \end{aligned} \quad (1.2)$$

with the Weinberg angle θ_W . In these mass eigenstates, the photon remains massless while the Z, W^+ and W^- bosons receive a mass. The Z boson and the photon carry no electric charge, while the W^\pm bosons carry an electric charge of $\pm 1e$. The large mass of the Z and W^\pm bosons is responsible for the short range of the weak interaction, as they have a lifetime of less than 10^{-24} s [10]. The effective strength can be estimated with the Fermi coupling constant of $G_F \approx 10^{-5} \text{ GeV}^{-2}$ of the weak interaction. The electromagnetic coupling constant has a value in the order of 10^{-2} .

With respect to $SU(2)_L$ transformations, the quarks and leptons of one generation are grouped into weak isospin doublets, consisting of two left-handed spinor fields with $I_3 = \pm 1/2$. Weak isospin singlets with $I_3 = 0$ are represented by right-handed spinor fields, which cannot couple to charged currents. By interacting through W^\pm bosons, flavor¹ changing charged currents are possible. They describe transitions of fermions

¹The flavor indicates the type of a particle. For example, there are six quark flavors (u, d, s, c, b, t) and six lepton flavors (e, μ , τ , ν_e , ν_μ , ν_τ).

within a weak isospin doublet, for example $u \rightarrow d + W^+$. For quarks, this current is given by

$$J_{CC}^\mu = \begin{pmatrix} \bar{u} & \bar{c} & \bar{t} \end{pmatrix} \gamma^\mu \frac{1 - \gamma^5}{2} \begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} \quad (1.3)$$

$$= \bar{u} \gamma^\mu \frac{1 - \gamma^5}{2} d' + \bar{c} \gamma^\mu \frac{1 - \gamma^5}{2} s' + \bar{t} \gamma^\mu \frac{1 - \gamma^5}{2} b' \quad ,$$

with the Dirac matrices γ^μ and γ^5 and $\bar{q} = q^\dagger \gamma_0$ denoting the Dirac adjunct spinor of q . The quarks described in table 1.1, are in their electroweak eigenstate and are distinguished from the observable mass eigenstates. Here, q' denotes a quark in its electroweak eigenstate and q in its mass eigenstate. The mass eigenstates of the quarks are therefore a mixture of their electroweak eigenstates and vice versa:

$$\begin{pmatrix} d' \\ s' \\ b' \end{pmatrix} = V \begin{pmatrix} d \\ s \\ b \end{pmatrix} = \begin{pmatrix} V_{ud} & V_{us} & V_{ub} \\ V_{cd} & V_{cs} & V_{cb} \\ V_{td} & V_{ts} & V_{tb} \end{pmatrix} \begin{pmatrix} d \\ s \\ b \end{pmatrix} \quad . \quad (1.4)$$

The 3×3 unitary matrix V is the Cabibbo-Kobayashi-Maskawa matrix (CKM-matrix) that allows transitions of quarks between two generations. The probability of these transitions is low due to small off-diagonal values of V [10]:

$$V = \begin{pmatrix} 0.97417 \pm 0.00021 & 0.2248 \pm 0.0006 & 0.00409 \pm 0.00039 \\ 0.220 \pm 0.005 & 0.995 \pm 0.016 & 0.0405 \pm 0.0015 \\ 0.0082 \pm 0.0006 & 0.0400 \pm 0.0027 & 1.009 \pm 0.031 \end{pmatrix} \quad . \quad (1.5)$$

The matrix V has four free parameters that cannot be predicted by the SM and have to be determined by measurements. By inserting equation 1.4 into equation 1.3, one can, for instance, obtain the current for a transition of a b quark to a c quark. The relevant part is given by

$$J_{CC,cb}^\mu = \bar{c} \gamma^\mu \frac{1 - \gamma^5}{2} V_{cb} b \quad , \quad (1.6)$$

which is suppressed by the factor of $V_{cb} = 0.0405$.

1.3 Quantum Chromodynamics

The strong interaction is responsible for the interaction between quarks and gluons and is described by quantum chromodynamics (QCD) with the underlying symmetry of the $SU(3)_C$ group. Eight gluons are needed to obtain a $SU(3)_C$ invariant description. The charge of this symmetry is called color, of which three different types exist: red,

green and blue. For each color, there is also a corresponding anticolor. As this symmetry is not broken, the gluons remain massless and indistinguishable. They carry one color and one anticolor charge, this allows interactions of gluons among themselves. Quarks on the other hand carry one color charge while antiquarks are charged with an anticolor.

The nature of the strong force leads to a potential for a quark-antiquark pair of

$$V(r) = -a \cdot \frac{\alpha_s(r)}{r} + b \cdot r \quad , \quad (1.7)$$

with the positive constants a and b , the radius between the two particles r and the strong couplings constant $\alpha_s(r)$, which is dependent on the radius and is in the order of 1^2 . For small radii, the first term is dominant and allows bound states. The lower the distance between two particles, the lower is the strong interaction between them, this effect is called asymptotic freedom. On the other hand, the potential increases linearly with the distance between the particles. If a quark in a bound state is pulled apart from the other quark(s) in this bound state, the energy of the system is steadily increasing. When the energy is high enough, a quark-antiquark pair is produced in order to form two color neutral objects with the separated particles. This effect is called confinement and is the reason that no color-charged isolated object exists. A color-neutral object that consists of quarks is called hadron.

Hadrons can be grouped into mesons and baryons. Mesons are quark-antiquark pairs, whereas baryons consist of three quarks. Besides the top quark, all quarks can be bound in hadrons, therefore a huge number of different hadrons exists. Hadrons are produced by the strong force and decay in flavor-changing charged currents as described in equation 1.3. Therefore, the lifetime of hadrons [10] varies from 10^{-8} s for charged pions ($u\bar{d}$, $\bar{u}d$), 10^{-12} s for B mesons ($u\bar{b}$, $\bar{u}b$, $\bar{b}d$, $b\bar{d}$) to values up to 10^{-24} s for hadrons in excited states like the ρ meson ($u\bar{d}$, $\bar{u}d$). The only stable hadron is the proton (uud), as the neutron (udd) is only stable in atomic nuclei. The top quark is the only quark that cannot occur in a bound state because its lifetime of about $5 \cdot 10^{-25}$ s [11] is roughly 20 times smaller than the hadronization time.

²For instance, at the energy of the mass of the Z boson, $M_Z = 91.188$ GeV, $\alpha_s(M_Z^2) = 0.1181 \pm 0.0013$ [10]

2 The Compact Muon Solenoid Experiment at the Large Hadron Collider

Particle accelerator experiments have a great history of success in the investigation of physical laws. With these machines, it was possible to prove the existence of different elementary particles of the standard model, for instance, the top quark [1]. Particle accelerators make use of the electric charge of particles to accelerate them to velocities close to the speed of light. In a circular collider, particles are moving in circles in opposite directions and are forced to head-on collisions. These collisions release a high amount of energy and because of the laws of physics, heavy particles are produced that are highly unstable and decay after a short time. Detectors are used to measure the properties of these particles directly or indirectly.

This chapter gives a brief overview of the collider and detector system that has produced and measured the events used in this work.

2.1 The Large Hadron Collider

The Large Hadron Collider (LHC) [12, 13] is a circular collider located at the European Organization for Nuclear Research (CERN) in the region of Geneva. With a circumference of 27 km, it is the world's largest and most powerful particle accelerator and lies in a tunnel about 100 m beneath the earth. The LHC is used to accelerate protons, xenon ions and lead ions. These particles undergo several steps of creation and pre-acceleration before they are injected into the LHC. An illustration of the system is shown in figure 2.1.

The LHC consists of two beam pipes, allowing the acceleration of protons in opposite directions. In this setup, for head-on collisions with equal beam energies, the center-of-mass energy is twice the beam energy E :

$$\sqrt{s} = 2 \cdot E \quad . \quad (2.1)$$

In 2010, the first collisions with a beam energy of 3.5 TeV started. In 2012, the center-of-mass energy was increased to $\sqrt{s} = 8$ TeV. From 2013 to 2015, the LHC was shut down for maintenance. After this, Run 2 started in 2015 with $\sqrt{s} = 13$ TeV. After a shutdown

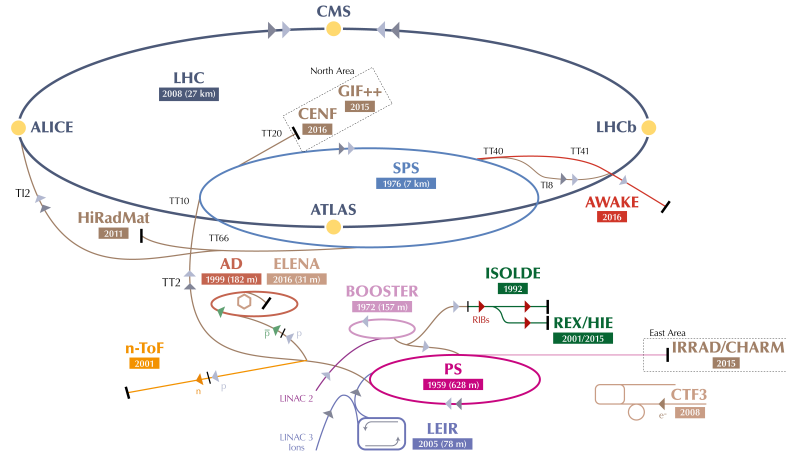


Figure 2.1: The CERN accelerator complex. The figure shows several accelerator facilities at CERN. The protons that end up in the LHC are obtained from hydrogen atoms and are first accelerated with the **LINAC2** accelerator to 50 MeV. Next, they are injected into the **Booster** accelerator and are accelerated to 1.4 GeV. It follows the acceleration in the **Proton Synchrotron (PS)** to 25 GeV and in the **Super Proton Synchrotron (SPS)** to 450 GeV. Finally, the protons are injected into the LHC in both directions. From there they reach a final energy of 6.5 TeV within 20 minutes. In addition, the four main detectors at the LHC are shown. Source: [14]

in 2018, the LHC is planned to run at $\sqrt{s} = 14$ TeV in 2021 for Run 3.

The protons in the beam are packed in up to 2808 bunches, each bunch consists of about $1.2 \cdot 10^{11}$ protons. For each beam, eight superconducting cavities, cooled down to 4.5 K, generate an electromagnetic potential with a frequency of 400 MHz to accelerate and tighten the bunches. Each of the cavities is delivering an accelerating field of 5 MV/m. To avoid collisions with gas molecules during operation, the beam pipes have a ultra-high vacuum pressure of 10^{-13} atm. Dipole magnets with a magnetic field strength of up to 7.74 T each, force the particle beam to stay on the circular path. Quadrupole magnets and magnets of higher order are used to focus the beam. To keep the magnets in superconducting state, superfluid helium is used to cool them down to 1.9 K.

Besides the center-of-mass energy, the luminosity is another important quantity for particle accelerators. The luminosity L determines the event rate and therefore, a high luminosity is desired:

$$\frac{dN}{dt} = \sigma L \quad , \quad (2.2)$$

where N is the number of events and σ is the cross section, given by the physical laws described in chapter 1. The luminosity depends on the properties of the particle

accelerator:

$$L = \frac{nN_1N_2f}{A} . \quad (2.3)$$

Here, n is the total number of bunches in the collider, N_1 and N_2 are the number of particles in the colliding bunches, f is the circulating frequency for one single bunch and A is the cross-sectional area. As the luminosity is inversely proportional to the area, the beam must be focused as much as possible to reach a high luminosity. The total number of events can be derived by integrating equation 2.1 over time to

$$N = \sigma \int L dt = \sigma \mathcal{L} , \quad (2.4)$$

with the integrated luminosity \mathcal{L} . In 2016, a peak luminosity of $1.5 \cdot 10^{34} \text{ cm}^{-2}\text{s}^{-1}$ was reached with an integrated luminosity of 41 fb^{-1} ($1 \text{ b} = 10^{-24} \text{ cm}^2$) for the total year [15]. After Run 3 at the LHC, a high luminosity upgrade is planned in which the luminosity will be increased to $5 \cdot 10^{34} \text{ cm}^{-2}\text{s}^{-1}$.

There are four interaction points at the LHC. At each point, one detector is located. One is for the ALICE experiment that studies collisions of lead ions. Another one is for the LHCb experiment, which is designed for the studies of b hadron physics. The ATLAS and the CMS experiments are general-purpose experiments, measuring the standard model particles and searching for new elementary particles.

2.2 The Compact Muon Solenoid Detector

The construction of the Compact Muon Solenoid (CMS) detector [16] was motivated by the search for the Higgs boson and the search for physics beyond the standard model. The former one has already been found [3, 4], but the search for alternative theories like supersymmetry or extra dimensions is still ongoing. Besides of these two big tasks, the CMS detector is used for several other studies, like the measurement of the couplings of different particles [17]. In order to achieve these goals, the detector is required to measure the properties of all particles with best possible precision. A sketch of a slice of the CMS detector with its different sub-detectors is shown in figure 2.2. To store only the interesting events and reduce the amount of data to an amount that can be stored for further analyses, a filter system, the so called trigger system, is applied. In this section, the sub-detectors and the trigger system are briefly described. At the beginning the coordinate system used within the CMS detector is introduced.

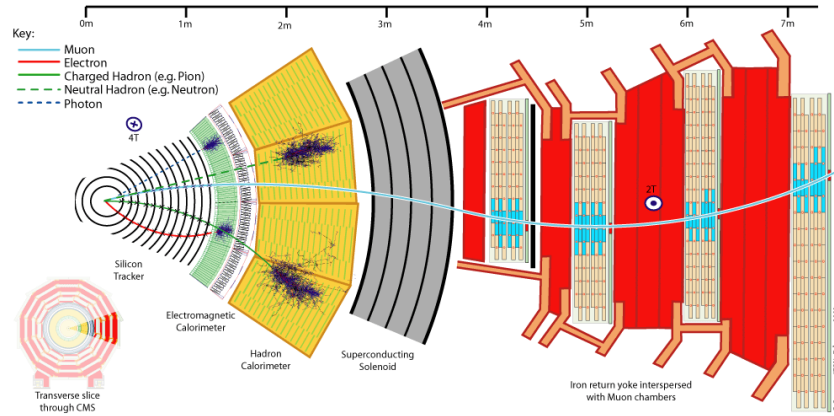


Figure 2.2: Slice of the CMS detector. The scheme shows the different detector systems as well as the tracks and the interactions of different particles with the detector. Source: [18]

Coordinate System

The origin of the coordinate system is chosen to be the intended collision point. The z axis points in beam direction. The x axis lies in the plane of the collider ring and points to its center. The y axis points vertically upwards. The azimuthal angle ϕ is defined as the angle in the x - y plane measured from the x -axis. The radial distance is $r = \sqrt{x^2 + y^2}$. The polar angle θ is measured from the z axis. Another important quantity is the rapidity, which is a measure for the velocity of a particle and defined as

$$y = \text{artanh} \left(\frac{v}{c} \right) , \quad (2.5)$$

where v is the velocity of the particle and c is the speed of light. y is in contrast to v unlimited and can take values in the range of $-\infty$ to ∞ , further one can just add two rapidities together instead of using the relativistic velocity-addition formula. The pseudorapidity η is a frequently used quantity in particle physics, in the relativistic approximation, η is equal to y and given by the equation

$$\eta = -\ln \left[\tan \left(\frac{\theta}{2} \right) \right] , \quad (2.6)$$

which is used to denote the angle between a vector and the z axis. A value of $\eta = 0$ is equal to $\theta = 90^\circ$; an angle of $\theta = 0^\circ$ equals $\eta = \infty$. The pseudorapidity is preferred over θ since the number of produced particles per η -interval is constant and the difference between two particles $\Delta\eta$ is invariant under lorentz boosts in z direction. The momentum and energy of the initial partons is not determined in a hadron collider. Because of this, mainly the quantities transverse to the beam axis are of interest. These quantities, the transverse momentum p_T and the transverse energy E_T are invariant

under lorentz boosts in z direction. They are defined as

$$p_T = \sqrt{p_x^2 + p_y^2} \quad \text{and} \quad E_T = \sqrt{E_x^2 + E_y^2} \quad . \quad (2.7)$$

Due to momentum conservation, the transverse momenta of all final particles sum up to zero. In the measurement there is always a missing amount of transverse momentum caused by gaps in the detector and neutrinos, which are not detectable within the CMS detector. The missing amount of transverse momentum is denoted as p_T^{miss} .

Silicon Tracker

Immediately after the collision, heavy objects are formed which decay after a short time and distance. The innermost part of the detector is of highest importance to reconstruct these objects and has to fulfill several requirements. First, a precise measurement of trajectories of charged particles is necessary to identify primary and secondary vertices. This also allows a momentum measurement as the full tracker is inside the superconducting solenoid, which produces a homogeneous magnetic field of 4 T. Next, at each bunch crossing, about 1000 particles on average are created and travel through the detector. To distinguish these particles from each other and to assign the trajectories to the corresponding vertices, a high granularity is needed. Further, the time between two bunch crossings is about 25 ns, therefore the response time and the cooling down period has to be short. Last, the detector components have to be resilient against this high amount of radiation.

To fulfill these requirements, silicon technology is used for the whole tracker system [19]. When charged particles travel through these materials, electron-hole pairs are produced, which in turn generate an electric current in the read-out electronics. In the innermost part from $r = 4.4$ cm to 10.2 cm, three barrel layers of pixel detectors are surrounding the beam axis. Each pixel detector has a size of $100 \times 150 \mu\text{m}^2$. With this design, the desired impact parameter resolution is achieved. The pixel detectors are surrounded by ten layers of silicon micro-strip detectors up to a radius of 1.1 m. As the particle flow decreases with the radius, larger detector elements can be used. This also reduces the amount of read-out electronics. The inner micro strip detectors have a size of $10 \text{ cm} \times 80 \mu\text{m}$, whereas the size of the outer ones is $25 \text{ cm} \times 180 \mu\text{m}$. At the endcaps, two disks of pixel detectors, three smaller and nine larger discs of strip detectors are installed. This allows a reconstruction of charged particle tracks up to $|\eta| < 2.5$. In total, the silicon tracker has a length of 5.8 m with a diameter of 2.5 m. At each bunch crossing, the occupancy of the detectors is at about 2-3% for the inner micro strip detectors and 1% or lower for the other ones. The track resolutions for charged particles of $1 \text{ GeV} < p_T < 10 \text{ GeV}$ and $|\eta| < 1.4$ are typically 1.5% in p_T , 25–90 μm in the transverse and 45–150 μm in the longitudinal impact parameter [20].

Electromagnetic Calorimeter

The electromagnetic calorimeter (ECAL) [21] surrounds the silicon tracker system hermetically and measures the energy of electrons, positrons and photons. Other electromagnetically interacting particles leave tracks but are not fully absorbed in the ECAL. The ECAL consists of several elements, the pseudorapidity interval up to $|\eta| < 1.48$ is covered by the barrel part while the endcap has a coverage of $1.48 < |\eta| < 3$. The ECAL consists of homogeneous lead tungstate (PbWO_4) crystals which is the absorber and scintillator material at the same time. When electromagnetically interacting particles propagate through it, they produce photons through bremsstrahlung, which in turn create electron-positron pairs through pair production. An electromagnetic shower emerges. When the energy of the electrons drops to a critical amount, they mainly excite the atoms, which emit photons with characteristic wavelengths between 360 nm to 570 nm. This process is called scintillation. The photons are then measured with photomultipliers. The number of photons is proportional to the deposited energy of the particle. The radiation length of PbWO_4 is $X_0 = 0.89$ cm. This means that after this distance, the energy of an electron or positron is reduced on average by the factor of $1/e$. Moreover a photon has an average free path length of $9/7 X_0$ before it undergoes a process of pair production. The thickness of the ECAL with 23 cm is equivalent to $25.8 X_0$, therefore all electrons, positrons and photons are absorbed completely.

The ECAL was designed to measure the $H \rightarrow \gamma\gamma$ decay as precise as possible. Therefore, it provides a good energy resolution. To identify the background in this decay channel, a preshower detector is installed between the tracker and the ECAL. This element helps to identify neutral pions that decay into two photons; it also improves the position determination of electrons, positrons and photons.

Hadron Calorimeter

Hadrons propagate through the ECAL losing only a small fraction of their energy. Therefore, a hadron calorimeter (HCAL) [22] surrounds the ECAL, which absorbs all hadrons and measures their energy. The HCAL is important for measuring the properties of hadron jets as well as the total energy of the collisions and the resulting missing transverse energy. Several elements allow to measure forward jets up to $|\eta| = 5.2$.

The HCAL is a sampling calorimeter which exists of alternating absorbing layers and scintillating layers. The absorbing layer consists of brass and has a nuclear interaction length of $\lambda_I = 16.42$ cm. This is the mean distance for a hadronic particle after which it undergoes an inelastic nuclear interaction. Similar to the ECAL, a chain reaction starts and a hadronic shower emerges. As the HCAL is located inside the solenoid, its size is limited. The total absorber material of ECAL and HCAL makes up less than $7 \lambda_I$. For hadrons that are not fully absorbed, an additional outer hadron calorimeter

is installed outside the solenoid. For the scintillating layers, plastic is used. When the hadronic shower hits these layers, the scintillator material gets excited and photons with characteristic wavelengths are emitted. They are guided through optical cables and measured by photodiodes.

Muon Detector

The muon detector [23] is the outermost part of the CMS detector and covers the pseudorapidity interval of $|\eta| < 2.4$ without any gaps. The only particles that reach this part of the detector are neutrinos, which cannot be detected directly with the CMS detector, and muons. Muons have a high mass and are often high energetic. Because of this, they lose only small parts of their energy by propagating through the inner parts and are usually not fully absorbed in the muon detector. Muon identification and position determination is done with gaseous detectors. As they pass through these detectors, muons are ionizing the gas atoms. The resulting electrons and ions can then be measured as an electric current. The muon system consists of an iron return yoke to guide the magnetic field lines of the solenoid through this part of the detector. This creates a magnetic field of 2 T. As a consequence, muons travel a circular path. The momentum can then be determined by measuring the radius of its track.

Trigger System

The LHC has a bunch crossing rate of about 40 MHz at each interaction point. This huge amount of data makes it impossible to store every event. Since interesting processes typically have low rates, it is sufficient for most analyses to store only a small fraction of promising events. To perform this data reduction, a trigger system [24] consisting of two filtering steps is used at the CMS detector.

The first filter is called the Level-1 (L1) trigger and is realized by hardware electronics. It decides in a few μs whether it is worth keeping an event. In this short time, only a part of the information from the calorimeters and the muon chamber can be used. The trigger checks if the information is consistent with a physics object like an electron or a muon. This procedure reduces the event rate to less than 100 MHz.

The second filter step is done by the High-Level Trigger (HLT). The HLT is realized with software tools where complex calculations and a more precise reconstruction can be done. In addition, the information of the tracking system is now included. This trigger reduces the event rate further to less than 1.5 KHz. These events are then stored offline for more detailed analyses.

3 Simulation and Reconstuction of Events

All the available information about a particle collision is the measured deposited energy of each detector component. Together with some results of the online reconstruction e.g. trigger signals, this information is stored for further reconstruction. In the offline reconstruction, physical objects like electrons and hadrons are reconstructed. To compare the measured data with the theory prediction, or to investigate the efficiencies and robustness of algorithms, a simulation is needed. In this chapter the simulation process is described followed by an explanation of the reconstruction process. Additionally, methods to correct differences between the measured data, shortly denoted as data, and the simulation are summarized.

3.1 Event Simulation

The simulation is done using as much theory input as possible. In the same order as the physical process, from the proton-proton scattering to the outgoing particles, one propagates the simulation. Due to insufficient theoretical descriptions, phenomenological models are used when necessary. After the outgoing particles are generated, the interaction with the detector materials is simulated. The emulation for the readout electronics generates signals that have the same signature as the data.

3.1.1 Proton-Proton Scattering

A proton-proton scattering process leads to the production of a specific final state. The cross section is proportional to the probability to produce this final state. The determination of the cross section σ is done in multiple steps, these steps are outlined in figure 3.1 and explained in the following.

Parton Distribution Functions

As described in section 1.3, the proton is not an elementary particle but consists of so called partons. The valence quarks (uud) determine the quantum numbers of the proton. Apart from these, also gluons exist inside of protons as they are the exchange particles of the strong interaction. The gluons can split into virtual quark-antiquark-pairs; these quarks are the so called sea quarks. In a deep inelastic proton-proton scattering, two partons from two colliding protons interact with each other. Each parton carries a momentum fraction x of the proton. The probability that a specific parton

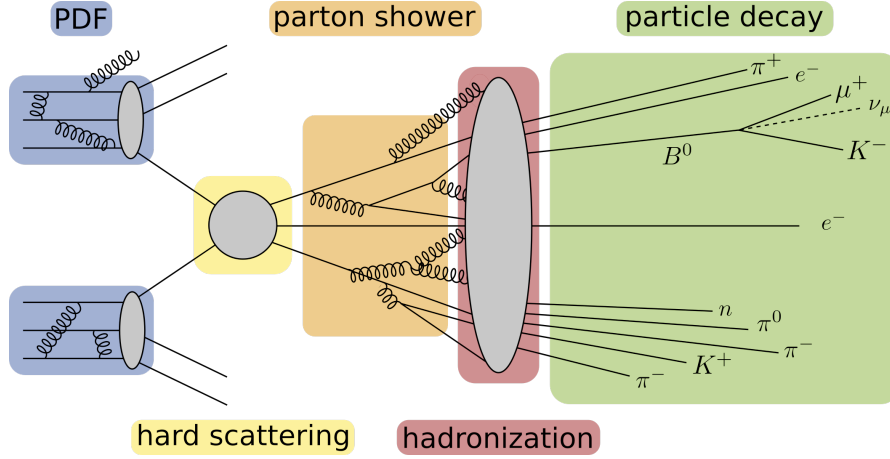


Figure 3.1: Proton-proton scattering process: The parton distribution functions (PDF) describe how the constituents of the proton are distributed. The deep inelastic scattering process of two elementary particles is calculated in the hard scattering. The subsequent processes are the parton shower, the hadronization and the particle decay.

interacts, depends on x and the absolute transferred momentum Q . It is given by a parton distribution function (PDF). As the PDFs cannot be computed by perturbation theory, they are determined by global fits to data from electron-proton scattering, fixed target experiments or hadron colliders. A classic approach is a polynomial ansatz for a specific PDF [25]. A more modern approach is using a three-layer feed-forward neural network with 37 parameters [26]. One can define a factorization scale μ and absorb low energetic gluons into the PDF, thereby infrared divergences are avoided. The use of the Dokshitzer-Gribov-Lipatov-Altarelli-Parisi (DGLAP)-equations [27–29] allows the evolution of gluon and quark PDFs to a desired value of Q . The PDFs for two different values of μ are shown in figure 3.2.

Hard Scattering Process

The differential cross section $d\sigma$ for the hard scattering process is weighted with the PDFs $f_{i,j}$, which give the probabilities to find certain partons with certain momentum fractions x and y of the colliding protons. It is given by

$$d\sigma = \frac{1}{3} \sum_{i,j} f_i(x, \mu^2) f_j(y, \mu^2) \frac{1}{2(xy s)^2} |\mathcal{M}_{i,j \rightarrow \text{final}}| \frac{d \cos \theta d\phi dx dy}{8(2\pi)^2} , \quad (3.1)$$

with the square of the center-of-mass energy s and the transition amplitude $\mathcal{M}_{i,j \rightarrow \text{final}}$ for an initial state to a final state, the so called matrix element. $\mathcal{M}_{i,j \rightarrow \text{final}}$ is obtained by summing up all Feynman diagrams for a specific process in a perturbation series. This is possible because the momentum of the partons is in a regime where $\alpha_s \ll 1$ and convergence of the series is given. Often it is sufficient to take the leading order

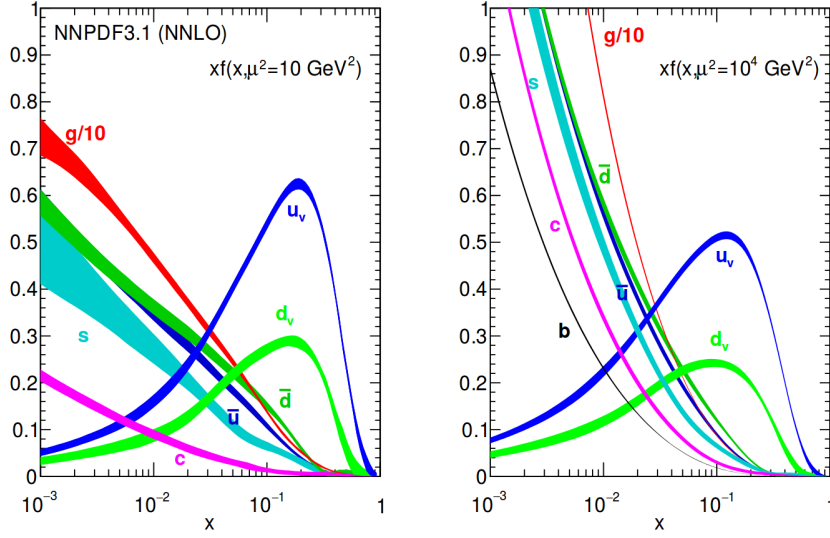


Figure 3.2: Neural Network Parton Distribution Functions (NNPDFs). Shown are the PDFs of gluons, valence quarks and sea quarks for different momentum fractions x of the proton momentum. At lower values of the factorization scale (left) the valence quarks carry the majority of the proton momentum. At higher values of factorization scale (right) gluons carry the dominant part. The PDFs are computed using the DGLAP-equations in next-to-next-to-leading-order (NNLO). Source: [26]

(LO) Feynman diagrams, which are the ones with the minimal amount of vertices that are needed for a specific process. Including next-to-leading order (NLO) Feynman diagrams improves the accuracy of the simulation but is more complex and computationally intensive in generation.

The simulation is done with the help of Monte Carlo (MC) methods [30]. MC methods can be used for numerical integration of higher dimensional phase spaces. In principle, to obtain one simulated event that obeys equation 3.1, the following can be done [31, p. 48]. Uniform distributed random numbers (θ , ϕ , x , y , i , j) are taken in the corresponding value ranges and $\langle d\sigma \rangle$ is calculated at one specific phase space point. The event is further processed if another uniform distributed random number g in the range $0 < g < d\sigma_{\max}$ is lower than $\langle d\sigma \rangle$, where $d\sigma_{\max}$ is the supremum of $d\sigma$. Otherwise it is discarded. Repeating this procedure equals an integration of the total phase space of equation 3.1.

Parton Shower, Hadronization and Particle Decay

In addition to the final state particles of the hard scattering process, gluons and photons can be produced in initial or final state radiation. Gluons can radiate off another gluon or split into quark-antiquark pairs; photons can split into fermion pairs. These processes

can recur and are described by the DGLAP-equations. A so called parton shower emerges. Because of the confinement (section 1.3), the partons form hadrons. This process cannot be described in a perturbation series as the assumption of $\alpha_s \ll 1$ does not hold for lower energies of the partons. The effect is described by phenomenological models like the Lund string model [32]. The hadrons can further decay, as most of them are unstable.

Underlying Event and Pileup

Besides the hard scattering process of two partons, the remaining partons can interact themselves, producing additional signals in the detector at the same time. This effect is called underlying event. In the same bunch crossing, several other proton pairs are interacting with each other, this is called in-time pileup. Out-of-time pileup considers previous bunch crossings. The pileup distribution depends on the collider's properties. The more protons are in one bunch and the more focused the beams are, the more pileup events happen. In the data recorded in 2016, there were in average about 27 pileup events per bunch crossing in the CMS detector [15]. The pileup events can be simulated as additional random scatterings and merged with the main event.

3.1.2 Simulation Tools

In the event simulation, several tools are used for the different steps. For the matrix element, one common generator is MADGRAPH5 [33]. It sums up all Feynman diagrams in LO for given initial and final state particles. The matrix element is generated but not evaluated yet. The evaluation is done with the MADEVENT package [34] at given phase space points with the MC method and the cross section is calculated. For this reason a simulated event is often called MC event.

MADGRAPH5_AMC@NLO [35] is a further developement of MADGRAPH5, combining the features of MadGraph5 with the ones of AMC@NLO, where AMC@NLO can generate matrix elements at NLO in QCD. These tools also provide different parton-shower matching schemes to combine the final state particles with the parton shower. The parton shower itself has to be added with a parton shower simulation tool. As explained earlier, the parton shower can cause additional gluons in the final state, but such gluons can also be produced in NLO matrix element generators. To avoid double counting, the parts of the parton shower that match the NLO Feynman diagrams are subtracted. This is implemented by using negative event weights.

Another matrix element generator that includes Feynman diagrams up to NLO in QCD is POWHEG [36–38]. POWHEG uses a different method than AMC@NLO to avoid double counting and negative event weights. It demands a parton shower simulation that

generates the emission with the highest p_T first and then corrects the NLO emission.

PYTHIA [39] is a general purpose tool for the simulation of the hard process as well as the parton shower, the hadronization, the particle decay and the underlying event. The hard process is only calculated in LO, but PYTHIA provides several interfaces to external programs. Therefore, the matrix element generators explained above can be used in combination with PYTHIA.

To be able to compare simulation with data, the detector response has to be simulated as well. This is done with the GEANT4[40] toolkit. In GEANT4, the different materials of the detector system, including dead materials, are modeled. The particles are propagated through these materials while taking into account the effect of the magnetic field on charged particles. Interactions like bremsstrahlung, multiple scattering and photon pair-production are simulated. The generated signals in the active detector materials are taken as input for emulators of readout electronics and trigger systems. After this step, the detector information of a simulated event is in the same manner as for data. To get the best comparison between data and simulation, the event reconstruction of the detector signals is done in the same way as for data. This will be subject of the next section.

3.2 Event Reconstruction

In this chapter, the aim to reconstruct physics objects out of the measured or simulated detector information is described. In a first step of the event reconstruction, particle tracks and the corresponding interaction points, called vertices, are reconstructed. In several subsequent steps, the identification and reconstruction of particles that penetrate the detector is done. These are electrons, muons, photons, charged hadrons, neutral hadrons and neutrinos. Afterwards, adjacent particles are bundled to so called jets.

3.2.1 Tracks and Vertices

Tracks of charged particles are reconstructed from the tracker information. Therefore, tracker hits are created from clustering signals of the silicon pixel and strip detectors. The tracks are produced using an iterative track finding algorithm. From tracker hits in the inside of the detector, where the occupancy of each detector element is low, tracker seeds are created. The track finding is performed with the Kalman filter approach [41]. In the first iteration step, the seeds have tight constraints. This gives only a moderate efficiency, but also a minimal fake rate. The energy deposits of the corresponding hits are then removed, which simplifies the track finding problem. At the next iteration step with looser seeding criteria, a higher efficiency is achieved. Repeating this, one gets an

efficiency to find isolated muons of 99.5% and more than 90% to find charged hadrons while the fake rate is of about 1% [42]. Each track has several quality criteria like the number of hits, a normalized χ^2 value and the compatibility to a certain vertex.

Because of pileup, there exists more than one vertex for each event. The vertex from where the hard scattering originates is the primary vertex (PI), it has to be identified in order to remove the tracks originating from pileup. The vertex reconstruction starts with a set of possible vertices and each track is extrapolated to a point z_i on the beam line. For each track and vertex pair a probability c_{ik} is estimated that the track is originating from the regarded vertex. A χ^2 function, weighted with these probabilities

$$\chi^2 = \sum_{i,k} c_{ik} \frac{(z_i - z_k)^2}{\sigma_i^2} \quad , \quad (3.2)$$

is minimized to determine the exact position of the vertices z_k . As the probability depends on the vertex position, an iterative procedure is used. After a final selection, a set of vertices is obtained. The primary vertex is selected as the one with the highest sum of p_T^2 of its tracks.

3.2.2 Particle Reconstruction

For reconstructing particles as precise as possible, the combination of all subdetectors is used within the particle flow (PF) algorithm [42, 43]. The identification of PF candidates, namely electrons, muons, photons, charged hadrons and neutral hadrons is explained in the following.

Calorimeter Clustering

The tracker system gives the most precise information about the position and through the curvature of the track, it gives also the best determination of the momentum of charged particles. The calorimeters support the determination of the energy and direction of charged particles in cases of high p_T or low quality tracks. Moreover the energy and direction of photons and neutral hadrons is measured while the contributions of charged particles are separated. This is done in the calorimeter clustering process, where each calorimeter element, the ECAL barrel, the ECAL endcap and the several HCAL elements, are handled separately. The process starts on calorimeter cells with a local maximum of deposited energy. Then, adjacent cells are combined, if their deposited energy exceeds a certain threshold. For each calorimeter element, one can have several calorimeter clusters. Together with the charged-particle tracks and the muon tracks, PF elements can be constructed.

Electrons

An electron is identified if a charged-particle track matches an ECAL cluster with a typical electron shower profile. A fit starting with a track seed is combining both signals and the momentum is determined. A complimentary algorithm for electrons with high p_T starts with taking a seed from the ECAL. In this algorithm, several ECAL clusters are combined to so called ECAL superclusters which have a characteristic form for electrons ???. The quality of an electron fit is described by several quantities like the matching of the track to the cluster or the distance of the track to the primary vertex. A tight requirement on these quantities can lower the fake rate.

Muons

For the muon reconstruction, tracks in the muon chamber are computed in a similar way as in the tracker. These muon tracks are propagated towards the beam line and combined if possible with the most suitable track of the inner tracker. A refit using the information of both sub detectors gives an improved muon track. A reconstruction process for low p_T muons that do not create a muon track, starts with propagating an inner tracker track and tries to find a minimal muon information in the muon tracker. Again one can choose looser or tighter demands on muon tracks afterwards to further reduce the fake rate.

Hadrons and Photons

For the reconstruction of charged hadrons, the energy depositions of clusters in ECAL and HCAL are compared to the momentum of the charged particle tracks. In case of an agreement with one or more tracks, a combined fit is performed. In case of disagreement, several scenarios including additional neutral hadrons or photons are considered. After all charged-particle tracks are matched to PF elements, their energy deposits are removed and neutral particles are identified. Neutral particles do not produce any hits in the tracker, therefore they have only bad direction information and an assignment to a vertex is not possible. Because of this, also neutral particles coming from pileup are included.

3.2.3 Jet Reconstruction

A jet is a particle shower of hadrons, leptons and photons that originates in the most cases from a single particle. There are different ways to reconstruct a jet, the default method used in the CMS experiment collects several nearby PF elements in a cone up to a given size in a sequential recombination process. This is done with the anti- k_t jet clustering algorithm [44].

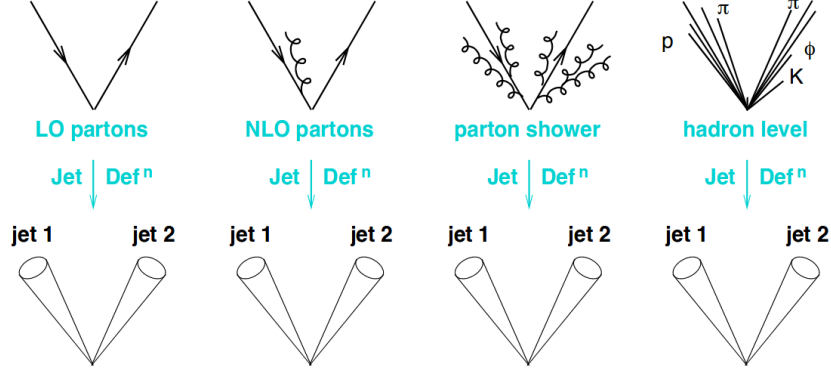


Figure 3.3: Infrared and collinear safety. A jet can be defined on different levels. In a theoretical description, it originates from one single particle and can therefore be defined as this particle. In perturbation theory, several vector bosons can be radiated in addition. On the detector level, the jet is defined by color neutral objects. With an infrared and collinear safe jet clustering algorithm, the resulting jets are independent of the level of definition. Source: [45]

An advantage of this algorithm is its infrared and collinear safety. This means, that the same jet is found with and without a soft or collinear radiation, for example a gluon radiation or a gluon splitting. Therefore, this algorithm is independent of the technical level of the jet definition as shown in figure 3.3. For the clustering process, a distance measure d_{ij} between two particles i and j is defined as

$$d_{ij} = \min(k_{t,i}^{-2}, k_{t,j}^{-2}) \frac{\Delta R_{ij}^2}{R^2} , \quad (3.3)$$

where $k_{t,i}$ is the momentum of a particle, $R^2 = (y^2 + \phi^2)$ is the radius in the y - ϕ plane with the rapidity y and $\Delta R_{ij}^2 = (y_i - y_j)^2 + (\phi_i - \phi_j)^2$ is a radial distance of two particles. A distance measure d_{iB} between a particle i and the beam axis is defined as

$$d_{iB} = k_{t,i}^{-2} . \quad (3.4)$$

For the general jet reconstruction of data taken from run 2 of the CMS experiment, a radius of $R = 0.4$ was chosen. In the sequential recombination algorithm, all possible d_{ij} are computed and two particles with the smallest d_{ij} are combined to new pseudo-particles. This is repeated until d_{iB} is the smallest distance. The remaining pseudo-particles become then the jets. In this way, the hardest particles (the ones with the highest p_T) are combined with nearby soft particles (the ones with low p_T). If the cones of two hard particles overlap, they share the softest particles and create two separate jets with partial cones.

3.2.4 Lepton Isolation

Muons and electrons often appear inside of QCD jets originating from b , c or s quark decays. Isolated leptons on the other hand appear often in rare processes, they are

good indicators for leptonic top quark decays ($t \rightarrow b + \nu_l + l^+$). Therefore an isolation criteria is useful to find events with bottom quarks without directly looking at them. An isolation can be described using the momenta and energies p_T^{charged} , E_T^{neutral} and E_T^γ of the PF elements for the charged particles, the neutral particles and the photons respectively inside a cone of

$$\Delta R = \sqrt{\Delta\eta^2 + \Delta\phi^2} = 0.4 \quad . \quad (3.5)$$

Since the tauon decays in the beam line, only for an electron or a muon with its momentum p_T^l an isolation can be defined as

$$I_l^{\text{PF}/\Delta\beta} = \frac{\sum p_T^{\text{charged}} + \max\left(0.0, E_T^{\text{neutral}} + \sum E_T^\gamma - 0.5 \sum E_T^{\text{charged}}(\text{PU})\right)}{p_T^l} \quad . \quad (3.6)$$

The so called $\Delta\beta$ correction is included where contributions of the neutral particles from pileup interactions are subtracted. They are estimated as half of the energy coming from charged particles in pileup $\sum E_T^{\text{charged}}(\text{PU})$. A lower value of $I_l^{\text{PF}/\Delta\beta}$ corresponds to a more isolated electron or muon.

3.2.5 Reconstruction of Neutrinos and Leptonically Decaying W Bosons

At the LHC, the transverse momentum $\vec{p}_T = (p_x, p_y)^T$ of the colliding particles is zero, therefore the sum of the \vec{p}_T of all generated particles after the collision has to be zero as well. As neutrinos cannot be measured within the CMS detector the missing part of the transverse momentum can be attributed to neutrinos. For historical reasons and as the momentum is equal to the energy in the relativistic regime using natural units, the missing part is denoted as the missing transverse energy

$$\vec{E}_T = - \sum_i \vec{p}_{T,i} \quad . \quad (3.7)$$

Decays of the W boson into a charged lepton and a neutrino are called leptonic W boson decays. With the measured \vec{E}_T and the momentum of the lepton, the transverse mass of the W boson $m_{T,W}$ can be reconstructed

$$m_{T,W} = \sqrt{(\vec{p}_{T,l} + \vec{p}_{T,\nu})^2 - (p_{x,l} + p_{x,\nu})^2 - (p_{y,l} + p_{y,\nu})^2} \quad , \quad (3.8)$$

with $\vec{p}_{T,\nu}$ the neutrino momentum set to \vec{E}_T . This method allows to reconstruct the W boson if there are no additional W bosons or leptons in the event.

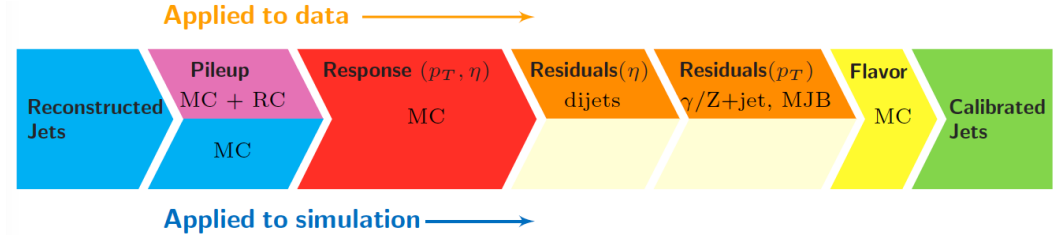


Figure 3.4: Jet Energy Corrections. In the first step, based on information from simulation, energy contributions from pileup and electronics noise are removed. In data, the residual differences between data and simulation as a function of η is flattened as well. Next, the response of the detector is corrected. Therefore, the residual differences in simulation between the reconstructed p_T and η and the corresponding true values are flattened out. The resulting correction is performed in data and simulation. A further step is done only on data, here small residual differences between data and simulation in p_T and η are corrected. An optional last step is the correction in dependence of the flavor of the jet using the truth information from the simulation. Source: [47]

3.3 Differences between Data and Simulation

As mentioned before, the simulation is not exact since in each step smaller or larger inaccuracies are made. These differences lead to discrepancies between data and simulation after the reconstruction. Several steps can be done to improve the agreement afterwards, these are explained in the following.

3.3.1 Jet Energy Corrections

In general, the measured jet energy does not match the true parton energy, where the jet originated from. One reason is pileup and underlying events. Due to the tracks, the charged particles from pileup can be removed reasonably well but the neutral particles not. Also neutrinos are not measured and they appear also in jets, therefore, their energy is missing. Moreover, there is a disagreement in data compared to simulation due to inaccuracies in the simulation. With jet energy corrections (JEC) [46], these differences are mitigated in several steps with some variation for data and simulation. These steps are outlined in figure 3.4.

3.3.2 Pileup Reweighting

The simulation is done before or during the actual data taking period, therefore a pileup distribution has to be assumed in the simulation. To correct most of the disagreements caused by the different pileup interactions, the simulated events can be assigned with a weight in a way that the pileup distributions in data and simulation are equal. The pileup weights are given by

$$w_{PU,i} = \frac{n_{data,i}}{n_{MC,i}} \quad , \quad (3.9)$$

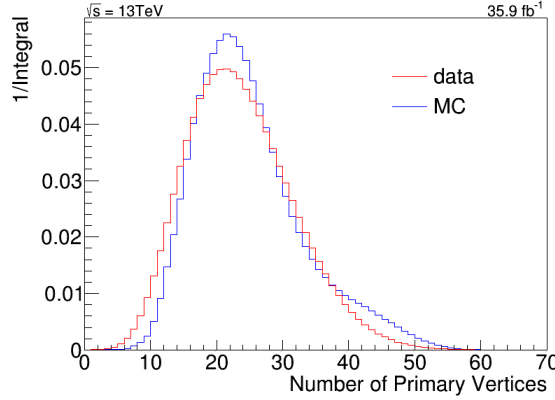


Figure 3.5: Pileup distribution for data and simulation. Shown is a normalized histogram of the events as a function of the number of primary vertices. In each bin where the data is above the simulation, the simulation gets up-weighted; when the data is lower than the simulation, the simulation gets down-weighted.

with the number of data events $n_{\text{data},i}$ and the number of simulated events $n_{\text{MC},i}$ in dependency of the number of primary vertices i . The pileup distributions for the 2016 data and simulation are shown in figure 3.5.

3.3.3 Trigger and Lepton Efficiencies

The efficiency of the triggers is also different in data and simulation. In order to mitigate this discrepancy, scalefactors are used to correct the simulation

$$sf_{\text{trigger}} = \frac{\epsilon_{\text{data}}(p_T, \eta, \dots)}{\epsilon_{\text{MC}}(p_T, \eta, \dots)} \quad , \quad (3.10)$$

where the efficiencies $\epsilon_{\text{data},\text{MC}}$ are functions of different quantities. For example a muon trigger has a strong dependency on the p_T of the corresponding muon.

Lepton scalefactors are needed whenever a selection of leptons is performed as the efficiencies of the identification of a lepton are different in data and simulation. The efficiencies are moreover dependent of the selection of the lepton, they are often separated for different parts of the selection (for example for the tracker selection and the isolation criteria) and can be combined. For some selection criteria, scalefactors are already measured.

4 Artificial Neural Networks

Artificial neural networks (ANNs) are a class of algorithms inspired by the biological brain and widely used in machine learning and pattern recognition. In this thesis, ANNs were used for a multivariate classification task, this problem is outlined in section 4.1. ANNs consist of several artificial neurons, in section 4.2, the functionality of a single artificial neuron is explained. In section 4.3 the kind of ANN, used in this thesis is described. The core subject of this thesis is the domain adaptation as an extension of ANNs to improve its data to simulation agreement. The domain adaptation is described in section 4.4. A comprehensive introduction to machine learning techniques is provided in [48] or [49].

4.1 Multivariate Classification

In a multivariate classification task one tries to assign a data point with characteristic variables to one of several classes. If the distributions of the characteristic variables overlap for different classes, the data points are not completely separable. That means only the probability that one data point belongs to a certain class can be obtained. Consider two classes, denoted as class one and class two. From the output of a classifier, a so called test statistic can be constructed to separate the two classes. The test statistic can be chosen on a fixed value, all data points with a test statistic higher (lower) than this value are said to be class one (two). Due to the fact that the classes are not completely separable, not all data points can be classified correctly. Consider class one (two) as the signal (background) class. The efficiency is the fraction of data points of class one that are correctly classified. The false positive rate (FPR) is the fraction of class two data points that are mistakenly classified as class one. The closer the value of the efficiency is to 1 and the closer the FPR is to 0, the better the classifier.

The multivariate classification in this thesis is done with ANNs by constructing a function that maps an input vector \vec{x} of characteristic variables (features) to an output vector \vec{y} that describes the probabilities of the classes. An ANN has various free parameters. In a process called training, these parameters are adjusted to get a function with high efficiency and low FPR. The process of training is done using so called training samples. For each data point of these samples the corresponding class is known, therefore the \vec{y} is labeled.

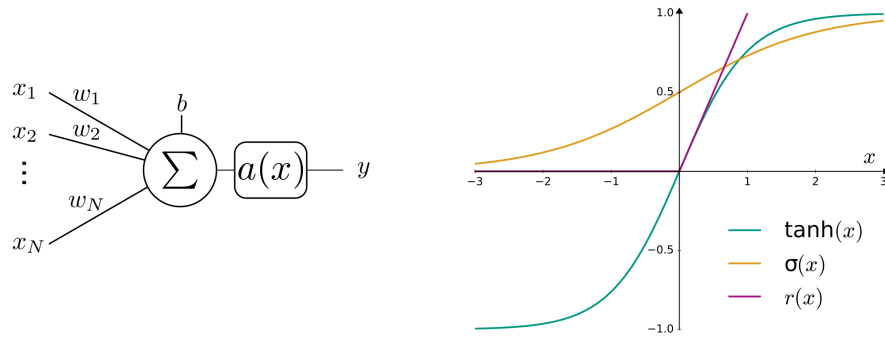


Figure 4.1: Perceptron (left). An ANN consisting of one neuron as shown on the left is called perceptron. Several neurons can be connected to a multilayer perceptron.
Activation functions (right). Three common activation functions are shown.

4.2 Perceptron

One single neuron can be used to construct simple mathematical functions. The first was introduced by rosenblatt and is the rosenblatt perceptron. The computation is done by taking all input values x_i , multiplying them with adaptable weights w_i and adding an adaptable bias b . The result is mapped by an activation function $a(x)$ to one single output value

$$y = a \left(\sum_i x_i w_i + b \right) . \quad (4.1)$$

Commonly used activation functions are the sigmoid function

$$\sigma(x) = \frac{1}{1 + \exp(-x)} , \quad (4.2)$$

and the hyperbolic tangent $\tanh(x)$. However, these are problematic when applied to more complex neural networks [50], therefore the rectifier activation function (ReLU) [51]

$$r(x) = \max(0, x) , \quad (4.3)$$

is preferred for them. The perceptron and the mentioned activation functions are shown in figure 4.1. This perceptron is able to solve linearly separable problems. For more complex problems, an ANN consisting of several neurons can be used.

4.3 Multilayer Perceptron

A multilayer perceptron (MLP) is the standard type of ANN where several neurons are aligned in layers. An MLP can be further distinguished according to its architecture. The architecture describes the configuration of the MLP, it determines the number of layers, the number of neurons of each layer, how the neurons are connected among each

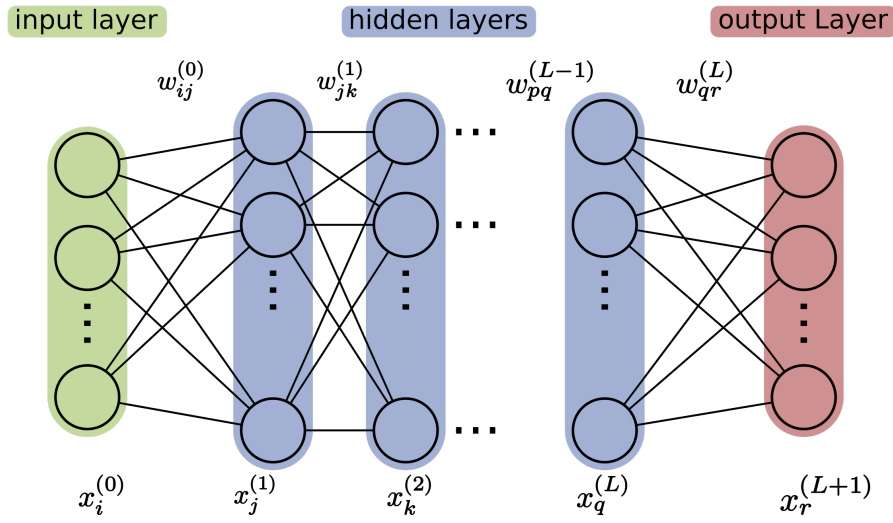


Figure 4.2: A multilayer perceptron with $L + 2$ fully connected, feed-forward layers. The first layer (green) is called the input layer and the last layer (red) is called the output layer. In between there are L layers (blue), referred to as hidden layers. The circles represent the neurons while the connections between them are the edges. Each edge has a weight w_{ij} where i and j indicate its source and target neuron respectively. Each neuron has a bias and delivers a value x_i , the biases are not shown.

other and the activation functions used. With the use of an appropriate architecture and training procedure, the output values can take on meaningful values.

4.3.1 Fully Connected Feed-Forward Networks

If the MLP has a feed-forward structure, the connections between neurons do not form circles. If it is additionally fully connected, each neuron takes input values from all neurons of the preceding layer while the output values are connected to all neurons of the following layer. An example of a fully connected feed-forward MLP is shown in figure 4.2. The number of input and output neurons is determined by the number of features and classes of the specific multivariate classification task. The number of hidden layers and the number of neurons in each hidden layer can be chosen freely. These determine the number of free parameters, the biases and the weights. MLPs with more than one hidden layer are called deep neural networks (DNN). The output of an arbitrary neuron can be calculated recursively by

$$x_i^{(k+1)} = a_i^{(k+1)} \left(\sum_{j=1}^{N^{(k)}} x_j^{(k)} w_{ij}^{(k)} + b_i^{(k)} \right) , \quad (4.4)$$

where k denotes the layer and $N^{(k)}$ the number of neurons in this layer. As an activation function in the final layer, it is reasonable to use a softmax function

$$y(\vec{x})_k = \frac{\exp(x_k)}{\sum_{j=1}^N \exp(x_j)} \quad , \quad (4.5)$$

to normalize the sum of all output values to 1. With the presented architecture, the input values can be propagated through each layer and the output values can be calculated. This process is called forward propagation.

4.3.2 Training

The training is done by minimizing a function that takes the computed output values of the ANN for given training samples and measures the deviation from the corresponding labels. This function is called the error function, it calculates one single value, the so called loss.

Consider a binary classification problem where the ANN has one single output neuron with a sigmoid activation function. Moreover each element of the training sample has a label l with $l = 1$ for one class and $l = 0$ for the other. An appropriate error function is the binary cross-entropy

$$E(\vec{x}; \vec{w}, \vec{b}) = - \sum_{n=1}^N (l_n \ln y_n + (1 - l_n) \ln(1 - y_n)) \quad , \quad (4.6)$$

where n indicates the elements of the training sample and $y_n = y_n(\vec{x}; \vec{w}, \vec{b})$ the computed output value of the ANN. One can show that the binary cross-entropy together with the sigmoid activation function (equation 4.2) is equivalent to the negative log likelihood function [48, p. 234-236], for this kind of problem.

Now assume a multiclass classification problem where the labels \vec{l} are in the 1-of-K coding scheme, which means that each entry is binary ($l_k \in \{0, 1\}$) and is 1 for the corresponding class and 0 otherwise. A good choice for the error function is the categorical cross-entropy

$$E(\vec{x}; \vec{w}, \vec{b}) = - \sum_{n=1}^N \sum_{k=1}^K l_{nk} \ln y_{nk} \quad , \quad (4.7)$$

where k indicates the classes. Given that the softmax function (equation 4.5) is used in the output layer, one can show in a similar way to the before mentioned binary cross-entropy that one obtains the negative log likelihood function [48, p. 234-236]. This allows to interpret the calculated output values as probabilities for the different classes. For this reason, the calculated output values can be called predictions.

The high dimensional error functions are in general too complex to compute the position of the global minimum. Typically, the global minimum is not found and also not intended to be found. Instead it is aimed for a local minimum that has a low value and is flat¹. This is done by a minimizing algorithm, the so called optimizer.

A simple optimizer is the gradient descent. The error function is calculated for all training samples and the gradient is computed. The free parameters $\vec{\theta} = (\vec{b}, \vec{w})^T$ are then updated against the gradient

$$\vec{\theta} \leftarrow \vec{\theta} - \eta \frac{\partial E}{\partial \vec{\theta}} \quad , \quad (4.8)$$

with a so called hyperparameter, the learning rate η that determines the step size. This is repeated several epochs. An epoch is defined as a full training cycle, which is when every element in the sample is processed. When the gradient is zero, a minimum or a saddle point is found. To minimize the risk of getting stuck in a saddle point, the training samples are split into batches and the updates in equation 4.8 are done for each batch. After each epoch, the samples are shuffled. This is called stochastic gradient descent (SGD) and has also the advantage that broader minimums are found, which are more generalizable.

Widely used is the ADAM optimizer [52] that takes previous gradients into account and leads to a faster convergence with less fine tuning of free hyperparameters. In this case, the updates are given by

$$\vec{\theta} \leftarrow \vec{\theta} - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \vec{m} \quad , \quad (4.9)$$

with

$$\begin{aligned} \vec{m} &= \frac{\vec{m}}{1 - \beta_1} & \vec{m} &\leftarrow \beta_1 \vec{m} + (1 - \beta_1) \frac{\partial E}{\partial \vec{\theta}} \quad , \\ \hat{v} &= \frac{v}{1 - \beta_2} & v &\leftarrow \beta_1 v + (1 - \beta_1) \left(\frac{\partial E}{\partial \vec{\theta}} \right)^2 \quad . \end{aligned} \quad (4.10)$$

The hyperparameters are the learning rate η , a damping parameter ϵ and the decay rates β_1 and β_2 . The decay rates β_1 and β_2 determine how much the past gradients effect the current batch update. In practice, the default values of $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$ proposed by [52] can be used and only an appropriate η has to be chosen. When a stationary point is obtained, a fine tuning of the optimizer can be done by a method called reduce-on-plateau. In this method, the learning rate of the optimizer is reduced if after a certain number of epochs, the training loss has not decreased.

¹The flatness leads to better generalizability as for small deviations of the minimum the loss value is still low.

The updating of the free parameters $\vec{\theta}$ is performed with the help of error backpropagation. Starting at the output layer, the chain rule can be applied:

$$\frac{\partial E}{\partial \theta_{ji}} = \frac{\partial E}{\partial a_j} \frac{\partial a_j}{\partial \theta_{ji}} \quad , \quad (4.11)$$

where a_j is the activation function of the corresponding output neuron. By re-applying the chain rule, the gradients for the free parameters in the hidden layers can be calculated.

4.3.3 Regularization

Regularization methods are used to prevent the training process from overfitting. Overfitting is an effect where the ANN approximates the training data too close and gets sensitive to statistical fluctuations. As a consequence it loses the generalizability to make predictions for additional data that origins from the same distribution. There are several different methods, the ones used in this thesis are explained in the following.

A common regularization method in the field of machine learning is called early stopping [53]. The available data is split into two parts, the training dataset and the validation dataset. After each epoch of training the ANN on the training dataset, the so called validation loss is calculated by evaluating the loss function on the validation dataset. The ANN parameters, for which the validation loss is minimum, are stored. Even if the training loss further decreases in later epochs, the stored parameters for which the validation loss is minimal are applied after completion of training.

Additionally one can monitor the validation loss instead of the training loss for the above explained reduce-on-plateau method.

Another regularization method used in ANNs is dropout [54]. During the training, randomly selected hidden neurons are deactivated for each batch update. In this way, one single neuron does not carry the total information for a certain prediction and the ANN spreads the information over several neurons. As a result the ANN is less able to memorize single samples and becomes more generalizable. The fraction of neurons that are deactivated in each batch update is given by the dropout rate and can be chosen freely.

A regularization method that has a similar effect is the use of batch normalization layers [55]. They normalize each neurons output value by subtracting the mean μ_B and dividing by the standard deviation σ_B^2 of the corresponding batch B . Afterwards, it multiplies the output by an adjustable scale factor γ and adds another adjustable shift

factor β . The output is then given by

$$y_i = \gamma \frac{x_i - \mu_B}{\sqrt{\sigma_B^2 + \epsilon}} + \beta \quad , \quad (4.12)$$

where ϵ is a small constant value that prevents the fraction of a division by zero. With this it is possible to scale and shift the output value by only changing two values. Without batch normalization, several weights would have to be adjusted to achieve the same. It prevents individual neurons from becoming too important if there is no general gain for the optimizer. Moreover, it speeds up the training since the input values are more in the scope of the common activation functions.

4.4 Domain Adaptation in Deep Neural Networks

For the training of a DNN, a huge set of labeled samples is needed, but often not available. The DNN is therefore trained on a similar, but differently distributed set of samples where labels are known, the so called source domain. When applied to the unlabeled set of samples, the so called target domain, it results in worse performance and reliability for the predictions. The objective of unsupervised domain adaptation is to transfer the target domain and the source domain into one common domain invariant feature space. Such a feature space is exemplified with and without domain adaptation in figure 4.3.

Various methods for unsupervised domain adaptation methods have been proposed [56]. For example re-weighting the source domain [57] or using an explicit feature space transformation [58]. In this thesis, two promising methods were tested on a multiclass classification task to mitigate the difference between simulated events and measured data, the source and target domains respectively. Compared to the previous mentioned methods, the following ones can be implemented straightforwardly to complement the existing DNN. They both aim at reducing the domain discrepancy in a latent feature space, but they use different mechanisms. The first method, the moments method, is realized through a modified loss function that punishes the domain discrepancy. The second method, the domain-adversarial method, uses an extension of the DNN via additional network components to implicitly achieve this goal by backpropagation. Both methods are explained in the following.

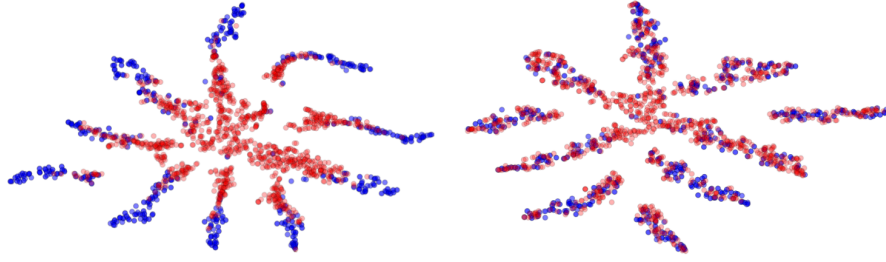


Figure 4.3: The effect of domain adaptation. The blue (red) points represent samples from the source (target) domain. Each point represents the position of an element in a two-dimensional feature space. Shown are features from the last hidden layer of a DNN classifier. In the left plot no domain adaptation is performed. The source samples form separated groups while the target samples are more centered and less separated. In the right plot, the domain adaptation is used during the training. Source and target samples cover the same areas, the predictions of target samples are more reliable. Source: [59]

4.4.1 Moments Method

Two distributions are the same if and only if the moments of these distributions are identical. The first moment of a distribution is the mean

$$\mu = E[X] = \frac{1}{n} \sum_{\{x \in X\}} x \quad , \quad (4.13)$$

where $x \in X$ are data points from the distribution X and n is the number of all data points. Central moments are defined as

$$\mu^{(k)} = \frac{1}{n} \sum_{\{x \in X\}} E \left[(x - E[X])^k \right] \quad , \quad (4.14)$$

where (k) denotes the order of the moment. The second central moment is the variance. Higher central moments are normalized by the standard deviation to the power of k

$$\hat{\mu}^{(k)} = \frac{\mu^{(k)}}{\sigma^k} \quad . \quad (4.15)$$

These moments can be used to measure the domain discrepancy in certain features. A simple measure is the maximum mean discrepancy (MMD) [60]

$$\text{MMD}_i(X_S, X_T) = |\mu_i(X_S) - \mu_i(X_T)| \quad , \quad (4.16)$$

where i denotes the feature and S (T) denotes the source (target) domain. The MMD is used in many domain adaptation approaches, for example [61]. It can be used as an additional term in the loss function to construct a domain invariant feature space. The resulting loss function is given by

$$L = L_C(X_S, y) + \lambda \sum_i \text{MMD}_i^2(X_S, X_T) \quad , \quad (4.17)$$

where the $L_C(X_S, y)$ is the classification loss with the labels y of the samples from the source domain. The hyperparameter λ determines how strong the domain discrepancy is punished. This approach is generic, one can use any features, for example the ones of some hidden layer or the output layer. The training is performed with samples from both domains at the same time while only samples from the source domain can be used for the training of the classification.

A better measure for the domain discrepancy includes higher central moments. The loss can be defined as

$$L = L_C(X_S, y) + \alpha \sum_i \text{MMD}_i^2(X_S, X_T) + \beta \sum_i \left(\mu_i^{(2)}(X_S) - \mu_i^{(2)}(X_T) \right)^2 + \dots, \quad (4.18)$$

with the hyperparameters α and β . The dots represent the possibility to include arbitrary higher moments.

4.4.2 Domain-Adversarial Method

The domain-adversarial method (or domain adaptation by backpropagation) was introduced in [59, 62], the proposed architecture is shown in figure 4.4. The method is realized through an extension to the original feed-forward DNN. The original DNN is divided into two parts. The first part is the feature extractor $G_f(\cdot; \theta_f)$ which maps the input features to a latent feature vector \vec{f} and forwards them to the second part. The second part is the label predictor $G_y(\cdot; \theta_y)$, which predicts the class labels y . The additional part is the domain classifier $G_d(\cdot; \theta_d)$, which is used to distinguish between the source domain and the target domain. As the label predictor, the domain classifier takes \vec{f} as its input features. The first layer is a gradient reversal layer (GRL) followed by an arbitrary number of feed-forward layers and one binary output neuron. The GRL has no free trainable parameters, it is the identity function in forward propagation. During backpropagation, the gradient is multiplied by the negative constant $-\lambda$.

During training, samples from the source domain and target domain are used at the same time. As a loss function for the label predictor, denoted as class loss L_y , the categorical cross entropy can be used. The class loss takes only the labeled samples from the source domain into account, backpropagation is used to update the free parameters θ_y of the label predictor and the free parameters θ_f of the feature extractor. For the domain classifier, a binary cross entropy can be used as a loss function, denoted as domain loss L_d . The domain loss includes samples from both domains. The free parameters θ_d of the domain classifier are updated by backpropagation in a way that the discrimination between the domains is improved. Through the gradient reversal layer, the gradient is flipped and the θ_f are updated towards a maximum of the domain loss. As a result \vec{f} becomes indistinguishable between source and target domain, this

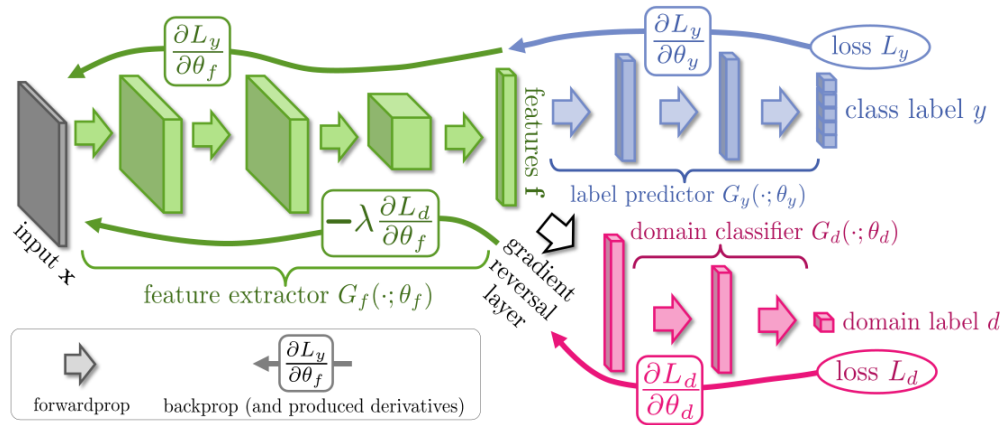


Figure 4.4: The domain-adversarial method. The input features \vec{x} , the feature extractor (green) and the label predictor (blue) form a conventional DNN. Additionally, a domain classifier (red) is added. For a more detailed description refer to the text. Source: [59]

is called \vec{f} becomes domain invariant. This also leads to domain invariant class labels. The updated free parameters can be computed using a standard optimizer. For the SGD, they are given by

$$\begin{aligned}
 \theta_f &\leftarrow \theta_f - \eta \left(\frac{\partial L_y}{\partial \theta_f} - \lambda \frac{\partial L_d}{\partial \theta_f} \right) , \\
 \theta_y &\leftarrow \theta_y - \eta \left(\frac{\partial L_y}{\partial \theta_y} \right) , \\
 \theta_d &\leftarrow \theta_d - \eta \left(\frac{\partial L_d}{\partial \theta_d} \right) .
 \end{aligned} \tag{4.19}$$

Note that this is done implicitly through the use of the GRL, no further changes have to be done to the optimizer. The first and third equation in 4.19 show that θ_d minimizes the domain loss while θ_f maximizes it, the factor λ is a trade-off factor between these two mechanisms. The domain classifier can therefore also be seen as an adversarial network, similar to the general adversarial network (GAN) approach of [63].

5 Heavy Flavor Jet Tagging

Jets that origin from the heavy flavor quarks c and b have characteristic properties compared to jets that origin from gluons or light flavor quarks, which are u , d , and s quarks. The properties are essentially determined through the bound state hadrons formed by the quarks and are explained in section 5.1. These properties can be used to distinguish the jets in respect of present quark flavors (tagging). This is done using a fully connected feed-forward MLP in the DeepCSV algorithm, which is explained in section 5.2. A more advanced DNN is used by the DeepFlavour algorithm that is described in 5.3.

5.1 Heavy Flavor Jet Properties

B (C) hadrons have a lifetime in the order of 1.5 ps (1 ps) [64], depending on their momentum, this results in a flight distance of some mm up to cm in the CMS detector. After this distance they decay at a secondary vertex (SV) where daughter particles arise. The SVs itself have characteristic properties, for example the vertex mass, which is the invariant mass of the SV. Another important variable is the flight distance (FD), defined as the distance from the PV to the SV. The FD can be measured in three spatial dimensions (3D), in the plane transverse to the beam line (2D) or in one dimension along the beam line (longitudinal). The tracks of the daughter particles can be propagated towards the beam line. The distance of their closest approach to the primary vertex is called impact parameter (IP). The daughter particles tend to have larger IPs compared to tacks that origin from the PV, they have displaced tracks. These aspects are outlined in figure 5.1. The IP can be measured similar to the FD in 3D, 2D or in longitudinal direction. The higher mass of the B or C hadrons compared to hadrons consisting of only light flavor quarks results in daughter particles with higher momentum perpendicular to the jet axis. Furthermore, b or c quark decays mostly happen through flavor-changing charged currents under the emission of a W boson, which in turn can decay into an electron or a muon. Therefore, b (c) jets contain an electron or a muon in about 20 % (10 %) [64] of the cases. Since b quarks decay mostly first in c and than in light quarks (see equation 1.5), they have in general several SVs and a higher probability of containing electrons or muons.

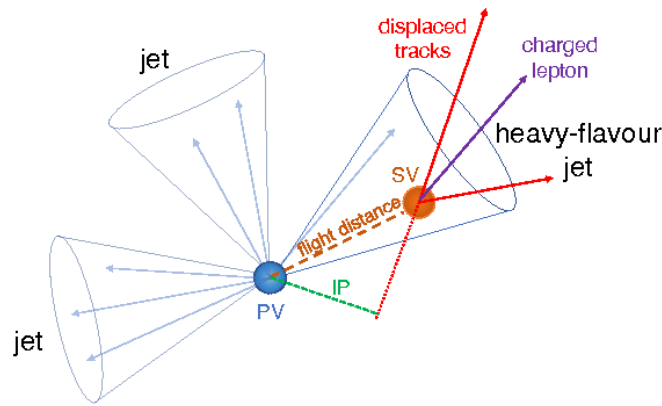


Figure 5.1: A heavy flavor jet. Several jet properties can be used to identify a heavy flavor jet. The main identifying properties are secondary vertices (SV) and displaced tracks with an impact parameter (IP) towards the primary vertex (PV). Source: [64]

5.1.1 Secondary Vertex Reconstruction

The reconstruction of the SV is done with the inclusive vertex finding (IVF) algorithm [65]. The IVF algorithm takes all tracks with $p_T > 0.8 \text{ GeV}$ and longitudinal IP $< 0.3 \text{ cm}$ into account. In a first step, seed tracks with thresholds on the IP and the IP significance (the IP value divided by its significance) are identified. Next, tracks are clustered if they form a compatible SV. The clustered set of tracks is then used to fit a SV. Afterwards, SVs are only kept if their 2D FD significance (the FD value divided by its significance) is larger than 2.5 and their 3D FD significance is larger than 0.5 and no other SV is compatible with the tracks. Furthermore are tracks removed if ΔR between the track and the SV flight direction is larger than 0.4. After this cleaning, a repeated SV fitting is performed followed by a repeated cleaning. The remaining SVs have a FD significance (the FD value divided by its significance) of 10 or more and less than 20 % of their tracks shared with other SVs.

5.2 The DeepCSV Algorithm

The DeepCSV algorithm used in this thesis consists of a DNN with five hidden layers with 100 neurons each. The DNN uses up to 66 input features to get a good separation of the jets in four different categories. This results in 66 neurons in the input layer and 4 neurons in the output layer. The implementation is done using the high-level API Keras [66] with the machine learning framework TensorFlow [67] as back end.

For the hidden layers, ReLu activation functions are used while the last layer has softmax activation functions. The loss function is the categorical cross-entropy. After

each hidden layer, a dropout layer with a dropout rate of 0.1 is used (see section 4.2 and 4.3).

5.2.1 Categories

In this thesis, jets were distinguished in four categories, which are b, bb, c and udsg. Jets containing two B hadrons are categorized as bb. Jets containing one B hadron are categorized as b. Jets in which no B hadron is present, but at least one C hadron, are categorized as c. If neither a B hadron nor a C hadron exists, the jet is categorized as udsg. Jets that origin from pileup remain uncategorized. The simulated jets are labeled corresponding to their categories in the 1-of-K coding scheme (as in section 4.3.2).

5.2.2 Input Features

The DeepCSV algorithm makes use of the properties described in section 5.1. It takes into account global jet properties with twelve features, six charged particle tracks with seven features each, four neutral particle tracks with one feature each and one secondary vertex with eight features. The particle tracks from PF elements and the reconstructed SVs are first selected by demanding some quality criteria. Afterwards, the six most displaced charged tracks and the four most displaced SVs are taken. For the neutral particles, the ones with the highest p_T are selected. The used variables and their description are found in [64].

During the training, the convergence of the DNN is faster if all input features have the same mean and variance. Therefore, each input feature i is rescaled by using its mean value μ_i and standard deviation σ_i . The normalized input values \hat{x}_i are then given by

$$\hat{x}_i = \frac{x_i - \mu_i}{\sigma_{x,i}} \quad , \quad (5.1)$$

and have a mean value of 0 and a standard deviation of 1. If a feature is missing for a certain jet, a default value is assigned that is not overlapping with the core of the feature distribution and close to zero.

5.3 The DeepFlavour Algorithm

The DeepFlavour algorithm [68] is a further development of the DeepCSV algorithm. In contrast to the DeepCSV algorithm, the selection of the tracks and vertices are much looser and much more candidates with more features of each PF candidate are included. Moreover the number of PF candidates varies for each jet. This requires an alternative flexible network structure that compresses the high amount of information.

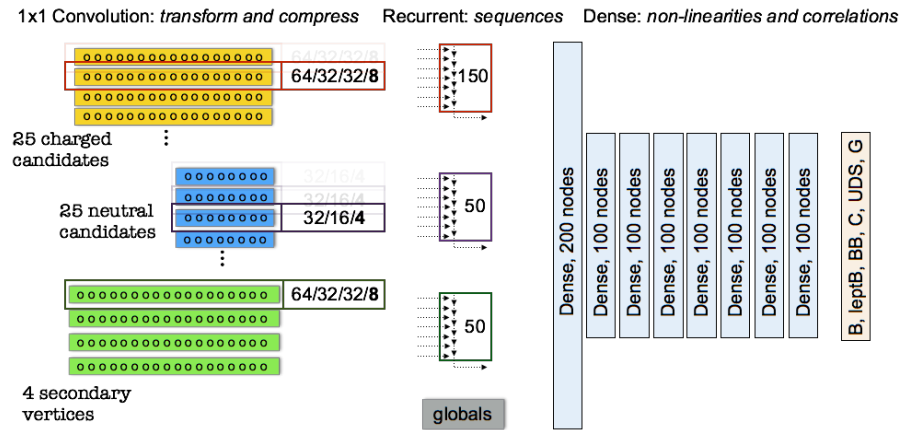


Figure 5.2: Architecture of the DeepFlavour algorithm. Charged particle tracks, neutral particle tracks, SVs and global jet information are inserted separately in the network and compressed through 1x1 convolutional layers. The subsequent use of recurrent sequences leads to a more flexible structure in respect of the number of objects that are present in the jet. Together with the global features, the correlations between the object can be extracted. Source: With kind permission of Jan Kieseler

More information of jet tagging with advanced machine learning tools can be found in [69].

5.3.1 Architecture

The network structure of the DeepFlavour algorithm is shown in figure 5.2. The features are divided in four groups, namely the charged candidates, the neutral candidates, the secondary vertices and the globals. The first three groups consist of several objects of which each object has the same features. Therefore, each group is arranged separately as a two dimensional feature layer. The first dimension represents the different features while the second dimension represents the various objects. A maximum of 25 charged particles, 25 neutral particles and 4 secondary vertices is given, but there is no issue for jets with less objects. The two dimensional layer is inserted into a 1x1 convolutional layer. This kind of layer is only fully connected in the first dimension and uses shared weights in the second dimension. The 1x1 refers to the so called filter kernel size and if a bigger filter kernel size is used, the layer performs a convolution of the input features with the so called filter, hence the name convolutional layer. A 1x1 convolutional layer is shown in figure 5.3.

The 1x1 convolutional layer generates output features which are again two-dimensional. In the first dimension, new latent features are constructed while the second dimension represents the same objects as before. For the charged particles and SVs, four 1x1 convolutional layers are connected in a row, they use 64, 32, 32 and 8 filters respectively. The neutral particles are processed in three 1x1 convolutional layers with 32, 16 and 4 filters

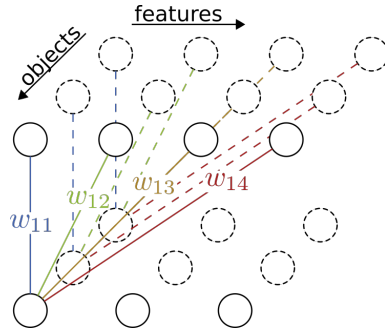


Figure 5.3: 1x1 convolutional layer. Shown are three objects with four features each. The objects share their weights among themselves, the lines in the same color represent the same weights. For each filter, one output neuron is generated for each object. In this case, three filters were used.

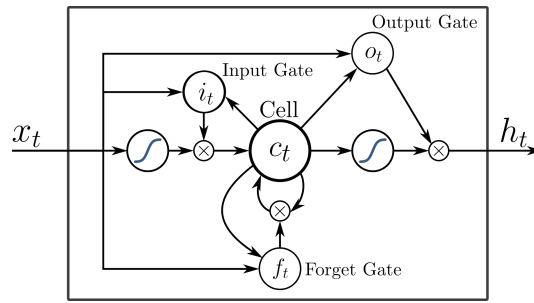


Figure 5.4: Long short-term memory (LSTM) unit. The LSTM unit transports the input vector x_t to three gates. The input gate i_t determines how much a new value influences the memory cell c_t , the forget gate f_t describes the degree the value of c_t is forgotten and the output gate defines the impact of c_t in the output. Adapted from [71].

each. Each two-dimensional output of the last 1x1 convolutional layer is then inserted into a long short-term memory (LSTM) [70] layer. The LSTM layer takes sequences of features as input, in this case the sequences are the various objects.

Each unit of the LSTM layer has a memory and is therefore capable of taking past sequences into account. One LSTM unit is shown in figure 5.4. The most information can be extracted from the last sequence, therefore the sequences have to be ordered in a way that the most important sequence is inserted as the last sequence. The output of the LSTM layer is one dimensional and can be chosen freely. For the DeepFlavour algorithm, 150 units were taken for the charged particles and 50 each for the neutral particles and SVs. The outputs of the three LSTM layers from the three categories respectively are concatenated together with the features of the globals into one flat layer. These are connected to a fully connected layer with 200 neurons and followed by seven fully connected layers of 100 neurons each.

The LSTM layer uses hyperbolic tangent activation functions. For the other hidden lay-

ers, ReLu activation functions were used. The output layer has again sigmoid activation function. The loss function is again the categorical cross-entropy and dropout layers are applied after each hidden layer with a dropout rate of 0.1. Additionally, a batch normalization layer is applied after each hidden layer (see section 4.2 and 4.3).

5.3.2 Categories

The categories are similar to the ones used in the DeepCSV algorithm with the difference that the DeepFlavour algorithm distinguishes between the light quarks u, d and s and gluons g , where the physics definition is used as explained in [72]. Moreover it differentiates between B hadron jets where the B hadron decays leptonically or hadronically. The labeling is done in the same way as in the DeepCSV algorithm.

5.3.3 Input Features

Compared to the DeepCSV algorithm, the set of input features has been increased. 15 global features, 16 features for each charged particle, 6 features for each neutral particle and 12 features for each SV were used. For the sequential processing of the LSTM layers, the objects have to be ordered. Charged particles and SVs are sorted in respect of the 2D IP significance in descending order. Charged candidates that were used in the PV fit do not have a 2D IP significance, they are sorted in ascending order of the $\Delta R_m(cPF, SV)$ value (ΔR of the charged candidate and its closest SV in the jet). For jets without SV, the particle p_T is used in descending order as a third criterion. Neutral PF candidates are sorted as a first criterion in ascending order by their $\Delta R_m(nPF, SV)$ value and as a second criteria descending by their p_T .

5.4 Performance of b Tagging Algorithms

To measure the performance of a tagging algorithm, the so called receiver operating characteristic (ROC) curve can be used. The ROC curve is computed by varying the test statistic (section 4.1) of the classifier output from its minimal to its maximal value. At each point the efficiency and the FPR are calculated. The ROC curve is commonly computed on simulated samples with the available labels. The performances of the DeepCSV and DeepFlavour algorithms are compared to the CSVv2 algorithm which is a well established b tagging algorithm in the CMS experiment. Their ROC curves are shown in figure 5.5. The classifier is applied at a fixed value of the test statistic, the so called working point. The working point is defined by the FPR. The loose working point (L) has a FPR of 1 %, the medium working point (M) has a FPR of 0.1 % and the tight working point (T) has a FPR of 0.01 %.

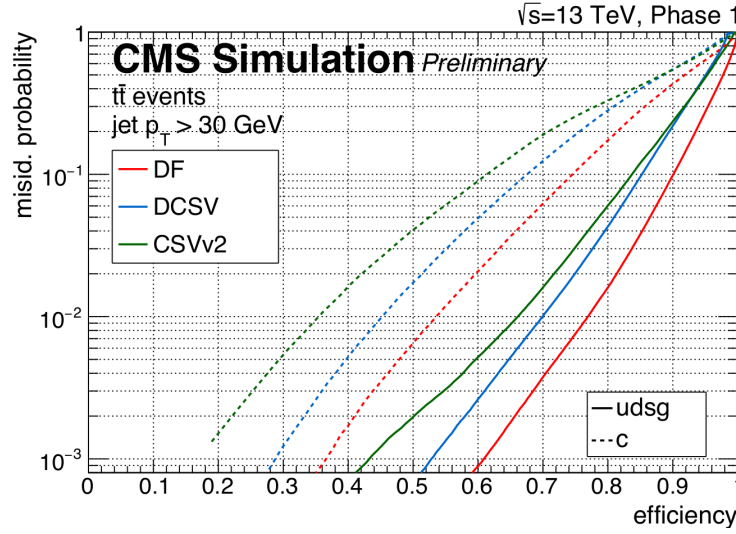


Figure 5.5: Performance of b tagging algorithms. The plot shows the misidentification probability (which is the FPR) against the b tagging efficiency measured on simulated samples. The b tagging efficiencies against light quarks and gluons (continuous lines) and c quarks (dashed lines) are shown for three different tagging algorithms. The CSVv2 algorithm has the worst performance. The DeepCSV (DCSV) is the currently proposed b tagging algorithm for CMS analyses. The DeepFlavour (DF) has the best performance, it is under development and the performance on real data is currently not known. Source: [68]

To calculate the efficiency in data, more complex methods have to be applied. Since these methods have model assumptions and the number of jets from data events is much more limited, they have high systematic and statistical uncertainties. With the efficiency on data ϵ_{data} and on simulated jets ϵ_{MC} on identical FPRs, a scalefactor can be computed as

$$SF_b = \frac{\epsilon_{\text{data}}}{\epsilon_{\text{MC}}} . \quad (5.2)$$

Scalefactors for the DeepCSV and CSVv2 algorithm computed with various methods are shown in figure 5.6. As expected, the scalefactor is in the most cases lower than one, this originates from the fact that the algorithms are trained on simulation and have therefore worse performance on data. More important than a scalefactor that is close to one is a scalefactor with low uncertainty. However, scalefactors which are far from one tend to have higher uncertainties. For the DeepFlavour algorithm, there exists no official measurement of the scalefactors, but scalefactors with higher uncertainties are expected since the unselected tracks and vertices, used by the DeepFlavour algorithm are less good modeled compared to the ones used by the DeepCSV algorithm.

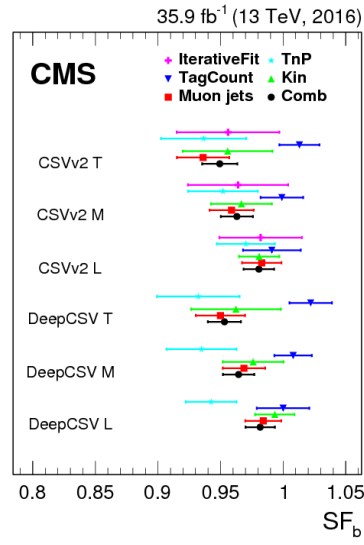


Figure 5.6: Scalefactor for b tagging algorithms. The scalefactor for the b tagging algorithms at different working points are shown for five methods. The measured scalefactors agree in their uncertainties. The black dot refers to the combined value and is mainly dominated by the Muon Jets method. Source: [64]

6 Event Selection

The subject of this thesis is to study the effect of domain adaptation methods on flavor tagging algorithms to improve the data to simulation agreement. In section 4.4 two domain adaptation methods were introduced. Both have the goal to transfer the source domain and the target domain into a domain invariant feature space that the classifier output becomes domain invariant. To achieve this goals, the source and the target distribution of their input features have to be similar. Furthermore, both domains should have the same distributions in their class compositions. Which means in this case they need to have the same amount of jets each quark flavor in data and simulation. In order to meet these requirements, an event selection is needed.

The event selection has several requirements to fulfill. It has to be performed in the same way for data and simulation and the data to simulation agreement for the selected events shall be as good as possible. The mismodeling of the simulation of the jets shall be the dominant reason for the domain discrepancy. The event selection has to be independent of the specific jets in each event. Moreover the selected events should contain balanced fractions of jets of the different categories used in the classifier. Finally the amount of jets should be high enough that the statistical uncertainties are negligible. The last two points are in direct conflict as categories like bb occur mostly for high p_T jets of which not many exist. Therefore the demand of balanced fractions is secondary.

6.1 Top Quark Decay

A good choice is to search for top quarks as they decay almost every time into a b quark and a W boson. Moreover, the $t\bar{t}$ process is under investigation by many analyses, these can be taken into comparison (for example ??). The W boson decays either into a lepton and a neutrino, or into a quark-antiquark pair. Top quarks are mainly produced in pairs at the LHC with a cross section of $\sigma_{t\bar{t}} = 831 \text{ pb}$ at $\sqrt{s} = 13 \text{ TeV}$ (predicted by the SM [?]). Their decay can be categorized by the decay of the W bosons as the potential lepton is a good indicator for those events. If both W bosons decay into quarks, it is called a hadronic decay. If both W bosons decay into leptons, it is called a dileptonic decay. If one W decays into a lepton while the other into quarks, it is called semileptonic decay. The decay rates for the W boson and $t\bar{t}$ pairs are listed in table 6.1. The leading Feynman diagrams for top quark pair production are shown in figure 6.1.

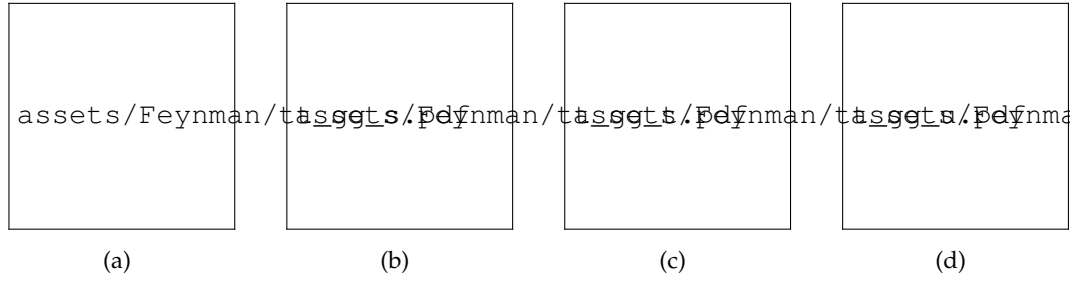


Figure 6.1: Dominant processes of top quark pair production at the LHC. Top quark pair production by gluon fusion is the dominant process at the LHC. It occurs in the s channel (a), in the t channel (b) and in the u channel (c). Another production process of the same order of perturbation theory is the quark-antiquark annihilation in the s channel (d). Because of the PDFs, this channel is less contributions.

The $t\bar{t}$ hadronic decay mode has the best statistic but is inappropriate for a selection as only jets are in the final states and other effects have the same signature and occur far more often. The $t\bar{t}$ semileptonic decay mode has also a high statistic and with the lepton in the final state, a good indicator for this process exists. As listed in table 6.1, $t\bar{t}$ semileptonic decays also provide some c quarks. Therefore, selection criteria are applied in respect of the $t\bar{t}$ semileptonic decay mode. This is described in section 6.3. A much lower statistic has the $t\bar{t}$ dileptonic decay mode. But events of this category that decay into a electron muon pair with opposite flavor and charge produce the clearest signal. In section 6.4 this selection is described.

Table 6.1: Decay modes of W bosons and top quark pairs. Shown are the measured branching ratios from [?]. In almost 50 % of the cases of a hadronic W decay, a c quark occurs, the X denotes a d or s quark. The τ lepton often decays into quarks and is therefore often seen as a hadronic decay mode. In this table, the τ is included as a lepton ℓ .

Particle	Decay mode	Branching ratio in %
W^+	$e^+ + \bar{\nu}_e$	10.71 ± 0.16
W^+	$\mu^+ + \bar{\nu}_\mu$	10.63 ± 0.15
W^+	$\tau^+ + \bar{\nu}_\tau$	11.38 ± 0.21
W^+	$q + \bar{q}$	67.41 ± 0.27
W^+	$c + X$	33.3 ± 2.6
$t\bar{t}$	hadronic (all-jets)	45.7
$t\bar{t}$	semileptonic ($\ell + \text{jets}$)	43.8
$t\bar{t}$	dileptonic ($\ell\ell$)	10.5

6.2 Particle Identification

The τ decays before it can be directly identified, therefore it is less suited for a selection. The main identifying objects are electrons and muons. In a first selection step, quality criteria are applied on these particles and an isolation is required. The set of remaining muons and lepton candidates have a low fake rate and can be used for the decay mode selection. Moreover, jets have a high fake rate as for example photons are often misidentified as jets. This requires a further selection.

6.2.1 Muon Selection

Muon-quality requirements are used in many analyses, several requirements are combined in so called muon IDs. In this thesis the muon is required to fulfill the tight muon ID. The tight muon ID contains requirements of the number of hits of the muon track in various detector components or a minimum value of χ^2 divided by the number of decreases of freedom of the fit. Apart from this, in this thesis muons with $|\eta| < 2.4$ are discarded as these are not fully covered from the muon chamber. A minimum p_T value is required as the modeling of low p_T muons is worse, the p_T cut has a different value in each selection. Additionally an isolation criteria of $I_l^{\text{PF}/\Delta\beta} < 0.15$ is applied as described in section 3.2.4. The set of selected muons is denoted as good muons.

6.2.2 Electron Selection

For the electrons, a similar ID, the tight electron ID is applied. Furthermore, to ensure the best performance of the silicon tracker the electron is required to have a value of

$|\eta| < 2.4$. However there is a gap in the ECAL between the barrel and the endcap part, therefore electrons in the interval of $1.4442 < |\eta_{SC}| < 1.5660$ are excluded, where η_{SC} refers to the position of the ECAL supercluster of the electron. Additionally, cuts on the IP have to be performed to reduce pileup contributions. The transverse IP has to be lower than 0.05 (0.1) while the longitudinal IP has to be lower than 0.1 (0.2) for electrons with $|\eta_{SC}| < 1.4442$ ($|\eta_{SC}| > 1.5660$). Furthermore a $p_T > 20$ GeV is required due to the mismodeling of low p_T electrons. The set of selected electrons is denoted as good electrons.

6.2.3 Jet Selection

The Jet selection is performed by requiring

- $p_T > 30$ as those jets have a low probability to come from the hard scattering process,
- $|\eta| < 2.4$ to ensure the full capability of the detector,
- neutral electromagnetic energy fraction < 0.99 to exclude photons modeled as jets,
- charged electromagnetic energy fraction < 0.99 to exclude electrons modeled as jets,
- neutral hadron energy fraction < 0.99 to exclude single neutral hadrons modeled as jets,
- charged hadron energy fraction > 0 as a jet should have charged hadrons,
- number of charged and neutral candidates > 1 to exclude single particles reconstructed as jets,
- and number of charged candidates > 0 as a jet should have charged particles.

Finally, jets are rejected if a good electrons or a good muons has a closer distance than $\Delta R = 0.4$. The set of selected jets is denoted as good jets.

6.3 Top Quark Pair - Semileptonic Selection

The final state particles of a $t\bar{t}$ semileptonic decay in LO in QCD are one lepton, one neutrino, two b quarks, one up-type and one down-type quark. The best signal is obtained in the case where the lepton is a muon, the selection criteria are

- one good muon with $p_T > 30$,
- no additional good muon or good electron,
- four good jets with no further requirements on the jets,
- a transverse mass of the W boson $m_{T,W} > 50$ GeV to take the neutrino into account,
- a trigger selection, this will be explained later in this section.

A $t\bar{t}$ semileptonic decay mode that satisfies this selection is shown in figure 6.2. The main contributions from other processes are from W boson production in association with jets and Drell-Yan processes ($qq \rightarrow X \rightarrow \ell\ell$ where X is a photon or Z boson) in association with jets where one of the leptons misses the detector. A minor contribution is coming from single top quark production in association with jets.

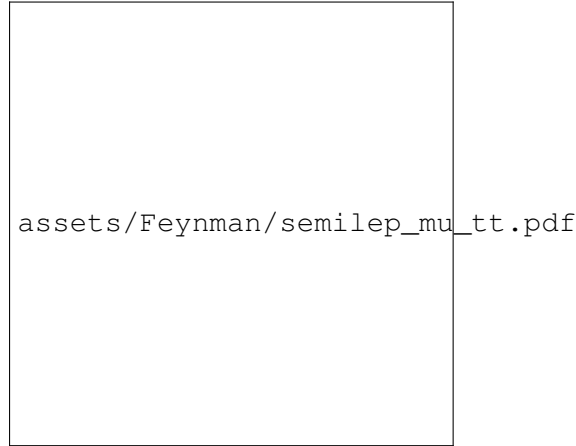


Figure 6.2: Semileptonic decay of a top quark pair. The process is in LO of QCD in respect of the final state particles. To be unbiased of the selected jets, no selection for b jets is done. The quarks from the W boson decay are preferably light flavor or c quarks. The charged conjugated process is also possible.

6.3.1 Datasets

The datasets for recorded events are sorted in respect of L1 triggers and the HLTs. The appropriate dataset for this selection is the single muon dataset. Each event was accepted by one or several HLTs of this category. For later steps, events are only taken if they passed one of two specific HLTs for isolated muons with $p_T > 24\text{GeV}$. The use of a HLT selection on data and simulation ensures that no events from other datasets for data are represented in the simulated datasets. For the data, the recorded events from all periods of 2016 were taken. The total recorded luminosity is $\mathcal{L} = 35.14\text{fb}^{-1}$, it was computed using the `BRILCALC` [73] tool.

The simulated events are sorted by the hard scattering process, additional jets can occur from pileup, initial or final state radiation or from the parton shower. The $t\bar{t}$ processes were generated by `POWHEG` including NLO processes in QCD. For the W boson processes with additional jets, a dataset with NLO in QCD exists, this dataset is modeled with `AMC@NLO` and has negative weights and the statistic is low. Alternative datasets exist in LO with more events. These are grouped in the number of jets in the final state of the hard process. The datasets for a W boson with two, three and four additional jets were taken. The Drell-Yan processes are generated with `AMC@NLO` and grouped in respect of the invariant mass M_{inv} of the virtual photon or Z boson. Two datasets, one for $10\text{GeV} < M_{\text{inv}} < 50\text{GeV}$ and one for $M_{\text{inv}} > 50\text{GeV}$ were taken. The single t processes are grouped in respect of their production mode and generated with `POWHEG`. The four main contributing production channels were taken, these are single top quark production in the t-channel and in association with an W boson and the corresponding processes for antitop quarks. The subsequent effects of

Table 6.2: Generated hard-scattering processes and their cross sections. The higher the cross section σ and the lower the number of generated events N_{gen} is, the worse the statistic. One can see, the $W + \text{jets}$ dataset has the worst statistic, in the semileptonic selection, it was chosen to take the jet grouped datasets instead. The processes where two bosons are involved have only a significant contribution in the dileptonic selection.

Dataset	N_{gen}	σ in pb
$t\bar{t}$	154 948 894	831.76[74]
DY($10 \text{ GeV} < M_{\text{inv}} < 50 \text{ GeV}$)	47 946 519	6025.2 [75]
DY($M_{\text{inv}} > 50 \text{ GeV}$)	81 781 052	22 635.1 [75]
$W + \text{jets}$	16 497 031	61 526.7 [75]
$W + 2 \text{ jets}$	29 878 415	3161 [76]
$W + 3 \text{ jets}$	19 798 117	948.2 [76]
$W + 4 \text{ jets}$	9 170 576	494.6 [76]
single t (t-channel)	67 240 808	136.02[77, 78]
single \bar{t} (t-channel)	38 811 017	80.95[77, 78]
Wt	6 952 830	35.6 [77, 78]
$W\bar{t}$	6 933 094	35.6 [77, 78]
WW	994 012	118.7 [79]
WZ	1 000 000	44.9 [79]
ZZ	998 034	15.4 [79]

all processes were modeled by PYTHIA 8 while the detector simulation is performed with GEANT4. The number of events and the cross section of each process are listed in table 6.2.

6.3.2 Data-to-Simulation Comparison

To compare the data with the simulation, the events from the simulation of a certain process i have to be scaled, this is done by assigning a weight for each event depending on the process. The weight is determined by the corresponding cross section σ_i and the efficiency ϵ_i . The ϵ_i is the number of selected events divided by the number of generated events $N_{i,\text{gen}}$. The events have also to be scaled with the luminosity of the data that is in comparison. Each event of the simulation gets an event weight of

$$w_i = \mathcal{L} \frac{\sigma_i}{N_{i,\text{gen}}} . \quad (6.1)$$

The ϵ_i results from summing up all events. Further improvements can be achieved by applying reweighting for each event depending on known scalefactors, this was

described in section 3.3. In this selection, scalefactors were applied for the tight muon ID (sf^{ID}), for the muon isolation (sf^{Iso}), for the muon efficiency in the tracker (sf^{Track}) and for the HLT selection (sf^{HLT}). These depend on the muon p_T and η value. Furthermore, a pileup reweighting w_{PU} was performed, depending on the number of PV of each event. Additionally one has to take negative event weights w_{lhe} for events generated with the AMC@NLO matrix element generator into account (as explained in section 3.1.2). These appear in Drell-Yan processes. For this dataset, one has to compute an effective number of generated events $N_{i,\text{gen}}^{\text{eff}}$, which is the number of events with positive weights subtracted by the number of events with negative weights. The scalefactors are different for the various periods of the total run 2 in 2016. Therefore, they have to be weighted with the corresponding luminosity. The event weight for each event is then given by

$$w_i(nPV, p_T, \eta, w_{\text{lhe}}) = \frac{\sigma_i}{N_{i,\text{gen}}^{\text{eff}}} \cdot w^{\text{PU}}(nPV) \cdot w_{\text{lhe}} \cdot \sum_{r \in \{\text{Periods}\}} \mathcal{L}_r \cdot sf_r^{\text{ID}}(p_T, \eta) \cdot sf_r^{\text{Iso}}(p_T, \eta) \cdot sf_r^{\text{Track}}(\eta) \cdot sf_r^{\text{HLT}}(p_T, \eta) \quad (6.2)$$

The data to simulation agreement can be investigated by looking at distributions of different variables. Four meaningful variables are shown in figure 6.3. The fact that the number of events in simulation is lower than in data is a hint for an additional process which was not included here. This could be QCD processes, however QCD processes are more complicated to implement. Since the shape is in a good agreement, and this is the more important fact, the selection is sufficient for the domain adaptation studies.



Figure 6.3: Distributions of variables from the semileptonic selection. Shown are the distributions of muon p_T (a), muon η (b), reconstructed transverse energy of the W boson m_T^W (c) and the missing transverse energy p_T^{miss} (d). The total amount of events in data is about 7 % higher compared to the simulation. The shape in each distribution is in good agreement. The statistic uncertainties are negligible, systematic uncertainties have not been computed.



Figure 6.4: Dileptonic decay of a top quark pair. With a selection of the muon and electron, a high fraction of $t\bar{t}$ events can be achieved, and therefore many b jets are obtained. Further cuts on the amount of jets or the missing p_T^{miss} are not necessary for this thesis. Additional jets through initial or final state radiation, pileup or through the parton shower are possible.

6.4 Top Quark Pair - Dileptonic Selection

The dileptonic decay mode can be further distinguished in respect of the leptons in the final state. The decay mode with two muons or two electrons in the final state are dominated by Drell-Yan processes which have mainly additional light flavor quark and gluon jets. The decay mode with a muon and an electron with opposite charge, shown in figure 6.4 has the clearest signal for $t\bar{t}$ and is therefore preferred. The selection criteria are

- one good muon with $p_T > 20$,
- one good electron with opposite charge,
- $M_{\text{inv}} > 20$ GeV of the electron-muon pair to reject backgrounds,
- the good electron or the good muon with $p_T > 25$ GeV,
- a trigger selection, which is explained in the following.

The main contributions from other processes are Drell-Yan processes where, for example one muon is misidentified as an electron. Minor contributions coming from vector boson pairs (WW, WZ and ZZ), W boson processes and single top quark processes.

6.4.1 Datasets

The measured events are taken from the muon-electron/photon dataset. Two HLTs were used in the selection where both require one isolated muon and one isolated electron. The first (second) has a limit for the muon of $p_T > 23$ GeV ($p_T > 8$ GeV) and for the electron of $p_T > 12$ GeV ($p_T > 23$ GeV). Unfortunately there was a problem during the periods of 2016, two different pairs of these HLTs were used. The total recorded

luminosity for this dataset was again computed using the `BRILCALC` tool to a value of $\mathcal{L} = 35.199 \text{ fb}^{-1}$.

For the simulation, the same datasets were used for the $t\bar{t}$, Drell-Yan, Wt , and $W\bar{t}$ processes. For the W boson with additional jets, the $W + \text{jets}$ datasets was used here. Single t and single \bar{t} processes in the t -channel have a negligible contribution and are not included. Instead, processes with two bosons were included, the `PYTHIA 8` program were used for the matrix element generation in NLO and the simulation of subsequent processes. The datasets are given in table 6.2.

6.4.2 Data-to-Simulation Comparison

The comparison was done in the same way than for the semileptonic selection in the last section. The fact that two different sets of trigger were used during the periods of the 2016 data has to be taken into account for the simulation. The event weights are therefore additionally multiplied with a flag $f \in \{0, 1\}$ of the HLT of the corresponding period. Moreover a trigger for the isolated electron sf^{IDandISO} has to be included. The event weight is then given by

$$\begin{aligned}
 w_i(nPV, p_{T,\mu}, \eta_\mu, p_{T,\text{el}}, \eta_{\text{el}}, \eta_{\text{SC,el}}, w_{\text{lhe}}, f_r) &= \frac{\sigma_i}{N_{i,\text{gen}}^{\text{eff}}} \cdot w^{\text{PU}}(nPV) \cdot w_{\text{lhe}} \\
 &\cdot \sum_{r \in \{\text{Periods}\}} \left(f_r \cdot \mathcal{L}_r \cdot sf_r^{\text{ID}}(p_{T,\mu}, \eta_\mu) \cdot sf_r^{\text{Iso}}(p_{T,\mu}, \eta_\mu) \cdot sf_r^{\text{Track}}(\eta_\mu) \right. \\
 &\quad \left. \cdot sf_r^{\text{IDandISO}}(p_{T,\text{el}}, \eta_{\text{SC,el}}) \cdot sf_r^{\text{HLT}}(\eta_{\text{el}}, \eta_\mu) \right) \quad (6.3)
 \end{aligned}$$

The selection has no requirement of the number of jets. The number of jets together with distributions for the leading (trailing) lepton, which is the good muon or good electron with the highest (lowest) p_T , is shown in figure 6.5.

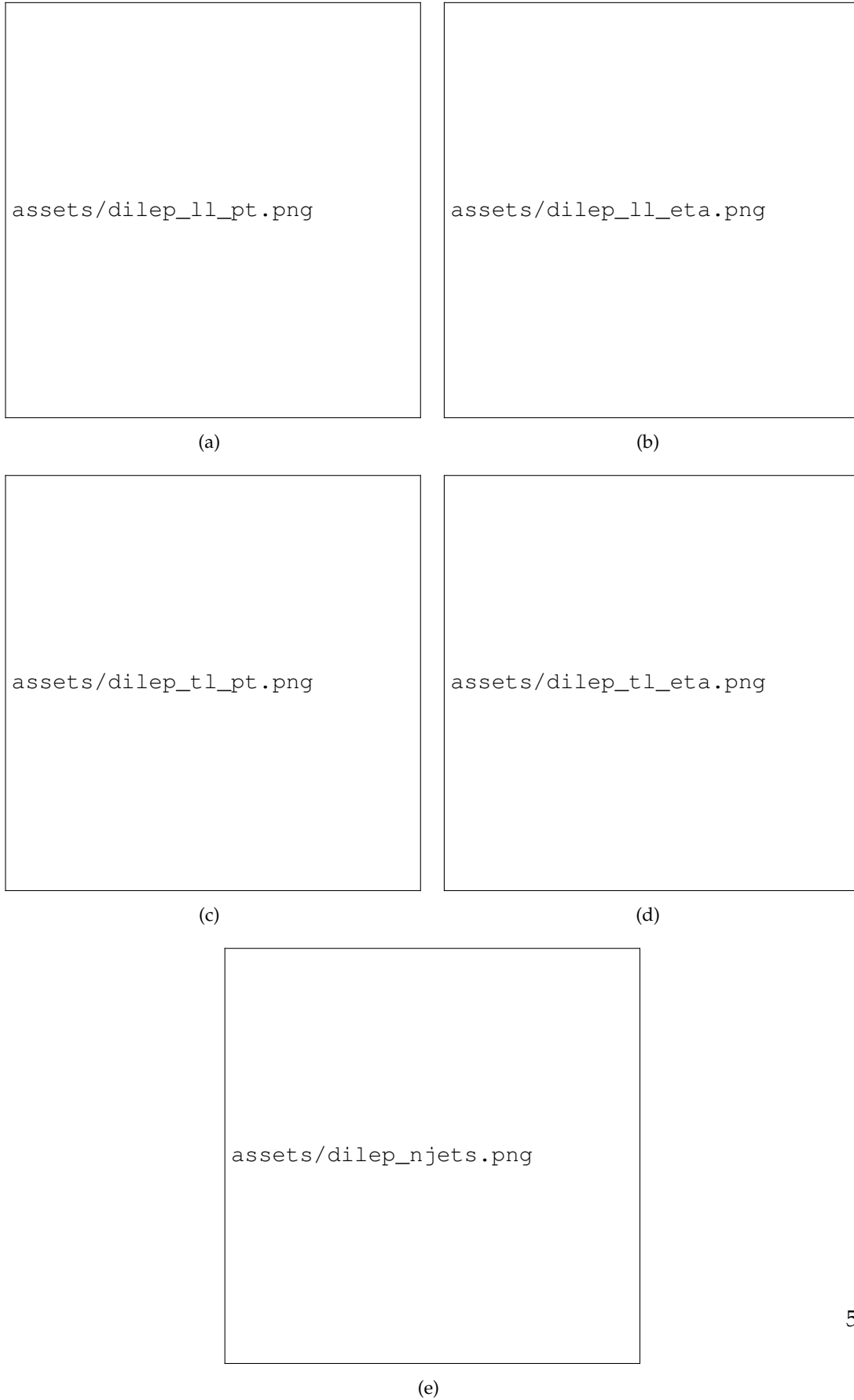


Figure 6.5: Distributions of variables from the dileptonic selection. Shown are the distributions of p_T (a) and η (b) of the leading lepton, and the p_T (c) and η (d) of the trailing lepton and the number of jets (e) in each event. The number of simulated events is about 7 % higher than the number of events in data. The reason is assumed in wrong scalefactors. However the shape is again in a good

7 Domain Adaptation Studies

7.1 Domain Adaptation on the DeepFlavour Algorithm

7.1.1 •

7.2 Domain Adaptation on the DeepCSV Algorithm

7.3 Two Tag Efficiencies

List of Tables

1.1	Fermions of the SM	6
1.2	Bosons of the standard model	7
6.1	Decay Modes of W Bosons and Top Quark Pairs	51
6.2	Hard-Scattering Processes	54

List of Figures

2.1	CERN Accelerator Complex	12
2.2	CMS Detector Slice	14
3.1	Proton-Proton Scattering Process	20
3.2	Parton Distribution Functions	21
3.3	Infrared and Collinear Safety of Jets	26
3.4	Jet Energy Corrections	28
3.5	Pileup Distributions for Data and Simulation	29
4.1	Perceptron and Activation Function	32
4.2	Multilayer Perceptron	33
4.3	The Effect of Domain Adaptation	38
4.4	Domain-Adversarial Method	40
5.1	Heavy Flavor Jet	42
5.2	Architecture of the DeepFlavour Algorithm	44
5.3	1x1 Convolutional Layer and LSTM Unit	45
5.4	Long Short-Term Memory Unit	45
5.5	Architecture of the DeepFlavour Algorithm	47
5.6	Scalefactors for b Tagging Algorithms	48

6.1	Dominant Processes of Top Quark Pair Production	50
6.2	Semileptonic Decay of a Top Quark Pair	53
6.3	Distributions of Variables from the Semileptonic Selection	56
6.4	Dileptonic Decay of a Top Quark Pair	57
6.5	Distributions of Variables from the Dileptonic Selection	59

Bibliography

- [1] F. Abe *et al.*, “Observation of top quark production in $\bar{p}p$ collisions”, *Physical Review Letters*, vol. 74, pp. 2626–2631, 1995.
- [2] K. Kodama *et al.*, “Observation of tau neutrino interactions”, *Physics Letters B*, vol. 504, pp. 218–224, 2001.
- [3] G. Aad *et al.*, “Observation of a new particle in the search for the standard model higgs boson with the atlas detector at the lhc”, *Physics Letters B*, vol. 716, pp. 1–29, 2012.
- [4] S. Chatrchyan *et al.*, “Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC”, *Physics Letters B*, vol. 716, pp. 30–61, 2012.
- [5] D. Hanneke, S. F. Hoogerheide, and G. Gabrielse, “Cavity Control of a Single-Electron Quantum Cyclotron: Measuring the Electron Magnetic Moment”, *Physical Review A*, vol. 83, p. 052122, 2011.
- [6] F. Halzen and A. D. Martin, “Quarks and Leptons: An Introductory Course in Modern Particle Physics”, 1984.
- [7] G. L. Kane, “Modern elementary particle physics: the fundamental particles and forces?”, updated ed., 4. pr. ed., 1998.
- [8] J. Schwichtenberg, “Physics from symmetry”, Undergraduate Lecture Notes in Physics, Springer, Cham, 2nd ed. 2018 ed., 2018.
- [9] M. E. Peskin and D. V. Schroeder, “An Introduction to quantum field theory”, Addison-Wesley, Reading, USA, 1995.
- [10] C. Patrignani *et al.*, “Review of Particle Physics”, *Chinese Physics C*, vol. 40, no. 10, p. 100001, 2016.
- [11] A. Quadt, “Top quark physics at hadron colliders”, *The European Physical Journal C - Particles and Fields*, vol. 48, pp. 835–1000, Dec 2006.
- [12] CERN - Education, Communications and Outreach Group, “LHC Guide”, Mar 2017. <http://cds.cern.ch/record/2255762>, last accessed on 2018-06-21.
- [13] CERN, “The Large Hadron Collider”, 2014. <http://home.cern/topics/large-hadron-collider>, last accessed on 2018-06-07.

- [14] C. De Melis, “The CERN accelerator complex. Complexe des accélérateurs du CERN”, 2016. <https://cds.cern.ch/record/2197559>, last accessed on 2018-06-07.
- [15] Lujan P., “CMS Luminosity - Public Results”, 2018. https://twiki.cern.ch/twiki/bin/view/CMSPublic/LumiPublicResults#2016_proton_proton_13_TeV_collis, last accessed on 2018-06-28.
- [16] S. Chatrchyan *et al.*, “The CMS Experiment at the CERN LHC”, *Journal of Instrumentation*, vol. 3, p. S08004, 2008.
- [17] The CMS Collaboration, “Search for the $t\bar{t}(H \rightarrow b\bar{b})$ process in pp collisions at $\sqrt{s} = 13$ TeV and study of Higgs boson couplings”, CMS-PAS-HIG-17-016, CERN, Geneva, 2018.
- [18] S. R. Davis, “Interactive Slice of the CMS detector”, 2016. <https://cds.cern.ch/record/2205172>, last accessed on 2018-06-08.
- [19] V. Karimäki *et al.*, “The CMS tracker system project: Technical Design Report”, Technical Design Report CMS, CERN, Geneva, 1997.
- [20] S. Chatrchyan *et al.*, “Description and performance of track and primary-vertex reconstruction with the CMS tracker”, *Journal of Instrumentation*, vol. 9, p. P10009, 2014.
- [21] The CMS Collaboration, “The CMS electromagnetic calorimeter project: Technical Design Report”, Technical Design Report CMS, Geneva, 1997.
- [22] The CMS Collaboration, “The CMS hadron calorimeter project: Technical Design Report”, Technical Design Report CMS, Geneva, 1997.
- [23] The CMS Collaboration, “The CMS muon project: Technical Design Report”, Technical Design Report CMS, Geneva, 1997.
- [24] V. Khachatryan *et al.*, “The CMS trigger system”, *Journal of Instrumentation*, vol. 12, no. 01, p. P01020, 2017.
- [25] S. Dulat *et al.*, “New parton distribution functions from a global analysis of quantum chromodynamics”, *Physical Review Journals D*, vol. 93, no. 3, p. 033006, 2016.
- [26] R. D. Ball *et al.*, “Parton distributions from high-precision collider data”, *The European Physical Journal C*, vol. 77, no. 10, p. 663, 2017.
- [27] Y. L. Dokshitzer, “Calculation of the Structure Functions for Deep Inelastic Scattering and $e^+ e^-$ Annihilation by Perturbation Theory in Quantum Chromodynamics.”, *Soviet Physics*, vol. 46, pp. 641–653, 1977. [Zh. Eksp. Teor. Fiz.73,1216(1977)].

-
- [28] V. N. Gribov and L. N. Lipatov, “Deep inelastic $e p$ scattering in perturbation theory”, *Soviet Journal of Nuclear Physics*, vol. 15, pp. 438–450, 1972. [Yad. Fiz.15,781(1972)].
- [29] G. Altarelli and G. Parisi, “Asymptotic Freedom in Parton Language”, *Nuclear Physics B*, vol. 126, pp. 298–318, 1977.
- [30] N. Metropolis and S. Ulam, “The monte carlo method”, *Journal of the American Statistical Association*, vol. 44, no. 247, pp. 335–341, 1949.
- [31] M. Bachtis, “Event simulation”, pp. 47–54. Springer International Publishing, Cham, 2014.
- [32] B. Andersson, G. Gustafson, G. Ingelman, and T. Sjöstrand, “Parton fragmentation and string dynamics”, *Physics Reports*, vol. 97, no. 2, pp. 31 – 145, 1983.
- [33] J. Alwall, M. Herquet, F. Maltoni, O. Mattelaer, and T. Stelzer, “MadGraph 5 : Going Beyond”, *JHEP*, vol. 06, p. 128, 2011.
- [34] F. Maltoni and T. Stelzer, “MadEvent: Automatic event generation with MadGraph”, *Journal of High Energy Physics*, vol. 02, p. 027, 2003.
- [35] J. Alwall *et al.*, “The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations”, *Journal of High Energy Physics*, vol. 07, p. 079, 2014.
- [36] P. Nason, “A New method for combining NLO QCD with shower Monte Carlo algorithms”, *Journal of High Energy Physics*, vol. 11, p. 040, 2004.
- [37] S. Frixione, P. Nason, and C. Oleari, “Matching NLO QCD computations with Parton Shower simulations: the POWHEG method”, *Journal of High Energy Physics*, vol. 11, p. 070, 2007.
- [38] S. Alioli, P. Nason, C. Oleari, and E. Re, “A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX”, *Journal of High Energy Physics*, vol. 06, p. 043, 2010.
- [39] T. Sjöstrand *et al.*, “An Introduction to PYTHIA 8.2”, *Comput. Phys. Commun.*, vol. 191, pp. 159–177, 2015.
- [40] S. Agostinelli *et al.*, “G4 - a simulation toolkit”, vol. 506, pp. 250–303, 07 2003.
- [41] R. Fruhwirth, “Application of Kalman filtering to track and vertex fitting”, *Nuclear Instruments and Methods A*, vol. 262, pp. 444–450, 1987.
- [42] The CMS Collaboration, “Particle-Flow Event Reconstruction in CMS and Performance for Jets, Taus, and MET”, 2009.

- [43] F. Beaudette, "The CMS Particle Flow Algorithm", in *Proceedings, International Conference on Calorimetry for the High Energy Frontier (CHEF 2013): Paris, France, April 22-25, 2013*, pp. 295–304, 2013.
- [44] M. Cacciari, G. P. Salam, and G. Soyez, "The Anti-k(t) jet clustering algorithm", *Journal of High Energy Physics*, vol. 04, p. 063, 2008.
- [45] G. Salam, "Jets", 2008. <https://gsalam.web.cern.ch/gsalam/repository/talks/2008-cteq-mcnet.pdf>, last accessed on 2018-07-15.
- [46] The CMS collaboration, "Determination of jet energy calibration and transverse momentum resolution in cms", *Journal of Instrumentation*, vol. 6, no. 11, p. P11002, 2011.
- [47] V. Sordini, "JEC levels", 2016. <https://twiki.cern.ch/twiki/bin/viewauth/CMS/IntroToJEC>, last accessed on 2018-07-16.
- [48] C. M. Bishop, "Pattern recognition and machine learning", Information science and statistics, Springer, New York, 2006.
- [49] I. Goodfellow, Y. Bengio, and A. Courville, "Deep learning", 2016. <http://www.deeplearningbook.org>, last accessed on 2018-07-26.
- [50] S. Hochreiter, "Untersuchungen zu dynamischen neuronalen netzen". Diplomarbeit im fach informatik, Technische Universität München, April 1991.
- [51] P. Ramachandran, B. Zoph, and Q. V. Le, "Searching for activation functions", *Computing Research Repository*, vol. abs/1710.05941, 2017.
- [52] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization", *Computing Research Repository*, vol. abs/1412.6980, 2014.
- [53] L. Prechelt, "Early stopping — but when?", pp. 53–67. Springer, Berlin, Heidelberg, 2012.
- [54] G. E. Hinton *et al.*, "Improving neural networks by preventing co-adaptation of feature detectors", *Computing Research Repository*, vol. abs/1207.0580, 2012.
- [55] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift", *Computing Research Repository*, vol. abs/1502.03167, 2015.
- [56] E. Tzeng *et al.*, "Adversarial discriminative domain adaptation", *Computing Research Repository*, vol. abs/1702.05464, 2017.
- [57] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function", *Journal of Statistical Planning and Inference*, vol. 90, pp. 227–244, 2000.

- [58] S. J. Pan *et al.*, “Domain adaptation via transfer component analysis”, *IEEE Transactions on Neural Networks*, vol. 22, pp. 199–210, Feb 2011.
- [59] Y. Ganin *et al.*, “Domain-adversarial training of neural networks”, *Journal of Machine Learning Research*, vol. 17, no. 59, pp. 1–35, 2016.
- [60] K. Borgwardt *et al.*, “Integrating structured biological data by kernel maximum mean discrepancy”, *Bioinformatics*, vol. 22, pp. e49–e57, Aug. 2006.
- [61] E. Tzeng *et al.*, “Deep domain confusion: Maximizing for domain invariance”, *Computing Research Repository*, vol. abs/1412.3474, 2014.
- [62] Y. Ganin and V. Lempitsky, “Unsupervised Domain Adaptation by Backpropagation”, *Journal of Machine Learning Research*, Sept. 2014.
- [63] I. J. Goodfellow *et al.*, “Generative Adversarial Networks”, *ArXiv e-prints*, June 2014. <https://arxiv.org/abs/1406.2661>, last accessed on 2018-08-01.
- [64] A. M. Sirunyan *et al.*, “Identification of heavy-flavour jets with the CMS detector in pp collisions at 13 TeV”, *Journal of Instrumentation*, vol. 13, no. 05, p. P05011, 2018.
- [65] V. Khachatryan *et al.*, “Measurement of $B\bar{B}$ Angular Correlations based on Secondary Vertex Reconstruction at $\sqrt{s} = 7$ TeV”, *Journal of High Energy Physics*, vol. 03, p. 136, 2011.
- [66] F. Chollet *et al.*, “Keras”, 2015. <https://keras.io>, last accessed on 2018-07-28.
- [67] M. Abadi *et al.*, “TensorFlow: Large-scale machine learning on heterogeneous systems”, 2015. <https://www.tensorflow.org/>, last accessed on 2018-07-28.
- [68] The CMS Collaboration, “CMS Phase 1 heavy flavour identification performance and developments”, May 2017.
- [69] D. Guest *et al.*, “Jet flavor classification in high-energy physics with deep neural networks”, *Physical Review D*, vol. 94, p. 112002, Dec 2016.
- [70] S. Hochreiter and J. Schmidhuber, “Long short-term memory”, vol. 9, pp. 1735–80, Dec 1997.
- [71] Wikimedia, “Long Short Term Memory”, 2015-10-04. https://commons.wikimedia.org/wiki/File:Long_Short_Term_Memory.png, last accessed on 2018-08-01.
- [72] T. Speer, “Jet Flavour Identification (MC Truth)”, 2018. <https://twiki.cern.ch/twiki/bin/view/CMSPublic/SWGuideBTagMCTools>, last accessed on 2018-07-31.

- [73] CERN, “Brilview” <https://brilview.readthedocs.io/en/stable/>, last accessed on 2018-08-03.
- [74] M. Czakon and A. Mitov, “Top++: A program for the calculation of the top-pair cross-section at hadron colliders”, *Computer Physics Communications*, vol. 185, no. 11, pp. 2930 – 2938, 2014.
- [75] R. Gavin, Y. Li, F. Petriello, and S. Quackenbush, “Fewz 2.0: A code for hadronic z production at next-to-next-to-leading order”, *Computer Physics Communications*, vol. 182, no. 11, pp. 2388 – 2403, 2011.
- [76] CMS, “GenXSecAnalyzer” <https://github.com/cms-sw/cmssw/blob/master/GeneratorInterface/Core/plugins/GenXSecAnalyzer.cc>, last accessed on 2018-09-06.
- [77] P. Kant *et al.*, “HatHor for single top-quark production: Updated predictions and uncertainty estimates for single top-quark production in hadronic collisions”, *Computer Physics Communications*, vol. 191, pp. 74–89, 2015.
- [78] M. Aliev *et al.*, “HATHOR: HAdronic Top and Heavy quarks crOss section calculatoR”, *Computer Physics Communications*, vol. 182, pp. 1034–1046, 2011.
- [79] J. M. Campbell, R. K. Ellis, and C. Williams, “Vector boson pair production at the lh_c”, *Journal of High Energy Physics*, vol. 2011, p. 18, Jul 2011.

Danksagung