Estimating information loss in LHC simulations: how to tackle the curse of dimensionality

Marius Sunde Sivertsen

Thesis submitted for the degree of
Master in Theoretical Physics
60 credits

Department of Physics
Faculty of mathematics and natural sciences

UNIVERSITY OF OSLO

Spring 2021

Estimating information loss in LHC simulations: how to tackle the curse of dimensionality

Marius Sunde Sivertsen

UNIVERSITAS OSLOENSIS

MDCCCXI

Estimating information loss in LHC simulations: how to tackle the curse of dimensionality

http://www.duo.uio.no/

Printed: Reprosentralen, University of Oslo

Abstract

In this project, we study the computationally challenging task of estimating the Kullback-Leibler divergence for high-dimensional probability distributions from particle physics. Our approach is based on using a trained classifier (a boosted decision tree) as a tool for dimensional reduction. As an interesting and challenging test case, we study simulated kinematic distributions for the production of supersymmetric particles at the Large Hadron Collider. We estimate the Kullback-Leibler divergence between kinematic distributions simulated at leading order and at next-to-leading order in perturbation theory, and find divergences of the order 10^{-2} bits for the studied examples.

Acknowledgments

First and foremost, I would like to give my most sincere gratitude to my supervisor Anders Kvellestad. Thank you for all the extraordinary effort making my scripts run correctly or being available nearly 24/7, and thank you for giving me this challenging and original thesis project! Thank you for all the interesting talks at the coffee machine, and for being an exceptional “kaffesjef” providing fresh coffee and other vital life supports for the theory group at the University of Oslo (UIO). Secondly, my co-supervisor Are Raklev has been a great support in arranging weekly meetings with a group of self promoted *Xsectioneers* from the theory group. Here we have shared our large range of difficulties, knowledge and exciting particle physics news across students involved in high energy physics research at UIO.

I also want to say thank you to all in the theory group at UIO for being so encouraging to love physics, so including and so helpful to understand the very non-trivial trivialities of a white introductory book in quantum field theory which title shall remain nameless. This project would not be possible without this great environment of people, and I will miss being a part of it on a day by day basis.

I would like to thank my mother, stepdad and my two sisters. You have always been proud of me, and encouraging me to pursue what I am passionate about. Thank you very much!

The last two years have been exceptional, exciting and very fruitful for my future. Time spent on what you are passionate about is always well spent time. During my time at UIO I have learned more about myself as a person, from the feeling of crushing failure to the wonderful feeling of accomplishments (and love). I have seen the huge value of having the opportunity to be educated, which in my opinion is an essential part in growing as a person.

Contents

Introduction

Given two *probability density functions* (pdfs) defined on the same space, how different are they from one another? A common way to quantify the difference between two pdfs $q(\mathbf{x})$ and $p(\mathbf{x})$ is the *Kullback-Leibler divergence* (KL divergence),

$$D_{KL}(p \,||\, q) = \int p(\mathbf{x}) \log \left[\frac{p(\mathbf{x})}{q(\mathbf{x})} \right],$$

from information theory. The KL divergence is a functional that takes two pdfs as input and essentially computes the integral over the ratio between p and q weighted with the p-distribution. However, in physics we often do not know the complete, analytical pdfs for the problem we are studying. If we can numerically generate samples from the pdfs, we can create histograms to use as an approximation for the pdfs. The question of how different the two pdfs are then becomes a question of the difference between two histograms.

Unfortunately, a problem arises when the pdfs are multi-dimensional – that is, when each sample is described by multiple variables. Populating multi-dimensional histograms is a computationally expensive task, and practically speaking not suitable for approximating pdfs in three dimensions or more. Therefore this naive approach is also not suitable for numerical evaluation of the pdf ratio $p(\mathbf{x})/q(\mathbf{x})$ that appears in the KL divergence. However, as shown in [1, sec. 2], it is possible to approximate a pdf ratio directly by using a classifier trained to differentiate samples from the two pdfs. This circumvents the need to populate two multi-dimensional histograms. With such a classifier at hand, we can compute the KL-divergence simply through Monte-Carlo integration using the p-samples.

As an interesting physics application, we will consider kinematic distributions for collision events at the Large Hadron Collider (LHC), when these distributions are simulated at leading order (LO) and next-to-leading order (LO+NLO) in perturbation theory. We will focus on the difference in the shapes of the LO and LO+NLO distributions across the kinematic space, and how these differences can be captured and mapped into a single number for information loss, namely the KL-divergence. Being able to accurately quantify the loss of information due to approximations is useful for LHC physics, in particular for theory studies, where simulations based on LO kinematics are often used due to the computational expense of NLO simulations. As a test case we will study a particular process from the Minimal Supersymmetric Standard Model (MSSM), which has long been a popular candidate theory for physics beyond the Standard Model of particle physics.

In the remainder of this thesis, we will in chapter 1 present how quantum field theory is used to analyze scattering experiments, and briefly introduce the Standard Model along with the MSSM. In chapter 2 we will discuss how to use information theory and statistical classifiers to quantify the overall difference of two unknown probability densities. As an application, we will consider the kinematic distributions at LO and at LO+NLO of an electroweak dislepton production process from the MSSM. We dedicate chapters 3 and 4 to explain the implementation of a boosted decision tree and the use of an event generator to construct appropriate kinematic datasets at LO and LO+NLO. The final results will be presented and discussed in chapter 5, and we end with a brief summary and outlook in chapter 6.

Chapter 1

Physics Background

Quantum field theory (QFT) is the framework in which modern theories of particle physics are formulated. It is constructed by combining the special theory of relativity with quantum mechanics, allowing us to describe fundamental physics through interactions between fields. These fields are known as *quantum fields* – operator-valued fields on spacetime. In the perturbative approach to QFT, computing interaction rates in a given theory are in principle done using perturbation theory on the free field solutions by adding an infinite number of diminishing correction terms – expanded in a power series of the physical coupling constants. However, the series must be truncated after a couple of terms due to the increasing computational complexity, resulting in approximated solutions.

In this chapter we will summarize how we understand particle scattering in QFT. Further, we will introduce the Standard Model of particle physics and discuss supersymmetry as a possible framework for physics beyond the Standard Model. This chapter, including conventions and notation for QFT, is based on [2, ch. 4] and [3, ch. 3].

1.1 Physics of Scattering Experiments

Scattering experiments have been key in the development of modern physics. From the 1911 discovery of the atomic nuclei by scattering alpha particles off a thin sheet of gold, to the 2012 discovery of the Higgs boson by proton-proton scattering in the LHC [4, 5], scattering experiments have been of key importance in our quest to understand the constituent parts of matter and the fundamental interactions.

The central quantity of interest in any scattering experiment is usually the *cross section* σ, which in a sense measures the effective size of scattering targets.

1.1.1 The Cross Section

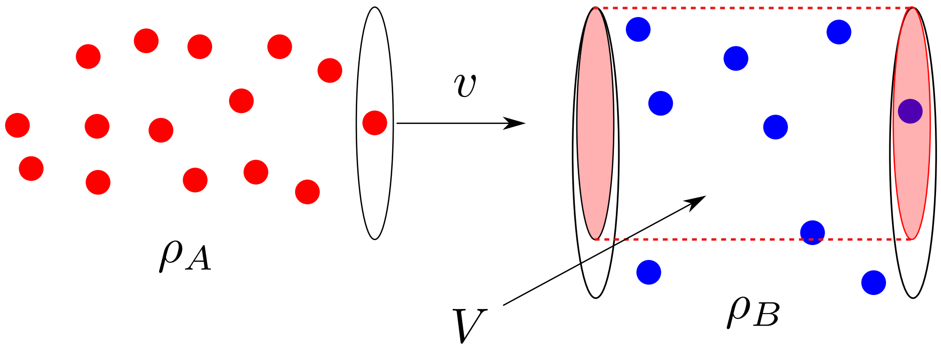


Figure 1.1: A cartoon to depict a bunch of particles of type A (red) with particle density ρ_A, and a bunch of particles of type B (blue) with particle density ρ_B. They are passing each other inside an interaction volume V, where we have picked a frame of reference where B is stationary and particles A are inbound with speed v.

In fig. 1.1 we can see a bunch of particles of type A with particle density ρ_A (number of particles per unit volume), and a bunch of particles of type B with particle density ρ_B. The two bunches will interact inside a volume V, referred to as an interaction volume. For convenience, we have picked a frame of reference where the particles of type A are moving with a speed v toward stationary particles of type B. Of course, any frame of reference can be picked where the particles will interact and collide at a speed $v = |v_A - v_B|$.

It is reasonable to expect that the number of scattering events per unit time (any type of event) is proportional to the rate of particles passing per unit area (incident flux) computed as $\phi_A = \rho_A v$, and proportional to the number of particles of type B within the overlapping area (red area in fig. 1.1), computed as $\rho_B V$. The number of scattering events per unit time is therefore

$$\dot{N} = \sigma\phi_A\rho_B V = v\rho_A\rho_B\sigma V, \tag{1.1}$$

where the proportionality constant σ is what we call the *cross section* for this scattering experiment. Note that this definition is symmetric in A and B, so σ would not change by putting A at rest and let B move inbound with speed v. The cross section σ has units of area, and classically it can be interpreted as an effective size of targets.

In reality, the number density in a beam of particles is typically not constant, where particles are mostly concentrated near the center. To get the count rate of scattering events in a real accelerator, simply integrate over the interaction volume V as

$$\dot{N} = \sigma \int_V \mathrm{d}^3x\, \phi_A(x)\rho_B(x) = \sigma v \int_V \mathrm{d}^3x\, \rho_A(x)\rho_B(x). \tag{1.2}$$

Moreover, detectors have deficiencies and a finite resolution which reduces the count rate by a factor $\epsilon < 1$ known as the detector efficiency. This is a highly important part of collider physics and deserves its own discussion, but that is beyond the scope of this thesis.

More importantly, we will see how we can use QFT to obtain a kinematic distribution of scattering events, which will be studied in more detail in what follows.

1.1.2 The Differential Cross Section and Kinematic Distributions

The count rate $\dot{N}$ from the previous section eq. (1.1) depends on the constant of proportionality σ defined as the cross section for that experiment. While all the other parts of eq. (1.1) describe the kinematic setup, σ is the quantity that captures the microscopic physics which is to say the interactions between the particles. We know from quantum mechanics that the microscopic physics in interactions are described as a superposition of specific processes that yield the same scattering outcome, with some processes being more likely than other. Thus, some scattering events will happen more often than other giving a non-trivial distribution of scattering events.

To formulate this properly, we can consider the cross section associated with a particular set of final state momenta, which will of course be infinitesimal. We write this as $\mathrm{d}^{3N}\sigma/(\mathrm{d}^3p_1 \cdots \mathrm{d}^3p_N)$, and it is simply the quantity that, when integrated over the small volume $\mathrm{d}^3p_1 \cdots \mathrm{d}^3p_N$, gives the cross section for scattering into that part of the momentum space. Do note however that four of the final state momenta will be set by 4-momentum conservation.

We now focus on a $2 \to N$ process which is typical for collider physics. In this case, all information about directional preference is captured in the *differential cross section*

$$\mathrm{d}^{3N}\sigma = \frac{1}{2E_A 2E_B|v_A - v_B|}\left(\prod_{i=1}^{N} \frac{\mathrm{d}^3p_i}{(2\pi)^3 2E_i}\right) \\ \times |\mathcal{M}(p_A, p_B \to \{p_f\})|^2 (2\pi)^4 \delta^4(p_A + p_B - \sum p_f). \quad (1.3)$$

Here, p_A, p_B, E_A and E_B are the momenta and energies of the initial particle states moving with relative speed $|v_A - v_B|$, and $\{p_f\}$ is the set of final state particle momenta with energies E_f. The object in eq. (1.3) that captures the details of the particle interactions in the scattering is $\mathcal{M}$, known as the *invariant matrix element.* The delta function at the end enforces 4-momentum conservation. This can be related to a particular set of global symmetries, as will be discussed in section 1.1.4.

Considering the transformation properties of eq. (1.3), the only object that transforms non-trivially under a Lorentz transformation is the prefactor

$$\frac{1}{E_A E_B |v_A - v_B|} = \frac{1}{|E_B p_A - E_A p_B|} = \frac{1}{|\epsilon_{\mu\nu xy} p_A^\mu p_B^\nu|},$$

which transforms exactly like a cross sectional surface area in the xy-plane being invariant under boosts along the conventionally chosen collision axis: the z-axis. Everything else is manifestly Lorentz invariant.

Phase Space of Final State Particles

To begin unraveling the rather intimidating eq. (1.3), we have to first understand its overall structure. Notice that the only part that depends on the physics of the interactions is contained solely within the invariant matrix element $\mathcal{M}$, while the other parts reflect universal physical constraints. This is an interesting point by itself which deserves a bit of attention – the distinction between *kinematics* and *dynamics.*

When a classical particle moves along some path $x^\mu(\tau)$ through spacetime, we want a full description of the motion along the path, *i.e.*, how the temporal x^0 and spatial coordinates x^i (i=$1, 2, 3$) change. This is known as kinematics. For convenience, we typically parameterize the path using the particle's proper time τ as the parameter since it is a Lorentz scalar[1]. The important point here is that this description is universal and independent of whatever caused the motion.

On the other hand, in physics we also attempt to understand and describe the causes of motion, that is, the fundamental interactions that influence a physical system. We refer to this as dynamics. This is where we attempt to quantify and understand the fundamental interactions that are present, giving a certain effect on a physical system. For a classical particle moving through spacetime, dynamics is concerned with how the conjugate coordinates of x^μ are affected by forces – that is, how the particle's 4-momentum p^μ is affected by external forces.

However, particles in QFT are not classical particles. *i.e.*, their state is not represented as their spacetime position x^μ and momentum p^μ. Rather, free particles are represented using quantum states. Their general state $|\phi\rangle$ as a wavepacket can be written as a super position of plane waves (momentum eigenstates) as

$$|\phi\rangle = \int \frac{\mathrm{d}^3k}{(2\pi)^3}\frac{1}{\sqrt{2E_\mathbf{k}}}\phi(\mathbf{k})\,|\mathbf{k}\rangle\,, \tag{1.4}$$

where $\phi(\mathbf{k})$ is the Fourier transform of the spatial wave function $\phi(\mathbf{x})$, and $|\mathbf{k}\rangle = \sqrt{2E_\mathbf{k}}\,|0\rangle$ is the associated momentum state with a proper relativistic normalization. The normalization $1/\sqrt{2E_\mathbf{k}}$ ensures that $\langle\phi|\phi\rangle = 1$ which is to say that all probabilities add up to 1.

Imagine now you have a $2 \to N$ process with N final state particles. How many momentum states are available in the range $[\mathbf{p_f}, \mathbf{p_f} + \mathrm{d}\mathbf{p_f}]$? By introducing a fictive box with

[1] That is, τ is invariant under Lorentz transformations since it can be written as an integral of a 4-vector contraction. Note that τ can only be used for massive particles.

side lengths a giving a volume $V = a^3$, we can expand the wave packets from eq. (1.4) as plane waves with their momenta being multiples of $2\pi/a$. Thus, each accessible state in momentum space occupies a tiny cube of size

$$\mathrm{d}^3p = \mathrm{d}p_x \mathrm{d}p_y \mathrm{d}p_z = \left(\frac{2\pi}{a}\right)^3 = \frac{(2\pi)^3}{V}. \tag{1.5}$$

Keep in mind that the volume V will not show up in any physical calculations since it is only used to parameterize the available states. In a physical calculation, the volume dependence on the phase space element will cancel with the normalization of the wave function within the box of volume V. Since it will not show up in the final answer, we can simply put $V = 1$ to get rid off it. This will normalize the phase space volume to have 1 particle state per unit volume resulting in

$$\mathrm{d}\Pi = \frac{\mathrm{d}^3p}{(2\pi)^3 2E} \tag{1.6}$$

number of available states within the infinitesimal volume d^3p_i. As explained in eq. (1.4), the wave function comes with a conventional factor $\sqrt{2E}$ which is compensated here by dividing by $2E$ in the phase space element (intuitively, this compensates for the Lorentz contraction $1/\gamma \sim 1/E$ of the volume V after a boost). With N final state particles, the number of states becomes

$$\prod_{i=1}^{N} \frac{d^3p_i}{(2\pi)^3 2E_i}. \tag{1.7}$$

However, how many degrees of freedom (dof) are there with N final state particles? Without any constraints, each particle represents 3 dof, which gives $3N$ dof overall. But conservation of 4-momentum introduces four constraints, leaving a total of $3N - 4$ degrees of freedom. By using a four-dimensional delta-function to account for the 4-momentum conservation, the final *Lorentz invariant phase space* (LIPS) element for a $2 \to N$ process becomes

$$\mathrm{d}\Pi_N = \left[\prod_{i=1}^{N} \frac{\mathrm{d}^3p_i}{(2\pi)^3 2E_i}\right] \delta^4\left(p_A + p_B - \sum_{i=1}^{N} p_i\right)(2\pi)^4. \tag{1.8}$$

One nice feature of eq. (1.8) is that it is manifestly Lorentz invariant by construction, allowing us to compute the LIPS in any frame we like which is very convenient for practical reasons.

Combining eq. (1.8) and eq. (1.3) allows us to rewrite the differential cross section as

$$\mathrm{d}^{3N}\sigma = \frac{1}{2E_A 2E_B |v_A - v_B|} \mathrm{d}\Pi_N |\mathcal{M}(p_A, p_B \to p_1, \ldots, p_N)|^2. \tag{1.9}$$

Writing it this way illuminates its structure more clearly: the differental cross section for scattering to final states with momenta in the range $[\mathbf{p}_f, \mathbf{p}_f + d\mathbf{p}_f]$ is proportional to the number of available such states and the square of the amplitude for transition to these states. As always in quantum mechanics, the probability to start in an initial state $|i\rangle$ and end up in a final state $|f\rangle$ is simply given by the square of their "overlap", *i.e.*, their inner product squared $|\langle f|i\rangle|^2$. Applying this to the two multi-particle states where $|\phi_A, \phi_B\rangle$ is the initial state and $|\phi_1, \ldots, \phi_N\rangle$ is the final state, the transition probability is computed as

$$P(\mathbf{p}_A, \mathbf{p}_B \to \mathbf{p}_1, \ldots, \mathbf{p}_N) = |\langle \phi_1, \ldots, \phi_N|\phi_A, \phi_B\rangle|^2 \propto |\langle \mathbf{p}_1, \ldots, \mathbf{p}_N|\mathbf{p}_A, \mathbf{p}_B\rangle|^2, \tag{1.10}$$

which is the starting point to compute the invariant matrix element $\mathcal{M}(p_A, p_B \to \{p_f\})$ and the full $2 \to N$ differential cross section starting from eq. (1.1). The arguments up until now are a big part of deriving eq. (1.3), but we will not complete the full derivation here since it is a standard derivation found in many text books on QFT or particle physics. For the full derivation, see for instance [2, sec. 4.5] or [3, sec. 3.4].

Example: General Two-Body Process

As an example on how to apply eq. (1.3), let us consider a special case where there are two final state particles ($2 \to 2$) and evaluate the differential cross section in the center-of-mass (CM) frame, *i.e.*, the frame where the total initial 3-momentum is $\mathbf{p_A} + \mathbf{p_B} = \mathbf{0}$.

Computing the differential cross section in eq. (1.3) involves computing the invariant matrix element $\mathcal{M}$ which can be a complicated function of the final state momenta. However, due to momentum conservation from all the delta functions, there are a couple of simplification we can do by partially evaluating the integrals of the phase space element from eq. (1.8). Labeling the final momenta as p_1 and p_2, we can immediately do the integration over the three components of $\mathbf{p_2}$ using the three delta functions forcing $\mathbf{p_2} = -\mathbf{p}_1$, expected from 3-momentum conservation. The integral over the three remaining momentum coordinates $\mathrm{d}^3p_1 = \mathrm{d}|\mathbf{p}_1|\,|\mathbf{p}_1|^2\mathrm{d}\Omega$ of the phase space element $\mathrm{d}\Pi_2$ reduces to

$$\int \mathrm{d}\Pi_2 = \int \frac{\mathrm{d}|\mathbf{p}_1|\,|\mathbf{p}_1|^2\mathrm{d}\Omega}{(2\pi)^3 2E_1 2E_2} 2\pi\delta(E_{\mathrm{CM}} - E_1 - E_2), \tag{1.11}$$

where $E_1 = \sqrt{m_1^2 + |\mathbf{p}_1|^2}$, $E_2 = \sqrt{m_2^2 + |\mathbf{p}_1|^2}$ and $E_{\mathrm{CM}} = E_A + E_B$ is the total initial energy. To compute the last integral over the final delta function where the argument is a function of $|\mathbf{p}_1|$, we can use the identity

$$\delta(g(x)) = \sum_i \frac{1}{|g'(x_i)|}\delta(x - x_i) \tag{1.12}$$

where the sum is over all the zeros x_i of a differentiable function g, assuming $g(x_i) \neq 0$ for all the zeros. Applying this identity to the delta function in eq. (1.11) with $g(p) = E_{\mathrm{com}} - \sqrt{m_1^2 + p^2} - \sqrt{m_2^2 + p^2}$, we see that the only zero is at $p = |\mathbf{p}_1|$ with a derivative

$$\left.\frac{\mathrm{d}g}{\mathrm{d}|\mathbf{p}_1|}\right|_{p=|\mathbf{p}_1|} = -\left(\frac{|\mathbf{p}_1|}{E_1} + \frac{|\mathbf{p}_1|}{E_2}\right) \tag{1.13}$$

which immediately simplifies the phase space integral to

$$\begin{aligned} \int \mathrm{d}\Pi_2 &= \int \mathrm{d}\Omega \frac{|\mathbf{p}_1|^2}{16\pi^2 E_1 E_2}\left(\frac{|\mathbf{p}_1|}{E_1} + \frac{|\mathbf{p}_1|}{E_2}\right)^{-1} \\ &= \int \mathrm{d}\Omega \frac{1}{16\pi^2}\frac{|\mathbf{p}_1|}{E_{\mathrm{CM}}}. \end{aligned} \tag{1.14}$$

If the reaction is symmetric about the collision axis (azimuthal symmetry), the integral over ϕ is trivial giving an extra factor 2π, *i.e.*,

$$\int \mathrm{d}\Pi_2 = \int \mathrm{d}(\cos\theta)\,\frac{1}{8\pi}\frac{|\mathbf{p}_1|}{E_{\mathrm{CM}}}. \tag{1.15}$$

Having the two-body phase space at hand, the differential cross section eq. (1.3) simplifies to

$$\left(\frac{\mathrm{d}\sigma}{\mathrm{d}\Omega}\right)_{\mathrm{CM}} = \frac{1}{E_A E_B |v_A - v_B|}\frac{|\mathbf{p}_1|}{64\pi^2 E_{\mathrm{CM}}}|\mathcal{M}(p_A, p_B \to p_1, p_2)|^2. \tag{1.16}$$

In the case we can neglect the masses of the initial and final state particles, making $E_A = E_B = |\mathbf{p}_1| = E_{\mathrm{CM}}/2$, the differential cross section simplifies even further to

$$\left(\frac{\mathrm{d}\sigma}{\mathrm{d}\Omega}\right)_{\mathrm{CM}} = \frac{|\mathcal{M}|^2}{64\pi^2 E_{\mathrm{CM}}^2}. \tag{1.17}$$

This is quite a simplification starting from eq. (1.3), and a practical result used in many situations.

The total cross section σ is obtained by simply integrating over the remaining phase space variables (like Ω), keeping in mind that if there are n identical particles in the final state, σ has to be divided by $n!$. This is because identical quantum particles are indistinguishable making eq. (1.3) overcount by a factor $n!$ because there will be $n!$ identical final states.

Kinematic Distributions

At last, we will introduce the *kinematic distribution* for a $2 \to N$ process. This is defined as the normalized differential cross section from eq. (1.9)

$$f(\mathbf{X}) = \frac{1}{\sigma}\frac{\mathrm{d}^{3N}\sigma}{\mathrm{d}^{3N}X}, \tag{1.18}$$

where $\mathbf{X}$ is a tuple of $3N$ kinematic variables, where we keep in mind that four of these will be fully determined by 4-momentum conservation. Notice that this object integrates to 1 by construction. It can be interpreted as the conditional probability distribution for the N particles to scatter into the specific kinematic configuration $\mathbf{X}$, given that a $2 \to N$ process is taking place. The kinematic distribution will be the main object of interest in our study.

1.1.3 Perturbative Computation of Cross Sections

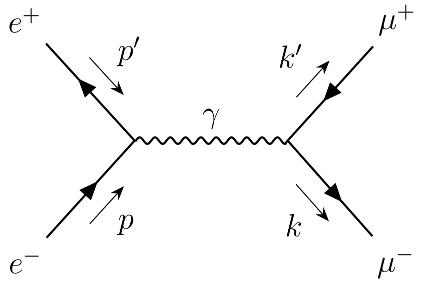
The perturbative approach to QFT gives us an elegant and systematic way to compute the invariant matrix element $\mathcal{M}(\{p_i\} \to \{p_f\})$ for particle processes. As demonstrated by Feynman, the perturbative contributions to $\mathcal{M}$ can be represented as graphs (now known as Feynman diagrams) consisting of simple edges connected by vertices. For every vertex and edge, there is a rule (now known as Feynman rules) that tells us how to translate that part of the diagram into a mathematical expression. See [2, sec. 4.4] for more details and the motivation behind Feynman diagrams.

To give an illustrative example, let us consider a classic process from the QFT of interactions between fermions and the electromagnetic field, known as *quantum electrodynamics* (QED).

Pair annihilation of two electrons

$$e^+e^- \to \mu^+\mu^- \tag{1.19}$$

to lowest order in perturbation theory is given by the diagram below. This process produces two final state particles, so there are $3 \times 2 - 4 = 2$ dof. To keep the focus on the kinematic distribution of this process we will not compute the diagram in detail like it is done in [2, p. 131-136]. The square of the invariant matrix element $\mathcal{M}$ averaged over the four possible initial state spin configurations is



$$\frac{1}{4}\sum_{\text{spins}} |\mathcal{M}|^2 = \frac{8e^4}{(p+p')^4}\big[(p\cdot k)(p'\cdot k' + (p\cdot k')(p'\cdot k) + m_\mu^2(p\cdot p'))\big], \tag{1.20}$$

with m_μ being the mass of the muon (electron masses are neglected) and e is the elementary charge unit. With the amplitude squared given, it is easy to evaluate the differential cross section in the center of mass frame using eq. (1.16),

$$\frac{\mathrm{d}\sigma}{\mathrm{d}\Omega} = \frac{\mathrm{d}^2\sigma}{\mathrm{d}\phi\,\mathrm{d}(\cos\theta)} = \frac{\alpha^2}{4s}\sqrt{1-\frac{m_\mu^2}{E^2}}\left[\left(1+\frac{m_\mu^2}{E^2}\right)+\left(1-\frac{m_\mu^2}{E^2}\right)\cos^2\theta\right], \tag{1.21}$$

where $s = E_{\text{CM}}^2 = 4E^2$ with E being the energy of the initial electron e^- (or e^+) and $\alpha = e^2/4\pi$ is the QED coupling constant. Equation (1.21) does not depend on the azimuthal angle ϕ, allowing us to immediately write down

$$\frac{\mathrm{d}\sigma}{\mathrm{d}(\cos\theta)} = 2\pi\frac{\mathrm{d}\sigma}{\mathrm{d}\Omega}. \tag{1.22}$$

To end up with a kinematic distribution, we can simplify the analysis by considering the cross section in the high energy limit $E \gg m_\mu$, *i.e.*, ,

$$\frac{\mathrm{d}\sigma}{\mathrm{d}\Omega} = \frac{\alpha^2}{4s}(1+\cos^2\theta). \tag{1.23}$$

It is now simple to compute the total cross section σ by integrating eq. (1.22) over $\cos\theta$ from -1 to 1, *i.e.*, θ from 0 to π giving

$$\sigma = \frac{4\pi\alpha^2}{3s}. \tag{1.24}$$

At last, combining the last two equations gives us the full two-dimensional kinematic probability distribution of final state particles in the high energy limit as

$$f(\phi, \theta) = \frac{1}{\sigma}\frac{\mathrm{d}^2\sigma}{\mathrm{d}\phi\,\mathrm{d}(\cos\theta)} = \frac{3}{16\pi}(1 + \cos^2\theta), \tag{1.25}$$

by using eq. (1.18) directly. For completeness, we can integrate out the trivial azimuthal angle ϕ to give an extra factor 2π, resulting in the one-dimensional distribution

$$f(\theta) = \frac{1}{\sigma}\frac{\mathrm{d}\sigma}{\mathrm{d}(\cos\theta)} = \frac{3}{8}(1 + \cos^2\theta). \tag{1.26}$$

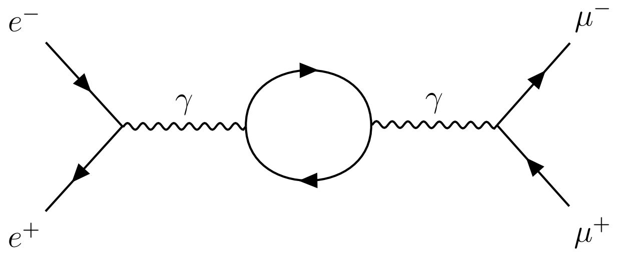
This is our first result of a kinematic distribution – a perfectly valid probability distribution which will be an object of high interest in this project.

We have just seen what is known as a *leading order* (LO) computation of the cross section of the QED process above. Keep in mind that, due to the perturbative approach to compute the matrix element $\mathcal{M}$, this is just an approximation of the “real” cross section. In general, the matrix element $\mathcal{M}$ has an infinite number of correction terms, and it is expanded as a power series in the coupling constant α as

$$\mathcal{M} = \mathcal{M}_{\mathrm{LO}} + \mathcal{M}_{\mathrm{NLO}} + \mathcal{M}_{\mathrm{NNLO}} + \ldots, \tag{1.27}$$

where $\mathcal{M}_{\mathrm{LO}} \sim \alpha$, $\mathcal{M}_{\mathrm{NLO}} \sim \alpha^2$ called *next-to-leading order*, $\mathcal{M}_{\mathrm{NNLO}} \sim \alpha^3$ called *next-to-next-to-leading order* and so forth. These higher orders are represented as more complicated Feynman diagrams involving more particles. For instance, the diagram below is a NLO correction proportional to α^2 to the process above. This is known as a *one-loop diagram* where there is a quantum correction to the exchange photon from an intermediate fermion/anti-fermion loop.

Since the coupling constant α is small for this process, the LO approximation of the cross section in eq. (1.21) from a single diagram is a good approximation. Diagrams without loops are often called *tree-level diagrams*.



In section 1.3.2, we will see the LO and NLO diagrams for the process we are studying in this project, and also address some complications that always show up with higher order corrections.

1.1.4 Symmetries and Conservation Laws

Many properties of physics and fundamental particles can be understood from certain symmetries that are present in the universe. A symmetry operation on an object, broadly speaking, is any type of transformation that leaves that object unchanged. In classical field theory, we are interested in symmetries that leave either the Lagrangian or the associated equations of motion unchanged.

As an example, consider the massless free scalar field Lagrangian

$$\mathcal{L} = \frac{1}{2}\partial_\mu\phi(x)\partial^\mu\phi(x) = \frac{1}{2}(\partial_\mu\phi(x))^2 \tag{1.28}$$

of a single kinetic term in ϕ. If we shift the spacetime position x^μ by a small amount a^μ, that is to say we transform

$$x^\mu \to x^\mu + a^\mu,$$

which induces a change in ϕ by an amount

$$\Delta\phi(x) = \phi(x+a) - \phi(x) = a^\mu\partial_\mu\phi(x) + \mathcal{O}\big(a^2\big). \tag{1.29}$$

What is the change in the Lagrangian? In general, from the Taylor expansion of $\mathcal{L}$ we have

$$\mathcal{L}(x+a) = \mathcal{L}(x) + \Delta\phi\frac{\partial\mathcal{L}}{\partial\phi} + (\partial_\mu\Delta\phi)\frac{\partial\mathcal{L}}{\partial(\partial_\mu\phi)} + \mathcal{O}\big(\Delta\phi^2\big), \tag{1.30}$$

where we can rewrite the third term using the product rule of differentiation

$$(\partial_\mu \Delta\phi)\frac{\partial\mathcal{L}}{\partial(\partial_\mu\phi)} = \partial_\mu\left(\Delta\phi\frac{\partial\mathcal{L}}{\partial(\partial_\mu\phi)}\right) - \partial_\mu\frac{\partial\mathcal{L}}{\partial(\partial_\mu\phi)}\Delta\phi.$$

Inserting this back in to eq. (1.30) and using the Euler-Lagrange equations to cancel the terms proportional with $\Delta\phi$, this allows us to write $\Delta\mathcal{L}$ as

$$\Delta\mathcal{L}(x) = \mathcal{L}(x+a) - \mathcal{L}(x) = \partial_\mu\left(\Delta\phi\frac{\partial\mathcal{L}}{\partial(\partial_\mu\phi)}\right) + \mathcal{O}\big(\Delta\phi^2\big), \tag{1.31}$$

saying that the change in the Lagrangian due to a small change in the field configuration $\phi(x)$ can in general be written as a total derivative. Thus, the $\mathcal{L}$ will always transform as

$$\mathcal{L} \to \mathcal{L} + \partial_\mu\mathcal{J}^\mu \tag{1.32}$$

for some $\mathcal{J}^\mu$. In the derivation of the equations of motion (Euler-Lagrange equations) from varying the action, surface terms do not contribute assuming that the fields vanishes at infinity. Since total derivatives can be written as surface terms evaluated at infinity through Gauss' divergence theorem, the equations of motion due to $\Delta\mathcal{L}$ are left unchanged.

We can use $\mathcal{J}^\mu(x)$ to define a *conserved current* $j^\mu(x)$ such that

$$\partial_\mu j^\mu(x) = 0 \quad \text{for} \quad j^\mu(x) = \frac{\partial\mathcal{L}}{\partial(\partial_\mu\phi)}\Delta\phi(x) - \mathcal{J}^\mu(x). \tag{1.33}$$

The zeroth component j^0 is often called a charge density, while the other components j^1, j^2 and j^3 make up the current flux density. We can define the *charge* of a conserved current as

$$Q(t) = \int \mathrm{d}^3x\, j^0(t,\mathbf{x}), \tag{1.34}$$

and due to eq. (1.33), it follows that

$$\begin{aligned}\frac{\mathrm{d}Q(t)}{\mathrm{d}t} &= \frac{\mathrm{d}}{\mathrm{d}t}\int \mathrm{d}^3x\, j^0(t,\mathbf{x}) \\ &= \int \mathrm{d}^3x\, \partial_0 j^0(t,\mathbf{x}) \\ &= -\int \mathrm{d}^3x\, \partial_i j^i(t,\mathbf{x}) \\ &= -\oint_{\partial S(\infty)} \mathrm{d}^2x\,(\hat{\mathbf{n}}\cdot\mathbf{j}(t,\mathbf{x}))\end{aligned} \tag{1.35}$$

where we have applied Gauss' divergence theorem in the last line. The surface integral on the last line vanishes since the fields vanishes at infinity. Thus, the charge $Q(t)$ is conserved at any time in any enclosed volume in space, which is an example of a conservation law derived from an associated conserved current $j^\mu(x)$.

The free-field Lagrangian eq. (1.28) after transforming ϕ with eq. (1.29) using eq. (1.31) can be simplified and rewritten into

$$\begin{aligned} \mathcal{L}(x+a) &= \mathcal{L}(x) + \partial_\mu \left(\Delta\phi \frac{\partial \mathcal{L}}{\partial(\partial_\mu \phi)} \right) = \mathcal{L}(x) + a^\mu \partial_\mu \mathcal{L}(x) \\ &= \mathcal{L}(x) + a^\nu \partial_\mu (\delta^\mu_\nu \mathcal{L}(x)) \end{aligned} \tag{1.36}$$

which is exactly the form of eq. (1.32) with $(\mathcal{J}^\mu(x))_\nu = \delta^\mu_\nu \mathcal{L}(x)$. This allows us to define four conserved currents

$$(j^\mu)_\nu \equiv T^\mu{}_\nu = \frac{\partial \mathcal{L}}{\partial(\partial_\mu \phi)} \partial_\nu \phi - \delta^\mu_\nu \mathcal{L} \tag{1.37}$$

where T is an object known as the *stress-energy tensor* of the field ϕ. The four associated conserved charges are

$$\begin{aligned} \nu = 0: \quad & H = \int \mathrm{d}^3x \, T^0{}_0 = \int \mathrm{d}^3x \left[\pi(t,\mathbf{x})\dot{\phi}(t,\mathbf{x}) - \mathcal{L}(t,\mathbf{x}) \right] = \int \mathrm{d}^3x \, \mathcal{H}(t,\mathbf{x}) \\ \nu = i: \quad & P^i = \int \mathrm{d}^3x \, T^{0i} = \int \mathrm{d}^3x \, \pi(t,\mathbf{x}) \partial^i \phi(t,\mathbf{x}), \end{aligned} \tag{1.38}$$

where $\pi(x) \equiv \partial\mathcal{L} \big/ \partial\dot{\phi}$ is the physical *momentum density* of the field ϕ and $\mathcal{H}$ is the physical energy density. Thus, the momentum and energy associated with the field ϕ is conserved in time, which establishes the well known conservation law of 4-momentum.

The original result, proved by Emmy Noether in 1915, states that there is a conservation law associated with every continuous symmetry in the Lagrangian or in the associated equations of motion. This result is known as *Noether's theorem*, and it plays a fundamental part in the description of fundamental physics through conservation laws.

1.2 The Standard Model

The Standard Model of particle physics (SM) is currently the most complete theory of the fundamental particles, describing all visible matter. It gives a consistent and accurate description of three of the four fundamental forces: the electromagnetic, the weak and the strong force. Gravity remains unaccounted for, but the predictive power of the SM remains solid since the effects of gravity are expected to be negligible up to the *Planck scale* (10^{19} GeV). Most recent high energy particle physics experiments have only achieved energies on the order of 10^4 GeV.

The main ingredients of the SM as a QFT is the *Dirac equation* describing the properties and dynamics of fermions. It also uses the *gauge principle* as a way to formulate and understand the interactions, identifying each class of interactions with a local symmetry of the SM Lagrangian. At last, non-zero particle masses, which would naively spoil these local gauge symmetries in the theory, are explained via the *Higgs mechanism* of spontaneous electroweak symmetry breaking. Here, the necessary mass terms in the Lagrangian are dynamically generated from an underlying, gauge-invariant Lagrangian. We discuss these ideas in some more detail below.

1.2.1 Fundamental Forces

The standard model is an example of a gauge theory which means that the Lagrangian is postulated to respect additional symmetries above the standard Lorentzian symmetries. These extra symmetries are stronger in the sense that they are defined locally, forcing the quantum fields to transform in a particular way from point to point on spacetime. To ensure this is the case, we are forced to introduce additional quantum fields giving rise to spin-1 particle states, namely the force mediating gauge bosons. These bosons are often refereed to as "force carriers" which mediate forces between the SM particles.

The gauge symmetry group of the SM is

$$SU(3)_C \times SU(2)_L \times U(1)_Y. \tag{1.39}$$

The subscripts indicate which fields have non-trivial transformations under the different symmetries: The subscript C in $SU(3)_C$ indicates that $SU(3)_C$ transformations affect fields with non-zero color charge; the L in $SU(2)_L$ means that these transformations impact left-chiral fields; and the Y in $U(1)_Y$ associate $U(1)_Y$ transformations with fields with non-zero weak hypercharge. The weak hypercharge Y is related to the electric charge

Q through the third component of the isospin I_3 by $Y = 2(Q - I_3)$.

While the electromagnetic force and the weak force are considered separately, they were unified through the work of Salam, Glashow and Weinberg in the 1960s into a more fundamental force known as the *electroweak force*. Described by the gauge group $SU(2)_L \times U(1)_Y$, this was one of the major steps towards the gauge group of the SM (eq. (1.39)), predicting the existence of four massless gauge bosons W_1, W_2, W_3 and B. The bosons had to be exactly massless to respect the gauge symmetry, but experiments suggested that three of the gauge bosons had to be massive to match the data. This forced the theorists to introduce the concept of *spontaneous symmetry breaking* of the electroweak symmetry ($SU(2)_L \times U(1)_Y \rightarrow U(1)_{EW}$), which allows W_1, W_2, W_3 and B to mix and form exactly three massive particle states and one massless state. Sure enough, these states are the observed weak gauge bosons $W^\pm$ and Z, and the observed photon γ. The breaking of electroweak symmetry was made possible by predicting the existence of a scalar field with a vacuum state that does not necessarily respect this symmetry, with the effect of creating massive gauge bosons. This scalar field is known as the *Higgs field* and gives rise to a spin-0 boson known as the *Higgs boson*, which was discovered experimentally by the ATLAS and CMS experiments at the LHC in 2012. The Higgs boson interacts with all massive particles in the SM, including itself.

1.2.2 Matter Particles

The SM predicts a number of spin-1/2 particles (fermions) as the fundamental building blocks of matter, interacting through the forces introduced above. There are two classes of fermions, namely *leptons* and *quarks*. The leptons include the electron $e^\pm$ and its associated electron neutrino followed by its two "heavier siblings" the muon $\mu^\pm$ and the tau $\tau^\pm$ with their associated neutrinos. We refer to their charge as ± to also include the associated anti-particles. Similarly, the quarks are arranged in three different classes from lightest to heaviest, and together with the leptons they make up the three different *generations* of the SM.

Particles are collected together in vectors under unitary representations of the gauge groups. If we consider $SU(2)_L$ for a moment, the fundamental representation of that group is built up of three 2×2-matrices which acts upon two-component vectors known as *doublets*. For instance, the muon μ^- and its neutrino ν_μ form a doublet under $SU(2)_L$ on the form

$$\begin{pmatrix} \mu_L \\ \nu_{\mu,L} \end{pmatrix},$$

meaning that you can apply any combination of these three 2×2-matrices on that doublet without changing the SM Lagrangian. Since the representation of $SU(2)$ is unitary, doublets are simply rotated around inside this three-dimensional space implying that an inner product of two doublets is left invariant. This also explains why the SM Lagrangian is invariant because all the doublet terms are purely built up of inner products. The same can be said about the two other gauge groups under their unitary representations.

1.3 Beyond the Standard Model

While SM has passed numerous experimental tests over several orders of magnitude in energy, it has shortcomings that leave us with several open questions. Below we will highlight two such important open questions, before we discuss supersymmetry as a framework for going beyond the Standard Model, and introduce the specific scattering process we will study. The theory discussions in this chapter are based on [6].

The Hierarchy Problem

The hierarchy problem stands as one of the most peculiar problems of the SM. Here the general question is why there seems to be a fine tuning of the model parameters. For instance, let us consider the physical mass of the Higgs boson. Theoretically, the Higgs mass is related to its bare mass[2] as

$$m_H^2 = (m_H^0)^2 + \Delta m_H^2, \tag{1.40}$$

[2] That is, the mass parameter obtained before renormalizing it to its real physical value.

where m_H^0 is the bare mass and Δm_H represents the one-loop corrections (or quantum corrections) to the mass from all the loop diagrams with massive fermions, bosons and even itself.

The one-loop correction to m_H^2 due to a massive fermion f with a momentum cut off at a scale Λ takes the form

$$(\Delta m_H^2)_f = -\frac{|\lambda_f|^2}{8\pi^2}\Lambda^2 + \ldots, \tag{1.41}$$

where λ_f is the Yukawa coupling of the fermion f with the Higgs. Since this coupling is proportional to the fermion mass, the largest contribution would be from the top quark, being the heaviest of all the SM particles.

The one-loop corrections with the same cut off Λ to the squared Higgs mass m_H^2 from a scalar particle S takes the form

$$(\Delta m_H^2)_S = \frac{\lambda_S}{16\pi^2}\Lambda^2 + \ldots, \tag{1.42}$$

where λ_S is the coupling of the scalar to the Higgs.

The Λ parameter can be interpreted as the scale where new physics will probably be important, which, if the Standard Model is a complete description of non-gravitational quantum physics, can be as large as the Planck scale, 10^{19} GeV. Since the correction to the Higgs mass squared goes as the square of the momentum scale, it is rather surprising that the Higgs mass is as low as it is when we expect huge quantum corrections. That is, within the Standard Model we would theoretically expect the Higgs mass to be comparable with some very high scale of new physics, but from experiment we know that it is around 125 GeV. From eq. (1.40), the only way this can happen within the Standard Model is if the bare mass $(m_H^0)^2$ is extremely *fine tuned* to a particular value, causing a massive cancellation with the loop corrections Δm_H^2.

Another hierarchy problem in the SM is related to why gravity is so much weaker than the weak force, differing by 24 orders of magnitude in their respective coupling strength.

Dark Matter

According to astrophysical observations, there are large amounts of weakly-interacting matter in the universe known as *dark matter*. The evidence for dark matter includes anomalous rotational curves in all galaxies[3] and gravitational lensing effects in a seemingly void of space suggesting a presence of invisible mass. The only possible candidates from the SM are the neutrinos, and despite being the most abundant class of particles in our universe, they are too light to fit the observational data. This has led astronomers and cosmologists to suggest that there might exist particles beyond the SM, often called *non-baryonic matter*.

1.3.1 Supersymmetry

In light of the problems above and others, search for physics beyond the SM has been going on for decades without any luck so far. Nevertheless, numerous theories have been suggested, and among the most popular ones are theories based on the idea of *supersymmetry* (SUSY).

The massive cancellation of the Higgs mass corrections discussed in the hierarchy problem in eq. (1.40) suggests a more appealing solution – that there is an underlying symmetry unaccounted for in the SM. Notice the relative minus sign between the scalar one-loop correction eq. (1.42) compared to the one-loop fermion correction eq. (1.41). Imagine now for the sake of argument that there is a new symmetry relating bosons and fermions. If we were to introduce two new scalars for every fermion in the SM, with $\lambda_S = |\lambda_f|^2$, notice now how all the loop corrections would perfectly cancel. This cancellation of the Higgs mass corrections, consistent with the measured value, is one appealing reason to postulate the existence of this fermion-boson symmetry – known as supersymmetry. This symmetry transforms bosons into fermions and vice versa, and it is regarded as a non-trivial extension of the spacetime symmetries. We will briefly discuss this extension below to give some more insight.

[3] Large discrepancies with theoretical predictions of the tangential speed of stars in a galaxy as a function of distance from the center.

All transformations that leave the spacetime interval $(x-y)^2$ unchanged form a group known as the *Poincaré group*. It is the group of all transformations on the form

$$x^\mu \to \Lambda^\mu{}_\nu x^\nu + a^\mu, \tag{1.43}$$

where a^μ is a constant displacement in spacetime, and $\Lambda^\mu{}_\nu$ are the components of the Lorentz transformations. These transformations are known as the *spacetime symmetries*. The generators of Lorentz transformations, $\mathcal{M}_{\mu\nu}$, and the generators for translation, P^μ, satisfy the so called *Poincaré algebra* summarized by

$$[P^\mu, P^\nu] = 0, \tag{1.44}$$

$$[M^{\mu\nu}, P^\sigma] = i(g^{\nu\sigma}P^\mu - g^{\mu\sigma}P^\nu) \tag{1.45}$$

$$[M^{\mu\nu}, M^{\rho\sigma}] = i(g^{\nu\rho}M^{\mu\sigma} + g^{\mu\sigma}M^{\nu\rho} - g^{\nu\sigma}M^{\mu\rho} - g^{\mu\rho}M^{\nu\sigma}). \tag{1.46}$$

If we want to extend the spacetime symmetries in a non-trivial way, *i.e.*, what other generators can possibly exist that do not trivially commute with $\mathcal{M}_{\mu\nu}$ and P^μ? It turns out, due to the work of Coleman and Mandula [7], that the only possibility is to introduce a pair of anti-commuting operators, Q_α and its adjoint $Q^\dagger_{\dot\alpha}$, where $\alpha, \dot\alpha = 1, 2$ are two distinct indices. These operators are fermionic by nature (anti-commuting), and they can be represented as two-component spinors[4] acting on Dirac spinors. The fundamental commutation relations with the spacetime generators above are shown in [6, sec. 3.1].

Nevertheless, the effect of applying Q and $Q^\dagger$ to particle states (bosons or fermions) is to change the spin quantum number by ±1/2 – effectively mapping fermions to bosons and vice versa. Qualitatively, the action of these *SUSY operators* can be summarized by

$$Q\,|\text{fermion}\rangle = |\text{boson}\rangle \quad \text{and} \quad Q\,|\text{boson}\rangle = |\text{fermion}\rangle\,. \tag{1.47}$$

However, SUSY can not be an exact symmetry of nature, as this would require the new "superpartner" particles predicted by SUSY to have the same masses as their corresponding SM particles. Clearly we have not observed any such particles, implying that, if they exist, they must be heavier. However, there are good reasons to expect that they are not much heavier than their SM partners. The reason is that the heavier the SUSY particles are, the less successful the cancellations between bosons and fermions in the hierarchy problem become.

[4] Also known as Weyl spinors from the Weyl representation of the Poincaré group.

1.3.2 The Minimal Supersymmetric Standard Model

We will now discuss a minimal SUSY-extension of the SM, based on extending the Poincaré group from eq. (1.43) with one SUSY generator Q and its conjugate, $\bar{Q}$. Both the spacetime symmetry generators and the SUSY generators commute with the gauge symmetry generators of the SM, allowing us to use the SM gauge symmetry group as it is for this theory. Since SUSY is a broken symmetry (not an exact symmetry of nature), a viable Lagrangian consists of a SUSY-invariant part ($\mathcal{L}_{\text{SUSY}}$) and a part with SUSY breaking terms ($\mathcal{L}_{\text{soft}}$),

$$\mathcal{L} = \mathcal{L}_{\text{SUSY}} + \mathcal{L}_{\text{soft}}. \tag{1.48}$$

This theory is known as the *Minimal Supersymmetric Standard Model* (MSSM), and it predicts a whole set of new particles due to the effect of eq. (1.47). This is because there are obviously no SM particles that differ in spin by ±1/2 while keeping all other quantum numbers the same. This implies that there must be other particles with these properties if the MSSM is a correct theory of nature.

Field Content

In this thesis, we will restrict ourselves to two classes of SUSY particles (or *sparticles*) predicted by the MSSM. New scalar particles are named by prepending an “s” to the SM particle name, while new fermion states are given names with an “ino” ending. Their symbols are equipped with a “tilde” character like in $\tilde{e}^+$.

The first class of particles are the charged *sleptons* $\tilde{l}$: *selectrons* $\tilde{e}$, *smuons* $\tilde{\mu}$ and *staus* $\tilde{\tau}$. These are the scalar SUSY partners of the corresponding charged leptons in the SM. At the end of this chapter, we will consider a hypothetical LHC process where we produce a pair of selectrons: $\tilde{e}^+\tilde{e}^-$.

The second class of particles are the so-called *neutralinos*: a special class of sparticles predicted to exist due to the symmetry breaking associated with the massive electroweak SM gauge bosons. The SUSY partners of the electroweak SM gauge bosons (W_1, W_2, W_3 and B) are the fermion states $\tilde{W}_1$, $\tilde{W}_2$, $\tilde{W}_3$ (winos) and $\tilde{B}$ (bino), known as the gauginos.

In addition, there are in the MSSM a total of eight scalar degrees of freedom in the Higgs sector, which gives rise to four fermionic SUSY partners known as *higgisinos*. The four gauginos and the four higgsinos mix to form eight different mass eigenstates, namely the neutralinos $\tilde{\chi}_i^0$ $(i = 1, 2, 3, 4)$ and the charginos $\tilde{\chi}_i^{\pm}$ $(i = 1, 2)$, indexed with ascending mass. Among all the four neutralinos, $\tilde{\chi}_1^0$ is the lightest one, and it will be a sparticle of particular interest for this project.

R-parity

In Standard Model processes, both the baryon number (B) and lepton number (L) are conserved. This is due to the fact that there are no renormalizable terms in the SM Lagrangian that violates these conservation laws. In the MSSM, B and L are not naturally conserved due to some terms in $\mathcal{L}_{SUSY}$ that violates these conservation laws. However, this is solved by introducing a new fundamental symmetry of the MSSM which automatically throws away SUSY-terms violating B and L conservation. This symmetry is called *R-parity* (or matter-parity), and it is defined for a given particle with spin s as

$$P_R = (-1)^{3(B-L)+2s}. \tag{1.49}$$

For processes in the MSSM, P_R is a mutliplicatively conserved quantum number from vertex to vertex in the associated Feynman diagrams. The definition of P_R assigns $P_R = +1$ to all the SM particles and the additional Higgs bosons predicted by the MSSM, while the sleptons, neutralinos and all other SUSY partners get $P_R = -1$.

By adding this discrete R-parity symmetry to the MSSM, there are a couple of important phenomenological consequences for the search of new physics as described by the MSSM:

1. Sparticles can only be produced in even numbers (typically two) in collider experiments.
2. There exists a *lightest supersymmetric particle* (LSP) with $P_R = -1$ which is absolutely stable.
3. A sparticle decay will eventually lead to a final state with an odd number of LSPs (typically just one).

The first point above comes from the fact that in any collider experiment we collide SM particles. Since all the SM particles have $P_R = +1$, the R-parity at the final state also has to be $+1$ which can only be obtained by an even number of sparticles in any vertex. The second point is interesting, since it suggests the LSP as a possible candidate for dark matter, given that it is neutral and very weakly interacting. The third point is analogous to the first point, but starting with $P_R = -1$.

Example: Hadronic Slepton Production

In this project we will consider hadronic slepton production of the form

$$pp \to \tilde{l}^+\tilde{l}^- \to l^+l^-\tilde{\chi}_1^0\tilde{\chi}_1^0, \tag{1.50}$$

at LO and at LO+NLO. Each produced slepton decays into a lepton and the lightest neutralino, which in this case is the stable LSP. As a Feynman diagram, the process can be visualized as shown in fig. 1.2. We limit our study to the production of first-generation sleptons, *i.e.*, selectrons, through s-channel electroweak exchange (γ or Z). We can write it as

As an explicit example of eq. (1.50), we will very briefly discuss the Feynman diagrams that contribute to production of the first generation of sleptons: selectrons. We can write it as

$$pp \to \tilde{e}^+\tilde{e}^-, \tag{1.51}$$

where we have omitted the final state leptons and neutralinos as they are not relevant for this discussion. In fig. 1.3 we can see the tree-level contribution to eq. (1.51) from an electroweak exchange. By summing these two diagrams, squaring the amplitude, summing over the three color states of the initial quarks and average over spins, you get the LO cross section for eq. (1.51). The result can be found in [8, eq. 50.68, Cross-section formula for specific processes].

Similarly, two next-to-leading (NLO) order contributions are shown in fig. 1.4. It is not obvious at first why a diagram with a final state gluon has to be considered for eq. (1.51) without any gluons, but it will become clear shortly.

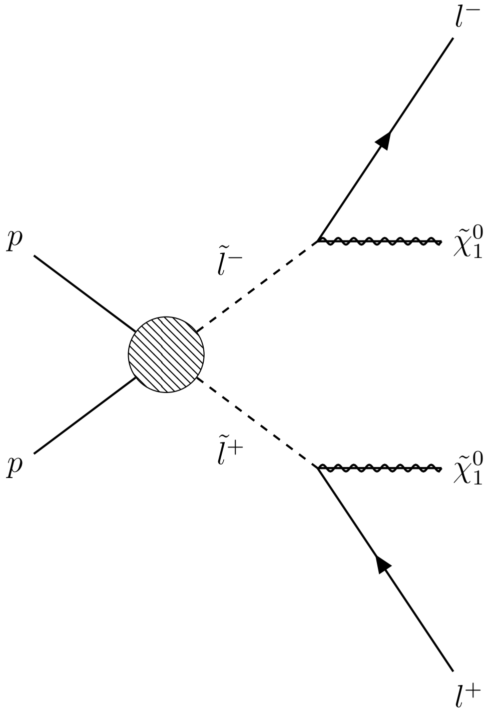


Figure 1.2: The general process we are considering in this thesis: hadronic production of sleptons $\tilde{l}^\pm$ that decay to charged leptons $l^\pm$ and lightest neutralinos $\tilde{\chi}_1^0$ (LSPs). Note that even though the sleptons show up as propagators here, they are treated as on-shell real particles.

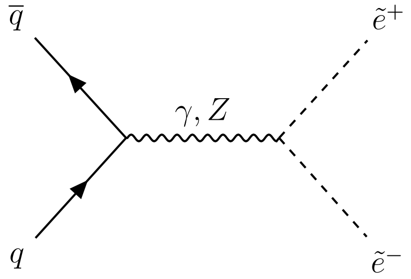


Figure 1.3: Electroweak tree-level contribution to **??** with annihilation of quarks to selectrons through a γ or Z.

It is quite common in QFT that single perturbative contributions to the invariant matrix element $\mathcal{M}$ diverges. This may sound like a problem at first, because predictions of any physical viable theory have to remain finite. However, we should remind ourselves what physical theories actually predict, namely observable quantities. Are single Feynman

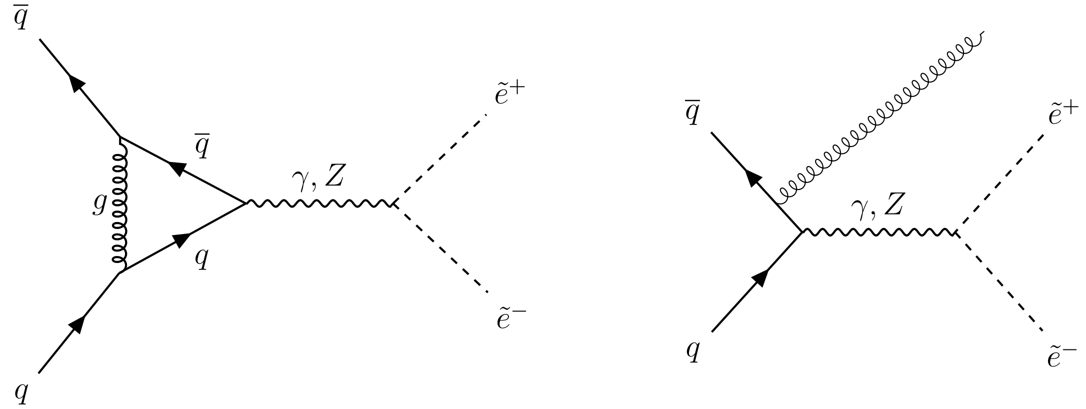


Figure 1.4: Examples of next-to-leading order diagrams contributing to eq. (1.51). The diagram to the left is an example of a one-loop contribution. In the diagram to the right, there is a gluon radiated off from the initial state quark. These diagrams diverges in the limit where the gluon is soft (momentum → 0), known as an infrared (IR) divergence.

diagrams observable? No, it is the cross section σ which we can measure in reality. The cross section is, using the perturbative approach, computed as an infinite sum of Feynman diagrams – implying that it is the sum of diagrams that is observable.

Looking at the loop diagram in fig. 1.4, it is not obvious what is problematic with that diagram by itself. We can first notice that there is an undetermined "momentum" in the loop, which can be picked arbitrary. However, the Feynman rule of a loop instructs us to integrate over this undetermined momentum, and by writing down this expression, it is not hard to see that this integral diverges. While there are two divergences associated with this diagram, we will only consider one of them which is closely related to the divergence in the right diagram with a radiated gluon.

The reason why the diagram with the radiated gluon has to be included has to do with a rather subtle detail. Considering this diagram alone, it describes a seemingly different process (2 → 3) with the radiation of a gluon from the initial state. If you write down the amplitude of this diagram and compute the corresponding cross section, you will see that the cross section actually diverges in the limit where the gluon momentum $k \to 0$. See [2, ch. 6] for a analogous detailed discussion. This is called *soft radiation*, and it introduces what we call *infrared divergences* (IR divergences) in QFT. To avoid letting this diagram diverge, we can parameterize it by giving the gluon a fictive mass μ which is put to 0 at the end.

Rather surprisingly, the IR divergence from the loop diagram and the IR divergence from the radiative diagram are identical, but with a relative sign. Thus, if we sum the cross section contribution from each diagram, these divergences cancel exactly leaving us with a finite cross section which we can compare with experiments. It makes sense to add this diagram to our process eq. (1.51) because in the limit of a vanishing gluon momentum, the radiative diagram becomes indistinguishable from our original $2 \to 2$ process.

It will always be the case that these infinities "cancel" each other if we are dealing with a physically good theory, formally known as a *renormalizable theory.* The reason is that when your theory is renormalizable, you only need a finite number of "counter" diagrams to cancel all the emerging infinities.

Chapter 2

Information Content in Probability Density Functions

The applications of probability theory and statistics are virtually endless. From forecasting the weather, teaching computers to tell hot dogs from hamburgers and evaluating the significance of certain signals above a background noise, it is hard to overstate its relevance.

In this chapter we will present some core ideas from probability and information theory, and how that can be used to restate the task of this master project – quantifying the overall difference between two complicated mathematical objects using the concept of information. This chapter is based on [9, ch. 1] regarding probability theory, [10, ch. 4] for information theory and entropy and [1] for density ratio approximations.

2.1 Probability Density Functions

Prior to defining information, it can be helpful to remind ourselves some basics from probability theory. A *probability density function* (pdf), *probability distribution* or *density*, is any integrable function $f : \Omega \to [0, \infty)$ of a continuous random variable X satisfying the normalization condition

$$\int_{\Omega} f(x)\mathrm{d}x = 1, \tag{2.1}$$

where Ω is known as the *sample space* – the set of possible outcomes for X. Currently, Ω can be any measurable continuous space depending on the application, but we are going to restrict ourselves to the set of real n-tuples of $\mathbb{R}^n$, which will become clear shortly.

2.1.1 Single-variable Densities

The idea behind eq. (2.1) and the *probability* for different values of X can be understood by considering a density f defined on single reals from $\mathbb{R}$. The probability that $X \in I = [a, b]$ is computed as an integral of f over the interval I, and it is written as

$$P(a \leq X \leq b) = \int_a^b f(x)\mathrm{d}x. \tag{2.2}$$

The probability for finding $X \in [x, x + dx]$ is simply $f(x)\mathrm{d}x$, which is the probability density at x times a tiny one-dimensional volume $\mathrm{d}x$. Thus, $f(x)\mathrm{d}x$ can be thought of as the "probability mass" at the point x. Clearly, if we sum up all these small probabilities across the whole line of real numbers we get the total summed probability to find X somewhere on that line, which has to be 1 consistent with eq. (2.1).

Another commonly used function derived from eq. (2.2) is the *cumulative probability function*

$$F(x) \equiv P(-\infty < X \leq x) = \int_{-\infty}^{x} f(t)\mathrm{d}t$$

which simply is the probability to find $X \leq x$ for some $x \in \mathbb{R}$. A property of F is that it will approach 1 from below as $x \to \infty$, and F is a non-decreasing function of x since f is non negative. Moreover, it is easy to see that F is bounded between 0 and 1.

An alternative, but equivalent definition commonly seen in the literature [9, p. 9], is to simply define the probability density as the derivative of F with respect to x, *i.e.*,

$$f(x) = \left.\frac{\mathrm{d}F}{\mathrm{d}x}\right|_{X=x}.$$

Note that F is now the fundamental object in the theory over continuous random variables defined with the properties above with the additional requirement that F has to be differentiable almost everywhere on its domain $\mathbb{R}$.

2.1.2 Multi-variable Densities

With single-variable densities at hand, it is easy to generalize to multiple variables. We say that for continuous random variables $X_1, X_2, \ldots, X_n = \mathbf{X} \in \mathbb{R}^n$ we can define a *joint probability density function* or *multi-variable density* $f_{\mathbf{X}} : \mathbb{R}^n \to [0, \infty)$ which computes the probability for $\mathbf{X}$ to be inside a domain $A \subseteq \mathbb{R}^n$ as

$$P(\mathbf{X} \in A) = \int_A f_{\mathbf{X}} \mathrm{d}^n \mathbf{x} \equiv \int \cdots \int_A f_{\mathbf{X}}(x_1, x_2, \ldots, x_n) \mathrm{d}x_1 \mathrm{d}x_2 \cdots \mathrm{d}x_n. \tag{2.3}$$

As in the single variable case, the intuition behind this is that the probability for $\mathbf{X} \in A$ is simply a sum of tiny "probability masses" $f_{\mathbf{X}} \mathrm{d}x_1 \mathrm{d}x_2 \cdots \mathrm{d}x_n$ over the domain to obtain its total mass, and this sum will be again 1 if the integration is performed over all of space $A = \mathbb{R}^n$, as eq. (2.1) requires.

Together with eq. (2.3) it is common to define what is known as a *marginal density function* obtained by integrating over all but a subset $\chi = \{X_i, \ldots, X_j\} \subset \{X_1, \ldots, X_n\}$ of the variables, *i.e.*,

$$f_\chi(x_i, \ldots, x_j) = \left(\prod_{X_k \notin \chi} \int \mathrm{d}x_k \right) f_{\mathbf{X}}(x_1, x_2, \ldots, x_n).$$

The marginal density can be thought of as the probability distribution of $\{X_i, \ldots, X_j\}$ irrespective of what values the other variables acquire. Formally, we say that f_χ is *marginalized.*

The normalized differential cross section discussed in chapter 1 is an example of a multi-variable density. Using the QED annihilation example from eq. (1.19) in chapter 1, the normalized differential cross section in eq. (1.25) represents the density over the two physical degrees of freedom, while eq. (1.26) is the marginalized density with ϕ integrated out.

2.1.3 Expectation and Variance

The expectation value $\mathrm{E}\left[g(X)\right]$ of a function g of a random variable X distributed according to the density $f(x)$ is defined as

$$\mathrm{E}\left[g(X)\right] = \int_{-\infty}^{\infty} g(x)f(x)\mathrm{d}x. \tag{2.4}$$

It is sometimes called the *mean* of $g(X)$, and it can be interpreted as the average value of $g(X)$ in a population of x-values distributed as $f(x)$.

The variance $\mathrm{Var}\left[g(X)\right]$ can be defined using the expectation value as

$$\mathrm{Var}\left[g(X)\right] = \mathrm{E}\left[(g(X) - E[g(X)])^2\right] = \int_{-\infty}^{\infty} (g(x) - E[g(X)])^2 f(x)\mathrm{d}x, \tag{2.5}$$

which is a measure of the spread of $g(X)$ around its mean. By using the linearity of $\mathrm{E}\left[g(X)\right]$ we can rewrite eq. (2.5) as

$$\mathrm{E}\left[(g(X){-}E[g(X)])^2\right] = \mathrm{E}\left[g(X)^2{-}2X\mathrm{E}\left[g(X)\right]{+}\mathrm{E}\left[g(X)\right]^2\right] = \mathrm{E}\left[g(X)^2\right]{-}\mathrm{E}\left[g(X)\right]^2, \tag{2.6}$$

which sometimes is a useful identity.

2.2 Quantifying Information and Information Difference

In 1948, Claude Shannon published a paper to discuss and formalize the concept of information by addressing the following problem: given all possible "messages" a source can transmit over a noisy channel to a given receiver, how can the message be reconstructed in a way to minimize the *loss of information*? Shannon's main result [11, thm. 10, p. 408], the *noisy channel coding theorem*, states that for a given channel capacity C transmitting information at a rate $R < C$, there exists a way to transmit the message with arbitrarily low probability of error. A fundamental mathematical quantity behind this result is the so called *Shannon entropy*, and it is a key concept in the definition of information.

In this section we will briefly formalize the concept of information, in particular defining the amount of information within a probability distribution. We will see how the concept of entropy naturally enters the description of information, and how that is used to describe the overall behavior of random variables.

2.2.1 Entropy in Information Theory

Imagine you have an unfair coin[1] favoring largely heads, and you toss it a large number of times. Intuitively, you expect to see mostly heads in the long run and it would not be a big surprise to get another head if you were to toss it again. In one sense, the amount of information contained in the series of random outcomes is *low* due to the high level of predictability. Conversely, there is more surprise associated with a fair coin as it is less predictable, *i.e.*, there is more to learn about its behavior.

Shannon Entropy

A common way to measure the *amount of information* in a random variable X with N possible outcomes is by using the Shannon entropy

$$H(X) = -\sum_{i=1}^{N} p_i \log_a p_i, \tag{2.7}$$

where p_i is the probability for outcome i. Here, $0 \cdot \log_a 0$ is defined as 0[2]. The unit of information depends on the choice of base a in the logarithm, but a common choice is simply base 2 as it has an intuitive interpretation in computer science: the *bit* of information. Another choice seen in the literature, but less popular, is the natural logarithm (base e) with the *nat* as the unit of information. We will stick to using logarithm base 2 making a bit the fundamental unit of information.

In simple terms, the Shannon entropy measures the "level of surprise" associated with a series of outcomes of X. To recast eq. (2.7) into the language of probability theory, the entropy is simply the expectation value of the so called *self information* $\log_a [1/p(X)]$ given the probability distribution p for X. In other words, the Shannon entropy tells us

[1] A tossed coin that lands on one side more often than the other.

[2] From the analytical fact that $x \log_a x \to 0$ as $x \to 0^+$

on average how many outcomes measured in number of bits that can be identified by identifying one of the outcomes. Alternatively, the Shannon entropy gives us the average number of bits needed to encode any message that is transmitted by X.

Example: The Lottery

To shine light on the rather cryptic conclusion from the previous paragraph, let us consider a classic lottery example with a small twist. Imagine a lottery with 100 playing numbers with one of them being the winning number. Little do we know that it is our good friend Alice that runs this lottery, and she decides to first pick one number at random and reveal if it is the winning number or not. If we assign the probability to reveal the winning number as $w = 1/100$ and consequently the probability to reveal non-winning numbers $l = 99/100$, we can understand the Shannon entropy of Alice's message (winning number or not) in the following way: if Alice does not reveal the winning number (probability 99/100), barley any information, $\log_2(100/99) \approx 0.014$ bits, is conveyed by Alice since it only allows us to identify the revealed number as one of the non-winning numbers. If Alice reveals the winning number (probability 1/100), she conveys a staggering $\log_2 100 \approx 6.6$ bits of information to us, since we based on that single message now can identify all the 99 non-winning numbers and the one winning number. While the information from a single reveal measures how many playing numbers we can throw away in the search for the winning number, the average value of these two scenarios is exactly the Shannon entropy, interpreted as the effective size of Alice's message in number of bits, *i.e.*,

$$H = \frac{99}{100}(0.014 \text{ bits}) + \frac{1}{100}(6.6 \text{ bits}) \approx 0.081 \text{ bits}.$$

Despite being a good friend, Alice will on average not be very helpful revealing the winning number if we were to repeat this game a large number of times.

Example: The Game of Sixty-Four

What does it mean that the Shannon entropy gives us the number of bits needed to encode a message? To understand this better, you are asked to play the following game with Alice: she thinks of a number from 1 to 64, let us say 17, and your task is to deduce the number by asking only yes-no questions. What is the smallest number of yes-no questions needed to identify the number? Intuitively, a good strategy is to divide the set of 64 possibilities into equal sized sets in the following manner:

1. Is the number higher than 32? No.
2. Is the number higher than 16? Yes.
3. Is the number higher than 24? No.
4. Is the number higher than 20? No.
5. Is the number higher than 18? No.
6. Is the number higher than 17? No.

After 6 questions, Alice's number is deduced to be 17. How much information is gained after every question? Assuming that each set of halves are equally likely to contain Alice's number, the information gain is $\log_2(1/0.5) = 1$ bit; in total 6 bits after 6 questions. This comes from the fact that half of the remaining numbers are identified after every question (being Alice's number or not) which is the fastest scheme to exclude possible numbers. In other words, 6 bits of information and at least 6 questions are needed to identify Alice's number which can be thought of as her message encoded as a string of yes-no answers. Interestingly, and not hard to prove, $\log_2 64 = \log_2 2^6 = 6$ bits of information is required to identify Alice's number independent of your strategy.

Largest Possible Entropy

For a given number of possible outcomes N, what configuration of outcome probabilities p_i gives the largest Shannon entropy $H(X)$? This is an easy optimization problem using Lagrange multipliers under the constraint that the variables p_i sum to 1, with the solution that all $p_1, p_2, \ldots, p_N$ are equal, *i.e.*, $p_i = 1/N$. The maximal Shannon entropy possible for X is therefore

$$H_{max}(X) = \log_2 N,$$

measured in number of bits. One immediate corollary from this observation is that entropy is a decreasing function of the probability distribution in the sense that moving away from a uniform distribution reduces $H(X)$. This is expected since the outcomes of X become more predictable and consequently less surprising, *i.e.*, there is less to learn about the behavior of X.

Continuing the example from the beginning, the maximal Shannon entropy associated with a series of coin flips is $\log_2 2 = 1$ bit. There is on average 1 bit of information needed to capture the behavior of a fair coin since it has two equally likely outcomes. Conversely, there is no information (0 bits) required to encode an unfair coin having "heads" on both sides as it will always result in heads.

2.2.2 Entropy in Statistical Mechanics

While the concept of information is relatively new, the related concept of entropy has older roots. After the birth of thermodynamics in the late 18th century industrialized Victorian England, the idea of thermodynamic entropy was developed to understand and improve heat machines. In simple terms, a heat machine works by extracting useful energy between two reservoirs that differ in macroscopic quantities such as pressure and temperature to do work. Entropy is then effectively a measure of how close the heat machine has reached thermal equilibrium with the reservoirs – the inevitable final state of maximal entropy with uniform temperature and pressure where all useful energy is exhausted. This is the point when no more work can be done by the machine.

In statistical mechanics, the most general definition for thermodynamic entropy in a system is the so called *Gibbs entropy*

$$S = -k_B \sum_{i=1}^{W} p_i \ln p_i, \tag{2.8}$$

where k_B is the Boltzmann constant and p_i is the probability to find the system in a micro state i out of W possible states. This definition of entropy is simply the Boltzmann constant times the Shannon entropy base e, giving the Gibbs entropy identical properties.

Analogous to maximizing the Shannon entropy eq. (2.7), the basic and founding assumption of statistical mechanics is that any of the possible micro states i are equally realizable. Thus, $p_i = 1/W$ which immediately yields

$$S = k_B \ln W$$

using eq. (2.8). This result is known as Boltzmann's entropy equation, and it gives in fact a nice interpretation of entropy in context of information. The thermodynamic entropy S/k_B is a measure of how much information on average is required to determine the exact micro state of a system characterized by a particular macro state.

2.2.3 Cross Entropy and Information Divergence

We have just shown that uniform (or flat) distributions have maximal entropy. Suppose now that we have a distribution q for a random variable X with N possible outcomes that is an approximation of a true distribution p, with the property that $p = 0$ whenever $q = 0$. In this context, q is often called the *reference distribution* – the one you compare with. We can then define what is known as the *cross entropy* or *Shannon-Jaynes entropy* as

$$K(p \,||\, q) = \sum_{i=1}^{N} p_i \log_2 \left[\frac{p_i}{q_i}\right], \tag{2.9}$$

where p_i and q_i are probabilities for outcome i. The cross entropy is simply the expectation value of $\log_2 [p/q] = \log_2 p - \log_2 q$ assuming p, and it can be thought of as how much information is lost by using q to approximate the true distribution p. Equivalently, it measures the *information gain* using p as opposed to using q.

Kullback-Leibler Divergence

By now we have seen how entropy is used to quantify the amount of information within discrete probability distributions. In analogy with eq. (2.9), we can also define the cross entropy known as the *Kullback-Leibler divergence* (KL divergence) for n-dimensional continuous probability distributions $q(\mathbf{x})$ and $p(\mathbf{x})$ as the functional

$$D_{KL}(p \,||\, q) = \int_{\Omega} p(\mathbf{x}) \log_2 \left[\frac{p(\mathbf{x})}{q(\mathbf{x})}\right] \mathrm{d}^n\mathbf{x}, \tag{2.10}$$

where Ω is the common domain of q and p. We require again that $p = 0$ whenever $q = 0$ (formally, the support of q is at least as big as the support of p). The KL divergence satisfies the following properties [12, sec. 1]:

1. $D_{KL}(p \,||\, p) = 0$.
2. $D_{KL}(p \,||\, q) = 0 \iff p = q$.
3. $D_{KL}(p \,||\, q) \geq 0 \quad \forall \quad p, q$.

One nice feature from this set of properties is that the KL divergence gives us a way to test if two arbitrary smooth distributions are identical by asserting $D_{KL}(p \,||\, q) = 0$.

While there are several ways to interpret the KL divergence, it is nevertheless a common tool to measure how different two probability distributions are. Following the analogy from eq. (2.9), it measures the gain in information using the true distribution p rather than the approximation q. While it is tempting to classify the KL divergence as a metric on the space of probability densities, it will fail as a metric since it is not symmetric in its arguments.

Example: Comparing Normal Distributions

To demystify the KL divergence and the cross entropy between two distributions, let us consider a couple of simple examples using one dimensional normal distributions.

Imagine you have a normal distribution $X_0 \sim \mathcal{N}(\mu_0, \sigma_0^2)$ with the goal to approximate another normal distribution $X \sim \mathcal{N}(\mu, \sigma^2)$. What do you lose in terms of information using the approximation? The density associated with a normal distributed variable $X \sim \mathcal{N}(\mu, \sigma^2)$ is

$$f(x; \mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left[-\frac{(x-\mu)^2}{2\sigma^2}\right],$$

which translates into $q = f(x; \mu_0, \sigma_0^2)$ and $p = f(x; \mu, \sigma^2)$. The KL divergence eq. (2.10) can then be evaluated as

$$D_{KL}(p \,||\, q) = \frac{1}{\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \exp\left[\frac{(x-\mu)^2}{2\sigma^2}\right] \log_2\left[\frac{\sigma_0}{\sigma}\exp\left[\frac{(x-\mu_0)^2}{2\sigma_0^2} - \frac{(x-\mu)^2}{2\sigma^2}\right]\right] \mathrm{d}x$$

which simplifies to

$$D_{KL}(p \,||\, q) = \log_2\left(\frac{\sigma_0}{\sigma}\right) + \frac{1}{\ln(2)\sqrt{2\pi\sigma^2}} \int_{-\infty}^{\infty} \left[\frac{(x-\mu_0)^2}{2\sigma_0^2} - \frac{(x-\mu)^2}{2\sigma^2}\right] \exp\left[\frac{(x-\mu)^2}{2\sigma^2}\right] \mathrm{d}x.$$

This is a matter of evaluating the expectation value of X^2 with respect to p, which is very simple using eq. (2.6) and $\mathrm{E}\,[X] = \mu$ directly giving

$$\mathrm{E}\,[X^2] = \mu^2 + \sigma^2.$$

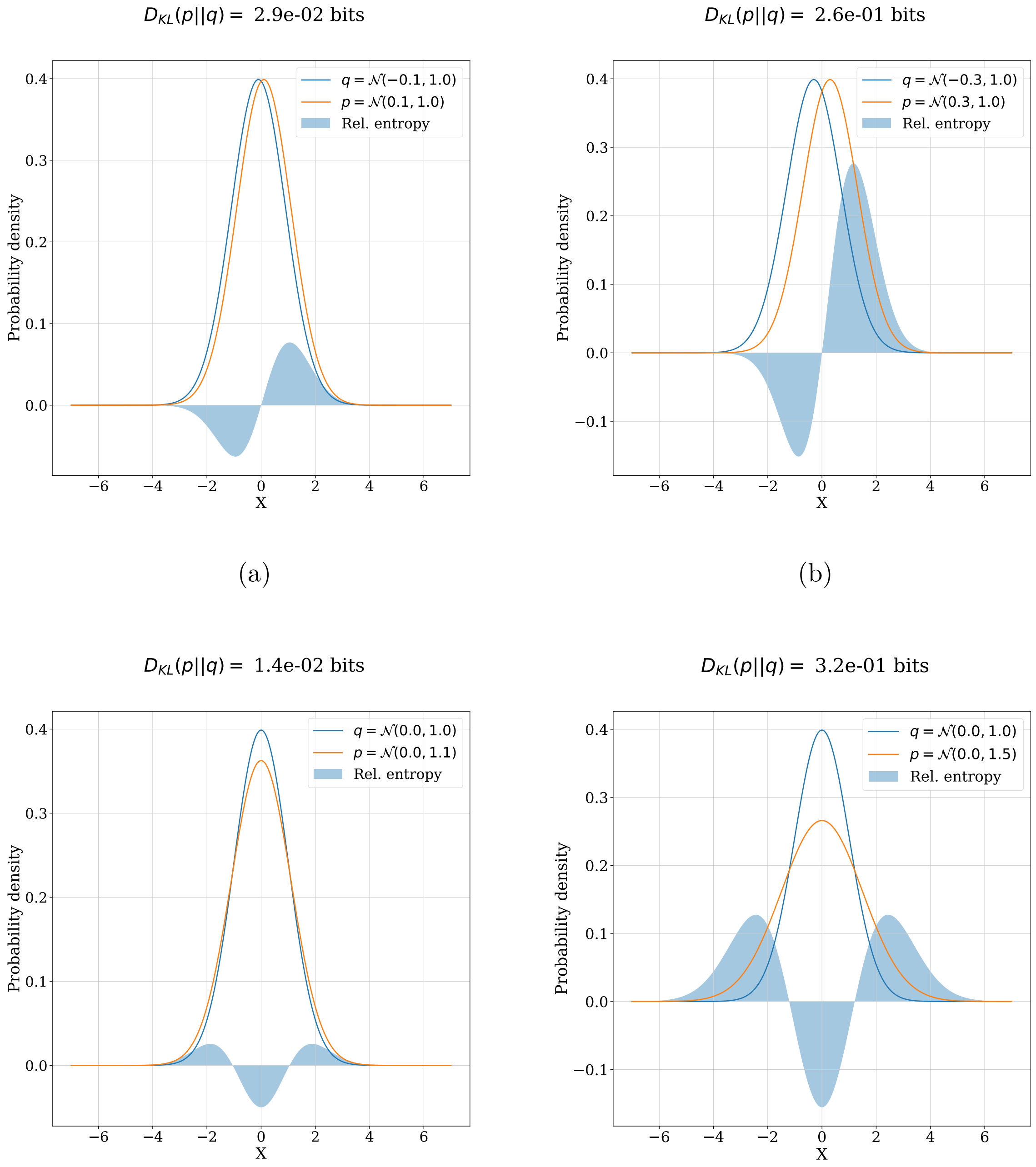
Thus, the final expression for the KL divergence from q to p in number of bits is

$$D_{KL}(p \,||\, q) = \frac{1}{2\ln(2)}\left[\frac{(\mu-\mu_0)^2 + \sigma^2 - \sigma_0^2}{\sigma_0^2} - \ln\left(\frac{\sigma^2}{\sigma_0^2}\right)\right]. \tag{2.11}$$

In fig. 2.1 we have depicted the effect on $D_{KL}(p \,||\, q)$ by shifting and scaling the normal distributions to different amounts. In figs. 2.1a and 2.1b, the two normal distributions are slightly shifted relative to each other, while in figs. 2.1c and 2.1d they are slightly scaled. It is clear that $D_{KL}(p \,||\, q)$ grows as the distributions are shifted or scaled more from each other, *i.e.*, becoming more different.

The blue shaded areas in figs. 2.1a, 2.1b, 2.1c and 2.1d represent the integrand in eq. (2.10), and the final KL divergence in every figure is therefore the net blue area. It is clear from the figures that this net blue area is non negative, consistent with the properties of the KL divergence. The final $D_{KL}(p \,||\, q)$ is computed using eq. (2.11).

For the observant reader, also hinted in the very introduction of this thesis, the ratio between $p(x)$ and $q(x)$ is strictly necessary to compute $D_{KL}(p \,||\, q)$. This can in general be notoriously hard, and quite a bit of effort has been dedicated to develop methods to attack this problem. In this thesis, we will explore one such way to approximate the said ratio using classifiers.



(a)

(b)

(c)

(d)

Figure 2.1: Two normal distributions q and p showing the effect of shifting and scaling on $D_{KL}(p \mid\mid q)$. The distributions on the top row demonstrates shifting, and the distributions on the bottom demonstrates scaling. The KL divergence $D_{KL}(p \mid\mid q)$ is computed analytically using eq. (2.11), and the blue shaded areas represent the integrand in eq. (2.10).

2.3 Approximating Density Ratios Using Classifiers

Approximating a multi-variable probability density by populating a multi-dimensional histogram is not an easy task due to the vast number of samples required. Imagine you had a simple population with equally distributed variables and made a histogram over one of them to approximate its marginal distribution. If 100 samples are needed to roughly resemble the distribution, then it would require roughly $100^2 = 10,000$ samples to populate a 2D histogram over two of the variables or $100^3 = 1,000,000$ samples over three of the variables! The number of samples required for a fixed sample density scales exponentially with the dimensionality of the sample space. This is one aspect of what is often called the *curse of dimensionality.*

In this section we will present a way to circumvent the need to populate multi-dimensional histograms to compute the ratio between two unknown multi-variable densities by recasting the task into a statistical classification problem.

2.3.1 Classifiers

If you as a statistician find yourselves in a garden picking fruits from different fruit trees, and your job is to sort fruits based on *features* such as type and size, then your task is what is known as a *classification problem.* Our brains can easily identify different fruits of different sizes, and therefore easily categorize them into different classes. In the language of statistics, our brain resembles a *classifier* that takes a fruit as an input and classifies the fruit based on its features.

More formally, a classifier on a particular population as defined in context of machine learning is a function

$$s = s(\mathbf{x})$$

that predicts which *class* a given sample with features $\mathbf{x}$ belongs to. The classifier uses a collection of samples obtained by sampling the population – the training data set – to associate different features to different classes, and the output can be a categorical type of data or simply a number. Classes are sometimes called *targets*, *labels* or *categories.*

Machine learning (ML) is typically used to create classifiers by training predictive models on different datasets, but other methods also exist. A particular ML technique common and powerful for classification problems are decision trees, which will be discussed in more detail in chapter 3.

It is increasingly common in science that a simulation samples from some generative model based on a theory θ using a distribution $p_{\mathbf{x}}(\mathbf{x}|\theta)$ which is difficult to evaluate directly, but can be numerically simulated. An example from particle physics is the use of a hypothesis test on a null θ_0 theory against an alternative θ_1 given the observed data points $D = \{\mathbf{x}_1, \ldots, \mathbf{x}_n\}$ by using the likelihood ratio

$$\lambda(D;\theta_0,\theta_1) = \prod_{\mathbf{x}\in D} \frac{p_{\mathbf{x}}(\mathbf{x}|\theta_0)}{p_{\mathbf{x}}(\mathbf{x}|\theta_1)} \tag{2.12}$$

as a test statistic [1, sec. 2.1], where it is typically hard to evaluate the multi-variable densities $p_{\mathbf{X}}$ directly. However, it is possible to use a discriminative classifier to evaluate the ratio between two densities in a relatively easy manner – which in fact is sufficient in many cases, including the example above eq. (2.12).

Given two multi-variable densities $q_{\mathbf{X}}(\mathbf{x})$ and $p_{\mathbf{X}}(\mathbf{x})$ defined on a domain Ω in n variables, it is possible to do a change of variable $S = s(\mathbf{X})$ which will induce two new single-variable densities $q_S(s = s(\mathbf{x}))$ and $p_S(s = s(\mathbf{x}))$ such that

$$r(\mathbf{x}) \equiv \frac{q_{\mathbf{X}}(\mathbf{x})}{p_{\mathbf{X}}(\mathbf{x})} = \frac{q_S(s = s(\mathbf{x}))}{p_S(s = s(\mathbf{x}))} \quad \forall \quad \mathbf{x} \in \Omega. \tag{2.13}$$

The key requirement for eq. (2.13) to be valid is that the function $s(\mathbf{X})$ has to be strictly monotonous in the density ratio $r(\mathbf{X})$, meaning that if you follow a path through Ω such that r increases (or decreases), s has to change monotonically. The proof of this result can be found in [1, thm. 1].

To shed light on eq. (2.13), evaluating density ratios can now be transformed into a classification problem with $s(\mathbf{x})$ being the classifier constructed to differentiate samples $\mathbf{x} \sim q_{\mathbf{X}}$ from $\mathbf{x} \sim p_{\mathbf{X}}$, which computationally speaking is far more feasible. This result also guarantees that any classifier will do the job as long as it is monotonous in the density ratio $r(\mathbf{x})$, allowing us to use for instance supervised learning algorithms from ML to construct $s(\mathbf{x})$ which is convenient.

Example: Computing Information Divergence of Normal Distributions with Classifiers

Equation (2.13) will be of high importance in this master project. To demonstrate its power, let us consider the normal distributions we created in fig. 2.1. Imagine now for the purpose of this example that q and p are unknown probability densities. We could solve this by numerically simulating q and p (generating samples) to approximate them with the resulting histograms, but this is generally not an option for arbitrary probability densities as discussed previously. Instead, we will use a technique from ML known as a *boosted decision tree* (BDT) to train a classifier $s(x)$ to differentiate samples $x \sim q$ from $x \sim p$, then use eq. (2.13) to compute the density ratio p/q to compute their KL divergence with eq. (2.10) using Monte Carlo integration. We are not going to explain the details around the implementation of the decision tree for this example since it is a mere distraction from the main point. The general method in details is laid out in chapter 3.

Explicitly, we compute the KL divergence of the resulting class histograms using

$$D_{KL}(p_s \mid\mid q_s) = \sum_i p_S(s_i) \log_2 \left[\frac{p_S(s_i)}{q_S(s_i)} \right] \Delta x \tag{2.14}$$

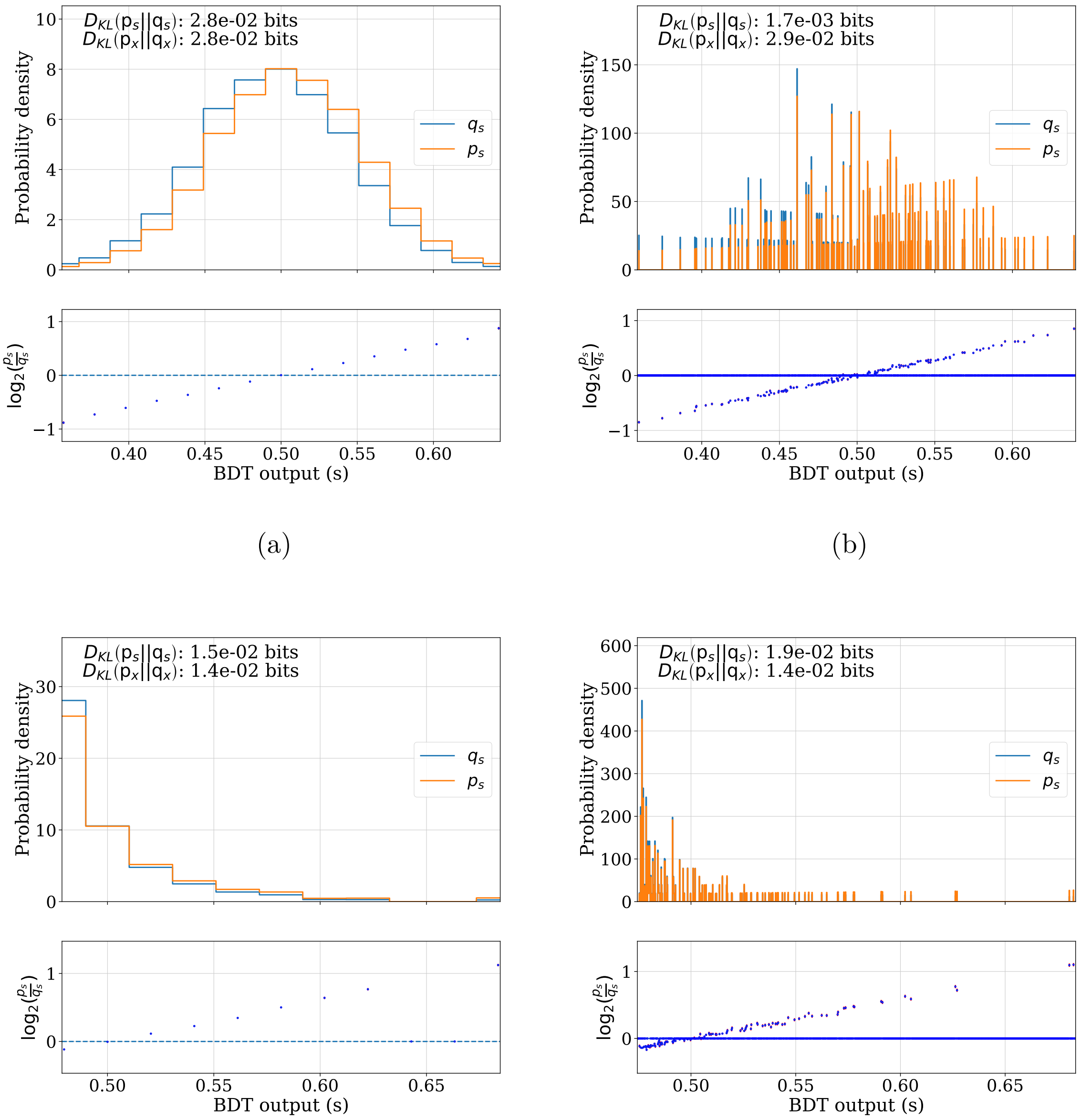
where Δx is the fixed bin width. The other, labeled as $D_{KL}(p_X \mid\mid q_X)$, is computed by

$$D_{KL}(p_X \mid\mid q_X) = \int p(x) \log_2 \left[\frac{p(x)}{q(x)} \right] \mathrm{d}x \tag{2.15}$$

using Monte Carlo (MC) integration by sampling numbers x from p_X, which computes the integral as the sample mean of $\log_2 [p_X(x)/q_X(x)]$. In fact, it is this integral we want to compute using the suggested classification method on our kinematic distributions from chapter 1. Now that we have trained a classifier to distinguish the samples from each other, we can use eq. (2.13) allowing us to compute the ratio inside the logarithm as the class ratio $p_s(s(x))/q_s(s(x))$. This is just the ratio of two one-dimensional distributions which can easily be approximated by populating their histograms, resolving the problem. These histograms are shown in fig. 2.2.

Figures 2.2a and 2.2b plot the class distributions of the output $s(x)$ from the boosted decision tree trained on q_X and p_X from fig. 2.1a. Similarly, figs. 2.2c and 2.2d show the class distributions where the BDT is trained on q_X and p_X from fig. 2.1c. As we can see from the figures, $D_{KL}(p \mid\mid q)$ and $D_{KL}(p_X \mid\mid q_X)$ match as expected – verifying eq. (2.13) in this simple scenario.

The choice of number of bins has a direct and obvious effect on the KL divergence estimate of the class distributions q_s and p_s since the bin width Δx enters explicitly, but only an indirect effect on $D_{KL}(p_X \mid\mid q_X)$. As shown in figs. 2.2b and 2.2d with 5000 bins, compared to the decently binned histograms in figs. 2.2a and 2.2c with 50 bins, $D_{KL}(p_s \mid\mid q_s)$ changes considerably when increasing the number of bins, but $D_{KL}(p_X \mid\mid q_X)$ on the other hand is affected only slightly – approaching the true KL divergences seen in fig. 2.1. This is due to the way $s(x)$ behaves as the output of a decision tree with a finite number of class leaves (see chapter 3), giving $s(x)$ a discrete set of values it can take. The effect of this is that histograms with different number of bins will in general "pick" up a different subset of s-values which alters the density approximations p_s and q_s, and thus the ratio between them p_s/q_s which is the way $D_{KL}(p_X \mid\mid q_X)$ is indirectly affected.



(a)

(b)

(c)

(d)

Figure 2.2: The output of a boosted decision tree as a classifier $s(x)$ to differentiate samples $x \sim q_X$ from $x \sim p_X$, where upper plots in every figure shows the distribution of the two possible classes q_s and p_s which is used to compute the KL divergence through eq. (2.13). The lower plots shows the logarithmic class ratios p_s/q_s which is used directly in the KL divergence estimation. There are 50 bins in figs. 2.2a and 2.2c, and 5000 bins in figs. 2.2b and 2.2d.

Chapter 3

Boosted Decision Trees for Classification

To utilize eq. (2.13) from the previous chapter we need a classifier that is monotonous in the pdf ratio $p(\mathbf{x})/q(\mathbf{x})$. All the heavy lifting in computing the high-dimensional pdf ratio is done with this classifier, making it an essential component for this project. In this chapter we will introduce some basic ideas and terminology from machine learning, and explain the classification technique known as *boosted decision tree*, which we employ in our project.

3.1 Machine Learning

Machine learning (ML) algorithms are a special class of algorithms that are able to improve automatically based on previous experience. The algorithm does this by creating a model based on a dataset of samples, known as *training data*. The model is then used to make predictions or decisions from new unseen samples, after having experienced the training data. In many cases, learning problems can be formulated as a minimization of a *loss function* during training, *i.e.*, an optimization problem.

It has become more common in the last decades to deploy machine learning techniques in science. From forecasting the weather, recognizing cancerous biological tissues or predicting stock prices, ML has a very wide range of applications, and is itself a highly active field of research. There exists a large number of different ML techniques, with *neural networks* and *decision trees* being two examples.

As an example of a very simple ML model we can consider a linear model,

$$y(\mathbf{x}) = w_0 + \sum_{i=1}^{N} w_i x_i, \tag{3.1}$$

where y is the output of the ML algorithm and $\mathbf{x}$ is the input describing a single sample as a tuple of N numbers (x_i). The numbers w_i are known as the *weights*, and they are the parameters that are optimized during training to make the model fit the dataset. This simple model is used in linear regression, where we attempt to produce a best-fit straight line to fit the given training set. The output of a ML model is often referred to as the *target value*, and the components of the input sample $\mathbf{x}$ are referred to as the *features* of the sample.

3.1.1 Decision Trees

In this thesis we will utilize a ML technique based on *decision trees*. A decision tree is maybe one of the simplest approaches to construct a ML model, and it is used in a large variety of problems involving regression, classification and decision making.

Terminology: A decision tree consists of a *root node* at the very top followed by intermediate *interior nodes*. The final bottom nodes of the tree are known as *leaf nodes* or just *leaves*, while the connection between nodes are referred to as *branches*.

In a sense, all humans use decision trees on daily basis. Imagine that are considering whether or not to play beach volleyball today. In the act of deciding, you might ask yourself if it is raining or not. If it is, you decide not to play volleyball. If it does not rain, you might next ask yourself if it is windy. If it is, you again do not want to play. However, if it is not windy, you conclude that today is a fine day for some beach volleball.

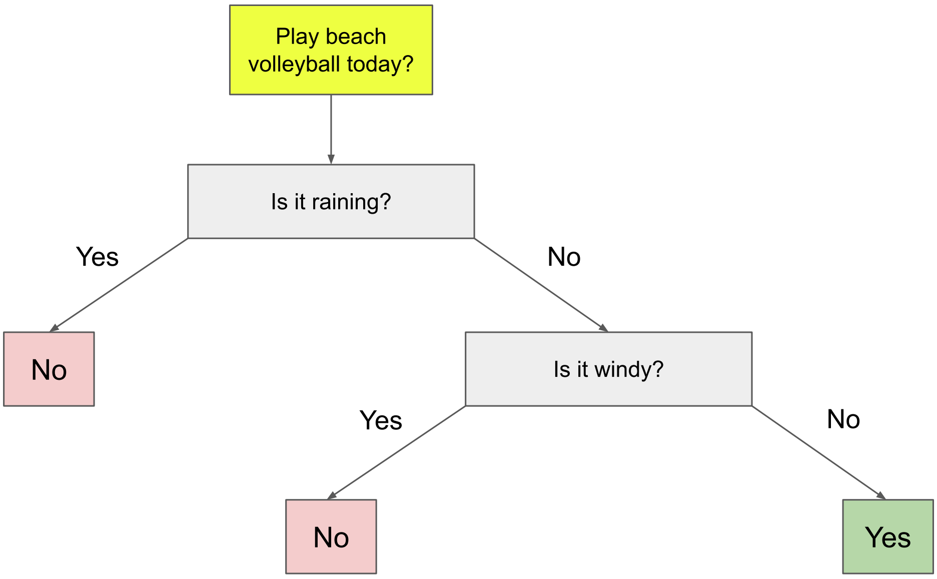


Figure 3.1: A simple decision tree to decide whether today is a good day or not to play beach volleyball. By checking true-false questions, the input falls into one of the leaves which represents the final output of the decision tree (interpreted as a decision).

This "tree" of true/false conditions is exactly what we mean by a decision tree, and it can be visualized graphically as in fig. 3.1.

In general, it works by categorizing the *features* of a dataset based on true/false conditions to finally predict a *target value*. The tree is built by "splitting" the data points along these conditions, and the end is reached at some maximum *depth*. A data point that has traversed down the tree ending in a leaf node will belong to a *class* of data points sharing the same features. This leaf node can represent a categorical data type representing a choice or a decision, or simply a number. A common way to compute the appropriate number for a class of data points is simply the average value of all the targets in that class of data points.

Two examples of typical decision trees:

- *Classification trees*: predict which class a particular input data belongs to. This is used when you want to categorize data (known as classification).
- *Regression trees*: output real numbers which is used to predict numerical targets (number of expected hospitalizations day by day, efficiency of drugs on patients etc.).

One benefit with decision trees is that they are really easy to implement and use, making them a great entry point to ML. Depending on how you split your dataset (what nodes you implement), the performance and accuracy of a decision tree can vary largely. There exist iterative methods that will split your dataset in such a way that minimizes the overall prediction error, for instance by minimizing the *mean square error* similar to finding the best fit for linear regressions. Another benefit is that a decision tree requires very little preparation of the data. As long as you can represent the data on the form

$$(x_1, x_2, \ldots, x_N, Y) = (\mathbf{x}, Y) \tag{3.2}$$

where $x_1, x_2, \ldots, x_N$ are the features collected as a tuple $\mathbf{x}$ and Y is the target value, a decision tree can automatically be created by choosing a particular set of splittings of the features to a desired set of classes.

However, there are a couple of limitations with decision trees to be aware of. First and foremost, trees are highly sensitive to how you construct them. A small change in the training data can have a large impact on the final predictions. Therefore, robust methods are needed to create a set of nodes that minimizes the prediction error. A pitfall associated with decision trees is that it is quite easy to make an *overfit*. This happens when there are too many splittings, making the tree over-complex which do not generalize well to new data.

A decision tree alone can be useful by itself, but sometimes it is necessary to deploy multiple trees at once. In what follows we will briefly discuss one way to do this, and the resulting algorithm will be the method to create a trained classifier for this project.

3.2 Boosted Decision Trees

With decision trees at hand, we can now combine multiple trees together to make a *predictive forest* or an *ensemble*. As an example of how to create such a forest, we will consider the *boosting algorithm*.

A boosting algorithm incrementally builds a forest by learning from previous mistakes. That is, the next tree in the building process is used to compensate for the "flaws" of the previous tree. Starting from a single decision tree constructed from a dataset with n data points $\mathbf{x}_i$ with an output $\hat{y}_i = F_1(\mathbf{x}_i)$, we can begin building the forest. The flaws are measured by using a *loss function*, for instance the mean squared error

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2, \tag{3.3}$$

where y_i is the associated target with a data point with features $\mathbf{x}_i$.

With the goal to minimize the MSE for our dataset, we can introduce a second decision tree with an output $F_2(\mathbf{x})$ such that

$$F_2(\mathbf{x}) = F_1(\mathbf{x}) + h(\mathbf{x}), \tag{3.4}$$

where $h(\mathbf{x})$ is known as the residual of the first tree. If we now attempt to fit the second tree to the data points obtained by computing

$$h(\mathbf{x}_i) = y_i - F_1(\mathbf{x}_i), \tag{3.5}$$

we will reduce the MSE because this is exactly the correction to add to eq. (3.4) to lower the errors from y_i. Continuing this process repeatedly, we can create a forest of N trees, where each successive tree attempts to correct the errors of the previous. Every iteration adding a new tree is called a *boosting round*.

Chapter 4

Generating Datasets and Analysis Implementation

In this chapter we present the data generation and analysis pipeline to produce the results in the upcoming chapter 5. Starting from initializing an event generator of the electroweak MSSM process presented in chapter 1 (eq. (1.51)), we want to end up at an estimate of the Kullback-Leibler divergence (KL divergence) between the kinematic distributions at LO and LO+NLO in perturbation theory. All the scripts we have written for this aim are all publicly available.[1].

In fig. 4.1 we have presented a pipeline that shows how the KL-divergence is computed on a module based level.

4.1 Generating and Combining Events

To construct a dataset of dislepton events for a BDT to train on, we need to generate dislepton production events which we will come back to in a moment. Due to radiative corrections at NLO, generating jets of gluons and quarks is inevitable. Thus, we have to allow for explicit production of hard jets (gluons and quarks) in the final state.

[1] https://github.com/SundeMarius/UIO-MPHYS-project

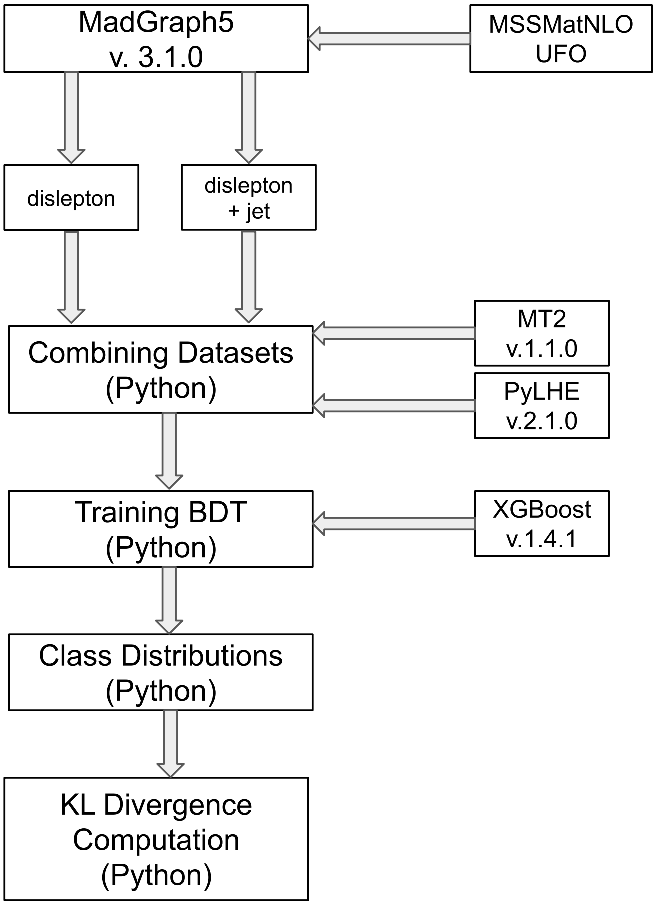


Figure 4.1: This flow chart highlights all the components involved in estimating the KL divergence between kinematic distributions of a simulated MSSM process using the *high level* kinematic variables from [13, table 5.2]. Starting from generating events with MadGraph5_aMC@NLO, it continues to a Python script to combine the events into appropriate datasets. Then we use XGBoost to create a BDT as a trained classifier with supervised learning. All the software versions used in this project are also displayed.

One simple way to accomplish this is the following: generate the process *without* any explicit jets at LO and at NLO to obtain two datasets, then repeat *with* an explicit jet both at LO and at NLO. You now have four processes with four associated total cross sections, but what you want are two processes that include jets in the correct proportion.

The proportion of jet-events is simply

$$p = \frac{\sigma_{\text{jet}}}{\sigma_{\text{jet}} + \sigma_{\text{zero-jet}}}, \tag{4.1}$$

which can be implemented in two ways. The first way is instead of giving each event a count of 1, we count jet events as multiples of p and zero-jet events as multiples of $1 - p$ (known as *weighing* the events). This will make sure that the total count of events are in the proportion given by eq. (4.1). Alternatively, you can turn the combination of events into a probabilistic sampling problem. You have two bunches of events to pick from, and you keep picking events until one of the sets is empty. If you pick a zero-jet event with probability p, the proportion of jet events is expected to approach eq. (4.1) in the limit of many events.

The benefit of the former method of combining events is that no events will be wasted, but it has the downside that it is more difficult to implement because we need to keep track of which dataset the event was picked from. The latter method however is the easiest to implement, but there will be some events wasted depending on which bunch will be emptied first. In this thesis, we have implemented the latter method because it is easy and fail safe. The loss of events turns out to be negligible because the cross sections for $pp \to \tilde{e}^+\tilde{e}^-$ with and without explicit jet production are quite similar.

4.1.1 Event Generator

The software we have used in this project to generate appropriate datasets for dislepton production is MadGraph5_aMC@NLO[14]. It is a program written and interfaced in Python to auto generate computer code (in C++ or Fortran) to primarily calculate tree level, but in some cases also higher order diagrams of user-specified particle physics processes. While MadGraph5_aMC@NLO has many models implemented by default, and it is possible to deploy your own model by specifying its Lagrangian as long as it is renormalizable. Examples of how to use this software can be found in [14, app. A,B,C].

To generate the process in eq. (1.51), we imported the MSSM model from [15] capable of computing LO+NLO kinematics:

```
    import model MSSMatNLO_UFO-full
```

followed by generating the LO processes with and without explicit jets as

```
    generate p p > el+ el- j
    output ``desired_output_folder''
```

```
    generate p p > el+ el-
    output ``desired_output_folder''
```

Generating the corresponding LO+NLO processes was just a slight modification of the commands above:

```
    generate p p > el+ el- j [QCD] / go
    output ``desired_output_folder''
```

```
    generate p p > el+ el- [QCD] / go
    output ``desired_output_folder''
```

where we excluded the gluino written as "go" in MadGraph5_aMC@NLO. While the gluino can show up in loop corrections for the process we are studying, we focus on MSSM scenarios where the gluino is too heavy to have any physical impact, and we therefore left it out of the simulations. Then, we generated $6M$ events from each of the four processes above at two different parameter points in the MSSM.

To decay the final state selectrons, we enabled MadSpin which is a tool included by default in MadGraph5_aMC@NLO event generation. This makes sure that the selectrons remain on-shell, and that they will decay to a specified set of final state particles. Since MadGraph5_aMC@NLO understands the *Les Houches event file format*, the following syntax can be added to the "parameter card" in the four processes:

```
    DECAY 1000011 2.136822e-01 # Width of selectron (sel-)
    1.000000e+00 2 11 1000022 # branching ratio for selectron (sel-) for
   decay to ``2'' and ``11'' pdg codes.
```

This sets the decay width of the selectron to $2.136\,822\times10^{-1}$ GeV, along with setting the branching ratio to 100% for our desired electron + neutralino final state (particle IDs 11 and 1000022 refer to the electron and neutralino, respectively).

4.2 Training Classifiers with Python

To combine and compute the kinematic variables associated with every generated event, we wrote multiple Python scripts to pipeline the process as shown in fig. 4.1. After generating the datasets outputted from MadGraph5_aMC@NLO in the Les Houches file format, we used the package PyLHE[16] to read these files into Python. Then we combined the datasets as described in section 4.1, did event selection based on kinematic cuts and computed the kinematic variables for every event. The events were stored in a *Pandas dataframe* which is a convenient and efficient data structure to store and process large amounts of data in Python.

4.2.1 Kinematic Variables

We implemented two sets of kinematic variables where we label one set as *low level features* (LLF) and the other *high level features* (HLF). The set with LLF is simply all the 4-momentum components of the four final state particles, and HLF consists of eight typical kinematic variables seen in ATLAS papers such as [17] and [18]. The exact variables with their definitions are listed in appendix A.

We implemented selection cuts for kinematic events. First and foremost, only events with a *leading-jet* $p_T < 20$ GeV were included. Since we are only considering the production of selectrons, we have to consider events which are effectively a $2 \to 2$ for a hypothetical detector. As discussed at the end of chapter 1, there are corrections with radiation of quarks and gluons but they can not be considered separately. That means, only events with a soft jet ($p_T < 20$ GeV) are considered such events. Referring this cut as the *base cut*, this is the only cut used to create the event dataset with LLF. For the event dataset with HLF, we implemented additional cuts following [13, table 3.2, p. 29] and in [17, table 2, SR-SF-0J, p. 9]. To see the effect of cuts more clearly, we created an event dataset with HLF using only the base cut.

4.2.2 Training the BDT

To create a trained classifier as a BDT, we used a tailored library for the purpose known as XGBoost [19]. It is a gradient boosting library that has implemented several efficient machine learning algorithms interfaced in many different computer languages such as Python and C++.

We used the training parameters in table 4.1 to train a BDT for each of the six event datasets (two MSSM parameter points with three datasets each as explained above). The first parameter (learning rate) decides how "aggressive" the algorithm is, while the second parameter limits the number of tree layers (to avoid overfit). The "gamma" parameter is a number from $[0, \infty)$ that sets the minimum loss reduction required to continue adding tree layers. These particular parameters were chosen such that the loss function were minimized by doing a simple parameter scan over the "learning rate" and the "maximum depth". To evaluate the performance of the BDT, we used the "area under the curve" known as *auc*. More information about the use of XGBoost in Python with clarifying examples can be found on the documentation pages[2].

Parameter	**Value**
Learning rate	0.1
Max depth of tree	4
Gamma	3
Objective	binary:logistic
Evaluation Metric	auc

Table 4.1: All the different XGBoost parameters used in the training of all the kinematic datasets using LLF, HLF and HLF base cut.

To avoid overfit, we used a feature in XGBoost that stops the training of the BDT if the loss function has not decreased in the last 20 boosting rounds.

[2] https://xgboost.readthedocs.io/en/latest/python/index.html

4.2.3 Computing Kullback-Leibler Divergence

We computed the Kullback-Leilber divergence for both the high-dimensional pdfs and for the one-dimensional class distributions. The former was computed using Monte-Carlo (MC) integration[3] on eq. (2.10) where we substituted the pdf ratio with eq. (2.13). In that case, we also computed the uncertainty using the corresponding MC variance. For the latter, we computed the Kullback-Leibler divergence by numerically integrating eq. (2.10) with q_s and p_s as inputs (approximated using their histograms).

[3] That is, you compute the mean of the dataset of samples.

Chapter 5

Information Loss using Leading Order Kinematics

By now we have performed training with BDTs on two types of collider event datasets; one with events sampled assuming LO kinematics and one with events sampled assuming LO+NLO kinematics, and we are ready to estimate the relevant KL-divergences to quantify the information loss using LO kinematics.

5.1 Results

As explained in chapter 4, the D_{KL} results are split into two categories: one based on low-level features (LLF), and one based on high-level features (HLF). We have also repeated the exercise for two different parameter points in the MSSM. For simplicity, we have picked two different mass points for sleptons and neutralinos with mass splittings 150 GeV and 50 GeV to use as benchmarks. The dataset is split into a standard training/test set in 80/20 proportion, and all the results here reflect the test dataset. Moreover, each BDT is trained with an equal number of LO and LO+NLO events.

We will often refer to the two distributions at play; namely the full LO kinematic distribution labeled as $q_{\mathbf{x}} \equiv q(\mathbf{x})$, and the LO+NLO kinematic distribution labeled as

$p_{\mathbf{x}} \equiv p(\mathbf{x})$. Here $\mathbf{x}$ is a tuple of LLF or HLF as defined in table A.1. The corresponding one-dimensional distributions of the BDT classifier $s = s(\mathbf{x})$ are labeled as $q_s(s)$ and $p_s(s)$, respectively.

As briefly discussed in chapter 2, eq. (2.13) allows us to equate the two ratios

$$r(\mathbf{x}) \equiv \frac{p(\mathbf{x})}{q(\mathbf{x})} = \frac{p_s(s(\mathbf{x}))}{q_s(s(\mathbf{x}))} \tag{5.1}$$

assuming that we have picked or trained an appropriate classifier $s(\mathbf{x})$ which is monotonous in the former ratio $p(\mathbf{x})/q(\mathbf{x})$. That is, if you follow a particular path in the phase space such that $r(\mathbf{x})$ increases or decreases, $s(\mathbf{x})$ changes monotonically. There is however a rather subtle, but important point to make before applying it in this situation: the monotonicity of p_s/q_s is a consequence of eq. (2.13) assuming we have picked a valid classifier, but it is not a sufficient condition for the validity of the theorem. From fig. 5.1 through fig. 5.6, the class ratio p_s/q_s is mostly monotonous on the support of q_s which suggests that our BDT is sufficiently well trained, but the minor deviations confirm that it is not perfect. This will effectively reduce the quality of our classifier and manifest itself as an uncertainty in the KL-divergence estimates. A more thorough discussion of this issue is found in section 5.2.

In every plot there are two similar, but subtly different measures of information loss using the KL-divergence from eq. (2.10). One, labeled as $D_{KL}(p_s \,||\, q_s)$, is simply computed by approximating the integral as a sum over bins from the two class distributions q_s and p_s, *i.e.*,

$$D_{KL}(p_s \,||\, q_s) = \sum_i p_s(s_i) \log_2 \left[\frac{p_s(s_i)}{q_s(s_i)}\right] \Delta x$$

where Δx is the fixed bin width. The other, labeled as $D_{KL}(p_{\mathbf{X}} \,||\, q_{\mathbf{X}})$, is computed by

$$D_{KL}(p_{\mathbf{X}} \,||\, q_{\mathbf{X}}) = \int p(\mathbf{x}) \log_2 \left[\frac{p(\mathbf{x})}{q(\mathbf{x})}\right] \mathrm{d}^n\mathbf{x} \tag{5.2}$$

using Monte Carlo (MC) integration by sampling events $\mathbf{X}$ from $p_{\mathbf{X}}$. The ratio inside the logarithm is simply $r(\mathbf{x})$ computed as $p_s(s(\mathbf{x}))/q_s(s(\mathbf{x}))$ via eq. (5.1). While these two methods look similar, the difference is that in the former case $\ln[p_s/q_s]$ is averaged over the p_s distribution, and in the latter case over $p_{\mathbf{x}}$ itself. The latter method is in fact the one that computes the true $D_{KL}(p_{\mathbf{X}} \,||\, q_{\mathbf{X}})$, *i.e.*, the information loss using the LO approximation $q_{\mathbf{X}}$ instead of the more accurate LO+NLO distribution $p_{\mathbf{X}}$.

The lower panel on all figures shows $\log_2(p_s/q_s)$. Viewing D_{KL} as the expectation value of $\log_2(p(\mathbf{x})/q(\mathbf{x}))$ assuming $p(\mathbf{x})$ as its distribution, then D_{KL} can simply be interpreted as the average value of $\log_2(p_s/q_s)$ weighted with $p(\mathbf{x})$.

The uncertainties displayed as red error bars in the lower panel of all the figures are computed by error propagating $\log_2[p_s/q_s]$ treating q_s and p_s as the variables. Assuming Poisson statistics, we take

$$\sigma_i = \sqrt{k_i} \tag{5.3}$$

as an uncertainty estimate for each bin i of the two histograms approximating q_s and p_s[1].

The KL-divergences $D_{KL}(p_{\mathbf{x}} \,||\, q_{\mathbf{x}})$ from fig. 5.1 through fig. 5.6 are collected in table 5.1 for easy reference.

$\mathbf{m}(\tilde{e},\, \tilde{\chi}_1^0)$ **[GeV]**	D_{KL}^{LLF} **[bits]**	D_{KL}^{HLF} **[bits]**	$D_{KL}^{HLFbase}$ **[bits]**
$(200, 50)$	$(3.8\pm0.1)\times10^{-3}$	$(22.1\pm0.5)\times10^{-3}$	$(19.7\pm0.3)\times10^{-3}$
$(200, 150)$	$(3.6\pm0.1)\times10^{-3}$	$(45.1\pm2.3)\times10^{-3}$	$(19.9\pm0.3)\times10^{-3}$

Table 5.1: The KL-divergence $D_{KL}(p_x||q_x)$ from fig. 5.1 through fig. 5.6 as a measure of information loss using the LO kinematic distribution as opposed to the LO+NLO kinematic distribution.

The classification of events using LLF and the corresponding class ratio p_s/q_s is shown in fig. 5.1 with mass splitting 150 GeV, and in fig. 5.2 with mass splitting 50 GeV. From the vanishing error bars in the lower panels of these figures, it is clear that the BDT is able to statistically differentiate events sampled from $q_{\mathbf{x}}$ and $p_{\mathbf{x}}$.

[1] One way to understand eq. (5.3) is that populating a single bin i can be considered a *binomial experiment* with a given success probability p_i for a sample to end up in that bin depending on the sample distribution, making k_i a binomially distributed variable. Since there are many bins and samples to pick between, p_i will be tiny ($\to 0$), but the number of repeated "trials" n will be large ($\to \infty$). If the expected number of samples $\mathrm{E}[k_i] = np_i$ remains finite in this limit, then this is exactly the limit where a binomial distribution approaches the Poisson distribution with parameter $\lambda = np_i$. Since the histogram is populated only once, resulting in k_i^{obs} observed samples in bin i, an unbiased maximum likelihood estimator for λ becomes $\hat{\lambda} = k_i^{obs}$, with a standard deviation $\sigma_i = \sqrt{\mathrm{Var}[\hat{\lambda}]} = \sqrt{k_i^{obs}}$ using that $\mathrm{Var}[X] = \lambda$ for a Poisson distributed variable X.

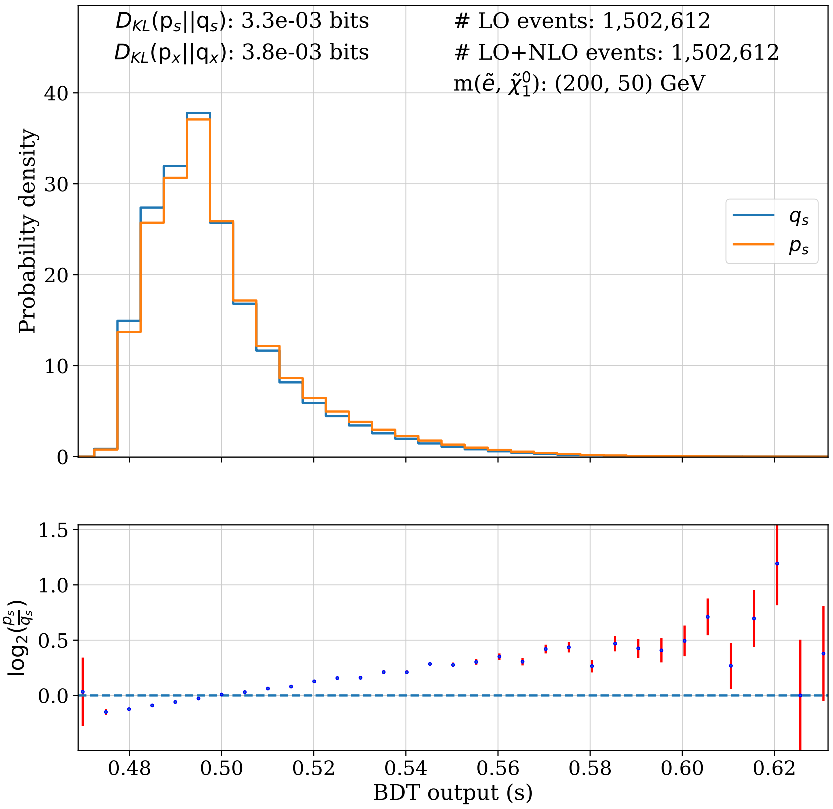


Figure 5.1: The class distributions q_s and p_s of the BDT output $s(\mathbf{x})$ trained on LLF (upper plot) with associated class ratios from each bin plotted with red error bars (lower plot). The mass splitting is 150 GeV.

In similar fashion, the classification of events using HLF and HLF base cut are shown in figs. 5.3 and 5.4 with mass splitting 150 GeV. Due to additional cuts, the majority of events are excluded leaving us only with the “tail” of the kinematic distributions. This explains the overall higher uncertainty in bin counts as shown by the red error bars on the lower plots. Moreover, there are far fewer events to train the BDT with, but in contrast there are now only eight kinematic features used in the classification which obviously requires smaller datasets.

The class distributions using HLF base cut are populated with about as many events as the class distributions using LLF. While HLF and HLF base cut are identical up to choice

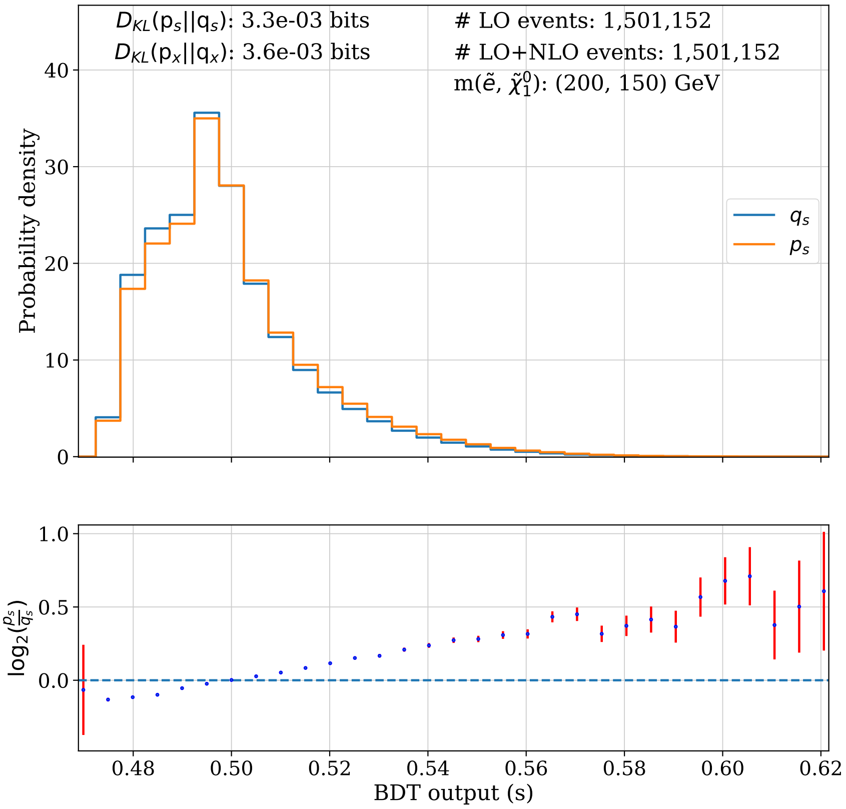


Figure 5.2: The class distributions q_s and p_s of the BDT output $s(\mathbf{x})$ trained on LLF (upper plot) with associated class ratios from each bin plotted with red error bars (lower plot). The mass splitting is 50 GeV.

of cuts, they resemble very different distributions of events. The former case covers more ground of the phase space to include a bigger variety of events, while the latter includes only the “tails” of the kinematic distributions due to the strict cuts.

5.2 Estimating Uncertainties

In this section, we will consider the uncertainties in our classification method to quantify the resulting uncertainties in the KL-divergence estimates.

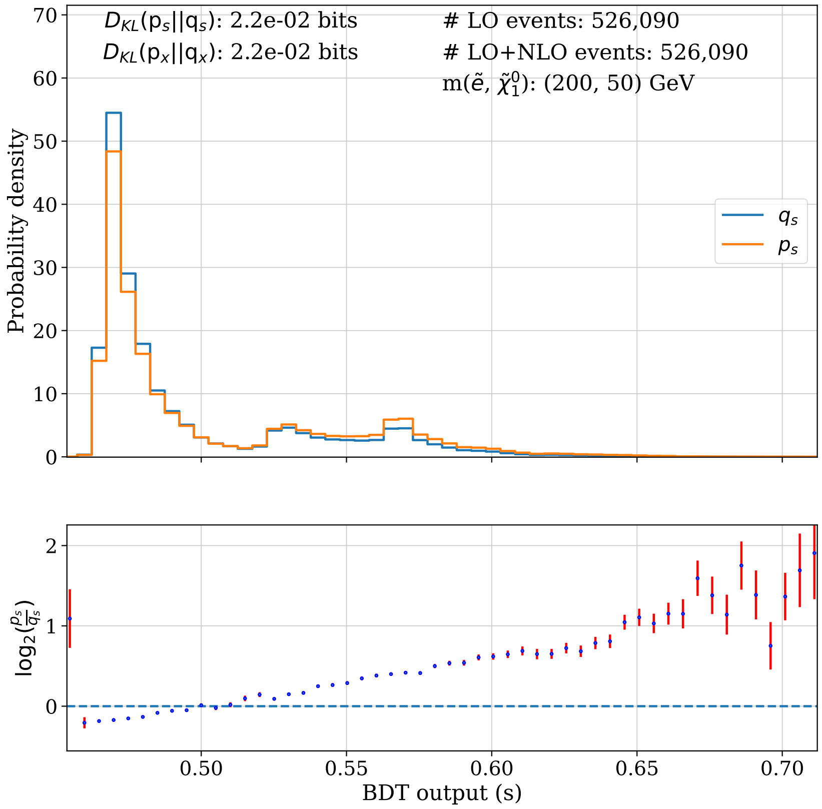


Figure 5.3: The class distributions q_s and p_s of the BDT output $s(\mathbf{x})$ trained on HLF (upper plot) with associated class ratios from each bin plotted with red error bars (lower plot). The mass splitting is 150 GeV.

5.2.1 Uncertainty due to MC Integration

The integral in eq. (5.2) is estimated by Monte Carlo integration. The variance of a mean estimator $\hat{\mu}$ using the sample mean is given by

$$\mathrm{Var}\left[\hat{\mu}\right] = \frac{\mathrm{Var}\left[\theta_i\right]}{N},$$

where θ_i is one of the N independent samples. Using $\theta_i = \log_2\left[p_s(s_i)/q_s(s_i)\right]$ and the sample variance as an estimate of $\mathrm{Var}\left[\theta_i\right]$, the uncertainty due to the finite number of

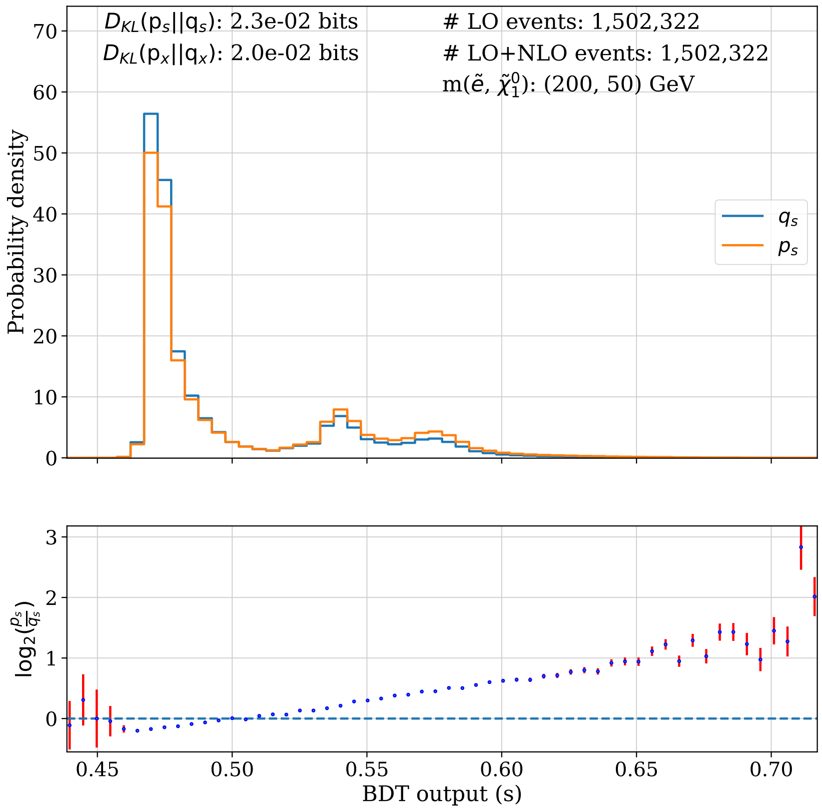


Figure 5.4: The class distributions q_s and p_s of the BDT output $s(\mathbf{x})$ trained on HLF base cut (upper plot) with associated class ratios from each bin plotted with red error bars (lower plot). The mass splitting is 150 GeV.

samples in the Monte Carlo integration is

$$\mathrm{Var}\left[D_{KL}\right] = \frac{1}{N}\mathrm{Var}\left[\log_2\left[\frac{p_s(s_i)}{q_s(s_i)}\right]\right]. \tag{5.4}$$

The square root of $\mathrm{Var}\left[D_{KL}\right]$ gives us the standard deviation of the D_{KL} estimates which we will refer to as Monte Carlo uncertainties. Applying eq. (5.4), the uncertainty in the D_{KL} estimates as in table 5.1 are at least 1% and at most 5%.

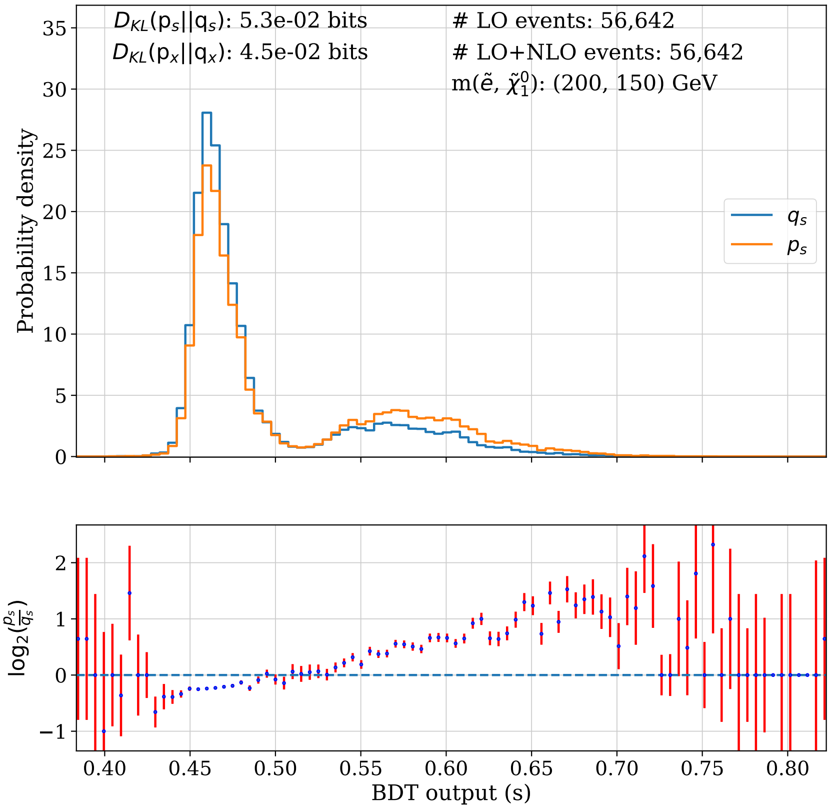


Figure 5.5: The class distributions q_s and p_s of the BDT output $s(\mathbf{x})$ trained on HLF (upper plot) with associated class ratios from each bin plotted with red error bars (lower plot). The mass splitting is 50 GeV.

5.2.2 Uncertainty due to Imperfect Classifier

The uncertainties shown in table 5.1 are purely due to random sampling, introduced above as Monte Carlo uncertainties. However, there is another source of error related to the quality of our classifiers. This error manifests itself as a non monotonic behavior of the class ratios p_s/q_s in fig. 5.1 through fig. 5.6, which ideally would be strictly monotonous for perfectly trained classifiers. This uncertainty is not taken into account in this project, but a couple of methods are presented on how that can be done.

To analyze the quality of our classifiers, there are a couple of suggested methods from

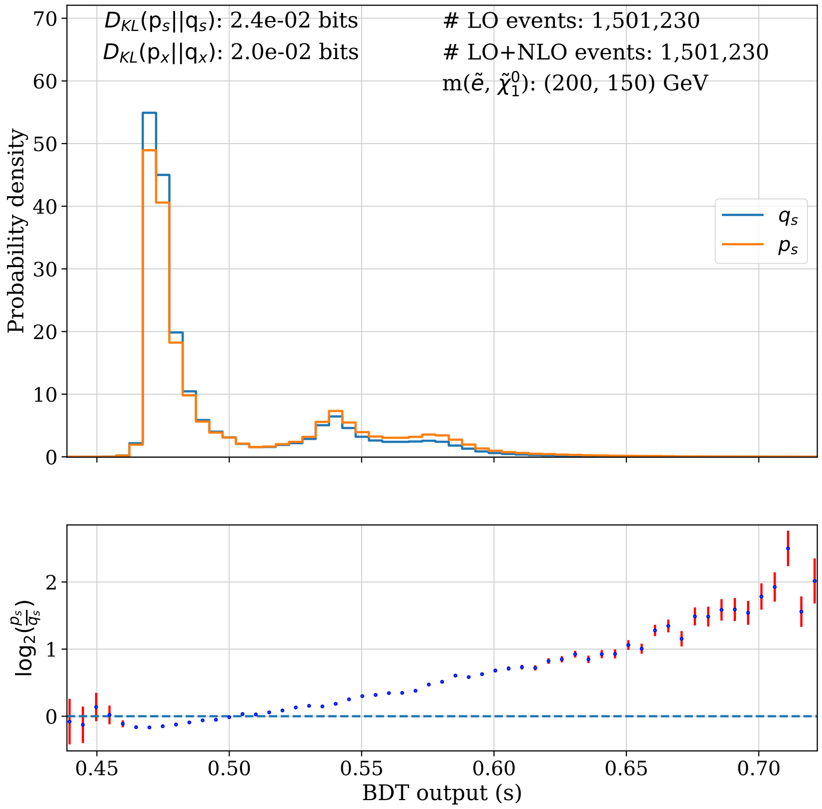


Figure 5.6: The class distributions q_s and p_s of the BDT output $s(\mathbf{x})$ trained on HLF base cut (upper plot) with associated class ratios from each bin plotted with red error bars (lower plot). The mass splitting is 50 GeV.

the literature. Since the true density ratio $r(\mathbf{x}) = p(\mathbf{x})/q(\mathbf{x})$ is not known, one way is to create an estimator of this ratio and compute an estimate from the dataset to compare with the available class ratios. Another way is to use the density ratio p_s/q_s to redefine

$$q_{\mathbf{x}} := \frac{q_s}{p_s} p_{\mathbf{x}},$$

then sample events from this new rescaled distribution. Now, if eq. (5.1) is valid, then it would be impossible for the classifier to distinguish events $\mathbf{x}$ from $q_{\mathbf{x}}$ and $p_{\mathbf{x}}$. This method is therefore a direct way to address the validity of eq. (5.1). For more information using these diagnostics, see [1, sec 3.5, p. 13].

5.3 Interpreting the Results and Outlook

There are some interesting features in all figures from fig. 5.1 through fig. 5.6. Clearly, which table 5.1 also shows, there is a near complete overlap between $q_{\mathbf{X}}$ and $p_{\mathbf{X}}$ yielding a tiny KL-divergence estimate meaning the gain of information is minute. Using the chosen LLF and HLF in direct dislepton production shows that one does learn virtually nothing more about the kinematic distribution going to NLO in perturbation theory, which is reassuring for theoretical physicists that uses analytical LO approximations to analyze and simulate similar particle processes. Another common feature is the growing uncertainty along the tails of the class distributions. However, all characteristic events $\mathbf{x}$ where $s(\mathbf{x})$ is considerably far from 0.5 corresponds to areas in the phase space where one distribution dominates the other. Such events, constituting an exceptionally small minority, will have a negligible effect on the D_{KL} estimate. Another way to view this is that $p_{\mathbf{X}} \to 0$ along the tail giving vanishing contributions to D_{KL}[2]. Moreover, the class ratios p_s/q_s are very close to 1 at $s = 0.5$ which is expected by the use of balanced datasets. In simple terms, $s = 0.5$ represents a set of events $\mathbf{x}_i$ where the classifier fails to distinguish $p(\mathbf{x})$ from $q(\mathbf{x})$ (strictly inconclusive). An explanation for why this is the case can be found in [20, p. 68, sec. 5.2.2].

There is a level of asymmetry in the class distributions about their mean. This is an interesting point as it can be directly linked to the different shapes of $q_{\mathbf{X}}$ and $p_{\mathbf{X}}$. The fact that there are many more events $\mathbf{x}$ classified as $s < 0.5$ signals that the "tail" of $q_{\mathbf{X}}$ in a particular direction outruns the corresponding tail in $p_{\mathbf{X}}$, making such extreme events classified mostly as LO. Consequently, to preserve the unit norm of the distributions, $p_{\mathbf{X}}$ will dominate in other domains of the phase space. Also notice that for $s < 0.5$ both q_s and p_s have a peak close to $s = 0.5$, but for $s > 0.5$ the distributions are more flat reaching relatively far away from $s = 0.5$. The fact that the distribution reaches far away for $s > 0.5$ indicates that $p_{\mathbf{X}}$ vastly dominates $q_{\mathbf{X}}$ in this region of the phase space. Similarly, the peaks just below $s = 0.5$ indicate that this is an area of the phase space where $p_{\mathbf{X}}$ and $q_{\mathbf{X}}$ are nearly equal but covers the majority of events.

[2] From the analytical fact that $x \ln x \to 0$ as $x \to 0^+$.

One important difference between classification across LLF and HLF is that while there are 16 features in the LLF classification, there are only 8 physical degrees of freedom[3] in the direct dislepton production implying that 8 of the features are redundant. These features will be constrained by conservation of 4-momentum, and the effect of this is that some of the information will be hidden away, reducing the KL-divergence estimate. Another difference is the average density of training points in the feature space which is related to the number of events available for each kinematic variable. If we picture the training points organized on a grid in feature space, the average training point densities would correspond to about six grid points in each direction in the HLF case, but only about two grid points per direction in the LLF case. The classifier using HLF has three times as many events in each direction to train with, which increases both the prediction accuracy and the KL-divergence estimate as well as reducing the associated uncertainty. Moreover, we should not necessarily expect a strong correlation between the KL-divergences obtained using classification across LLF and HLF. This is because HLF and LLF represent two entirely different classification problems, effectively exploring two completely different multi variable densities.

As mentioned in chapter 4, the datasets are generated in a rather ideal and simplified manner through MadGraph5 with MadSpin. This raises the important question what effect a more physically appropriate constructed dataset would have on the final KL-divergence estimates – for instance adding the effect of showers and the resulting hadronisation, or adding different detector effects that shows up in real experiments. Ideally, final state particles are modelled as sharp momentum states represented by delta functions at particular momenta which would be captured by a perfect detector. However, a real detector has a finite resolution for measuring and pinpointing the momentum of absorbed particles, resulting in a distribution of measurements centered at a narrow peak with a tiny spread. Therefore, there will be an uncertainty to the true momentum of a particle, and this effect is known as *smearing*. In our case, the smearing would wash out subtle differences between $p_{\mathbf{x}}$ and $q_{\mathbf{x}}$ since the smaller differences can not be resolved by the detectors, resulting in a lower KL-divergence estimate. With showering, the creation of secondary particles such as mesons and pions from hadronisation and electron/anti electron pairs

[3] From chapter 1 the number of degrees of freedom in an unpolarized $2 \to N$ process is $3N - 4$.

from hard photons, alters which particles that are actually observed in the detectors. The process we are considering creates dileptons, missing energy and soft jets (quarks and gluons) in the final state, but the observed particles may be different. This affects the value of measured high level kinematical variables such as leading p_T and $\Delta\phi\big(p_T^{miss}, p_T^{ll}\big)$ since some of the secondary particle tracks can be hard to detect and therefore potentially get lost in the energy balance.

As a note to future work, it would be interesting to use one of the diagnostic methods explained previously to investigate the uncertainty from using poor classifiers. In favor of time, we did not take this uncertainty into account. However, the class ratios behave to a big degree monotonically across the histograms and we do believe the associated uncertainty due to the minor deviations is not bigger than the Monte-Carlo uncertainty.

Chapter 6

Conclusion and Outlook

In this thesis we have investigated a new approach for estimating the Kullback-Leibler divergence (KL divergence) for high-dimensional probability distributions in particle physics. The core of this approach is the use of a trained classifier to estimate the ratio between two pdfs appearing in the KL divergence. Specifically, we applied this method to electroweak production of disleptons with final state dileptons and LSPs as predicted by the MSSM with R-parity conservation. We created two types of kinematic datasets using the event generator MadGraph5, where one was labeled “low level features” with 16 distinct features, and the other with 8 “high level features”. A boosted decision tree, playing the role as the classifier, was trained for each dataset using XGBoost.

As the results of electroweak production of sleptons from the previous chapter show, there is very little to learn about the differential cross section at LO+NLO as compared to LO for the given parameter points. This implies that the kinematic distributions are vastly similar throughout feature space, which we can expect from the nature of electroweak interactions. The noticeable differences are found in the tails where one distribution can dominate the other, which has a significant effect on the the class distributions by making them highly asymmetrical about their means. We picked this process in particular because it stands as a challenging test of the sensitivity of our trained classifier, compared to other processes where NLO kinematics has more influence. Note however that even though the kinematic distributions are virtually identical, the effect of LO+NLO can still be vital in predicting the total cross section which is the integral of the differential cross section over the kinematic phase space.

One aspect which could be investigated more is the impact on the uncertainty in the KL-divergence estimates due to poorly trained classifiers. Moreover, it would be interesting to repeat this exercise more than once using other parameter points in the MSSM. By repeating the process across the parameter space, we can use this method as a tool to see how much there is to gain using LO+NLO kinematics in different parts of the space which might be useful for particle physicists doing parameter scans and model fits. For a single parameter point, the whole procedure from generating and combining event datasets to estimating the KL divergence between LO and LO+NLO kinematics took approximately 1 hour.

Despite being applied to particle physics, this procedure using trained classifiers remains universal and does not make any explicit links to physics which makes the method a wide-applicable tool across different disciplines. It would be interesting to study how our classifer-based approach performs compared to other suggested methods for KL divergence estimation, see e.g. [21, 22, 23]. Currently this remains an open question.

Appendix A

Kinematic Variables

This is a short overview of all the variables used to specify kinematic events for the process defined in eq. (1.51). While the low level features in the left column of table A.1 are self-explanatory, we will below define the high level features shown in the right column.

LLF	**HLF**
p^μ for e^-	m_{ll}
p^μ for e^+	m_T
p^μ for $\tilde{\chi}_1^0$	m_{T2}
p^μ for $\tilde{\chi}_1^0$	h_T
	E_T^{miss}
	E_T^{miss}/h_T
	$\Delta\phi(\mathbf{p}_T^{\text{ll}}, \mathbf{p}_T^{\text{miss}})$
	ΔR_{ll}

Table A.1: Low level features and high level features as the two sets of variables used in this thesis. There are 16 low level features and 8 high level features.

- m_{ll} is the *invariant mass* of a lepton pair. If they have momentum p_1^μ and p_2^μ

respectively, their invariant mass is

$$m_{ll} \equiv \sqrt{(p_1+p_2)_\mu (p_1+p_2)^\mu} = \sqrt{(E_1+E_2)^2 - (\mathbf{p}_1+\mathbf{p}_2)^2} \tag{A.1}$$

- m_T is known as the *transverse mass* of a particular process with missing energy, and it is defined as

  $$m_T = \sqrt{2|\mathbf{p}_T^{ll}||\mathbf{p}_T^{\text{miss}}|(1-\cos\Delta\phi)} \tag{A.2}$$

  where $\mathbf{p}_T^{ll} = \mathbf{p}_{T,1} + \mathbf{p}_{T,2}$ is the total 3-momentum of the lepton pair (the lepton system), and $\mathbf{E}_T^{\text{miss}}$ and $\Delta\phi$ are defined below.

- m_{T2} is known as the *stransverse mass* of a particular process with missing energy, and it is defined as

  $$m_{T2}(\mathbf{p}_{T,1}, \mathbf{p}_{T,2}, \mathbf{p}_T^{\text{miss}}) = \min_{\mathbf{q}_{T,1}+\mathbf{q}_{T,2}=\mathbf{p}_T^{\text{miss}}} \{\max\left[m_T(\mathbf{p}_{T,1}, \mathbf{q}_{T,1}), m_T(\mathbf{p}_{T,2}, \mathbf{q}_{T,2})\right]\} \tag{A.3}$$

- h_T is defined as the scalar sum of $p_T = |\mathbf{p}_T|$ for leptons, which in our case is simply

  $$h_T = p_{T,1} + p_{T_2}. \tag{A.4}$$

- $\Delta\phi(\mathbf{p}_T^{\text{ll}}, \mathbf{p}_T^{\text{miss}})$ is the difference in azimuthal angle between the lepton system and the direction of missing energy (angular difference in the transverse plane).

- ΔR is the distance between the leptons in the angular space spanned by $\phi = \arctan(p_y/p_z)$ and $\eta = -1/2\ln\left[\tan^2(\theta/2)\right]$, *i.e.*,

  $$\Delta R = \sqrt{(\Delta\phi_{ll})^2 + (\Delta\eta_{ll})^2}, \tag{A.5}$$

  where ϕ is the standard azimuthal angle and θ is the polar angle (angle from z-axis to the leptons direction of motion).

Bibliography

[1] K. Cranmer, J. Pavez, and G. Louppe, *Approximating likelihood ratios with calibrated discriminative classifiers*, arXiv:1506.02169.

[2] M. E. Peskin and D. V. Schroeder, *An Introduction to quantum field theory.* Addison-Wesley, Reading, USA, 1995.

[3] M. Thomson, *Modern particle physics.* Cambridge University Press, New York, 2013.

[4] ATLAS: 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*, *Phys. Lett. B* **716** (2012) 1–29, [arXiv:1207.7214].

[5] CMS: S. Chatrchyan *et. al.*, *Observation of a New Boson at a Mass of 125 GeV with the CMS Experiment at the LHC*, *Phys. Lett. B* **716** (2012) 30–61, [arXiv:1207.7235].

[6] S. P. Martin, *A Supersymmetry primer*, *Adv. Ser. Direct. High Energy Phys.* **18** (1998) 1–98, [hep-ph/9709356].

[7] S. Coleman and J. Mandula, *All possible symmetries of the s matrix*, *Phys. Rev.* **159** (1967) 1251–1256.

[8] Particle Data Group: P. Zyla *et. al.*, *Review of Particle Physics*, *PTEP* **2020** (2020) 083C01.

[9] G. Cowan, *Statistical data analysis.* Oxford university press, 1998.

[10] D. J. MacKay, *Information theory, inference and learning algorithms.* Cambridge university press, 2003.

[11] C. E. Shannon, *A mathematical theory of communication*, *The Bell System Technical Journal* **27** (1948) 379–423.

[12] J. R. Hershey and P. A. Olsen, *Approximating the kullback leibler divergence between gaussian mixture models*, in *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07* **4** (2007) IV–317–IV–320.

[13] M. Anderssen, *Performance of deep learning in searches for new physics phenomena in events with leptons and missing transverse energy with the atlas detector at the lhc*, Master's thesis, University of Oslo, Institute for Physics, 2020.

[14] J. Alwall, M. Herquet, F. Maltoni, O. Mattelaer, and T. Stelzer, *MadGraph 5 : Going Beyond*, *JHEP* **06** (2011) 128, [arXiv:1106.0522].

[15] S. Frixione, B. Fuks, *et. al.*, *Automated simulations beyond the Standard Model: supersymmetry*, *JHEP* **12** (2019) 008, [arXiv:1907.04898].

[16] L. Heinrich and M. Feickert, "pylhe: v0.2.1."

[17] ATLAS: G. Aad *et. al.*, *Search for electroweak production of charginos and sleptons decaying into final states with two leptons and missing transverse momentum in $\sqrt{s} = 13$ TeV pp collisions using the ATLAS detector*, *Eur. Phys. J. C* **80** (2020) 123, [arXiv:1908.08215].

[18] ATLAS: G. Aad *et. al.*, *Search for chargino-neutralino production with mass splittings near the electroweak scale in three-lepton final states in $\sqrt{s}$=13 TeV pp collisions with the ATLAS detector*, *Phys. Rev. D* **101** (2020) 072001, [arXiv:1912.08479].

[19] T. Chen and C. Guestrin, *Xgboost: A scalable tree boosting system*, arXiv:1603.02754.

[20] I. Strümke, *Parameter Scans and Machine Learning for beyond Standard Model Physics.* PhD thesis, 2019.

[21] Q. Wang, S. R. Kulkarni, and S. Verdú, *Divergence estimation of continuous distributions based on data-dependent partitions*, *IEEE Transactions on Information Theory* **51** (2005) 3064–3074.

[22] X. Nguyen, M. J. Wainwright, and M. I. Jordan, *Estimating divergence functionals and the likelihood ratio by convex risk minimization*, *IEEE Transactions on Information Theory* **56** (2010) 5847–5861.

[23] M. Kato and T. Teshima, *Non-negative bregman divergence minimization for deep direct density ratio estimation*, 2020.