UNIVERSITY
OF OSLO

Master's thesis

Model independent search for Dark Matter using Machine Learning

In final states with dileptons and Missing Transverse Energy with the ATLAS detector at the LHC

Ruben Guevara

Physics: Nuclear and Particle Physics

60 ECTS study points

Department of Physics

Faculty of Mathematics and Natural Sciences

Spring 2023

Ruben Guevara

Model independent search for Dark Matter using Machine Learning

In final states with dileptons and Missing Transverse Energy with the ATLAS detector at the LHC

Supervisors:

Professor Farid Ould-Saada

Dr. Eirik Gramstad

Abstract

In this master thesis, we compared the performance of a Neural Network (NN) and a Boosted Decision Tree (BDT) Machine Learning (ML) algorithm for the binary classification of Dark Matter (DM) signal and the Standard Model (SM) background. We conducted searches based on models from different theoretical principles that share common experimental signatures, including three models based on a new Z' vector boson coupled to a DM candidate: the Dark Higgs model, Light Vector model, and inelastic Effective Field Theory model. We also studied the direct slepton production model in Supersymmetry, and a Two Higgs Doublet Model with an additional pseudoscalar mediator. In the process of networks optimization we explored methods to mitigate the phenomena of missing variables on datasets, as well as how to weigh simulated samples that have negative weights. Our study involved two approaches: a model dependent approach training one BDT for each model and a model independent approach training three BDTs in kinematically orthogonal regions. We demonstrated that the model independent approach consistently achieved higher mass exclusion limits for all studied models compared to the model dependent approach. These findings demonstrate the efficacy of the model independent approach for new physics searches using ML techniques.

Acknowledgements

First and foremost I want to thank my supervisor Farid Ould-Saada for giving me the opportunity of pursuing this project and supporting my crazy ideas, such as me trying to make a QML algorithm to use in HEP, and to include new models to update my results two weeks before I hand in. I also want to thank Eirik Gramstad for putting up with my constant nagging and the in depth discussions we had about ML.

Finish this!

Most importantly I want to thank the coffee machine at the section for High Energy Particle Physics for the incredible support in times of need, as the coffee was free.

Contents

List of Figures

List of Tables

List of Acronyms

SM Standard Model

DM Dark Matter

MET Missing Transverse Energy

DH Dark Higgs

LV Light Vector

EFT Effective Field Theory

HDS Heavy Dark Sector

LDS Light Dark Sector

SUSY Supersymmetry

2HDMa Two Higgs Doublet Model with a pseudoscalar a

ML Machine Learning

NN Neural Network

BDT Boosted Decision Tree

Chapter 1

Introduction

After the Higgs boson, the last ingredient of the Standard Model (SM), was discovered in 2012 [1], we still question its nature. Is it the long-awaited SM particle or the first of a series of scalar particles? With the advent of higher energies and higher collision rates, the Large Hadron Collider (LHC) continues the voyage towards new physics phenomena. The ambitious LHC physics programme may shed light on some of the greatest mysteries in physics today. The focus of this work is to perform an independent search for new physics phenomena, such as evidence of a Dark Matter (DM) candidate and its plausible mediator using supervised learning. A Neural Network (NN) and a Boosted Decision Tree (BDT) will be trained on various Monte Carlo (MC) simulated data samples featuring well-defined predictions from a set of new physics theories. The analysis uses proton-proton (pp) collisions at the LHC from Run II at 13 TeV, recorded by the ATLAS detector. The data consists of dilepton final states with Missing Transverse Energy (MET). By comparing the empirical data collected from Run II with SM simulated samples we will carefully select the features to be used in the training phase of the Machine learning (ML) networks.

One of the biggest motivations for this project is to search for Dark Matter (DM), one of the biggest mysteries in science to this day. It can be a Weakly Interacting Particle (WIMP) [2] or the Axion [3] which are both postulated by cosmological constraints. This project will focus on DM particles that are described by WIMP theories. The primary focus is to set up a NN and a BDT to be trained on a set of models predicting DM particles, such as supersymmetry, two Higgs doublet models, simplified DM models involving

a mediator, including those models based on effective field theory (EFT). This way we aim at a model independent search of a DM particle, possibly together with its mediator.

We concentrate on dilepton and Missing Transverse Energy (MET) final states, $pp \rightarrow ll\chi\chi = ll$ MET. Where the MET is due to DM since we cannot detect it directly in particle detectors. There are various Beyond Standard Model (BSM) models with a DM candidate χ leading to this process final state. The standard practice in new physics searches today is to choose one model and do a thorough data analysis to test it. However, the emphasis of this project as aforementioned will be to let an ML algorithm learn the features of various DM models predicting a dilepton + MET final state, thus optimizing relevant signal search regions, based on, among others, the invariant mass of the 2 leptons and the missing transverse energy, thus reaching better search sensitivities. Utilizing the powerful tool of ML it might help recognize a pattern that is common through all the different models studied, at least within each of the signal regions defined, which might in turn bring us closer to identifying an empirical signal of DM in the collected data. The goal is to study various BSM models, such as: Mono-Z', Dark Higgs, Light Vector and inelastic EFT [4], Two Higgs Doublet Model with an additional pseudoscalar [5] and Supersymmetry [6]. These models are built upon different theoretical principles, making them phenomenologically different from each other, with some common experimental features. Thus, this approach of building generic ML algorithms to be simultaneously trained on all of the models could help reduce the computational time needed testing new models in the future.

This thesis is organized into three main parts: Background, Methods, and Results. The Background part provides a theoretical foundation, while the Methods part details the data preparation and machine learning optimization techniques. Finally, the Results part presents the findings of the thesis, as well as the conclusion and outlook.

We will start this thesis by introducing the theoretical foundation behind the SM using Quantum Field Theory in Chapter 2. Thereafter, we will present the theory behind the DM models we will study in Chapter 3. After the field theory description of the background and signal that will be used on the ML algorithms, we will present how we

can actually measure anything from this, both from a kinematical and experimental point of view. Afterwards, we will present the ATLAS detector, the cut and count method of searching for new physics, including the statistical tools that we will utilize, all of this is in Chapter 4. After that we will give an overview of both NNs and BDTs, as well as the tools that will be used to evaluate their performances in Chapter 5.

In Chapter 6 we will present the methods we will use to prepare the dataset for ML, meaning the event selection as well as the feature selection. In this chapter we will discuss the challenges that arise using when making the datasets for an ML study. After that we will show the number of events in the dataset and how we plan to do our model independent approach, this we do in Chapter 7. In this chapter we present the challenges that arise with the datasets for NNs and BDTs, and what methods we use to mitigate these challenges. Lastly, we will present the results in Chapter 8 and discuss the results in Chapter 9.

Part I

Background

Chapter 2

The Standard Model of Particle Physics

The Standard Model (SM) of particle physics is the framework of particle physics, it unveils the secrets of the universe's building blocks and their interactions. With its framework, it offers a rich understanding of the structure and behavior of matter, solidifying its place as a bedrock in the realm of modern physics. At its core, the Standard Model describes three of the fundamental forces of nature: the electromagnetic force, the weak force, and the strong force. These forces are mediated by particles known as gauge bosons, which act as carriers of the forces between matter particles.

As the SM is described using Quantum Field Theory (QFT), we need the description of what a *field* is. A quantum field is a fundamental concept in quantum physics that describes the underlying fabric of reality, where particles and their interactions are represented as excitations or vibrations of these fields. In this chapter we will delve into every of the aforementioned forces, the spontaneous symmetry breaking phenomenum that gives mass to particles, as well as how to calculate the cross-section of processes. We will start by describing electromagnetism in terms of QFT, namely Quantum Electrodynamics.

The theory of this section is mainly based on Peskin's and Schroeder's "An Introduction to Quantum Field Theory" [7] and partly on Thomson's "Modern Particle Physics" [8].

2.1 Quantum Electrodynamics

In the beginning there was nothing; *then God said, "Let there be light," and there was light.* The first part of the Standard Model, Quantum Electrodynamics, or QED for short, described the processes in nature involving the photon. The Lagrangian of QED can be seen below

$$\mathcal{L}_{QED} = \overline{\psi}\left(i\not{D} - m\right)\psi - \frac{1}{4}B^{\mu\nu}B_{\mu\nu} \tag{2.1}$$

with $\overline{\psi} = \psi^{\dagger}\gamma^0$, where ψ is a bispinor field of spin-1/2[1], m is the mass of ψ, γ^μ are the Dirac matrices, defined as

$$\gamma^0 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix}, \quad \gamma^1 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix},$$

$$\gamma^2 = \begin{pmatrix} 0 & 0 & 0 & -i \\ 0 & 0 & i & 0 \\ 0 & i & 0 & 0 \\ -i & 0 & 0 & 0 \end{pmatrix}, \quad \gamma^3 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \\ -1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}$$

Furthermore we have $\not{D} \equiv \gamma^\mu D_\mu$ where $iD_\mu = i\partial_\mu - eB_\mu$ is the covariant derivative, where e is the coupling constant[2], $B_{\mu\nu} = \partial_\mu B_\nu - \partial_\nu B_\mu$ is the electromagnetic field tensor[3]. B_μ is the covariant four-potential[4] of the electromagnetic field, also known as the gauge field.

The above Lagrangian is described by a $U(1)$ symmetry group, meaning that it corresponds to the *unitary* group of one unitary matrix with determinant 1[5]. The consequence of QED being described by this symmetry group is, recalling Noether's theorem, that there is only one unique degree of freedom, this is the rotation of phase angle, $\alpha(x)$,

[1] In QED, it is the electron or positron field

[2] Which in QED is the electric charge of ψ.

[3] Taking $\partial_\mu B^{\mu\nu} + \star\partial_\mu B^{\mu\nu} = 0$ gives all of Maxwell's equations.

[4] Defined as $B_\mu = (\frac{1}{c}\phi, \mathbf{B})$ where ϕ is the electric potential and $\mathbf{B}$ is the magnetic potential we know and love from electrodynamics.

[5] For more information about group theory we refer the reader to Georgi's "LIE ALGEBRAS IN PARTICLE PHYSICS. FROM ISOSPIN TO UNIFIED THEORIES" [9]

of the field ψ. This means that

$$\psi \to \psi' = e^{ie\alpha}\psi \tag{2.2}$$

Inserting ψ' into the Eq. (2.1) yields that $\mathcal{L} \to \mathcal{L}' = \mathcal{L}$, meaning that our theory is locally gauge invariant.

2.2 Yang-Mills Theory

While QED is expressed by the Lagrangian in Eq. (2.1), which is part of a $U(1)$ symmetry group, it can still be described by a more general Lagrangian using *Yang-Mills* theory. The differences between the Yang-Mills and QED Lagrangian being that the gauge freedom changes from a plane rotation of the phase angle α of the field ψ, to a more general gauge in α^a in the field f_i (using now the infinitesimal transformation notation)

$$f_i \to \left(1 + ig\alpha^a t^a + \mathcal{O}(\alpha^2)\right) f_i$$

The covariant derivative also changes to a more general

$$iD_\mu = i\partial_\mu - eB_\mu \to iD_\mu = i\partial_\mu - gA^a_\mu t^a$$

where e is replaced by a coupling constant g, the vector field B_μ changes to a more general A^a_μ which transforms as

$$A^a_\mu \to A^a_\mu + \frac{1}{g}\partial_\mu \alpha^a + f^{abc} A^b_\mu \alpha^c$$

where f^{abc} is a set of numbers called the *structure constants*. The important thing to note about the Yang-Mills is that the covariant derivative is *non-Abelian*[6], meaning that the commutator of the operator becomes

$$[D_\mu, D_\nu] = -igF^a_{\mu\nu} t^a$$

this new t^a factor is a Lie algebra generator, which for our purposes represent a local gauge symmetry we have on the field. The structure constant has to fulfill the criteria

[6]Except in the trivial case where we use the $U(1)$ symmetry group

that

$$[t^a, t^b] = if^{abc}t^c \tag{2.3}$$

The difference between the Yang-Mills and the QED Lagrangian is most noticeable when fully writing

$$F^a_{\mu\nu} = \partial_\mu A^a_\nu - \partial_\nu A^a_\mu + gf^{abc}A^b_\mu A^c_\nu$$

Where the last term takes into account the *self interaction* of bosons. Adding all of this up we get the following Lagrangian

$$\mathcal{L}_{YM} = \overline{f}\left(i\not{D} - m\right)f - \frac{1}{4}F^a_{\mu\nu}F^{\mu\nu}_a \tag{2.4}$$

where we use the notation $f \equiv \begin{pmatrix} f_1 & f_2 & \cdots & f_n \end{pmatrix}^T$ where f_i is a field. Using the Euler-Lagrange equation with the Yang-Mills Lagrangian above yields the equation of motion

$$\partial^\mu F^a_{\mu\nu} + gf^{abc}A^{b\mu}F^c_{\mu\nu} = -g\overline{\psi}\gamma_\nu t^a\psi \tag{2.5}$$

As using Noether's theorem on the equation of motion gives us the conserved quantity of the theory, we will apply it for every symmetry group in the SM. There is also a theorem[7] that states that for an $SU(n)$ symmetry group there are $n^2 - 1$ vector fields.

If we were to choose $U(1)$ as our local symmetry group the generator $t^a = \mathbb{I}$ and all the structure constants are $f^{abc} = 0$, meaning that we get the QED back. Noether's theorem states that the conserved quantity of the Lagrangian is the electric charge e.

2.3 Weak theory

Using the Yang-Mills Lagrangian from Eq. (2.4) on $SU(2)$, choosing $t^a = \frac{\sigma^i}{2}$, where σ^i are the Pauli matrices given as

$$\sigma^1 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \sigma^2 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}, \quad \sigma^3 = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \tag{2.6}$$

[7] See Georgi's book [9]

The structure constants are the Levi-Civita $f^{abc} = \epsilon^{ijk}$. Where $i, j, k \in [1, 2, 3]$. The Weak theory gives us the gauge freedoms of

$$f \to \left(1 + i\alpha^i \sigma^i + \mathcal{O}(\alpha^2)\right) f$$

Using the notation of $A_\mu \to W_\mu$ we get that there are $2^2 - 1$, meaning three vector fields

$$W_\mu^i \to W_\mu^i + \frac{1}{g}\partial_\mu \alpha^i + \epsilon^{ijk} W_\mu^j \alpha^k$$

We cannot directly apply Noether's theorem on this Lagrangian, as it is not renormalizable, but when arriving to *electroweak* theory we would get that two quantities are conserved[8], the *hypercharge* and *isospin*. In this theory we have isospin doublets, meaning

$$f = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix} \to \begin{pmatrix} U \\ D \end{pmatrix}$$

For *up and down type* particles. i.e. $\begin{pmatrix} l \\ \nu_l \end{pmatrix}$ for leptons, and $\begin{pmatrix} u \\ d \end{pmatrix}$ for quarks.

[8] See Section 2.5

2.4 Quantum Chromodynamics

Using the Yang-Mills Lagrangian from Eq. (2.4) on an $SU(3)_C$, choosing $t^a = \frac{\lambda^a}{2}$, where λ^a, the *Gell-Mann matrices*, are the $SU(3)$ equivalent of the $SU(2)$ Pauli matrices:

$$\lambda^1 = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \lambda^2 = \begin{pmatrix} 0 & -i & 0 \\ i & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}, \quad \lambda^3 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{pmatrix},$$

$$\lambda^4 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}, \quad \lambda^5 = \begin{pmatrix} 0 & 0 & -i \\ 0 & 0 & 0 \\ i & 0 & 0 \end{pmatrix}, \quad \lambda^6 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}, \tag{2.7}$$

$$\lambda^7 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -i \\ 0 & i & 0 \end{pmatrix}, \quad \lambda^8 = \frac{1}{\sqrt{3}} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & -2 \end{pmatrix}$$

The structure constant have to obey Eq. (2.3), thus we get

$$\begin{aligned} f^{123} &= 1 \\ f^{147} = -f^{156} = f^{246} = f^{257} = f^{345} = -f^{367} &= \frac{1}{2} \\ f^{458} = f^{678} &= \frac{\sqrt{3}}{2} \end{aligned}$$

Using the notation of $A_\mu \to G_\mu$ we get that there are $3^2 - 1$, meaning *eight* vector fields

$$G^a_\mu \to G^a_\mu + \frac{1}{g_s}\partial_\mu \alpha^a + \epsilon^{abc} G^b_\mu \alpha^c$$

If one were to use Noether's theorem on the $SU(3)_C$ Yang-Mills Lagrangian, we would get that there are three conserved quantities for this symmetry group. These are the colors, r, g and b. With all the information above we can fully describe Quantum Chromodynamics, which describe quark interactions. The eight vector fields are the eight different gluons that are in the theory.

In this theory we have color triplets

$$f = \begin{pmatrix} f_1 \\ f_2 \\ f_3 \end{pmatrix}$$

where f_i is the color state of the particle we are studying.

2.5 Electroweak unification and the BEH Mechanism

We have so far introduced the three symmetry groups that form the SM, but this is not the full picture yet. To get the full SM we need to talk about the electroweak $U(1)_Y \otimes SU(2)_L$ unification that can be explained when studying the spontaneous symmetry breaking (SSB) of the electroweak symmetry.

If we introduce a new complex scalar field ϕ that transforms under the $SU(2)$ representation, we can write the Lagrangian

$$\mathcal{L} \supset \mathcal{L}_\phi = (D_\mu \phi)^\dagger (D^\mu \phi) - \mu^2 (\phi^* \phi) - \lambda^2 (\phi^* \phi) \tag{2.8}$$

with

$$D_\mu \phi = \partial_\mu \phi - igW_\mu^a \sigma^a - ig' y_\phi \phi \tag{2.9}$$

where W_μ and B_μ are vector fields of the $SU(2)_L$ and $U(1)_Y$ group respectively. And where $\mu^2 (\phi^* \phi) - \lambda^2 (\phi^* \phi) = V(\phi)$ is the potential of the new complex scalar field. After developing *phi* we can minimize the potential to find the minima we can rewrite it as $V(\phi) = -\lambda \left(\phi^\dagger \phi - \frac{v^2}{2} \right)^2$. After rotating away the three phases due to $SU(2)_L$ gauge symmetry we can arrive at

$$\phi = \begin{pmatrix} 0 \\ v + H(x) \end{pmatrix} \tag{2.10}$$

where H is real scalar field. This expands the potential to be

$$V(H) = \frac{1}{2} m_H^2 H^2 + \frac{1}{2v} m_H^2 H^3 + \frac{1}{8v^2} m_H^2 H^4 \tag{2.11}$$

where we have used that $m_H = \sqrt{2\lambda}v$. This potential is the so-called *Sombrero potential*, and we can reveal that this scalar field H is the Higgs boson. The consequences of writing our complex scalar field using Eq. (2.10) comes when from the shape of the potential, this gives rise to the mass of particles. The SSB comes from the fact that the fields have degenerate minima on the potential but can move around in a higher dimensional space (around the sombrero edge) to get to the other minima.

Another thing that follows from Eq. (2.10) is that expanding on Eq. (2.9) gives rise (after a lot of math) to the following relations

$$W_\mu^\pm = \frac{W_\mu^1 \mp iW_\mu^2}{\sqrt{2}} \tag{2.12}$$

$$Z_\mu = \frac{gW_\mu^3 - g'B_\mu}{\sqrt{g^2+e^2}} \tag{2.13}$$

$$A_\mu = \frac{gB_\mu - g'W_\mu^3}{\sqrt{g^2+e^2}} \tag{2.14}$$

These are the W and Z boson as well as the photon, respectively. To explicitly show how the SSB phenomena gives mass to every other particle, one can transform the Lagrangian to the the unitary gauge, this adds an extra term on the Lagrangian called the $Yukawa$ Lagrangian

$$\mathcal{L}_{YU} = \psi_i y_{ij} \psi_j \phi$$

where $y_{i,j}$ are the Yukawa couplings between the scalar, ϕ, and fermion field, ψ. As we are looking at a complex scalar field in the $SU(2)$ representation we have from the Yang-Mills theory

$$f = \begin{pmatrix} U \\ D \end{pmatrix}$$

where U, D is the field *up and down type* spinor. For example this can be the first generation quark two component spinor $Q = \begin{pmatrix} u \\ d \end{pmatrix}$ consisting of the *up* and *down* quark or the lepton two component spinor $L = \begin{pmatrix} l \\ \nu_l \end{pmatrix}$. The standard for weak interactions is to

define $right$ and $left$ handed helicity using the operators

$$L = \frac{1}{2}(\mathbb{I} - \gamma_5), \qquad \text{and} \quad R = \frac{1}{2}(\mathbb{I} + \gamma_5)$$

where $\gamma_5 = i\gamma^0\gamma^1\gamma^2\gamma^3$. Such that we can write

$$U_L = L \begin{pmatrix} U \\ D \end{pmatrix} = \begin{pmatrix} U \\ 0 \end{pmatrix}, \qquad \text{and} \quad D_R = R \begin{pmatrix} U \\ D \end{pmatrix} = \begin{pmatrix} 0 \\ D \end{pmatrix}$$

as well as their hermitian conjugates.

2.6 S-matrix expansion

But how do we predict the occurrences of processes? To explain this generally, we can conduct an S-matrix expansion. To be more precise and use terms from quantum field theory rather than quantum mechanics, we can use the scattering operator $\mathcal{S}$ on the interaction term of the SM Lagrangian $\mathcal{L}_{int}$. Starting by defining the general operator

$$\begin{gathered} \mathcal{S} = \sum_{i=vertices} \mathcal{S}^{(i)}, \qquad \text{where} \quad \mathcal{S}^{(0)} = \mathbb{I} \qquad \text{and} \\ \mathcal{S}^{(n)} = (-i)^n \mathcal{T} \int_{-\infty}^{\infty} d^4\mathbf{x}_1 \cdots \int_{-\infty}^{\infty} d^4\mathbf{x}_n \left[\mathcal{L}_{int}(x_1) \cdots \mathcal{L}_{int}(x_n)\right] \end{gathered} \tag{2.15}$$

where $\mathcal{T}$ is the *time operator* ordering the terms such that we predict our desired transition amplitude, and $\mathbf{x}_i$ are coordinates in spacetime. The $\mathcal{S}$ operator sums over all *vertices* of a *Feynman diagram*. To calculate the probability amplitude we can start with an easy example using QED, the Bhabha scattering process $e^-e^+ \to e^-e^+$, in particular the annihilation channel where we have $e^+e^- \to \gamma$ on a vertex level. The interaction Lagrangian of interest for this process is $\mathcal{L}_{int} = e\overline{\psi}(x)\gamma^\mu\psi(x)A_\mu$. We use the scattering operator in the following way

$$\langle f| \mathcal{S}^{(n)} |i\rangle \to \left\langle e^-e^+\right| S^{(2)} \left|e^-e^+\right\rangle$$

$$= \left\langle e^-e^+\right| \left(-e^2\mathcal{T} \int_{-\infty}^{\infty} d^4\mathbf{x} \int_{-\infty}^{\infty} d^4\mathbf{y} \overline{\psi}(x)\gamma^\mu\psi(x)A_\mu(x)\overline{\psi}(y)\gamma^\nu\psi(y)A_\nu(y)\right) \left|e^-e^+\right\rangle$$

Starting by using the time operator $\mathcal{T}$ on the integrals, which gives us the *normal order* of the operations, we can start by using *Wicks theorem* to "connect" the two points in spacetime $\mathbf{x}$ and $\mathbf{y}$. As we know there are no photons in the initial or final state, we can connect the photons as *propagators*, meaning

$$\mathcal{S}^{(2)} = -\frac{e^2}{2}\int_{-\infty}^{\infty} d^4\mathbf{x}\int_{-\infty}^{\infty} d^4\mathbf{y}\mathcal{N}\left(\overline{\psi}(x)\gamma^\mu\overbrace{A_\mu(x)\psi(x)\overline{\psi}(y)\gamma^\nu}A_\nu(y)\psi(y)\right)$$

where

$$\overbrace{A_\mu(x)A_\nu(y)} = \langle 0|\,\mathcal{T}(A_\mu(x)A_\nu(y))\,|0\rangle = \cdots = D^F_{\mu\nu}(x-y) = \int_{-\infty}^{\infty}\frac{d^4q}{(2\pi)^4}D^F_{\mu\nu}(q)e^{-iq(x-y)}$$

with

$$D^F_{\mu\nu}(k) = \frac{-g_{\mu\nu}+r_{\mu\nu}}{(q^2+i\epsilon)}, \tag{2.16}$$

where $q = q_\mu q^\mu$ is the contracted four-momentum of the particle, and $r_{\mu\nu} = \frac{q_\mu q_\nu}{q^2}$ we will set to zero as we will use the Feynman gauge. Here we have omitted the mathematical details, but to summarize it with words. We calculated the photon propagator which connects two spacetime points $\mathbf{x}$ and $\mathbf{y}$ by calculating the Greens function of the Minkowski space using among other the residue theorem from complex analysis. If we were now to define $|e^-e^+\rangle, \langle e^-e^+|, \overline{\psi}$ and ψ we could directly calculate the scattering amplitude by setting in the values, as this usually involves solving integrals that use Dirac delta function[9]. To complete our example we define

$$\left|e^-e^+\right\rangle = (\sqrt{2E_p}a_p^\dagger)(\sqrt{2E_k}b_k)\,|0\rangle\,, \qquad \text{and} \quad \left\langle e^-e^+\right| = (2\sqrt{E_pE_k})\,\langle 0|\,a_pb_k^\dagger$$

where E_i is the energy of the particle and a_i, b_i and $a_i^\dagger, b_i^\dagger$ are creation and annihilation operators, respectively. In general fermion fields have

$$\psi \equiv \int\frac{d^3p}{(2\pi)^3}\frac{1}{\sqrt{2E_\mathbf{p}}}\left(\sum_s a^s_\mathbf{p}u^s(p)e^{-ip\cdot x} + b^{s\dagger}_\mathbf{p}v^s(p)e^{ip\cdot x}\right) = \psi_p^+ + \psi_p^-$$

Where, depending on ψ^+ (for $u(p)$) and ψ^- (for $v(p)$) are the particle and antiparticle fields, respectively. It is standard convention to say that ingoing particles and anti-particles on an interaction are written as ψ^+ and $\overline{\psi}^-$, respectively, and their hermitian

[9] At least while omitting loops, which we will do in this thesis

conjugate when they are outgoing. With this information we can connect the initial and final particles to their fields, solving the bra we have

$$\left\langle e^- e^+ \right| \overline{\psi}(x)\psi(x) = (2\sqrt{E_p E_k}) \left\langle 0 \right| a_p b_k^\dagger \overline{\psi}_{k'}^- \psi_{p'}^+,$$

Giving

$$\langle e^- e^+ | \, \overline{\psi}(x)\psi(x) = \langle 0 | \int \tfrac{d^3p'}{(2\pi^3)} \int \tfrac{d^3k'}{(2\pi^3)} \sqrt{\tfrac{E_p E_k}{E_{p'} E_{k'}}} \left(\textstyle\sum_{s,t} a_{\mathbf{p}'}^{\dagger,s} u^s(p') e^{-ip'\cdot x} + b_{\mathbf{k}'}^t \overline{v}^t(k') e^{-ik'\cdot x} \right) a_p b_k^\dagger$$

using the relation $a_{p's} a_{pt}^\dagger = (2\pi)^2 \delta^{(3)}(p' - p) - a_{p's}^\dagger a_{pt}$ where the latter part kills vacuum, we get

$$\left\langle e^- e^+ \right| \overline{\psi}(x)\psi(x) = u(x)\overline{v}(x) \left\langle 0 \right|$$

Doing the same for the ket, and setting all of this into the initial equation we get

$$\langle f | \, \mathcal{S}^{(n)} \, | i \rangle = -\frac{e^2}{2} \langle 0 | \int_{-\infty}^{\infty} d^4\mathbf{x} \int_{-\infty}^{\infty} d^4\mathbf{y} u(x)\gamma^\mu \overline{v}(x) D_{\mu\nu}^F(x-y) \overline{u}(y)\gamma^\nu v(y) \, |0\rangle$$

Giving us

$$(2\pi)^4 \delta^{(4)}(p_1 + p_2 - p_1' - p_2') \overline{v}(p_1)\gamma^\mu u(p_2) \frac{g_{\mu\nu}}{q^2} \overline{u}(p_1')\gamma^\nu v(p_2')$$

we can relate the scattering amplitude to a variable called the *transition amplitude*, $\mathcal{M}_{fi}$ by using

$$\langle f | \, \mathcal{S} \, | i \rangle = i(2\pi)^4 \delta^{(4)} \left(\sum_i p_i - \sum_f p_f \right) \mathcal{M}_{fi}$$

Meaning that we have

$$\mathcal{M}_{bhabha} = \overline{v}(p_1)\gamma^\mu u(p_2) \frac{g_{\mu\nu}}{q^2} \overline{u}(p_1')\gamma^\nu v(p_2')$$

The reason we did all this is to calculate the *differential cross-section* (in the center of mass frame) from this, using

$$\frac{d\sigma}{d\Omega} = \frac{1}{64\pi^2 s} \frac{p_f *}{p_i *} |\mathcal{M}_{fi}|^2 \tag{2.17}$$

where $p_f *$ and $p_i *$ are the magnitudes of the final- and initial-state momenta, respectively. From the differential cross-section we can also calculate the cross-section by integrating

over $d\Omega$.

$$\sigma = \int_0^{2\pi} \int_0^{\pi} \frac{1}{64\pi^2 s} \frac{p_f*}{p_i*} |\mathcal{M}_{fi}|^2 \sin\theta d\theta d\phi \tag{2.18}$$

This method is a general way to calculate the cross-section of an event we want to study, but usually one just constructs the matrix element $\mathcal{M}_{fi}$ from *Feynman diagrams*. In this calculation we already introduced the photon propagator in Eq. (2.16), and we implicitly simplified it to be $\frac{g_{\mu\nu}}{q^2}$, which is valid for our purposes, and can be used in any event where the photon is the propagator. Feynman realized this and made Feynman rules to construct diagrams. The rules are shown in Figure 2.1.

Meaning that following these rules we just constructed the Feynman diagram for Bhabha scattering shown in Figure 2.2

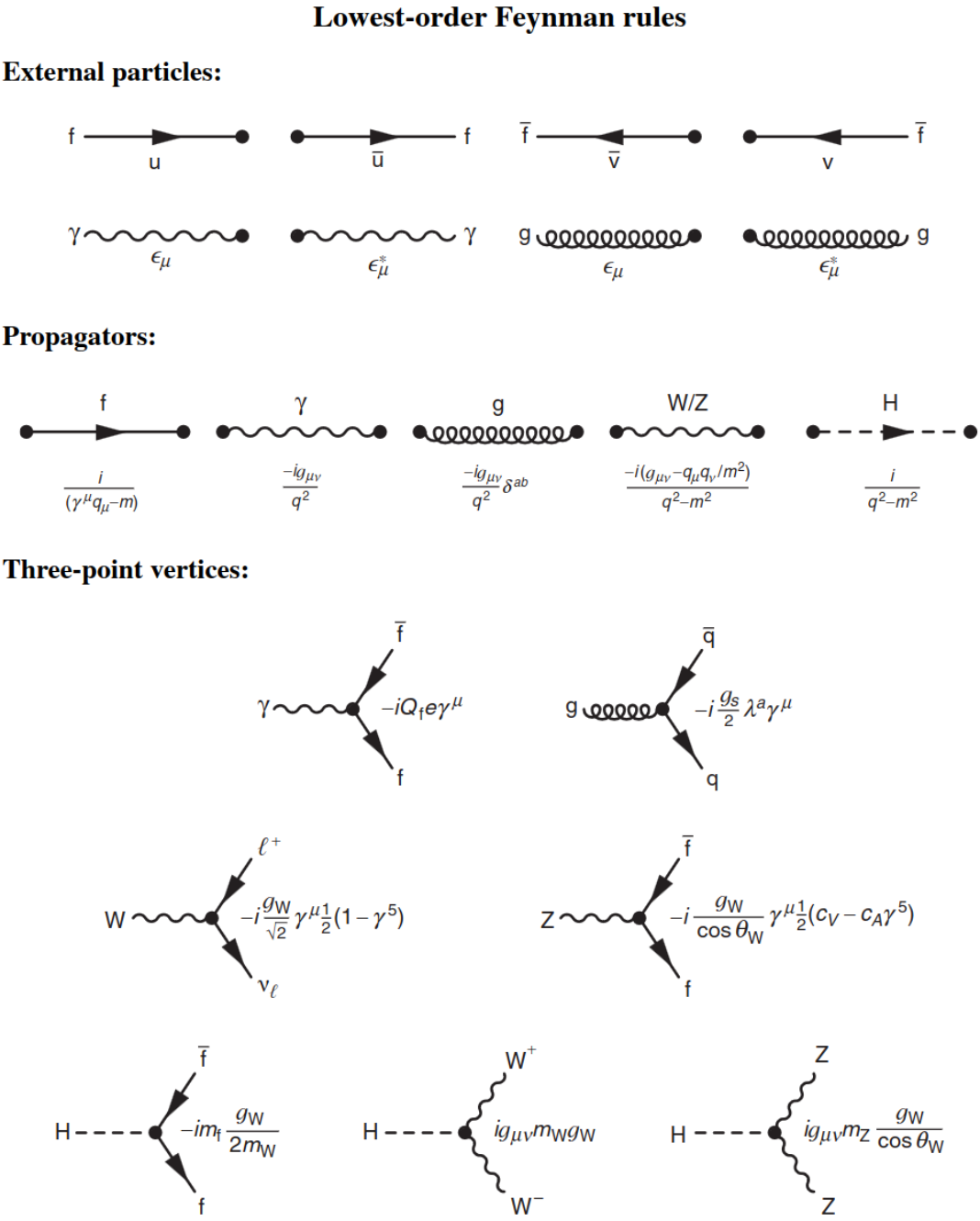


Figure 2.1: The Standard Model Feynman rules. Image taken from Thomson's "Modern Particle Physics" [8]

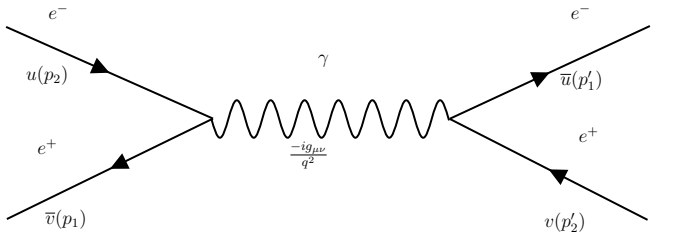


Figure 2.2: Feynman diagram of Bhabha scattering annihilation

2.7 Adding it all up

Combining the three groups $U(1)_Y \otimes SU(2)_L \otimes SU(3)_C$ as well as introducing the Higgs field from Eq. (2.10) gives us the SM Lagrangian. To give an illustration of the elements of the SM we can see Figure 2.3

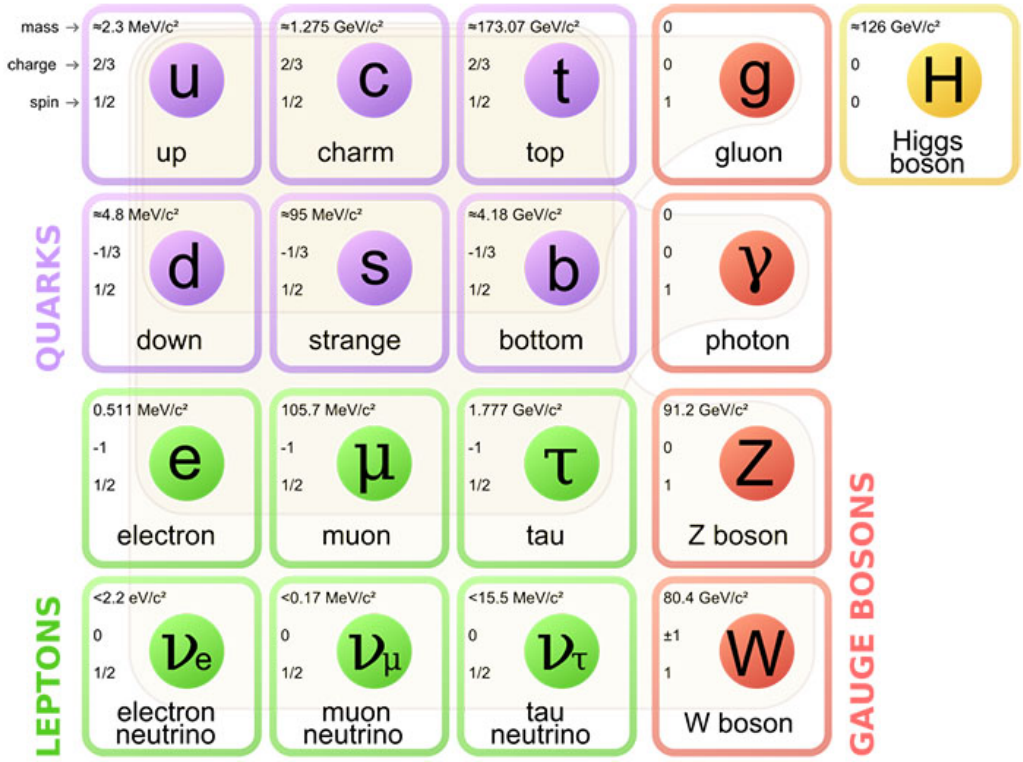


Figure 2.3: The Standard Model of particle physics. Image taken from Wikipedia [10]

From Figure 2.3 we can see all the particles of the standard model. These include the quarks u, d, c, s, t and b, the leptons e, μ and τ as well as their neutrino partners, the carrier of the strong force g, the carriers of the weak force $W^{\pm}$ and Z, the carrier of the electromagnetic force γ and the scalar Higgs boson H. From the figure we can also see which particles can interact with each of the bosons.

These are the building blocks of nature, as mesons, subatomic particles consisting of quark-antiquark pairs, and baryons, meaning subatomic particles consisting of three quarks (such as the proton and neutron) can further build atoms with electrons to furthermore create chemistry via quantum mechanical processes.

However, although the SM successfully accounts for the majority of observable matter in the universe, it represents only approximately 5% of the total energy content. Approximately 70% of the universe is composed of *Dark Energy* which we know is there because of the universe's expansion, and is represented in the cosmological constant in Einsteins Field Equations [11], and the remaining 25% of the energy in the universe is *Dark Matter*. The next chapter of this thesis will present a brief description of Dark Matter, and present the Beyond SM models with Dark Matter candidates we will study.

Chapter 3

Beyond Standard Model Dark Matter

While the standard model of particle physics has been incredibly successful in describing the behavior of subatomic particles, it is not a complete description of the universe. For example, the standard model cannot accommodate the mysterious gravitational forces that hold galaxies together, and it fails to account for the abundance of matter that we observe in the universe.

These mysteries have led scientists to propose the existence of Dark Matter (DM), a mysterious substance that makes up a significant fraction of the matter in the universe. The name DM, stems from the understanding that while most DM particles do not interact with matter via any of the forces described by the SM. However, it is important to note that certain DM candidates can indeed interact weakly through the weak force as described by the SM. This makes it extremely difficult to detect directly. However, its presence can be inferred through its gravitational effects on visible matter, such as stars and galaxies, which is why we know it is there [12, 13]. In this thesis we will look at so-called Weakly Interacting Massive Particles (WIMP) DM candidates. There is no formal definition of a WIMP, but broadly it is a new elementary particle which interacts via gravity, possibly the weak interaction, and potentially other forces outside the SM.

There are several extensions of the SM that include DM candidate particles interacting via a so-called Beyond Standard Model (BSM) Lagrangian. Such as Supersymmetry [6], Universal Extra Dimensions [14] and little Higgs models [15]. In this chapter we will present the BSM models we will be studying that have a dilepton and MET final state.

3.1 Mono-Z' candidates

Six of the models that will be studied in this thesis come from a new theoretical gauge boson that behaves like much heavier Z boson. The theory is based on the papers by Bauer et al. [4]. The models we will study assume a $U(1)'$ symmetry for a new Z' gauge boson, from this we can extend the SM Lagrangian, to include the Z' coupling to the SM particles by

$$\mathcal{L} \supset -\sum_q g_q \bar{q} \gamma^\mu q Z'_\mu \tag{3.1}$$

where the coupling g_q is a free parameter of the model. The three models we will study are based on three interactions of this new Z' (decaying into two leptons) and DM candidates χ which we describe now.

3.1.1 Z' Dark Higgs

The DM Z' models come with an additional scalar responsible for SSB. A new scalar particle that couples to the Z', which we will call the Dark Higgs (DH), h_D, plays the role of portal to DM. Analogous to the SM process of Higgs-boson radiation from a W or Z, the DH is radiated from the Z' in a dark-Higgsstrahlung process. In addition, we will assume that this DH boson couples with invisible states, such as DM. A minimal model for this process, with the $U(1)'$ symmetry with a charged scalar field Φ_D as well as an invisible singlet scalar ϕ_χ:

$$\mathcal{L} \supset |D_\mu \Phi_D|^2 + \mu_D^2 |\Phi_D|^2 - \lambda_D |\Phi_D|^2 - \frac{1}{4}(F'_{\mu\nu})^2 + \frac{1}{2}(\partial_\mu \phi_\chi) - \lambda_\chi |\Phi_D|^2 \phi_\chi^2 - V(\phi_\chi) \tag{3.2}$$

with $\Phi_D = \frac{1}{\sqrt{2}}(v_D + h_D)$, thus obtaining a vacuum expectation value giving mass to the Z', $m_{Z'}$. The coupling of the DH with Z' is

$$Q_h g_Z m_{Z'} h_D Z'_\mu Z'^\mu \equiv g_{h_D} m_{Z'} h_D Z'_\mu Z'^\mu \tag{3.3}$$

where Q_h is the charge of the Φ_D which is a free parameter absorbed in the coupling g_{h_D} above. The Feynman diagram for this dark Higgs model is depicted in Figure 3.1

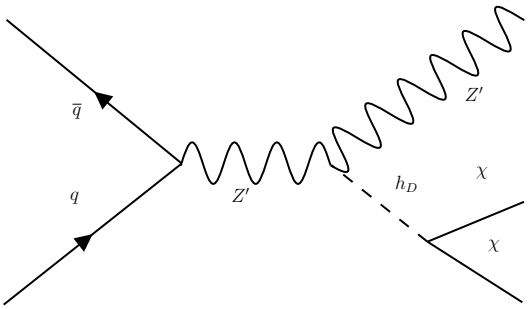


Figure 3.1: Z' Dark Higgs Model

3.1.2 Z' Light Vector

The second mono-Z' model we will look at, is when the Z' is relatively light such that it can be produced in $q\overline{q}$ annihilation and decays to dark states. As an example we can consider a Z' coupled to a new fermion which has both Dirac and Majorana masses, M_d and M_m respectively. Where the Majorana mass can be generated from the $U(1)'$ through a $y_\chi \Phi_\chi \overline{\chi}\chi^c$, such that

$$\mathcal{L} \supset \overline{\chi}(i\not{D} - M_d)\chi - \frac{M_m}{2}(\overline{\chi}\chi + h.c)$$

This leads to two Majorana states $\chi_{1,2}$ with masses $M_{1,2} = |M_m \pm M_d|$. We will study the process $Z' \to \chi_1\chi_2$, where $\chi_2 \to Z'\chi_1$. This model is represented in the Feynman diagram in Figure 3.2

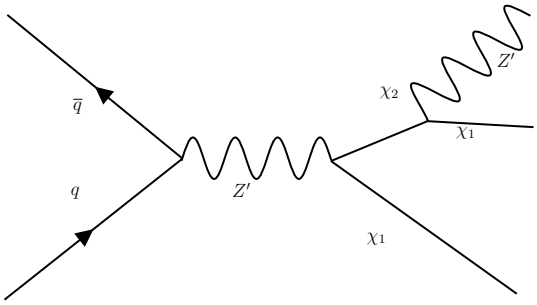


Figure 3.2: Z' Light Vector Model

It is assumed that the mass splitting of the two states, $\chi_{1,2}$, is larger than $m_{Z'}$, such that the heavier state, χ_2 can decay into an on-shell Z' and a stable DM candidate χ_1. The $\chi_{1,2}$ interaction with the Z' is off-diagonal, and taking $M_m > M_d$ we can write the interaction

$$\frac{g_\chi}{2} Z'_\mu(\overline{\chi}_2\gamma^\mu\gamma^5\chi_1 + \overline{\chi}_1\gamma^\mu\gamma^5\chi_2) \tag{3.4}$$

3.1.3 Z' Effective Field Theory

The aforementioned models rely on the Z' coupling to the quarks as seen in Eq. (3.1). In the third model, rather than producing the DM candidates through the new Z', we consider the possibility that the dark states χ_1 and χ_2 are produced through a new constant interaction:

$$\frac{1}{2\Lambda^2}\overline{q}\gamma_\mu q(\overline{\chi}_2\gamma^\mu\gamma^5\chi_1 + \overline{\chi}_1\gamma^\mu\gamma^5\chi_2), \tag{3.5}$$

where Λ is the scale at which new physics may appear. Similarly to the Light Vector model we assume the two dark states $\chi_{1,2}$ with an off-diagonal coupling to the Z', however, the intermediate s-channel Z' in Figure 3.2 has effectively been replaced with a heavy Z'_H, which has been integrated out to give the operator in Eq. (3.5). This results in the process of in Figure 3.3.

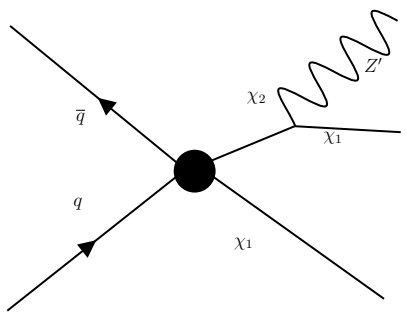


Figure 3.3: Z' Effective Field Theory Model

3.2 Supersymmetry

Supersymmetry, or SUSY for short, is another SM extension that introduces a new symmetry between fermions and bosons. A supersymmetrical operator, Q, changes a particle from a fermionic state, $|F\rangle$, to a bosonic state, $|B\rangle$, and vice versa by changing the spin by 1/2 unit

$$Q\,|F\rangle = |B\rangle\,, \quad \text{and} \quad Q\,|B\rangle = |F\rangle$$

This new symmetry has as a consequence that for each fermion degree of freedom in the SM, there must be a boson degree of freedom and vice versa. What this means is that every SM particle has a superpartner, such that SM particles are grouped with their supersymmetric counterpart into supermultiplets according to their quantum numbers.[1]

This extension addresses many of the problems in the SM, but for the purposes of this thesis we will only mention the *WIMP miracle* [6]. To give a brief description of the WIMP miracle, if we put in the dark matter density that the universe requires today, we can infer how many WIMPs[2] we need of a given mass to make it up. With this we can compute what the self-annihilation cross-section must be in order to get the right abundance of dark matter in the universe. That value turns out to be in the range of the electroweak scale. For SUSY the DM mass scale is in the ballpark of 100 GeV to 1 TeV, suggesting that the self-annihilation of SUSY WIMPs in the early universe could have naturally resulted in the observed dark matter abundance we see today. During the cooling of the universe, the self-annihilation of WIMPs eventually stopped, known as freeze-out. This freeze-out process led to a constant population of WIMPs, preserving a certain relic density. This relic density is suggested to be what we observe today as dark matter, indicating the importance of freeze-out in determining the abundance of dark matter particles in the universe.

The neutralino, $\tilde{\chi}_1^0$, is a leading candidate for dark matter because it naturally emerges as the lightest supersymmetric particle and is in the SUSY models studied in this thesis assumed to be typically stable. As a WIMP, the neutralino possesses the appropriate properties to be a dark matter candidate, interacting weakly with other particles and

[1] The supersymmetric operator, Q, does not change any of the quantum numbers except for the spin
[2] Assuming they interact only through the weak force and gravity

having a mass scale consistent with the observed dark matter density in the universe. Because of this we will look at a model which has neutralinos in the final state. The model we will study, which has been studied by the ATLAS collaboration [16], is the *direct slepton production* channel as this has a dilepton and MET final state, which this thesis is based on.

For this thesis we will study a process involving superpartners of the leptons, the sleptons $\tilde{\ell}$, as illustrated in the Feynman diagram in Figure 3.4.

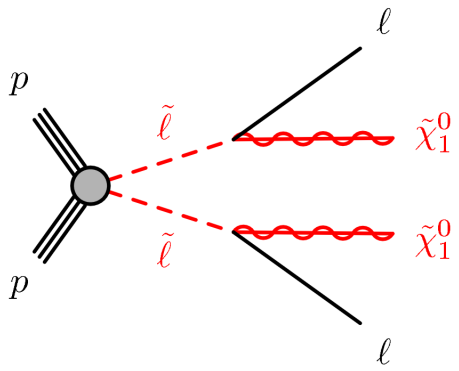


Figure 3.4: Supersymmetric direct slepton production model

In this model we study sleptons produced in the $pp-$collisions at the LHC, which then decay to their SM partner, the leptons, in addition to the DM candidate the neutralino $\tilde{\chi}_1^0$. In this thesis only the 1st and 2nd generation [3] are included in the models, which are assumed to be mass degenerate. The staus in the models are assumed to be heavier.

[3] The superymmetric partners of the electron and muons, the selectron and smuons

3.3 Two Higgs doublet model

The Two Higgs Doublet Model, or 2HDM for short, is an extension of the SM that introduces an additional Higgs doublet[4] [17]. This model is motivated by various theoretical considerations and provides a framework to study the properties of the Higgs boson and explore new physics phenomena, including implications for DM searches.

The 2HDM includes two Higgs doublets instead of one, resulting in an expanded scalar sector. This means there are additional Higgs bosons, which include neutral and charged ones. The presence of these extra Higgs bosons opens up new channels for DM interactions and allows for the possibility of DM particles being connected to the Higgs sector.

The 2HDM offers different scenarios, In this thesis we will study the scenario of having a charged Higgs doublet with an additional pseudoscalar a, which couples to DM.

The model we will study in this thesis is one version of the 2HDM with the addition of a pseudoscalar mediator a which mediates the interactions between the visible and dark sectors.

We will study the channel depicted in Figure 3.5, which includes a charged Higgs, H^-.

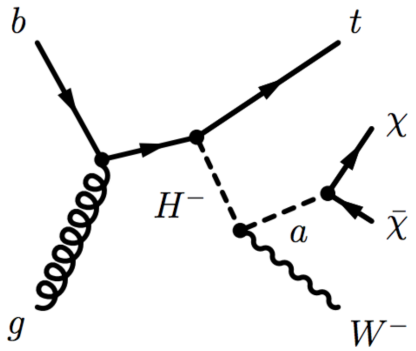


Figure 3.5: Two Higgs Doublet Model + pseudoscalar decay channel with a dilepton and MET final state stemming from $t \to W^+b \to l^+\nu_l b$ and $W^- \to l^-\overline{\nu}_l$

We study this specific channel because of the potential dilepton final state that can come from the $W^+ \to l^+\nu_l$ coming from the top-decay and the other W^- boson decaying to $l^-\overline{\nu}_l$ (See Section 6.1), in addition to the MET from the DM candidates $\chi\chi$ and the

[4]Meaning a new Eq. (2.10)

neutrinos from the W decays. An important value we will study is the ratio between the vacuum expectation values of the two Higgs doublets, written as $\tan\beta$. This model has been studied by the ATLAS collaboration [18], and we will from now on refer to it as the $2HDM + a$ model.

3.4 Summary

To summarize the models we will be studying in thesis. We have three models that stem from a $U(1)'$ extension of the SM, this $U(1)'$ extension predicts a new vector boson Z' which can interact with DM candidates. The first of the three mono-Z' models we study is a Dark Higgs model, where the Z' couples to a new scalar particle behaving like the Higgs, h_D. The second model assumes that the Z' is a Light Vector, which interacts directly with heavy dark states χ_1, χ_2, χ_2 finally decaying to a real Z' in addition to the DM candidate χ_1. The third model is an Effective Field Theory version of the last model, with unknown production inetaction, expected to show up at some high energy scale Λ. Where we assume that the Z' is lighter than the dark state χ_2.

The fourth model we will study is a Supersymmetric direct slepton model, where the sleptons, the scalar superpartners of the SM leptons, decay into leptons and neutralinos, $\tilde{\ell} \to \ell\tilde{\chi}_1^0$, where $\tilde{\chi}_1^0$ is assumed to be the DM candidate. The last model we will study in this thesis is a Two Higgs Doublet Model with an additional pseudoscalar a that decays into DM candidates, $\chi\chi$.

Now that we have presented the theoretical framework behind the SM and the DM models, we have essentially showcased our signal and background for the ML search. But as only having the theoretical framework is not enough to make a dataset to use in ML, we have to show how we can obtain experimental measurements to confront these theories. This is the subject of the next chapter, which presents how to get kinematical variables from Lorentz vectors, and how one identifies real particles in detectors. We will also present the cut and count method used to discover new physics as well as the statistical tools we will apply to our searches.

Chapter 4

Production, Detection and Analysis

Now that we have established the necessary theoretical groundwork of particle physics in Chapter 2 and explored our DM candidates in 3, it is time to explore how this knowledge can be applied. This leads us to ask important questions such as, how can we measure what we have learned? How do we put it into practice? Most importantly, how can we use this knowledge to uncover new physics phenomena?

To answer these questions, we have divided this chapter into three sections, each of which will focus on a different aspect of experimental particle physics. The first section will delve into particle production, followed by an examination of particle detection with the ATLAS detector at the LHC, and finally, we will explore the intricacies of data analysis in particle physics.

4.1 Particle production

Having already been introduced to the SM we are now ready to dive into the subject of how we can produce the particles that we wish to detect. In this chapter we will start from the basic kinematics of particles and then move to more complex variables that will be of use when analyzing data from detectors. The material for the first section is based on Thomson's book Modern Particle Physics [8], Jacksons' "Kinematics" [19] and Vadla's PhD. thesis [20].

4.1.1 Particle kinematics

When working on a relativistic setting, such as we do in high energy particle physics, four vectors are the natural object to consider to generally describe our particles. As we are mainly interested in the motion of the particles, we will look at the four-momentum. Instead of using general variables, we will describe the particles using the four-momentum in terms of the geometry of the detectors (See Section 4.2), that means we will use the polar angle, θ, and the azimuthal angle, ϕ, such that we have

$$p^\mu = (E, p_x, p_y, p_z) \xrightarrow{Lab} (E, p_T \cos\phi, p_T \sin\phi, |\mathbf{p}| \cos\theta) \tag{4.1}$$

where p_T is the *transverse momentum* expressed as

$$p_T \equiv \sqrt{p_x^2 + p_y^2} = |\mathbf{p}| \sin\theta \tag{4.2}$$

Where the relativistic energy and momentum are given as, $E = \gamma m$ and $\mathbf{p} = \gamma m \boldsymbol{\beta}$, with $\gamma = 1/\sqrt{1-\beta^2}$ and $\boldsymbol{\beta} = \mathbf{v}/c$[1], m is the mass of the particle and c is the speed of light in vacuum. By contracting[2] two four-momenta we get the important Lorentz invariant square of the *invariant mass*

$$m^2 = p_\mu p^\mu = E^2 - |\mathbf{p}|^2$$

[1] As this is a particle physics thesis I will convert to Natural Units where we set $c = 1$, $\hbar = 1$

[2] Using the particle physicists convention of the Minkowski metric tensor $\eta_{\mu\nu}$, (+, -, -, -)

which can be generalized for a system containing n particles as

$$m^2 = p_\mu p^\mu = \left(\sum_{i=1}^{n} E_i\right)^2 - \left(\sum_{i=1}^{n} \mathbf{p}_i\right)^2 \tag{4.3}$$

As this thesis will focus on a dilepton (and missing transverse energy) final state, which is of the type $2 \to 4$ where two are invisible, but the invariant mass of the two visible leptons in the final state will be of interest, we will denote this as m_{ll}. From this we can also get another interesting variable, the *transverse energy*. This follows directly from Eq. (4.3) by using the transverse momentum

$$E_T = \sqrt{m^2 + p_T^2} \tag{4.4}$$

The invariant mass is what we measure in the final state only. As we analyze data[3] from Run II on LHC, which operated at $\sqrt{s} = 13$ TeV, it is important to consider the determination of the center-of-mass energy, which characterizes the initial state of the colliding protons. While the LHC controls the initial state, it is crucial to understand that the 13 TeV energy value associated with the incoming protons does not directly define the center-of-mass energy of the initial state. Instead, it is the partons, primarily gluons but also including quarks and anti-quarks, that contribute to the center-of-mass energy, see Section 4.1.2 for more details.

The aim of this thesis is to search for DM, which we know interacts weakly with matter in the "same way" as neutrinos, meaning it leaves no signal in detectors such as ATLAS[4]. So how can we detect its presence? As we know that the transverse momenta before a particle collision is zero[5], then we know that the sum of transverse momenta must be zero after the collision as well, to conserve momentum. From this we often can infer the presence of the non-interacting particles from the presence of *missing transverse energy*[6]

[3] And mostly simulations mimicking the ATLAS detector

[4] Neutrinos interact in dense and large mediums however

[5] Because protons travel along the beam axis

[6] Also called *missing momentum*. While energy is a scalar quantity without direction, we use the term "missing transverse energy" to emphasize the magnitude of the missing transverse momentum vector. This convention is adopted due to the nature of our energy measurements in calorimeters, which actually give us directions

(MET), which is defined by

$$E_T^{miss} = \left|\mathbf{p}_T^{miss}\right|, \quad \text{where} \quad \mathbf{p}_T^{miss} \equiv -\sum_i \mathbf{p}_{T,i} \tag{4.5}$$

where the sum extends over the measured momenta of all the observed particles in an event, as well as all the low energy tracks which are reconstructed, but not associated with any particle[7]. From this formula, if all particles produced in the collision have been detected, then this sum should be zero. Meaning that in ideal cases, significant MET is an indicative of the presence of undetected particles.

Another useful kinematic variable is the *hadronic activity* which is the scalar sum of the transverse momentum of all jets in an event, defined as

$$H_T = \sum_{i \in \{jets\}} ||\mathbf{p}_{T,i}|| \tag{4.6}$$

This gives a measurement of the hadronic energy scale of an event. Another handy trick comes from the realization that the centre-of-mass frame is between the hadrons in jets, where the total momentum is given as a function of the energy of the hadrons. This means that the final state particles are boosted along the beam axis. With this realization we can now introduce a Lorentz invariant[8] kinematic property known as the *rapidity, y* used to express the polar angles

$$y \equiv \frac{1}{2} \ln \left(\frac{E + p_z}{E - p_z} \right) \tag{4.7}$$

We can use that $p_Z = E \cos\theta$ in the high-energy limit as the mass is negligible. In this limit we can use the *pseudorapidity*, η, defined by

$$y \approx \eta \equiv -\ln \left(\tan \frac{\theta}{2} \right) \tag{4.8}$$

The pseudorapidity is an interesting variable as, in the same was as θ, it can tell us how close to the beam the final state particles are[9], but this time independently of the boost as it does not include the mass of the particle. The pseudorapidity also gives us that for a

[7] Referred to as *soft terms*
[8] Under boosts along the beam axis
[9] Where the higher $|\eta|$ means closer to the beam

single particle the phase space is more uniformly distributed on (η, ϕ) than $(\theta, \phi)-$space. The η variable can also be negative, meaning it is boosted along the other beam direction. From this we can define a new variable which will come handy with particle identification, which is called the *R-cone*, that defines a circle in (η, ϕ)-space. It is defined as

$$\Delta R = \sqrt{(\Delta\eta)^2 + (\Delta\phi)^2} \tag{4.9}$$

Another interesting variable is the *transverse mass*, defined as

$$m_T^2 = m^2 + p_T^2 \tag{4.10}$$

where m^2 is the invariant mass defined in Eq. (4.3). What is interesting about this variable is that it is a generalization of the invariant mass in the transverse plane and can thus accommodate the MET coming from invisible particles. We can take this further by looking at a variable which calculates a transverse mass for two leptons by distributing the total p_T^{miss} among the two systems, and minimizing the maximum of the two transverse masses by varying the distribution of the $p_T^{miss}-$vector in terms of the size of p_T. This is called the *stransverse mass* and is defined by

$$m_{T2}^2(\chi) = \min_{\not{\mathbf{q}}_T^{(1)} + \not{\mathbf{q}}_T^{(2)} = \not{\mathbf{p}}_T} \left[\max \left\{ m_T^2 \left(\mathbf{p}_T^{\ell_1}, \not{\mathbf{q}}_T^{(1)}; \chi \right), m_T^2 \left(\mathbf{p}_T^{\ell_2}, \not{\mathbf{q}}_T^{(2)}; \chi \right) \right\} \right] \tag{4.11}$$

where $\not{\mathbf{q}}_T$ are "dummy 2-vectors", χ is a free parameter used to "guess" the mass of the invisible particle, and $m_T^2(\mathbf{p}_T, \mathbf{q}_T)$ is an application of Eq. (4.10) using two particles:

$$m_T^2(\mathbf{p}_T, \mathbf{q}_T) = 2(p_T q_T - \mathbf{p}_T \cdot \mathbf{q}_T)$$

For a more detailed explanation and interpretation of the stransverse mass we refer the reader to the paper by Barr et al. [21]. Even though the stransverse mass was made with neutralinos in mind, it can still be used to calculate SM processes. For example, if we want to reduce W^+W^- background events, we can first recall that each boson can decay as $W^+ \to l^+ + \nu_l$ (and $W^- \to l^- + \overline{\nu}_l$) with the W mass, m_W, as an endpoint. Meaning that we can use m_{T2} to reduce the W^+W^- events in a dilepton final state by requiring that $m_{T2} > m_W$.

4.1.2 Proton-proton collisions

With all the kinematics out of the way the question of how the particles are produced still remains. The answer could be an electron-positron collider, as they did in LEP, a proton anti-proton collider, like the Sp$\overline{\text{p}}$S. As this thesis uses LHC data we will look at proton-proton collisions. Protons are made of elementary particles, two *up*- and one *down*-quarks to be specific. Because of this it is not hard to realize that the Feynman rules acquired from the SM (Figure 2.1) also apply here. In this subsection we will study how different effects of $pp-$collisions affect the cross-section, and therefore the expected number of events to occur, from a kinematical point of view.

Parton Distribution Functions

Although the proton is made up of two up quarks and one down quark, called the *valence quarks*, the proton also consists of gluons and other quarks and anti-quarks, called *sea quarks*. These sea quarks become important in deep inelastic scattering, where the proton breaks apart due the high energies in the collisions. As we accelerate the protons before colliding them, we also accelerate the quarks and "gluons" inside it, each of the partons carry a momentum fraction x of the proton, referred to as the Bjorken x. We can then calculate the reduced centre-of-mass, $\hat{s}$ of two colliding partons q_1 and q_2, as seen in Figure 4.1, with the proton momentum fraction x_1 and x_2, from the proton's momentum p_1 and p_2 respectively as

$$\hat{s} = x_1 x_2 s$$

where s is the centre-of-mass energy squared of the $pp-$system. The valence quarks in the proton do not only interact with the other valence quarks in the other proton, but they might also emit gluons which split into quark anti-quarks pairs or gluon-gluon, making a "sea" of gluons, quarks and anti-quarks around the valence quarks. The momentum of the partons inside the proton are dependent on the momentum transfer Q^2 and is represented by an experimentally determined momentum distribution, known as the *parton distribution function* (PDF) [22] $f(x, Q^2)$. In other words, the PDFs give the probability of a parton to carry the momentum fraction x of a proton. The shape and form the PDFs play an important role in estimating the process cross-section that occur after the $pp-$collisions, and therefore are crucial when simulating events using Monte Carlo [23]. If

we take as an example the process $pp \to l^+l^- + X$ where X denotes any hadrons formed by the remaining quarks. Figure 4.1 showcases the process.

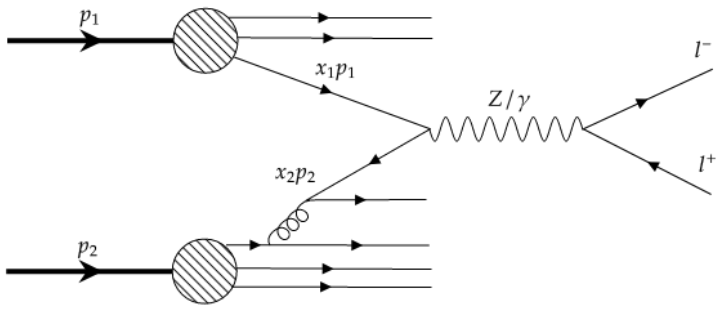


Figure 4.1: Feynman diagram depicting the $pp \to l^+l^- + X$ process where X denotes any hadrons formed by partons not taking part in the hard process $q\bar{q} \to l^+l^-$. The colliding valence quarks are defined as $q_i = x_i p_i$ where each quark q_i is carrying the momentum fraction x_i of the proton momentum p_i.

The cross-section for the process is

$$\sigma\left(p(p_1)p(p_2) \to l^+l^- + X\right) =$$

$$\int_0^1 \int_0^1 \sum_f f_f(x_1) f_{\bar{f}}(x_2) \cdot \sigma\left(q_f(x_1p_1) + \bar{q}_f(x_2p_2) \to l^+l^-\right) dx_2 dx_1 \tag{4.12}$$

where $\sigma\left(q_f(x_1p_1) + \bar{q}_f(x_2p_2) \to l^+l^-\right)$ is calculated using the Feynman rules given in Figure 2.1, and f are the PDFs.

Breit-Wigner resonance

Another important aspect when looking at particle collisions is the *Breit-Wigner resonance*. All unstable particles have a decay rate or width, Γ, (given as inverse of the lifetime τ) which is present in the wave function, $\Psi \propto e^{-i(m-i\Gamma/2)}$. This decay rate also becomes apparent when the unstable particle is the propagator of the interaction we are studying

$$\sigma \propto \frac{1}{(s-m^2)^2 - m^2\Gamma^2} \tag{4.13}$$

From this we can see that, as the square of the centre-of-mass energy s, approaches the unstable particles mass m, there will be a resonance at the invariant mass of the final state particles that "show" the mass of the unstable particle, this is called the Breit-Wigner resonance. It is because of resonances like this that we can identify particles such as the

Z boson. A new resonance Z' resonance is predicted by some BSM DM models studied in this thesis.

Expected events

The most important value we need to know when studying $pp-$collisions is the number of events N expected for a process, this is defined as

$$N = \sigma \int \mathcal{L}(t)dt, \qquad \text{where} \quad \mathcal{L} = f\frac{n_1 n_2}{\sigma_x \sigma_y} \tag{4.14}$$

where σ is the cross-section of the process as expressed in Eq (2.18) while also taking into account the PDF functions in Eq. (4.12) and the decay widht in Eq. (4.13). The last three symbols come from accelerator kinematics where $\sigma_{x,y}$ denotes the beam size, f is the frequency of bunch crossings, and $n_{1,2}$ is the number of protons in bunches.

4.2 The ATLAS detector

We have so far in this chapter discussed how particles are produced. But we have not yet explained how we actually detect them, arguably the most important matter in the field of experimental high energy particle physics. This section of the chapter is just about that, and we will explain how the detection happens in A Toroidal LHC ApparatuS, or more commonly know as the ATLAS detector. Figure 4.2 showcases the detector and its size. The information of this section is largely based on the original ATLAS Technical Design Report [24].

The ATLAS detector is a general multipurpose[10] detector located at the LHC and covers nearly the entire solid angle around the collision point, as described in the reference frame depicted in Figure 4.2. The ATLAS detector consists of four main subdetectors; (i) an inner tracking detector (ID), an (ii) electromagnetic calorimeter (ECAL), a (iii) hadronic calorimeter (HCAL), and lastly (iv) a muon spectrometer (MS). Figure 4.3 visualizes in the transverse plane the four $(i) - (iv)$ main sub-detectors, along with how the different particle types interact with each layer. A brief description of each layer is given in this section.

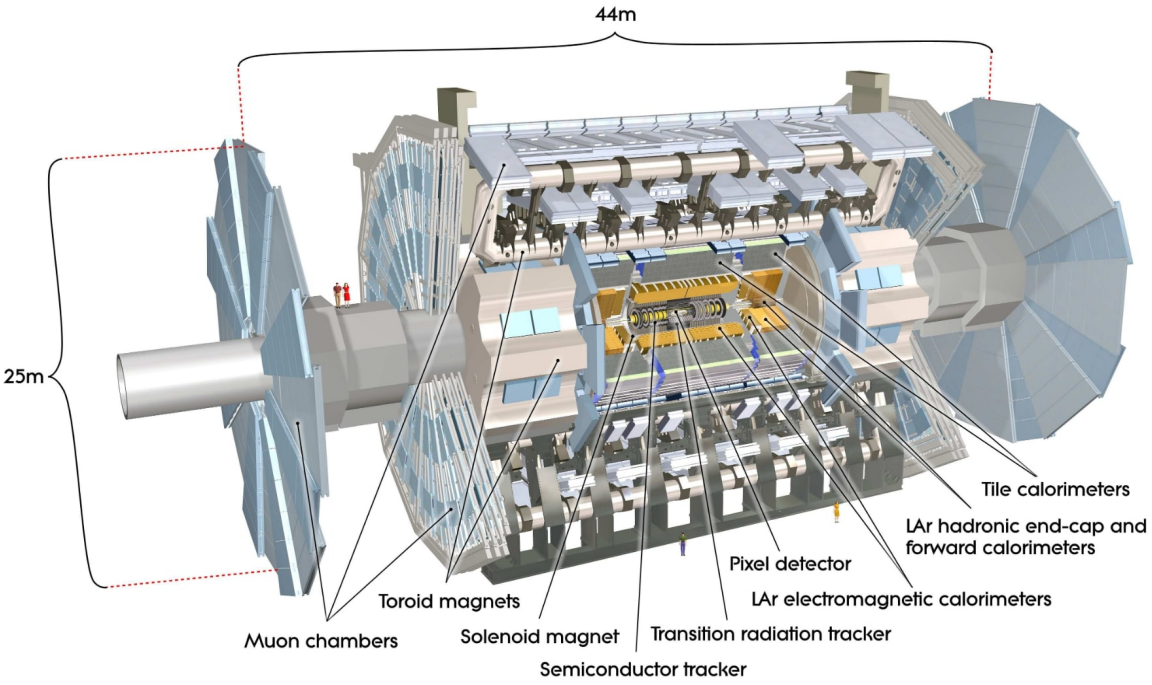


Figure 4.2: Cut-away view of the ATLAS detector, image taken from Ref. [24]

[10] Probing $pp-$ and $AA-$ (heavy ions) collisions.

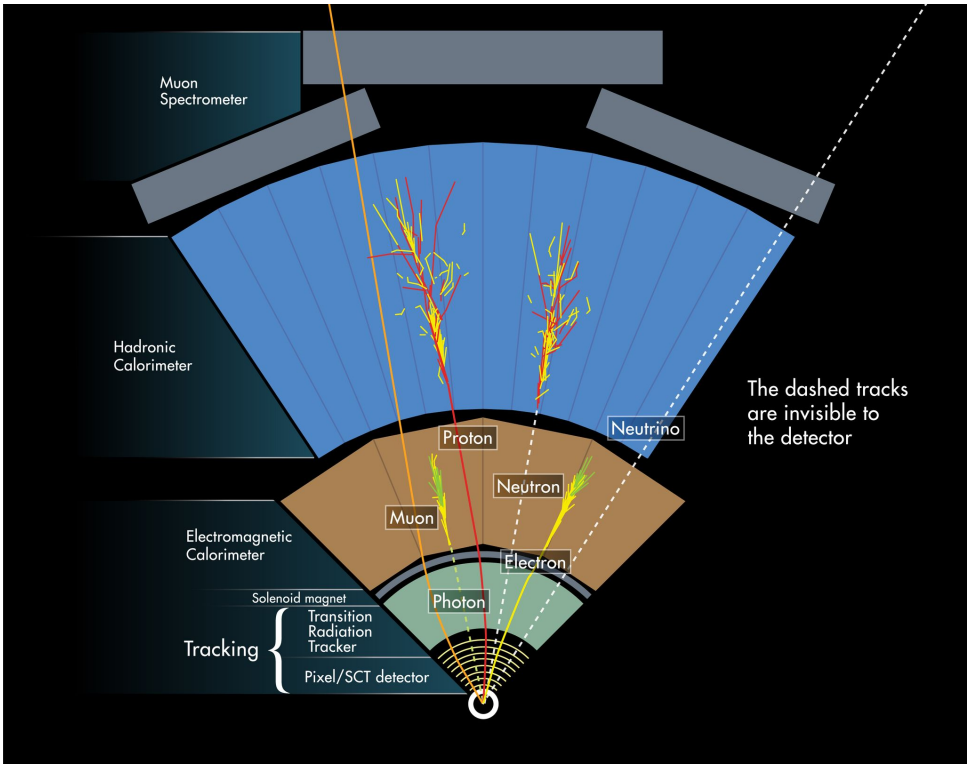


Figure 4.3: Illustration of the ATLAS detector layers, image taken from Ref. [25]

4.2.1 Inner detector

The inner-detector (ID) system is immersed in a 2T axial magnetic field and provides charged-particle tracking in the range $|\eta| < 2.5$. The ID provides the first measurements of the momentum and charge of electrically charged particles, as these can be determined by the curvature of their reconstructed tracks. The ID is made of three independent systems; the pixel detector, the semiconductor tracker (SCT) and the transition radiation tracker (TRT).

The pixel detector is made up of 80 million silicon pixel sensors, each of size $50 \times 400 \mu m^2$, and spread over four layers. Outside the pixel layers are the SCT, which consists of silicon microstrip detectors, also placed on multiple layers. The SCT covers the pseudorapidity region $|\eta| < 2.5$. Lastly we have the TRT, which consist of 4 mm in diameter straw tubes, which enable track-following up to $|\eta| = 2.0$ and allows for electron identification.

4.2.2 The calorimeters

The ATLAS detector has two types of calorimeters, the *electromagnetic calorimeter* (ECAL) and the *hadronic calorimeter* (HCAL), both designed to fully stop certain types of particles. The ECAL is immediately surrounding the inner detector and is divided into a barrel part ($|\eta| < 1.475$) and two end-cap components ($1.375 < |\eta| < 3.2$). The ECAL consists of absorbing lead plates, with liquid Argon (LAr) in between. The thickness of the calorimeter is made to fully measure the shower of photons and electrons/positrons. The muons will only lose a small fraction of their energy as they have longer interaction lengths in lead.

The HCAL is immediately surrounding the ECAL on all sides and consists of two types of detectors. In the barrel ($|\eta| < 1.0$) and the extended barrel regions ($0.8 < |\eta| < 1.7$), the HCAL is made of steel plates with plastic scintillator tiles as active material. While on the end-cap regions ($1.5 < |\eta| < 3.2$) there are hadronic LAr detectors, with absorbing copper plates as active material; in the forward region ($3.1 < |\eta| < 4.9$) a combination of copper and tungsten plates are used as active material. The active materials are chosen to maximize the interaction cross-section with hadrons, such as neutrons, protons and charged pions. The depth of the HCALs is also designed to fully stop hadrons and their showers, which corresponds to the deposited energies. Hadrons are efficiently stopped at the HCALs, meaning that only muons and invisible particles, such as neutrinos and potentially dark matter, leave the HCAL.

4.2.3 Muon spectrometer

The outermost layer of the ATLAS detector is the *muon spectrometer* (MS), dedicated to the measurement and identification of the muons momenta. The MS consists of multiple layers of detector material, and is immersed in a strong magnetic field to bend the trajectories of the charged muons. The MS is made of four different types of detector component: (*i*) Monitored Drift Tubes (MDTs) in the barrel, (*ii*) Cathode Strip Chambers (CSCs) dealing with the events closer to the beam line in the end cap, (*iii*) Resistive Plate Chambers (RPCs) in the barrel and (*iv*) Thin Gap Chambers (TGCs) in the end caps. The MDTs and CSCs are used for tracking while the RPCs and TGCs are used for

triggering. The tracking is provided for pseudorapidities up to $|\eta| < 2.7$, and the trigger system only extends to $|\eta| < 2.5$.

4.3 Data analysis

The time has come to explore how we can search for new physics phenomena, now that it has been established how particles are produced and how we detect them. In this section we will take into account the classic way of searching for new physics which is called the *cut and count method*, but there are other methods to search for new physics, such as Machine Learning (ML) which is the method pursued in this thesis. There might be other methods, such as Quantum ML, but as of today we are still in a too early stage of the technology [26]. To give a short description of the cut and count method, it makes kinematical *cuts* on various variables to isolate signal from background. The way the signal and background[11] are made is by Monte Carlo (MC) simulations, which is necessary for guiding us on where to place efficient cuts and understand our sensitivity to a given BSM model. As it would not be a real experimental physics discovery without making a statistical analysis of the results we will also explain how we utilize this tool. To guide us through this process I will use $ZZ^{(*)}$ channel in the discovery of the Higgs Boson in 2012 [1] as an example of the success of this method.

4.3.1 Cut and count method

The cut and count method is what has been the standard method of doing data analysis with LHC data. As the name implies, the cut and count method works by making cuts on kinematical variables and afterwards counting how many events are left. The goal of using this method is to make cuts such that we remove as many background events as possible while also keeping as many signal events as possible. For example, if we were to study a new physics model with a new light vector boson behaving similarly to the Z boson, but with a higher mass, then a good kinematical cut to remove many background processes would be to require that $m_{ll} > 100$ GeV, as this would remove the majority of Z-resonance from the final state, making it easier to "find" the new physics model.

[11] There are also data driven methods, but on this thesis we will focus on simulations

To more thoroughly explore the cut and count method we can look at the Higgs discovery, in particular the $H \rightarrow ZZ^{(*)} \rightarrow 4l$[12] channel. The event selection consist of kinematical cuts used. The kinematical cuts include (aside from the *acceptane cuts*), were:

- Single-lepton or dilepton triggers
- Four leptons final state with ordered $p_T > 20, 15, 10, 7$ GeV
- Higgs-boson candidates are formed by selecting two same flavor opposite charge lepton pairs

The first "cut" is to make sure that the event contains leptons with sufficient quality to have fired one of the relevant triggers. The second cut is to have the sufficient number of leptons in the final state with p_T above the trigger thresholds. And lastly we want to have two lepton pairs of the same-flavor with opposite-charge, this is to make sure that one lepton pair actually decays from a Z-boson. What now remains is to explore the "count" part of this method. When counting the events that pass the event selection one usually counts the background events that pass, the data points that pass, and also the signal events that pass. For the 2012 Higgs discovery, the Higgs channel with a Higgs mass of 125 GeV was used as signal. The results of the 2012 discovery is shown in Figure 4.4.

Although this section made the process look simple, it was through the effort of many scientists working together that made this happen. The great computational power needed to *correctly* simulate events and reconstruct objects from detector signal was, and still is a big challenge. Not to mention the state-of-the-art technology to be able to both accelerate the protons to an energy high enough to "create" new physics, and to actually be able to detect it. This alone was not enough to claim the discovery, to claim anything we need to look at statistics, which is the subject of the next section.

4.3.2 Statistical tools

To make any sort of claim in modern physics we should be absolutely certain that what we are claiming is true, as just making the cuts and isolating a signal to background is

[12]Where l is for lepton, but only means $e^{\pm}$ or $\mu^{\pm}$

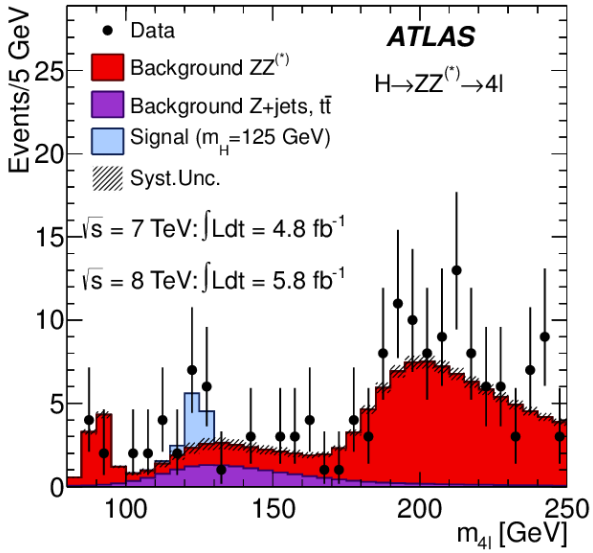


Figure 4.4: The Higgs discovery on the $ZZ^{(*)}$ channel, image taken from Ref. [1]

not enough. To be specific we need to be "*at least* 5σ *sure*" to claim any new discoveries. But as not everything in experimental high energy particle physics is a new discovery, we will also look at *exclusions*. In this section we will present some mathematical details behind the statistical tools that will be used in this thesis.

Fix this intro

This thesis follows the example set by the ATLAS Collaboration article "Search for new particles in events with one lepton and missing transverse momentum in pp collisions at $\sqrt{s} = 8$ TeV with the ATLAS detector" [27], where a Bayesian analysis is performed to set limits on the existence of a new W' boson. Using the signal+background hypothesis, the expected number of events in each lepton channel, $ee, \mu\mu$, of the process $W^- \to l\nu$ is

$$N_{\text{exp}} = \varepsilon_{\text{sig}} L_{\text{int}} \sigma B + N_{\text{bkg}}$$

where L_{int} is the integrated luminosity, ε_{sig} is the signal selection efficiency defined as the fraction of signal events that satisfy the event selection criteria, N_{bkg} is the expected number of background events, and σB is the cross-section times branching ratio of the process. Using Poisson statistics, the likelihood to observe N_{obs} events is

$$\mathcal{L}(N_{\text{obs}}|\sigma B) = \frac{(N_{\text{exp}})^{N_{\text{obs}}} e^{-N_{\text{exp}}}}{N_{\text{obs}}!} \tag{4.15}$$

We include uncertainties by introducing nuisance parameters θ_i, each with a probability density function $g_i(\theta_i)$, and integrating the product of the Poisson likelihood with the probability density function. The integrated likelihood is

$$\mathcal{L}_B(N_{\text{obs}}|\sigma B) = \int \mathcal{L}(N_{\text{obs}}|\sigma B) \prod g_i(\theta_i) d\theta_i \tag{4.16}$$

where a log-normal distribution is used for the $g_i(\theta_i)$. The nuisance parameters are taken to be: $L_{\text{int}}, \varepsilon_{\text{sig}}$ and N_{bkg}. The measurements of the two decay channels (muon or electron final state for $W' \to l\nu$) are combined assuming the same branching fraction for each, thus Eq. (4.16) remains valid with the Poisson likelihood replaced by the product of the Poisson likelihoods for the two channels. The integrated luminosities for the electron and muon channels are fully correlated. We can further use Bayes' theorem which gives the posterior probability that the signal has signal strength σB:

$$P_{\text{post}}(\sigma B|N_{\text{obs}}) = N\mathcal{L}_B(N_{\text{obs}}|\sigma B)P_{\text{prior}}(\sigma B) \tag{4.17}$$

where $P_{\text{prior}}(\sigma B)$ is the assumed prior probability, here chosen to be flat in σB, for $\sigma B > 0$. The constant factor N normalizes the total probability to one. The posterior probability is evaluated for each mass and decay channel as well as for their combination, and then used to set a limit on σB.

As we can see, the inputs for the evaluation of $\mathcal{L}_B$ (and P_{post}) are ε_{sig}, L_{int}, N_{bkg} and N_{obs} and the uncertainties of the first three. The uncertainties for these should account for experimental and theoretical systematic effects as well as the statistics of the simulated samples. For this thesis the systematic uncertainties will not be calculated, but will rather be assumed to be flat and $\pm$ 20% of the background.

To make exclusions we can use Eq. (4.17) to establish a *confidence limit* (CL). CLs are defined as the probability to observe the number of events observed in an experiment, N_{obs}, or *less* given signal+background. We usually define a signal+background hypothesis to be excluded when $\text{CL}_{s+b} < 5\%$. Meaning a 95% CL, such that the probability to falsely exclude an existing signal(+background) is 5%. We will use CL is to *set limits* on theoretical models, rather than exclude them.

On the other side, to claim any discovery in particle physics we need to know the *significance* of any statistical fluctuation. Before getting to the significance we can discuss the *p-value*, defined as the probability to observe the number of events observed in the experiment, n_{obs}, or *more* given only background

$$p = P(N \geq N_{\text{obs}} | \lambda = N_{\text{bkg}}) = \sum_{k=N_{\text{obs}}}^{\infty} \mathcal{L}(k|N_{\text{bkg}}) \tag{4.18}$$

The smaller the $p-$value, the less compatible an observation is with the background only hypothesis, meaning more likely to be a discovery. In Figure 4.5 we can see how the $p-$value could look for an arbitrary distribution

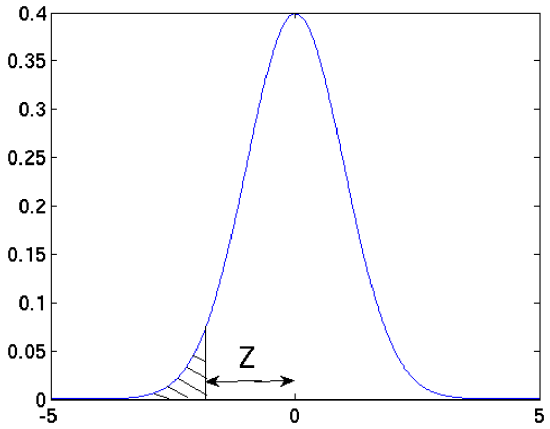


Figure 4.5: Illustration showing how the $p-$value of an arbitrary distribution might look (shaded area). We can also see the relation of the $p-$value to the significance Z in this illusteation.

From this figure we also see the relation between the $p-$value and the significance Z. Mathematically the relation is expressed by

$$p = \int_{-\infty}^{-Z} \frac{e^{-x^2/2}}{\sqrt{2\pi}} dx$$

As mentioned in the start of this subsection, a discovery in particle physics is defined to be at least a $Z = 5\sigma$ deviation from the background hypothesis, meaning that we would have a $p-$value of $p \leq 2.87 \times 10^{-7}$. In other words, with a 5σ deviation, the probability to falsely discover something is at worst one in roughly 3.5 million.

As the significance is an interesting quantity we can give it its own definition. We will

use the low statistics formula for the significance, as this is the most general one. We can either define the significance as the *observed significance* with the equation

$$Z = \sqrt{2\left[N_{\text{obs}} \ln \frac{N_{\text{obs}}}{N_{\text{bkg}}} - N_{\text{obs}} + N_{\text{bkg}}\right]}$$

However, for our purposes, as we will use the signal+background hypothesis, we will use the *expected significance*, which we get by changing $N_{\text{obs}} \to N_{\text{sig}} + N_{\text{bkg}}$, where N_{sig} is the number of signal events

$$Z = \sqrt{2\left[(N_{\text{sig}} + N_{\text{bkg}}) \ln\left(1 + \frac{N_{\text{sig}}}{N_{\text{bkg}}}\right) - N_{\text{sig}}\right]} \tag{4.19}$$

However, as Eq. (4.18) did not include any nuisance parameters, it used Eq. (4.15) instead of Eq. (4.16). We want to express the significance with uncertainties. From "Discovery sensitivity for a counting experiment with background uncertainty" from Glen Cowan [28], we can use then Eq. (17) on his paper that reads

$$Z = \left[-2\left(N_{\text{obs}} \ln\left[\frac{N_{\text{obs}} + m}{(1+\tau)N_{\text{obs}}}\right] + m \ln\left[\frac{\tau(N_{\text{obs}} + m)}{(1+\tau)m}\right]\right)\right]^{1/2}$$

where $m = \tau N_{\text{bkg}}$ and where we have Eq. (19) on his paper that says

$$\tau = \frac{N_{\text{bkg}}}{\sigma^2_{\text{bkg}}}$$

where σ_{bkg} is the uncertainty of the background. Using the prior definitions of m and τ, as well as changing $N_{\text{obs}} \to N_{\text{sig}} + N_{\text{bkg}}$ gives us

$$Z = \sqrt{-2\left((N_{\text{sig}} + N_{\text{bkg}}) \ln\left[\frac{(N_{\text{sig}}+N_{\text{bkg}}) + \frac{N^2_{\text{bkg}}}{\sigma^2_{\text{bkg}}}}{(1+\frac{N_{\text{bkg}}}{\sigma^2_{\text{bkg}}})(N_{\text{sig}}+N_{\text{bkg}})}\right] + \frac{N^2_{\text{bkg}}}{\sigma^2_{\text{bkg}}} \ln\left[\frac{\frac{N_{\text{bkg}}}{\sigma^2_{\text{bkg}}}((N_{\text{sig}}+N_{\text{bkg}}) + \frac{N^2_{\text{bkg}}}{\sigma^2_{\text{bkg}}})}{(1+\frac{N_{\text{bkg}}}{\sigma^2_{\text{bkg}}})\frac{N^2_{\text{bkg}}}{\sigma^2_{\text{bkg}}}}\right]\right)} \tag{4.20}$$

Which makes for a better estimate of the significance one has in reality.

The 95 % CL limit results from the 2012 Higgs discovery [1] can be seen in Figure 4.6, where we can see that there is a statistical fluctuation of the observed data compared to the background with $m_H = 125$ GeV

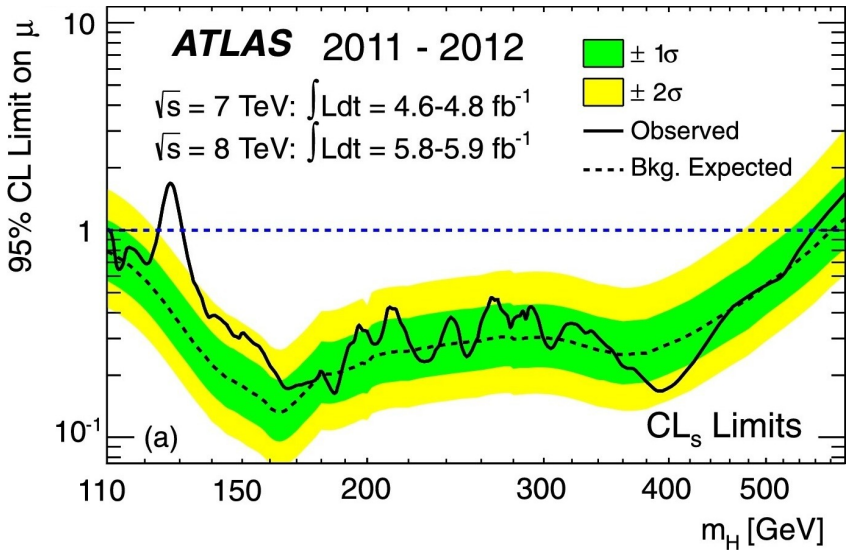


Figure 4.6: Confidence limit on the 2012 Higgs discovery, excerpt taken from Ref. [1]

Question: I show some results from the Higgs discovery, should I rather show the mass exclusion of the W' we saw in FYS555, as this is closer to the mass exclusion of the Z' I studied? Should I also show some SUSY exclusions to show how one excludes using the significance?

4.4 Summary

In this chapter, we have studied how one can discover new physics from calculating the number of expected events from $pp-$collisions. Using special relativity as a tool, we can express the four-momentum of the particles with detector-coordinates, $p^\mu = (E, p_T \cos\phi, p_T \sin\phi, |p| \cos\theta)$. From this four-momentum vector we can thereafter calculate interesting kinematic variables such as the invariant mass m_{ll}, missing transverse energy E_T^{miss}, stransverse mass, m_{T2}, among others. Using the advanced ATLAS detector with the main four parts; *inner-detector* (ID), *electromagnetic calorimeter* (ECAL), *hadronic calorimeter* (HCAL), and the *muon spectrometer* (MS), we can get the four-momentum can be recorded from the accelerated protons at the LHC, see Figure 4.3 which shows visually how different particles interact with the detector.

fix this

With the recorded data and MC simulations taking into account the experimental features such as the PDFs and Breit-Wigner resonance, as well as the ATLAS kinematics, we can compare how the simulated events fare with the data recorded. By playing around with the kinematical variables of the particles and making cuts to isolate new physics signal, we can see if there is a discrepancy between the data recorded and the SM background. After creating this signal region with the cut and count method we conduct a statistical analysis to see how the new theory/observed data deviates from our current understanding of physics.

This state-of-the-art method is what currently is being used at CERN and has lead to a great advancement in the field. However, with the rise of new technologies, such as *machine learning*, which excel at classification tasks, a door has been opened to try new methods. In this thesis we will use ML to hopefully create a better and more general signal region than what the current cut and count method does. Before describing how, we will explain what machine learning is, this is the subject of the next chapter.

Chapter 5

Machine Learning

Machine Learning (ML) has emerged as a powerful tool for analyzing complex datasets and making accurate predictions. Its applications span across various fields, from natural language processing to image recognition, and it has been used successfully to solve a range of problems.

The main approach of this thesis will be to use ML as its popular rise has proven to be effective at binary classification tasks [29, 30] in High Energy Particle Physics. For our purposes it will be a powerful tool to attempt to classify events as SM background or as DM signal.

To give a short description of the essence of ML we can start by considering a general parameter $\boldsymbol{\beta} = \{\beta_1, \beta_2, \cdots, \beta_n\}$ for an n-dimensional problem, which for our purposes can be seen as what are called *weights* and *biases* ($\boldsymbol{\beta} = \{w, b\}$), the goal is to choose these parameters $\boldsymbol{\beta}$ such that we minimize a cost (also called loss) function $C(\boldsymbol{\beta})$ with respect to a set of data points given by a matrix $\mathbf{X}$. This matrix $\mathbf{X}$ will be our dataset containing n *features* for each event m, and is of the following form

$$\mathbf{X} = \begin{pmatrix} x_{11} & x_{21} & x_{31} & \cdots & x_{n1} \\ x_{12} & x_{22} & x_{32} & \cdots & x_{n2} \\ x_{13} & x_{23} & x_{33} & \cdots & x_{n3} \\ \vdots & \vdots & \vdots & \cdots & \vdots \\ x_{1m} & x_{2m} & x_{3m} & \cdots & x_{nm} \end{pmatrix} \tag{5.1}$$

This project will only focus on Supervised Learning, meaning that we know that the output is a binary representation of signal and background, such that we can use target values $\boldsymbol{t}$. Then we give the network a score depending on how close the predicted output is to the real values $\boldsymbol{t}$. Then we repeat the process after tweaking the parameters $\boldsymbol{\beta}$ and see if the score gets better.

To this end, we will first provide a mathematical foundation for ML, starting with a brief overview of the different types of ML algorithms we will study, such as Neural Networks (NN) and Boosted Decision Trees (BDT). Other aspects such as the importance of feature selection and feature engineering in preparing the data for ML algorithms, as well as the concept of model evaluation and optimization will be discussed in Chapter 6 and Chapter 7, respectively.

5.1 Neural Networks

The theoretical foundation for this chapter is mainly based of Hjort-Jensen's lecture notes [31, 32, 33]. Before beginning, we can briefly explain the idea behind NNs. As stated by Hjorth-Jensen in [31]:

> *The idea of NN is to mimic the neural networks in the human brain, which is composed of billions of neurons that communicate with each other by sending electrical signals. Each neuron accumulates its incoming signals, which must exceed an activation threshold to yield an output. If the threshold is not overcome, the neuron remains inactive, i.e. has zero output*

That takes us to what a *neuron* is.

5.1.1 Artificial neurons

To describe the behavior of a neuron mathematically we can use the following model that mimics how one neuron works

$$y = f\left(\sum_{i=1}^{n} w_i x_i\right) = f(z) \tag{5.2}$$

Where y, the output of the neuron, is the value of its *activation function* (See section 5.1.3), which has the weighted, w_i, sum of signals $x_i, \cdots, x_n$ received by n neurons that are in a presiding layer. We will call this weighted sum for z to ease the notation.

The goal of NNs is to mimic the biological nervous system by letting each neuron interact with each other by sending signals, for us, it is in the form of a mathematical function between each layer, called the activation function. Most NNs consist of an input layer, an output layer and intermediate layers, called hidden layers. All the layers can contain an arbitrary number of neurons, and each connection between two neurons is associated with a weight variable w_i. The goal of using NNs is to teach the network the patterns of the data to then predict some target. In the context of our search for DM, by giving a NN our dataset of events as "its input layer", we can then train the network to classify events as signal or background.

Explained in greater detail if we were to look at a single data event, we start with an input with all the relevant features of the event, $\boldsymbol{X}$[1]. Using Eq. (5.2) on every neuron on the next layer we can teach the network if there are any connections between the n features, we can repeat this process for as many *hidden layers* as we want. As an output we want a single neuron to see if it has predicted the event to be a signal or background, since this is binary output. After analyzing the prediction we can use the labels on the target data $\boldsymbol{t}$ to tell (the network) whether it predicted correctly or wrong. We can then use a *cost function* and a specific *metric* to evaluate numerically how well the network predicted the output by giving it a score. Seeing how the results fare we can then back-propagate to shift the weights and biases and repeat the process until we are satisfied with our result. Each of these iterations is called an epoch.

To generalize our artificial neuron to a whole network we can look at a Multilayer Perceptron (MLP). An MLP is a network consisting of at least three layers of neurons, the input, one or more hidden layers, and an output. The number of neurons can vary for each layer. The above explanation is a very dense and simplified one. In reality, it is complicated to find out which cost function, activation function, metric, etc. are best suited to the given problem. But before we get into the gory details we can explore the mathematical model that illustrates what was explained above.

5.1.2 Optimizers

The way we "tweak the parameters $\boldsymbol{\beta}$ to see if the network prediction gets better" is by using an *optimizer*. We will mainly focus on the theory behind the *Stochastic Gradient Descent* (SDG) optimizer as it is easier to digest. Before explaining the SDG we have to look at the Gradient Descent (GD).

Given a cost function $C(\boldsymbol{\beta})$ we can get closer to the minimum by calculating the gradient $\nabla_\beta C(\boldsymbol{\beta})$ wrt. the unknown parameters $\boldsymbol{\beta}$ from the NN. If we were to calculate the gradient at a specific point $\boldsymbol{\beta}_i$ in the parameter space, the negative gradient would correspond to the direction where a small change $d\boldsymbol{\beta}$ in this parameter space would result in the biggest decrease in the cost function, in the same way we in physics would de-

[1] See Eq. (5.1)

termine where the local (or global) minima are in a complex multidimensional potential numerically. In GD, we can choose a step size η (which needs to be optimized) related to the size of an iteration in the parameter space; this is called the *learning rate*. The mathematical function for an iteration in parameter space to optimize the parameter $\boldsymbol{\beta}$ such that it minimizes the cost function is given as

$$\boldsymbol{\beta}_{i+1} = \boldsymbol{\beta}_i - \eta \nabla_\beta C(\boldsymbol{\beta}_i) \tag{5.3}$$

To converge towards a minimum we should select a learning rate η small enough to not "step over" the minimum point of the cost-function-space, but also not too small to get stuck on a local minimum rather than the global minimum. Thus using the learning rate as a hyperparameter in a grid search is a good way to optimize a NN for a given task.

In GD one computes the cost function and its gradient globally for all data points. This quickly becomes computationally heavy when dealing with large datasets. Thus, a common approach is to compute the gradient over batches of the data. For our purposes it would be optimal to use GD, but our data size is massive, of the order of 10^8 events (13 GB), becoming computationally challenging. Thus instead of making an $n \times 10^8$ data matrix (see Eq. (5.1)), we could for example split it into ten smaller matrices of $n \times 10^7$ to then perform a parameter update, making the computation affordable. This is where SGD comes in, for each step, or epoch the data is divided randomly into N batches of size n. Then for each batch we use the cost function minimization, Eq. (5.3), to update the parameters, thus updating $\boldsymbol{\beta}_{i+1}$ N-times for each epoch. The idea of SGD comes from the observation that the cost function can almost always be written as a sum over n data points. The main advantage of SGD is not to ease the computation however, as using more batches also reduces the risk of getting stuck in a local minimum since the SGD introduces a randomness of which part of the parameter space we move through, but this is at the cost of reducing statistics which might not be ideal for every problem.

There are other optimization algorithms we could use, such as the popular ADAM [34]. The Adaptive Moment Estimation (ADAM) is a more advanced optimization algorithm that uses adaptive learning rates. It computes individual learning rates for each weight

based on the average of past gradients and their variances. ADAM also uses momentum[2] to accelerate the convergence of the optimization algorithm. We will test both ADAM and SGD when optimizing our networks.

5.1.3 Activation functions

As seen in Section 5.1.1, an important aspect of NNs are activation functions and cost functions. As shall become apparent in Section 5.1.4, when evaluating an activation function we get the neuron output, but what are these activation functions? Mathematically speaking, activation functions are: Non-constant, Bounded, Monotonically-increasing and continuous functions. In this project we will utilize two different activation functions at different layers. The first one is a sigmoid activation function

$$f(x) = \frac{1}{1 + e^{-x}} \tag{5.4}$$

which is the most basic activation function, this will be used from the input layer to our first hidden layer as the risk of having little neuron activation is minimal here. On all other layers we will utilize a Rectified Linear Unit (ReLU)

$$f(x) = x^+ = \max(0, x) = \begin{cases} x & \text{if } x > 0, \\ 0. & \text{otherwise.} \end{cases} \tag{5.5}$$

ReLU has better gradient propagation, meaning that there are fewer vanishing gradient problems compared to the sigmoid function.

5.1.4 Feed Forward network

To describe how the network "guesses" outputs in a mathematical model we can start by looking at Eq. (5.2) where we got an output y from an activation function f that receives x_i as input. We can expand the function's input as follows

$$y = f\left(\sum_{i=1}^{n} w_i x_i + b_i\right) = f(z) \tag{5.6}$$

[2]Past gradients influence current updates, improving optimization by considering direction and magnitude of previous gradients.

where w_i is still the weight, and we now use the previously mentioned bias b_i which is normally needed in case of zero activation weights or inputs[3]. The difference comes now in the interpretation; in the activation $z = (\sum_{i=1}^{n} w_i x_i + b_i)$ the inputs x_i are now the outputs of the neurons in the preceding layer. Furthermore, an MLP is fully-connected, meaning that each neuron received a weighted sum of the output of **all** neurons in the previous layer. To expand the activation in Eq. (5.6) we can first look at the output of every neuron i in a weighted sum z_i^1 for each input x_j on a layer

$$z_i^1 = \sum_{j=1}^{M} w_{ij}^1 x_j + b_i^1, \tag{5.7}$$

where M stands for all possible inputs to a given neuron i in the *first* layer. Such that if we evaluate the weighted sum in an activation function f_i for each neuron i, then the output of all neurons in layer 1 is y_i^1

$$y_i^1 = f(z_i^1) = f\left(\sum_{j=1}^{M} w_{ij}^1 x_j + b_i^1\right)$$

To generalize this for l-layers, which may have different activation functions, we write it as

$$y_i^l = f^l(z_i^l) = f^l\left(\sum_{j=1}^{N_{l-1}} w_{ij}^l y_j^{l-1} + b_i^l\right) \tag{5.8}$$

Where N_l is the number of neurons in layer l. Thus, when the output of all the nodes in the first hidden layer is computed, the values of the subsequent layer can be calculated and so forth until the output is obtained. With this we can show that we only need the inputs x_n to calculate the output with l hidden layers

$$y^{l+1} = f^{l+1}\left[\sum_{j=1}^{N_l} w_{ij}^{l+1} f^l\left(\sum_{k=1}^{N_{l-1}} w_{jk}^l \left(\cdots f^1\left(\sum_{n=1}^{N_0} w_{mn}^1 x_n + b_m^1\right)\cdots\right) + b_j^l\right) + b_i^{l+1}\right] \tag{5.9}$$

This shows that an MLP is nothing more than an analytic function, specifically a mapping of real-valued vectors $\hat{x} \in \mathbb{R}^n \to \hat{y} \in \mathbb{R}^m$. We can also see that Eq (5.9) is essentially a nested sum of scaled activation functions of the form

$$f(x) = c_1 f(c_2 x + c_3) + c_4$$

[3]The bias allows the model to introduce a constant offset output of a neuron

where the parameters c_i are the weights and biases. By adjusting these parameters we shift the activation function to better match the label we are training the data on, this is the flexibility of a NN. Something else we can note is that Eq. (5.9) can easily be changed into matrix notation, as hinted with Eq. (5.1). However, this realization can help make computing the values a much easier task by for example utilizing `TensorFlow` [35] or other mathematical packages in `Python`. An illustration showing the main idea of how a Feed forward network is set up is shown in Figure 5.1.

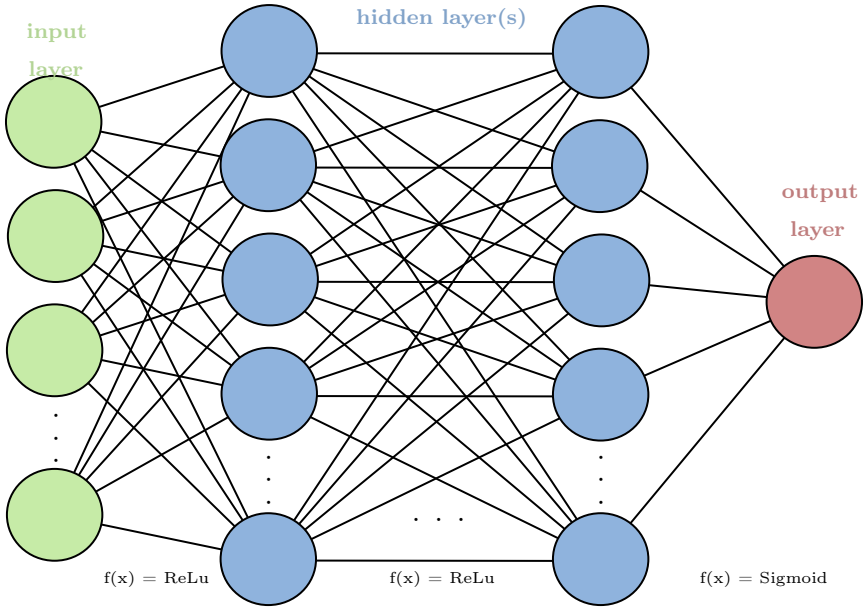


Figure 5.1: Basic illustration of a network with two hidden layers.

5.1.5 Back Propagation algorithm

So far we have only explained Feed Forward networks, which help us to compute the output of the NN in terms of basic linear algebra. We mentioned the possibility to adjust the weight and biases, but never explained how. Now is the time to dive into that subject, as we will explain the back propagation algorithm. What we want to know is how the changes in the biases and weights in the network change the cost function, and how we could use the final output to modify the weights? Before we derive these equations we start by a plain regression problem, using the Mean Squared Error (MSE) as a cost function for pedagogical reasons

$$C(\hat{W}) = \frac{1}{2}\sum_{i=1}^{n}(y_i - t_i)^2 \tag{5.10}$$

where $\hat{W}$ is the matrix containing all the weights, y_i is the network output and (more importantly) t_i are the targets, which are the labels of events telling whether we have a signal or background event. To generalize this we first have to generalize Eq. (5.8) for a layer l

$$z_i^l = \sum_{j=1}^{M} w_{ij}^l y_j^{l-1} + b_i^l \Leftrightarrow \hat{z}^l = \left(\hat{W}^l\right)^T \hat{y}^{l-1} + \hat{b}^l$$

where the right side is written in matrix notation. From the definition of z_j^l with an activation function, Eq. (5.6), we have

$$\frac{\partial z_j^l}{\partial w_{ij}^l} = y_i^{l-1} \tag{5.11}$$

and

$$\frac{\partial z_j^l}{\partial y_i^{l-1}} = w_{ij}^l$$

which again, with the definition of the sigmoid activation function, Eq. (5.4), gives us

$$\frac{\partial y_j^l}{\partial z_j^l} = y_j^l(1 - y_j^l) = f(z_j^l)(1 - f(z_j^l)) \tag{5.12}$$

Furthermore, we need to take the derivative of Eq. (5.10) with respect to the weights, doing so for a respective layer $l = L$ we have

$$\frac{\partial C(\hat{W}^L)}{\partial w_{jk}^L} = \left(y_j^L - t_j\right) \frac{\partial y_j^L}{\partial w_{jk}^L}$$

where the last partial derivative is easily computed using the chain rule with Eq. (5.11) and Eq. (5.12)

$$\frac{\partial y_j^L}{\partial w_{jk}^L} = \frac{\partial y_j^L}{\partial z_j^L} \frac{\partial z_j^l}{\partial w_{jk}^L} = y_j^L(1 - y_j^L) y_k^{L-1}$$

Such that

$$\frac{\partial C(\hat{W}^L)}{\partial w_{jk}^L} = \left(y_j^L - t_j\right) y_j^L(1 - y_j^L) y_k^{L-1} := \delta_j^L y_k^{L-1} \tag{5.13}$$

where we have defined the error

$$\delta_j^L := \left(y_j^L - t_j\right) y_j^L(1 - y_j^L) = f'(z_j^L) \frac{\partial C}{\partial y_j^L} \tag{5.14}$$

or in matrix form

$$\delta^L = f'(\hat{z}^L) \circ \frac{\partial C}{\partial \hat{y}^L}$$

where on the right-hand side we wrote this as a Hadamard product[4]. This error δ^L is an important expression, since as we can see in the index form of this expression in Eq. (5.14), we can measure how fast the cost function is changing as a function of the j-th output activation. This means that if the cost function does not depend on a particular neuron j, then δ_j^L would be small.

We also notice that everything in Eq. (5.14) is easily computed. Thus, we can also see how the weights change the cost function using Eq. (5.13) quite easily. Another thing we can compute with Eq. (5.14) is

$$\delta_j^L = \frac{\partial C}{\partial z_j^L} = \frac{\partial C}{\partial y_j^L}\frac{\partial y_j^L}{\partial z_j^L}$$

which can be interpreted in terms of the biases b_j^L as

$$\delta_j^L = \frac{\partial C}{\partial b_j^L}\frac{\partial b_j^L}{\partial z_j^L} = \frac{\partial C}{\partial b_j^L} \tag{5.15}$$

where the error δ_j^L is exactly equal to the rate of change of the cost function as a function of the bias.

Something interesting is that when using Eq. (5.13 - 5.15) we see that if a neuron output y_j^L is small, then the gradient term, Eq. (5.13), will also be small. We say then that the weight learns slowly, meaning that the contribution of said neuron is less important "to fix" than those that have a higher weight. Of course this example is a very simple one to wrap our heads around, but the magic comes when the algorithm is evaluating a random neuron in a layer n, after using many layers the NN becomes a *black box* for us to wrap our heads around!

It is also worth noting that when the activation function is flat at some specific values[5] the derivative will tend towards zero, making the gradient small, meaning the network

[4] Also called *element-wise* product, $(A \circ B)_{ij} = (A \odot B)_{ij} = (A)_{ij}(B)_{ij}$

[5] For the sigmoid it is at values $|x| > 1$ and for ReLu $x < 0$

is learning slow as well. To finish up our back propagation algorithm we still need one equation. We are now going to propagate backwards in order to determine the weights and biases. We start by representing the error in the layer before the final one $L-1$ in terms of the errors of the output layer. If we try to express Eq. (5.14) in terms of the output layer $l+1$, using the chain rule and summing over all k entries we get

$$\delta_j^l = \sum_k \frac{\partial C}{\partial z_k^{l+1}} \frac{\partial z_k^{l+1}}{\partial z_j^l} = \sum_k \delta_k^{l+1} \frac{\partial z_k^{l+1}}{\partial z_j^l}$$

recalling Eq. (5.7) (replacing 1 with $l+1$) we get

$$\delta_j^l = \sum_k \delta_k^{l+1} w_{kj}^{l+1} f'(z_j^l) \tag{5.16}$$

Which is the final equation we needed to start back propagating.

5.1.6 Summary

To summarize the whole process of the NN

- First take the input data $\mathbf{x}$ and the activation $\mathbf{z}_1$ of the input later, and then compute the activation function $f(z)$ to get the next neuron outputs $\mathbf{y}^1$. Mathematically this is taking the first step of the feed forward algorithm, i.e. choosing $l=0$ in Eq. (5.9)

- Secondly we commit all the way in Eq. (5.9) and compute all $\mathbf{z}_l$, activation function and $\mathbf{y}^l$.

- After that we compute the output error $\boldsymbol{\delta}^L$ by using Eq. (5.14) for all values j.

- Then we back-propagate the error for each $l = L-1, L-2, \cdots, 2$ with Eq. (5.16).

- The last step is then to update the weights and biases using Eq. (5.3) for each l and updating using

$$w_{jk}^l \leftarrow w_{jk}^l - \eta \delta_j^l y_k^{l-1}$$

and

$$b_j^l \leftarrow b_j^l - \eta \delta_j^l$$

This whole procedure is usually called an epoch, which we can repeat as many times as we want to better reduce the cost function in hope of converging to the global minima.

5.2 Boosted Decision Trees

In the previous chapter, we explored how NNs can aid in the task of signal and background classification, such as the one in this thesis where we are looking for a DM signal on top of SM background. In this chapter, we will delve into Boosted Decision Trees (BDTs), another powerful tool for binary classifications. BDTs, unlike neural networks, are not based on simulating biological neurons. Instead, they rely on decision trees, a simple yet powerful idea that has been around for decades [36, 37]. Decision trees are built by recursively splitting data based on the values of features until a stopping condition is met. BDTs take this idea one step further by training an ensemble of decision trees, where each tree attempts to correct the mistakes of its predecessors.

BDTs have several advantages over other machine learning algorithms, such as neural networks, for binary classification problems. They are highly interpretable, which allows us to understand how they arrived at their decisions. This is particularly important in this field, high energy particle physics, where the ability to interpret results is crucial. BDTs are also highly resilient to overfitting and can handle missing data effectively. Given their strengths, BDTs have been widely used in particle physics for binary classification tasks, such as distinguishing signal from background events. The idea of the BDTs used in high energy physics today was proposed as early as 2001 by Friedmann in the paper *Greedy Function Approximation* [38], and since then have become an indispensable tool in the field. In particular, in the ATLAS collaboration challenge *The Higgs boson machine learning challenge* [39] the creation of a BDT package called `XGBoost` [40] increased the popularity even more as they won the challenge. Today `XGBoost` has become a standard ML tool used in the field.

say what this is?

In this chapter, we will explore the theory behind BDTs in the context of *extreme gradient boosting*, for this we are mainly interested in two ways of making DTs, a set of Classification And Regression Trees (CART). The theory is mainly based on Hjort-Jensen's lecture notes [41, 42] and section 2 on the original `XGBoost` paper [40].

5.2.1 Decision trees

We will begin by explroing Decision Trees (DT) in the context of this DM search. The idea behind DTs is to find the most important features which contain the optimal information about the signal (DM) and background (SM), and then split the dataset along the values of these features with the goal of creating a dataset containing pure signal. As we are going to use kinematical variables as features we will use this to both augment and automate the classical cut and count method (Section 4.3), which is based on making kinematical cuts on the variables. To find which kinematical features are most important in splitting the dataset we have to achieve a stopping criterion where we end up on a so-called *leaf node*.

A DT is typically divided into *root nodes*, *interior nodes* and a final *leaf nodes*, also called leaves, which are all connected by *branches*, hence the name decision *tree*. As mentioned in the start of this chapter, we will look at two ways of making DTs, the first one we will look at is the *regression tree*.

5.2.2 Regression trees

As previously mentioned the leaves contain the prediction of a trained network, and are used to make new predictions on new datasets based on the information of the DT learning in the branching process. As how to construct trees, there are mainly two steps we need to follow:

1. Split the predictor space (set of possible values $x_1, x_2, \cdots, x_p$[6]) into J distinct and non-overlapping regions, $R_1, R_2, \cdots, R_J$

2. For every observation that falls into the region R_j, we make the same prediction, which is the mean of the response values for the training observations in R_J

But how do we construct the regions $R_1, R_2, \cdots, R_J$? In theory these regions could have any shape, but for simplicity and pedagogical reasons we will choose to divide the predictor shape into high-dimensional rectangles, or boxes. This means that the goal is to find boxes $R_1, R_2, \cdots, R_J$ that minimize a *cost function*, which again, for pedagogical

[6]Meaning all features in one event m in Eq. (5.1)

reasons will be the Mean Square Error (MSE) (Eq. (5.10)), defined as

$$\sum_{j=1}^{J}\sum_{i\in R_j}(y_i - \bar{y}_{R_j})^2$$

where y_i is the network output, and $\bar{y}_{R_j}$ is the mean response for the training observations withing box j. Ideally we would consider every possible partition of the feature space into J boxes, but his is computationally challenging. Thus, the common approach is to begin at the top of the tree, where all observations belong to a single region, and then split the predictor space, where every split is indicated via two new branches down in the tree. This is greedy (as in *Greedy Function Approximation* [38]) since the best split is made in the first step, rather than looking ahead and picking a split that might lead to a better tree in a future step.

To make any split we start by selecting the predictor x_j and a cut point s that splits the predictor space into two regions R_1 and R_2

$$\{X|x_j < s\} \quad \text{and} \quad \{X|x_j \geq s\}$$

so that we obtain the lowest MSE,

$$\sum_{i:x_i\in R_1}(y_i - \bar{y}_{R_1})^2 + \sum_{i:x_i\in R_2}(y_i - \bar{y}_{R_2})^2$$

where we consider all predictors $x_1, x_2, \cdots, x_p$ and each possible value s for each of them, where these values can be randomly assigned numbers. For any j and s we define the pair of half-planes where $\bar{y}_{R_1}$ and $\bar{y}_{R_2}$ are the mean responses for the training observations in $R_1(j,s)$ and $R_2(j,s)$ respectively. The goal is to find the value j and s such that we minimize the cost function, which can be done quickly depending on the number of predictors.

Then we repeat the process looking for the best predictor and best cut point, but instead of stopping with two regions, we split one of the two into another two regions creating a total of three regions. This is called the depth of a tree, and we can in principle continue to split indefinitely, however one usually sets a stopping criterion on how many events should be at a leaf.

The method explored above is straight forward, but often leads to overfitting and unnecessarily large and complicated trees. To mediate this one usually uses a so-called *Cost complexity* pruning algorithm. For regression procedures, the algorithm, rather than considering every possible subtree, considers a sequence of trees indexed by a non-negative tuning parameter α. For each value of this α there corresponds a subtree $T \in T_0$ such that

$$\sum_{m=1}^{\bar{T}} \sum_{i:x_i \in R_m} (y_i - \bar{y}_{R_m})^2 + \alpha T \tag{5.17}$$

is as small as possible. In the equation above T is the number of terminal nodes of the tree $\bar{T}$, R_m is the rectangle corresponding to the m-th terminal node. The tuning parameter α controls a trade-off between the subtree's complexity and its fit to the training data. As α increases, the above equation will give us a higher value, meaning that our trees would normally prioritize less complex models. This procedure is repeated until we minimize Eq. (5.17) and choose a suitable value for α. This is the essence of regression trees.

5.2.3 Classification trees

The second DT type we will study is the *classification tree*. Classification trees are very similar to regression trees, except that they are used to predict a qualitative response rather than a quantitative one. This means that classification trees are used for predictive modeling in which the output variable is discrete, such as classifying an email as spam or not spam, or predicting whether an event in our dataset is a DM signal or SM background event.

Like regression trees, classification trees recursively split the data based on the values of input variables until a stopping condition is met. However, the splitting criteria for classification trees are different. Classification trees cannot use the MSE function used in regression trees as a criterion for making binary splits, instead they use the *classification error rate*. The classification error rate, classification trees have discrete outputs, is simply the fraction of the training observations in a region that do not belong to the most common class.

Classification trees use measures such as Gini index or the entropy to determine quality

of a split at each node. The idea behind these quality measures is to select the split that maximizes the separation between the different categories of the output variable.

If a classification task takes for example $k = 1, 2, \cdots, K$ values as outputs, we can define a probability distribution function p_{mk} that represents the number of observations of k in a region R_m with N_m observations. We can represent this likelihood function in terms of the proportion $I(y_i = k)$ of observations of this output in region R_m as

$$p_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i = k)$$

where p_{mk} represents the majority class of observations in region m. The tree most common ways of splitting a node are given by: The misclassification error

$$p_{mk} = \frac{1}{N_m} \sum_{x_i \in R_m} I(y_i \neq k) = 1 - p_{mk}$$

The Gini index G

$$G = \sum_{k=1}^{K} p_{mk}(1 - p_{mk}) \tag{5.18}$$

and the entropy s

$$s = -\sum_{k=1}^{K} p_{mk} \log(p_{mk})$$

5.2.4 The CART algorithm

As we are going to look at extreme gradient boosting, the main algorithm for this type of DT is the Classification And Regression Tree (CART). For classification, the CART algorithm splits the dataset in two subsets using a single feature k and threshold t_k. This could for example be a threshold set on the value of the invariant mass of an event we are trying to classify as background or signal. The way we optimize the pair (k, t_k) such that we get the purest subset is by for example using the Gini factor G. With this the cost function tries to minimize

$$C(k, t_k) = \frac{m_{left}}{m} G_{left} + \frac{m_{right}}{m} G_{right}$$

where $G_{left/right}$ is the impurity of the left/right subset given in Eq. (5.18) and $m_{left/right}$ is the number of events in the left/right subset. Once the algorithm successfully splits the training set in two, it splits the subsets using the same algorithm, and so on, recursively. For our purposes we will make the DT stop searching for the pair (k, t_k) once we have reached the maximum depth we chose, or whenever the sum of the weights on a leaf is one. Both of these are hyperparameters that need to be optimized, as will be done in Section 7.3.2.

The CART algorithm for regression works similarly to the one for classification, but instead of trying to split the training set in a way that minimizes the Gini or entropy impurity, it tries to split the training set in a way that minimizes the (MSE). Meaning that we have

$$C(k, t_k) = \frac{m_{left}}{m} MSE_{left} + \frac{m_{right}}{m} MSE_{right}$$

with

$$MSE_{node} = \frac{1}{m_{node}} \sum_{i \in node} (\bar{y}_{node} - y_i)^2$$

and

$$\bar{y}_{node} = \frac{1}{m_{node}} \sum_{i \in node} y_i$$

5.2.5 Ensemble modeling and (extreme) gradient boosting

Now that we have explained how DTs work, we still need more information to get the full picture of *Boosted* DTs. As we are going to look at a massive dataset, a single tree is not strong enough to be used in practice. What is actually used is the *ensemble model*, which sums up the prediction of multiple trees together. What is important in these models is that the different trees we are summing together try to complement each other. For example if the first split on one tree is on the invariant mass of a dilepton pair, while on another tree it is the MET, then we know physically that these are important features that DM + dilepton models have as a signature, meaning that it should be a good combination. Mathematically, we can write the prediction of an ensemble model in the form

$$\hat{y}_i = \sum_{k=1}^{K} f_k(x_i), \quad f_k \in \mathcal{F} \tag{5.19}$$

where K is the number of trees we want to sum over, f_k is a function of the functional space $\mathcal{F}$, where $\mathcal{F}$ is the set of all possible CARTs. With this we want to minimize the objective function of the form

$$\mathcal{L} = \sum_{i}^{n} C(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k)$$

where we sum over n leaves, C is training cost function (for example it can be the MSE), and $\Omega(f_k)$ is the complexity of the tree f_k, which penalizes the more complex models. We can see from both functions above that we need to teach all trees at once to make predictions, this is however both computationally demanding and intractable. Instead, we can use an additive strategy, where we fix what we have learned, and add a new tree at a time. Using Eq. (5.19) with 0 trees and going up we see that we get

$$\begin{aligned}
\hat{y}_i^{(0)} &= 0 \\
\hat{y}_i^{(1)} &= f_1(x_i) = \hat{y}_i^{(0)} + f_1(x_i) \\
\hat{y}_i^{(2)} &= f_1(x_i) + f_2(x_i) = \hat{y}_i^{(1)} + f_2(x_i) \\
\hat{y}_i^{(3)} &= f_1(x_i) + f_2(x_i) + f_3(x_i) = \hat{y}_i^{(2)} + f_3(x_i)
\end{aligned}$$

and so on. Mathematically we see that a prediction at step t as in $\hat{y}_i^{(t)}$ takes the form

$$\hat{y}_i^{(t)} = \sum_{k=1}^{t} f_k(x_i) = \hat{y}_i^{(t-1)} + f_t(x_i) \tag{5.20}$$

Such that we can now write the objective to be

$$\mathcal{L}^{(t)} = \sum_{i}^{n} C(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \tag{5.21}$$

from which we can set up our choice for cost function. In general case we take the Taylor expansion of the loss function up to the second order such that we get

$$\mathcal{L}^{(t)} = \sum_{i}^{n} \left[C(y_i, \hat{y}_i^{(t-1)}) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) + const$$

where

$$g_i \equiv \partial_{\hat{y}_i^{(t-1)}} C(y_i, \hat{y}_i^{(t-1)}) \quad \text{and} \quad h_i \equiv \partial^2_{\hat{y}_i^{(t-1)}} C(y_i, \hat{y}_i^{(t-1)})$$

Such that the specific objective at step t becomes

$$\sum_{i=1}^{n} \left[g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \tag{5.22}$$

where we removed all constants, such that Eq (5.22) becomes the optimization goal for the new tree. As we can see the above equation can take into account any cost function as it is written in a general form with a Taylor expansion. Furthermore, we can rewrite this by expanding

$$\Omega(f_t) \rightarrow \alpha T + \frac{1}{2} \lambda \sum_{j=1}^{T} w_j^2$$

where the first term is the same as in Eq. (5.17), and the second is called the *regularization term* (see Section 5.3.1 for a detailed explanation). Inserting this into Eq. (5.22) we get

$$\mathcal{L}^{(t)} = \sum_{j=1}^{T} \left[\left(\sum_{i=1}^{n} g_i \right) w_j + \frac{1}{2} \left(\sum_{i=1}^{n} h_i + \lambda \right) w_j^2 \right] + \alpha T$$

such that with a fixed tree structure, q, we can define the optimal weight w_j^* on a leaf j as

$$w_j^* = -\frac{\sum_{i=1}^{n} g_i}{\sum_{i=1}^{n} h_i + \lambda} \tag{5.23}$$

and correspondingly the optimal objective reduction

$$\mathcal{L}^{(t)}(q) = -\frac{1}{2} \sum_{j=1}^{T} \frac{(\sum_{i=1}^{n} g_i)^2}{\sum_{i=1}^{n} h_i + \lambda} + \alpha T \tag{5.24}$$

With both of these equations we have all we need to use BDTs, the extreme part of the gradient boosting is something the `XGBoost` algorithm has implemented in terms of system optimization, such as parallelizing the tree boosting task, and can be read in more detail in section 4 of the original paper [40].

5.2.6 Summary

So in short BDTs split datasets based on different values of the features on the dataset, Figure 5.2 shows an illustration of how DTs work.

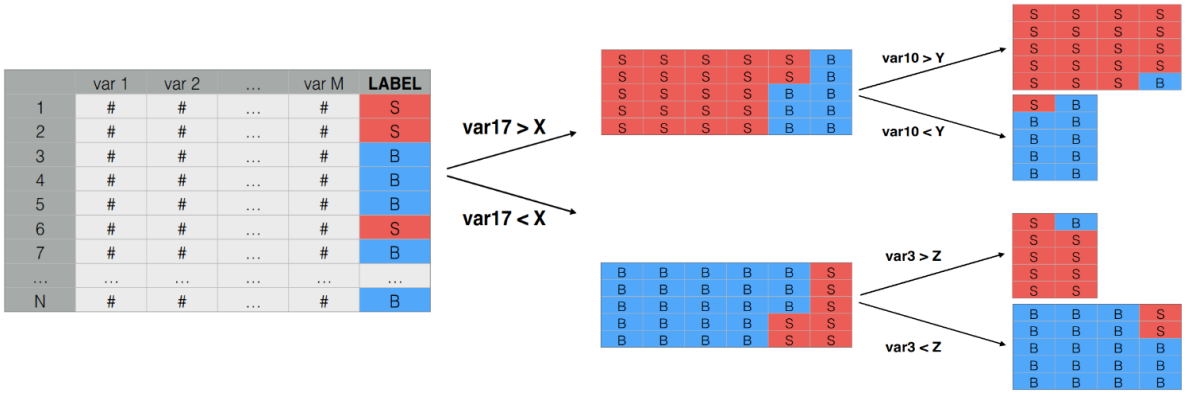


Figure 5.2: Illustration showcasing how decision trees work by splitting a dataset of N events with M features whenever a threshold X, Y or Z for a feature is passed. The label S and B state whether the event is signal or background respectively.

The illustration in Figure 5.2just shows how one tree might work, in reality we can choose to have an arbitrary number of trees and combine their results with the ensembling and boosting method. Then after combining the generated tree structures we use the optimal weight in Eq. (5.23) and the optimal objective reduction Eq. (5.24) to minimize the objective to get the most accurate predictions as possible. But this will come later when applying these to data in Section 7.3.2.

5.3 Tools and evaluation methods used for both algorithms

Now that we have explained how both NNs and BDTs work, we will present the tools and evaluating method that we will use in this thesis.

5.3.1 Cost functions

Cost functions have been mentioned throughout this chapter, but what are they? Cost functions are what we will utilize to evaluate how well the output of the network fares against the target, i.e. if our network "guesses" right whether an event is signal or background, thus making this a very important part of our network! Before getting into this we first have to look at logistic regression. Since we are studying a binary classification task where the output is either $t_i = 0$ or $t_i = 1$, meaning background or signal. We can introduce a polynomial model of order n as

$$\hat{y}_i = \beta_0 + \beta_1 x_i + \beta_2 x_i^2 + \cdots + \beta_n x_i^n$$

where the hat notation, $\hat{y}$, symbolizes a condensed matrix form of the output of a layer. We can define the probabilities of getting $t_i = 0$ or 1 given our input x_i and $\boldsymbol{\beta}$ (weights and biases) with the help of a logistic function. Using this we get the probability as

$$p(t_i = 1|x_i, \boldsymbol{\beta}) = \frac{1}{1 + e^{-t_i}}$$

and

$$p(t_i = 0|x_i, \boldsymbol{\beta}) = 1 - p(t_i = 1|x_i, \boldsymbol{\beta})$$

We want to then define the total likelihood for all possible outcomes from a dataset $\mathcal{D} = \{(t_i, x_i)\}$, with the binary labels $t_i \in \{0, 1\}$, applying the Maximum Likelihood Estimation (MLE) principle. This gives us

$$P(\mathcal{D}|\boldsymbol{\beta}) = \prod_{i=1}^{n} \left[p(t_i = 1|x_i, \boldsymbol{\beta})\right]^{t_i} \left[1 - p(t_i = 1|x_i, \boldsymbol{\beta})\right]^{1-t_i}$$

from which we obtain the log-likelihood

$$C(\boldsymbol{\beta}) = \sum_{i=1}^{n} (t_i \log p(t_i = 1|x_i, \boldsymbol{\beta}) + (1 - t_i) \log[1 - p(t_i = 1|x_i, \boldsymbol{\beta})])$$

By taking the parameter $\boldsymbol{\beta}$ to first order in x_i and reordering the logarithm we get

$$C(\boldsymbol{\beta}) = -\sum_{i=1}^{n} (y_i(\beta_0 + \beta_1 x_i) - \log(1 + \exp(\beta_0 + \beta_1 x_i))) \tag{5.25}$$

This equation we will use throughout the thesis and is known as the *cross entropy*. The two beta parameters used are the weight and biases, $\beta_1 = w$ and $\beta_0 = b$. The goal is to optimize these parameters such that it minimizes the cost function.

L2-regularization

Furthermore, we will add an extra term to the cost function, proportional to the size of the weights. We do this to constrain the size of the weights, so they do not grow out of control, this is to reduce *overfitting*. We will use the so-called *L2-norm* where the cost function becomes

$$C(\boldsymbol{\beta}) \to C(\boldsymbol{\beta}) + \lambda \sum_{ij} w_{ij}^2 \tag{5.26}$$

meaning that we add a term where we sum up **all** the weights squared. The factor λ is called the regularization parameter. The L2-norm combats overfitting by forcing the weights to be small, but not making them exactly zero. This is so that less significant features still have some influence over the final prediction, although small.

5.3.2 Sample weight

A "sample weights" array is an array of numbers that specify how much weight each sample in a batch should have in computing the total loss. It is commonly used in imbalanced classification problems (the idea being to give more weight to rarely-seen classes) [43]. To mathematically illustrate how this weight works is by multiplying a constant term χ_i into the cost/loss function that is used further on the error propagation.

As an example we can extend the simple MSE, Eq. (5.10) to

$$C(\hat{W}) = \frac{1}{2}\sum_{i=1}^{n} \chi_i (y_i - t_i)^2 \tag{5.27}$$

This will become an important feature when discussing network optimization in Chapter 7, and especially for particle physicists as we want to re-weight Monte Carlo events to expected events to correctly showcase kinematical distributions.

5.3.3 Area under the ROC-curve

To evaluate how well our networks do, we will use the accuracy, which literally tells us how the fraction of events it guessed right. But we will also take into account a more advanced metrics, this is the Area Under the "Receiver Operating Characteristic (ROC)" Curve (AUC). The theory for this section is based on the Wikipedia page [44]. Before explaining what the AUC is we first need to explore the ROC curves. The ROC curves helps us illustrate how successful a network is doing *binary classification*. The ROC curve, as the name states, is a curve, where we plot the *True Positive Rate* (TPR) against the *False Positive Rate* (FPR). The TPR is defined by dividing the *True Positive* (TP), which is when the network correctly guesses a signal event to be a signal event, by the *Positive* (P), which is the total number of signal events in the data.

$$\text{TPR} = \frac{\text{TP}}{\text{P}}$$

On the other hand, the FPR is when we divide the *False Positive* (FP), when the network guesses a background event to be signal event, divided by the *Negatives* (N), the total number of background events in the data.

$$\text{FPR} = \frac{\text{FP}}{\text{N}}$$

To get a numerical value of how well a network classifies data we thus calculate the AUC. The TP and the FP are both going to be predictions from the networks, which for this thesis will give an event a score from 0 to 1[7], meaning that if we had a perfect network

[7]Where 0 means that the network believes an event has 0% chance of being a signal event, and 1 means that the networks believes with 100% certainty that an event is a signal event.

which guessed everything correctly we would only get TPs and thus an AUC of 1. If we were to get an area of 0.5 this would mean that the network is randomly guessing whether an event is a signal or background, practically making a coin toss for every event. The goal is to have a network to give an AUC score as close to 1 as possible.

While only using the accuracy as a metric is not a bad start, it is not favorable to use as a metric if datasets are unbalanced. As en example, if we had a dataset with 100 events, where 95 were background events and 5 were signal events, if we only used accuracy as a metric we would be inclined to think that an accuracy of 95% is great. However, as the dataset is unbalanced, the network could easily take a shortcut and guess every event to be a background event. Meaning that the network learned nothing. This is where the AUC comes into play, as this metric highlights whenever a network guesses wrongly in a binary classification. Meaning that if we used the worst case scenario with 95% accuracy when guessing everything to be background, we would have a TPR = 0 and an FPR = 0, if we were to plot this we would get a flat line on the FPR and TPR axis, meaning that it would give us an AUC of 0.5. Highlighting that our network is randomly guessing, and that it needs to be optimized further. In Figure 5.3 we see an example of a ROC curve.

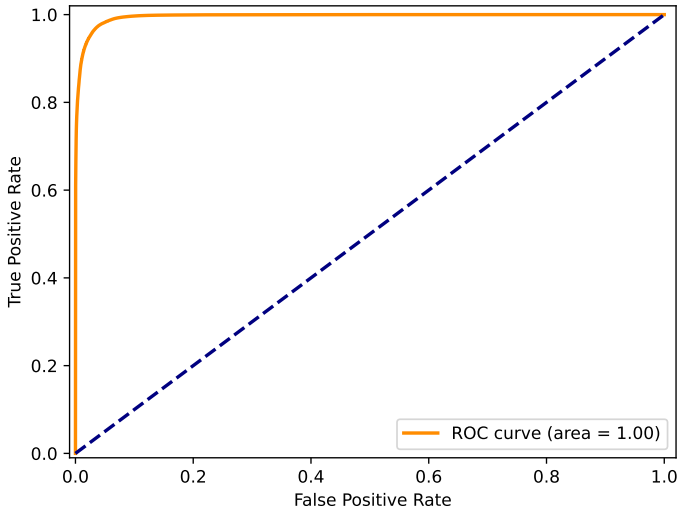


Figure 5.3: Illustration of a ROC curve. The orange line is the result of a network, while the dashed blue line is to illustrate a random guess (AUC = 0.5)

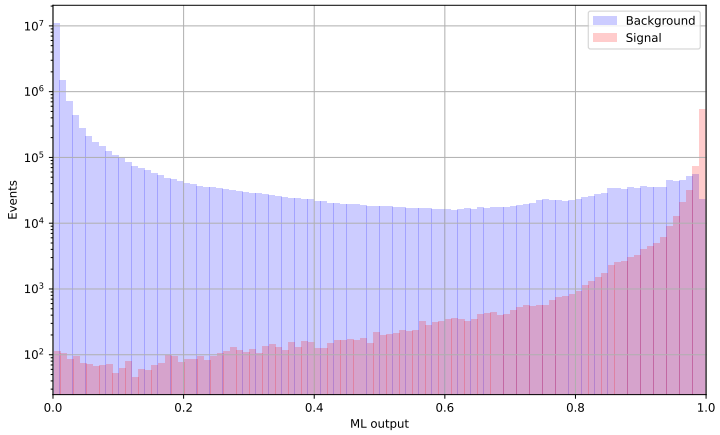
5.3.4 Validation plots

Another tool that we will use to evaluate how well a network does is by making so-called *validation plots*. The way these plots work is by making a distribution with histograms of the ML output for both signal and background. As a score closer to 1 means that the network predicts an event to be a signal event, and 0 means that the network predicts a background event. Then we would ideally have all the signal events shoved to the right and all the background events shoved to the left. Usually this is not the case, as some background events might be similar to signal events. We will therefore utilize this kind of plots to see the distribution of the network predictions.

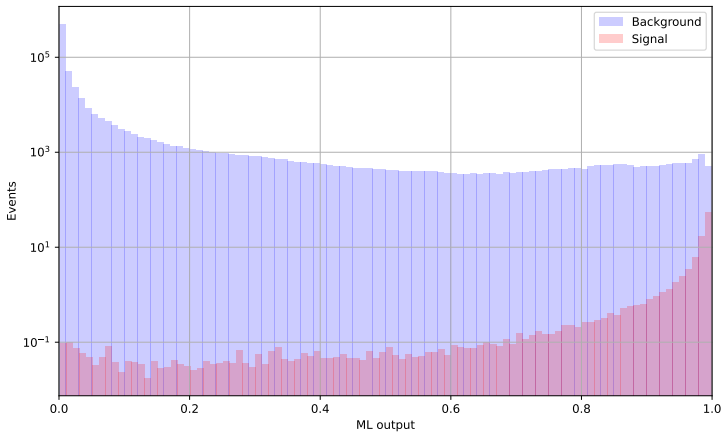
In addition to just sorting events by the ML score, we will also re-weight the events (which are MC events) into expected events[8], which takes into account the cross-section of a process in an event basis. The difference this makes can be seen in Figure 5.4, where we can see how an arbitrary distribution changes when applying re-weighting, the most notable difference is in the last bins where we stop having more signal events than background events. We can however still use the non-re-weighted plots to see how well the network has learned to categorize, but these plots will not be used in this thesis, and can rather be found on the GitHub repo[9].

It is standard practice to make fill the histograms with semi-transparent colors, such that one can visualize the whole spectra for both background and signal. In this thesis we will use this, as well as validation plots where we divide the SM events into their respective background channels and fill them with opaque colors, while leaving the signal unfilled. We will in addition include real data points in a control region, meaning up to a score of 0.6, to see how well SM simulations agree with the data. Ideally we want our data to not agree with the SM simulations when the network predicts a score close to one, as this discrepancy could be a hint of new physics. But as we cannot show signal regions before a data unbinding procedure we will not pursue the latter case.

[8] See Section 7.1.1 for more information
[9] Available in Plots here: https://github.com/rubenguevara/Master-Thesis/tree/master/



(a) No re-weighing MC events to expected events



(b) Re-weighing MC events to expected events

Figure 5.4: Illustration of the validation plots. Plot a) shows how the network learned a dataset without re-weighting events, while b) shows how it looks for re-weighted events. For this example we will not show any numbers, but the plot a) has more signal events than the luminosity allows for, while plot b) is corrected.

5.3.5 Significance plots

To create signal regions which will be used for statistical analysis we will also look at significance plots. The way these plots work is by using the expected significance, Eq. (4.19), on the number of signal and background events that are present after making a cut in the ML output on the validation plots. As an example, Figure 5.5 shows the re-weighted validation plot from the previous section (upper figure). We can calculate the expected significance by counting the number of signal and background events that have an ML score greater than 0.5, 0.6, 0.7, 0.8, 0.9 and 0.99 respectively. The significance distribution is shown in the lower plot of Figure 5.5. Here we see that the greatest expected significance lies on the last bin of the upper distribution, the example used reaches an expected significance of $\approx 2.35\sigma$. We will use this as a metric to choose where we will make a kinematical cut before making exclusions.

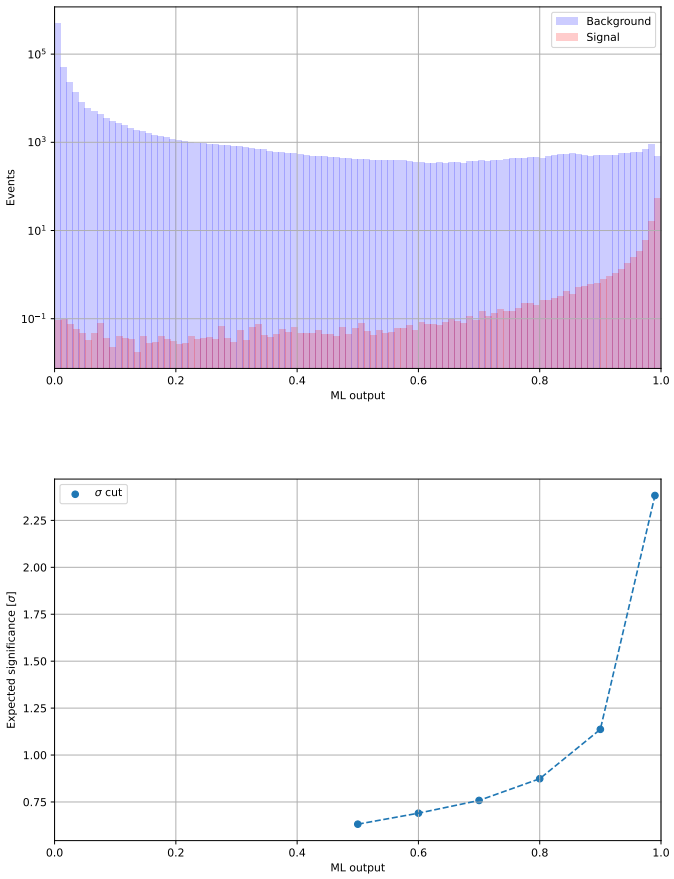


Figure 5.5: Plot a) shows an arbitrary ML score distribution. Plot b) shows the expected significance that is calculated from the distribution shown in plot a) when making cuts on the ML output score. The dashed lines are to guide the eye.

5.3.6 BDT and NN exclusive plots

Feature importance

For BDTs, as we can calculate which features had the most impact when splitting the data, we will also include feature importance plots. As the name states these plots tell us how important the features in the dataset were in the training to separate signal from background. As we are using the `XGBoost` package when working with BDTs, we will use the inbuilt functions of the package to plot the feature importance.

When plotting the feature importance there are several metrics we can use to determine how important they are, for this thesis we will mainly look at

- Gain: which measures the relative contribution of a feature to the model's performance. This is the standard metric used by `XGBoost`.
- Weight: the number of times a feature appears on a tree to split the data
- Cover: the number of samples affected by the split with a feature

Metric and loss evolution

For NNs we can plot how the metrics (AUC and binary accuracy) as well as the loss function change over the training epochs. We will plot these changes for both the training and testing sets. The goal is that the metric and loss converge to the same point for the training and testing set.

5.4 Summary

Now that we have explored the theoretical foundation of the ML algorithms we will be studying, as well as the tools we will be using to test how well the algorithm has performed, we are ready to start telling about the methods we will be utilizing in this thesis. The next part will be about the data preparation process, which shows the kinematical variables that we will use as features, as well as the size of the dataset. We will also present in the next part the challenges that arise when using extreme datasets as we do in high energy physics, including the methods used to mitigate these problems.

Part II

Methods

Chapter 6

Data Preparation

Now is the time to focus on the critical task of preparing the data for our DM search using ML. This task is essential because accurately distinguishing between signal and background events is a key challenge in searching for any new signal event, as seen in Chapter 4.3. In this chapter we will define what *background* and *signal* actually are, as well as selecting appropriate kinematical cuts to define the preselection region for our search. In addition, we will explain the data preparing procedure such that it is in a format that is well-suited for our ML algorithms. The reason this process is of great importance is, so we can maximize the chances of our ML algorithm to accurately identify any DM signals. Moreover, by providing a detailed explanation of the data preparation process, we can ensure that our dataset is both reliable and effective, and that our analysis is robust and trustworthy. In short, this chapter represents a critical step towards achieving our goal of using ML to classify DM from SM events using ATLAS data, and this is essential to the success of our overall research goal.

6.1 Standard Model background estimation

As we are going to look at dilepton final states with missing transverse energy to try to teach our ML to learn DM signatures, then we need to take into account all possible SM backgrounds events that share the same final states. The SM backgrounds processes we will look at are explained in the next subsections. Appendix H gives the full list of dataset IDs for background processes used in this thesis.

6.1.1 W and Drell Yan

W+jets and Z^*/γ^*+jets can lead to final states that mimic the one we are searching for. The W boson can decay to a lepton and neutrino $W \to l\nu_l$, which features MET, and $Z/\gamma \to ll$, which ends as a dilepton final state. The dominant production mechanisms for these background processes are shown in Figure 6.1. Where the quark jet can keep on decaying, potentially weakly creating another lepton and neutrino giving rise to a dilepton final state. We will divide these background processes into two, W and *Drell Yan* processes, where the latter is Z/γ^* + jets. To simulate these background processes Sherpa 2.2.11 [45] was used.

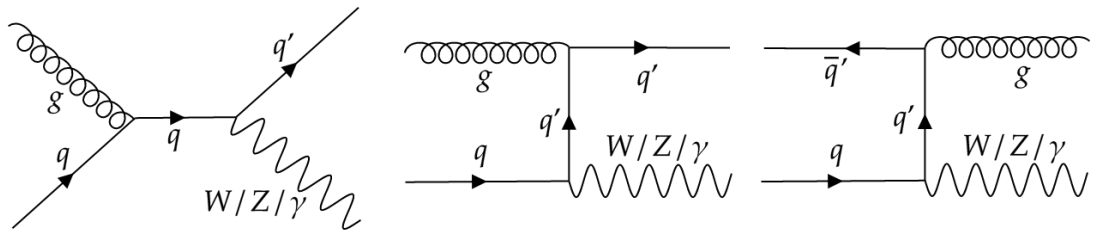


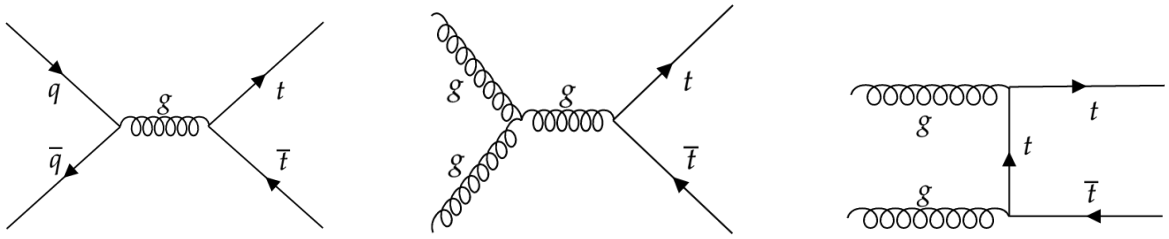
Figure 6.1: Diagrams showcasing SM W and Drell Yan production. A quark marked with a prime, indicated that the quark might be a different flavor after interacting with a boson.

6.1.2 Top pair

Another SM background process we will look at are $t\bar{t}$ processes. Since the top has a close to 100% branching ratio of decaying into a b-quark[1] and a W boson, $t \to bW^+$, where the W boson can again decay into a neutrino and lepton, where the first leads to MET. The overall final state is dilepton with MET together with two b-jets. The main

[1]Making $b-$tagging an effective method to reduce this background channel

production mechanism for these processes can be seen in Figure 6.2. To simulate these background processes Powheg-Box v2 [46] interfaced with Pythia 8 [47] was used.

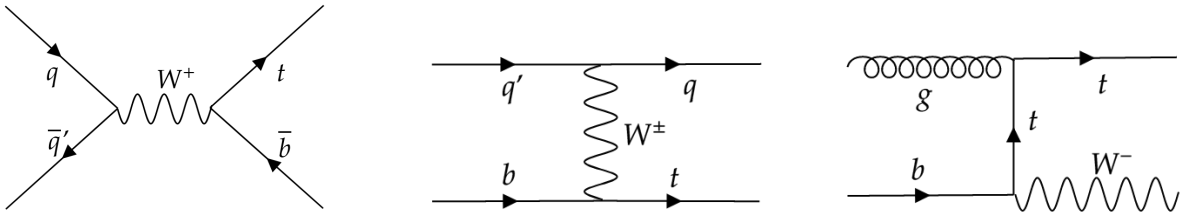


(a) s-channel (b) Gluon fusion (c) t-channel

Figure 6.2: Diagrams showcasing SM $t\bar{t}$ production

6.1.3 Single top

On the same note we have processes with a single top, where again the top has a high branching ratio of decaying into a b-quark and a W boson, $t \to bW^+$, where the W boson can again decay into a lepton and neutrino, giving MET. The main production mechanism for these processes can be seen in Figure 6.3. The b-mesons, and quark (for t-channel) can occasionally decay semi-leptonically giving us the second lepton and another neutrino. To simulate these background processes Powheg-Box v2 [46] interfaced with Pythia 8 [47] was used.



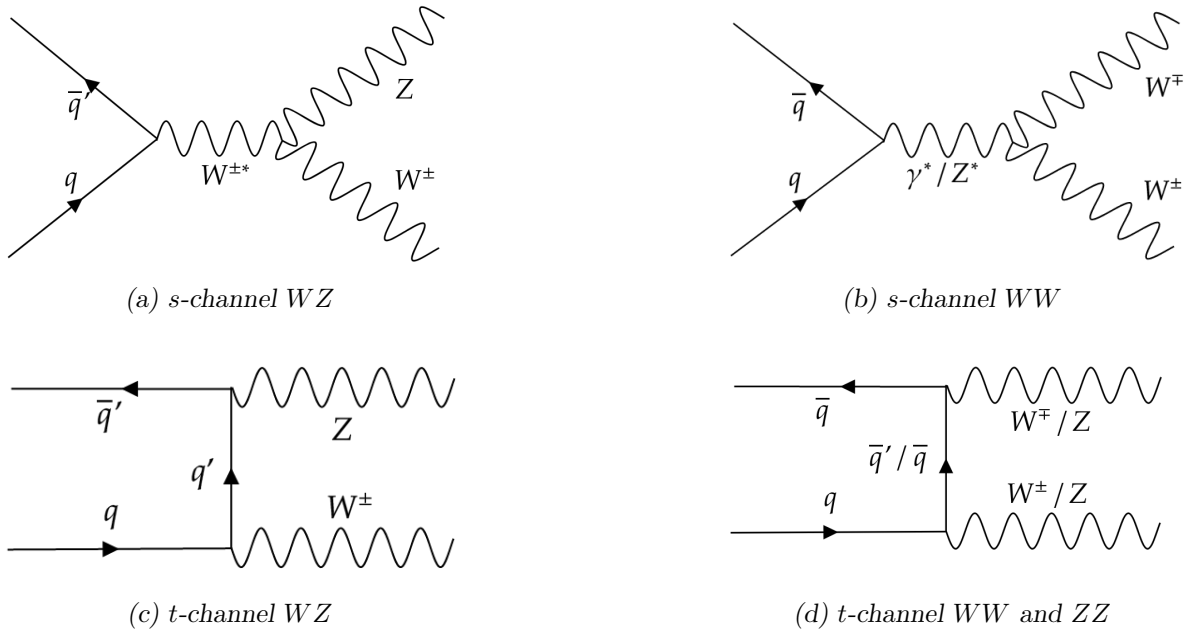
(a) s-channel (b) t-channel (c) tW

Figure 6.3: Diagrams showcasing SM single top productions. A quark marked with a prime, indicated that the quark might be a different flavor after interacting with a boson.

6.1.4 Diboson

The last SM background process we will look at are processes containing two bosons, called *diboson* backgrounds. The two SM bosons we will consider when looking into these final states are the W and Z, as these can decay as $Z \to ll$ or $W \to l\nu_l$, where the prime means a different lepton flavor. The main production mechanism for these

processes can be seen in Figure 6.4. From these diagrams we have that $WW \to ll + \nu\nu$, $WZ \to l\nu + ll$[2] and $ZZ \to ll + \nu\nu$, which all lead to a dilepton with MET final state. To simulate these background processes Sherpa 2.2.11 [45] was used.



(a) s-channel WZ *(b) s-channel WW*

(c) t-channel WZ *(d) t-channel WW and ZZ*

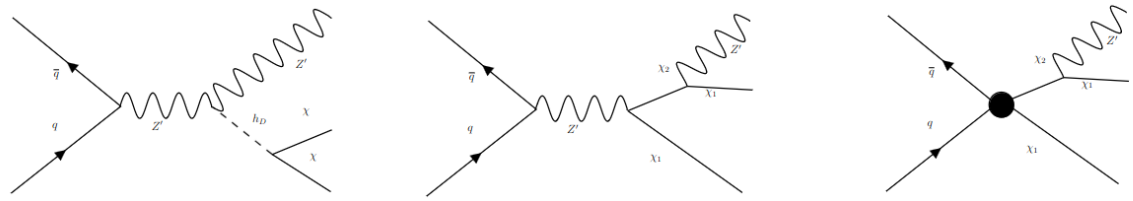
Figure 6.4: Diagrams showcasing SM diboson production. A star superscript on a boson indicates that the boson needs to be virtual and off mass shell. A quark marked with a prime, indicated that the quark might be a different flavor after interacting with a boson.

[2]Where one lepton is not reconstructed

6.2 Dark Matter samples

6.2.1 Mono-Z'

The $Z'-$aware DM models described in Chapter 3 showcased in Figure 6.5.



(a) *Dark Higgs* (b) *Light Vector* (c) *Inelastic EFT*

Figure 6.5: Mono Z' models

These models contain three relevant couplings

- the coupling g_D between the Z' and the dark sector particles ($Z'Z'h_D$ (See Figure 6.5a) or $Z'\chi_1\chi_2$ (See Figure 6.5b and 6.5c))
- the coupling g_q between the Z' and quarks
- and the coupling g_l between the Z' and leptons

The coupling to quarks and leptons are assumed to be the same for all generations. When simulating the couplings were set to

- $g_D = 1$,
- $g_q = 0.1$,
- and $g_l = 0.01$.

In addition to the couplings, as mentioned on Chapter 3, we will divide the three models, Dark Higgs, Light Vector and inelastic EFT into two more regions. The difference being the masses, Table 6.1 showcases the definition of the Light Dark Sector (LDS) and Heavy Dark Sector (HDS).

Table 6.1: Dark sector masses in light- and heavy dark sector models

	Dark Higgs	Light Vector / Inelastic EFT
Light Dark Sector	$m_\chi = 5$ GeV $m_{h_D} = 125$ GeV	$m_{\chi_1} = 5$ GeV $m_{\chi_2} = m_{\chi_1} + m_{Z'} + 25$ GeV
Heavy Dark Sector	$m_\chi = 5$ GeV $m_{h_D} = m_{Z'}$	$m_{\chi_1} = m_{Z'}/2$ $m_{\chi_2} = 2m_{Z'}$

The simulated masses of the new Z' boson for this thesis are

$$m_{Z'} = [130, 200, 400, 600, 800, 900, 1100, 1200, 1300, 1400, 1500] \text{ GeV},$$

for the DH and LV model and

$$m_{Z'} = [130, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 1100, 1200, 1300, 1400, 1500] \text{ GeV},$$

or the EFT model.

6.2.2 Supersymmetric direct slepton production

For the supersymmetric direct slepton production model, $\tilde{\ell}\tilde{\ell} \to \ell\ell\tilde{\chi}_1^0\tilde{\chi}_1^0$, depicted in the Feynman diagram in Figure 6.6

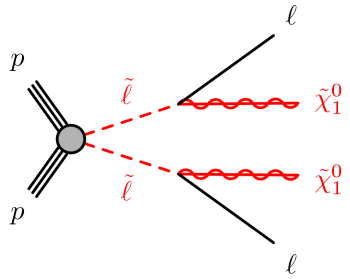


Figure 6.6: Direct slepton production

for this thesis the masses of the sleptons, $m_{\tilde{\ell}}$, and neutralinos, $m_{\tilde{\chi}_1^0}$ are the parameters that will be changed. The mass pairs of the neutralinos and sleptons used in this thesis are shown in a scatter plot in Figure 6.7.

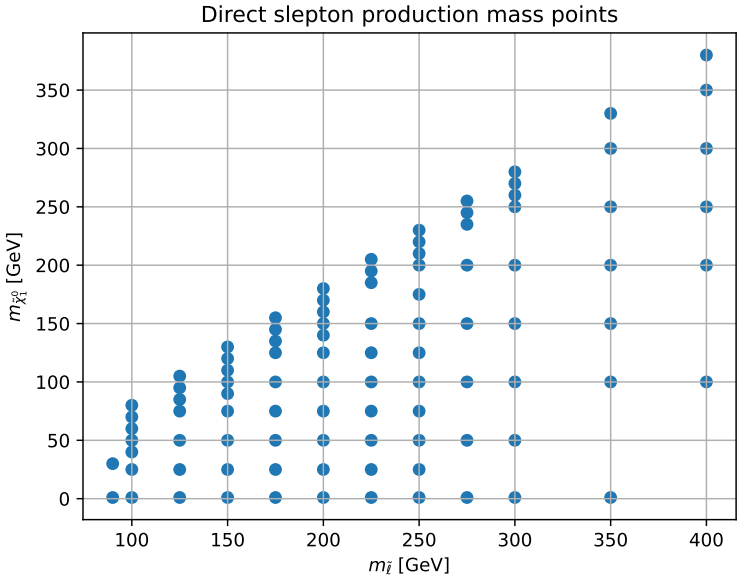


Figure 6.7: Scatter plot showcasing the $\tilde{\chi}_1^0$ and $\tilde{\ell}$ mass pairs for direct slepton production studied in this thesis. Each dot represents a simulated direct slepton process using the masses.

6.2.3 2HDM + a

For the Two Higgs Doublet Model with an additional pseudoscalar a (2HDM +a) which decays into DM shown by the Feynman diagram in Figure 6.8

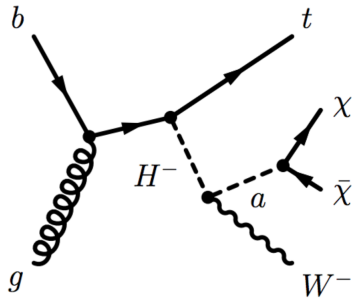


Figure 6.8: 2HDM + a

There are three important parameters for this model the mass of the new charged Higgs, m_{H^-}, the mass of the pseudoscalar a, m_a, and the ratio between the vacuum expectation values of the Higgs doublet, $\tan\beta$. The values used for this thesis can be sen below. In addition to these values, we assume the DM candidate mass to be $m_\chi = 10$ GeV.

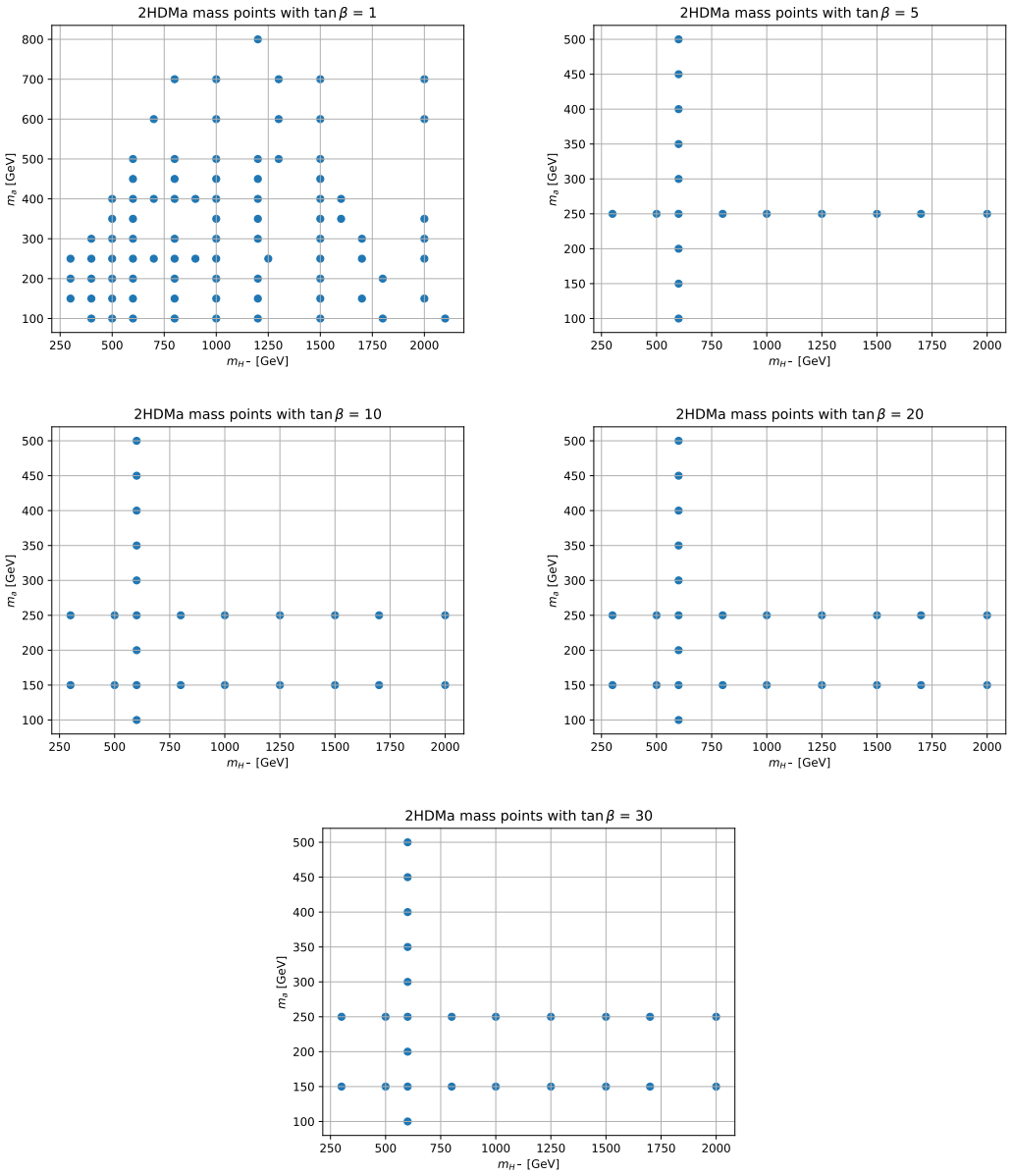


Figure 6.9: Scatter plot showcasing the m_a and m_{H^-} mass pairs for 2HDM + a model studied in this thesis. Each dot represents a simulated 2HDM + a process using the masses. As we have different values for $\tan\beta$ we show the different simulated samples for each mass for each value.

6.3 Object selection

Before getting into our DM search and the regions we will study we have to define how standard objects are defined. By standard objects we mean electrons, muons and jets. As this thesis is made using ATLAS data and simulations we will be following the central recommendations.

6.4 Preselection region

Now that we have defined what our backgrounds and signals are, the next step is to create a so-called *preselection region*. The preselection region is the kinematical region we will use as a base for our search. As we are conducting a model independent search we want our kinematical cuts to be minimal and as general as possible. As we are looking at a dilepton final state, then we need to first define what we mean by that. If we were only searching for DM with the Z' model, then a sensible definition would be Same Flavor Opposite Sign (SFOS) leptons ($e^{\pm}e^{\mp}, \mu^{\pm}\mu^{\mp}$). However, to stay as general, and model independent, as possible we will also study other possible combinations as these might be important for theories such as SUSY or Lepton Flavor Violating (LFV) models. These are Different Flavor Same Sign (DFSS), DFOS and SFSS lepton pairs.

Since we are looking for DM, which we expect to behave similarly to a neutrino, then a nice kinematical variable to set a general cut to isolate signal from background is the MET, we can see the distribution of MET on all of Run II in a dilepton final state in Figure 6.10, where we have applied a cut of $m_{ll} > 10$ GeV to exclude hadrons, as well as the central recommendations cuts.

As we are conducting a model independent search, we want to use minimal cuts. The MET cut made in this thesis was chosen to be of 50 GeV (violet line in plot), meaning we will *only* look at events where the MET is greater than this. As we can see from Figure 6.10 by making a kinematical cut on 50 GeV we are cutting out a massive part of the background processes while only losing a small part of the signal.

Other than the object selection criteria, these are the only cuts that will be used to define the preselection region. Their summary can be seen in Table 6.2

Table 6.2: Table showcasing the cuts used to define the preselection region for our model independent search.

Feature	Selection criteria
Dilepton final state	$\ell^{\pm}\ell^{\mp}$, $\ell^{\pm}\ell^{\pm}$, $\ell^{\pm}\ell'^{\mp}$ and $\ell^{\pm}\ell'^{\pm}$
Missing Transverse Energy	$E_T^{miss} > 50$ GeV

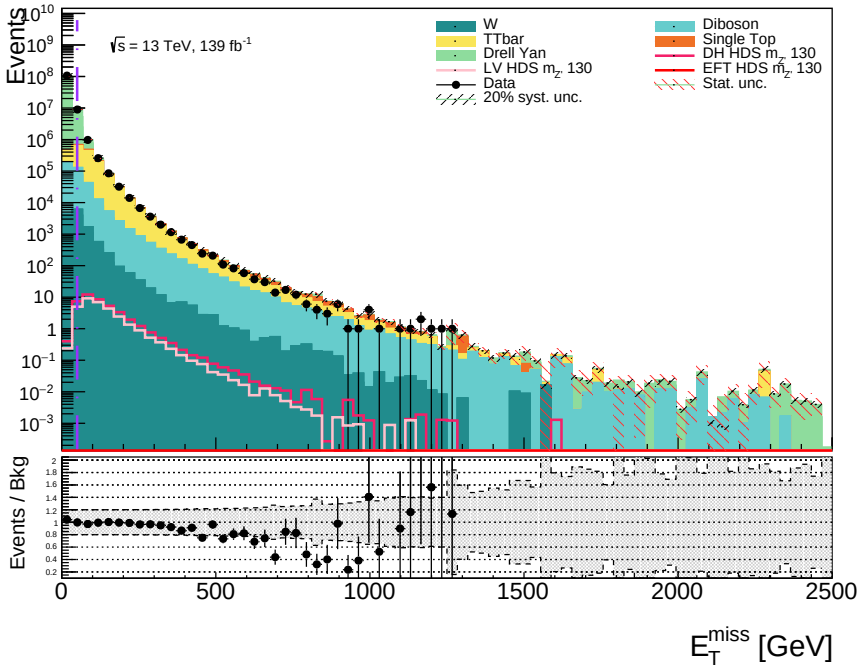


Figure 6.10: Distribution of MET when looking at a dilepton final state in all of Run II. The violet line shows a MET cut of 50 GeV to create a preselection region. Three DM models are also shown, in addition to the SM expectations.

6.5 Feature selection for ML

For this thesis there are many possible kinematic variables that can be used as features for our ML algorithms, in this section we will make use of the kinematical variables presented in Section 4.1.1 as well as introduce new variables that might help our ML algorithm to correctly learn the patters of SM background and DM signal events.

As we are studying a final state with two leptons it is natural to look at the kinematics for both of these objects. The first thing we will look at is the transverse momentum, p_T, of each lepton as defined in Eq. (4.2). We will also look at the azimuthal angle, ϕ and the pseudorapidity, η defined in Eq. (4.8). With this we have practically constructed a four-momentum from which we could learn all particle kinematics. However, we want to help our ML algorithm as much as possible in the task of learning SM background and DM signals. A powerful kinematical variable for this is therefor the invariant mass, m_{ll} defined in Eq. (4.3), which might as an example can help the ML algorithm sort out resonant models. Another variable of interest is the transverse energy, E_T defined in Eq.

(4.4) for the lepton pair. The distribution of the invariant mass can be seen in Figure 6.11.

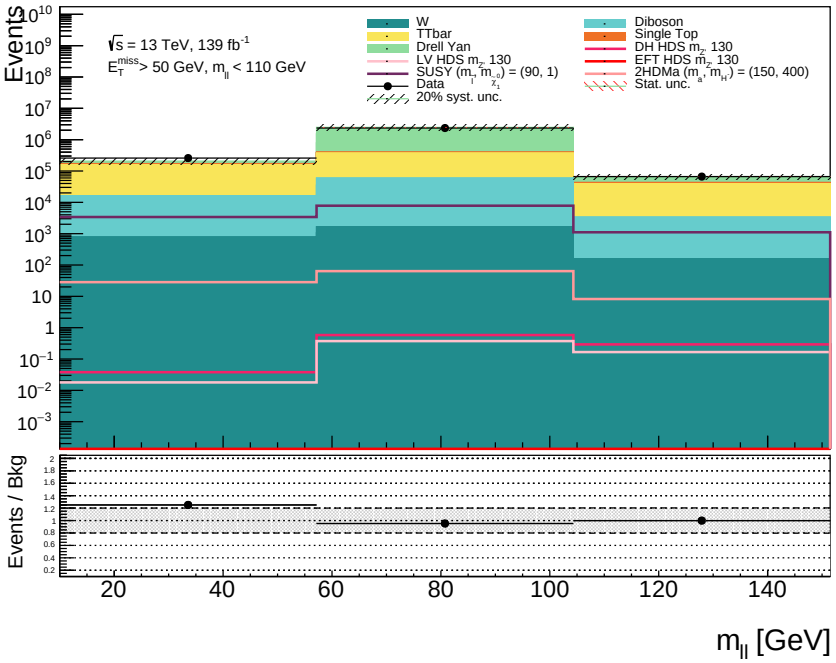


Figure 6.11: Distribution of m_{ll} in control region. For the 2HDM + a we only include the distribution for $\tan\beta = 1$.

As we are studying DM, an invisible particle, the most important kinematical variable to distinguish background from signal is the missing transverse energy, E_T^{miss}. Another version of the MET is the so-called *Object-based* E_T^{miss} *significance*, or $E_T^{miss,sig}$ for short, this variable is used to deal with artificial or fake E_T^{miss}. The way $E_T^{miss,sig}$ works is by weighing the value of E_T^{miss} by the precision of its reconstruction. It is defined as

$$E_T^{miss,sig} = \frac{E_T^{miss}}{\sigma(E_T^{miss})} \tag{6.1}$$

where $\sigma(E_T^{miss})$ is the uncertainty of the reconstruction of the E_T^{miss}, which consider the individual uncertainties of the objects that enter the E_T^{miss} calculation. The distribution of this variable can be seen in Figure 6.12. In this thesis we will use both E_T^{miss} and $E_T^{miss,sig}$ even though they are correlated, as the algorithm might choose in different instances[3] which of these is of more importance.

[3]Meaning different *signal regions* when conducting a model independent search. See Chapter 7.1.2

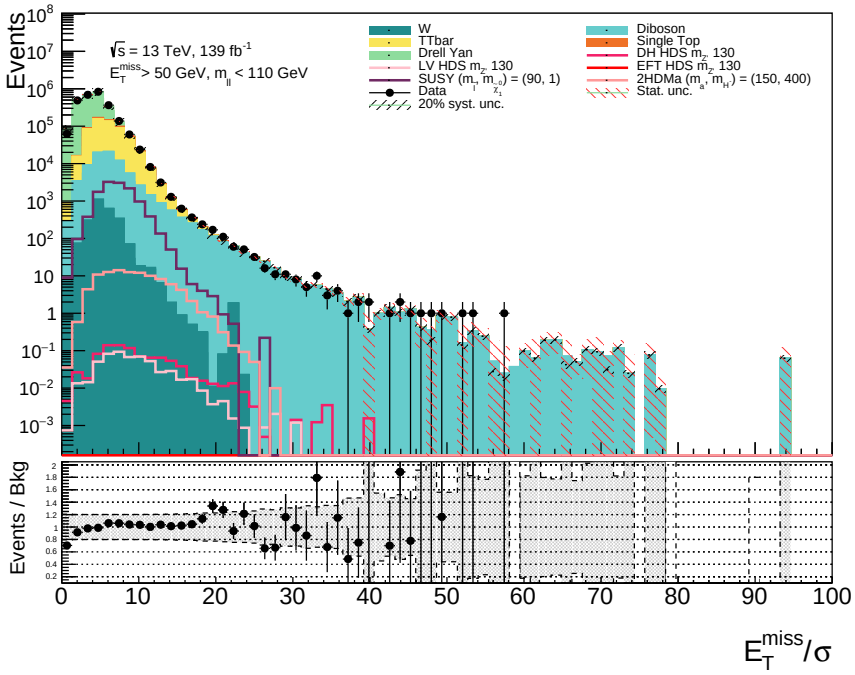


Figure 6.12: Distribution of $E_T^{miss,sig}$ in control region. For the 2HDM + a we only include the distribution for $\tan\beta = 1$.

To keep the trend of variables for invisible particles, we will also study the transverse mass, m_T defined in Eq. (4.10), and stransverse mass, m_{T2} defined in Eq. (4.11). The distribution of the stransverse mass can be seen in Figure 6.13. Moving on to jet related variables, we will count the number of b- and light jets with $p_T \geq 30, 40$ GeV respectively. As the jet reconstruction is tricky, especially when requiring a MET cut. In Appendix D we can see how the data and MC simulations of the SM agree when counting the number of b- and light jets with a p_T cut of ≥ 30 GeV and ≥ 40 GeV respectively. The agreement with different p_T cuts can be seen in Appendix D. We will also look at the p_T, η and ϕ of the three most energetic jets, as these have the best MC and data agreement, and the invariant mass of the two most energetic jets m_{jj}.

Another variable we will look at is the hadronic activity, H_T defined in Eq. (4.6), from which we can also get the ratio between the MET and hadronic activity, E_T^{miss}/H_T.

To know the distance between the lepton and MET the difference in azimuthal angle is studied between: the lepton pair, $\Delta\phi(l_1, l_2)$, the dilepton system and MET, $\Delta\phi(ll, E_T^{miss})$,

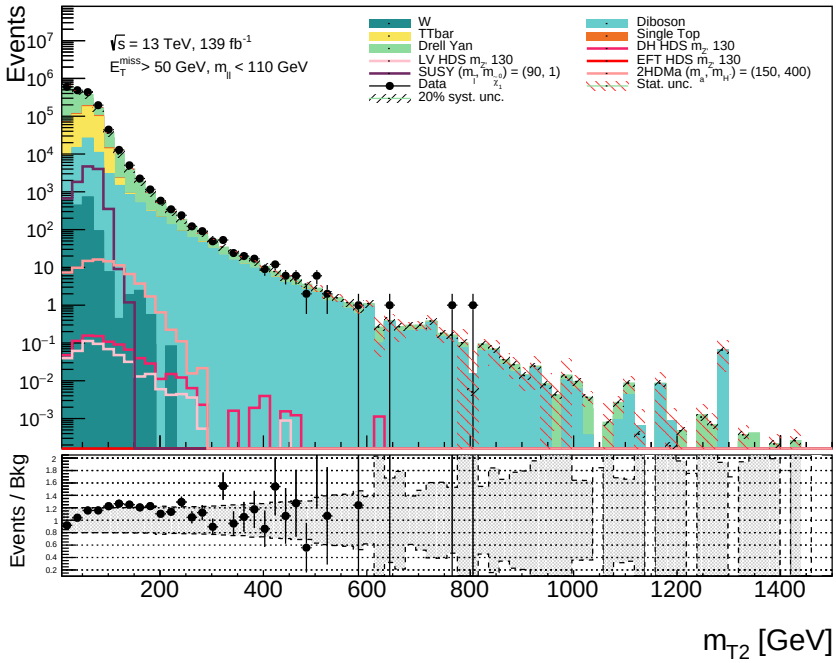


Figure 6.13: Distribution of m_{T2} in control region. For the 2HDM + a we only include the distribution for $\tan\beta = 1$.

the leading lepton and MET, $\Delta\phi(l_l, E_T^{miss})$, and the lepton closest to the MET and the MET, $\Delta\phi(l_c, E_T^{miss})$. In some of the models we are studying it is expected for DM and the lepton pair to be back to back, the distribution of $\Delta\phi(ll, E_T^{miss})$ shown in Figure 6.14.

See if you have examples

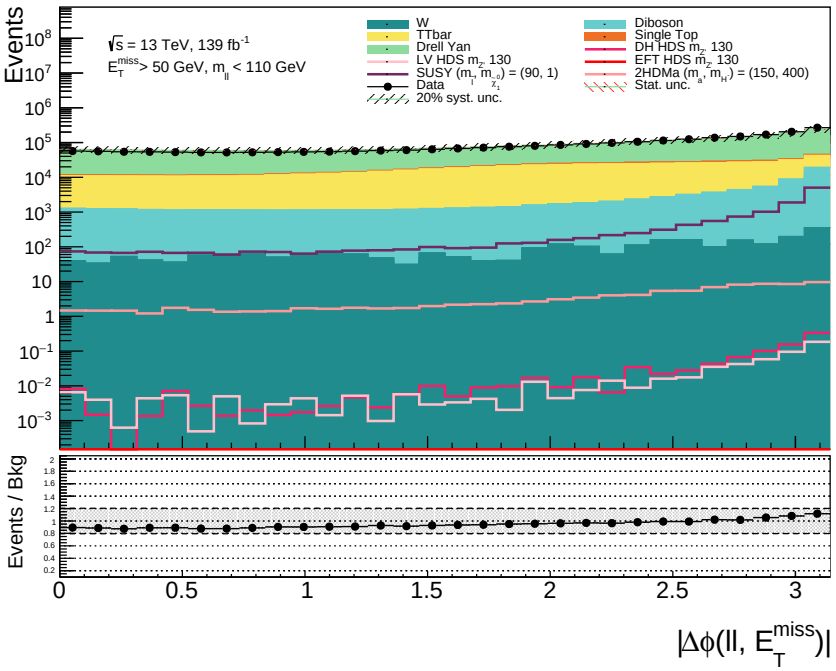


Figure 6.14: Distribution of $\Delta\phi(ll, E_T^{miss})$ in control region. For the 2HDM + a we only include the distribution for $\tan\beta = 1$.

The kinematic variables used are summarized in Table 6.3 and the distribution of the remaining kinematical variables are shown in Appendix C.

Table 6.3: Table showcasing the kinematic variables that will be used as features on our ML algorithm.
*The variables marked with * have poor MC and data agreement. The variables marked with † create jagged arrays. What this means and what we will do with this is explored in Section 6.5.1 and 7.2.1, respectively.*

Kinematic variable	Feature name
p_T of both leptons	lep1pt & lep2pt
ϕ of both leptons	lep1phi & lep2phi
η of both leptons	lep1eta & lep2eta
Invariant mass of dilepton pair, m_{ll}	mll
Missing transverse energy in event, E_T^{miss}	met
Missing transverse energy significance in event, $E_T^{miss,sig}$	met_sig
Transverse mass in an event, m_T	mt
Stransverse mass in an event, m_{T2}	mt2
Transverse energy of lepton pair, E_T	et
ϕ between lepton pair, $\Delta\phi(l_1, l_2)$*	dPhiLeps
ϕ between lepton pair and MET jet, $\Delta\phi(ll, E_T^{miss})$	dPhiLLMet
ϕ between leading lepton and MET jet, $\Delta\phi(l_l, E_T^{miss})$	dPhiLeadMet
ϕ between the closest lepton and MET jet, $\Delta\phi(l_c, E_T^{miss})$*	dPhiCloseMet
Hadronic activity, H_T	ht
Ratio between E_T^{miss} and H_T, E_T^{miss}/H_T	rt
Number of b-jets	nbjets
Number of light jets*	nljets
p_T of three jets with highest p_T†	jet1pt & jet2pt & jet3pt
ϕ of three jets with highest p_T†	jet1phi & jet2phi & jet3phi
η of three jets with highest p_T†	jet1eta & jet2eta & jet3eta
Invariant mass of two jets with highest p_T, m_{jj}†	mjj

6.5.1 MC and data disagreement

There is a challenge, with the final states that are not e^+e^- or $\mu^+\mu^-$ (SFOS), as the MC generated background tends to be lower than the recorded data. The number of events that are not SFOS are minimal though[4], and we think the reason it does not fit the data is because we are not including fake leptons[5]. The features where these are prominent are marked by a * In Table 6.3. As we want the MC simulations of the background to

MAKE THE FIGURES

[4]For all kinematical variables divided into their respective dilepton final state see the GitHub repo, available here:

[5]Fake leptons refer to reconstructed particles that appear to be leptons but are actually misidentified hadrons or non-prompt particles.

agree as much as possible, we will not use the features marked with a * when using our ML algorithms.

6.6 Transfer to ML-friendly syntax

We have so far explained how the data will be used and carefully selected, however the question of how the data is made, and the number of samples we will work with still remains unanswered.

The MC simulations are made available in `ROOT` [48] NTuples which contain the information of the objects passing the selection criteria in each event. What remains is to set the kinematical cuts, so we have our preselection region. To do this, as well as saving the remaining events into a new file which will be fed into the ML algorithm, we utilized the algorithms on `EventSelector`[6], which also saved the events that passed the event selection criteria as `ROOT` histograms (to make plotting the distributions easier). After saving the events that passed the selection criteria we used the algorithms on `DataPrep`[7] to plot the actual distributions of kinematical variables, and more importantly converting all the events that passed the event selection into an ML-friendly syntax. For this thesis we are converting from `ROOT` NTuples to `pandas DataFrame` [49] which can furthermore be saved as `h5` files to be read more efficiently. The reason we chose `pandas DataFrame` is because of the easily readable kinematics per event, the easily applicable kinematical cuts to the whole dataset (to more effectively create signal regions), and most importantly because of the compatibility with `XGBoost` [40] and `TensorFlow` [35] which will be the ML packages we will utilize for both BDTs and NNs. With this we are ready to discuss the preparation of our ML algorithms.

[6] Available here: https://github.com/rubenguevara/Master-Thesis/tree/master/EventSelector
[7] Available here: https://github.com/rubenguevara/Master-Thesis/tree/master/DataPrep

Chapter 7

Machine Learning preparation

We have now presented the dataset and theory, and are thus ready to start making our ML algorithms. The first part of this chapter will be a presentation of the dataset in more details, included all the events and the *weights* variable. We will also explore how we divide our dataset into a training and testing set. Afterwards we will discuss the crucial task of optimizing the architecture and hyperparameters of the ML models for obtaining high-performance classifiers. In this field, the challenge is often to distinguish between signal events from rare physics phenomena, and the much more common SM background processes. Since the signals events are extremely rare, the classifiers need to be highly optimized in order to effectively separate them from the background. In addition, the datasets have numerous features, which can lead to overfitting or poor generalization if not properly optimized. Another hardship is how to mitigate the phenomena of *jagged arrays* and missing variables.

It is essential to carefully choose the model architecture and hyperparameters, as well as the dataset pre-processing techniques, in order to achieve the best possible performance in the search for the rare DM physics phenomena. This can involve optimizing parameters such as the learning rate and the regularization strength. Exclusively for the NNs we have the number of layers and the number of neurons per layer. Exclusively for the BDTs we have the number of trees and the tree depth. The second and third part of this chapter will be just about optimization for the NNs and BDTs respectively.

7.1 The datasets

The way we are making datasets in this project, to keep it as close to model independent as possible, will be of the following format

- Make a dataset with all the SM backgrounds and one DM model.
- Make a dataset with all the SM backgrounds and all the DM models, divided into different signal regions and statistically combine the results of each signal region

The idea behind the first one is to start with a model dependent approach, where we train an ML algorithm to learn just one model at a time. The reason for taking this approach, is because it is still closer to being more model independent than normal ML HEP searches, as these usually train on the whole SM background dataset with just one signal model with fixed masses and parameters. While what we would be doing is combining all the different masses and parameters into one dataset for each model.

For the second one, if we created Signal Regions (SR) in kinematically orthogonal regions, then some models might be more important than other in each region. Meaning that the approach would teach the ML algorithm the signatures of the DM physics in the final state, rather than making an algorithm that is good at learning one individual model. Furthermore, the plan is to combine the results of each respective region to get an overall view of all the signal regions studied.

In this project we will look at real data from the ATLAS detector from all running periods of Run II. This is from period 2015-2016, 2017 and 2018 each with an integrated luminosity of $36.4, 44.3$ and $58.5\ fb^{-1}$, respectively. In Table 7.1 we list the overall number of corresponding MC simulated events for each process, together with the expected events (i.e. scaled to the relevant cross-section and integrated luminosity), for all SM background and the as DM models considered. In addition to using the features in Table 6.3, we will also include the EventID, dataset ID (DSID), period, dilepton final state, label telling us whether an event is signal or background and the *weight* of each event. From this dataset we can see one of the main challenges that will follow throughout the thesis, especially with the model dependent approach. This is the unbalance of the dataset, as an example lets look at the DH HDS model using the model dependent

Table 7.1: Table showcasing the number of simulated events and expected events for every SM background channel and DM model that will be used in this thesis.

Channel	Simulated Events	Expected events for 139 fb^{-1}
W	38,684	5,436
Drell Yan	47,201,697	2,198,258
TTbar	28,126,697	978,531
Single top	729,624	93,899
Diboson	10,983,689	116,654
Standard model total	87,080,391	3,392,777
Z' Dark Higgs Heavy Dark Sector	498,621	54.92[1]
Z' Dark Higgs Light Dark Sector	892,365	155.37
Z' Light Vector Heavy Dark Sector	521,759	39.82
Z' Light Vector Light Dark Sector	470,352	235.70
Z' Effective Field Theory Heavy Dark Sector	715,409	0.0004
Z' Effective Field Theory Light Dark Sector	640,963	0.0007
Supersymmetric direct slepton production	-	-
2HDM + a	-	-

approach. Here we would train using all 87,080,391 simulated background events and 498,621 simulated signal events, as the numbers show, the majority of the dataset consists of background events, making the task of learning harder for any ML algorithm. This becomes even harder when applying the weights used for re-weighting simulated events to expected events at 139 fb^{-1}. What these weights used are is the subject of the next section.

7.1.1 Weights

The weights are crucial for most of what is to come further in this thesis. The weights only apply to MC simulations and can be interpreted as corrections to the simulations. The weights are defined as Eq. (7.1)

$$W = w_{mc} \times w_{pu} \times w_{xs} \times w_{lsf} \times w_{nlo,ew} \times w_{ttbar} \times w_{jet} \times w_{bjet} \times \frac{\text{lumi}_{period}}{\sum w_{mc,DSID}} \tag{7.1}$$

This equation has a lot of information in it, so every part of this will be discussed. Starting from the MC weight, w_{mc}, which as stated in [47] gives the relative probability of an event within a sample, the pile-up weight w_{pu}, is a weight used to correct object reconstruction as it can be possible for multiple objects to cross on the identification process. [50]

The cross-section weight w_{xs}, defined as $w_{xs} := xs \times w_{kf} \times g_{eff}$ where xs is the cross-section of the process, w_{kf} is the $k-$factor which tries to take into account the higher-order QCD corrections, for more details we refer the reader to the study done by Catani et al. [51]. g_{eff} is the filter efficiency, which is the expected fraction of MC events that pass through the acceptance criteria.

The Lepton and trigger Scale Factor (LSF), w_{lsf}, which are applied in order to correct for differences in reconstruction in MC with respect to data, the Next to Leading Order (NLO) electroweak correction $w_{nlo,ew}$, which corrects for higher order EW diagrams, for more details we refer the reader to the paper by Denner et al. [52].

The w_{ttbar}, w_{jet} and w_{bjet}, which are reconstruction corrections for $t\bar{t}$ [53], jets [54] and b-jets [55], respectively. And lastly the luminosity of the period, meaning

$$\text{lumi}_{period} = \begin{cases} 36.4 \text{ fb}^{-1}, & \text{for period} = \text{2015-2016} \\ 44.3 \text{ fb}^{-1}, & \text{for period} = \text{2017} \\ 58.5 \text{ fb}^{-1}, & \text{for period} = \text{2018} \end{cases}$$

divided by the Sum Of Weights (SOW), $\sum w_{mc}$ of every sample for each dataset ID.

7.1.2 Signal regions

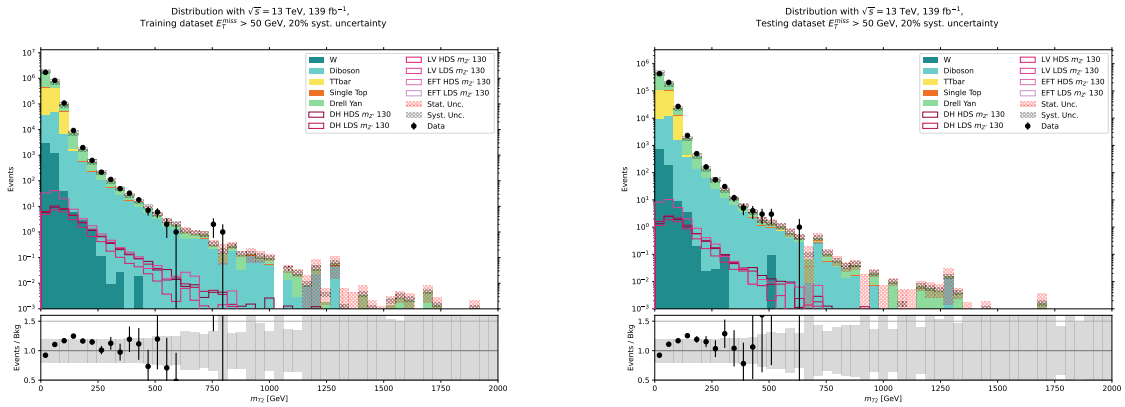
As a first step towards achieving less model dependence we will create three Signal Regions (SR). The goal is then to train one network in each respective SR that contains all the DM models we wish to study. The SR we will study are defined below

- SR1: $m_{ll} > 110$ GeV and $E_T^{miss} \in [50, 100]$ GeV
- SR2: $m_{ll} > 110$ GeV and $E_T^{miss} \in [100, 150]$ GeV
- SR3: $m_{ll} > 110$ GeV and $E_T^{miss} > 150$ GeV

where the m_{ll} cut is made such that we remove the $Z-$peak before training and where the MET regions are orthogonal. The goal of having kinematically orthogonal SRs when training is to allow the combination of the results.

7.1.3 Train and test split

When training our networks we will use 80% of the dataset, the remaining 20% will be used to test whether the networks are learning any physical patterns or just getting really good at guessing events in the given dataset. When splitting the datasets however, we need to be absolutely certain that the distributions follow the same shape and have the same ratio of background channels as well as having good MC and data agreement in the training and testing data sets. In this thesis we will utilize a function from `scikit learn` [56] called `train_test_split` which can easily split our dataset into a training and testing set of a chosen size, while also splitting the label of the dataset, meaning whether the event is signal or background, in the same manner.



(a) Training distribution *(b) Testing distribution*

Figure 7.1: Distribution of the stransverse mass when dividing SM background and DM samples into 80% training and 20% testing datasets. The integrated luminosity for each distribution is 80% and 20% of 139 fb^{-1} respectively. We also included an estimated 20% flat systematic uncertainty to the simulated events.

In Figure 7.1 we can see that the distribution has not been altered when we split the dataset into 80-20. The data and MC agree reasonably in both datasets, meaning the distributions are split correctly. To see the distribution of every other kinematical variable see the GitHub repo[2].

[2] Available here: https://github.com/rubenguevara/Master-Thesis/tree/master/Plots/Data_Analysis/train_test_split

7.2 Neural Network Training

For this thesis we utilize `TensorFlow v. 2.7.1 GPU` for all NNs. After creating the dataset, the first and most important thing is to marinate the `batch_size` whenever we try anything while using the dataset. This is to optimize wrt. to both the size of the dataset and the imbalance between signal and background events.

The highest possible batch size that could be used for this thesis was 2^{22},[3] leading to roughly 4 million simulated events per batch. This is the best that a dedicated GPU, `NVIDIA A100-PCIE-40GB`, could handle. The batch size also decreases the more complex the NN becomes, as this requires greater computational power. If we reduce the batch size for any of the tests we will explicitly mention what we changed it to.

With this out of the way there are still a few challenges to mitigate to optimize our NN. These are described below.

7.2.1 Padding of data

There are two difficulties to overcome when utilizing NNs as compared to BDTs. The hardest to solve, as there is no general solution yet, is the padding of *jagged arrays*. As indicated in Table 6.3 with the †, we will record events with up to three jets in the final state. However, there are not always three jets in the recorded events that we will be studying, this creates jagged arrays which we can interpret as arrays with missing values. This is an unwanted feature that we need to avoid when training NNs.

To mediate this problem we chose to set the p_T to zero for the missing jets and m_{jj} to zero if there are less than two jets, this is something that is physically reasonable as it does not violate any conservation laws. More problematic however is the η and ϕ when there are no jets. To mediate this we have set the values to -999, which has no physical meaning and is impossible to achieve, this we did so it becomes easier for us to identify the jagged arrays that need different interventions.

[3] It has to be a power of two because of the alignment of virtual processors (VP) onto physical processors (PP) of the GPU. As the number of PP is often a power of 2, using a number of VP different from a power of 2 leads to poor performance.

As mentioned before, there is no consensus on how the padding should be done, and there are many methods of doing so. The classical data scientist way of solving this problem would be to just take the mean of every feature and use that as a variable for every event with missing values. That means replacing every $p_T, m_{jj} = 0$ and $\eta, \phi = -999$ with the mean of every p_T, m_{jj}, η and ϕ (excluding the 0's and -999's respectively). However, this is not popular among physicists since it breaks conservation laws when we say there are jets present in an event when in reality there are none. Another approach is to use Bayesian statistics or ML to estimate the missing values, these options will not be pursued in this thesis due to time constraints, but might be of interest for future projects. Another approach, is setting all the missing values to zero, as this might mean that there is not anything there, but this also breaks conservation laws since $\eta, \phi = 0$ have physical meaning, this is also highly looked down upon by data scientists since this could affect the weighting when training the network and potentially create a false pattern for the network to follow which would lead to a bias.

The jet p_T and m_{jj} being 0 is a valid form of padding the dataset, as this does not break any fundamental law of physics. However, setting ϕ as something outside $[-\pi, \pi]$ does not make much sense as this is the angle around the detector. Setting a high value of $|\eta|$ might be possible in principle, but as of today the ATLAS detector has a $|\eta| < 4.9$ (see Chapter 4.2) as the pseudorapidity states how close to the beam line the objects recorded are. However, having a p_T of a jet equal to zero while still recording the η and ϕ breaks the laws of physics, so this is a problem that needs to be fixed.

Another approach is to remove the features with missing values to conserve statistics, albeit make it harder for the network to see any pattern that we might miss, but this is also not a desirable mitigation. Instead, we have tried to limit ourselves to the use of number of jets of defined categories that work around the need of padding. These features are just *event features*, meaning that we count the number of jets that fulfill some criteria, such as the number of b-jets with $p_T > 20$ GeV, the number of light jets with $p_T > 40$ GeV, the number of jets recorded in the central calorimeter ($|\eta| < 2.5$), and the number of jets with $p_T > 50$ GeV recorded in the forward calorimeter ($|\eta| > 2.5$).

The p_T cuts are optimized to allow a good agreement between data and simulations, the full distributions with different cuts in the preselection region can be seen in Appendix E. The padding variables included in the training are shown in Table 7.2.

To come

Table 7.2: Table showcasing features that will not need padding.

Kinematic variable	Feature name
Number of b-jets with $p_T > 20$ GeV	n_bjetPt20
Number of light-jets with $p_T > 40$ GeV	n_ljetPt40
Number of jets recorded in Central calorimeter	n_jetsetaCentral
Number of jets recorded in Forward calorimeter with $p_T > 50$ GeV	n_jetsetaForward50

When training our NN with these new variables instead of dropping features with missing variables we hope that the NN learns more physics by hopefully recognizing patterns between all high level features. Having addressed the challenge of jagged arrays, we will continue to address the second step to prepare the dataset for the NN, the normalization procedure.

Should I link to the results?

7.2.2 Normalization of data

Since neural networks send a lot of data into multiple neurons and multiple layers using activation functions and carrying weights and biases that change for every back propagation iteration, it is important to make sure that a neuron output does not vanish when moving around the network. Meaning that a neuron output becomes negligible compared to others when navigating the loss-phase space. A fast way for a neuron output's to vanish is to not normalize the data and send it through the network as it is available. The reason it might vanish, is because we send in numbers which vary significantly from each other, i.e. the p_T might be as high as thousands GeV, while E_T^{miss}/σ might be as low as 0.1. What might happen when sending such different numbers is that the network might think "obviously the high number is more important than the low number" thus making the activation function worse for the feature, even though this feature is of high importance when looking at MET final states. A way to fix this problem is to normalize all features. There are many ways to do this, one could do *min max scaling* which normalizes every feature from $[0, 1]$, completely solving the problem above. Mathematically

speaking this is done by

$$X_{norm} = \frac{X - X_{min}}{X_{max} - X_{min}} \tag{7.2}$$

Where X is the array containing all events for a given feature, while X_{min} and X_{max} are the lowest and highest values in the said array. Another way to normalize the data is to make the mean of every feature such that the mean of all the features is distributed according to a standard normal distribution (mean 0 and std. 1), this is called *Z-score normalization*

$$X_{norm} = \frac{X - \bar{X}}{\sqrt{\sigma_X^2}} \tag{7.3}$$

where $\bar{X}$ is the mean of said array and σ_X^2 is the variance. One could also use pre-built functions in `TensorFlow` that provide a normalization, such as `Batch_normalization` of the data entering the network per batch. This is usually used in Convolutional NN's as it improves computational speed. Another one, `Normalize`, provides the same as Eq. (7.3) for the whole training set going in. This is however computationally heavy to use. The `Layer_normalization`, which normalizes the activations of the previous layer in a batch *independently*, rather than *across* a batch like `Batch_Normalization`. Both `Batch_Normalization` and `Layer_Normalization` use an optimized version of the Z-score when normalizing the data, meaning that all the features take are normalized following a standard normal distribution.

There is a big difference when normalizing data ourselves or using `TensorFlow`. `TensorFlow` remembers how the data was normalized when training such that the test data will be normalized the same way, making testing easier. While if we use Eq. (7.2) or Eq. (7.3) ourselves, we have to make sure that we use the same values for $X_{max}, X_{min}, \bar{X}$ or σ_X^2 when normalizing the test data.

7.2.3 Balancing of signal and background

The biggest challenge to overcome in this thesis is what we should use as sample weights (see Section 5.3.2). If we were to not use any form of sample weights to mitigate the imbalance in our data set it could potentially lead to *majority class classification* where the networks could get "lazy" and guess that everything is background.

To combat the majority class classification, we will as mentioned make use of the sample weights. We will study three cases

1. Unweighted training, meaning that we will be setting the sample weight to one
2. Weighted training, meaning that we will be setting the sample weight to the weights used to re-weight MC events to the expected events as explained in Chapter 7.1.1
3. Balanced training, we will weigh the background and signal samples in an attempt to balance the number of events available in each of the two categories,

where the latter would in terms of weights mimic a 50-50 distribution of signal and background.

7.2.4 Re-weighting MC to expected events

Even if the weighting method previously described might help the NN give us better results, we also want to include the weights used to re-weight MC events to expected events. This is desirable in the sense that we want to show our ML networks the true kinematical distributions of each feature. As the re-weighting weights are generated with simulation corrections as well as luminosity and cross-section in mind, it is heavily desirable to also apply these corrections when training our networks, such that it can correctly make predictions in the test dataset regardless of their weight. This is particularly crucial when using real data on our predictions, as these have no weights.

Ideally one would take into account both the data imbalance between the signal and background as well as the re-weighting weights when training a network. To do this using `TensorFlow` we could make use of two parameters when training the network: `class_weight` and `sample_weight`. The `class_weight` works as a dictionary that weighs events differently, based on a dictionary key, for our purposes this key would be whether the event is a signal or background event. The `sample_weight` takes in individual weights for every single event that goes into the network, meaning that it is crucial that we know that the desired weight matches the desired event. Ideally we would use both weighting methods, `class_weight` to balance the signal to background ratio and `sample_weight` with the re-weighting weights that include the cross-section. However, there is a bug

in `TensorFlow` (up to version `GPU 2.7.1`) that prevents the program to run when using both parameters. This is not a big problem though, as when looking at the source code [57] one can see that what `TensorFlow` does with both weights is multiply them together. This means that to mitigate this problem we will use element-wise multiplication between the re-weighting weights array and the balancing weights array to create a new array that will be used as `sample_weights`.

For this thesis we tried testing four different methods to make the sample weights. For the re-weighting array, all background events will be re-weighted to expected events using the weights from Section 7.1.1 while the signal events will have a value of one. The balancing array was made by expanding the balancing ratio from Section 7.2.3 to be

1. $\frac{N_{sig,MC}}{N_{bkg,MC}}$ where we weigh down all background events wrt. the ratio of total simulated signal events over the total simulated background events
2. $\frac{N_{bkg,MC}}{N_{sig,MC}}$ where we weigh up all signal events wrt. the ratio of total simulated background events over the total simulated signal events
3. $\frac{N_{sig,MC}}{N_{bkg,exp}}$ where we weigh down all background events wrt. the ratio of total simulated signal events over the total expected background events at 139 fb^{-1}
4. $\frac{N_{bkg,exp}}{N_{sig,MC}}$ where we weigh up all signal events wrt. the ratio of expected background events at 139 fb^{-1} over the total simulated signal events,

and one if we do not weight the event. Note that we are not re-weighting the signal events to expected signal events when training because this would in principle remove all expected events (due small cross-sections) from the NN as there are so few events (see Table 7.1). Taken all of these factors into account what remains is to choose a network architecture and which hyperparameters to use to best fit our task.

7.2.5 Architecture and hyperparameter tuning

The architecture of the NN utilized in this project is of the form shown in Figure 5.1, where a NN with an arbitrary number of layers is shown, each of the hidden layers use the ReLu (Eq. (5.5)) activation function, and where the output is in the form of a single neuron using the sigmoid activation function (Eq. (5.4)). An algorithm showing one possible way to create this NN can be seen in Appendix B.1. To get the best performance on our NN, we need to find which hyperparameters help the network reach the highest significance. To do this, we need to do a grid search. For our neural network we will mainly focus on four hyperparameters explained in section 5.1:

- The learning rate η
- The L2-regressor variable λ
- The number of neurons on each hidden layer `n_neuron`
- The number of layers `n_layers`, excluding the output. Meaning that `n_layers` = 1 means no hidden layer

The metrics that will be used to estimate the best hyperparameters are Area Under the curve (AUC), binary accuracy and most importantly the expected significance[4]. The expected significance for this section has been calculated using the low statistics formula Eq. (4.19) without uncertainties. The expected significance will be calculated by making a lower cut of 0.85 on the network prediction score, as explained in Chapter 5.3.5. For some models where the expected events are too low, the expected significance will be 0 or `NaN`. If this happens we will use the AUC on the testing set as a metric to find the best hyperparameters. This procedure will be performed for every single network we will explore. The results of the best method used to optimize NNs is showcased in Section 7.4, and a more in depth showcase of the methods can be seen in Appendix A. In the next section we will discuss the optimization of BDTs.

[4]See Chapter 5.3 for explanation of metrics.

7.3 Boosted Decision Tree Training

When working with BDTs there are not as many challenges to overcome as with NNs. For example the padding and normalization of data can be completely avoided, making the whole procedure a lot easier when one uses challenging dataset as we do in high energy physics. The weights are still an obstacle that we need to overcome when using BDTs. This will be discussed in Section 7.3.1.

For this project we will, as mentioned utilize, the eXtreme Gradient Boosting, or `XGBoost` for short, package [40] made for the Higgs ML Challenge [39] whenever we mention BDTs. This project utilized version `1.5.0` without GPU adaptability. `XGBoost` also helps to avoid padding as it is integrated with a `missing_variable` variable where we can simply write the number of the variable that is missing. For the dataset in this project this implies setting the `missing_variable` value to -999,

7.3.1 Sample weights

For `XGBoost` there is a different problem when it comes to weights. XGBoost has a variable called `scale_pos_weight` where we can help the network deal with imbalanced data, such as the one we have. Meaning that the whole problem of combining the re-weighting weights with the balancing weights from Chapter 7.2.4 completely disappears. Albeit there is a caveat, `XGBoost` does not have the possibility to include negative weights, which the datasets explored in this work have. The reason `XGBoost` does not include negative weights, is because when calculating the number of events in a leaf node, which is made by taking the sum of sample weights, we cannot have a negative value [58, 59]. As for the MC generators, Sherpa [45] takes into account higher order diagrams and needs to add negative weights to "counter" the over counting of diagrams [60], which are important to correctly scale the simulated events to real data.

A method to mitigate this problem is to use the absolute value of the weights when training. This is however not generally accepted as a solution, and some even say it should be avoided. There are other options however, one of these options is to not include events with negative weights in the training set. This is an okay thing to do, as we

can imagine that if we were to only include events with positive weights in the training, it might be the same as putting the negative weights in the "testing dataset" (Chapter 7.1.3). However, both methods are equally problematic if the positive and negative weights distributions are not equal for all the features.

Another method that has been used in a published ATLAS article [61] is to normalize the weights when using the absolute value with respect to the sum of weights over the sum of absolute weights. The reason behind this is that the sum of weights is obviously not the same when we take the absolute value. Mathematically speaking, if we have an array of weights W, we can update this like

$$W \to |W| \frac{\sum_i W_i}{\sum_i |W_i|} \tag{7.4}$$

such that the weights are at least on the same scale.

7.3.2 Architecture and hyperparameter tuning

Making a BDT for our purposes is fairly easy as well using XGBoost. One way to do it is using the algorithm shown in Algorithm B.2. To get the best performance on our BDT we have to do a grid search here as well. The trainable hyperparameters here are different from NN's, with XGBoost we will focus on the following hyperparameters

- Tree depth: how many times we split the data
- Number of estimators: how many trees we use to do the gradient boosting
- The learning rate η
- L2-regressor λ, to stop overtraining

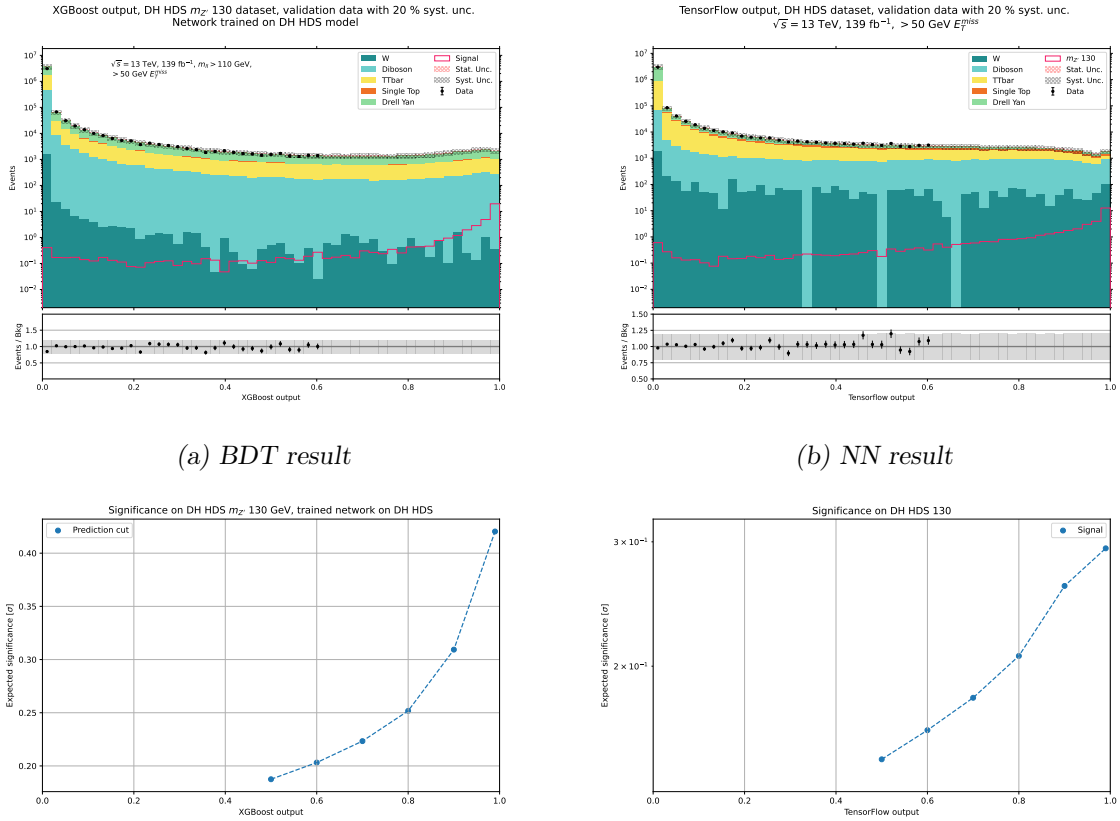
The same procedure for hyperparameter optimizations as described for NNs in Section 7.2.5 will be used for BDTs as well. The results of the best method used to optimize BDTs is showcased in the next section, a more in depth showcase of the methods can be seen in Appendix A.

7.4 Results of optimization methods

We tested the normalization, balancing of signal and background, re-weighting of events using NNs, as well as the use of sample weights with BDTs. In this section of the thesis we show the main result of the network optimization procedure, the full results are detailed in Appendix A. We observed that for the NNs using `Batch_normalization` as the normalization method yielded better results, we also saw that the NN was better at sorting signal from background using ADAM as the optimizer rather than using SGD. Concerning the sample weights, the method that yielded the best results was the one that re-weighted every simulated background event into the expected events with a luminosity of 139 fb^{-1}, and balanced the dataset by weighing up all signal events by the ratio of expected number of background events over simulated signal events, $\frac{N_{bkg,exp}}{N_{sig,MC}}$. We also observed that the new padding method, which counted the number of (light)b-jets with $p_T > (40)20$ GeV, the number of jets in the central calorimeter, and the number of jets with $p_T > 40$ GeV in the forwards calorimeter, performed better than the other method of removing jet features with missing variables.

For the BDT we observed that while conducting a grid search for the optimal hyperparameters, we never hit a plateau where the scores seemed to flatten out when increasing the tree depth. However, we decided to take the conservative approach of having a tree depth of maximum 6. The sample weights we decided to use further with BDTs were the ones that only looked at the positive weights of the dataset, instead of using the scaled up absolute value of the weights. To balance the signal and background events in the dataset we weighed down the background events with the ratio of total simulated signal events over the total expected background events at 139 fb^{-1}.

The best results from the BDT and NN trained using the model dependent approach on the Dark Higgs Heavy Dark Sector model are compared in Figure 7.2.



(a) BDT result

(b) NN result

(c) Significance of (a)

(d) Significance of (b)

Figure 7.2: Comparison of BDT and NN. Both networks were tested on the DH HDM dataset using the model dependent approach. The output on the x-axis on all plots above reflects the score of the ML output ranging from 0-1, where 1 means that the ML network classifies an event as signal and 0 means background. In plot (a) and (b) we can see the distribution of events given the output. In plot (c) and (d) we see the expected significance as a function of the low cut on the output.

Figure 7.2 showcases the ML distribution score from 0-1[5] for every event using both algorithms (upper plots), and the expected significance from the distribution plots as a function of the lower cut on the output. From the results we can see that the BDTs perform better than NNs as the expected significance on the last bin is higher by $\approx 0.13\sigma$. As using BDTs is easier than NNs, as well as BDTs having greater interpretability, we will for the remainder of the thesis opt to only use BDTs as our main ML algorithm. We have now explored the methods that will be used for this search, and all the challenges that need to be mitigated. The next part of this thesis will present the results of both the model dependent and independent approach for every model.

[5]Where 0 means background and 1 means signal

Part III

Results and conclusion

Chapter 8

Results

The time has come to test how our networks fare on learning the eight models with DM candidates we are studying: The three mono-Z' models, Dark Higgs (DH), Light Vector (LV) and light Effective Field Theory (EFT) in the Heavy Dark Sector (HDS) and Light Dark Sector (LDS)[1], the supersymmetric direct slepton production, and the Two Higgs Doublet Model with an additional pseudoscalar a (2HDM + a). As we got better results when using BDTs rather than NNs (see Section 7.4), only the BDT will be followed up.

This chapter showcases the results of both methods we used to train a BDT to classify signal (DM events) and background (SM events). The first method, which we called the model dependent approach, which trains one BDT for each model we have, including all the mass points and simulated samples as seen in Section 6.2.

The second method, which we called the model independent approach, where we split the dataset in orthogonal Signal Regions (SR). The SRs we studied are

- SR1: $m_{ll} > 110$ GeV and $E_T^{miss} \in [50, 100]$ GeV
- SR2: $m_{ll} > 110$ GeV and $E_T^{miss} \in [100, 150]$ GeV
- SR3: $m_{ll} > 110$ GeV and $E_T^{miss} > 150$ GeV

To test how well the networks have learned the models we will create a mass exclusion limits for every model. We decided to make a cut of $m_{ll} > 110$ GeV for the mono-Z'

[1]Where HDS and LDS denotes the assumed DM masses as shown in Table 6.1

models when using the model dependent approach to reduce computational time.

To compare the model independent approach to the model dependent approach we will statistically combine the mass exclusion results for each SR for each final state channel. To navigate through the process we will look at the Dark Higgs Heavy Dark Sector model, using the model dependent method, as the model independent method does the same with three SRs. Afterwards we will present the combined exclusion limits for all eight models.

8.1 Model dependent approach

We conducted a grid search to optimize the BDT and trained the network using 80% of the SM background samples and 80% of every different Z' mass sample of the DH HDS model. When training the BDT on the DH HDS model we used the feature importance feature to see which kinematical variables were most important for the BDT in categorizing signal (DM events) from background (SM events), this is illustrated in Figure 8.1.

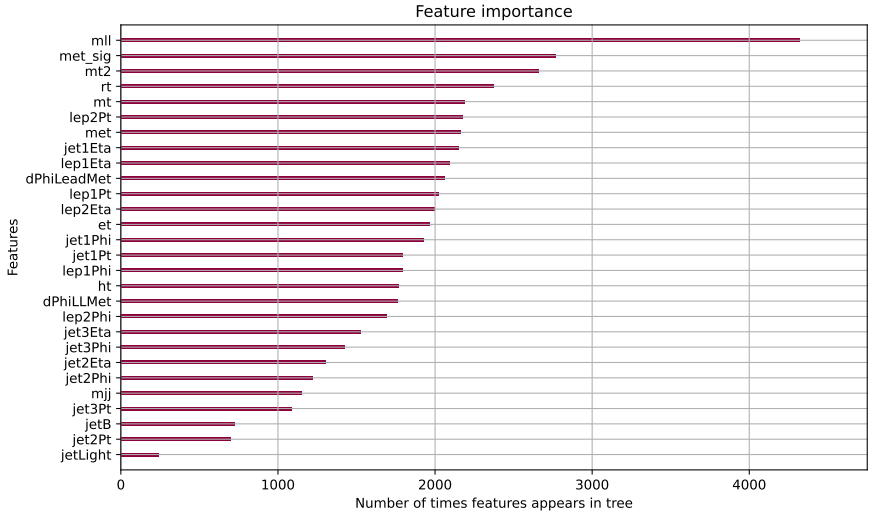


Figure 8.1: Feature importance for network trained on Z' DH HD model using the model dependent approach using the weight metric, which shows how many times a feature appears in a tree. The features in this plot use the labeling presented in Table 6.3. To see the feature importance plots with the cover metric, which show the number of samples affected by the split using the feature, and gain metric, which measures the relative contribution of a feature to the network's performance, see Appendix F.

From Figure 8.1 we can see that the most important kinematical variable for the BDT in classifying DM signal from the DH HDS model from SM background was the invariant mass m_{ll}, this is what we expect as the DH HDS model is based on a mono-Z' theory including a new Z' boson decaying to two leptons, and this includes a resonance on the mass of the Z' as explained by the Breit-Wigner resonance in Eq (4.13). Furthermore, we see that the three following features are all MET related, which is how we predict to measure the presence of DM. These are the MET-significane, $E_T^{miss,sig}$, the stransverse mass m_{T2}, and the ratio between MET and hadronic activity E_T^{miss}/H_T. Something to note of the features in Figure 8.2 is that we use both E_T^{miss} and $E_T^{miss,sig}$ which are correlated, the reason for using both is because the BDT might find that one is better at classifying DM signal from SM background when training on other models.

To showcase how the network categorizes signal from background the validation plots in Figure 8.2 is made on the 20% test samples for some the signal samples, together with the SM predictions and data. The data in the plots will not be shown for an output score of 0.6 or greater as this is the signal region, and we need to go through an unblinding procedure to show data in this region.

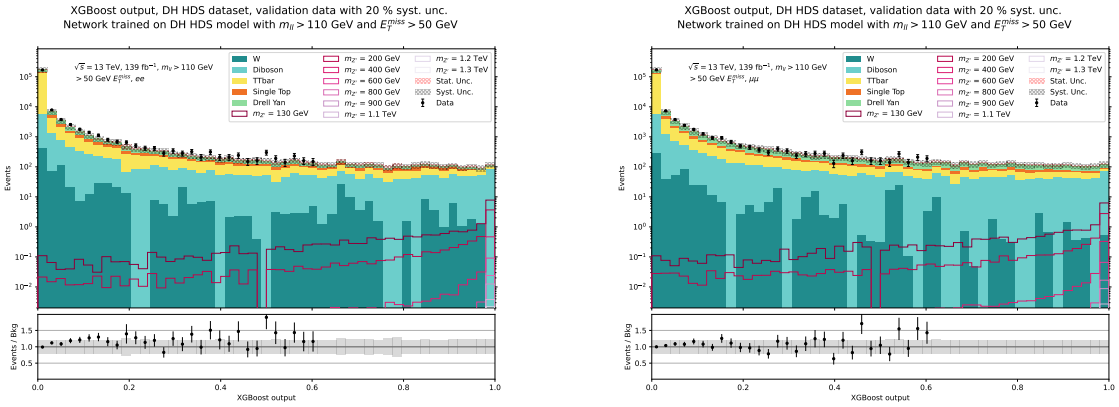


Figure 8.2: Validation plots for network trained on Z' DH HDS model using the model dependent approach. The validation plots show the distribution of the BDT score for every event from 0-1, where 1 is signal. On the left we have the results for the ee channel, and on the right we have the result for the μμ channel. Both results were trained by the same network consisting of a general dilepton final state.

While the validation plot might indicate that the network is not doing an impressive job at sorting signal from background, we can actually see how well the network learns the

different mass points on the DH HDS model by looking at the ROC curves for each mass point, shown in Figure 8.3 for the *ee* channel (left) and the $\mu\mu$ channel (right). Here we see that the network struggles a tiny bit to learn the DH HDS models with the lowest $m_{Z'}$, but the AUC score is still 0.96, meaning that it actually learned to sort the signal from SM background.

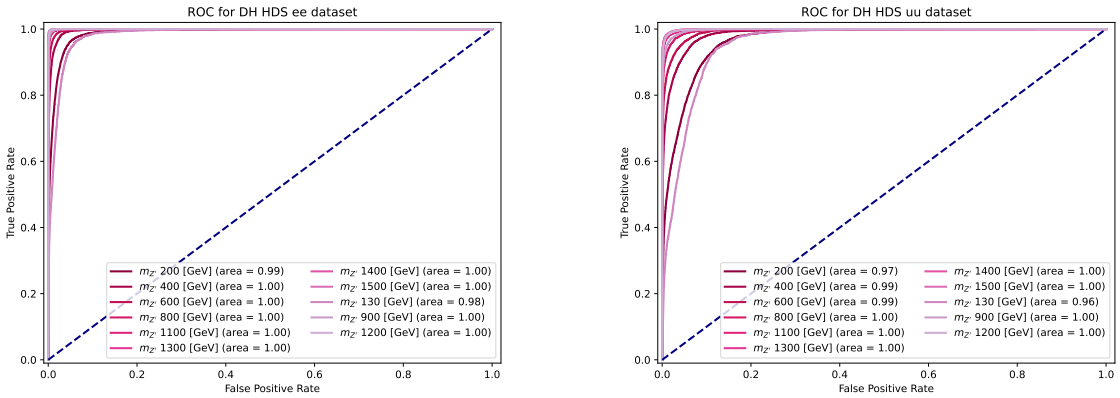


Figure 8.3: ROC plots for every Z' mass point on network trained on Z' DH HDS model using the model dependent approach. On the left we have the results for the ee channel, and on the right we have the result for the $\mu\mu$ channel. Both results were trained by the same network consisting of a general dilepton final state. The dashed blue line showcases the score of a random guess (AUC = 0.5)

To define the region we will use to set any expected exclusion limits, we can choose the validation plot bin that gives us the highest expected significance. In Figure 8.4 we show the calculation of the expected significance as a function of the lower cut on the BDT output for a few mass points.

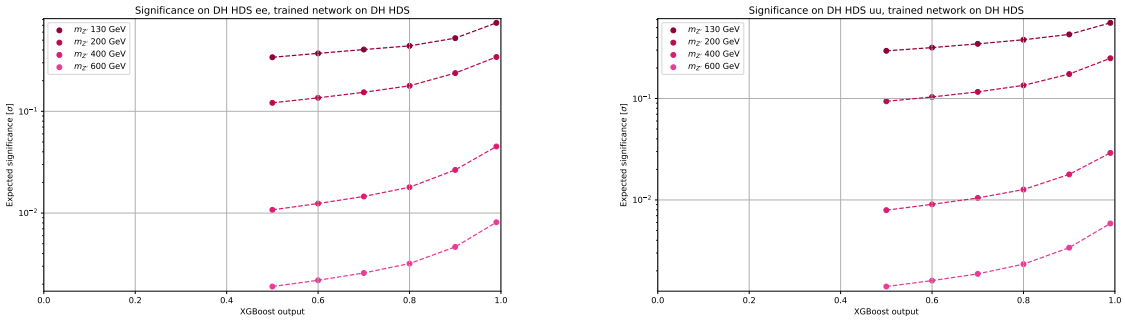


Figure 8.4: Expected significance plots for Z' mass points as a function on the lower cut on the BDT output on BDT trained on the Z' DH HDS model using the model dependent approach

In Figure 8.4 we see that we get the best expected significance of approximately 0.7σ

(without uncertainties), even though the network got an AUC score of 0.96 for the mass point on the muon channel. From this we see that even with an AUC of 0.96 we do not manage to exclude the model.

As the last bin of the BDT output on the validation plots scores the greatest expected significance for every $m_{Z'}$, we can effectively make a cut based on the BDT score. Counting the number of events in this last bin, as well as their uncertainties we can calculate a mass exclusion for both the ee and $\mu\mu$ channels. Utilizing Bayesian statistics with the signal+background hypothesis, we can use the values of ε_{sig}, N_{bkg} and σB from Table 8.1 for each mass points for each leptonic channel, to make a 95% CL expected exclusion.

The mass exclusion limits can be seen in Figure 8.5 for the ee (left) and $\mu\mu$ (right) channel for Z' Dark Higgs Heavy Dark Sector model using the model dependent approach. The y-axis of both plots represents the cross-section times branching ratio of the process we are studying. The x-axis is the mass of the Z' boson. We did not interpolate between the available masses we had simulated, and have rather just connected the values calculated for each mass point in Table 8.1 by connecting the points. There we see the expected 95% CL limit using the values of from Table 8.1 with a 1σ and 2σ deviation. Included in the plots we show how the exclusions look when varying the value of the lepton coupling g_l between the leptons and the Z' boson. The simulated events in this thesis utilized the value $g_l = 0.01$, we increase this coupling to 0.05 and 0.1 to see how the exclusions change.

We see that we cannot exclude any mass point of the DH HDS model (as the dashed red lines do not cross the dashed black line) when the lepton coupling is set to $g_l = 0.01$, we can however see that as we increase the lepton coupling, we can exclude more points. The highest mass exclusion in both the ee and $\mu\mu$ channel is roughly at $m_{Z'} = 300$ GeV (meaning we exclude masses where the dashed line is above the expected limit) when setting $g_l = 0.1$

Table 8.1: Inputs for the $Z'h_D \to l^+l^-\chi\chi$ HDS σB calculations. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the predicted number of signal events after the cuts. The last column is the number of background events, N_{bkg}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% flat systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	1.11	ee	0.25 ± 0.05	7.80 ± 1.58	108.4 ± 23.0
		$\mu\mu$	0.20 ± 0.04	6.28 ± 1.27	124.9 ± 26.1
200	2.46×10^{-1}	ee	0.54 ± 0.11	3.67 ± 0.74	114.1 ± 24.4
		$\mu\mu$	0.41 ± 0.08	2.78 ± 0.56	123.2 ± 25.8
400	1.49×10^{-2}	ee	1.13 ± 0.23	$4.67 \times 10^{-1} \pm 9.37 \times 10^{-2}$	107.0 ± 23.4
		$\mu\mu$	0.79 ± 0.16	$3.29 \times 10^{-1} \pm 6.60 \times 10^{-2}$	127.5 ± 26.6
600	2.35×10^{-3}	ee	1.40 ± 0.28	$9.12 \times 10^{-2} \pm 1.83 \times 10^{-2}$	126.3 ± 26.7
		$\mu\mu$	1.01 ± 0.20	$6.59 \times 10^{-2} \pm 1.32 \times 10^{-2}$	126.3 ± 26.3
800	5.43×10^{-4}	ee	1.59 ± 0.32	$2.40 \times 10^{-2} \pm 4.81 \times 10^{-3}$	118.8 ± 25.6
		$\mu\mu$	1.11 ± 0.22	$1.67 \times 10^{-2} \pm 3.36 \times 10^{-3}$	113.2 ± 23.6
900	2.82×10^{-4}	ee	1.60 ± 0.32	$1.25 \times 10^{-2} \pm 2.51 \times 10^{-3}$	119.3 ± 25.7
		$\mu\mu$	1.12 ± 0.22	$8.78 \times 10^{-3} \pm 1.76 \times 10^{-3}$	114.8 ± 24.0
1100	8.40×10^{-5}	ee	1.63 ± 0.33	$3.81 \times 10^{-3} \pm 7.64 \times 10^{-4}$	114.3 ± 24.4
		$\mu\mu$	1.16 ± 0.23	$2.70 \times 10^{-3} \pm 5.42 \times 10^{-4}$	118.6 ± 24.7
1200	4.75×10^{-5}	ee	1.65 ± 0.33	$2.18 \times 10^{-3} \pm 4.37 \times 10^{-4}$	118.4 ± 25.1
		$\mu\mu$	1.14 ± 0.23	$1.50 \times 10^{-3} \pm 3.01 \times 10^{-4}$	125.6 ± 26.3
1300	2.73×10^{-5}	ee	1.69 ± 0.34	$1.28 \times 10^{-3} \pm 2.57 \times 10^{-4}$	123.9 ± 26.5
		$\mu\mu$	1.15 ± 0.23	$8.72 \times 10^{-4} \pm 1.75 \times 10^{-4}$	130.7 ± 27.2
1400	1.60×10^{-5}	ee	1.67 ± 0.33	$7.43 \times 10^{-4} \pm 1.49 \times 10^{-4}$	115.8 ± 24.5
		$\mu\mu$	1.16 ± 0.23	$5.15 \times 10^{-4} \pm 1.03 \times 10^{-4}$	125.4 ± 26.1
1500	9.42×10^{-6}	ee	1.69 ± 0.34	$4.44 \times 10^{-4} \pm 8.90 \times 10^{-5}$	123.9 ± 26.5
		$\mu\mu$	1.13 ± 0.23	$2.97 \times 10^{-4} \pm 5.96 \times 10^{-5}$	130.7 ± 27.2

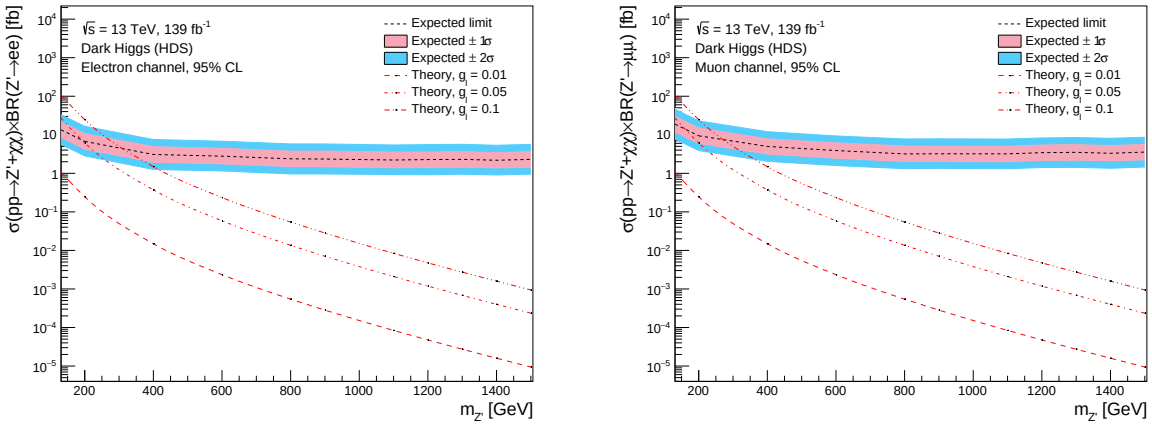


Figure 8.5: Mass exclusion limits of ee (left) and $\mu\mu$ (right) channel for Z' Dark Higgs Heavy Dark Sector model using the model dependent approach. The y-axis of both plots represents the cross-section times branching ratio of the process we are studying. The x-axis is the mass of the Z' boson. We did not interpolate between the available masses we had simulated, and have rather just connected the values calculated for each mass point in Table 8.1 by connecting the points. The dashed black line is the expected 95% CL limit calculated using the values of from Table 8.1 with a 1σ and 2σ deviation. The different dashed lines represent the theoretical cross-section times branching ratio of the process when varying the value of the lepton coupling g_l between the leptons and the Z' boson. The simulated events in this thesis utilized the value $g_l = 0.01$, we include the cross-section times branching ratio when increasing this coupling to 0.05 and 0.1 to see how the exclusions change.

When changing the lepton coupling g_l we have assumed that the efficiency of the cuts stays the same, as well as the number of background events in the last bin. This assumption is noteworthy due to the fact that if we increased the lepton coupling value of our model, this would increase the branching ratio of the leptonic decays. Meaning that when using MC to simulate we would have gotten more events. As one of the greatest challenges in this thesis was the imbalanced dataset, this fact would have helped mitigate the problem. In addition, a general rule of thumb in ML is that the more statistics one has when training a network, the better the network will learn. This means that if we instead simulated new events with a greater lepton coupling, the network could have achieved a greater efficiency. However, due to time constrains on this thesis we did not have the chance to explore the possibility of simulating more events, with varying lepton couplings.

We can furthermore statistically combine both of the ee and $\mu\mu$ results from Figure 8.5 to get the exclusion on a dilepton final state. Following this method for all other seven models[2] we can get the results in Figure 8.6 for the direct slepton production and 2HDM + a (only showing $\tan\beta = 30$) and in Figure 8.7 for the mono-Z' models.

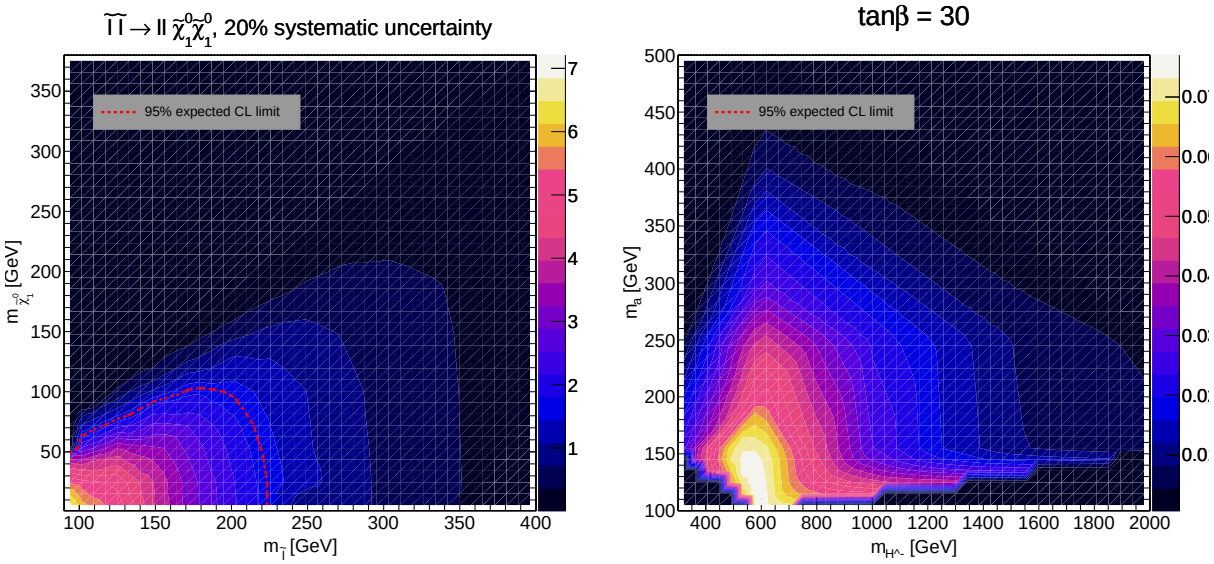


Figure 8.6: Mass exclusion limits of combined ee and $\mu\mu$ channel for direct slepton production (left) and 2HDM + a for $\tan\beta = 30$ (right) using the model dependent approach. The plots here have the two varying masses as the axes, the z-axis is the expected significance calculated using Eq. (4.20) with uncertainties. The expected 95% CL limit was chosen using Frequentist statistics using the significance $Z = 1.645$. For the direct slepton production model we have the slepton mass, $m_{\tilde{\ell}}$, on the x-axis, and the neutralino mass on $m_{\tilde{\chi}_1^0}$ the y-axis. For the 2HDM + a model we have the charged Higgs mass, m_{H^-}, on the x-axis, and the pseudoscalar a mass m_a on the y-axis. To see the exclusions for the other values of $\tan\beta$ on the 2HDM + a model see Appendix F.

[2]For more plots look at the GitHub repo: in https://github.com/rubenguevara/Master-Thesis/tree/master/Plots/XGBoost/DH_HDS changing "DH_HDS" for the model of interest

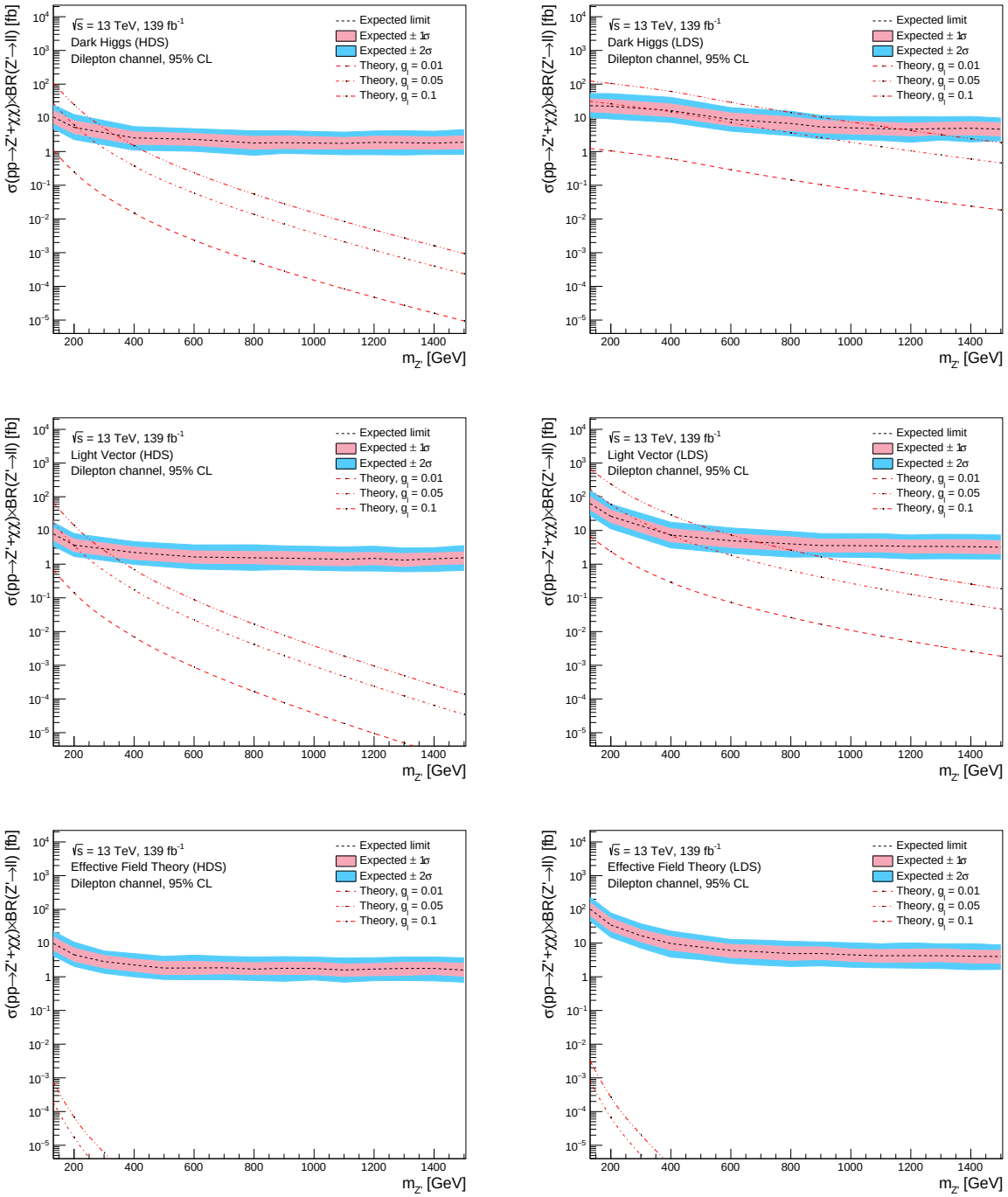


Figure 8.7: Mass exclusion limits of combined ee and μμ channel for all mono-Z' models in both the Heavy Dark Sector (HDS) and Light Dark Sector (LDS) using the model dependent approach. In the top row we have the mass exclusion of the Dark Higgs HDS (left) and Dark Higgs LDS (right) models. In the middle row we have the mass exclusions of the Light Vector HDS (left) and Light Vector LDS (right) models. In the bottom row we have the mass exclusion of the inelastic EFT HDS (left) and inelastic EFT LDS (right) models. The y-axis on all plots represents the cross-section times branching ratio of the process we are studying. The x-axis is the mass of the Z' boson. We did not interpolate between the available masses we had simulated, and have rather just connected the values calculated for each mass point by connecting the points. The dashed black line is the expected 95% CL limit calculated using Bayesian statistics with a 1σ and 2σ deviation. The different dashed lines represent the theoretical cross-section times branching ratio of the process when varying the value of the lepton coupling g_l between the leptons and the Z' boson. The simulated events in this thesis utilized the value $g_l = 0.01$, we include the cross-section times branching ratio when increasing this coupling to 0.05 and 0.1 to see how the exclusions change.

8.2 Model independent approach

The second method we utilized in this analysis was to train three BDTs in different orthogonal Signal Regions (SR) of kinematical phase space using a dataset that included all the available DM models with a dilepton and Missing Transverse Energy (MET) final state. The DM models we studied were a supersymmetric direct slepton production model, a Two Higgs Doublet Model with an additional pseudoscalar a (2HDM + a), and three mono-Z' models, Dark Higgs (DH), Light Vector (LV) and inelastic EFT. The three mono-Z' models were furthermore divided into a Heavy Dark Sector (HDS) and a Light Dark Sector (LDS). As this method uses models that share similar experimental features, like the final state, but are based of different theoretical principles, we called this method for the model independent approach. The three SR we chose to divide the dataset in were

- SR1: $m_{ll} > 110$ GeV and $E_T^{miss} \in [50, 100]$ GeV
- SR2: $m_{ll} > 110$ GeV and $E_T^{miss} \in [100, 150]$ GeV
- SR3: $m_{ll} > 110$ GeV and $E_T^{miss} > 150$ GeV,

where the invariant mass cut $m_{ll} > 110$ GeV was chosen to remove the Z-boson peak.

To not repeat how we arrive at the expected mass exclusion limits, which was discussed in detail in Section 8.1, we will present how this method was used by looking at the expected mass exclusion limit when testing on the DH HDS model. As the SRs we chose are orthogonal in MET we assume that they are not correlated, meaning that we can statistically combine the results for each SR to get a combined mass exclusion limit in a combined SR. This is shown in Figure 8.8 where we can see how the mass exclusion looks for the electron channel in every SR, including the combined SR limit.

In Figure 8.8 we can see that we can not exclude anything in SR1 (top left plot), we also see that as we go to SR2 (top right plot) the exclusion gets better, and we can exclude the model up to $m_{Z'} \approx 250$ GeV when we set $g_l = 0.1$[3]. This exclusion gets higher when we go to SR3 (bottom left plot), where we can exclude the model up to $m_{Z'} \approx 425$ GeV

[3]See Section 8.1 for discussion on lepton coupling variations with the mono-Z' models.

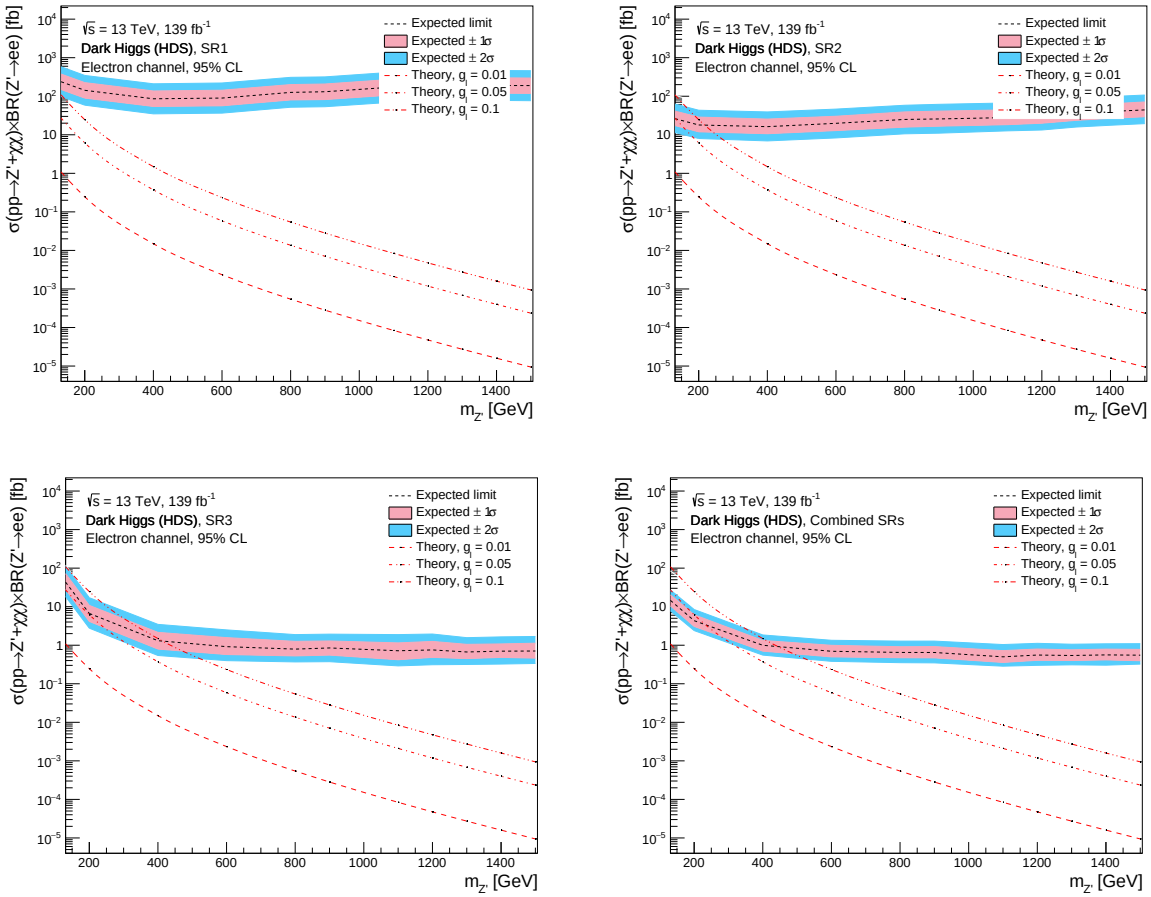


Figure 8.8: Mass exclusion limits results for Dark Higgs Heavy Dark Sector model on the ee channel in all SRs, including the combined SR channel. All SRs used in this thesis had $m_{ll} > 110$ GeV. In the top left we have the expected mass exclusion in SR1 where $E_T^{miss} \in [50, 100]$ GeV. In the top right we have the expected mass exclusion in SR2 where $E_T^{miss} \in [100, 150]$ GeV. In the bottom left we have the expected mass exclusion limit in SR3 where $E_T^{miss} > 150$ GeV. In the bottom right we have the expected mass limit for the model when statistically combining the result of SR1, SR2 and SR3. The y-axis on all plots represents the cross-section times branching ratio of the process we are studying. The x-axis is the mass of the Z' boson. We did not interpolate between the available masses we had simulated, and have rather just connected the values calculated for each mass point by connecting the points. The dashed black line is the expected 95% CL limit with a 1σ and 2σ deviation. The different dashed lines represent the theoretical cross-section times branching ratio of the process when varying the value of the lepton coupling g_l between the leptons and the Z' boson. The simulated events in this thesis utilized the value $g_l = 0.01$, we include the cross-section times branching ratio when increasing this coupling to 0.05 and 0.1 to see how the exclusions change.

when setting $g_l = 0.1$, but the expected limit is slightly lower than for SR2. However, when statistically combining the results for every SR (bottom right plot) we now get that we can exclude the model up to $m_{Z'} \approx 450$ GeV with $g_l = 0.1$, while having an overall lower expected limit with smaller 1σ and 2σ deviations. This means that this method of dividing into several SRs yields higher exlcusions. **Question: Should I state already**

that this is better than the model dependent approach or should I keep this for the next section?

This can be generalized for every model studied, the exclusion limits for the combined SRs in the dilepton channel for the direct slepton production and 2HDM + a model can be seen in Figure 8.9, and in Figure 8.10 for the mono-Z' models. To see all the plots used to arrive at these limits for every model you can see the GitHub repo[4], and to see all the tables with inputs see Appendix I.

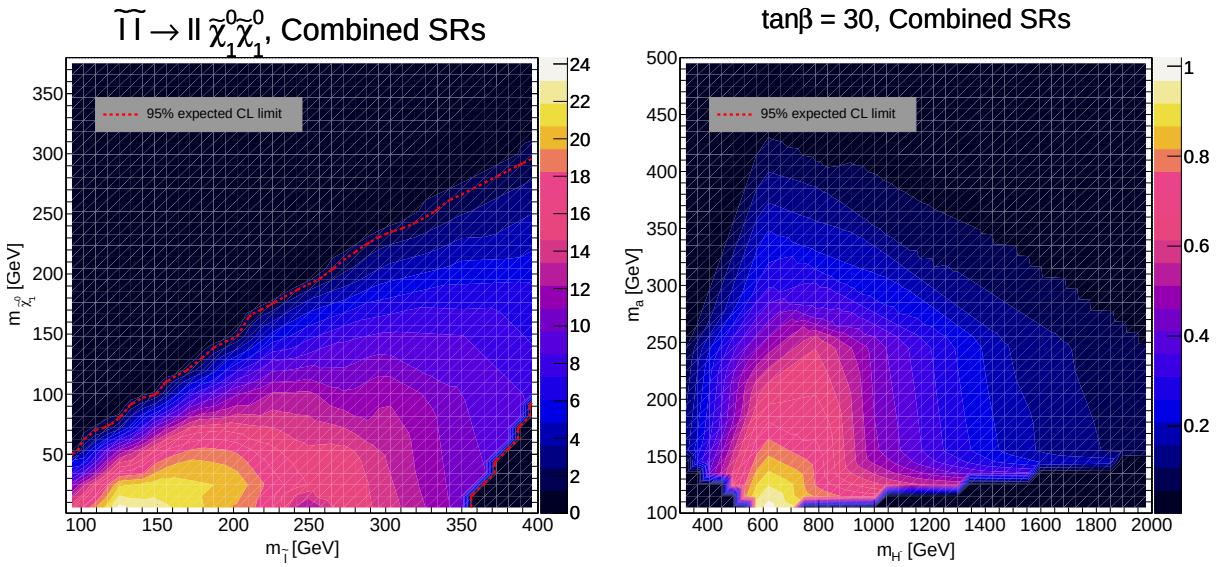


Figure 8.9: Mass exclusion limits of combined ee and $\mu\mu$ channel for direct slepton production (left) and 2HDM + a for $\tan\beta = 30$ (right) using the model independent approach. The plots here have the two varying masses as the axes, the z-axis is the expected significance calculated using Eq. (4.20) with uncertainties. The expected 95% CL limit was chosen using Frequentist statistics using the significance $Z = 1.645$. For the direct slepton production model we have the slepton mass, $m_{\tilde{\ell}}$, on the x-axis, and the neutralino mass on $m_{\tilde{\chi}_1^0}$ the y-axis. For the 2HDM + a model we have the charged Higgs mass, m_{H^-}, on the x-axis, and the pseudoscalar a mass m_a on the y-axis. To see the exclusions for the other values of $\tan\beta$ on the 2HDM + a model see Appendix F. Something to note about the direct slepton production plot (left), is the lack of exclusions in the bottom right corner where $m_{\tilde{\ell}} > 350$ GeV and $m_{\tilde{\chi}_1^0} < 50$ GeV, the reason for this is because we did not have any simulated event at that mass range, to see the samples we had available see Figure 6.7.

[4]Available here: https://github.com/rubenguevara/Master-Thesis/tree/master/Plots/XGBoost/Model_independent

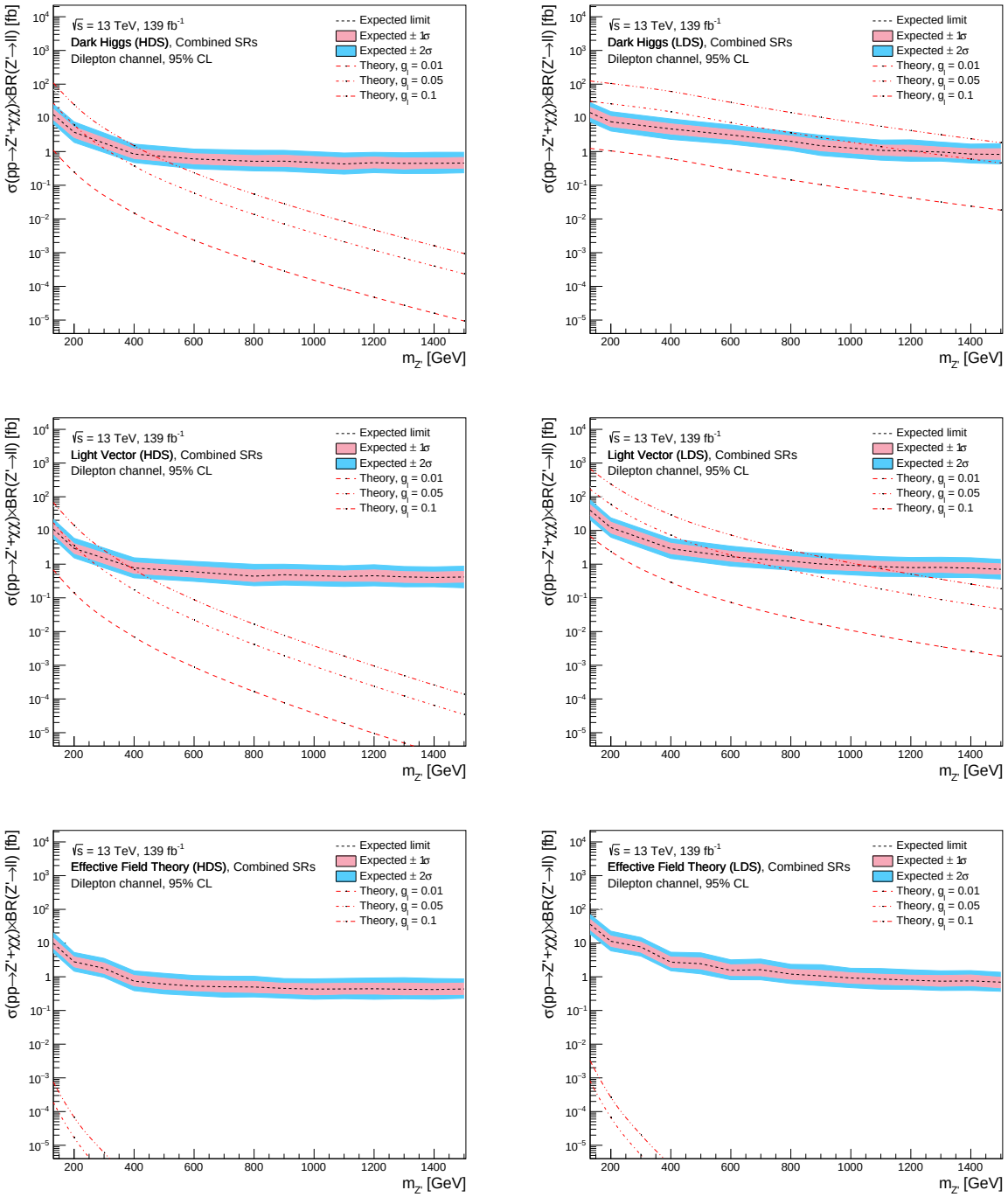


Figure 8.10: Mass exclusion limits of combined ee and $\mu\mu$ channel for all mono-Z' models in both the Heavy Dark Sector (HDS) and Light Dark Sector (LDS) using the model independent approach. In the top row we have the mass exclusion of the Dark Higgs HDS (left) and Dark Higgs LDS (right) models. In the middle row we have the mass exclusions of the Light Vector HDS (left) and Light Vector LDS (right) models. In the bottom row we have the mass exclusion of the inelastic EFT HDS (left) and inelastic EFT LDS (right) models. The y-axis on all plots represents the cross-section times branching ratio of the process we are studying. The x-axis is the mass of the Z' boson. We did not interpolate between the available masses we had simulated, and have rather just connected the values calculated for each mass point by connecting the points. The dashed black line is the expected 95% CL limit calculated using Bayesian statistics with a 1σ and 2σ deviation. The different dashed lines represent the theoretical cross-section times branching ratio of the process when varying the value of the lepton coupling g_l between the leptons and the Z' boson. The simulated events in this thesis utilized the value $g_l = 0.01$, we include the cross-section times branching ratio when increasing this coupling to 0.05 and 0.1 to see how the exclusions change.

8.3 Comparison of results

To more easily compare the model dependent approach, where we train one network for every DM model using all the available simulated events, to the model independent approach, where we train three networks in kinematically orthogonal regions containing every simulated event for all DM models, we present a side by side comparison of the expected mass exclusion plots. To remind how these plots work, the goal is to exclude as many of the mass points as possible. For the one-dimensional mono-Z' exclusions, we want as much of the theoretical cross-section times branching ratio to be over the expected 95% CL limit. While for the two-dimensional exclusions (SUSY and 2HDM + a), we want as many of the mass points to be inside the 95% CL limit as these are excluded.

In Figure 8.11 we can see the comparison of the direct slepton production model (upper row) and the Two Higgs Doublet Model with an additional pseudoscalar a model (lower row). The comparisons of the mono-Z' models, the Dark Higgs, Light Vector and inelastic EFT can be seen in Figure 8.12 for the Heavy Dark Sector, and Figure 8.13 for the Light Dark Sector. For every single model we see that the model independent approach yields better results as it excludes more mass points.

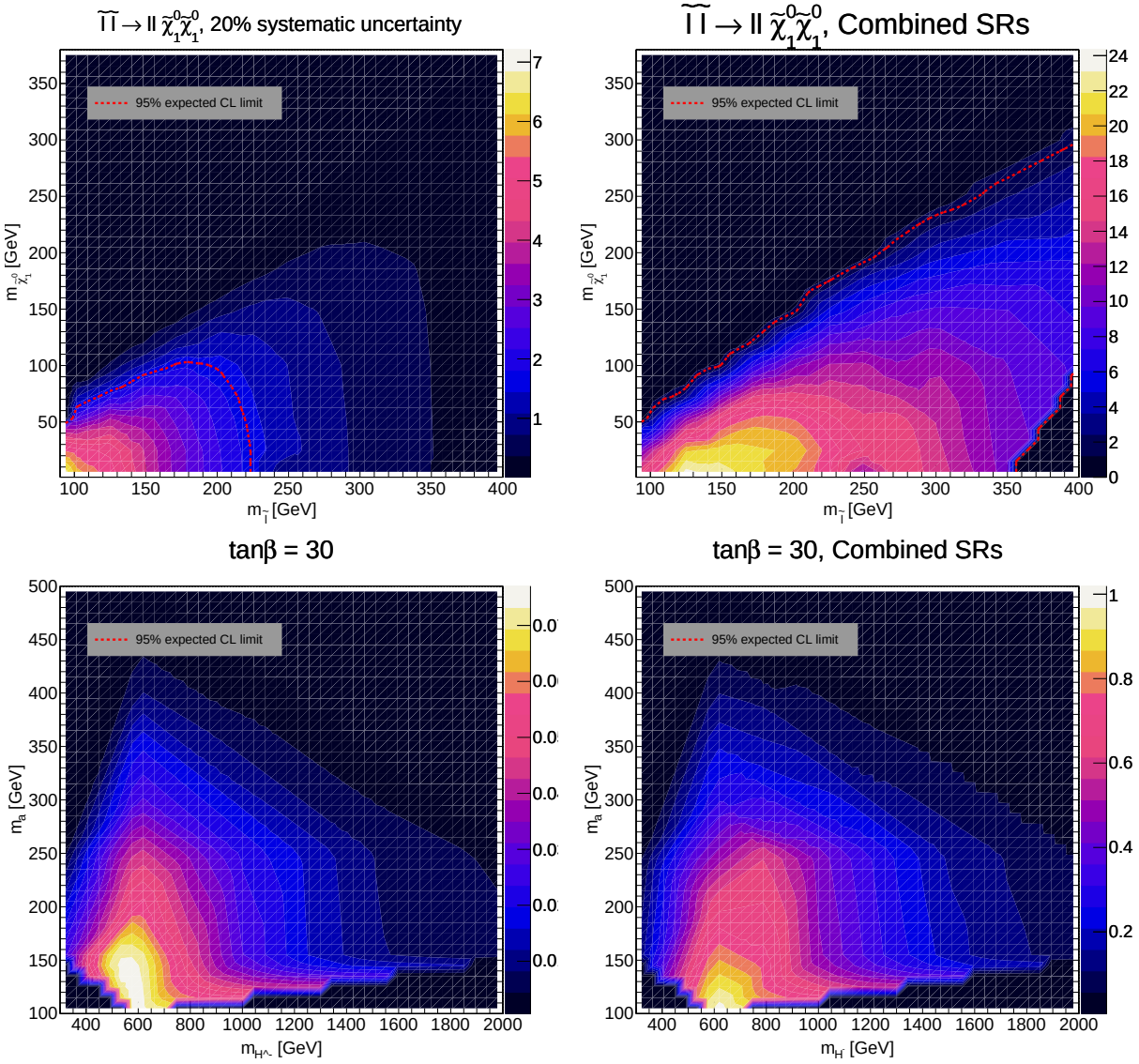


Figure 8.11: Comparison of mass exclusion limit using the model dependent (left) and model independent (right) approach for direct slepton production (upper plots) and 2HDM + a (lower plots). The plots here have the two varying masses as the axes, the z-axis is the expected significance calculated using Eq. (4.20) with uncertainties. The expected 95% CL limit was chosen using Frequentist statistics using the significance $Z = 1.645$. For the direct slepton production model we have the slepton mass, $m_{\tilde{\ell}}$, on the x-axis, and the neutralino mass on $m_{\tilde{\chi}_1^0}$ the y-axis. For the 2HDM + a model we have the charged Higgs mass, m_{H^-}, on the x-axis, and the pseudoscalar a mass m_a on the y-axis. To see the exclusions for the other values of $\tan\beta$ on the 2HDM + a model see Appendix F. Something to note about the direct slepton production plot (top right), is the lack of exclusions in the bottom right corner where $m_{\tilde{\ell}} > 350$ GeV and $m_{\tilde{\chi}_1^0} < 50$ GeV, the reason for this is because we did not have any simulated event at that mass range, to see the samples we had available see Figure 6.7.

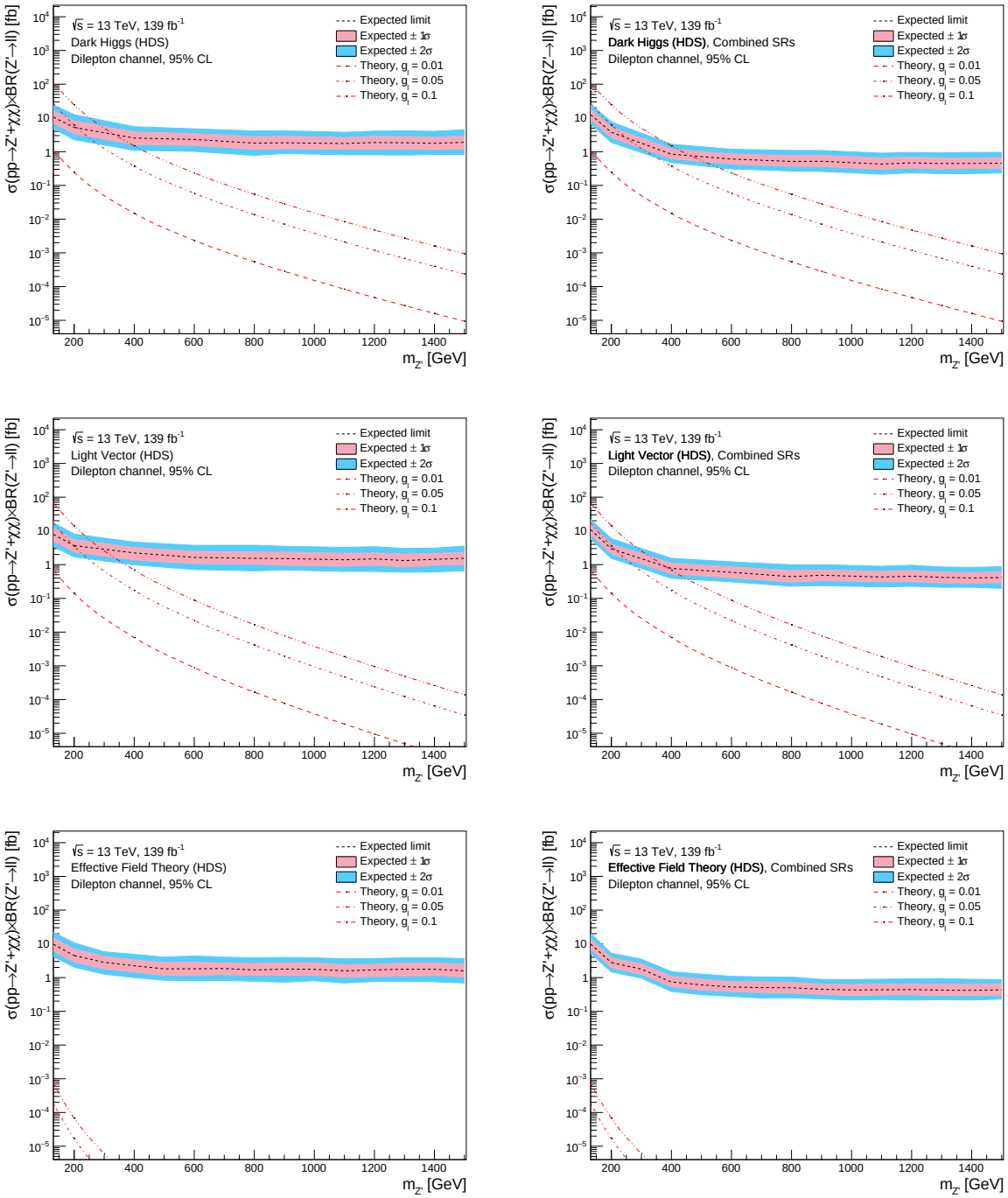


Figure 8.12: Comparison of the mass exclusion limits of combined ee and $\mu\mu$ channel for all mono-Z' models using the model dependent approach (left) and model independent approach (right) in the Heavy Dark Sector (HDS). In the top row we have the mass exclusion of the Dark Higgs model. In the middle row we have the mass exclusions of the Light Vector model. In the bottom row we have the mass exclusion of the inelastic EFT model. The y-axis on all plots represents the cross-section times branching ratio of the process we are studying. The x-axis is the mass of the Z' boson. We did not interpolate between the available masses we had simulated, and have rather just connected the values calculated for each mass point by connecting the points. The dashed black line is the expected 95% CL limit calculated using Bayesian statistics with a 1σ and 2σ deviation. The different dashed lines represent the theoretical cross-section times branching ratio of the process when varying the value of the lepton coupling g_l between the leptons and the Z' boson. The simulated events in this thesis utilized the value $g_l = 0.01$, we include the cross-section times branching ratio when increasing this coupling to 0.05 and 0.1 to see how the exclusions change.

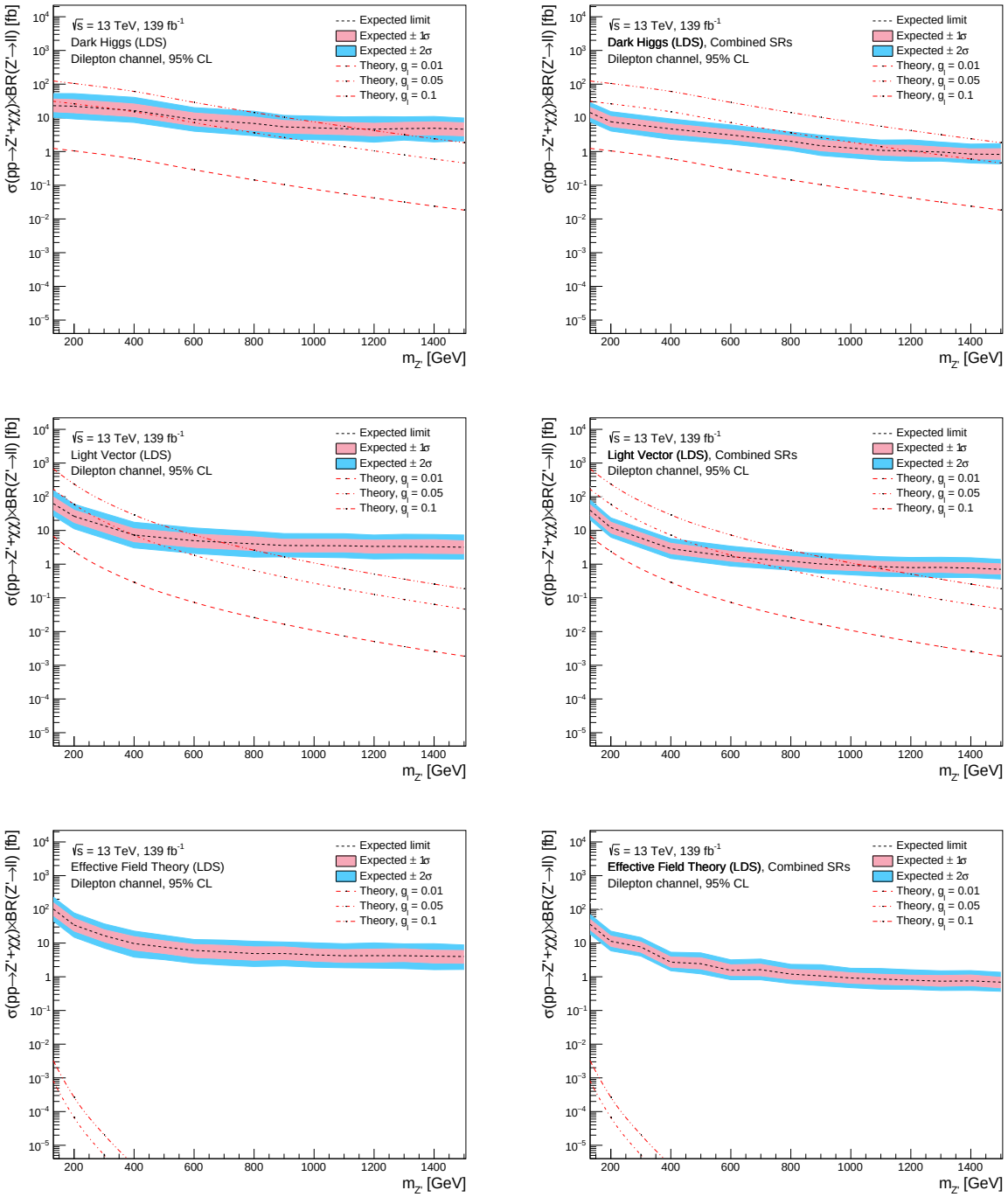


Figure 8.13: Comparison of the mass exclusion limits of combined ee and $\mu\mu$ channel for all mono-Z' models using the model dependent approach (left) and model independent approach (right) in the Light Dark Sector (LDS). In the top row we have the mass exclusion of the Dark Higgs model. In the middle row we have the mass exclusions of the Light Vector model. In the bottom row we have the mass exclusion of the inelastic EFT model. The y-axis on all plots represents the cross-section times branching ratio of the process we are studying. The x-axis is the mass of the Z' boson. We did not interpolate between the available masses we had simulated, and have rather just connected the values calculated for each mass point by connecting the points. The dashed black line is the expected 95% CL limit calculated using Bayesian statistics with a 1σ and 2σ deviation. The different dashed lines represent the theoretical cross-section times branching ratio of the process when varying the value of the lepton coupling g_l between the leptons and the Z' boson. The simulated events in this thesis utilized the value $g_l = 0.01$, we include the cross-section times branching ratio when increasing this coupling to 0.05 and 0.1 to see how the exclusions change.

Chapter 9

Conclusion and outlook

9.1 Conclusion

After exploring the intricacies of preparing both a NN and a BDT ML algorithm to learn a binary classification task of DM signal and SM background, we concluded, due to time, that a BDT ML approach was best suited for this search. The searches we conducted were based on models from three different theoretical principles. The first one containing a new $U(1)'$ vector boson, Z', that can decay and couple to a DM WIMP candidate. The first one being Z' coupling to a new scalar boson h_D, which we call the Dark Higgs (DH) model. The second is an off-shell Z' decaying into two dark states χ_1 and χ_2, where χ_2 decays again into a Z' and χ_1, which is the DM candidate, this we called the Light Vector (LV) model. The third model, is an inelastic Effective Field Theory model, which is similar to the LV model, with the exception being that there are no assumptions about the quark coupling to the Z', we called this the EFT model. The three models were furthermore split into two groups, the difference being the mass of the DM candidate, we called these for the Heavy- and Light Dark Sector (HDS and LDS) for heavy and light DM respectively. The second theoretical principle we studied was Supersymmetry, in particular a direct slepton production model, where the bosonic superpartner of the lepton, the slepton, $\tilde{\ell}$, decays into a lepton and a first generation neutralino $\tilde{\chi}_1^0$, which is a DM candidate. The las model we studied was a Two Higgs Doublet Model with the addition of a pseudoscalar mediator, a, which mediates the interactions between the visible and dark sector, we called this model for 2HDM + a for short.

As the ultimate goal for the thesis was to test model independent approaches for new physics searches using ML, we first tested what we called the model dependent approach. This approach consisted of using a dataset with all the simulated SM background events containing a dilepton and MET final state, and one of the DM models, including all the simulated events. To better the performance and the computational time, we only looked at events with MET $>$ 50 GeV, for the mono-Z' we included an additional cut of $m_{ll} > 110$ GeV. We trained one BDT for each model using this type of dataset. To test the results of this approach we computed mass exclusion limits using Bayesian and Frequentist statistics for every model.

The second approach, which we called the model independent approach, consisted of a dataset containing all the DM models and all the SM background. The difference however was that we trained three BDTs in orthogonal MET spaces which we called Signal Regions (SRs). All the networks only looked at events with $m_{ll} > 110$ GeV to reduce the Z-boson peak. The three SR we chose to divide the dataset in were

- SR1: $m_{ll} > 110$ GeV and $E_T^{miss} \in [50, 100]$ GeV
- SR2: $m_{ll} > 110$ GeV and $E_T^{miss} \in [100, 150]$ GeV
- SR3: $m_{ll} > 110$ GeV and $E_T^{miss} > 150$ GeV

After training the three networks on each SR we tested each model by computing the expected mass exclusion limit using Bayesian and Frequentist statistics in every respective SR. To compare the model independent approach with the model dependent approach, we statistically combined the mass exclusion limit of all SRs into one combined SR for all eight models.

Doing this we observed that we were able to compute higher mass exclusion limits for every model we studied using the model independent approach, the side by side comparison can be seen in Section 8.3.

9.2 Outlook

For the future, there are intriguing possibilities to further enhance the model independent approach and achieve even better results with fewer ML networks, building on the promising findings in this thesis. It would be fascinating to explore the potential of Deep Neural Networks (DNNs) with an expanded range of Dark Matter (DM) models, as DNNs thrive when provided with larger datasets and more statistical information.

Another avenue worth investigating is the Parametrized Neural Network (PNN) approach proposed by Baldi et al. [29]. In this approach, an additional feature is included on the NN input layer to specify the mass of the particle being studied as a signal. Exploring the combination of a DNN and PNN, forming a Deep-Parametrized Neural Network (DPNN), could yield a powerful tool for the model independent approach. By dividing a dataset consisting of multiple models, all sharing the same experimental signatures, into different regions of kinematical phase space, such a general ML network could rapidly test new models and swiftly provide mass exclusion limits.

By developing more efficient and versatile ML algorithms, we move closer to understanding physics beyond the standard model. Additionally, these advancements in ML techniques may have broader applications in various scientific domains, further amplifying their impact. This progress, in turn, holds the potential to advance our comprehension of spacetime.

Appendix A

Network optimization

A.1 Neural Network Training

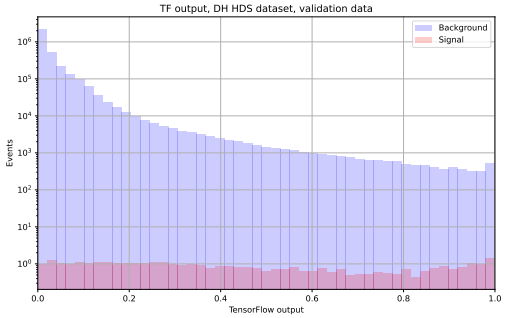
For most of the NN optimization methods we trained a NN with the following hyperparameters:

- One hidden layer
- 100 neurons in the hidden layer
- 0.1 learning rate η
- 10^{-5} L2-regularization parameter λ
- The ADAM optimizer

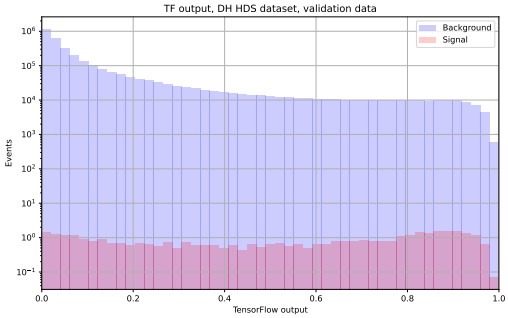
We will mention whenever these parameters were not used. The results for the different network optimization methods explained in Chapter 7.2 follow from here. Starting with the normalization of data.

A.1.1 Normalization of data

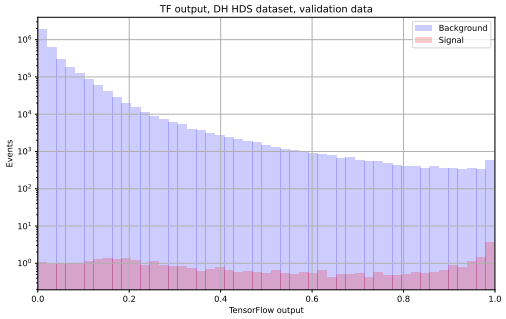
We trained a network using 80% of the whole SM background events as well as 80% of all the Z' DH HDS samples. As sample weights we only balanced the signal and background using MC events. We did this by using method 3. on Chapter 7.2.3. We tested on the remaining 20% of the SM background events, as well as 20% of Z' DH HDS events where $m_{Z'} = 130$ GeV. The different normalization methods explained in Chapter 7.2.2 have been tested and can be seen in the Figure A.1.



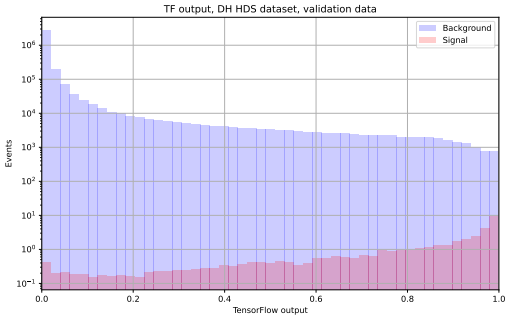
(a) No normalization



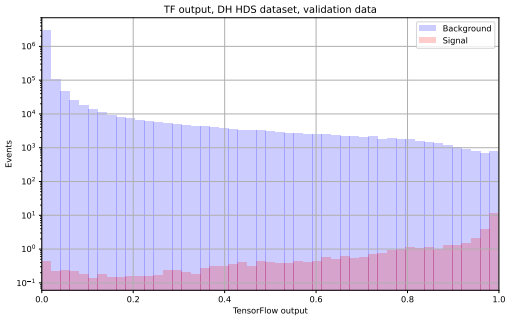
(b) Layer normalization



(c) Min max scaling



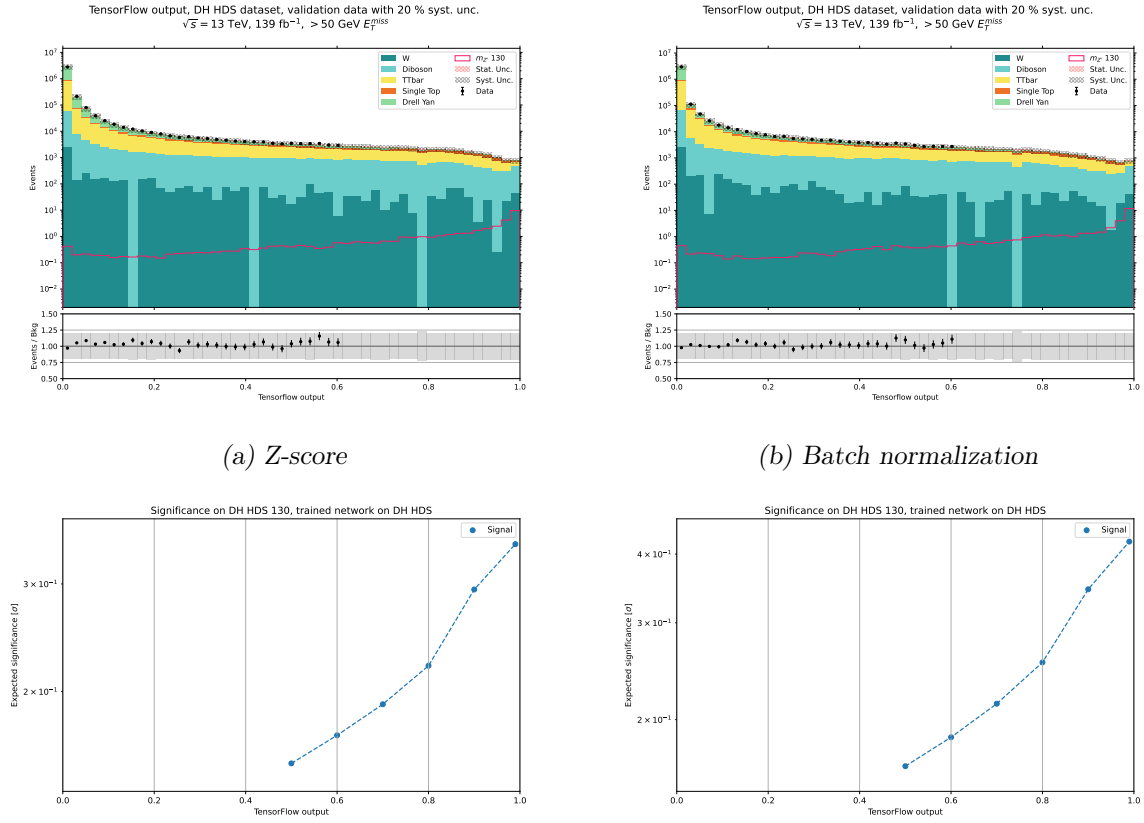
(d) Z-score



(e) Batch normalization

Figure A.1: NN prediction when using different normalization methods. This is testing a dataset with 20% of the Z' DH HDS $m_{Z'} = 130$ GeV events.

Including data points as well as uncertainties on the best performing normalization methods, as well as their calculate expected significance as explained in Chapter 5.3.5, yields the plots shown in Figure A.2. For more Figures showing NN training results see the GitHub repo[1].



(a) Z-score *(b) Batch normalization*

(c) The expected significance of a) *(d) The expected significance of b)*

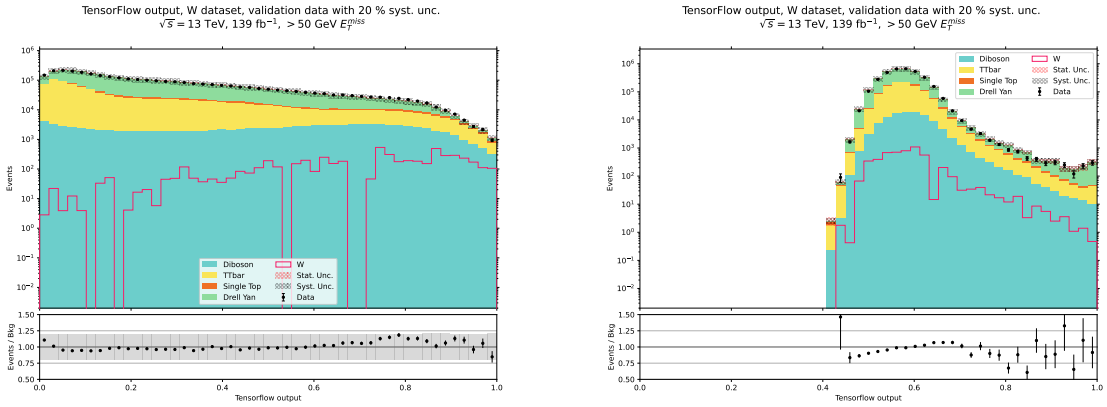
Figure A.2: Comparison of the best normalization methods. Figure a) and b) show the validation data of both cases, c) and d) show the expected significance of the validation plots when making a cut on the output.

As we can see the `Batch_normalization` method gives us the highest signal and background but is it reasonable to use this method when one is not using a CNN? The reason batch normalization might work best for our case is because when we divide the data by batches it might unevenly represent the SM / signal and their ratio. But by using batch normalization it takes the average of all the batches creating a closer to real distribution. For the following examples in this chapter we will use batch normalization to make the optimal network.

[1] Available here: https://github.com/rubenguevara/Master-Thesis/tree/master/Plots/NeuralNetwork/Normalization_method

A.1.2 Balancing of signal and background

To try the different sample weight methods explained in Chapter 7.2.3 we used a dataset consisting of only SM events where the goal was to treat the W channel as signal and try to isolate it from other SM processes. To train we used `Batch_normalization` and 80% of the SM background events. To test we used the remaining 20% of SM events. We also tested the difference in performance when using the SGD and ADAM optimizers. The difference in distributions when using different optimizers can be seen in Figure A.3, here the balancing method (3. on Chapter 7.2.3) is used.



(a) Using ADAM *(b) Using SGD*

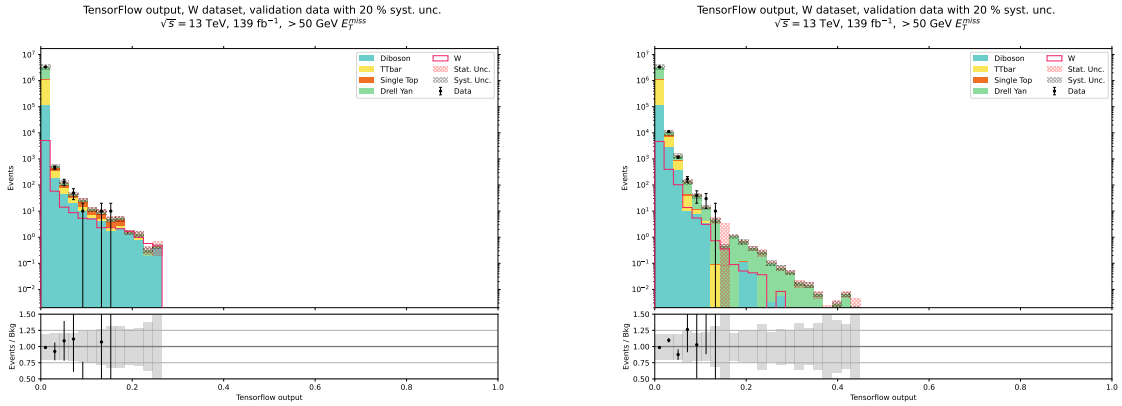
Figure A.3: Validation plots using SGD and ADAM. This was done using a dataset where the goal was to isolate the W background process from other SM background processes

As ADAM is far better at sorting signal from background we will only use this optimizer further. The results for the different weighting methods can be seen in Figure A.4, which shows the validation plots and in Figure A.5 which shows the ROC score. For more Figures showing NN training results see the GitHub repo[2].

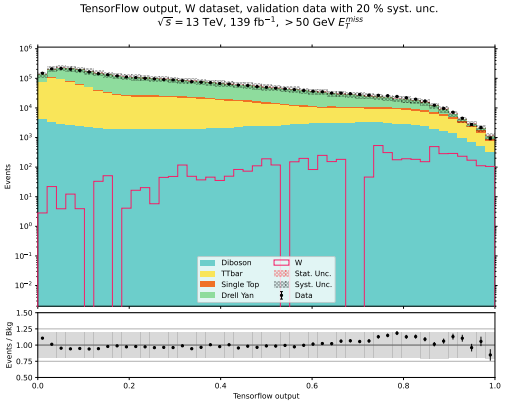
From the figures we see that the only way the network does not predict every event to be a background event[3] is when we introduce the balancing method. We also see that the AUC increases more as well. Meaning that we must balance our dataset. Something else to mention, as to why the network does such a poor job at classifying the W background, is that the network here was not optimized for the search. If we were to conduct a

[2]Available here: https://github.com/rubenguevara/Master-Thesis/tree/master/Plots/NeuralNetwork/W

[3]Since the output is the score from 0-1 our network gives every event, where 0 means that the network predicts 0% chance for an event to be signal



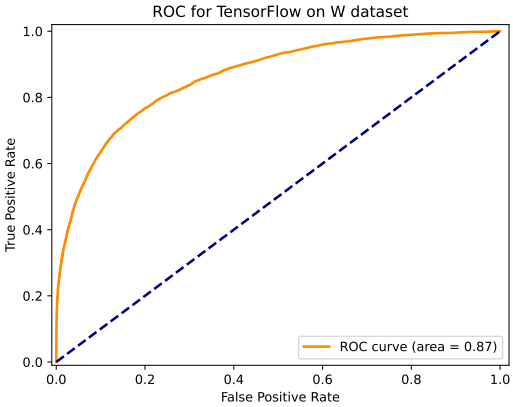
(a) Using no weights *(b) Using re-weighting weights*



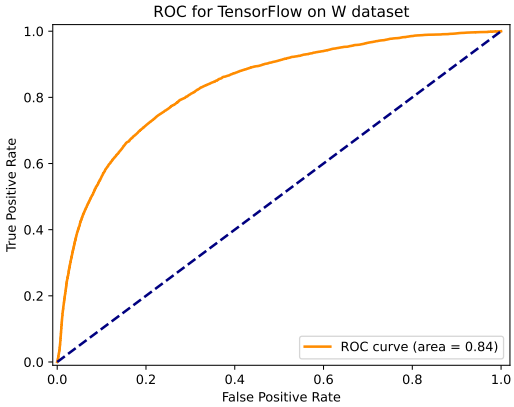
(c) Using $\frac{N_{sig}}{N_{bkg}}$

Figure A.4: Validation plots of different balancing methods. This was done using a dataset where the goal was to isolate the W background process from other SM background processes

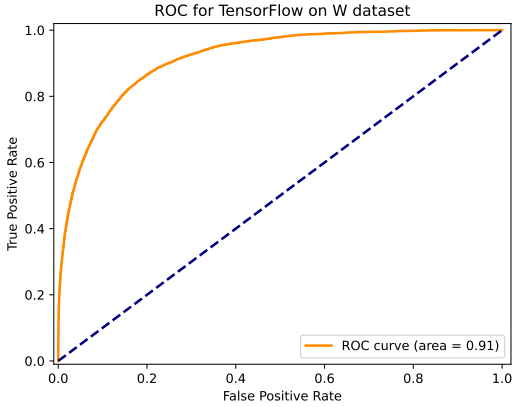
thorough grid search of all hyperparameters it would yield greater results, but as this chapter is for testing methods rather analyzing data we will not delve further into it for now,



(a) Using no weights



(b) Using re-weighting weights

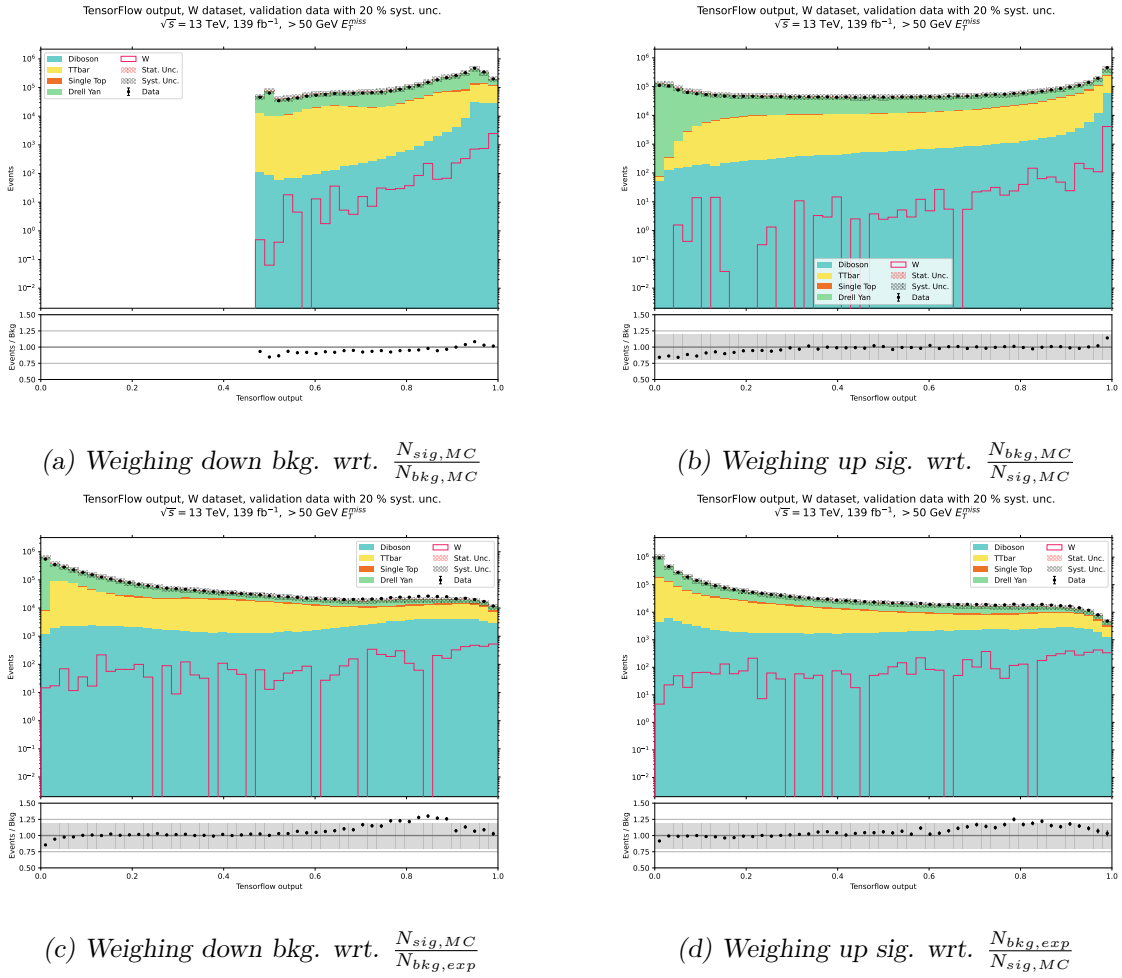


(c) Using $\frac{N_{sig}}{N_{bkg}}$

Figure A.5: ROC plots of different balancing methods. This was done using a dataset where the goal was to isolate the W background process from other SM background processes

A.1.3 Sample weights to get expected events

To try the different sample weight methods explained in Chapter 7.2.4 which include the weights (Chapter 7.1.1), we used a dataset consisting of only SM events where the goal was to treat the W channel as signal and try to isolate it from other SM processes. To train we used `Batch_normalization` and 80% of the SM background events. To test we used the remaining 20% of SM events. The results can be seen in Figure A.6, which shows the validation plots and in Figure A.7 which shows the ROC score. For more Figures showing NN training results see the GitHub repo[4].



(a) *Weighing down bkg. wrt.* $\frac{N_{sig,MC}}{N_{bkg,MC}}$

(b) *Weighing up sig. wrt.* $\frac{N_{bkg,MC}}{N_{sig,MC}}$

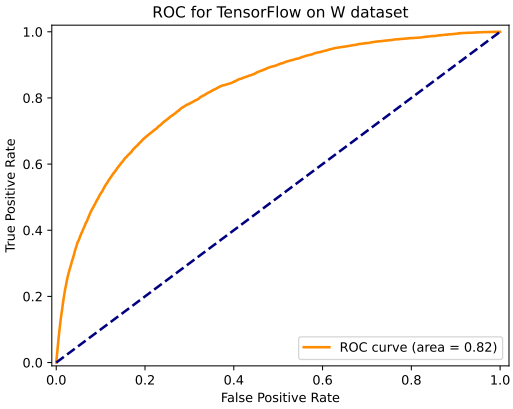
(c) *Weighing down bkg. wrt.* $\frac{N_{sig,MC}}{N_{bkg,exp}}$

(d) *Weighing up sig. wrt.* $\frac{N_{bkg,exp}}{N_{sig,MC}}$

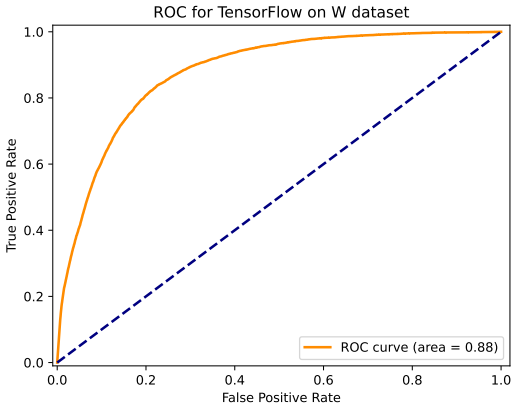
Figure A.6: Validation plots of different balancing methods when re-weighting background events to expected events. This was done using a dataset where the goal was to isolate the W background process from other SM background processes

As we are only re-weighting the background events, we can see from the figures that we get the best results when balancing with respect to the expected number of background

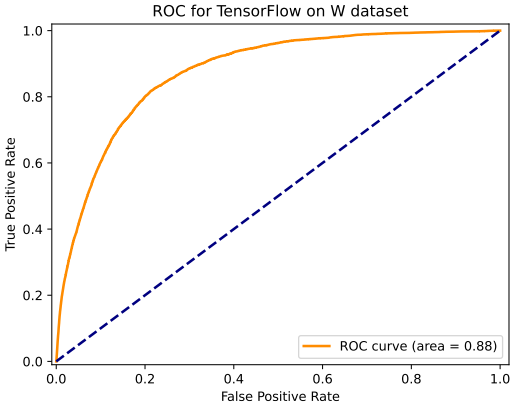
[4] Available here: https://github.com/rubenguevara/Master-Thesis/tree/master/Plots/NeuralNetwork/W



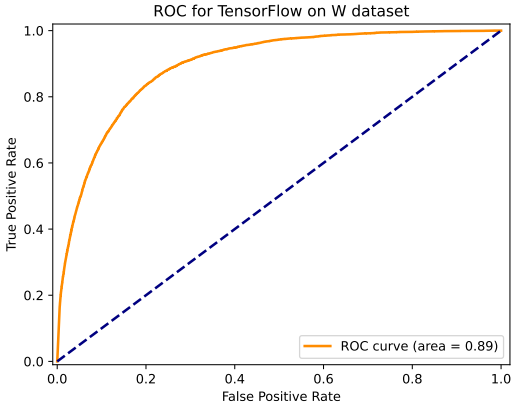
(a) *Weighing down bkg. wrt.* $\frac{N_{sig,MC}}{N_{bkg,MC}}$



(b) *Weighing up the signal wrt.* $\frac{N_{bkg,MC}}{N_{sig,MC}}$



(c) *Weighing down bkg. wrt.* $\frac{N_{sig,MC}}{N_{bkg,exp}}$

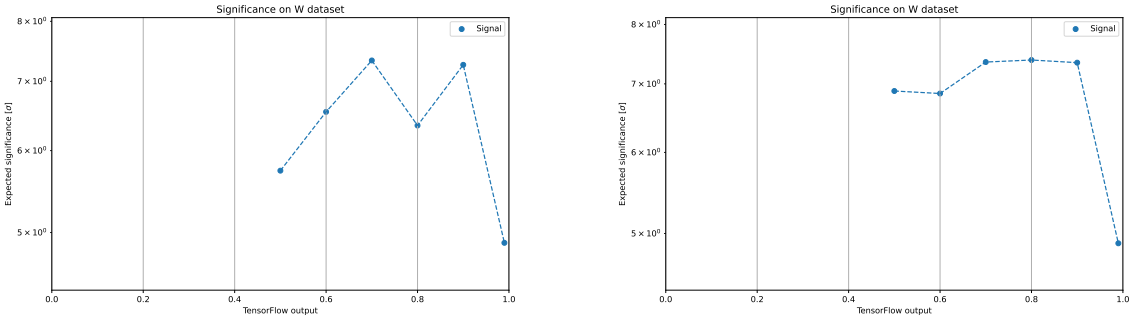


(d) *Weighing up sig. wrt.* $\frac{N_{bkg,exp}}{N_{sig,MC}}$

Figure A.7: ROC plots of different balancing methods when re-weighting background events to expected events. This was done using a dataset where the goal was to isolate the W background process from other SM background processes

events. Which is optimal, as this is what goes into the network. Whether it is better to weigh down the background or weight up the signal is not clear however, from the AUC of the ROC curves it is slightly better to weigh up the signal. To check which gives a higher expected significance however we can look at the expected significance, this is shown in Figure A.8. Here we see that there is a greater expected significance when weighing up the signal events.

As a last note for the testing of these methods, the networks, while still getting over 5σ expected significance (without errors) on the W channel, do not have the best distribution on the validation plots. The reason for this might be because we did not optimize the networks we tested, but rather used the same network for test.



(a) Weighing down bkg. wrt. $\frac{N_{sig,MC}}{N_{bkg,exp}}$

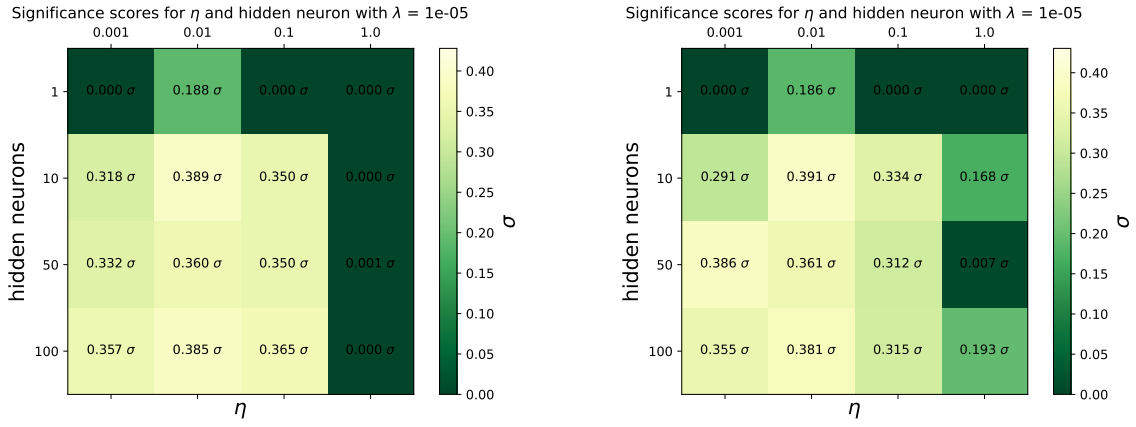
(b) Weighing up sig. wrt. $\frac{N_{bkg,exp}}{N_{sig,MC}}$

Figure A.8: Expected significance plots of the best balancing methods when re-weighting background events to expected events. This was done using a dataset where the goal was to isolate the W background process from other SM background processes

A.1.4 Padding of data

For the padding problem. We will as explained in Chapter 7.2.1 try the new variables presented in Table 7.2. The other method we tried was to remove the features with jagged arrays, that means the p_T, η, ϕ of the three most energetic jets, as well as the invariant mass of the two most energetic ones, m_{jj}. The trained a network using 80% of the whole SM background events as well as 80% of all the Z' DH HDS samples. As sample weights we used the best method from the previous section, which was to re-weight every background event and balance the dataset by weighing up all signal events by the ratio of expected number of background events over signal MC events, $\frac{N_{bkg,exp}}{N_{sig,MC}}$. As the best normalization method was `Batch_normalization`, this method was used here. We also utilized the ADAM optimizer instead of SGD.

As changing features changes the whole dataset, then to get the best results as possible we went through a full grid search following the steps in Chapter 7.2.5 for both networks. The result for the hyperparameters that gave the highest significance can be seen in Figure A.9.



(a) When including new features *(b) When dropping features*

Figure A.9: Grid search result for pad testing on NN. This is training a dataset with 80% of all Z' DH HDS events.

This means that the best hyperparameters for both networks coincidentally is the same, meaning: `n_neuron = 10, eta = 0.01, lamda = 1e-5`. The loss, AUC and binary accuracy over epochs for the best networks can be seen in Figure A.10 and Figure A.11.

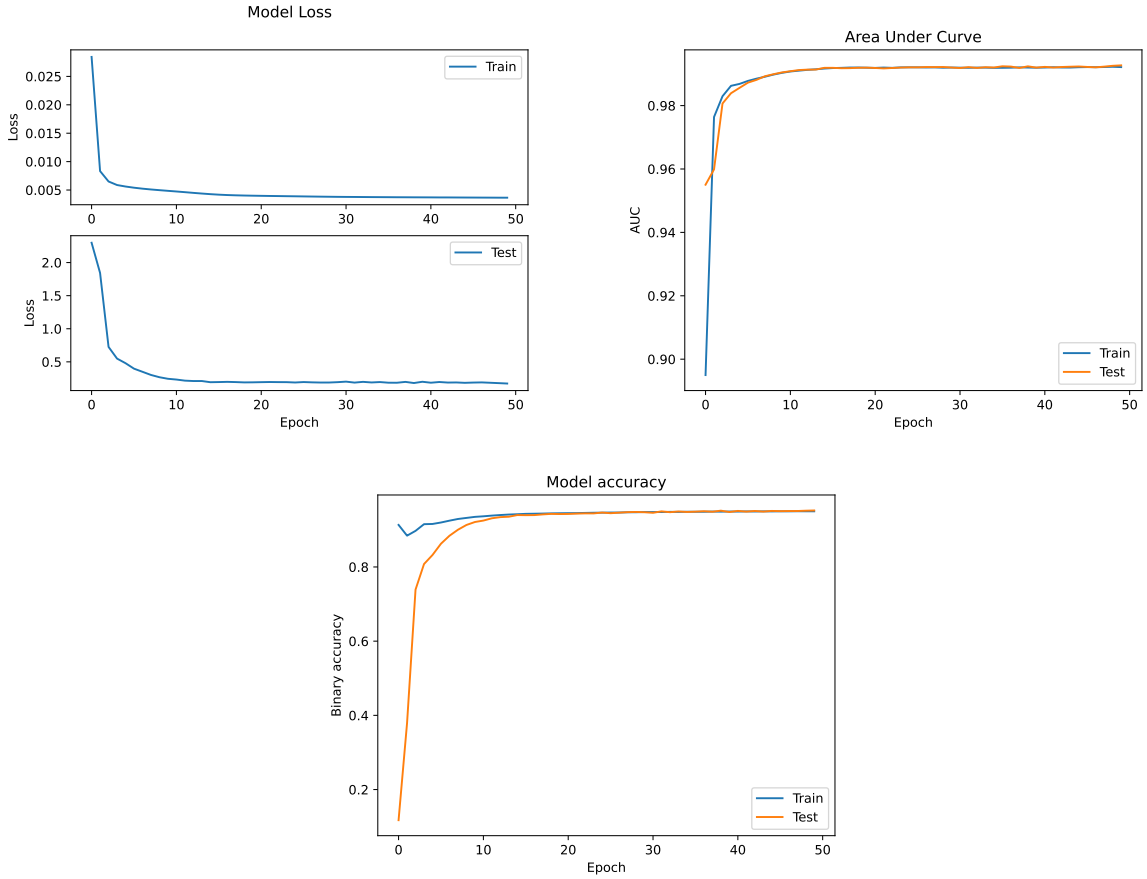


Figure A.10: NN parameters after 50 epochs with new features. This is training a dataset with 80% of all Z' DH HDS events.

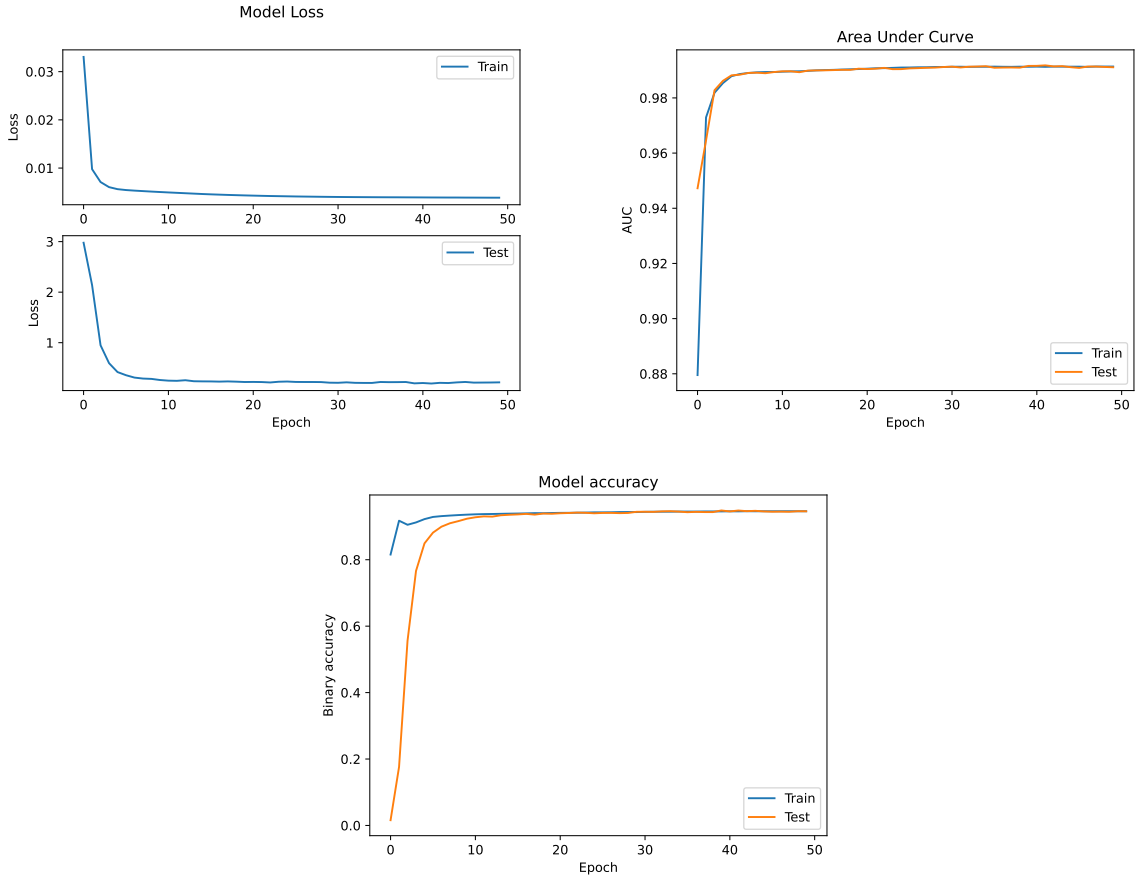
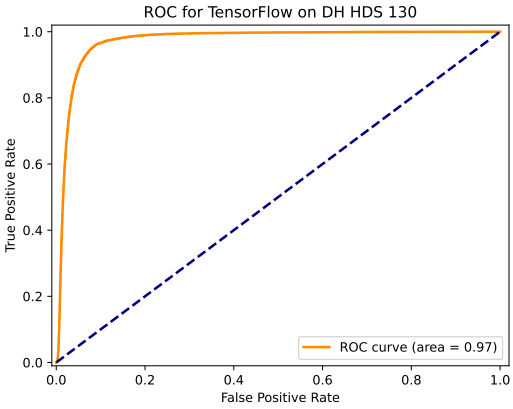
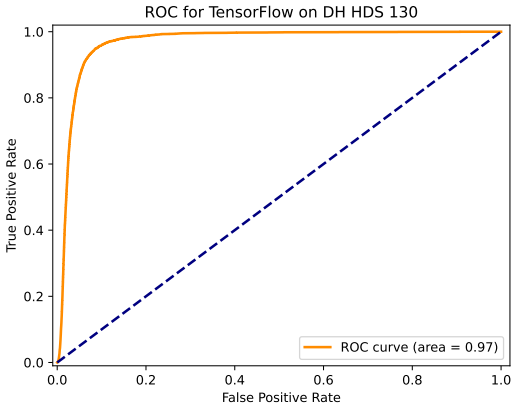


Figure A.11: NN parameters after 50 epochs when dropping features. This is training a dataset with 80% of all Z' DH HDS events.

We tested on the remaining 20% of the SM background events, as well as 20% of Z' DH HDS events where $m_{Z'} = 130$ GeV. The ROC scores for each network can be seen in Figure A.12. The validation plots can be seen in Figure A.13

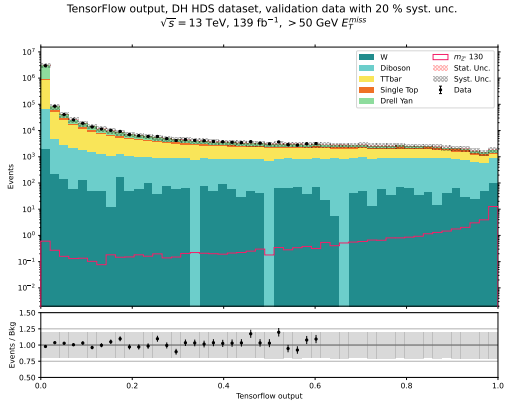


(a) When including new features

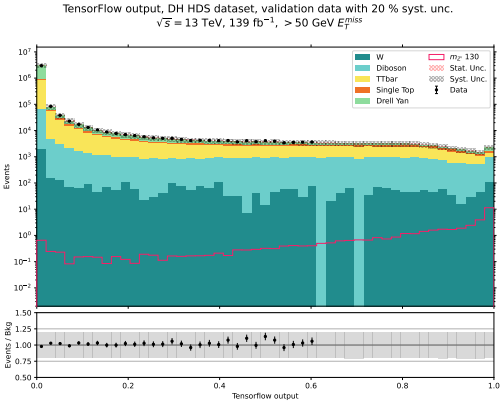


(b) When dropping features

Figure A.12: ROC plots for both padding methods. This is testing a dataset with 20% of the Z' DH HDS $m_{Z'} = 130$ GeV events.



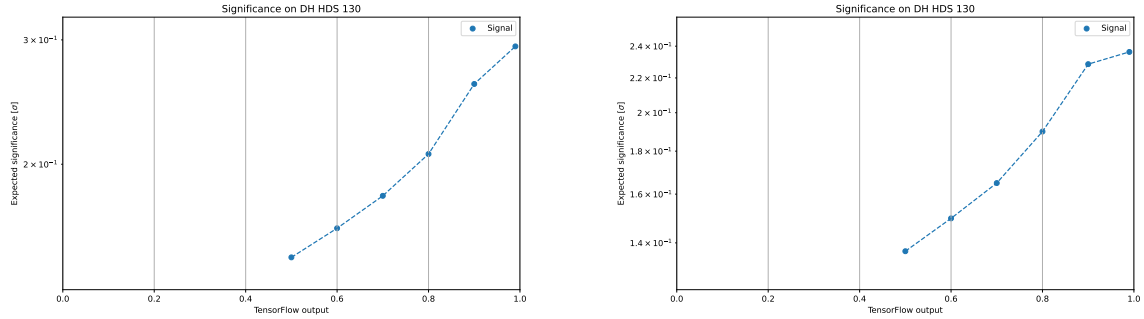
(a) When including new features



(b) When dropping features

Figure A.13: Validation plots for both padding methods. This is testing a dataset with 20% of the Z' DH HDS $m_{Z'} = 130$ GeV events.

As we can see the performance of both methods is the same, to check if the sensitivity increases more with the new padding method or not we can check the significance of each model. This can be seen in Figure A.14, which shows a slight improvement on the sensitivity of the network when using the features.



(a) When including new features *(b) When dropping features*

Figure A.14: Significance plots for both padding methods. This is testing a dataset with 20% of the Z' DH HDS $m_{Z'} = 130$ GeV events.

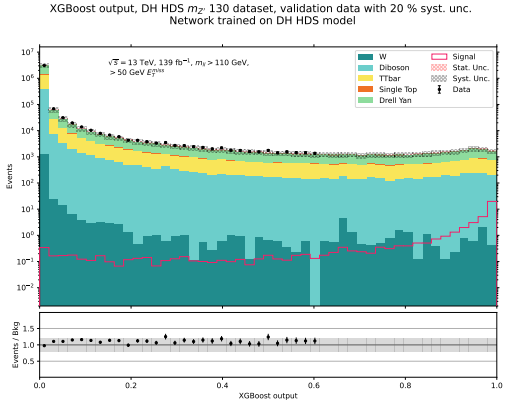
A.2 Boosted Decision Tree Training

As the only technique that needed to be tested for BDTs was the different weighting methods, we conducted these here. We only tested the weighting techniques where we only look at the positive weights, and where we scale the weights wrt. the sum of the weights over the sum over the absolute value of weights. To compare we also included the unweighted method (only balancing data). The hyperparameters used to train the different networks in this chapter were

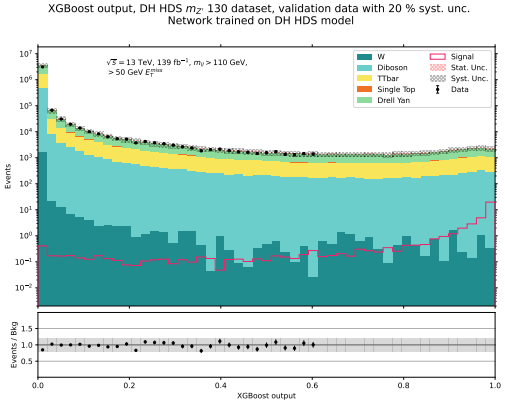
- L2-$\lambda = 10^{-5}$
- Number of trees = 200
- Depth of trees = 6
- Learning rate $\eta = 0.1$

A.2.1 Weights

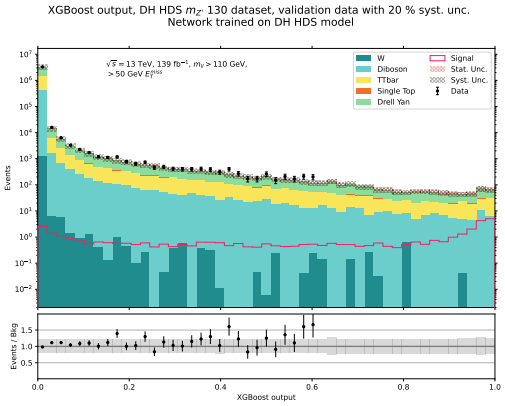
The results of the different weighting methods can be seen in Figure A.15.



(a) Using scaled absolute value of weights

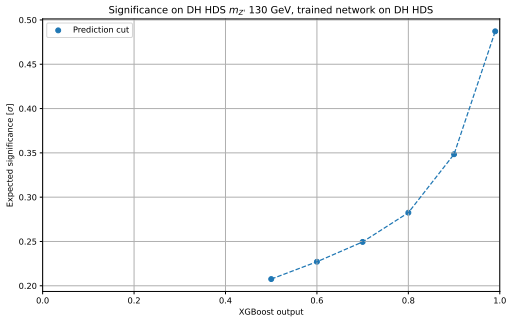


(b) Using only positive weights

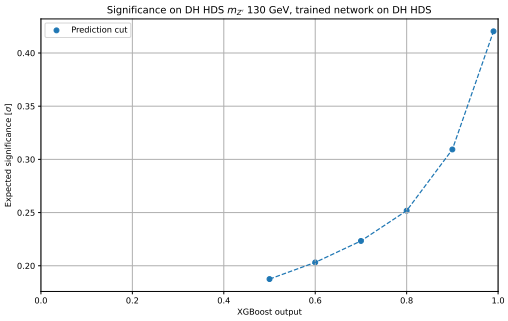


(c) Using no weights

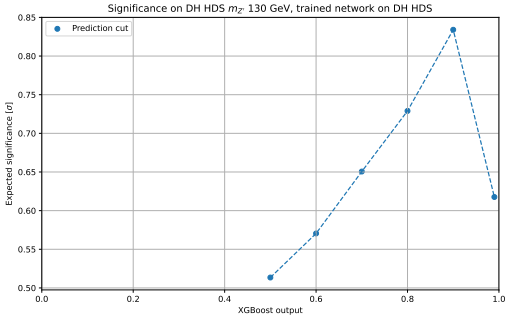
Figure A.15: Difference when using different weighting methods. All networks were trained using the balancing method explained in Section 7.3.1



(a) Using scaled absolute value of weights



(b) Using only positive weights



(c) Using no weights

Figure A.16: Difference when using different weighting methods. All networks were trained using the balancing method explained in Section 7.3.1

A.3 Discussion (draft)

Comment on lack of time and DNN thoughts.

The last thing to be noted is that having a DNN completely removes the possibility of combining the results of multiple networks trained on a single model, as the imbalance becomes too much for the network to see anything. A solution to this however, is that instead of combining the results of multiple networks trained on a singular model, one could try the Parametrized NN approach used by Baldi et al. [29], which could potentially avoid the imbalance problem, but this is proposed as a plausible new research project due to time constrain on this thesis.

Comment on fully converting to XG-Boost.

XGBoost » TensorFlow

Appendix B

Algorithms for BDTs and NNs

Algorithm B.1: Neural network definition using TensorFlow

```
import tensorflow as tf
from tensorflow.keras import layers

def Neural_Network(inputsize, n_layers, n_neuron, eta, lamda):

    model=tf.keras.Sequential()

    for i in range(n_layers):
        if (i==0):
            model.add(layers.Dense(n_neuron, activation='relu', kernel_regularizer=
             tf.keras.regularizers.l2(lamda), input_dim=inputsize))
        else:
            model.add(layers.Dense(n_neuron, activation='relu', kernel_regularizer=
                    tf.keras.regularizers.l2(lamda)))

    model.add(layers.Dense(1,activation='sigmoid'))

    opt=tf.optimizers.SGD(learning_rate=eta) # or Adam!

    model.compile(loss=tf.losses.BinaryCrossentropy(),
                optimizer=opt,
                metrics = [tf.keras.metrics.BinaryAccuracy()])
    return model
```

Algorithm B.2: Boosted Decision Tree definition using XGBoost

```
import xgboost as xgb

Boosted_Decision_Tree = xgb.XGBClassifier(
             max_depth,
             use_label_encoder=False,
             n_estimators,
             learning_rate,
             reg_lambda,
             predictor = 'cpu_predictor',
             tree_method = 'hist',
             scale_pos_weight = sow_bkg/sow_sig,
             objective = 'binary:logistic',
             eval_metric = 'auc',
             min_child_weight = 1,
             missing = -999,
             random_state = 42,
```

```
17                  verbosity = 1)
18
```

Appendix C

Kinematical variables' distribution in control region

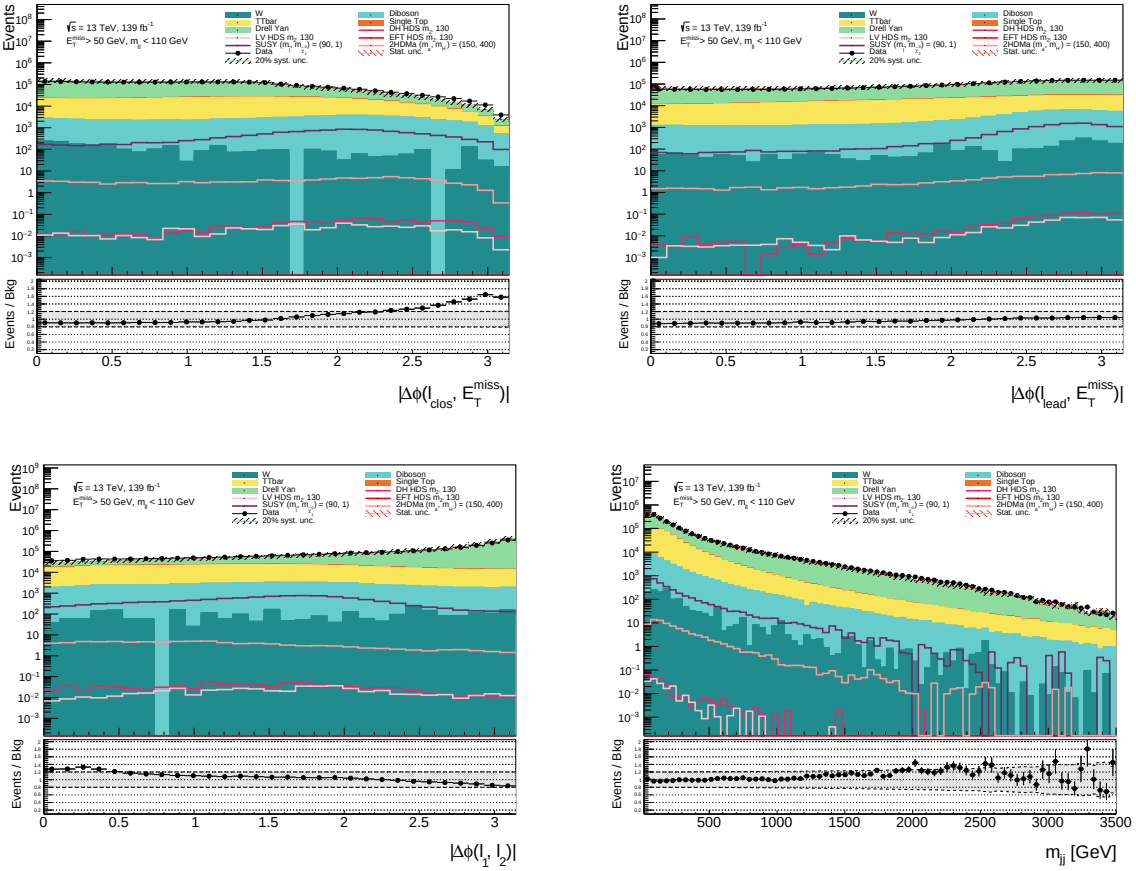


Figure C.1: $\Delta\phi(l_c, E_T^{miss})$, $\Delta\phi(l_l, E_T^{miss})$, $\Delta\phi(l_1, l_2)$ and m_{jj} distribution in the control region. For the 2HDM + a we only include the distribution for $\tan\beta = 1$.

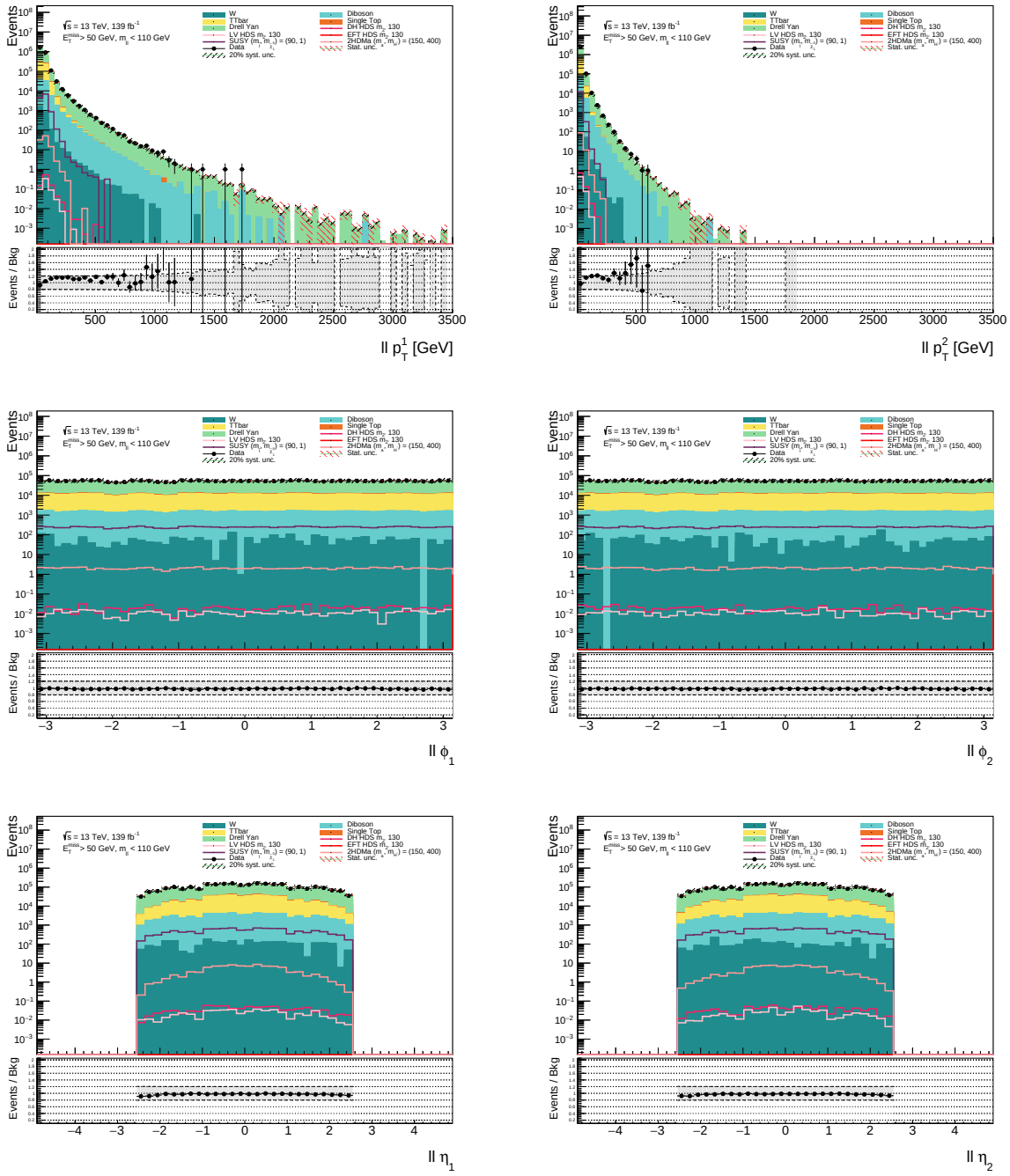


Figure C.2: Leptons p_T, ϕ and η distribution in the control region. For the 2HDM + a we only include the distribution for $\tan\beta = 1$.

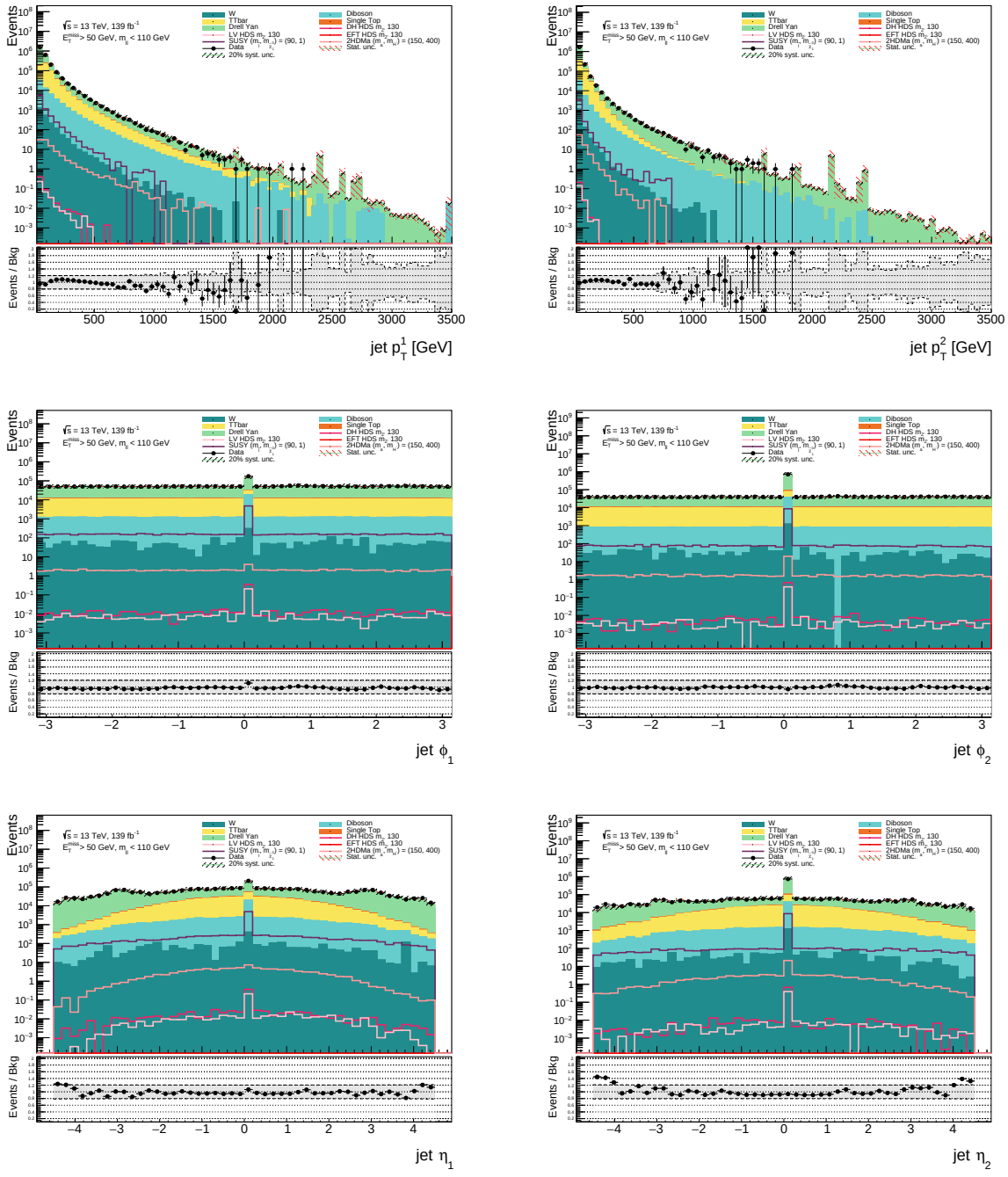


Figure C.3: Leading jets p_T, ϕ and η distribution in the control region. For the 2HDM + a we only include the distribution for $\tan\beta = 1$.

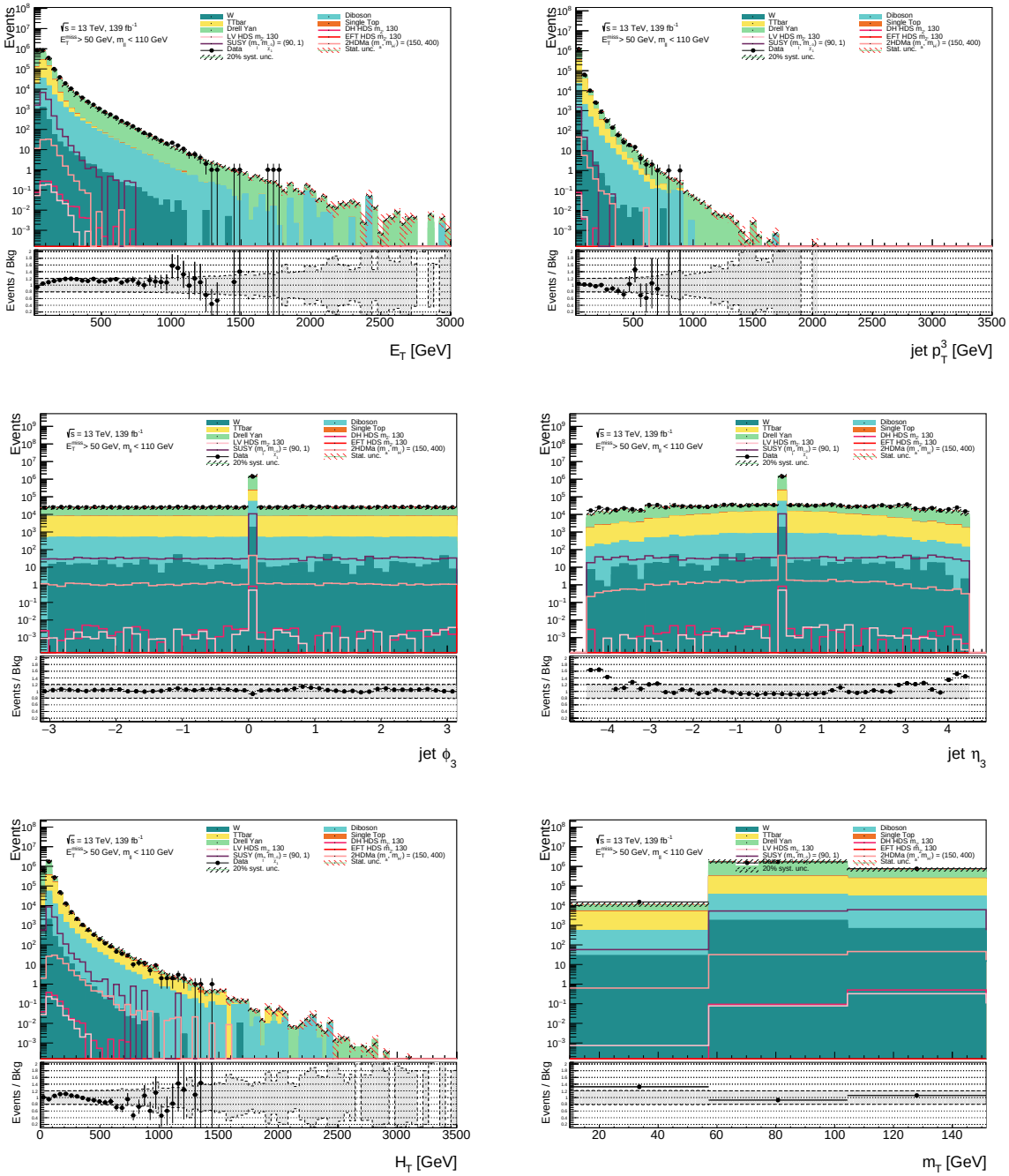


Figure C.4: p_T, ϕ and η of third leading jet and the dilepton pairs E_T, H_T and m_T distribution in the control region. For the 2HDM + a we only include the distribution for $\tan\beta = 1$.

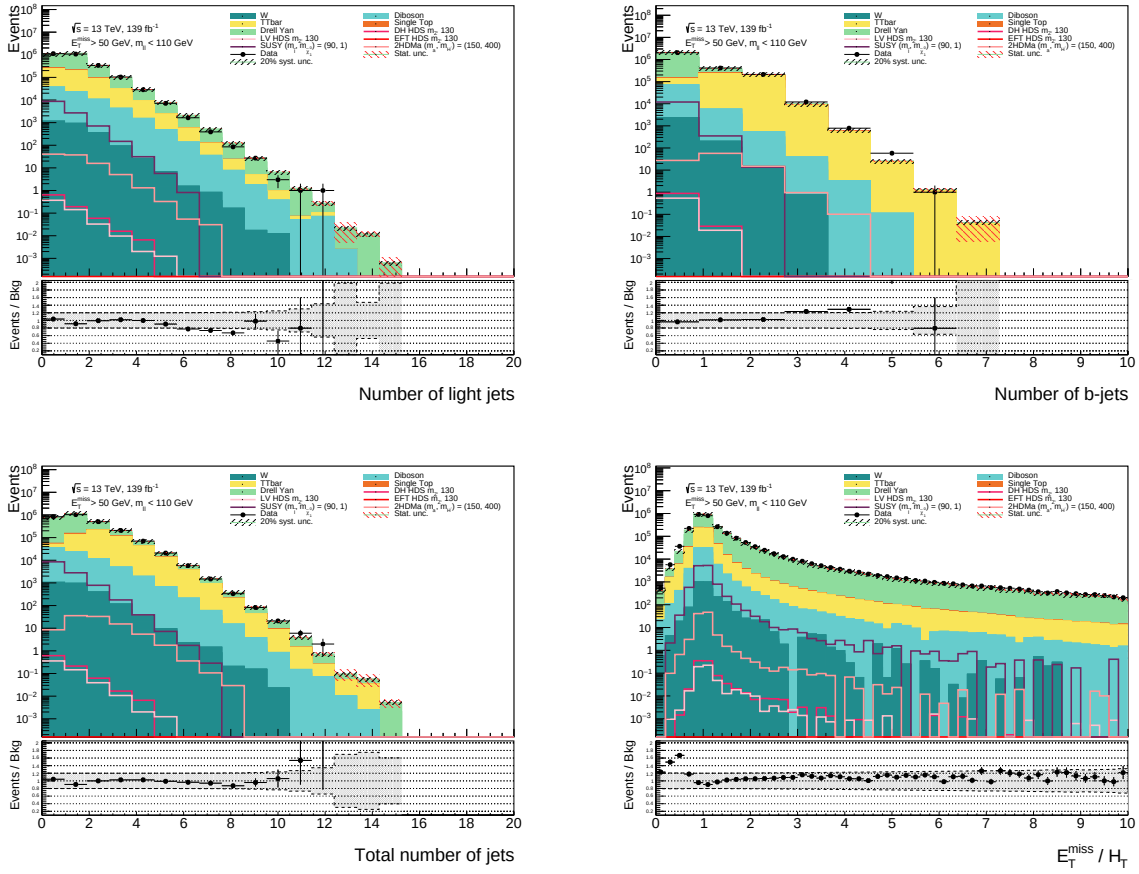


Figure C.5: Number of light, b- and total jets and E_T^{miss}/H_T distribution in the control region. For the 2HDM + a we only include the distribution for $\tan\beta = 1$.

Appendix D

Data and MC agreement of jets in preselection region

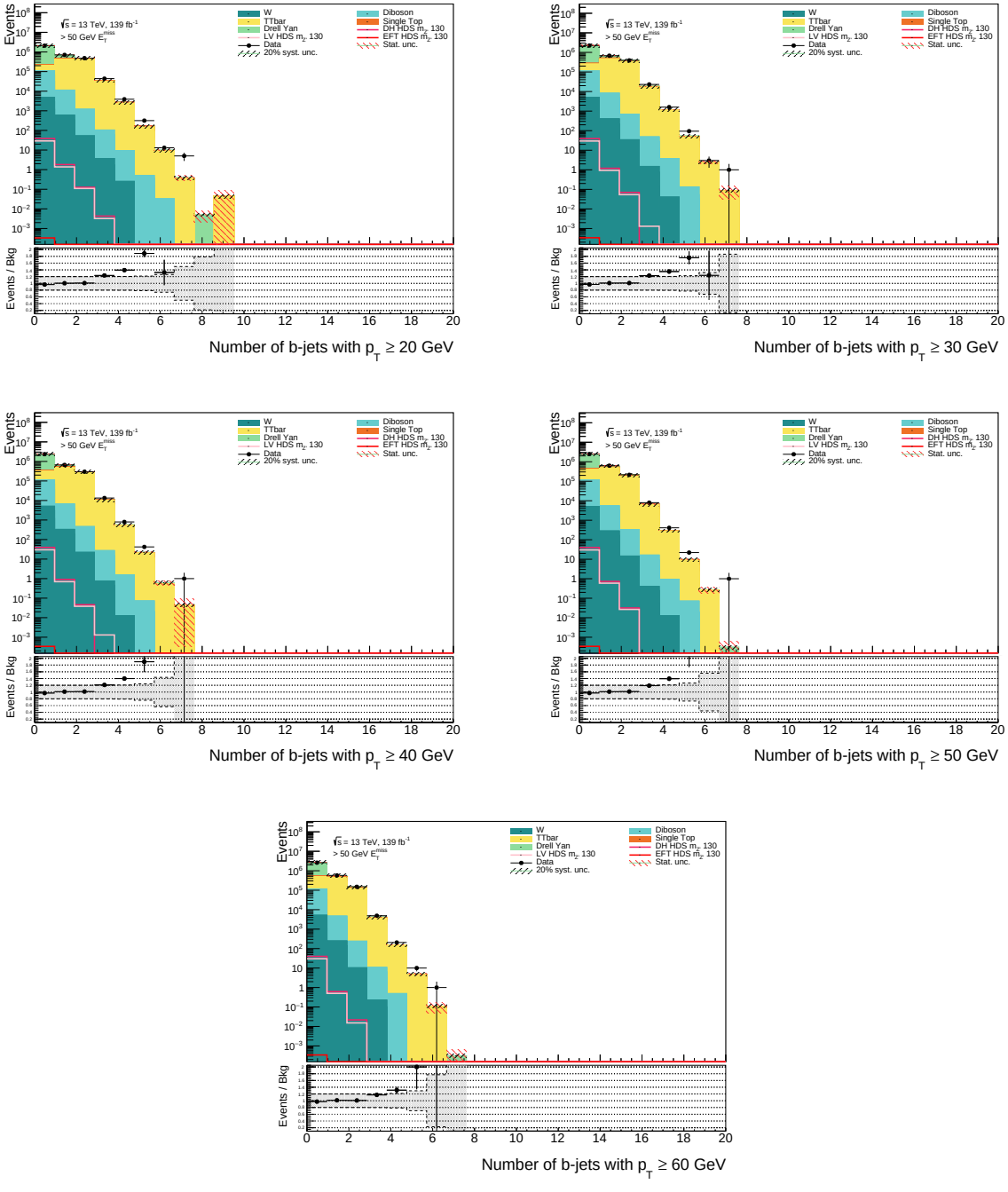


Figure D.1: Data and MC agreement on number of b- jets with different p_T cuts in CR.

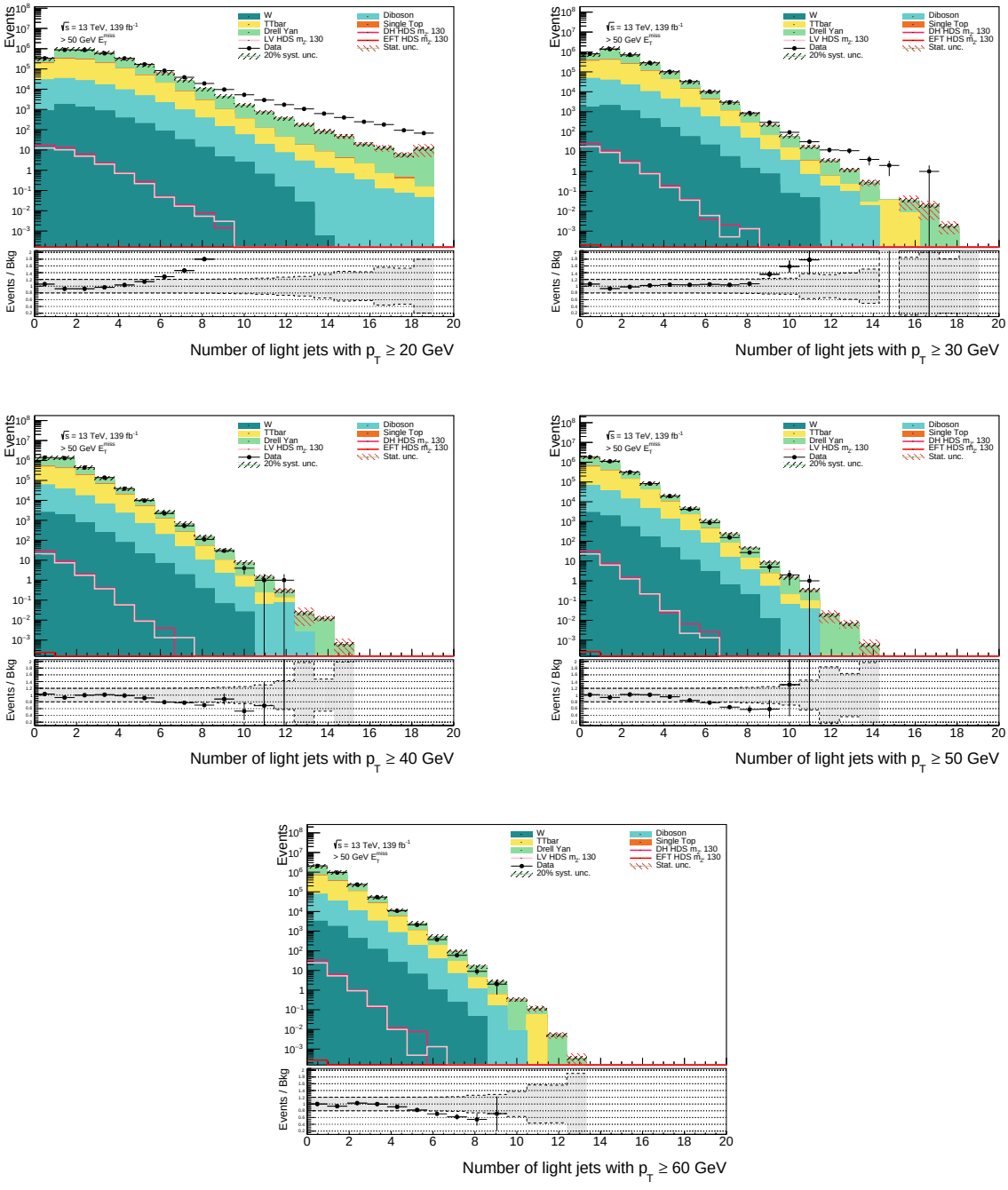


Figure D.2: Data and MC agreement on number of light jets with different p_T cuts in CR.

Appendix E

Distribution of new features to avoid padding

Soon to come

Appendix F

Model dependent approach

F.1 Dark Higgs Heavy Dark Sector

F.2 Dark Higgs Light Dark Sector

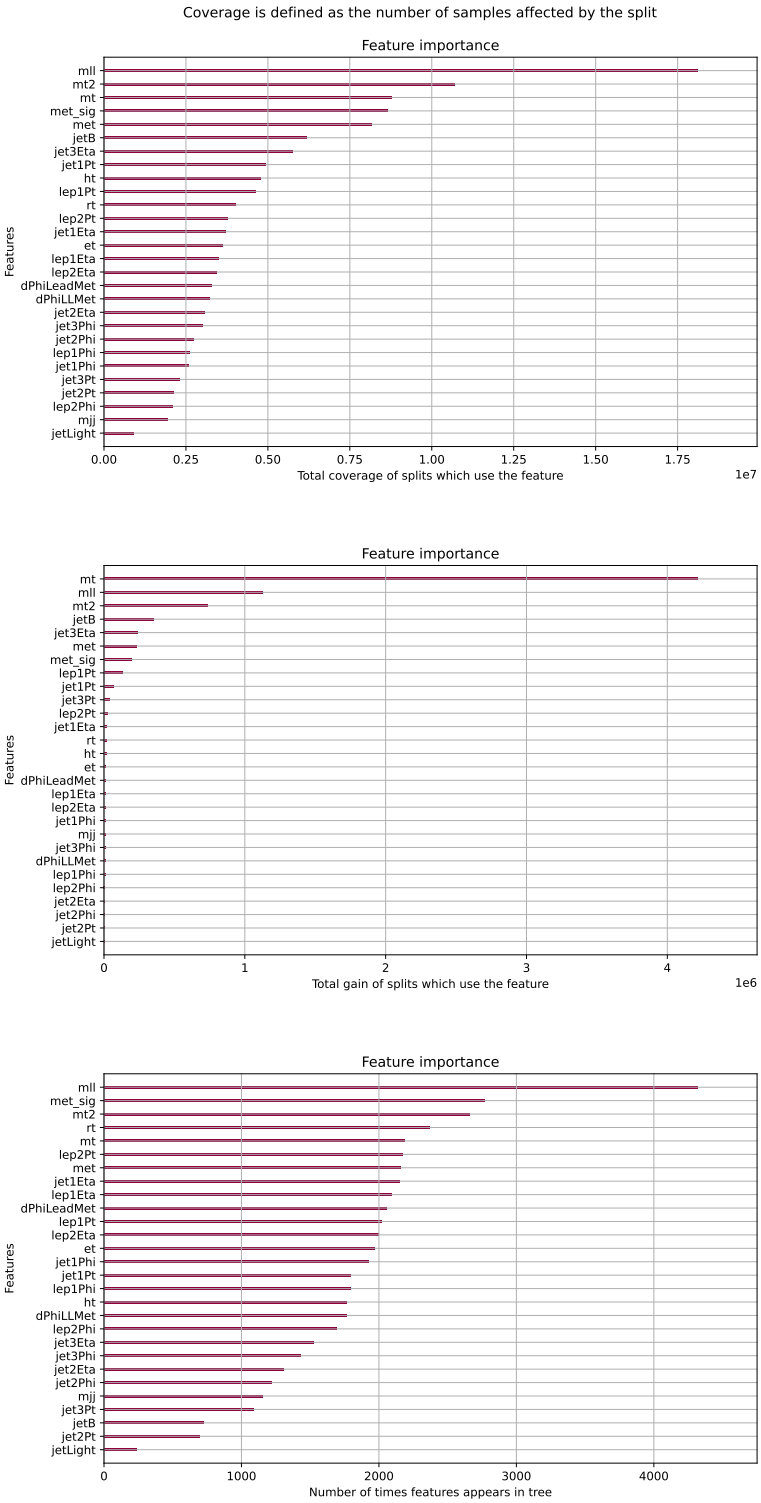


Figure F.1: Full feature importance for network trained on Z' DH HDS

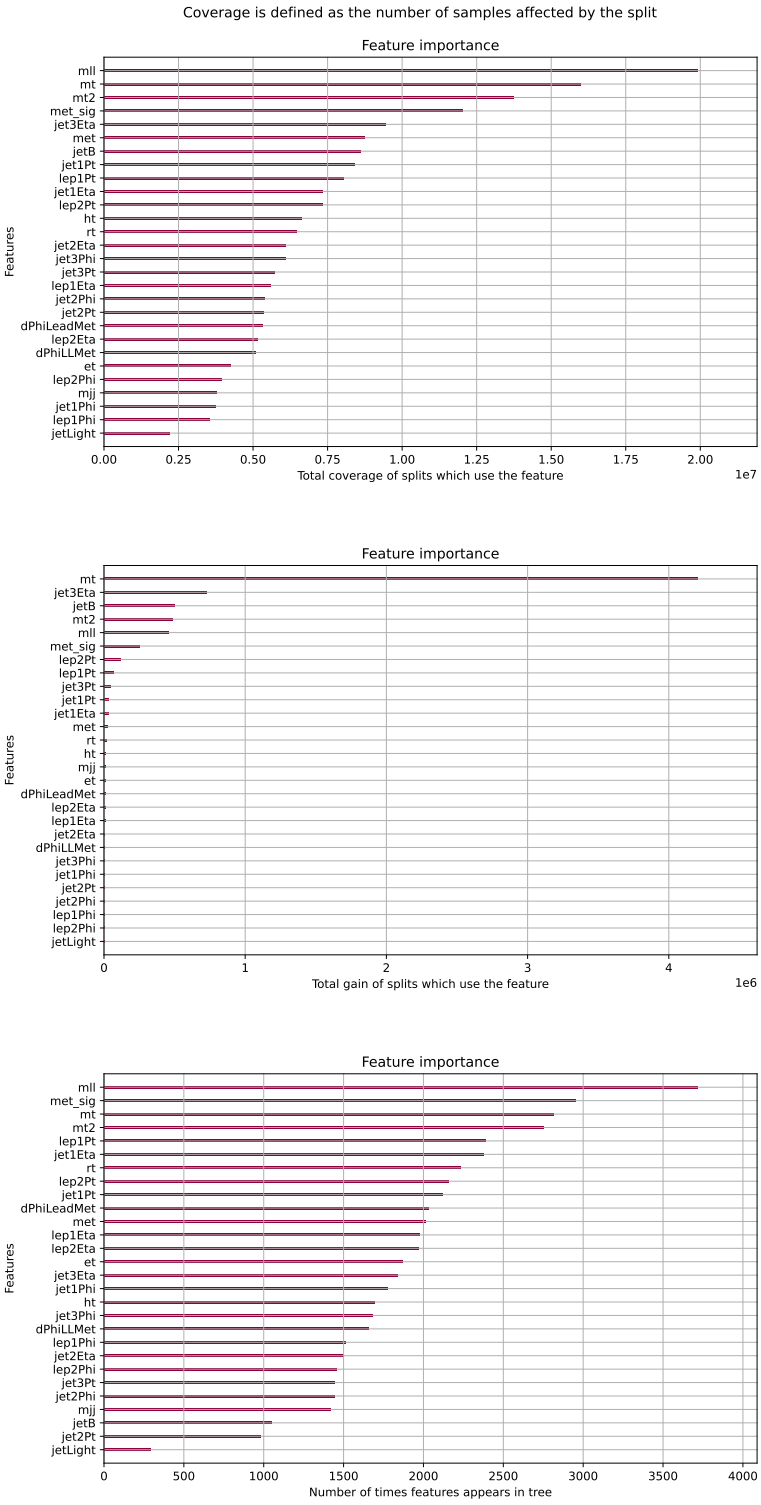


Figure F.2: Feature importance for network trained on Z' DH LDS

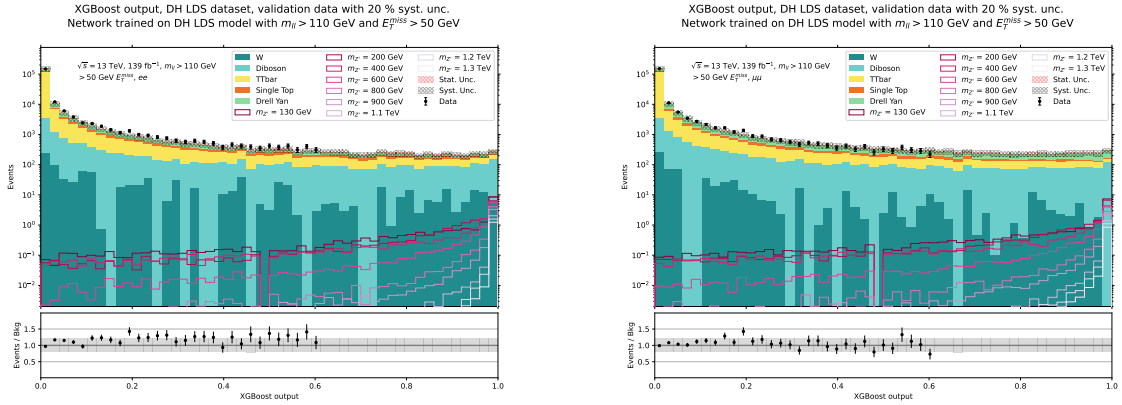


Figure F.3: Validation plots for network trained on Z' DH LDS

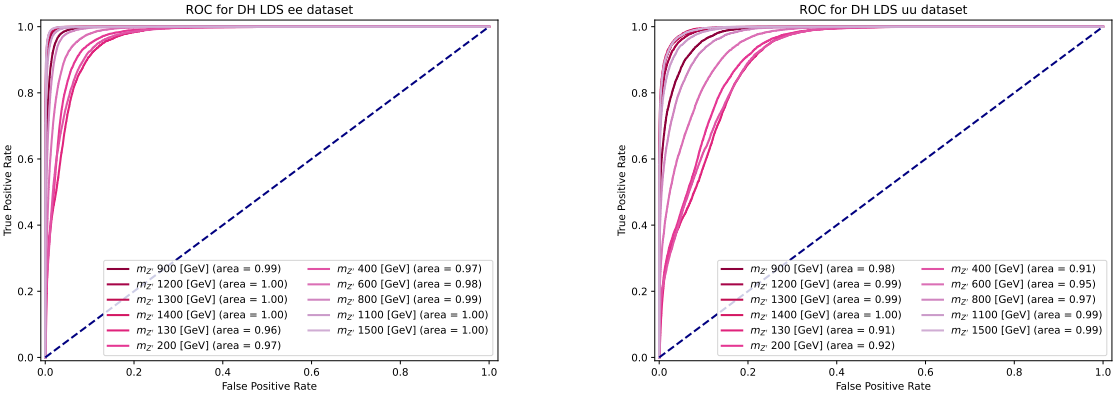


Figure F.4: ROC plots for every Z' mass point on network trained on Z' DH LDS

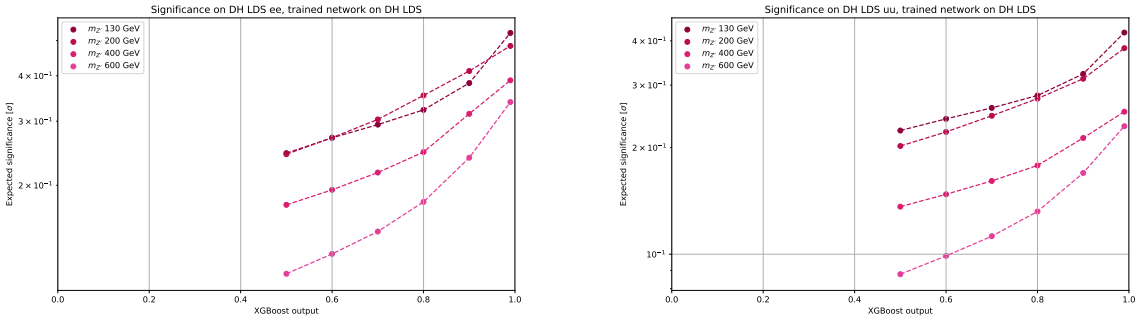


Figure F.5: Expected significance plots for Z' mass points on network trained on Z' DH LDS

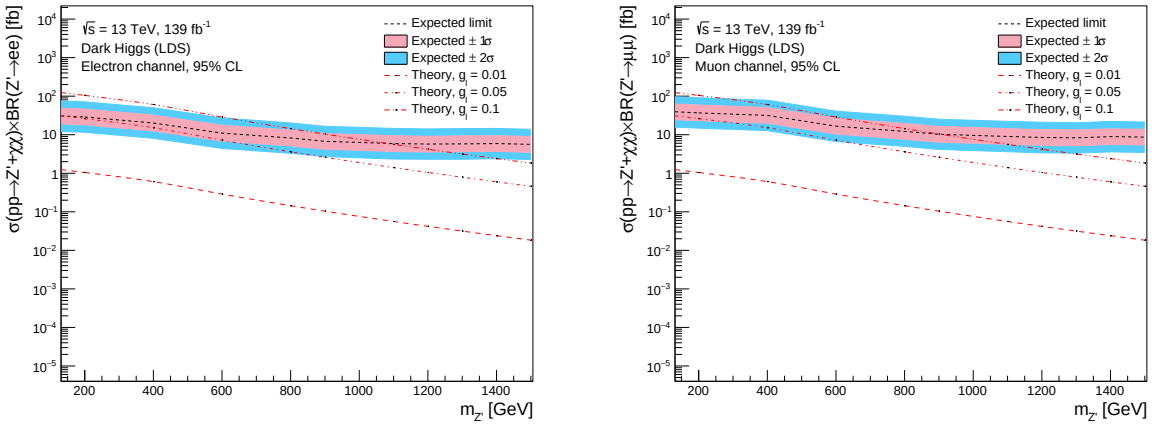


Figure F.6: Mass exclusion limits of ee and $\mu\mu$ channel for all Z' DH LDS model

F.3 Light Vector Heavy Dark Sector

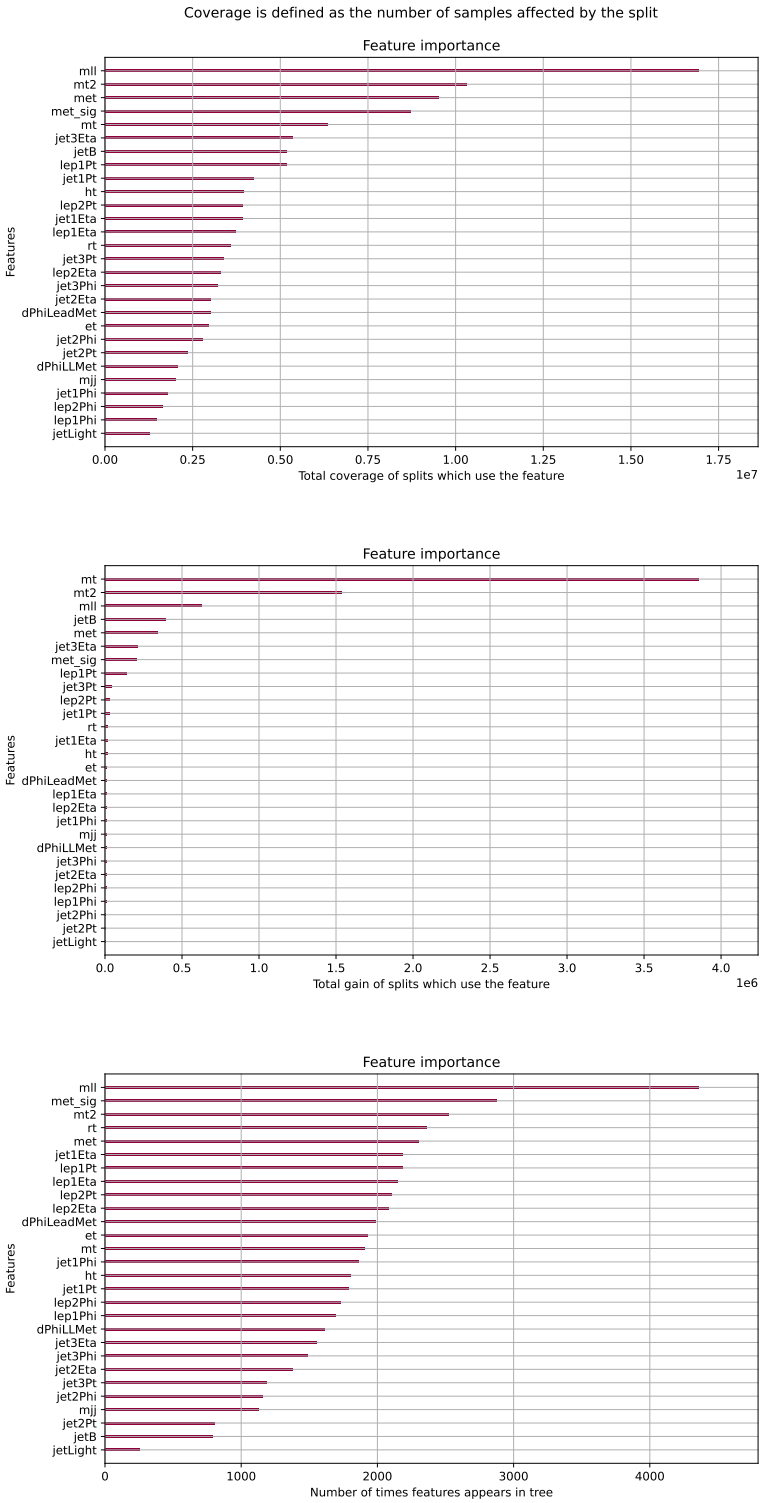


Figure F.7: Feature importance for network trained on Z' LV HDS

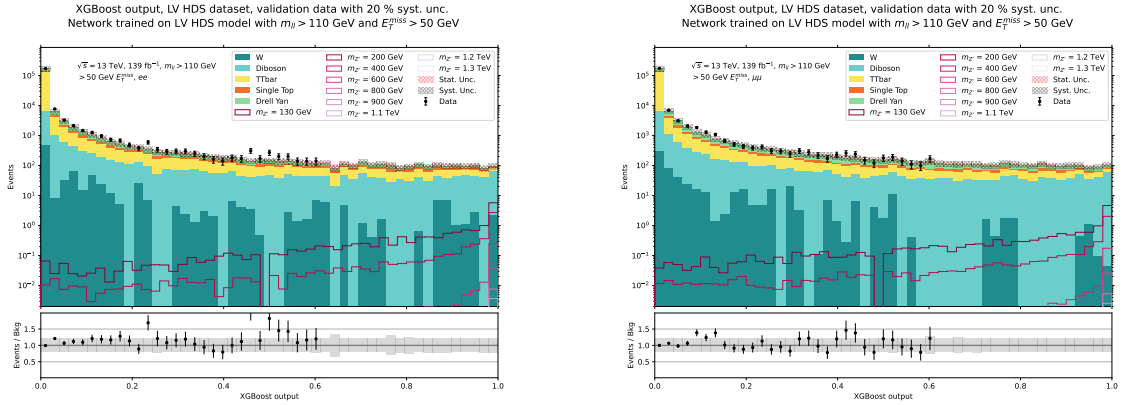


Figure F.8: Validation plots for network trained on Z' LV HDS

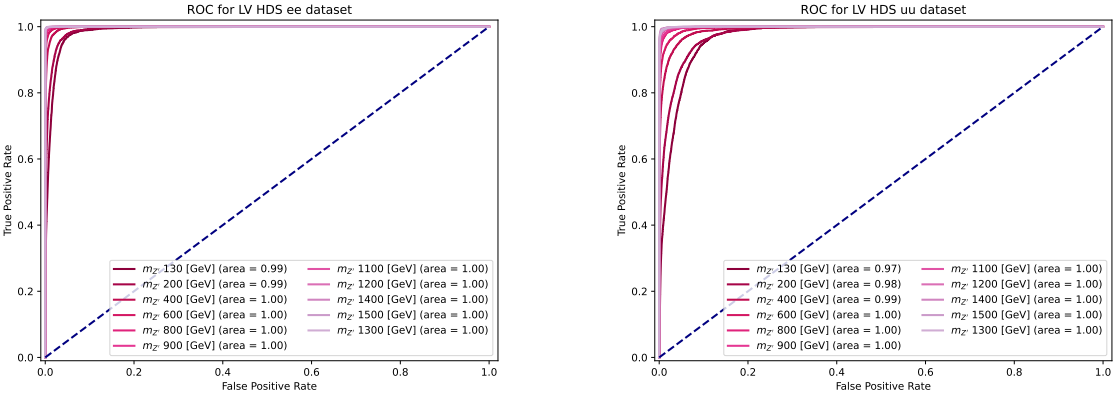


Figure F.9: ROC plots for every Z' mass point on network trained on Z' LV HDS

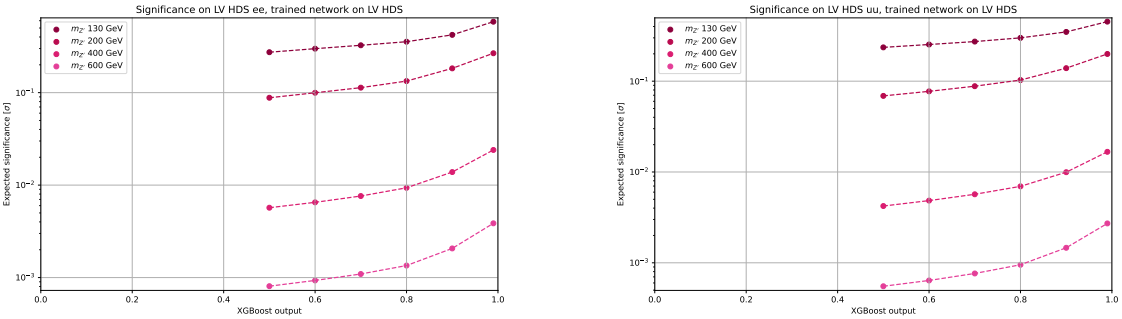


Figure F.10: Expected significance plots for Z' mass points on network trained on Z' LV HDS

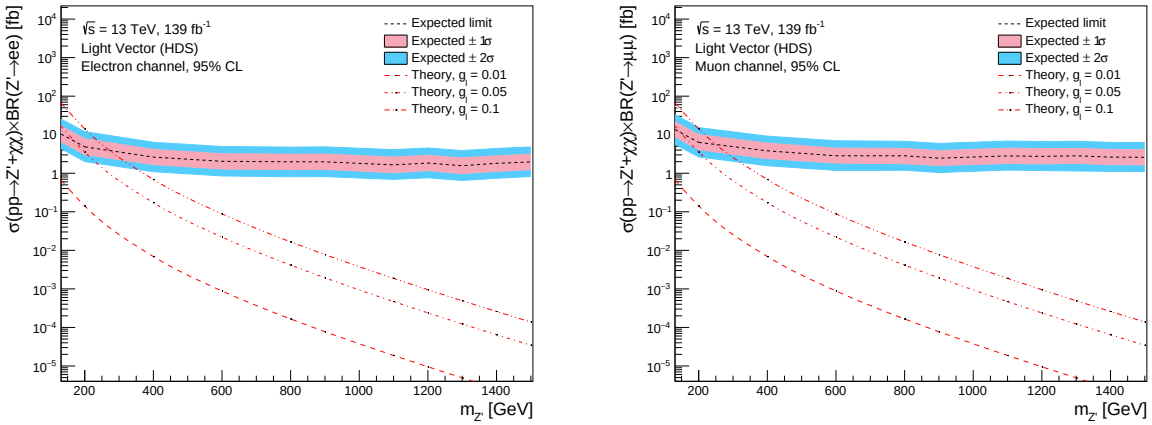


Figure F.11: Mass exclusion limits of ee and $\mu\mu$ channel for all Z' LV HDS model

F.4 Light Vector Light Dark Sector

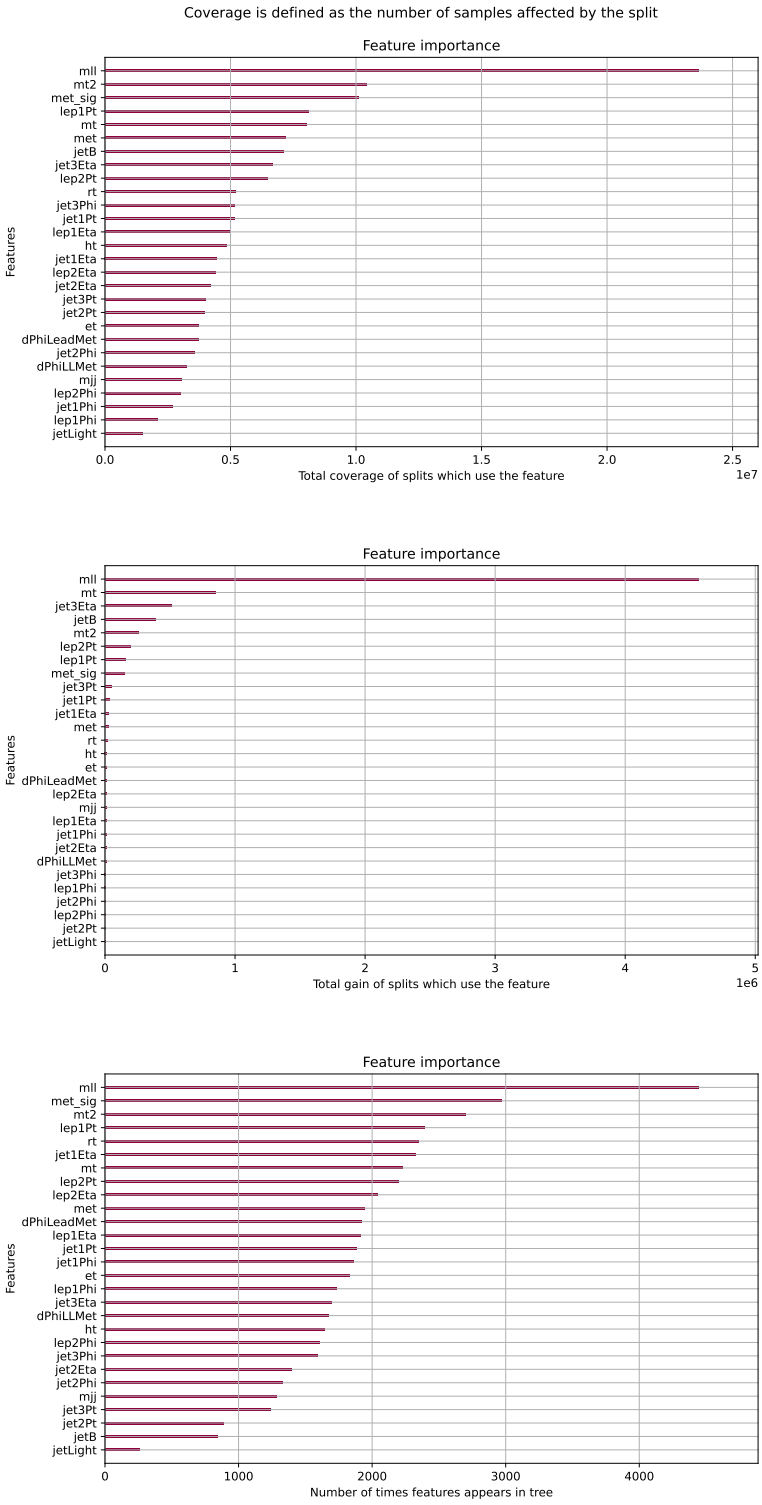


Figure F.12: Feature importance for network trained on Z' LV LDS

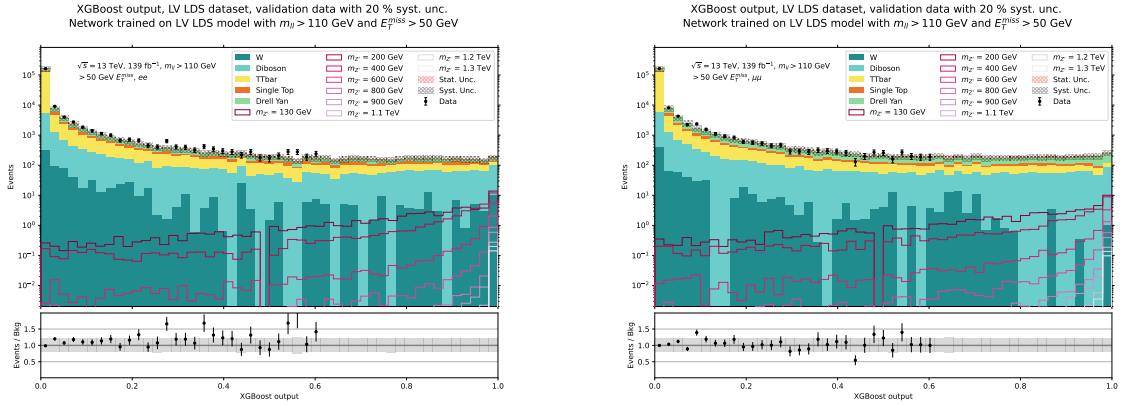


Figure F.13: Validation plots for network trained on Z' LV LDS

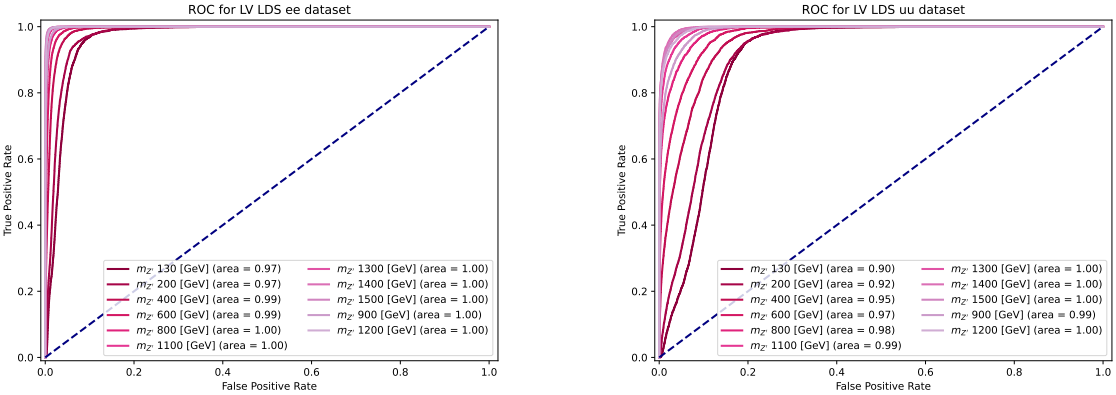


Figure F.14: ROC plots for every Z' mass point on network trained on Z' LV LDS

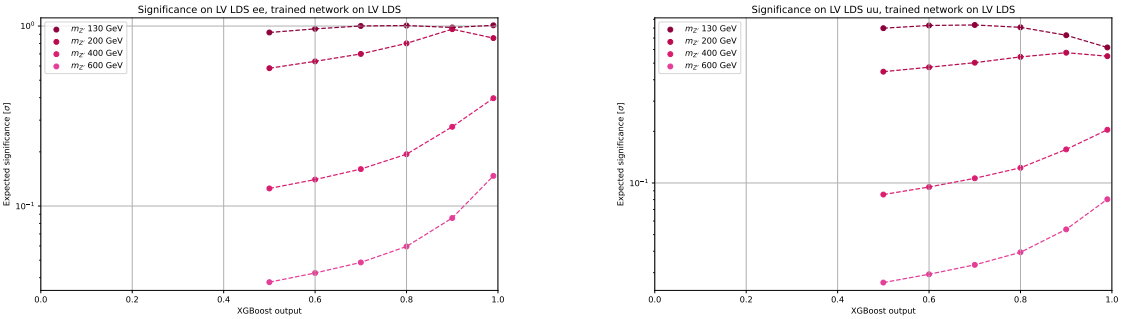


Figure F.15: Expected significance plots for Z' mass points on network trained on Z' LV LDS

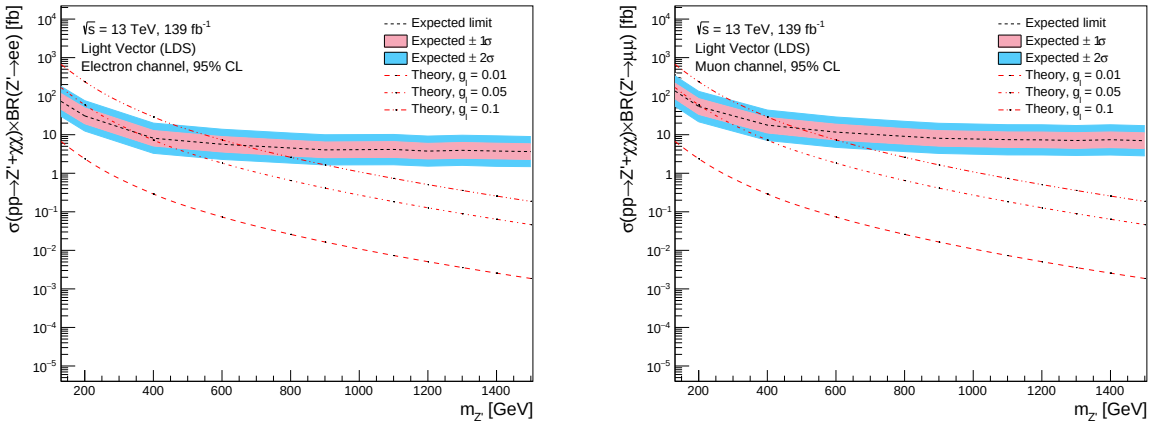


Figure F.16: Mass exclusion limits of ee and $\mu\mu$ channel for all Z' LV LDS model

F.5 Effective Field Theory Heavy Dark Sector

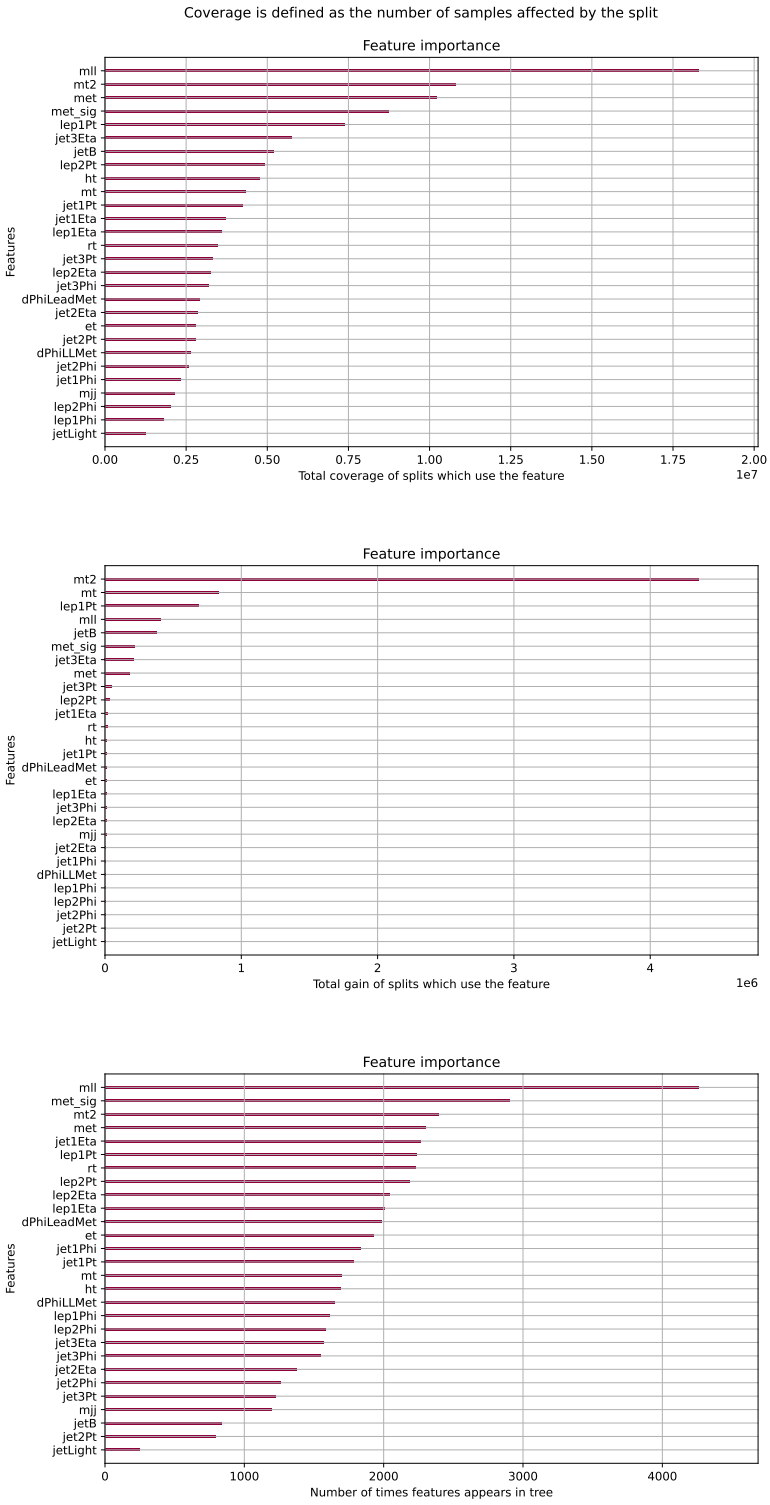


Figure F.17: Feature importance for network trained on Z' EFT HDS

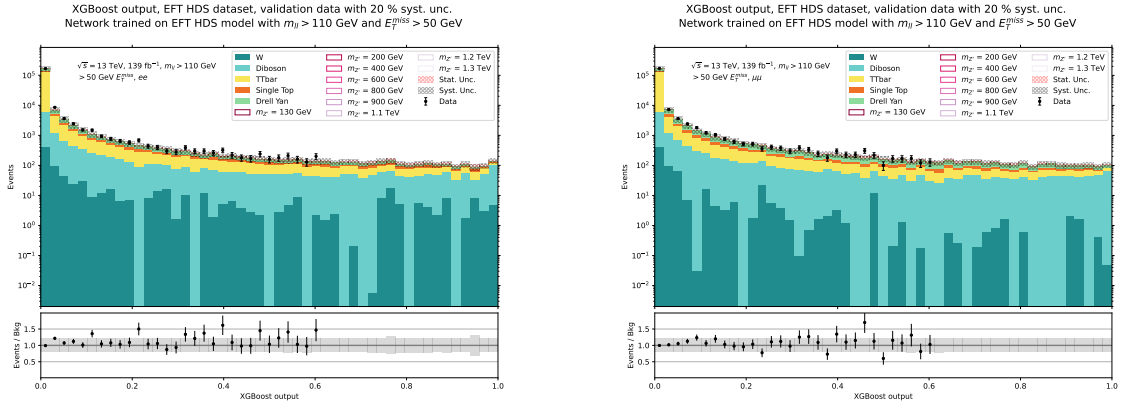


Figure F.18: Validation plots for network trained on Z' EFT HDS

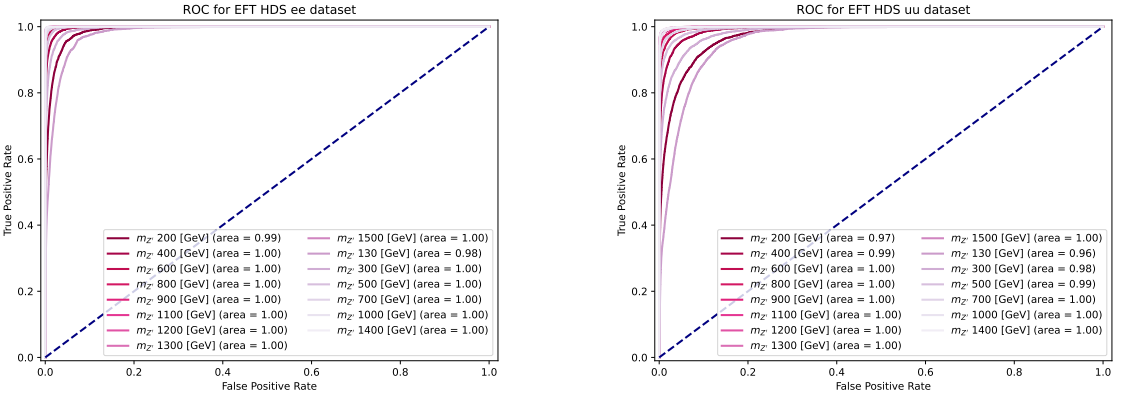


Figure F.19: ROC plots for every Z' mass point on network trained on Z' EFT HDS

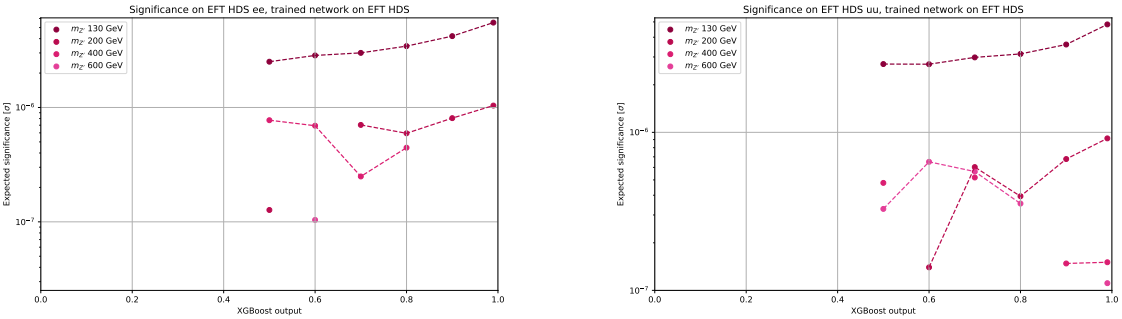


Figure F.20: Expected significance plots for Z' mass points on network trained on Z' EFT HDS

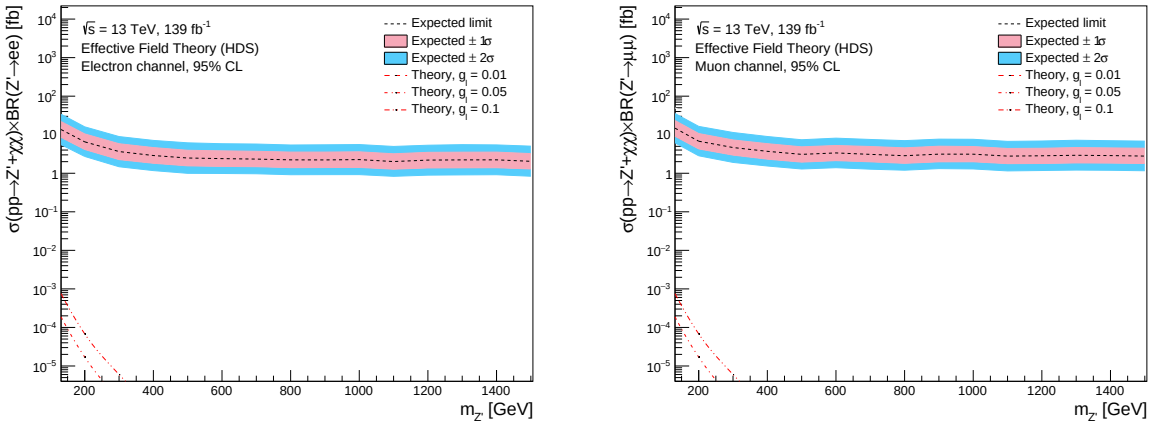


Figure F.21: Mass exclusion limits of ee and $\mu\mu$ channel for all Z' EFT HDS model

F.6 Effective Field Theory Light Dark Sector

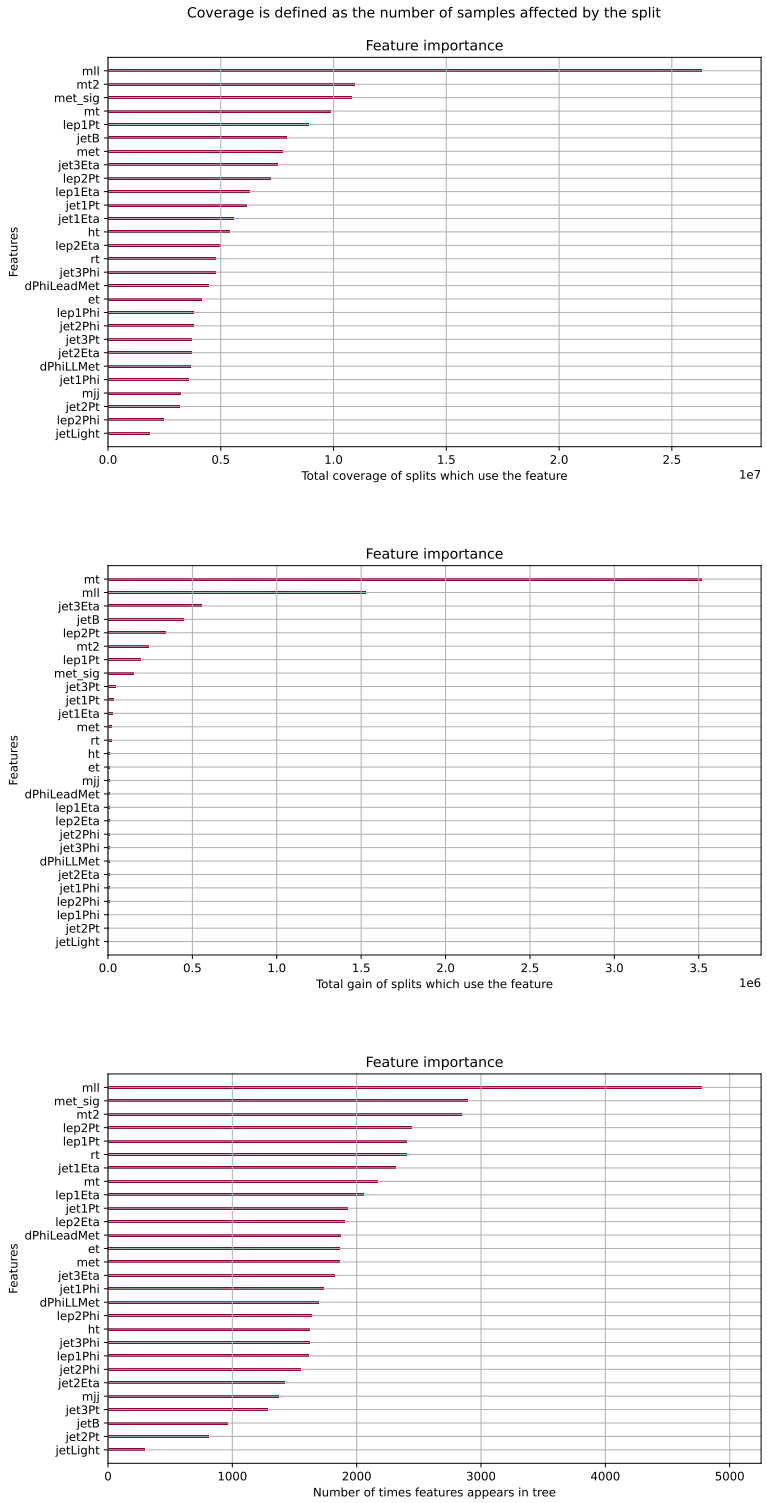


Figure F.22: Feature importance for network trained on Z' EFT LDS

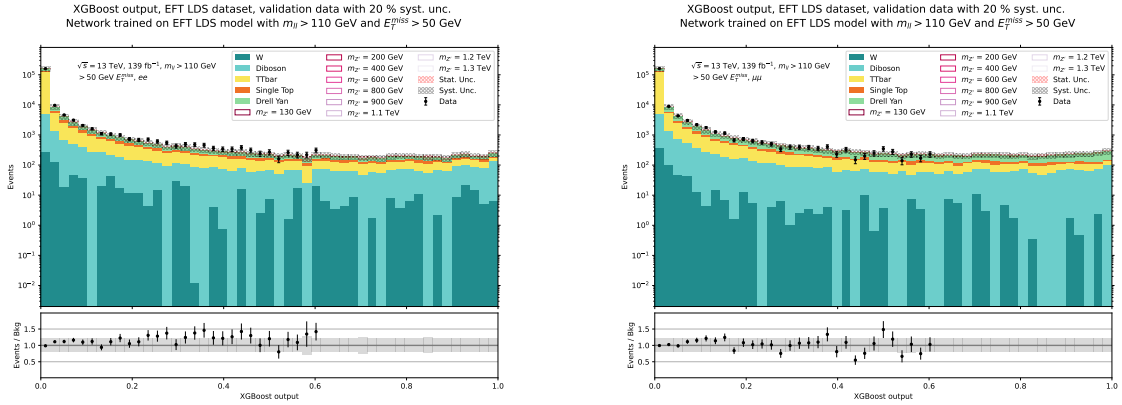


Figure F.23: Validation plots for network trained on Z' EFT LDS

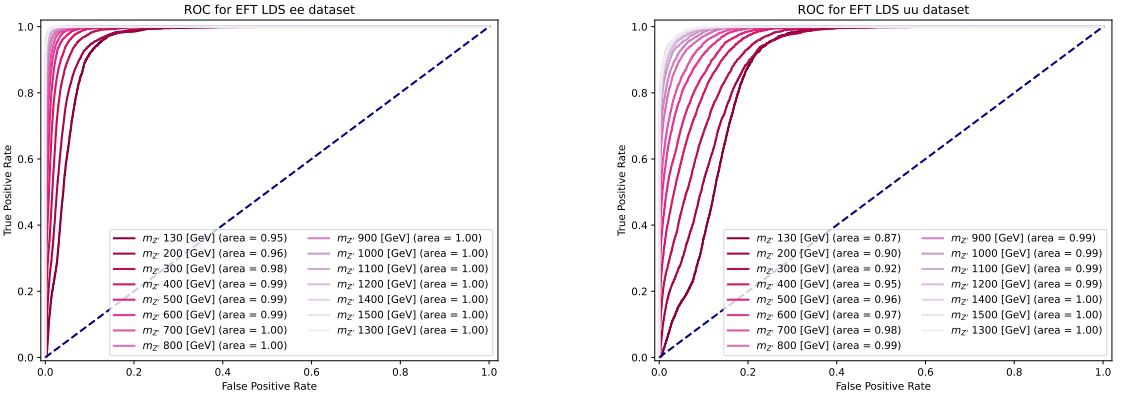


Figure F.24: ROC plots for every Z' mass point on network trained on Z' EFT LDS

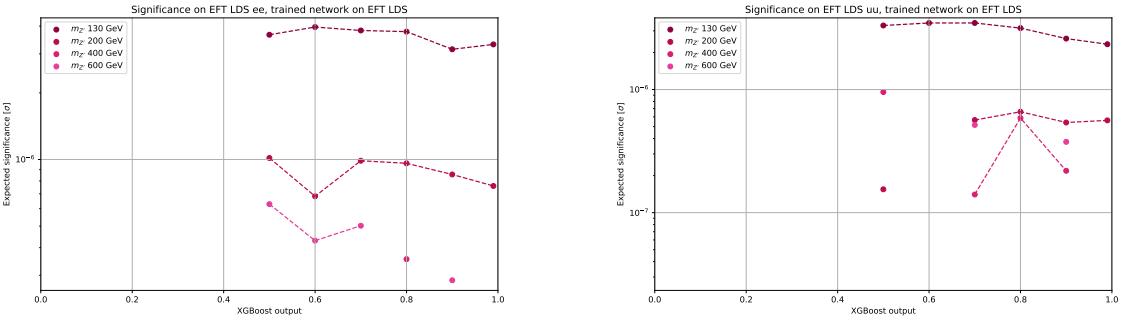


Figure F.25: Expected significance plots for Z' mass points on network trained on Z' EFT LDS

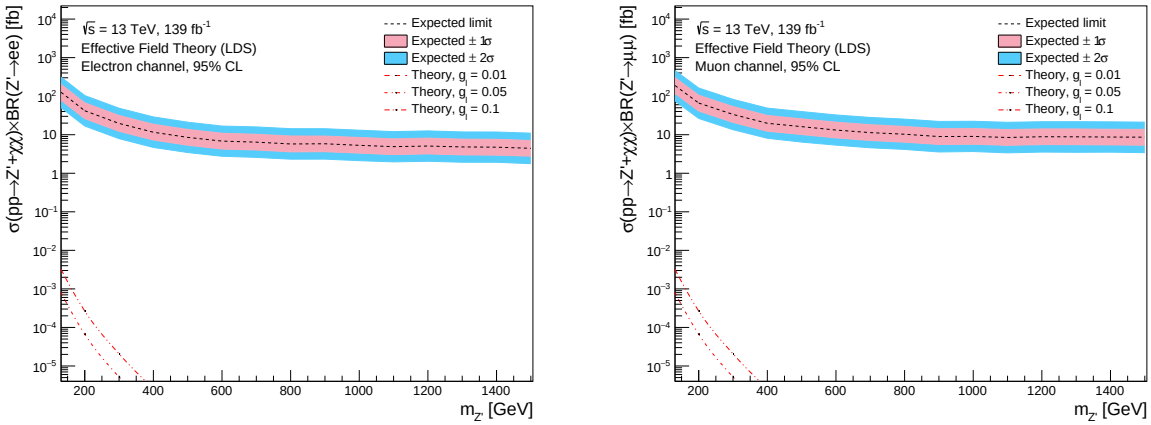


Figure F.26: Mass exclusion limits of ee and $\mu\mu$ channel for all Z' EFT LDS model

Appendix G

Model independent approach

G.1 Dark Higgs Light Dark Sector

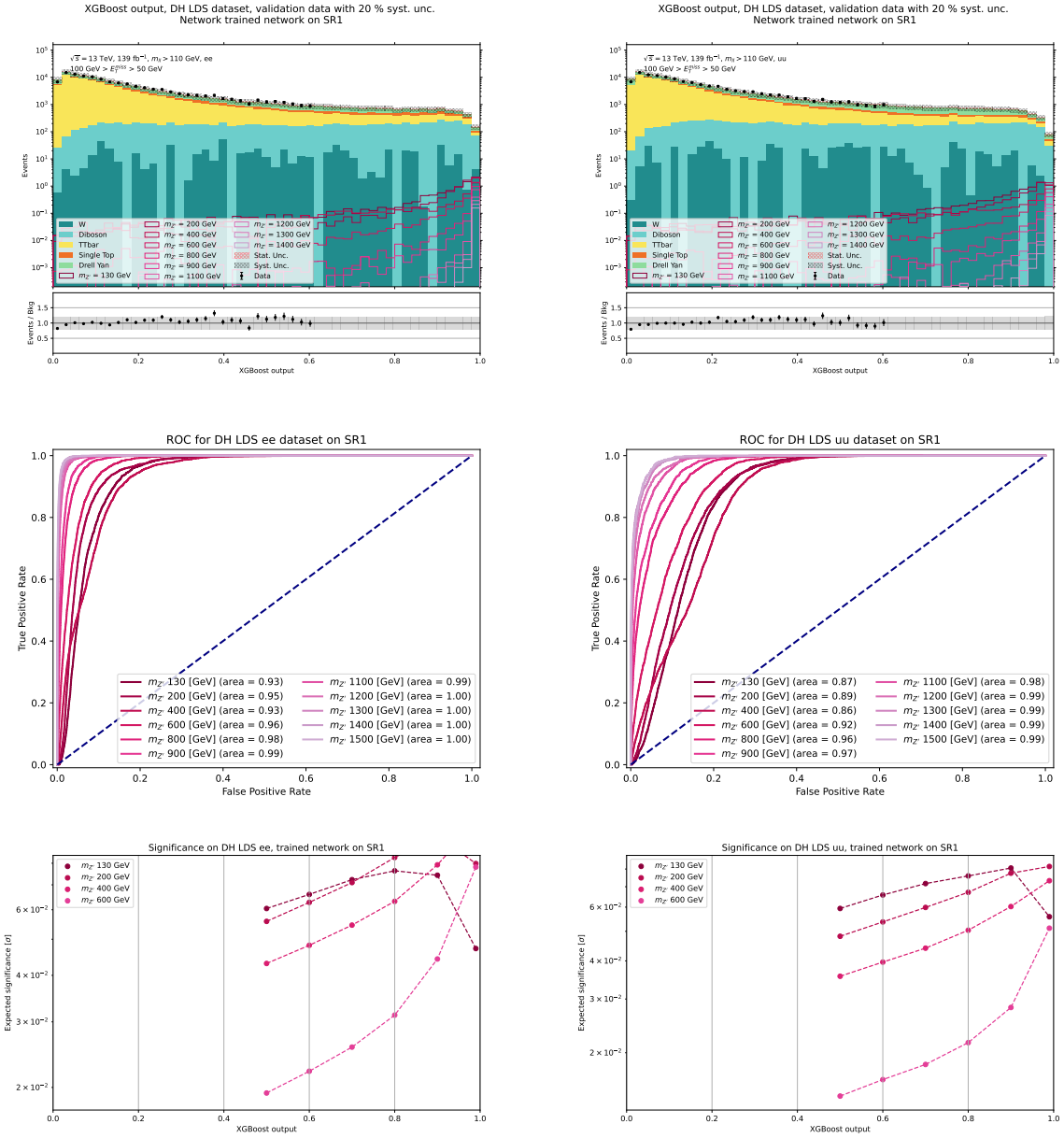


Figure G.1: XGBoost results for DH LDS model on ee and μμ channel in SR1

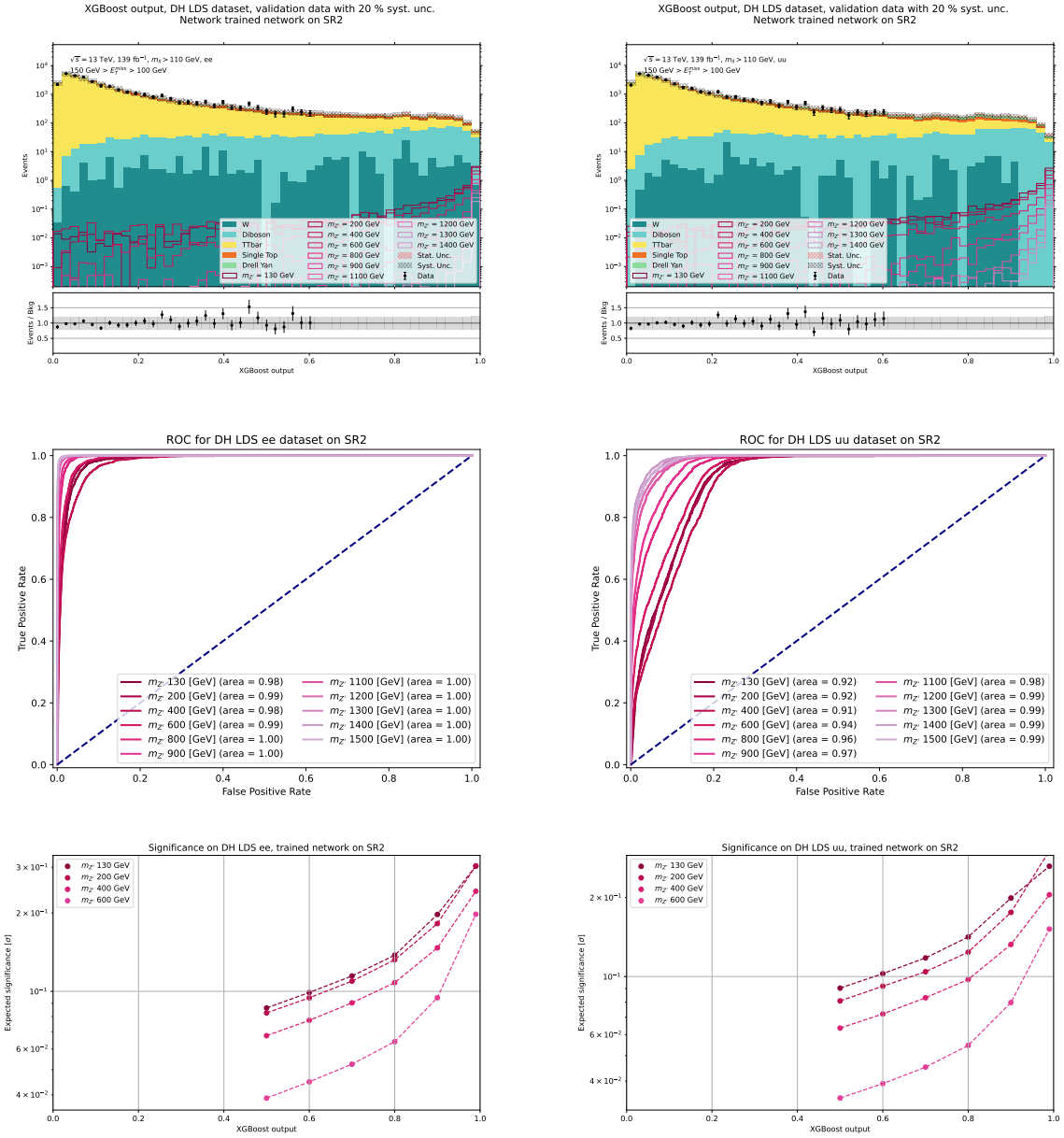


Figure G.2: XGBoost results for DH LDS model on ee and μμ channel in SR2

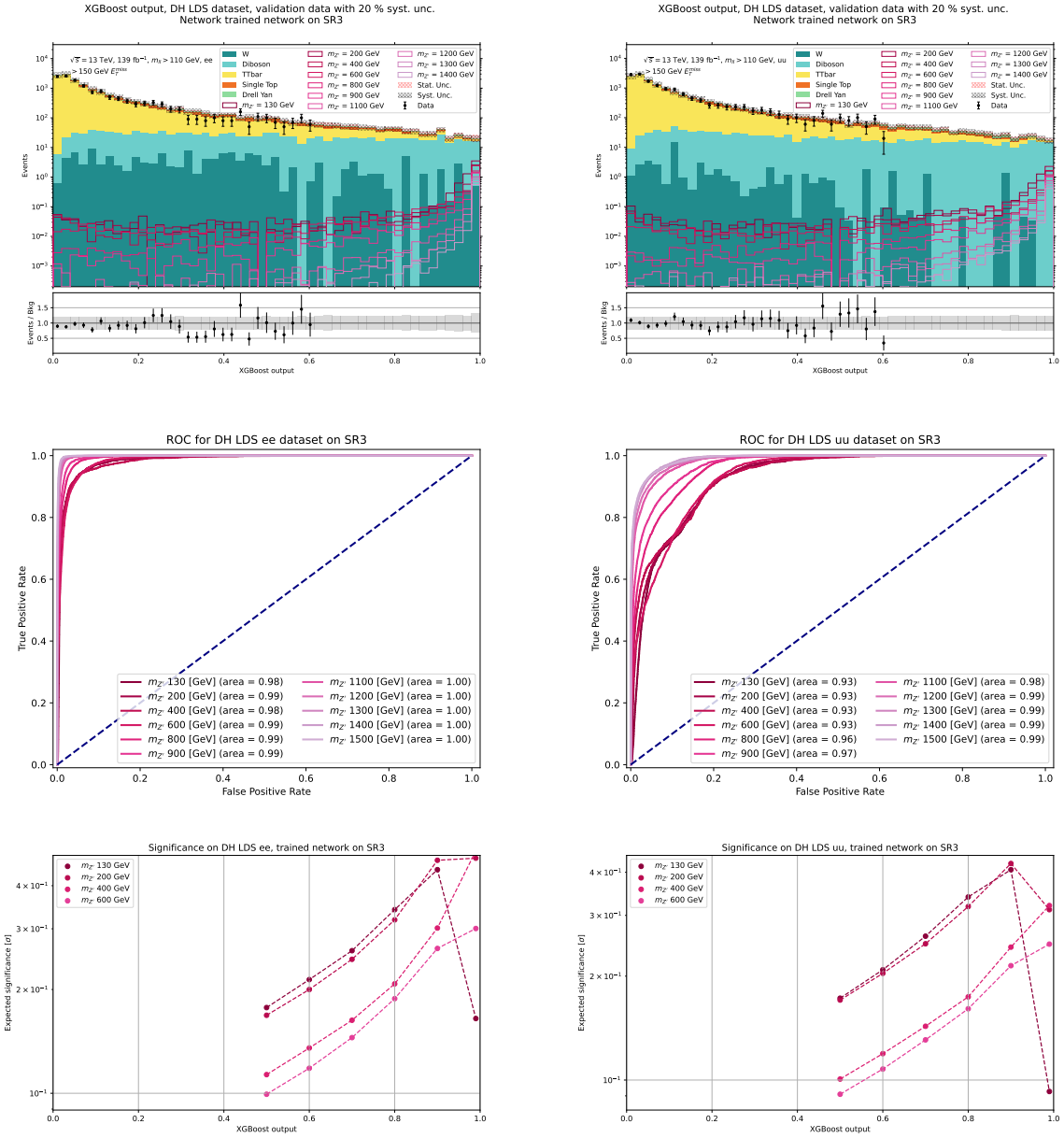


Figure G.3: XGBoost results for DH LDS model on ee and $\mu\mu$ channel in SR3

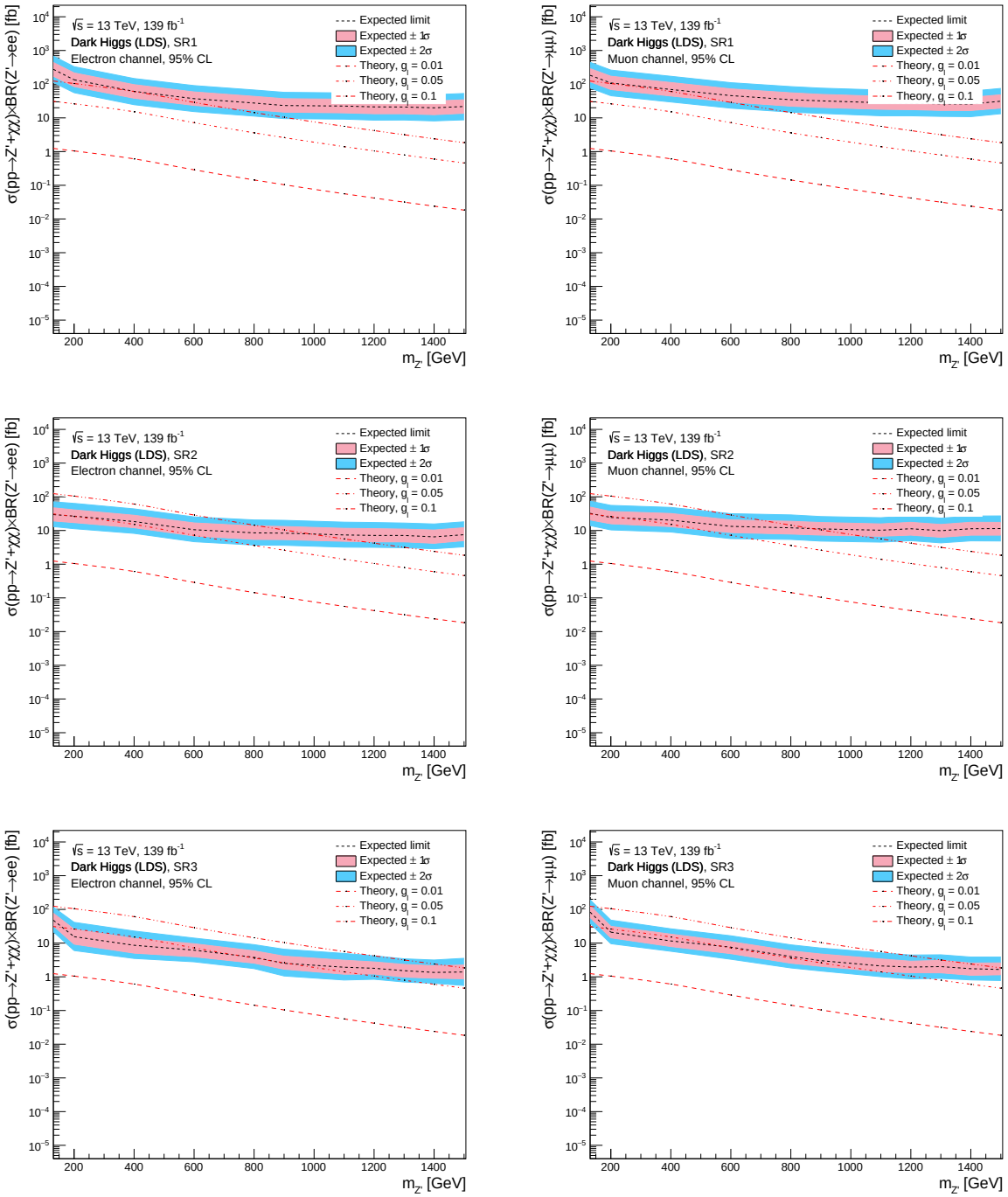


Figure G.4: Mass exclusion limits results for DH LDS model on ee and $\mu\mu$ channel in all SRs

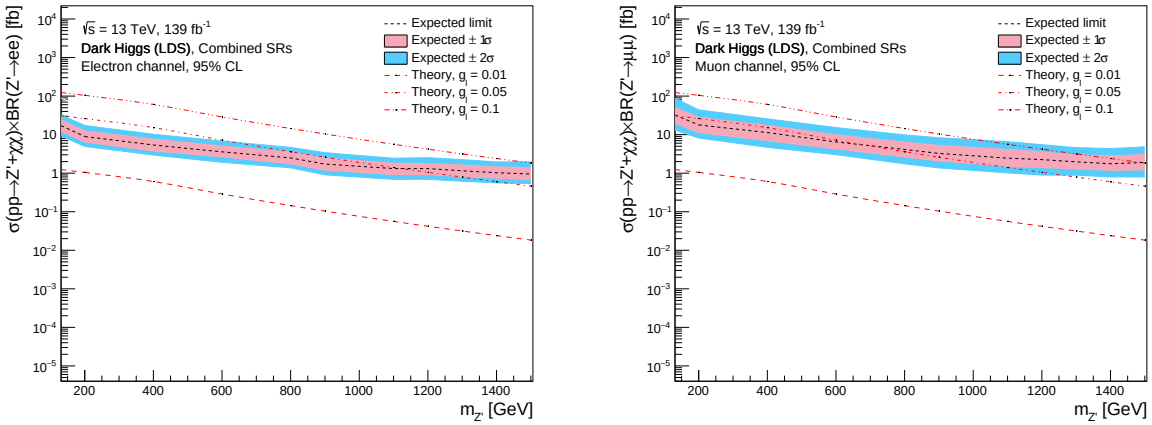


Figure G.5: Mass exclusion limits results for DH LDS model on ee and $\mu\mu$ channel in combined SRs

G.2 Light Vector Heavy Dark Sector

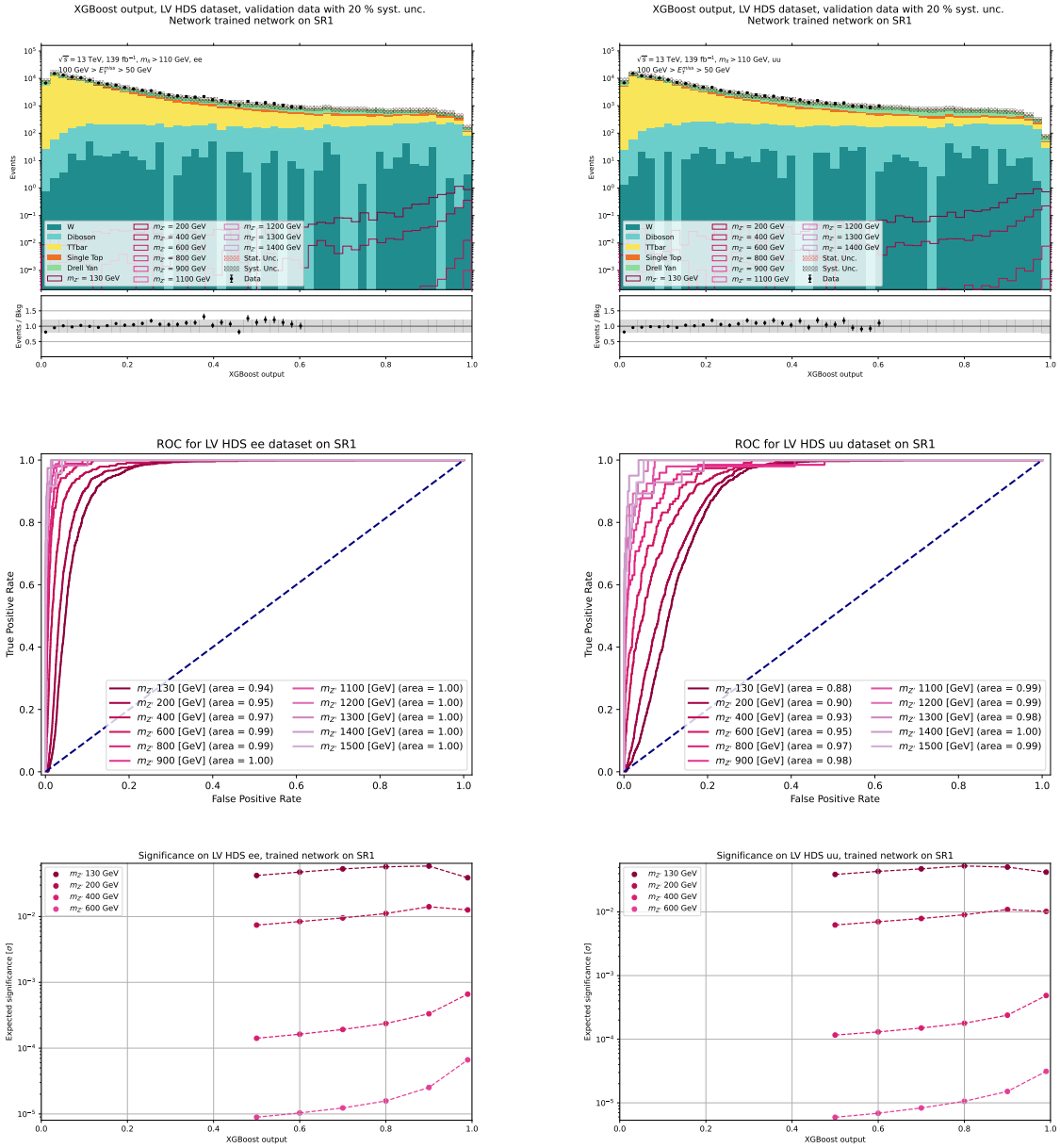


Figure G.6: XGBoost results for LV HDS model on ee and μμ channel in SR1

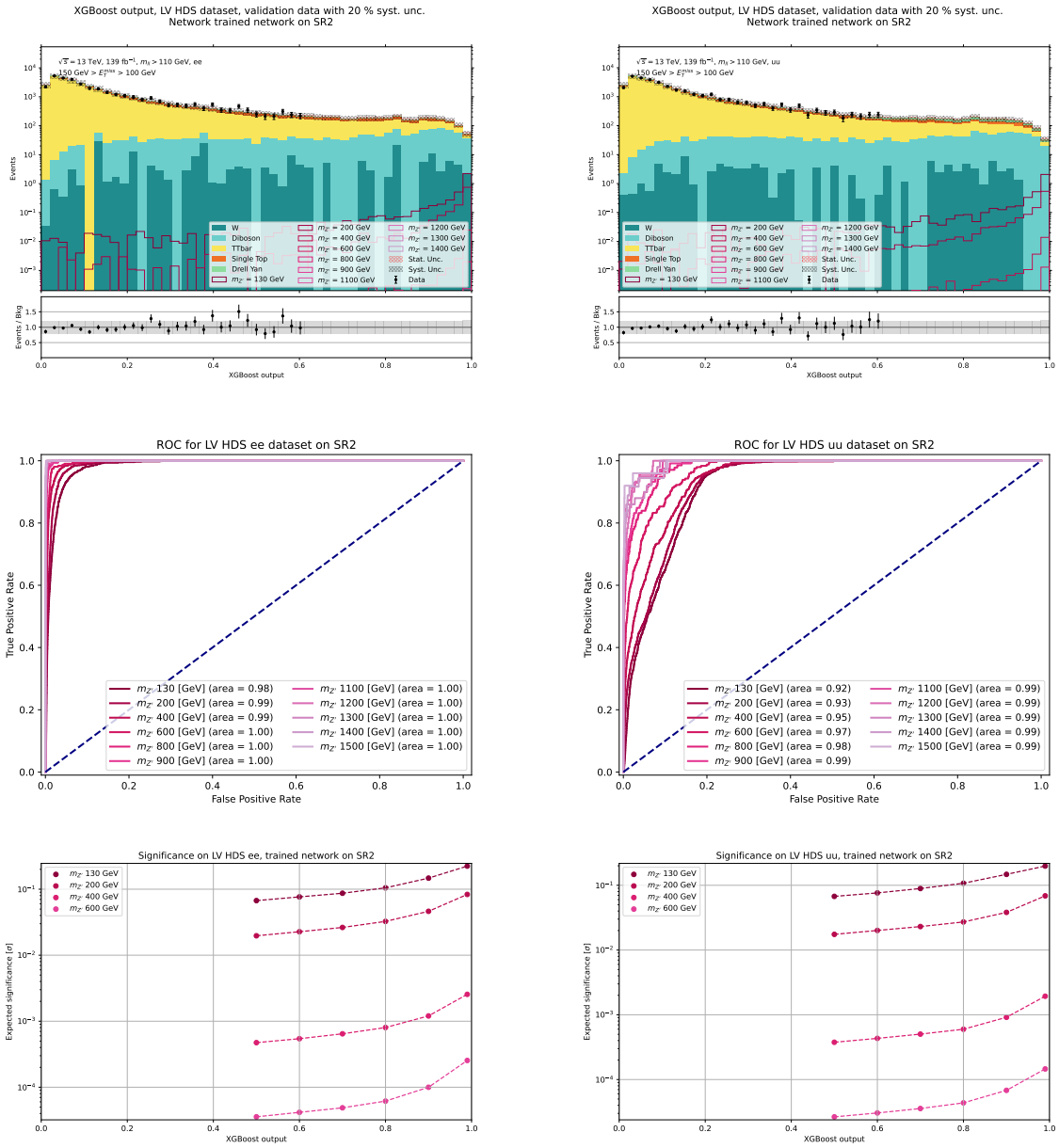


Figure G.7: XGBoost results for LV HDS model on ee and $\mu\mu$ channel in SR2

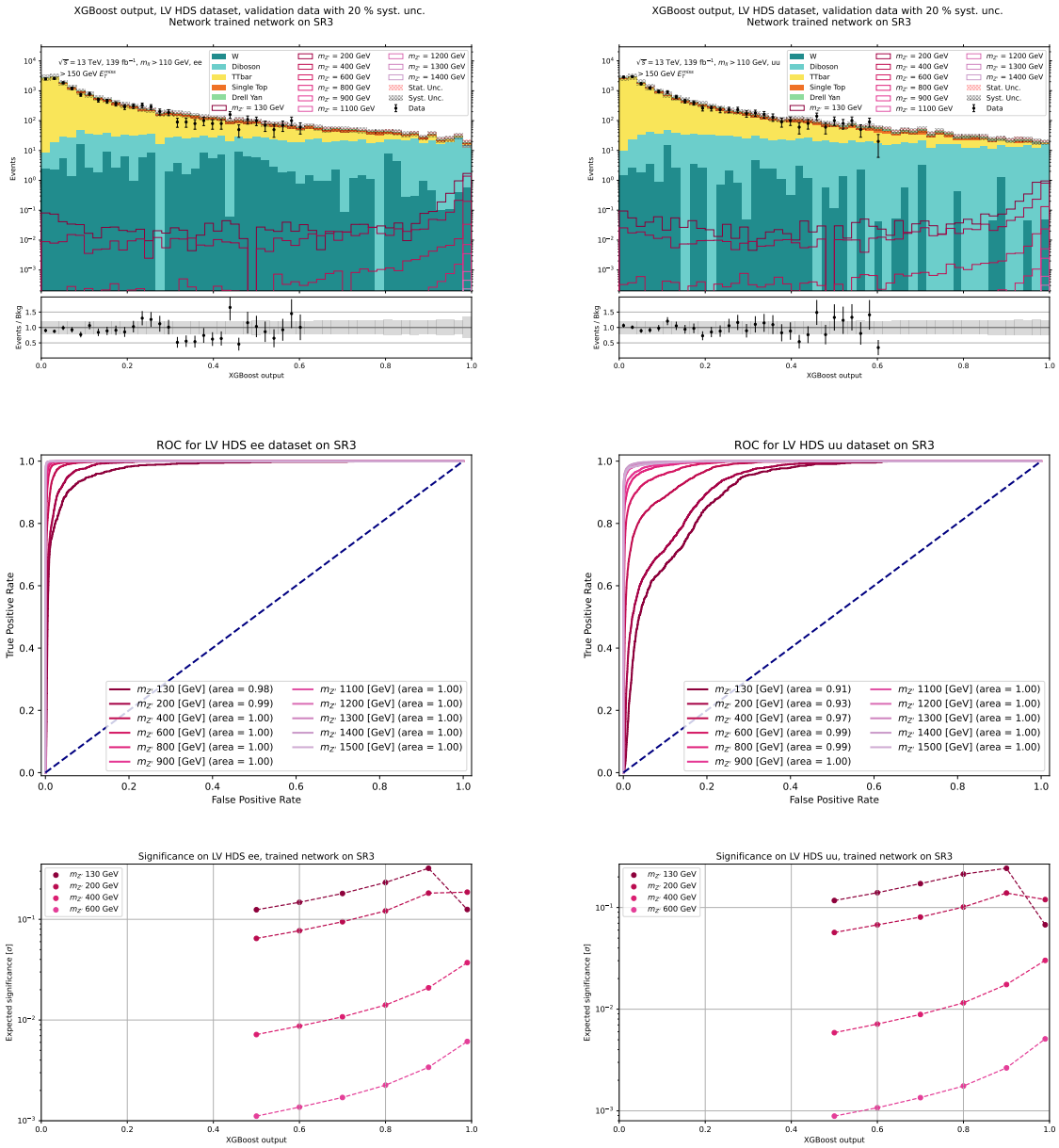


Figure G.8: XGBoost results for LV HDS model on ee and $\mu\mu$ channel in SR3

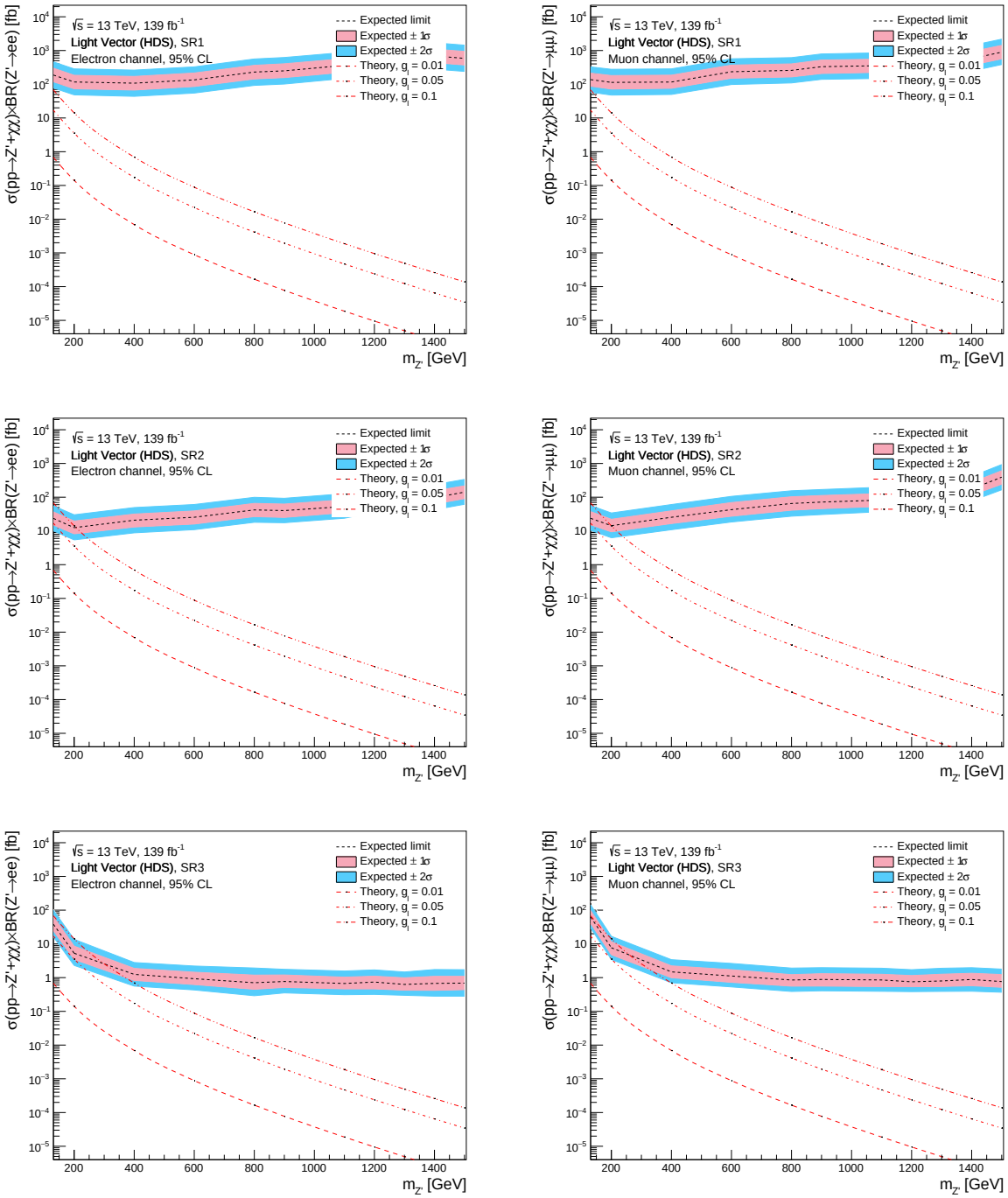


Figure G.9: Mass exclusion limits results for LV HDS model on ee and $\mu\mu$ channel in all SRs

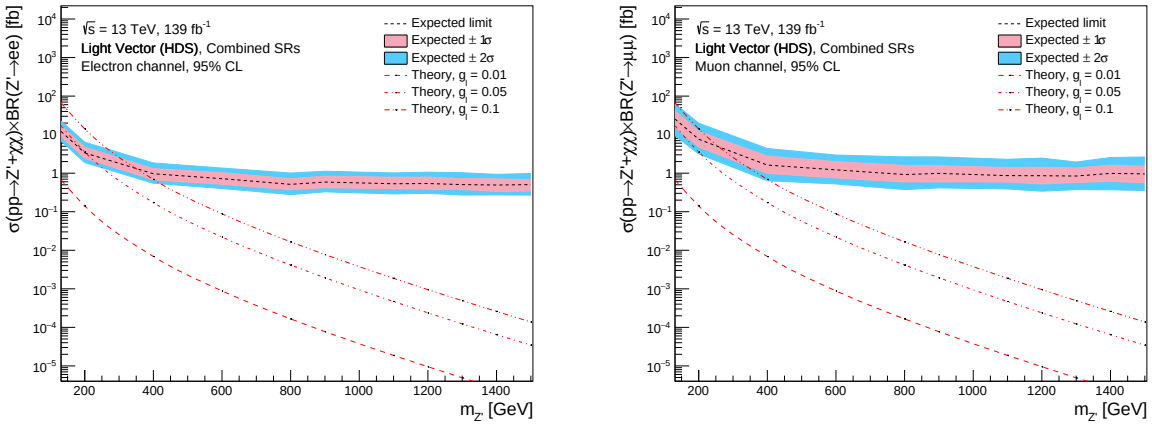


Figure G.10: Mass exclusion limits results for LV HDS model on ee and $\mu\mu$ channel in combined SRs

G.3 Light Vector Light Dark Sector

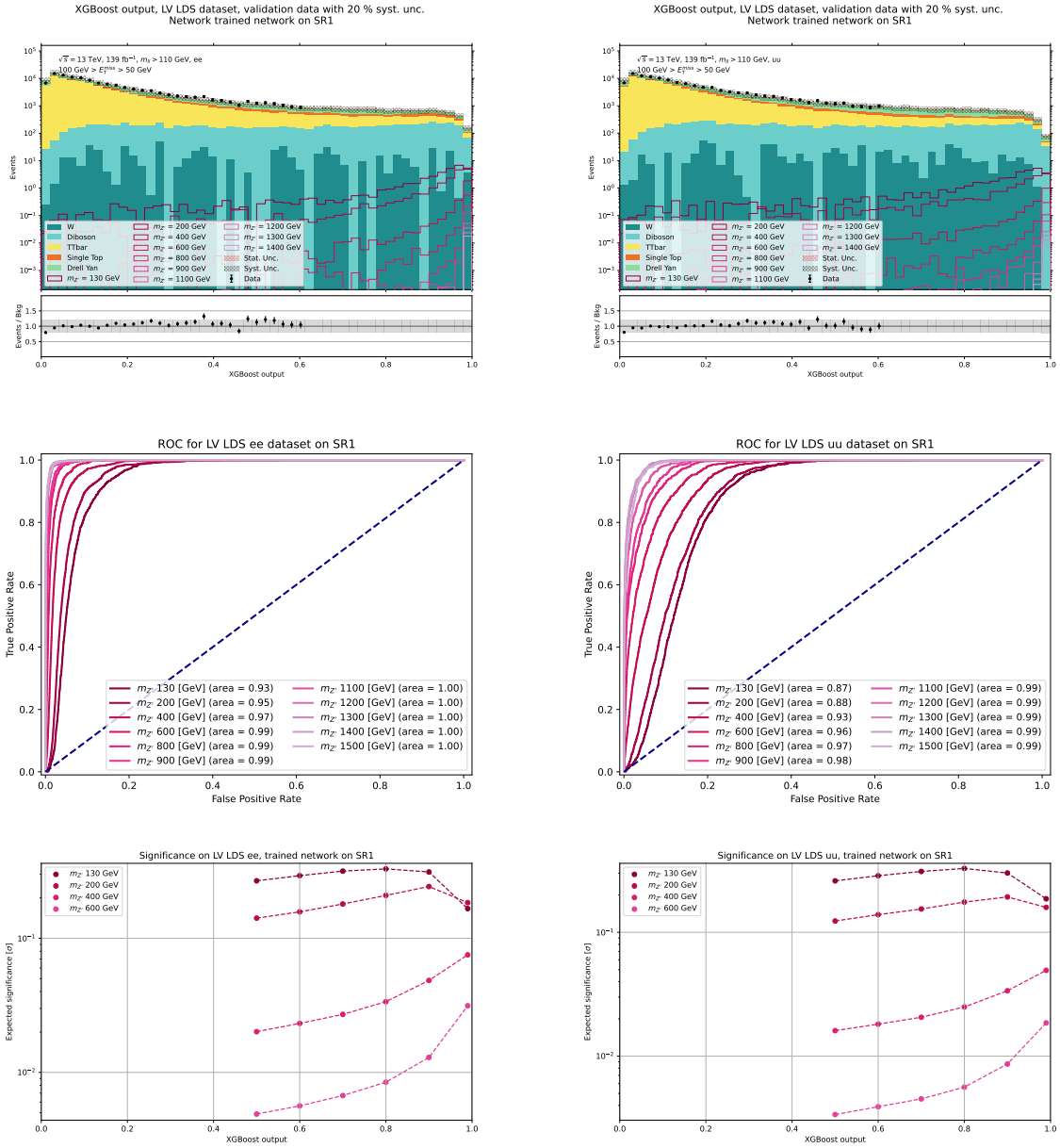


Figure G.11: XGBoost results for LV LDS model on ee and $\mu\mu$ channel in SR1

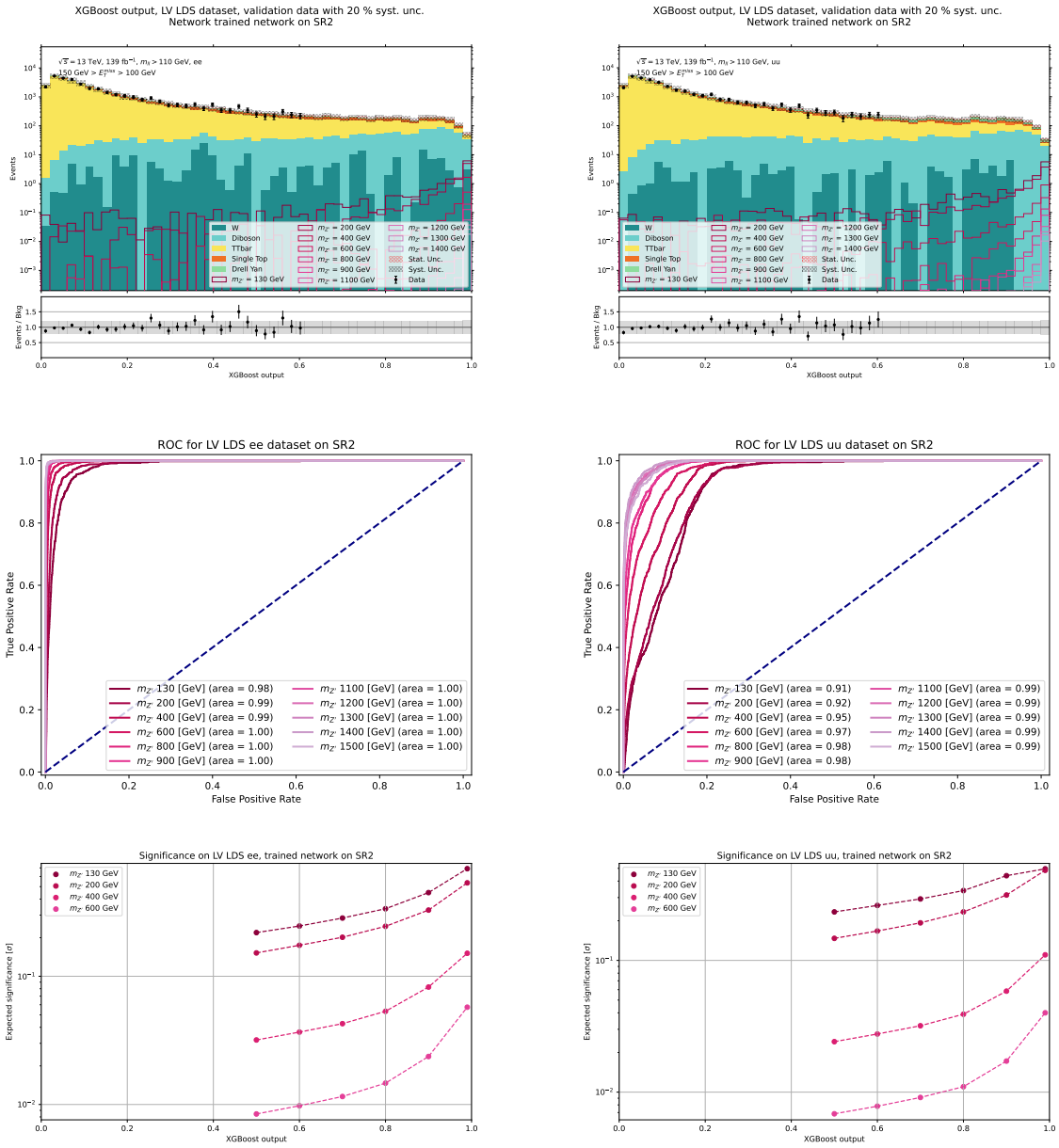


Figure G.12: XGBoost results for LV LDS model on ee and $\mu\mu$ channel in SR2

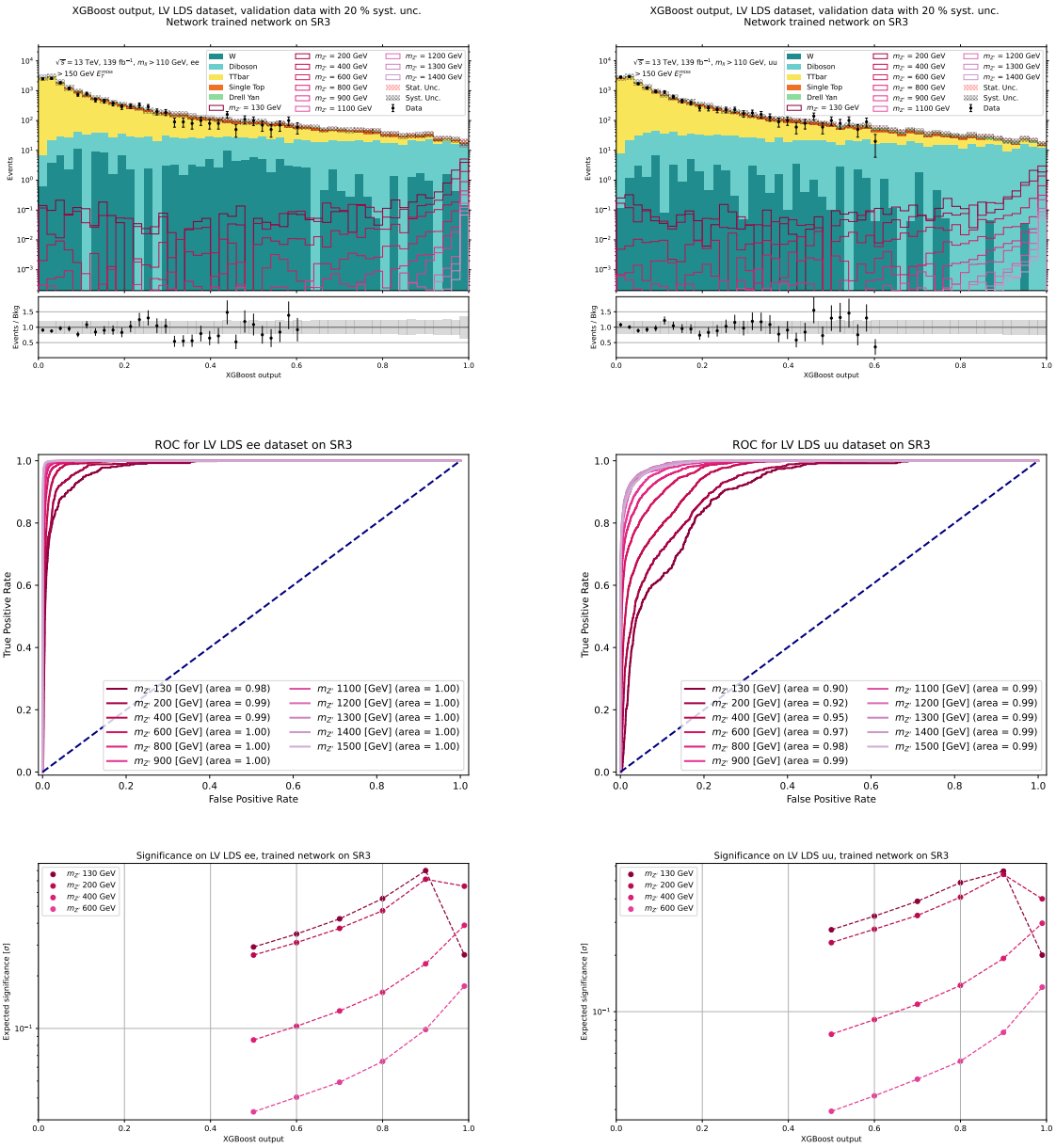


Figure G.13: XGBoost results for LV LDS model on ee and $\mu\mu$ channel in SR3

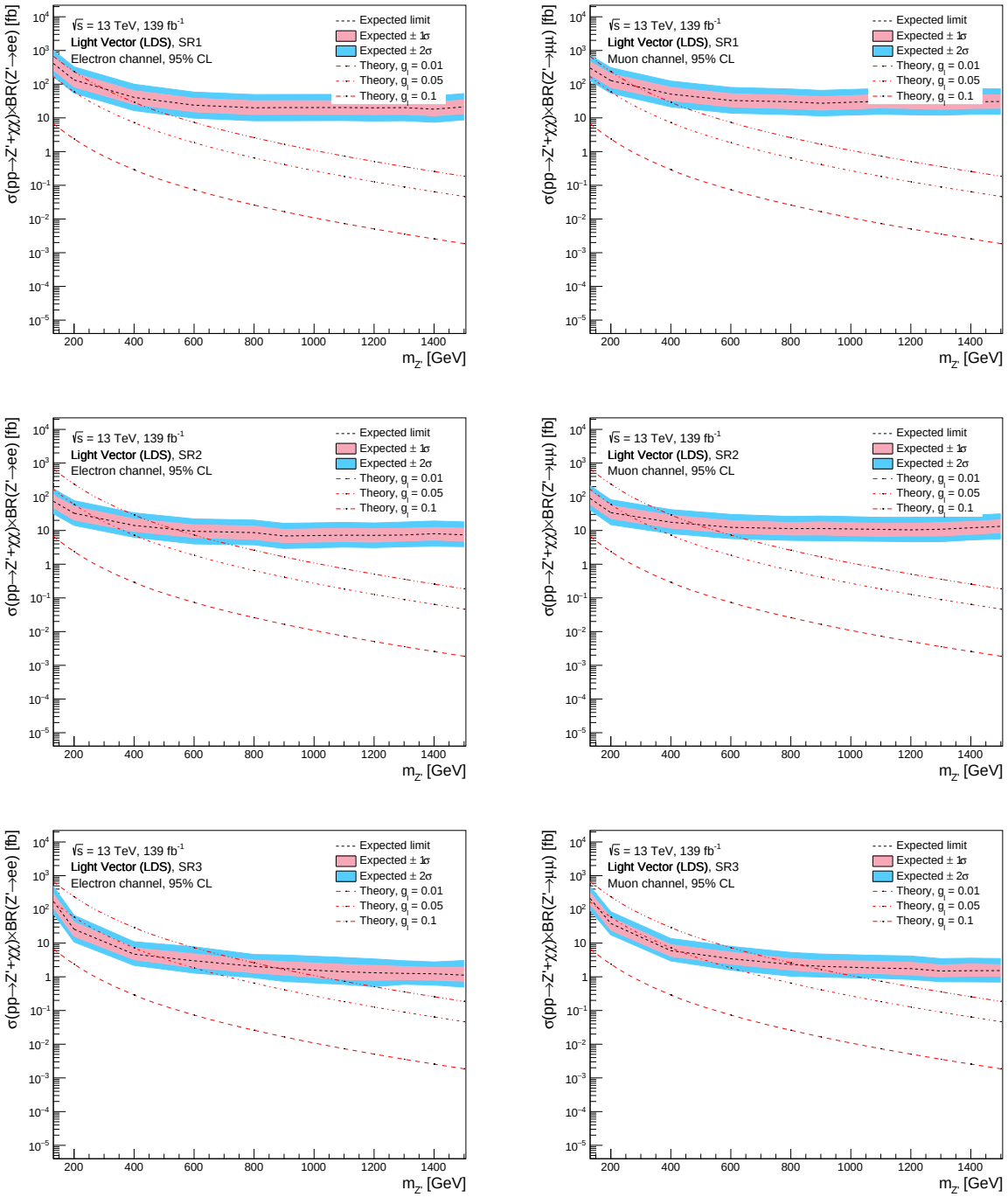


Figure G.14: Mass exclusion limits results for LV LDS model on ee and $\mu\mu$ channel in all SRs

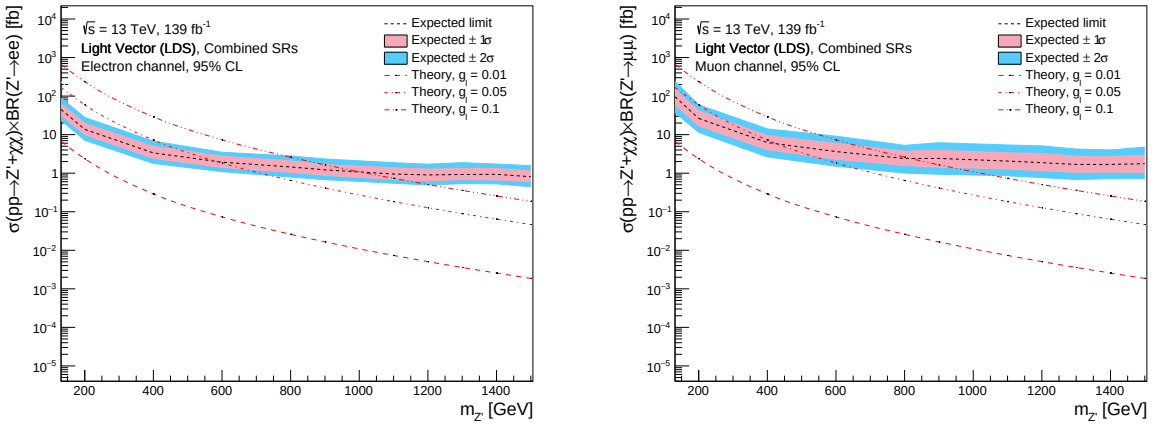


Figure G.15: Mass exclusion limits results for LV LDS model on ee and $\mu\mu$ channel in combined SRs

G.4 Effective Field Theory Heavy Dark Sector

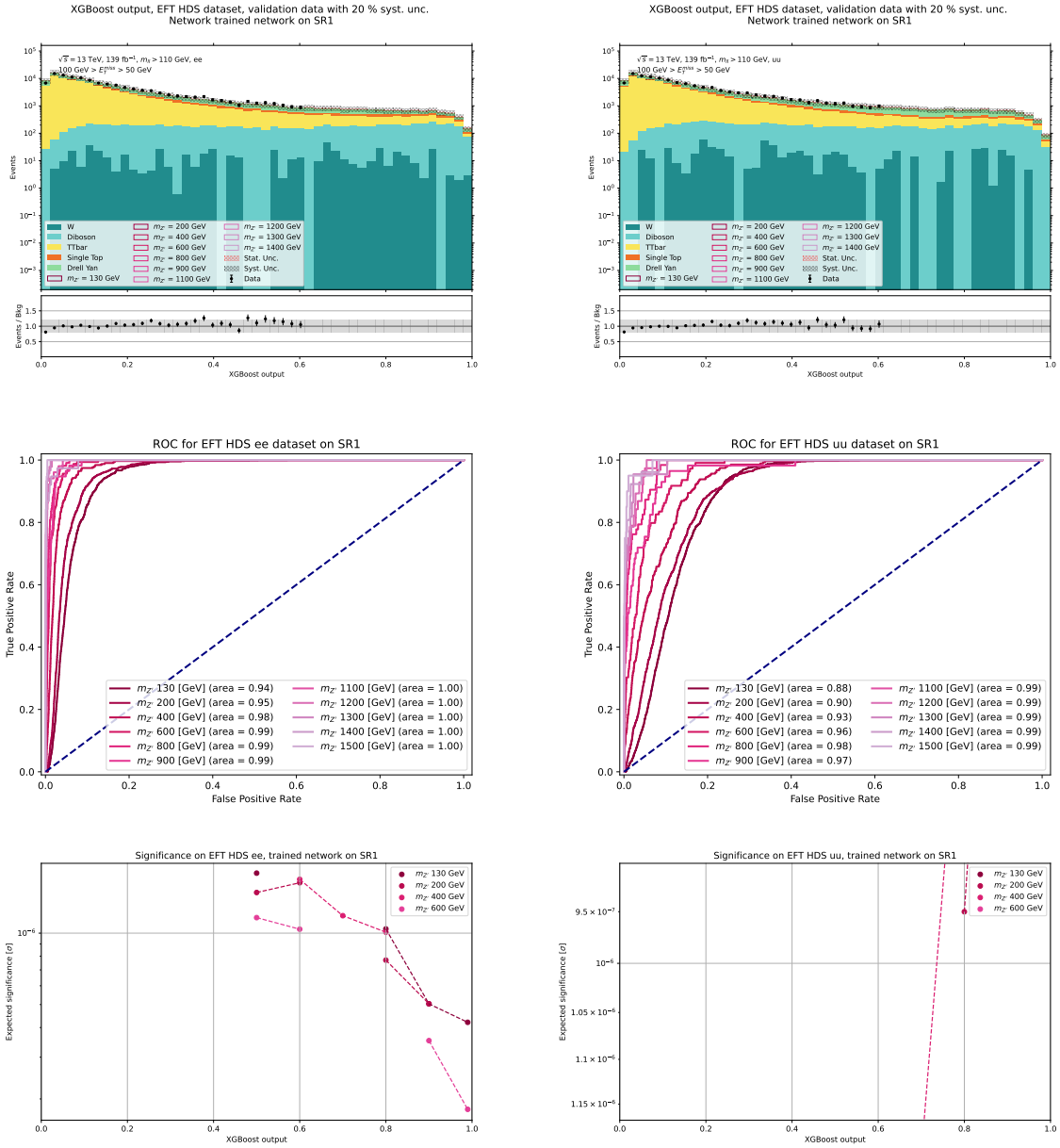


Figure G.16: XGBoost results for EFT HDS model on ee and $\mu\mu$ channel in SR1

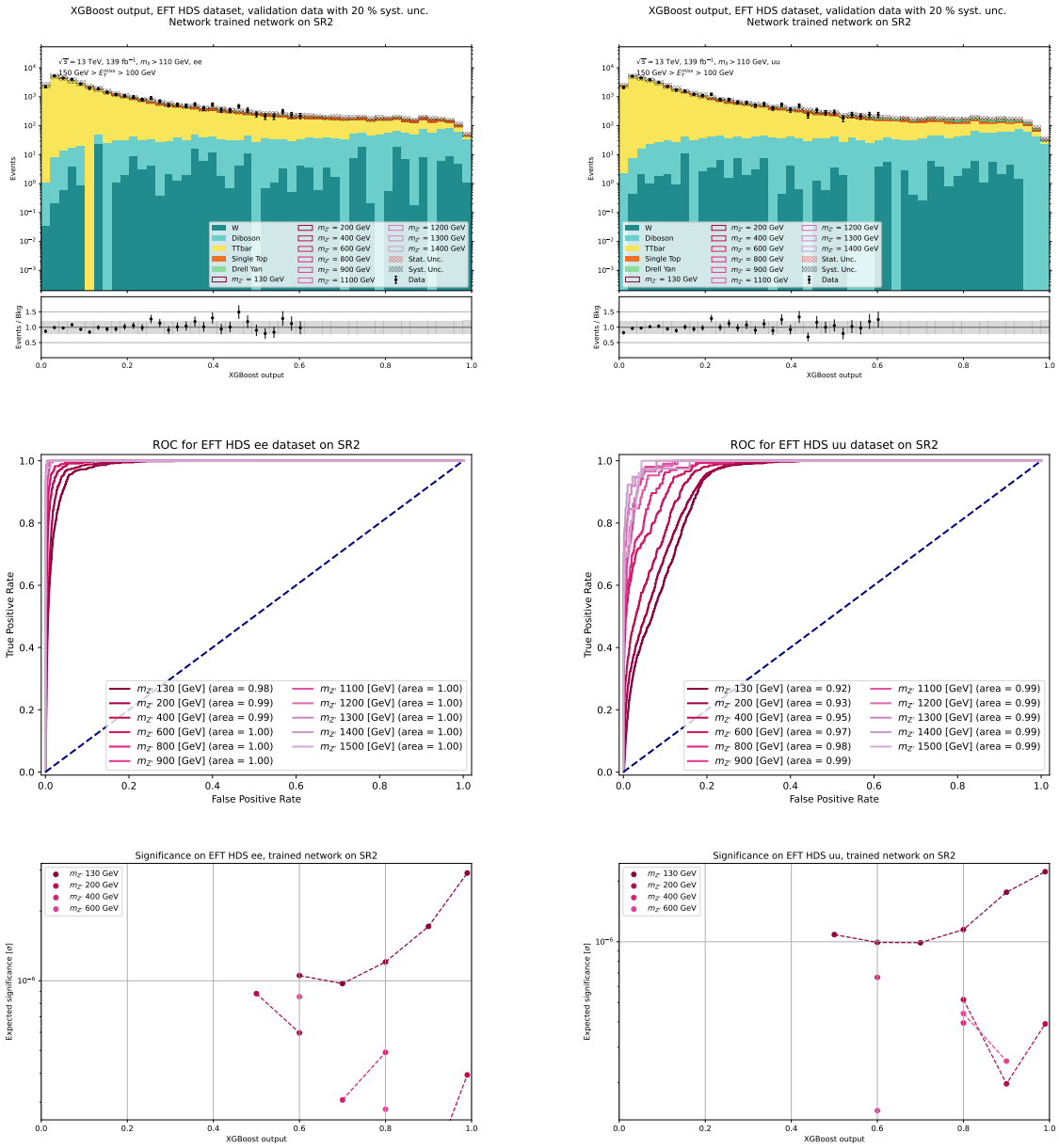


Figure G.17: XGBoost results for EFT HDS model on ee and μμ channel in SR2

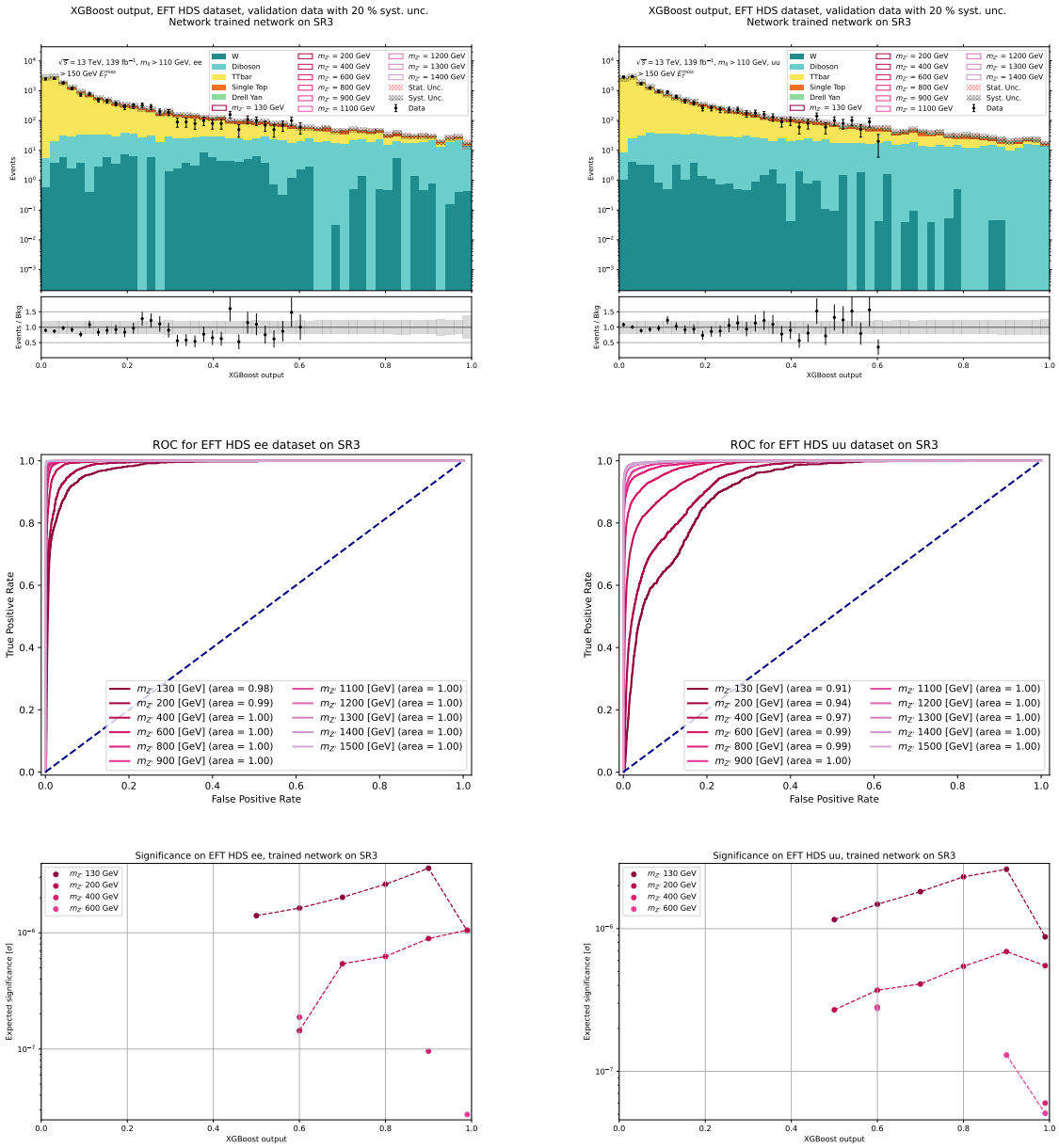


Figure G.18: XGBoost results for EFT HDS model on ee and μμ channel in SR3

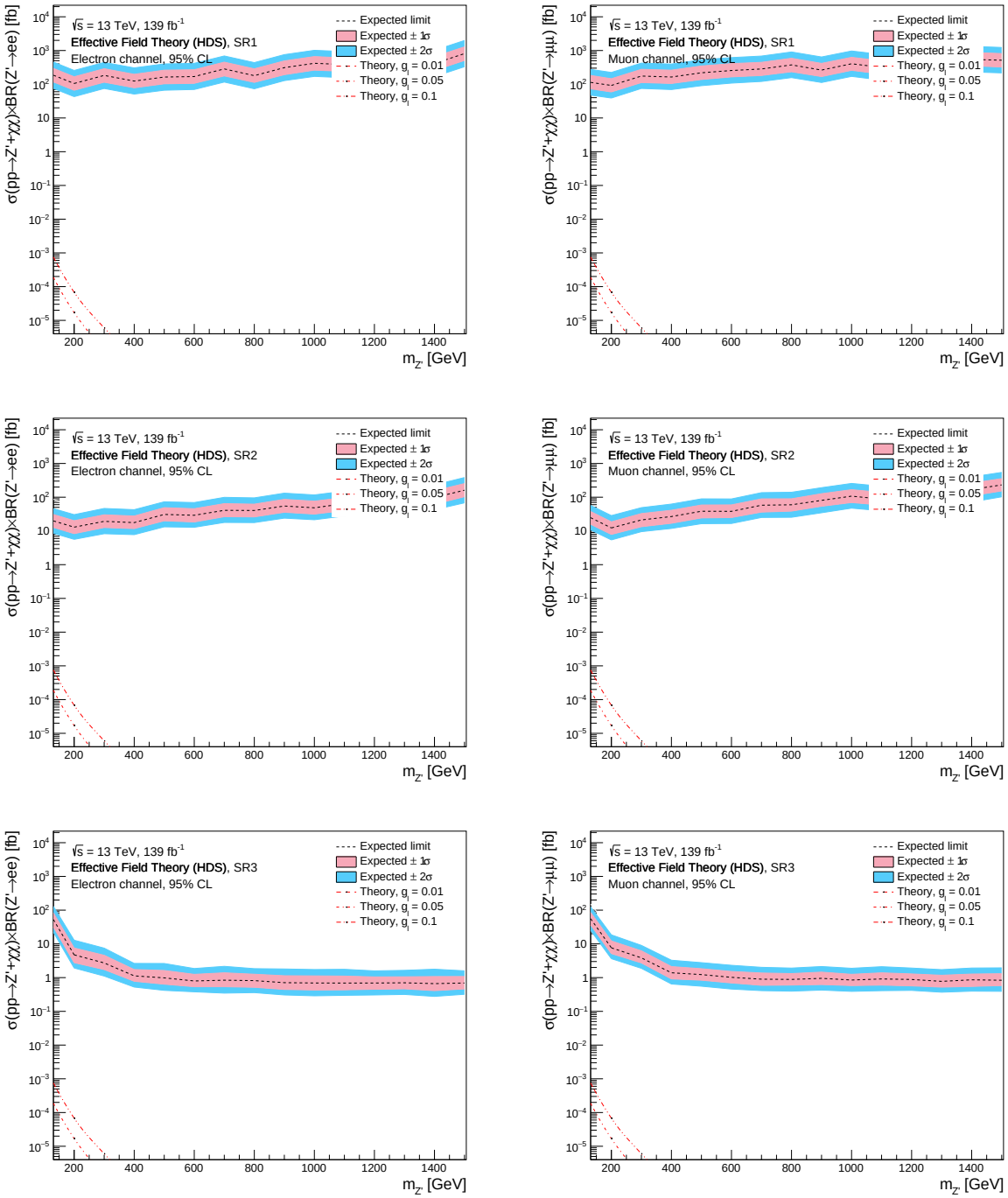


Figure G.19: Mass exclusion limits results for EFT HDS model on ee and $\mu\mu$ channel in all SRs

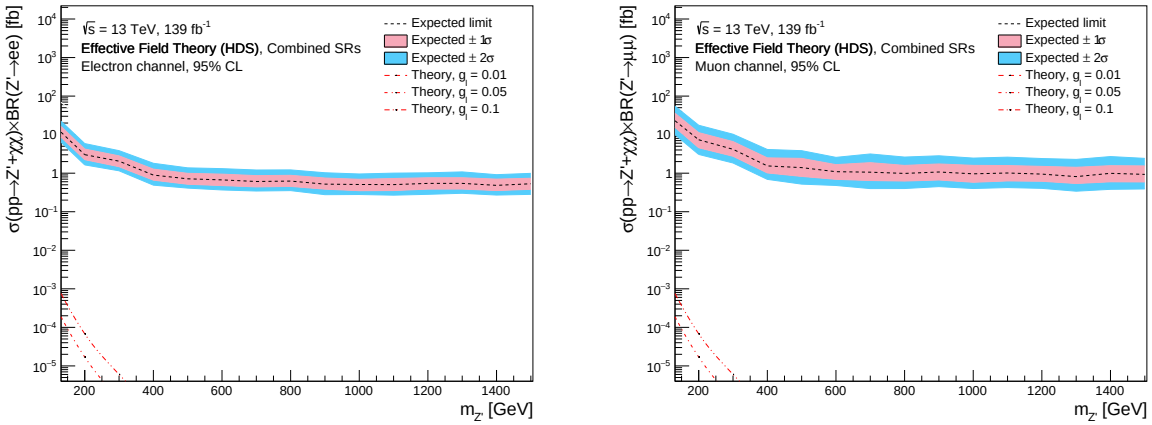


Figure G.20: Mass exclusion limits results for EFT HDS model on ee and $\mu\mu$ channel in combined SRs

G.5 Effective Field Theory Light Dark Sector

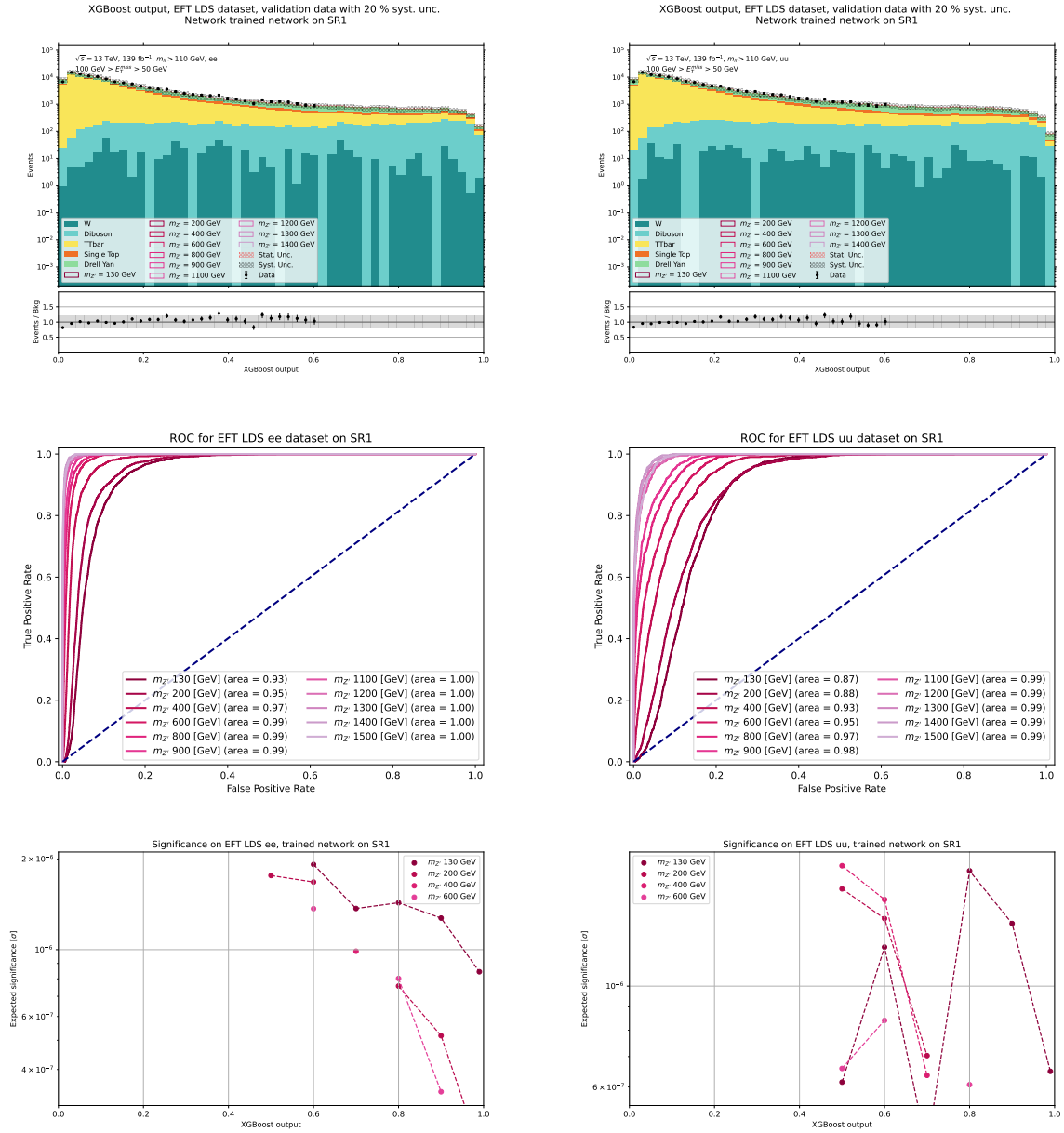


Figure G.21: XGBoost results for EFT LDS model on ee and $\mu\mu$ channel in SR1

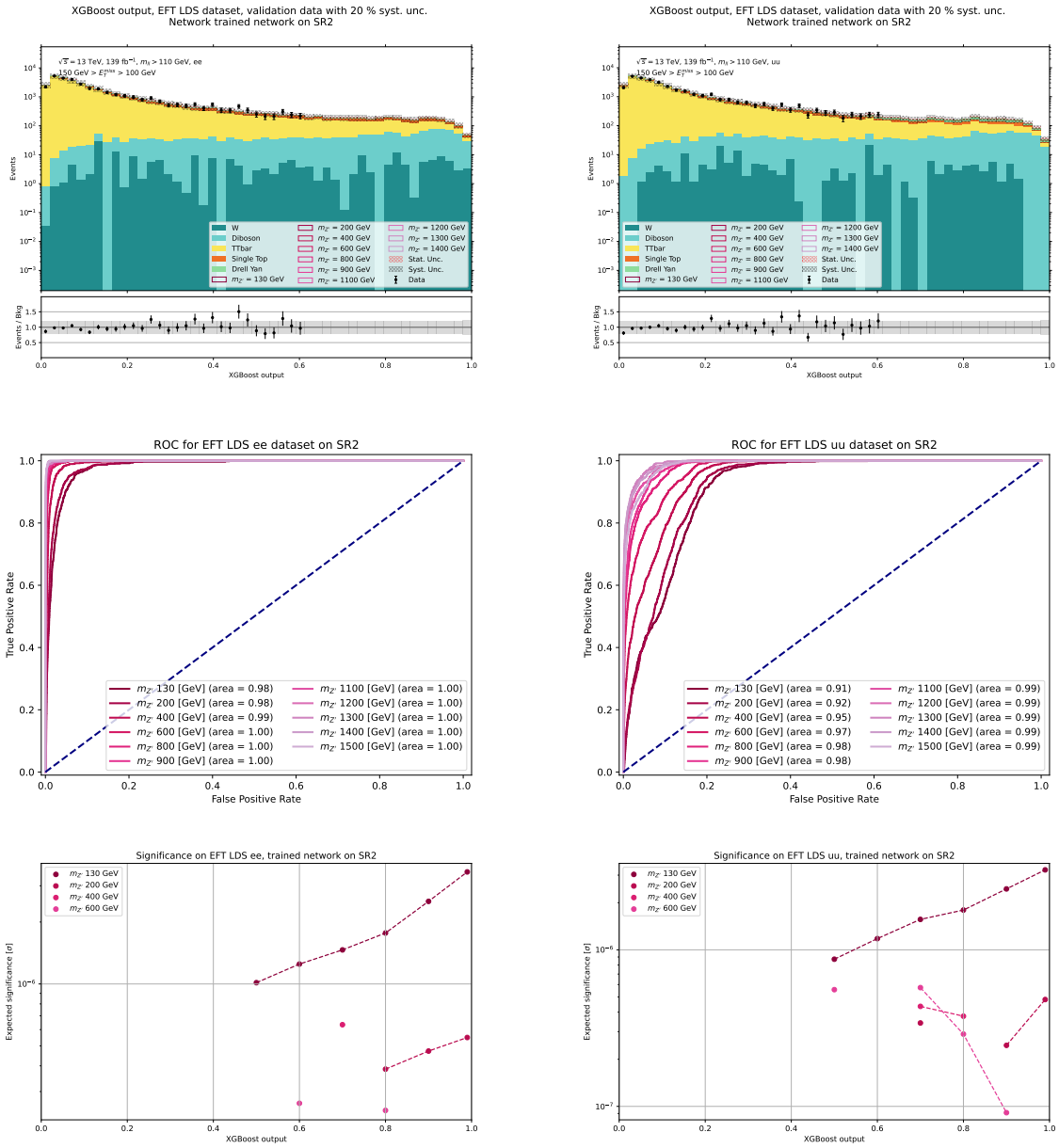


Figure G.22: XGBoost results for EFT LDS model on ee and μμ channel in SR2

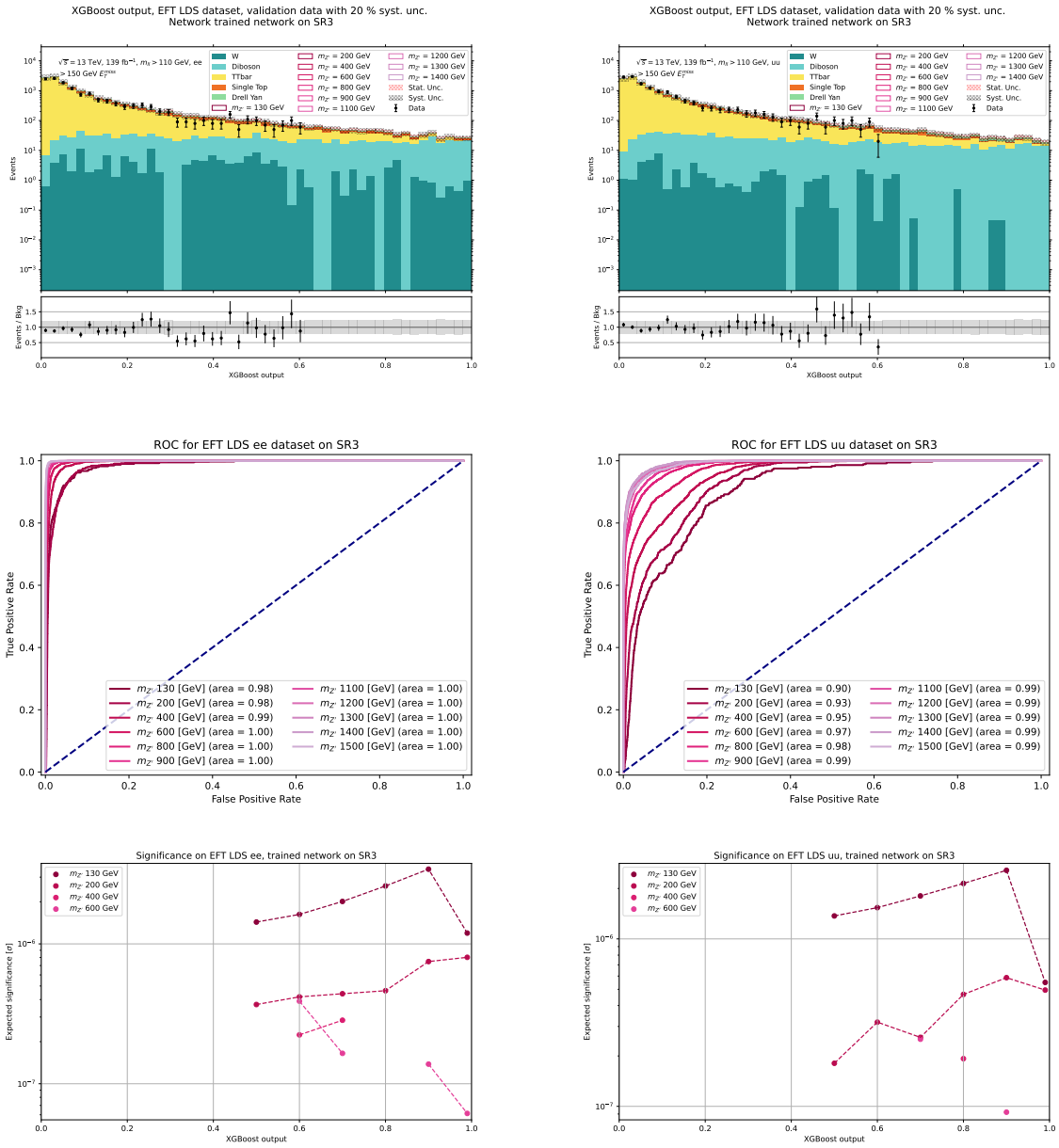


Figure G.23: XGBoost results for EFT LDS model on ee and μμ channel in SR3

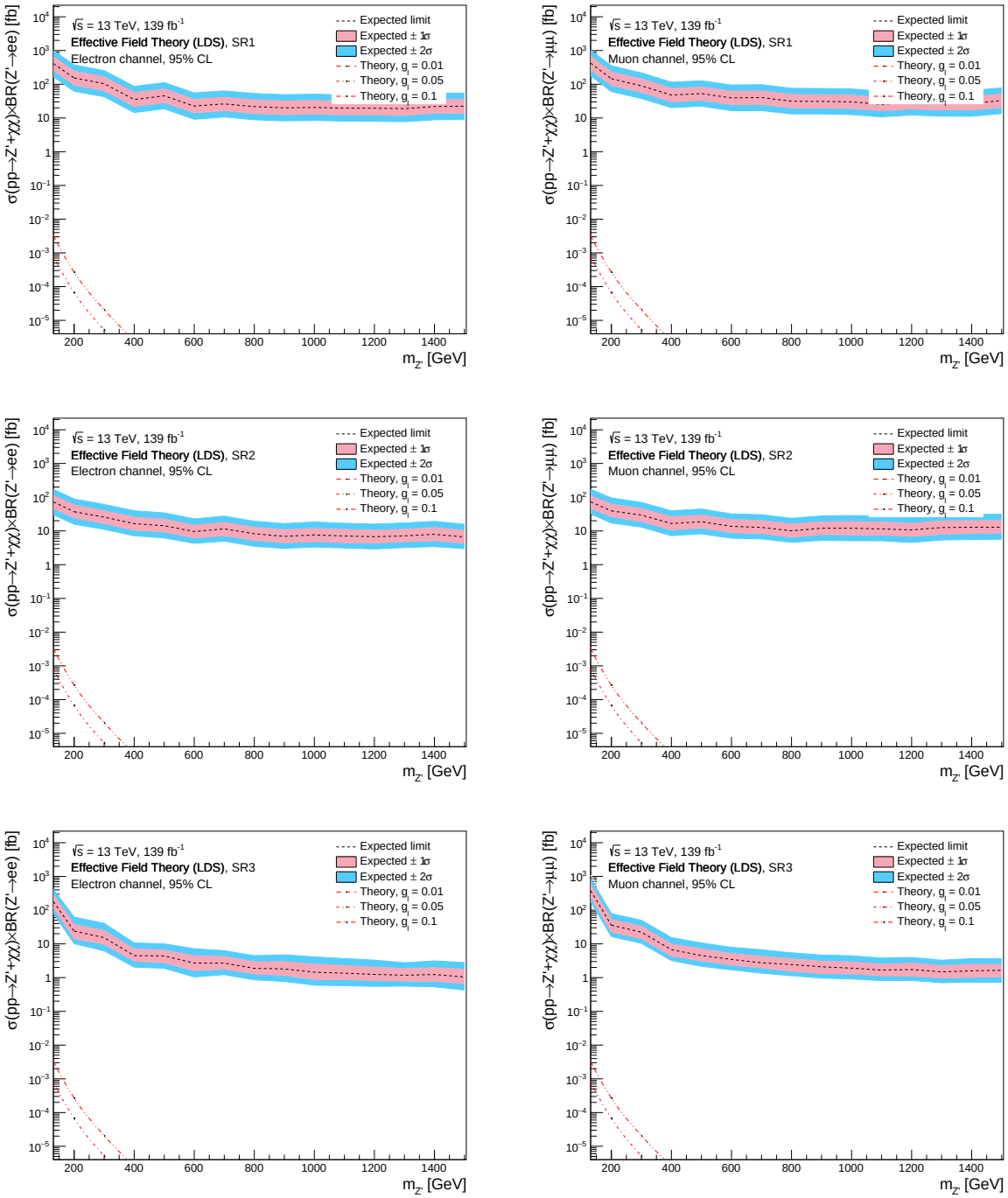


Figure G.24: Mass exclusion limits results for EFT LDS model on ee and $\mu\mu$ channel in all SRs

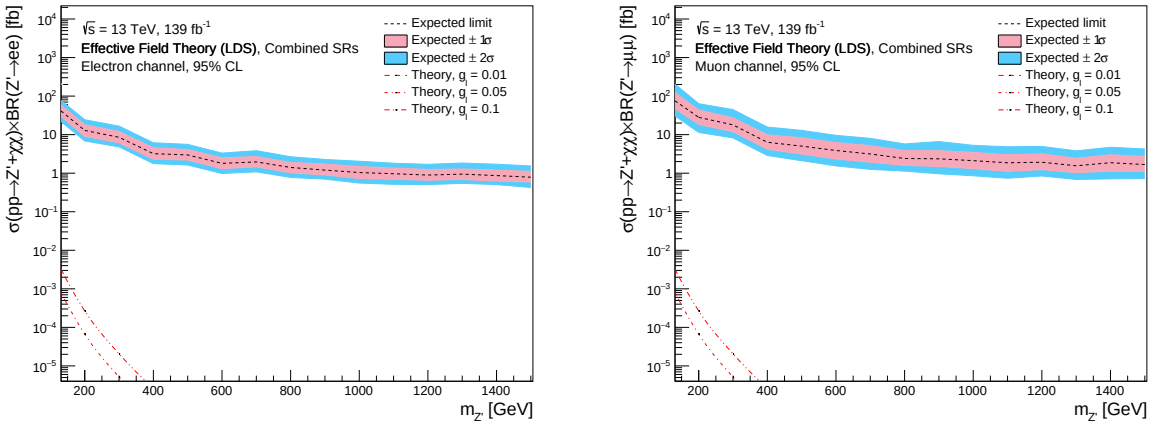


Figure G.25: Mass exclusion limits results for EFT LDS model on ee and $\mu\mu$ channel in combined SRs

Appendix H

Dataset IDs for MC samples

Table H.1: Drell Yan background MC samples

Dataset ID	Process
700320	Zee_maxHTpTV2_BFilter
700321	Zee_maxHTpTV2_CFilterBVeto
700322	Zee_maxHTpTV2_CVetoBVeto
700323	Zmumu_maxHTpTV2_BFilter
700324	Zmumu_maxHTpTV2_CFilterBVeto
700325	Zmumu_maxHTpTV2_CVetoBVeto
700326	Ztautau_LL_maxHTpTV2_BFilter
700327	Ztautau_LL_maxHTpTV2_CFilterBVeto
700328	Ztautau_LL_maxHTpTV2_CVetoBVeto
700452	Zee_mZ_120_ECMS_BFilter
700453	Zee_mZ_120_ECMS_CFilterBVeto
700454	Zee_mZ_120_ECMS_CVetoBVeto
700455	Zmumu_mZ_120_ECMS_BFilter
700456	Zmumu_mZ_120_ECMS_CFilterBVeto
700457	Zmumu_mZ_120_ECMS_CVetoBVeto
700458	Ztautau_mZ_120_ECMS_BFilter
700459	Ztautau_mZ_120_ECMS_CFilterBVeto
700460	Ztautau_mZ_120_ECMS_CVetoBVeto

Table H.2: Single top and TTbar background MC samples

Dataset ID	Process
410472	$t\bar{t}$ dilepton filtered
410644	Single top s-channel (top)
410645	Single top s-channel (anti-top)
410648	$W + t$ associated production dilepton filtered (top)
410649	$W + t$ associated production dilepton filtered (anti-top)
410658	Single top t-channel (top)
410659	Single top t-channel (anti-top)

Table H.3: Diboson background MC samples

Dataset ID	Process
363356	$Z \to qq + Z \to ll$
363358	$W \to qq + Z \to ll$
363359	$W^+ \to qq + W^- \to l\nu$
363360	$W^+ \to l\nu + W^- \to qq$
363489	$W \to l\nu + Z \to qq$
364250	$llll$
364253	$lll\nu$
364254	$ll\nu\nu$
364255	$l\nu\nu\nu$

Table H.4: W background MC samples

Dataset ID	Process
364156	$W \to \mu\nu$ maxHTpTV0_70_CVetoBVeto
364157	$W \to \mu\nu$ maxHTpTV0_70_CFilterBVeto
364158	$W \to \mu\nu$ maxHTpTV0_70_BFilter
364159	$W \to \mu\nu$ maxHTpTV70_140_CVetoBVeto
364160	$W \to \mu\nu$ maxHTpTV70_140_CFilterBVeto
364161	$W \to \mu\nu$ maxHTpTV70_140_BFilter
364162	$W \to \mu\nu$ maxHTpTV140_280_CVetoBVeto
364163	$W \to \mu\nu$ maxHTpTV140_280_CFilterBVeto
364164	$W \to \mu\nu$ maxHTpTV140_280_BFilter
364165	$W \to \mu\nu$ maxHTpTV280_500_CVetoBVeto
364166	$W \to \mu\nu$ maxHTpTV280_500_CFilterBVeto
364167	$W \to \mu\nu$ maxHTpTV280_500_BFilter
364168	$W \to \mu\nu$ maxHTpTV500_1000
364169	$W \to \mu\nu$ maxHTpTV1000_E_CMS
364170	$W \to e\nu$ maxHTpTV0_70_CVetoBVeto
364171	$W \to e\nu$ maxHTpTV0_70_CFilterBVeto
364172	$W \to e\nu$ maxHTpTV0_70_BFilter
364173	$W \to e\nu$ maxHTpTV70_140_CVetoBVeto
364174	$W \to e\nu$ maxHTpTV70_140_CFilterBVeto
364175	$W \to e\nu$ maxHTpTV70_140_BFilter
364176	$W \to e\nu$ maxHTpTV140_280_CVetoBVeto
364177	$W \to e\nu$ maxHTpTV140_280_CFilterBVeto
364178	$W \to e\nu$ maxHTpTV140_280_BFilter
364179	$W \to e\nu$ maxHTpTV280_500_CVetoBVeto
364180	$W \to e\nu$ maxHTpTV280_500_CFilterBVeto
364181	$W \to e\nu$ maxHTpTV280_500_BFilter
364182	$W \to e\nu$ maxHTpTV500_1000
364183	$W \to e\nu$ maxHTpTV1000_E_CMS
364184	$W \to \tau\nu$ maxHTpTV0_70_CVetoBVeto
364185	$W \to \tau\nu$ maxHTpTV0_70_CFilterBVeto
364186	$W \to \tau\nu$ maxHTpTV0_70_BFilter
364187	$W \to \tau\nu$ maxHTpTV70_140_CVetoBVeto
364188	$W \to \tau\nu$ maxHTpTV70_140_CFilterBVeto
364189	$W \to \tau\nu$ maxHTpTV70_140_BFilter
364190	$W \to \tau\nu$ maxHTpTV140_280_CVetoBVeto
364191	$W \to \tau\nu$ maxHTpTV140_280_CFilterBVeto
364192	$W \to \tau\nu$ maxHTpTV140_280_BFilter
364193	$W \to \tau\nu$ maxHTpTV280_500_CVetoBVeto
364194	$W \to \tau\nu$ maxHTpTV280_500_CFilterBVeto
364195	$W \to \tau\nu$ maxHTpTV280_500_BFilter
364196	$W \to \tau\nu$ maxHTpTV500_1000
364197	$W \to \tau\nu$ maxHTpTV1000_E_CMS

Appendix I

Limit calculation tables

To calculate the signal efficiency ε_{sig}, we used the following formula

$$\varepsilon_{sig} = \frac{N_{sig}}{\sigma B \times \text{lumi}_{RunII}}$$

where N_{sig} are the number of signal events after the cut, and where we calculated

$$\sigma B \times \text{lumi}_{RunII} = \sum_{period} \sigma B_{period} \times \text{lumi}_{period}$$

extracting the cross-section of the DM sample from the DSID on each MC period and converting it to fb units, and using

$$\text{lumi}_{period} = \begin{cases} 36.4 \text{ fb}^{-1}, & \text{for period} = \text{a} \\ 44.3 \text{ fb}^{-1}, & \text{for period} = \text{d} \\ 58.5 \text{ fb}^{-1}, & \text{for period} = \text{e} \end{cases}$$

Table I.1: Inputs for the $Z'h_D \to l^+l^-\chi\chi$ LDS σB calculations. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E^{miss}_{T,min} = 50GeV$ and the invariant mass threshold is $m^{min}_{ll} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	1.24	ee	0.25 ± 0.05	8.70 ± 1.75	272.0 ± 56.0
		$\mu\mu$	0.21 ± 0.04	7.33 ± 1.48	299.4 ± 60.7
200	1.05	ee	0.29 ± 0.06	8.34 ± 1.68	294.9 ± 60.3
		$\mu\mu$	0.23 ± 0.05	6.68 ± 1.34	304.4 ± 61.7
400	6.08×10^{-1}	ee	0.38 ± 0.08	6.35 ± 1.28	264.9 ± 55.0
		$\mu\mu$	0.25 ± 0.05	$4.30 \pm 8.66 \times 10^{-1}$	288.9 ± 58.7
600	2.88×10^{-1}	ee	0.70 ± 0.14	5.64 ± 1.13	275.2 ± 56.6
		$\mu\mu$	0.50 ± 0.10	$3.98 \pm 7.98 \times 10^{-1}$	298.3 ± 60.4
800	1.44×10^{-1}	ee	1.01 ± 0.20	$4.05 \pm 8.11 \times 10^{-1}$	295.9 ± 60.5
		$\mu\mu$	0.70 ± 0.14	$2.81 \pm 5.63 \times 10^{-1}$	307.5 ± 62.4
900	1.04×10^{-1}	ee	1.14 ± 0.23	$3.30 \pm 6.61 \times 10^{-1}$	271.3 ± 55.8
		$\mu\mu$	0.80 ± 0.16	$2.30 \pm 4.60 \times 10^{-1}$	293.2 ± 59.4
1100	5.61×10^{-2}	ee	1.30 ± 0.26	$2.03 \pm 4.07 \times 10^{-1}$	270.8 ± 55.7
		$\mu\mu$	0.90 ± 0.18	$1.40 \pm 2.81 \times 10^{-1}$	290.0 ± 58.8
1200	4.19×10^{-2}	ee	1.35 ± 0.27	$1.58 \pm 3.16 \times 10^{-1}$	273.9 ± 56.2
		$\mu\mu$	0.95 ± 0.19	$1.10 \pm 2.21 \times 10^{-1}$	289.4 ± 58.8
1300	3.16×10^{-2}	ee	1.41 ± 0.28	$1.24 \pm 2.48 \times 10^{-1}$	292.8 ± 60.0
		$\mu\mu$	0.97 ± 0.19	$8.49 \times 10^{-1} \pm 1.70 \times 10^{-1}$	288.6 ± 58.5
1400	2.40×10^{-2}	ee	1.43 ± 0.29	$9.51 \times 10^{-1} \pm 1.91 \times 10^{-1}$	298.8 ± 61.2
		$\mu\mu$	0.97 ± 0.19	$6.47 \times 10^{-1} \pm 1.30 \times 10^{-1}$	311.7 ± 63.2
1500	1.84×10^{-2}	ee	1.44 ± 0.29	$7.36 \times 10^{-1} \pm 1.47 \times 10^{-1}$	287.9 ± 59.1
		$\mu\mu$	0.96 ± 0.19	$4.90 \times 10^{-1} \pm 9.82 \times 10^{-2}$	298.7 ± 60.6

Table I.2: Inputs for the $Z'\chi_2 \to l^+l^-\chi_1\chi_1$ HDS σB calculations. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E^{miss}_{T,min} = 50GeV$ and the invariant mass threshold is $m^{min}_{ll} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	6.93×10^{-1}	ee	0.30 ± 0.06	5.72 ± 1.16	93.6 ± 20.6
		$\mu\mu$	0.24 ± 0.05	$4.66 \pm 9.44 \times 10^{-1}$	104.4 ± 21.9
200	1.41×10^{-1}	ee	0.69 ± 0.14	$2.71 \pm 5.45 \times 10^{-1}$	102.5 ± 22.7
		$\mu\mu$	0.51 ± 0.10	$2.02 \pm 4.08 \times 10^{-1}$	102.4 ± 21.4
400	6.90×10^{-3}	ee	1.30 ± 0.26	$2.48 \times 10^{-1} \pm 4.99 \times 10^{-2}$	107.2 ± 22.7
		$\mu\mu$	0.91 ± 0.18	$1.75 \times 10^{-1} \pm 3.52 \times 10^{-2}$	110.2 ± 23.1
600	8.81×10^{-4}	ee	1.56 ± 0.31	$3.83 \times 10^{-2} \pm 7.67 \times 10^{-3}$	98.2 ± 21.1
		$\mu\mu$	1.06 ± 0.21	$2.60 \times 10^{-2} \pm 5.21 \times 10^{-3}$	91.4 ± 19.7
800	1.66×10^{-4}	ee	1.66 ± 0.33	$7.65 \times 10^{-3} \pm 1.53 \times 10^{-3}$	101.6 ± 21.6
		$\mu\mu$	1.16 ± 0.23	$5.33 \times 10^{-3} \pm 1.07 \times 10^{-3}$	100.5 ± 21.2
900	7.74×10^{-5}	ee	1.71 ± 0.34	$3.67 \times 10^{-3} \pm 7.36 \times 10^{-4}$	106.0 ± 22.5
		$\mu\mu$	1.19 ± 0.24	$2.56 \times 10^{-3} \pm 5.14 \times 10^{-4}$	89.2 ± 19.0
1100	1.87×10^{-5}	ee	1.73 ± 0.35	$9.03 \times 10^{-4} \pm 1.81 \times 10^{-4}$	84.2 ± 18.8
		$\mu\mu$	1.19 ± 0.24	$6.22 \times 10^{-4} \pm 1.25 \times 10^{-4}$	106.1 ± 22.3
1200	9.53×10^{-6}	ee	1.76 ± 0.35	$4.65 \times 10^{-4} \pm 9.33 \times 10^{-5}$	100.2 ± 21.4
		$\mu\mu$	1.22 ± 0.24	$3.22 \times 10^{-4} \pm 6.46 \times 10^{-5}$	105.1 ± 22.0
1300	4.93×10^{-6}	ee	1.73 ± 0.35	$2.38 \times 10^{-4} \pm 4.76 \times 10^{-5}$	77.0 ± 17.7
		$\mu\mu$	1.20 ± 0.24	$1.64 \times 10^{-4} \pm 3.30 \times 10^{-5}$	104.5 ± 22.0
1400	2.59×10^{-6}	ee	1.79 ± 0.36	$1.29 \times 10^{-4} \pm 2.60 \times 10^{-5}$	98.5 ± 21.3
		$\mu\mu$	1.21 ± 0.24	$8.73 \times 10^{-5} \pm 1.75 \times 10^{-5}$	97.8 ± 20.6
1500	1.38×10^{-6}	ee	1.79 ± 0.36	$6.86 \times 10^{-5} \pm 1.38 \times 10^{-5}$	111.1 ± 23.5
		$\mu\mu$	1.20 ± 0.24	$4.61 \times 10^{-5} \pm 9.26 \times 10^{-6}$	96.3 ± 20.2

Table I.3: Inputs for the $Z'\chi_2 \to l^+l^-\chi_1\chi_1$ LDS σB calculations. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	6.80	ee	0.07 ± 0.01	13.8 ± 2.83	182.6 ± 38.0
		$\mu\mu$	0.05 ± 0.01	10.0 ± 2.07	260.9 ± 52.9
200	2.37	ee	0.18 ± 0.04	12.0 ± 2.43	191.4 ± 39.8
		$\mu\mu$	0.13 ± 0.03	8.88 ± 1.81	258.9 ± 52.4
400	2.89×10^{-1}	ee	0.68 ± 0.14	5.45 ± 1.10	186.6 ± 38.7
		$\mu\mu$	0.41 ± 0.08	$3.28 \pm 6.61 \times 10^{-1}$	256.9 ± 52.0
600	7.34×10^{-2}	ee	1.01 ± 0.20	$2.06 \pm 4.13 \times 10^{-1}$	194.8 ± 40.2
		$\mu\mu$	0.65 ± 0.13	$1.32 \pm 2.65 \times 10^{-1}$	268.6 ± 54.4
800	2.59×10^{-2}	ee	1.21 ± 0.24	$8.71 \times 10^{-1} \pm 1.75 \times 10^{-1}$	182.7 ± 38.4
		$\mu\mu$	0.80 ± 0.16	$5.74 \times 10^{-1} \pm 1.15 \times 10^{-1}$	256.9 ± 52.0
900	1.65×10^{-2}	ee	1.29 ± 0.26	$5.90 \times 10^{-1} \pm 1.18 \times 10^{-1}$	176.1 ± 36.5
		$\mu\mu$	0.86 ± 0.17	$3.92 \times 10^{-1} \pm 7.88 \times 10^{-2}$	243.3 ± 49.3
1100	7.34×10^{-3}	ee	1.35 ± 0.27	$2.76 \times 10^{-1} \pm 5.52 \times 10^{-2}$	190.8 ± 39.4
		$\mu\mu$	0.92 ± 0.18	$1.87 \times 10^{-1} \pm 3.75 \times 10^{-2}$	240.9 ± 48.8
1200	5.08×10^{-3}	ee	1.41 ± 0.28	$2.00 \times 10^{-1} \pm 4.00 \times 10^{-2}$	179.8 ± 37.2
		$\mu\mu$	0.96 ± 0.19	$1.36 \times 10^{-1} \pm 2.73 \times 10^{-2}$	250.8 ± 50.8
1300	3.58×10^{-3}	ee	1.43 ± 0.29	$1.42 \times 10^{-1} \pm 2.84 \times 10^{-2}$	190.7 ± 39.5
		$\mu\mu$	0.98 ± 0.20	$9.72 \times 10^{-2} \pm 1.95 \times 10^{-2}$	245.3 ± 49.6
1400	2.56×10^{-3}	ee	1.46 ± 0.29	$1.04 \times 10^{-1} \pm 2.08 \times 10^{-2}$	185.9 ± 38.9
		$\mu\mu$	0.99 ± 0.20	$7.09 \times 10^{-2} \pm 1.42 \times 10^{-2}$	260.4 ± 52.9
1500	1.86×10^{-3}	ee	1.48 ± 0.30	$7.68 \times 10^{-2} \pm 1.54 \times 10^{-2}$	185.1 ± 38.2
		$\mu\mu$	0.98 ± 0.20	$5.10 \times 10^{-2} \pm 1.02 \times 10^{-2}$	245.4 ± 49.7

Table I.4: Inputs for the EFT $\rightarrow Z'\chi\chi$ HDS σB calculations. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E^{miss}_{T,min} = 50GeV$ and the invariant mass threshold is $m^{min}_{ll} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	7.81×10^{-6}	ee	0.28 ± 0.06	$6.08 \times 10^{-5} \pm 1.23 \times 10^{-5}$	121.2 ± 26.6
		$\mu\mu$	0.23 ± 0.05	$4.96 \times 10^{-5} \pm 1.01 \times 10^{-5}$	105.8 ± 22.4
200	6.81×10^{-7}	ee	0.65 ± 0.13	$1.22 \times 10^{-5} \pm 2.46 \times 10^{-6}$	139.3 ± 29.1
		$\mu\mu$	0.49 ± 0.10	$9.19 \times 10^{-6} \pm 1.85 \times 10^{-6}$	103.0 ± 21.6
300	6.01×10^{-8}	ee	1.01 ± 0.20	$1.69 \times 10^{-6} \pm 3.39 \times 10^{-7}$	113.8 ± 25.0
		$\mu\mu$	0.76 ± 0.15	$1.28 \times 10^{-6} \pm 2.57 \times 10^{-7}$	113.3 ± 23.8
400	8.42×10^{-9}	ee	1.30 ± 0.26	$3.06 \times 10^{-7} \pm 6.13 \times 10^{-8}$	118.0 ± 25.7
		$\mu\mu$	0.92 ± 0.18	$2.16 \times 10^{-7} \pm 4.33 \times 10^{-8}$	107.9 ± 22.6
500	1.80×10^{-9}	ee	1.46 ± 0.29	$7.32 \times 10^{-8} \pm 1.47 \times 10^{-8}$	108.6 ± 24.9
		$\mu\mu$	1.03 ± 0.21	$5.16 \times 10^{-8} \pm 1.04 \times 10^{-8}$	97.9 ± 20.7
600	4.83×10^{-10}	ee	1.55 ± 0.31	$2.09 \times 10^{-8} \pm 4.18 \times 10^{-9}$	117.8 ± 25.0
		$\mu\mu$	1.09 ± 0.22	$1.46 \times 10^{-8} \pm 2.92 \times 10^{-9}$	115.5 ± 24.2
700	1.49×10^{-10}	ee	1.64 ± 0.33	$6.82 \times 10^{-9} \pm 1.37 \times 10^{-9}$	122.2 ± 26.1
		$\mu\mu$	1.15 ± 0.23	$4.76 \times 10^{-9} \pm 9.56 \times 10^{-10}$	113.3 ± 23.9
800	5.12×10^{-11}	ee	1.68 ± 0.34	$2.39 \times 10^{-9} \pm 4.80 \times 10^{-10}$	118.5 ± 25.0
		$\mu\mu$	1.17 ± 0.23	$1.66 \times 10^{-9} \pm 3.34 \times 10^{-10}$	105.3 ± 22.1
900	1.90×10^{-11}	ee	1.71 ± 0.34	$9.05 \times 10^{-10} \pm 1.81 \times 10^{-10}$	122.2 ± 25.8
		$\mu\mu$	1.15 ± 0.23	$6.09 \times 10^{-10} \pm 1.22 \times 10^{-10}$	116.5 ± 24.4
1000	7.47×10^{-12}	ee	1.70 ± 0.34	$3.53 \times 10^{-10} \pm 7.07 \times 10^{-11}$	123.0 ± 26.0
		$\mu\mu$	1.17 ± 0.23	$2.44 \times 10^{-10} \pm 4.89 \times 10^{-11}$	117.0 ± 24.6
1100	3.07×10^{-12}	ee	1.74 ± 0.35	$1.48 \times 10^{-10} \pm 2.97 \times 10^{-11}$	108.7 ± 23.3
		$\mu\mu$	1.17 ± 0.23	$1.00 \times 10^{-10} \pm 2.01 \times 10^{-11}$	100.9 ± 21.3
1200	1.31×10^{-12}	ee	1.75 ± 0.35	$6.41 \times 10^{-11} \pm 1.28 \times 10^{-11}$	122.8 ± 26.0
		$\mu\mu$	1.23 ± 0.25	$4.49 \times 10^{-11} \pm 9.01 \times 10^{-12}$	109.7 ± 23.1
1300	5.80×10^{-13}	ee	1.75 ± 0.35	$2.82 \times 10^{-11} \pm 5.65 \times 10^{-12}$	120.9 ± 26.6
		$\mu\mu$	1.23 ± 0.25	$1.99 \times 10^{-11} \pm 3.99 \times 10^{-12}$	115.4 ± 24.4
1400	2.63×10^{-13}	ee	1.74 ± 0.35	$1.27 \times 10^{-11} \pm 2.55 \times 10^{-12}$	123.9 ± 26.1
		$\mu\mu$	1.18 ± 0.24	$8.64 \times 10^{-12} \pm 1.73 \times 10^{-12}$	107.3 ± 22.6
1500	1.22×10^{-13}	ee	1.78 ± 0.36	$6.02 \times 10^{-12} \pm 1.21 \times 10^{-12}$	115.3 ± 24.6
		$\mu\mu$	1.22 ± 0.24	$4.15 \times 10^{-12} \pm 8.32 \times 10^{-13}$	108.3 ± 22.8

Table I.5: Inputs for the EFT $\rightarrow Z'\chi\chi$ LDS σB calculations. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	3.35×10^{-5}	ee	0.06 ± 0.01	$5.20 \times 10^{-5} \pm 1.07 \times 10^{-5}$	248.4 ± 51.7
		$\mu\mu$	0.04 ± 0.01	$3.98 \times 10^{-5} \pm 8.28 \times 10^{-6}$	292.5 ± 59.2
200	2.67×10^{-6}	ee	0.17 ± 0.03	$1.23 \times 10^{-5} \pm 2.50 \times 10^{-6}$	245.0 ± 50.3
		$\mu\mu$	0.12 ± 0.02	$9.11 \times 10^{-6} \pm 1.86 \times 10^{-6}$	291.6 ± 59.0
300	2.06×10^{-7}	ee	0.36 ± 0.07	$2.06 \times 10^{-6} \pm 4.15 \times 10^{-7}$	248.1 ± 51.0
		$\mu\mu$	0.24 ± 0.05	$1.36 \times 10^{-6} \pm 2.75 \times 10^{-7}$	284.7 ± 57.7
400	2.65×10^{-8}	ee	0.61 ± 0.12	$4.52 \times 10^{-7} \pm 9.09 \times 10^{-8}$	250.6 ± 51.7
		$\mu\mu$	0.40 ± 0.08	$2.93 \times 10^{-7} \pm 5.91 \times 10^{-8}$	281.9 ± 57.0
500	5.46×10^{-9}	ee	0.82 ± 0.16	$1.24 \times 10^{-7} \pm 2.48 \times 10^{-8}$	243.1 ± 50.3
		$\mu\mu$	0.51 ± 0.10	$7.69 \times 10^{-8} \pm 1.55 \times 10^{-8}$	293.3 ± 59.4
600	1.47×10^{-9}	ee	1.03 ± 0.21	$4.21 \times 10^{-8} \pm 8.44 \times 10^{-9}$	244.9 ± 50.8
		$\mu\mu$	0.62 ± 0.12	$2.53 \times 10^{-8} \pm 5.08 \times 10^{-9}$	294.2 ± 59.5
700	4.72×10^{-10}	ee	1.13 ± 0.23	$1.48 \times 10^{-8} \pm 2.97 \times 10^{-9}$	256.1 ± 52.4
		$\mu\mu$	0.73 ± 0.15	$9.56 \times 10^{-9} \pm 1.92 \times 10^{-9}$	298.1 ± 60.3
800	1.73×10^{-10}	ee	1.22 ± 0.24	$5.89 \times 10^{-9} \pm 1.18 \times 10^{-9}$	245.5 ± 51.2
		$\mu\mu$	0.82 ± 0.16	$3.93 \times 10^{-9} \pm 7.89 \times 10^{-10}$	301.5 ± 61.1
900	7.02×10^{-11}	ee	1.25 ± 0.25	$2.45 \times 10^{-9} \pm 4.91 \times 10^{-10}$	257.9 ± 52.7
		$\mu\mu$	0.88 ± 0.18	$1.71 \times 10^{-9} \pm 3.43 \times 10^{-10}$	281.0 ± 57.0
1000	3.09×10^{-11}	ee	1.35 ± 0.27	$1.16 \times 10^{-9} \pm 2.32 \times 10^{-10}$	251.1 ± 51.5
		$\mu\mu$	0.89 ± 0.18	$7.66 \times 10^{-10} \pm 1.54 \times 10^{-10}$	290.3 ± 58.8
1100	1.45×10^{-11}	ee	1.40 ± 0.28	$5.63 \times 10^{-10} \pm 1.13 \times 10^{-10}$	238.6 ± 49.0
		$\mu\mu$	0.95 ± 0.19	$3.85 \times 10^{-10} \pm 7.72 \times 10^{-11}$	291.4 ± 59.0
1200	7.19×10^{-12}	ee	1.39 ± 0.28	$2.78 \times 10^{-10} \pm 5.57 \times 10^{-11}$	249.4 ± 51.0
		$\mu\mu$	0.95 ± 0.19	$1.89 \times 10^{-10} \pm 3.80 \times 10^{-11}$	300.7 ± 60.8
1300	3.74×10^{-12}	ee	1.45 ± 0.29	$1.51 \times 10^{-10} \pm 3.02 \times 10^{-11}$	242.4 ± 50.0
		$\mu\mu$	0.97 ± 0.19	$1.01 \times 10^{-10} \pm 2.03 \times 10^{-11}$	307.5 ± 62.2
1400	2.02×10^{-12}	ee	1.48 ± 0.30	$8.31 \times 10^{-11} \pm 1.67 \times 10^{-11}$	244.0 ± 50.7
		$\mu\mu$	1.00 ± 0.20	$5.65 \times 10^{-11} \pm 1.13 \times 10^{-11}$	315.4 ± 63.9
1500	1.13×10^{-12}	ee	1.49 ± 0.30	$4.69 \times 10^{-11} \pm 9.40 \times 10^{-12}$	230.8 ± 47.4
		$\mu\mu$	0.98 ± 0.20	$3.07 \times 10^{-11} \pm 6.16 \times 10^{-12}$	301.5 ± 61.1

Table I.6: Inputs for the $Z'h_D \to l^+l^-\chi\chi$ HDS σB calculations in SR1. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	1.11	ee	0.04 ± 0.01	1.21 ± 0.26	283.9 ± 58.7
		$\mu\mu$	0.03 ± 0.01	1.06 ± 0.23	282.7 ± 57.2
200	2.46×10^{-1}	ee	0.09 ± 0.02	$5.84 \times 10^{-1} \pm 1.22 \times 10^{-1}$	273.9 ± 55.9
		$\mu\mu$	0.05 ± 0.01	$3.45 \times 10^{-1} \pm 7.37 \times 10^{-2}$	275.4 ± 59.3
400	1.49×10^{-2}	ee	0.09 ± 0.02	$3.90 \times 10^{-2} \pm 8.12 \times 10^{-3}$	281.9 ± 57.4
		$\mu\mu$	0.05 ± 0.01	$2.09 \times 10^{-2} \pm 4.49 \times 10^{-3}$	294.8 ± 59.7
600	2.35×10^{-3}	ee	0.07 ± 0.01	$4.59 \times 10^{-3} \pm 9.69 \times 10^{-4}$	267.8 ± 54.5
		$\mu\mu$	0.03 ± 0.01	$2.07 \times 10^{-3} \pm 4.60 \times 10^{-4}$	302.5 ± 61.2
800	5.43×10^{-4}	ee	0.05 ± 0.01	$7.29 \times 10^{-4} \pm 1.58 \times 10^{-4}$	296.4 ± 60.4
		$\mu\mu$	0.03 ± 0.01	$4.96 \times 10^{-4} \pm 1.11 \times 10^{-4}$	317.0 ± 64.1
900	2.82×10^{-4}	ee	0.04 ± 0.01	$3.40 \times 10^{-4} \pm 7.42 \times 10^{-5}$	270.4 ± 55.1
		$\mu\mu$	0.02 ± 0.00	$1.78 \times 10^{-4} \pm 4.11 \times 10^{-5}$	297.2 ± 60.1
1100	8.40×10^{-5}	ee	0.03 ± 0.01	$7.13 \times 10^{-5} \pm 1.62 \times 10^{-5}$	276.8 ± 57.1
		$\mu\mu$	0.02 ± 0.00	$4.47 \times 10^{-5} \pm 1.06 \times 10^{-5}$	284.8 ± 57.7
1200	4.75×10^{-5}	ee	0.02 ± 0.00	$2.85 \times 10^{-5} \pm 6.70 \times 10^{-6}$	276.0 ± 57.2
		$\mu\mu$	0.02 ± 0.00	$2.57 \times 10^{-5} \pm 6.13 \times 10^{-6}$	293.6 ± 59.4
1300	2.73×10^{-5}	ee	0.03 ± 0.01	$2.23 \times 10^{-5} \pm 5.08 \times 10^{-6}$	286.0 ± 58.2
		$\mu\mu$	0.02 ± 0.00	$1.32 \times 10^{-5} \pm 3.19 \times 10^{-6}$	265.9 ± 57.6
1400	1.60×10^{-5}	ee	0.02 ± 0.00	$1.10 \times 10^{-5} \pm 2.54 \times 10^{-6}$	279.5 ± 57.0
		$\mu\mu$	0.02 ± 0.00	$7.78 \times 10^{-6} \pm 1.89 \times 10^{-6}$	308.7 ± 62.4
1500	9.42×10^{-6}	ee	0.02 ± 0.00	$5.83 \times 10^{-6} \pm 1.36 \times 10^{-6}$	268.5 ± 55.5
		$\mu\mu$	0.01 ± 0.00	$3.13 \times 10^{-6} \pm 8.26 \times 10^{-7}$	303.1 ± 61.3

Table I.7: Inputs for the $Z'h_D \to l^+l^-\chi\chi$ HDS σB calculations in SR2. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E^{miss}_{T,min} = 50GeV$ and the invariant mass threshold is $m^{min}_{ll} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	1.11	ee	0.10 ± 0.02	3.17 ± 0.65	59.6 ± 15.2
		$\mu\mu$	0.08 ± 0.02	2.41 ± 0.50	80.8 ± 16.9
200	2.46×10^{-1}	ee	0.16 ± 0.03	1.07 ± 0.22	66.5 ± 16.4
		$\mu\mu$	0.12 ± 0.02	$7.89 \times 10^{-1} \pm 1.62 \times 10^{-1}$	78.9 ± 16.7
400	1.49×10^{-2}	ee	0.16 ± 0.03	$6.43 \times 10^{-2} \pm 1.32 \times 10^{-2}$	55.5 ± 13.4
		$\mu\mu$	0.10 ± 0.02	$4.34 \times 10^{-2} \pm 8.99 \times 10^{-3}$	83.4 ± 17.6
600	2.35×10^{-3}	ee	0.13 ± 0.03	$8.19 \times 10^{-3} \pm 1.69 \times 10^{-3}$	55.5 ± 13.6
		$\mu\mu$	0.08 ± 0.02	$5.44 \times 10^{-3} \pm 1.14 \times 10^{-3}$	77.9 ± 16.3
800	5.43×10^{-4}	ee	0.08 ± 0.02	$1.27 \times 10^{-3} \pm 2.67 \times 10^{-4}$	63.6 ± 13.9
		$\mu\mu$	0.07 ± 0.01	$9.86 \times 10^{-4} \pm 2.09 \times 10^{-4}$	78.5 ± 16.7
900	2.82×10^{-4}	ee	0.08 ± 0.02	$6.43 \times 10^{-4} \pm 1.35 \times 10^{-4}$	64.9 ± 14.6
		$\mu\mu$	0.06 ± 0.01	$4.35 \times 10^{-4} \pm 9.30 \times 10^{-5}$	80.8 ± 17.1
1100	8.40×10^{-5}	ee	0.06 ± 0.01	$1.43 \times 10^{-4} \pm 3.06 \times 10^{-5}$	59.8 ± 13.3
		$\mu\mu$	0.05 ± 0.01	$1.17 \times 10^{-4} \pm 2.53 \times 10^{-5}$	83.0 ± 17.3
1200	4.75×10^{-5}	ee	0.06 ± 0.01	$7.42 \times 10^{-5} \pm 1.59 \times 10^{-5}$	53.8 ± 14.3
		$\mu\mu$	0.04 ± 0.01	$4.67 \times 10^{-5} \pm 1.03 \times 10^{-5}$	80.8 ± 17.0
1300	2.73×10^{-5}	ee	0.05 ± 0.01	$3.81 \times 10^{-5} \pm 8.25 \times 10^{-6}$	64.3 ± 14.3
		$\mu\mu$	0.04 ± 0.01	$2.81 \times 10^{-5} \pm 6.21 \times 10^{-6}$	78.0 ± 16.4
1400	1.60×10^{-5}	ee	0.04 ± 0.01	$1.88 \times 10^{-5} \pm 4.13 \times 10^{-6}$	54.7 ± 13.5
		$\mu\mu$	0.03 ± 0.01	$1.32 \times 10^{-5} \pm 2.99 \times 10^{-6}$	82.7 ± 17.4
1500	9.42×10^{-6}	ee	0.04 ± 0.01	$1.17 \times 10^{-5} \pm 2.55 \times 10^{-6}$	48.9 ± 13.6
		$\mu\mu$	0.03 ± 0.01	$7.75 \times 10^{-6} \pm 1.75 \times 10^{-6}$	87.6 ± 18.3

Table I.8: Inputs for the $Z'h_D \to l^+l^-\chi\chi$ HDS σB calculations in SR3. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	1.11	ee	0.08 ± 0.02	2.62 ± 0.54	17.9 ± 4.7
		$\mu\mu$	0.05 ± 0.01	1.54 ± 0.33	23.6 ± 5.8
200	2.46×10^{-1}	ee	0.28 ± 0.06	1.91 ± 0.39	15.7 ± 6.7
		$\mu\mu$	0.19 ± 0.04	1.27 ± 0.26	24.5 ± 6.0
400	1.49×10^{-2}	ee	0.82 ± 0.16	$3.38 \times 10^{-1} \pm 6.79 \times 10^{-2}$	15.3 ± 4.6
		$\mu\mu$	0.57 ± 0.11	$2.34 \times 10^{-1} \pm 4.72 \times 10^{-2}$	21.9 ± 5.2
600	2.35×10^{-3}	ee	1.19 ± 0.24	$7.74 \times 10^{-2} \pm 1.55 \times 10^{-2}$	12.7 ± 6.4
		$\mu\mu$	0.84 ± 0.17	$5.47 \times 10^{-2} \pm 1.10 \times 10^{-2}$	20.8 ± 4.9
800	5.43×10^{-4}	ee	1.37 ± 0.27	$2.06 \times 10^{-2} \pm 4.14 \times 10^{-3}$	14.0 ± 6.4
		$\mu\mu$	0.97 ± 0.19	$1.46 \times 10^{-2} \pm 2.93 \times 10^{-3}$	20.1 ± 5.2
900	2.82×10^{-4}	ee	1.44 ± 0.29	$1.13 \times 10^{-2} \pm 2.27 \times 10^{-3}$	15.2 ± 4.7
		$\mu\mu$	1.04 ± 0.21	$8.14 \times 10^{-3} \pm 1.63 \times 10^{-3}$	21.9 ± 5.2
1100	8.40×10^{-5}	ee	1.55 ± 0.31	$3.62 \times 10^{-3} \pm 7.25 \times 10^{-4}$	18.0 ± 6.7
		$\mu\mu$	1.08 ± 0.22	$2.52 \times 10^{-3} \pm 5.06 \times 10^{-4}$	23.7 ± 5.8
1200	4.75×10^{-5}	ee	1.60 ± 0.32	$2.11 \times 10^{-3} \pm 4.23 \times 10^{-4}$	14.2 ± 4.5
		$\mu\mu$	1.10 ± 0.22	$1.46 \times 10^{-3} \pm 2.92 \times 10^{-4}$	20.9 ± 5.0
1300	2.73×10^{-5}	ee	1.61 ± 0.32	$1.22 \times 10^{-3} \pm 2.45 \times 10^{-4}$	18.0 ± 6.7
		$\mu\mu$	1.11 ± 0.22	$8.42 \times 10^{-4} \pm 1.69 \times 10^{-4}$	19.3 ± 5.1
1400	1.60×10^{-5}	ee	1.62 ± 0.32	$7.20 \times 10^{-4} \pm 1.44 \times 10^{-4}$	19.2 ± 5.3
		$\mu\mu$	1.10 ± 0.22	$4.88 \times 10^{-4} \pm 9.80 \times 10^{-5}$	21.5 ± 5.2
1500	9.42×10^{-6}	ee	1.64 ± 0.33	$4.30 \times 10^{-4} \pm 8.62 \times 10^{-5}$	14.4 ± 5.0
		$\mu\mu$	1.08 ± 0.22	$2.84 \times 10^{-4} \pm 5.70 \times 10^{-5}$	19.4 ± 4.6

Table I.9: Inputs for the $Z'h_D \to l^+l^-\chi\chi$ LDS σB calculations in SR1. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E^{miss}_{T,min} = 50GeV$ and the invariant mass threshold is $m^{min}_{ll} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	1.24	ee	0.04 ± 0.01	1.32 ± 0.27	289.1 ± 59.0
		$\mu\mu$	0.03 ± 0.01	1.08 ± 0.23	272.3 ± 58.7
200	1.05	ee	0.07 ± 0.01	2.16 ± 0.44	276.7 ± 56.6
		$\mu\mu$	0.05 ± 0.01	1.32 ± 0.27	297.5 ± 60.1
400	6.08×10^{-1}	ee	0.12 ± 0.02	1.98 ± 0.40	268.2 ± 55.2
		$\mu\mu$	0.06 ± 0.01	1.10 ± 0.23	292.4 ± 59.1
600	2.88×10^{-1}	ee	0.17 ± 0.03	1.37 ± 0.28	276.6 ± 56.5
		$\mu\mu$	0.10 ± 0.02	$7.91 \times 10^{-1} \pm 1.61 \times 10^{-1}$	300.0 ± 60.7
800	1.44×10^{-1}	ee	0.24 ± 0.05	$9.67 \times 10^{-1} \pm 1.95 \times 10^{-1}$	273.4 ± 56.2
		$\mu\mu$	0.14 ± 0.03	$5.77 \times 10^{-1} \pm 1.17 \times 10^{-1}$	268.9 ± 58.1
900	1.04×10^{-1}	ee	0.26 ± 0.05	$7.39 \times 10^{-1} \pm 1.49 \times 10^{-1}$	289.5 ± 59.5
		$\mu\mu$	0.14 ± 0.03	$4.06 \times 10^{-1} \pm 8.21 \times 10^{-2}$	304.2 ± 61.5
1100	5.61×10^{-2}	ee	0.25 ± 0.05	$3.93 \times 10^{-1} \pm 7.91 \times 10^{-2}$	283.6 ± 57.7
		$\mu\mu$	0.14 ± 0.03	$2.13 \times 10^{-1} \pm 4.32 \times 10^{-2}$	293.3 ± 59.3
1200	4.19×10^{-2}	ee	0.25 ± 0.05	$2.93 \times 10^{-1} \pm 5.89 \times 10^{-2}$	287.9 ± 58.7
		$\mu\mu$	0.14 ± 0.03	$1.60 \times 10^{-1} \pm 3.24 \times 10^{-2}$	297.7 ± 60.2
1300	3.16×10^{-2}	ee	0.24 ± 0.05	$2.14 \times 10^{-1} \pm 4.31 \times 10^{-2}$	296.1 ± 60.7
		$\mu\mu$	0.13 ± 0.03	$1.17 \times 10^{-1} \pm 2.38 \times 10^{-2}$	284.8 ± 61.1
1400	2.40×10^{-2}	ee	0.24 ± 0.05	$1.57 \times 10^{-1} \pm 3.17 \times 10^{-2}$	274.4 ± 56.2
		$\mu\mu$	0.13 ± 0.03	$8.35 \times 10^{-2} \pm 1.69 \times 10^{-2}$	303.4 ± 61.4
1500	1.84×10^{-2}	ee	0.23 ± 0.05	$1.20 \times 10^{-1} \pm 2.42 \times 10^{-2}$	277.0 ± 56.6
		$\mu\mu$	0.12 ± 0.02	$5.99 \times 10^{-2} \pm 1.22 \times 10^{-2}$	308.6 ± 62.4

Table I.10: Inputs for the $Z'h_D \to l^+l^-\chi\chi$ LDS σB calculations in SR2. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E^{miss}_{T,min} = 50GeV$ and the invariant mass threshold is $m^{min}_{ll} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	1.24	ee	0.09 ± 0.02	3.07 ± 0.63	52.8 ± 15.1
		$\mu\mu$	0.08 ± 0.02	2.65 ± 0.54	81.0 ± 17.0
200	1.05	ee	0.10 ± 0.02	2.84 ± 0.58	62.5 ± 15.7
		$\mu\mu$	0.07 ± 0.01	2.17 ± 0.44	72.9 ± 15.3
400	6.08×10^{-1}	ee	0.13 ± 0.03	2.16 ± 0.44	67.0 ± 16.6
		$\mu\mu$	0.09 ± 0.02	1.53 ± 0.31	82.6 ± 17.4
600	2.88×10^{-1}	ee	0.21 ± 0.04	1.67 ± 0.34	59.4 ± 13.2
		$\mu\mu$	0.15 ± 0.03	1.18 ± 0.24	83.8 ± 17.5
800	1.44×10^{-1}	ee	0.27 ± 0.05	1.07 ± 0.22	57.1 ± 13.9
		$\mu\mu$	0.18 ± 0.04	$7.07 \times 10^{-1} \pm 1.43 \times 10^{-1}$	74.5 ± 15.6
900	1.04×10^{-1}	ee	0.28 ± 0.06	$8.02 \times 10^{-1} \pm 1.62 \times 10^{-1}$	65.9 ± 14.6
		$\mu\mu$	0.18 ± 0.04	$5.20 \times 10^{-1} \pm 1.05 \times 10^{-1}$	68.1 ± 14.5
1100	5.61×10^{-2}	ee	0.29 ± 0.06	$4.47 \times 10^{-1} \pm 9.00 \times 10^{-2}$	62.9 ± 14.3
		$\mu\mu$	0.19 ± 0.04	$2.98 \times 10^{-1} \pm 6.02 \times 10^{-2}$	81.5 ± 17.2
1200	4.19×10^{-2}	ee	0.31 ± 0.06	$3.58 \times 10^{-1} \pm 7.21 \times 10^{-2}$	56.2 ± 14.6
		$\mu\mu$	0.18 ± 0.04	$2.06 \times 10^{-1} \pm 4.16 \times 10^{-2}$	86.7 ± 18.2
1300	3.16×10^{-2}	ee	0.29 ± 0.06	$2.52 \times 10^{-1} \pm 5.07 \times 10^{-2}$	56.7 ± 13.1
		$\mu\mu$	0.18 ± 0.04	$1.54 \times 10^{-1} \pm 3.11 \times 10^{-2}$	84.9 ± 17.7
1400	2.40×10^{-2}	ee	0.28 ± 0.06	$1.86 \times 10^{-1} \pm 3.75 \times 10^{-2}$	60.0 ± 15.0
		$\mu\mu$	0.17 ± 0.03	$1.17 \times 10^{-1} \pm 2.36 \times 10^{-2}$	83.2 ± 17.5
1500	1.84×10^{-2}	ee	0.28 ± 0.06	$1.41 \times 10^{-1} \pm 2.83 \times 10^{-2}$	50.4 ± 11.6
		$\mu\mu$	0.17 ± 0.03	$8.82 \times 10^{-2} \pm 1.78 \times 10^{-2}$	83.4 ± 17.6

Table I.11: Inputs for the $Z'h_D \to l^+l^-\chi\chi$ LDS σB calculations in SR3. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	1.24	ee	0.07 ± 0.01	2.55 ± 0.52	15.1 ± 6.6
		$\mu\mu$	0.05 ± 0.01	1.58 ± 0.33	20.5 ± 5.0
200	1.05	ee	0.12 ± 0.02	3.51 ± 0.71	22.5 ± 5.7
		$\mu\mu$	0.08 ± 0.02	2.27 ± 0.46	18.8 ± 4.7
400	6.08×10^{-1}	ee	0.14 ± 0.03	2.44 ± 0.49	20.1 ± 5.2
		$\mu\mu$	0.10 ± 0.02	1.66 ± 0.34	20.1 ± 4.8
600	2.88×10^{-1}	ee	0.20 ± 0.04	1.61 ± 0.33	18.4 ± 5.2
		$\mu\mu$	0.15 ± 0.03	1.19 ± 0.24	14.0 ± 4.0
800	1.44×10^{-1}	ee	0.36 ± 0.07	1.45 ± 0.29	22.1 ± 6.0
		$\mu\mu$	0.27 ± 0.05	1.09 ± 0.22	14.1 ± 3.9
900	1.04×10^{-1}	ee	0.47 ± 0.09	1.37 ± 0.27	15.4 ± 6.4
		$\mu\mu$	0.35 ± 0.07	1.01 ± 0.20	15.0 ± 3.8
1100	5.61×10^{-2}	ee	0.65 ± 0.13	1.01 ± 0.20	21.6 ± 5.4
		$\mu\mu$	0.48 ± 0.10	$7.47 \times 10^{-1} \pm 1.50 \times 10^{-1}$	17.0 ± 4.4
1200	4.19×10^{-2}	ee	0.74 ± 0.15	$8.59 \times 10^{-1} \pm 1.72 \times 10^{-1}$	18.9 ± 5.4
		$\mu\mu$	0.53 ± 0.11	$6.20 \times 10^{-1} \pm 1.24 \times 10^{-1}$	16.9 ± 4.3
1300	3.16×10^{-2}	ee	0.80 ± 0.16	$7.04 \times 10^{-1} \pm 1.41 \times 10^{-1}$	20.1 ± 5.5
		$\mu\mu$	0.57 ± 0.11	$5.03 \times 10^{-1} \pm 1.01 \times 10^{-1}$	16.6 ± 4.5
1400	2.40×10^{-2}	ee	0.84 ± 0.17	$5.64 \times 10^{-1} \pm 1.13 \times 10^{-1}$	12.2 ± 6.5
		$\mu\mu$	0.61 ± 0.12	$4.07 \times 10^{-1} \pm 8.16 \times 10^{-2}$	19.9 ± 4.9
1500	1.84×10^{-2}	ee	0.89 ± 0.18	$4.53 \times 10^{-1} \pm 9.09 \times 10^{-2}$	24.0 ± 5.8
		$\mu\mu$	0.63 ± 0.13	$3.20 \times 10^{-1} \pm 6.42 \times 10^{-2}$	19.0 ± 4.7

Table I.12: Inputs for the $Z'\chi_2 \to l^+l^-\chi_1\chi_1$ HDS σB calculations in SR1. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E^{miss}_{T,min} = 50GeV$ and the invariant mass threshold is $m^{min}_{ll} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	6.93×10^{-1}	ee	0.05 ± 0.01	$8.85 \times 10^{-1} \pm 1.90 \times 10^{-1}$	290.3 ± 59.0
		$\mu\mu$	0.04 ± 0.01	$7.31 \times 10^{-1} \pm 1.58 \times 10^{-1}$	296.4 ± 59.9
200	1.41×10^{-1}	ee	0.09 ± 0.02	$3.64 \times 10^{-1} \pm 7.58 \times 10^{-2}$	285.5 ± 58.4
		$\mu\mu$	0.06 ± 0.01	$2.24 \times 10^{-1} \pm 4.77 \times 10^{-2}$	287.7 ± 61.7
400	6.90×10^{-3}	ee	0.07 ± 0.01	$1.25 \times 10^{-2} \pm 2.66 \times 10^{-3}$	275.8 ± 56.5
		$\mu\mu$	0.04 ± 0.01	$7.67 \times 10^{-3} \pm 1.68 \times 10^{-3}$	304.0 ± 61.4
600	8.81×10^{-4}	ee	0.04 ± 0.01	$9.99 \times 10^{-4} \pm 2.20 \times 10^{-4}$	278.9 ± 57.5
		$\mu\mu$	0.02 ± 0.00	$4.92 \times 10^{-4} \pm 1.16 \times 10^{-4}$	296.3 ± 59.9
800	1.66×10^{-4}	ee	0.02 ± 0.00	$1.12 \times 10^{-4} \pm 2.58 \times 10^{-5}$	275.4 ± 56.6
		$\mu\mu$	0.01 ± 0.00	$6.37 \times 10^{-5} \pm 1.61 \times 10^{-5}$	290.0 ± 58.7
900	7.74×10^{-5}	ee	0.02 ± 0.00	$4.86 \times 10^{-5} \pm 1.14 \times 10^{-5}$	285.9 ± 58.7
		$\mu\mu$	0.01 ± 0.00	$2.43 \times 10^{-5} \pm 6.25 \times 10^{-6}$	297.8 ± 60.3
1100	1.87×10^{-5}	ee	0.01 ± 0.00	$7.79 \times 10^{-6} \pm 1.94 \times 10^{-6}$	302.9 ± 61.8
		$\mu\mu$	0.01 ± 0.00	$5.56 \times 10^{-6} \pm 1.48 \times 10^{-6}$	277.3 ± 59.7
1200	9.53×10^{-6}	ee	0.02 ± 0.00	$4.03 \times 10^{-6} \pm 1.03 \times 10^{-6}$	296.8 ± 60.5
		$\mu\mu$	0.01 ± 0.00	$1.83 \times 10^{-6} \pm 5.31 \times 10^{-7}$	294.5 ± 59.6
1300	4.93×10^{-6}	ee	0.01 ± 0.00	$1.42 \times 10^{-6} \pm 3.93 \times 10^{-7}$	276.6 ± 57.0
		$\mu\mu$	0.01 ± 0.00	$1.08 \times 10^{-6} \pm 3.10 \times 10^{-7}$	302.1 ± 61.1
1400	2.59×10^{-6}	ee	0.01 ± 0.00	$5.90 \times 10^{-7} \pm 1.77 \times 10^{-7}$	274.3 ± 56.0
		$\mu\mu$	0.01 ± 0.00	$3.97 \times 10^{-7} \pm 1.23 \times 10^{-7}$	296.7 ± 60.0
1500	1.38×10^{-6}	ee	0.01 ± 0.00	$2.90 \times 10^{-7} \pm 8.66 \times 10^{-8}$	274.9 ± 56.4
		$\mu\mu$	0.00 ± 0.00	$1.82 \times 10^{-7} \pm 6.31 \times 10^{-8}$	282.0 ± 57.1

Table I.13: Inputs for the $Z'\chi_2 \to l^+l^-\chi_1\chi_1$ HDS σB calculations in SR2. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	6.93×10^{-1}	ee	0.12 ± 0.02	2.22 ± 0.46	61.5 ± 14.2
		$\mu\mu$	0.11 ± 0.02	2.06 ± 0.43	73.5 ± 15.5
200	1.41×10^{-1}	ee	0.19 ± 0.04	$7.46 \times 10^{-1} \pm 1.52 \times 10^{-1}$	54.7 ± 13.8
		$\mu\mu$	0.14 ± 0.03	$5.39 \times 10^{-1} \pm 1.11 \times 10^{-1}$	81.3 ± 17.0
400	6.90×10^{-3}	ee	0.13 ± 0.03	$2.49 \times 10^{-2} \pm 5.13 \times 10^{-3}$	63.6 ± 13.9
		$\mu\mu$	0.07 ± 0.01	$1.38 \times 10^{-2} \pm 2.91 \times 10^{-3}$	77.8 ± 16.6
600	8.81×10^{-4}	ee	0.08 ± 0.02	$1.99 \times 10^{-3} \pm 4.17 \times 10^{-4}$	49.2 ± 13.1
		$\mu\mu$	0.05 ± 0.01	$1.29 \times 10^{-3} \pm 2.78 \times 10^{-4}$	83.0 ± 17.3
800	1.66×10^{-4}	ee	0.05 ± 0.01	$2.34 \times 10^{-4} \pm 5.05 \times 10^{-5}$	58.1 ± 13.3
		$\mu\mu$	0.03 ± 0.01	$1.49 \times 10^{-4} \pm 3.31 \times 10^{-5}$	74.3 ± 15.8
900	7.74×10^{-5}	ee	0.05 ± 0.01	$1.07 \times 10^{-4} \pm 2.33 \times 10^{-5}$	59.9 ± 13.4
		$\mu\mu$	0.03 ± 0.01	$6.38 \times 10^{-5} \pm 1.44 \times 10^{-5}$	84.5 ± 17.9
1100	1.87×10^{-5}	ee	0.03 ± 0.01	$1.72 \times 10^{-5} \pm 3.86 \times 10^{-6}$	46.2 ± 13.2
		$\mu\mu$	0.02 ± 0.00	$9.42 \times 10^{-6} \pm 2.26 \times 10^{-6}$	73.5 ± 15.6
1200	9.53×10^{-6}	ee	0.02 ± 0.00	$5.76 \times 10^{-6} \pm 1.37 \times 10^{-6}$	72.5 ± 15.5
		$\mu\mu$	0.02 ± 0.00	$4.05 \times 10^{-6} \pm 1.01 \times 10^{-6}$	77.9 ± 16.5
1300	4.93×10^{-6}	ee	0.02 ± 0.00	$3.18 \times 10^{-6} \pm 7.52 \times 10^{-7}$	74.0 ± 16.3
		$\mu\mu$	0.01 ± 0.00	$1.80 \times 10^{-6} \pm 4.62 \times 10^{-7}$	84.1 ± 17.7
1400	2.59×10^{-6}	ee	0.02 ± 0.00	$1.39 \times 10^{-6} \pm 3.34 \times 10^{-7}$	57.3 ± 14.2
		$\mu\mu$	0.01 ± 0.00	$6.88 \times 10^{-7} \pm 1.95 \times 10^{-7}$	68.8 ± 14.6
1500	1.38×10^{-6}	ee	0.01 ± 0.00	$5.54 \times 10^{-7} \pm 1.45 \times 10^{-7}$	48.6 ± 14.6
		$\mu\mu$	0.01 ± 0.00	$2.79 \times 10^{-7} \pm 8.17 \times 10^{-8}$	80.5 ± 17.0

Table I.14: Inputs for the $Z'\chi_2 \to l^+l^-\chi_1\chi_1$ HDS σB calculations in SR3. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	6.93×10^{-1}	ee	0.09 ± 0.02	1.71 ± 0.36	20.0 ± 5.1
		$\mu\mu$	0.05 ± 0.01	$9.52 \times 10^{-1} \pm 2.02 \times 10^{-1}$	24.6 ± 5.8
200	1.41×10^{-1}	ee	0.35 ± 0.07	1.39 ± 0.28	17.1 ± 6.5
		$\mu\mu$	0.20 ± 0.04	$8.05 \times 10^{-1} \pm 1.64 \times 10^{-1}$	21.8 ± 5.2
400	6.90×10^{-3}	ee	1.04 ± 0.21	$1.99 \times 10^{-1} \pm 4.00 \times 10^{-2}$	20.9 ± 5.4
		$\mu\mu$	0.69 ± 0.14	$1.32 \times 10^{-1} \pm 2.65 \times 10^{-2}$	21.6 ± 5.4
600	8.81×10^{-4}	ee	1.38 ± 0.28	$3.38 \times 10^{-2} \pm 6.78 \times 10^{-3}$	20.3 ± 5.3
		$\mu\mu$	0.96 ± 0.19	$2.36 \times 10^{-2} \pm 4.74 \times 10^{-3}$	18.9 ± 4.8
800	1.66×10^{-4}	ee	1.54 ± 0.31	$7.10 \times 10^{-3} \pm 1.42 \times 10^{-3}$	18.4 ± 5.2
		$\mu\mu$	1.11 ± 0.22	$5.12 \times 10^{-3} \pm 1.03 \times 10^{-3}$	20.4 ± 4.8
900	7.74×10^{-5}	ee	1.63 ± 0.33	$3.51 \times 10^{-3} \pm 7.04 \times 10^{-4}$	18.5 ± 6.6
		$\mu\mu$	1.15 ± 0.23	$2.48 \times 10^{-3} \pm 4.97 \times 10^{-4}$	19.7 ± 4.8
1100	1.87×10^{-5}	ee	1.70 ± 0.34	$8.87 \times 10^{-4} \pm 1.78 \times 10^{-4}$	25.1 ± 6.3
		$\mu\mu$	1.18 ± 0.24	$6.16 \times 10^{-4} \pm 1.24 \times 10^{-4}$	22.9 ± 5.5
1200	9.53×10^{-6}	ee	1.71 ± 0.34	$4.52 \times 10^{-4} \pm 9.07 \times 10^{-5}$	21.4 ± 5.4
		$\mu\mu$	1.17 ± 0.23	$3.09 \times 10^{-4} \pm 6.20 \times 10^{-5}$	20.0 ± 4.8
1300	4.93×10^{-6}	ee	1.70 ± 0.34	$2.33 \times 10^{-4} \pm 4.67 \times 10^{-5}$	22.5 ± 5.8
		$\mu\mu$	1.18 ± 0.24	$1.62 \times 10^{-4} \pm 3.26 \times 10^{-5}$	17.3 ± 4.3
1400	2.59×10^{-6}	ee	1.73 ± 0.35	$1.25 \times 10^{-4} \pm 2.50 \times 10^{-5}$	15.2 ± 6.3
		$\mu\mu$	1.20 ± 0.24	$8.62 \times 10^{-5} \pm 1.73 \times 10^{-5}$	21.2 ± 5.2
1500	1.38×10^{-6}	ee	1.79 ± 0.36	$6.88 \times 10^{-5} \pm 1.38 \times 10^{-5}$	16.0 ± 6.3
		$\mu\mu$	1.17 ± 0.23	$4.48 \times 10^{-5} \pm 8.98 \times 10^{-6}$	18.8 ± 4.5

Table I.15: Inputs for the $Z'\chi_2 \to l^+l^-\chi_1\chi_1$ LDS σB calculations in SR1. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	6.80	ee	0.03 ± 0.01	4.92 ± 1.05	269.8 ± 55.1
		$\mu\mu$	0.02 ± 0.00	3.53 ± 0.77	285.8 ± 57.8
200	2.37	ee	0.08 ± 0.02	5.41 ± 1.11	279.9 ± 57.3
		$\mu\mu$	0.05 ± 0.01	3.33 ± 0.70	300.5 ± 60.8
400	2.89×10^{-1}	ee	0.23 ± 0.05	1.83 ± 0.37	271.4 ± 55.3
		$\mu\mu$	0.12 ± 0.02	$9.94 \times 10^{-1} \pm 2.04 \times 10^{-1}$	294.1 ± 59.5
600	7.34×10^{-2}	ee	0.27 ± 0.05	$5.51 \times 10^{-1} \pm 1.12 \times 10^{-1}$	277.0 ± 56.6
		$\mu\mu$	0.15 ± 0.03	$2.99 \times 10^{-1} \pm 6.10 \times 10^{-2}$	274.4 ± 59.1
800	2.59×10^{-2}	ee	0.29 ± 0.06	$2.09 \times 10^{-1} \pm 4.23 \times 10^{-2}$	266.7 ± 54.9
		$\mu\mu$	0.15 ± 0.03	$1.07 \times 10^{-1} \pm 2.19 \times 10^{-2}$	300.8 ± 60.8
900	1.65×10^{-2}	ee	0.27 ± 0.05	$1.25 \times 10^{-1} \pm 2.52 \times 10^{-2}$	275.8 ± 56.9
		$\mu\mu$	0.14 ± 0.03	$6.33 \times 10^{-2} \pm 1.30 \times 10^{-2}$	292.1 ± 59.1
1100	7.34×10^{-3}	ee	0.26 ± 0.05	$5.20 \times 10^{-2} \pm 1.06 \times 10^{-2}$	272.1 ± 55.5
		$\mu\mu$	0.13 ± 0.03	$2.60 \times 10^{-2} \pm 5.35 \times 10^{-3}$	294.8 ± 59.6
1200	5.08×10^{-3}	ee	0.25 ± 0.05	$3.53 \times 10^{-2} \pm 7.16 \times 10^{-3}$	282.2 ± 57.7
		$\mu\mu$	0.13 ± 0.03	$1.85 \times 10^{-2} \pm 3.80 \times 10^{-3}$	290.0 ± 58.7
1300	3.58×10^{-3}	ee	0.24 ± 0.05	$2.37 \times 10^{-2} \pm 4.81 \times 10^{-3}$	268.5 ± 55.0
		$\mu\mu$	0.13 ± 0.03	$1.26 \times 10^{-2} \pm 2.60 \times 10^{-3}$	294.2 ± 59.5
1400	2.56×10^{-3}	ee	0.25 ± 0.05	$1.79 \times 10^{-2} \pm 3.63 \times 10^{-3}$	293.2 ± 59.7
		$\mu\mu$	0.12 ± 0.02	$8.54 \times 10^{-3} \pm 1.76 \times 10^{-3}$	274.7 ± 59.1
1500	1.86×10^{-3}	ee	0.25 ± 0.05	$1.29 \times 10^{-2} \pm 2.61 \times 10^{-3}$	309.1 ± 63.1
		$\mu\mu$	0.11 ± 0.02	$5.71 \times 10^{-3} \pm 1.18 \times 10^{-3}$	271.1 ± 58.6

Table I.16: Inputs for the $Z'\chi_2 \to l^+l^-\chi_1\chi_1$ LDS σB calculations in SR2. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E^{miss}_{T,min} = 50GeV$ and the invariant mass threshold is $m^{min}_{ll} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	6.80	ee	0.03 ± 0.01	6.16 ± 1.30	72.5 ± 15.6
		$\mu\mu$	0.03 ± 0.01	5.65 ± 1.20	75.8 ± 15.9
200	2.37	ee	0.08 ± 0.02	4.99 ± 1.03	50.6 ± 11.8
		$\mu\mu$	0.06 ± 0.01	3.72 ± 0.77	82.6 ± 17.2
400	2.89×10^{-1}	ee	0.21 ± 0.04	1.66 ± 0.34	63.5 ± 14.4
		$\mu\mu$	0.11 ± 0.02	$8.96 \times 10^{-1} \pm 1.84 \times 10^{-1}$	79.6 ± 16.7
600	7.34×10^{-2}	ee	0.25 ± 0.05	$5.09 \times 10^{-1} \pm 1.03 \times 10^{-1}$	69.4 ± 15.0
		$\mu\mu$	0.16 ± 0.03	$3.26 \times 10^{-1} \pm 6.64 \times 10^{-2}$	77.8 ± 16.6
800	2.59×10^{-2}	ee	0.27 ± 0.05	$1.95 \times 10^{-1} \pm 3.95 \times 10^{-2}$	63.4 ± 13.9
		$\mu\mu$	0.18 ± 0.04	$1.32 \times 10^{-1} \pm 2.69 \times 10^{-2}$	83.5 ± 17.6
900	1.65×10^{-2}	ee	0.28 ± 0.06	$1.27 \times 10^{-1} \pm 2.58 \times 10^{-2}$	64.6 ± 14.4
		$\mu\mu$	0.18 ± 0.04	$8.03 \times 10^{-2} \pm 1.64 \times 10^{-2}$	71.7 ± 15.1
1100	7.34×10^{-3}	ee	0.27 ± 0.05	$5.54 \times 10^{-2} \pm 1.12 \times 10^{-2}$	60.0 ± 15.2
		$\mu\mu$	0.16 ± 0.03	$3.28 \times 10^{-2} \pm 6.70 \times 10^{-3}$	78.3 ± 16.5
1200	5.08×10^{-3}	ee	0.26 ± 0.05	$3.72 \times 10^{-2} \pm 7.54 \times 10^{-3}$	59.2 ± 13.3
		$\mu\mu$	0.15 ± 0.03	$2.17 \times 10^{-2} \pm 4.43 \times 10^{-3}$	82.5 ± 17.3
1300	3.58×10^{-3}	ee	0.27 ± 0.05	$2.65 \times 10^{-2} \pm 5.36 \times 10^{-3}$	60.0 ± 13.4
		$\mu\mu$	0.17 ± 0.03	$1.67 \times 10^{-2} \pm 3.41 \times 10^{-3}$	83.7 ± 17.4
1400	2.56×10^{-3}	ee	0.25 ± 0.05	$1.79 \times 10^{-2} \pm 3.63 \times 10^{-3}$	48.2 ± 14.2
		$\mu\mu$	0.15 ± 0.03	$1.04 \times 10^{-2} \pm 2.12 \times 10^{-3}$	74.9 ± 15.8
1500	1.86×10^{-3}	ee	0.26 ± 0.05	$1.33 \times 10^{-2} \pm 2.69 \times 10^{-3}$	58.2 ± 14.8
		$\mu\mu$	0.14 ± 0.03	$7.25 \times 10^{-3} \pm 1.49 \times 10^{-3}$	78.6 ± 16.5

Table I.17: Inputs for the $Z'\chi_2 \to l^+l^-\chi_1\chi_1$ LDS σB calculations in SR3. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	6.80	ee	0.02 ± 0.00	3.96 ± 0.86	17.5 ± 5.1
		$\mu\mu$	0.01 ± 0.00	1.88 ± 0.44	19.0 ± 4.5
200	2.37	ee	0.08 ± 0.02	5.14 ± 1.06	18.1 ± 6.7
		$\mu\mu$	0.05 ± 0.01	2.99 ± 0.63	18.5 ± 4.5
400	2.89×10^{-1}	ee	0.24 ± 0.05	1.97 ± 0.40	20.0 ± 5.1
		$\mu\mu$	0.17 ± 0.03	1.38 ± 0.28	23.8 ± 5.6
600	7.34×10^{-2}	ee	0.44 ± 0.09	$8.95 \times 10^{-1} \pm 1.80 \times 10^{-1}$	22.0 ± 5.3
		$\mu\mu$	0.32 ± 0.06	$6.49 \times 10^{-1} \pm 1.31 \times 10^{-1}$	19.0 ± 4.9
800	2.59×10^{-2}	ee	0.61 ± 0.12	$4.40 \times 10^{-1} \pm 8.85 \times 10^{-2}$	22.1 ± 7.2
		$\mu\mu$	0.44 ± 0.09	$3.14 \times 10^{-1} \pm 6.33 \times 10^{-2}$	20.5 ± 5.3
900	1.65×10^{-2}	ee	0.71 ± 0.14	$3.27 \times 10^{-1} \pm 6.56 \times 10^{-2}$	18.8 ± 5.0
		$\mu\mu$	0.48 ± 0.10	$2.21 \times 10^{-1} \pm 4.45 \times 10^{-2}$	20.4 ± 5.3
1100	7.34×10^{-3}	ee	0.83 ± 0.17	$1.69 \times 10^{-1} \pm 3.39 \times 10^{-2}$	22.2 ± 7.2
		$\mu\mu$	0.56 ± 0.11	$1.14 \times 10^{-1} \pm 2.30 \times 10^{-2}$	20.3 ± 4.8
1200	5.08×10^{-3}	ee	0.87 ± 0.17	$1.23 \times 10^{-1} \pm 2.48 \times 10^{-2}$	22.4 ± 5.9
		$\mu\mu$	0.61 ± 0.12	$8.65 \times 10^{-2} \pm 1.74 \times 10^{-2}$	20.0 ± 4.9
1300	3.58×10^{-3}	ee	0.94 ± 0.19	$9.35 \times 10^{-2} \pm 1.88 \times 10^{-2}$	16.0 ± 6.6
		$\mu\mu$	0.65 ± 0.13	$6.49 \times 10^{-2} \pm 1.31 \times 10^{-2}$	16.3 ± 4.4
1400	2.56×10^{-3}	ee	0.96 ± 0.19	$6.86 \times 10^{-2} \pm 1.38 \times 10^{-2}$	21.1 ± 5.9
		$\mu\mu$	0.67 ± 0.13	$4.76 \times 10^{-2} \pm 9.57 \times 10^{-3}$	20.1 ± 5.1
1500	1.86×10^{-3}	ee	1.02 ± 0.20	$5.26 \times 10^{-2} \pm 1.06 \times 10^{-2}$	17.1 ± 6.8
		$\mu\mu$	0.67 ± 0.13	$3.47 \times 10^{-2} \pm 6.97 \times 10^{-3}$	18.7 ± 4.5

Table I.18: Inputs for the EFT $\rightarrow Z'\chi\chi$ HDS σB calculations in SR1. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	7.81×10^{-6}	ee	0.06 ± 0.01	$1.29 \times 10^{-5} \pm 2.73 \times 10^{-6}$	298.4 ± 60.7
		$\mu\mu$	0.05 ± 0.01	$9.82 \times 10^{-6} \pm 2.10 \times 10^{-6}$	290.6 ± 58.8
200	6.81×10^{-7}	ee	0.08 ± 0.02	$1.54 \times 10^{-6} \pm 3.24 \times 10^{-7}$	273.0 ± 55.9
		$\mu\mu$	0.06 ± 0.01	$1.12 \times 10^{-6} \pm 2.37 \times 10^{-7}$	289.1 ± 58.5
300	6.01×10^{-8}	ee	0.04 ± 0.01	$7.16 \times 10^{-8} \pm 1.56 \times 10^{-8}$	273.5 ± 56.0
		$\mu\mu$	0.02 ± 0.00	$3.68 \times 10^{-8} \pm 8.57 \times 10^{-9}$	304.3 ± 61.5
400	8.42×10^{-9}	ee	0.07 ± 0.01	$1.55 \times 10^{-8} \pm 3.28 \times 10^{-9}$	296.3 ± 60.4
		$\mu\mu$	0.04 ± 0.01	$8.75 \times 10^{-9} \pm 1.93 \times 10^{-9}$	295.7 ± 59.9
500	1.80×10^{-9}	ee	0.04 ± 0.01	$2.00 \times 10^{-9} \pm 4.39 \times 10^{-10}$	281.8 ± 58.2
		$\mu\mu$	0.02 ± 0.00	$9.02 \times 10^{-10} \pm 2.17 \times 10^{-10}$	286.8 ± 58.0
600	4.83×10^{-10}	ee	0.04 ± 0.01	$5.58 \times 10^{-10} \pm 1.22 \times 10^{-10}$	273.8 ± 56.0
		$\mu\mu$	0.02 ± 0.00	$2.61 \times 10^{-10} \pm 6.14 \times 10^{-11}$	293.0 ± 59.4
700	1.49×10^{-10}	ee	0.03 ± 0.01	$1.08 \times 10^{-10} \pm 2.48 \times 10^{-11}$	279.3 ± 57.2
		$\mu\mu$	0.02 ± 0.00	$7.49 \times 10^{-11} \pm 1.80 \times 10^{-11}$	278.5 ± 59.9
800	5.12×10^{-11}	ee	0.03 ± 0.01	$4.35 \times 10^{-11} \pm 9.86 \times 10^{-12}$	294.4 ± 59.9
		$\mu\mu$	0.02 ± 0.00	$2.20 \times 10^{-11} \pm 5.46 \times 10^{-12}$	288.6 ± 58.5
900	1.90×10^{-11}	ee	0.02 ± 0.00	$8.79 \times 10^{-12} \pm 2.16 \times 10^{-12}$	285.2 ± 58.2
		$\mu\mu$	0.01 ± 0.00	$5.86 \times 10^{-12} \pm 1.54 \times 10^{-12}$	266.9 ± 57.7
1000	7.47×10^{-12}	ee	0.01 ± 0.00	$2.65 \times 10^{-12} \pm 6.95 \times 10^{-13}$	293.1 ± 60.0
		$\mu\mu$	0.01 ± 0.00	$2.70 \times 10^{-12} \pm 6.86 \times 10^{-13}$	299.3 ± 60.6
1100	3.07×10^{-12}	ee	0.02 ± 0.00	$1.29 \times 10^{-12} \pm 3.22 \times 10^{-13}$	302.9 ± 61.8
		$\mu\mu$	0.01 ± 0.00	$8.56 \times 10^{-13} \pm 2.31 \times 10^{-13}$	277.3 ± 59.7
1200	1.31×10^{-12}	ee	0.01 ± 0.00	$5.00 \times 10^{-13} \pm 1.27 \times 10^{-13}$	284.4 ± 57.8
		$\mu\mu$	0.01 ± 0.00	$2.68 \times 10^{-13} \pm 7.76 \times 10^{-14}$	280.5 ± 56.8
1300	5.80×10^{-13}	ee	0.01 ± 0.00	$1.63 \times 10^{-13} \pm 4.49 \times 10^{-14}$	279.0 ± 57.2
		$\mu\mu$	0.01 ± 0.00	$8.75 \times 10^{-14} \pm 2.78 \times 10^{-14}$	295.9 ± 59.9
1400	2.63×10^{-13}	ee	0.01 ± 0.00	$8.52 \times 10^{-14} \pm 2.24 \times 10^{-14}$	263.4 ± 54.6
		$\mu\mu$	0.01 ± 0.00	$4.49 \times 10^{-14} \pm 1.42 \times 10^{-14}$	290.9 ± 58.9
1500	1.22×10^{-13}	ee	0.01 ± 0.00	$1.95 \times 10^{-14} \pm 6.16 \times 10^{-15}$	267.1 ± 54.5
		$\mu\mu$	0.01 ± 0.00	$2.27 \times 10^{-14} \pm 7.01 \times 10^{-15}$	310.0 ± 62.7

Table I.19: Inputs for the EFT $\rightarrow Z'\chi\chi$ HDS σB calculations in SR2. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E^{miss}_{T,min} = 50GeV$ and the invariant mass threshold is $m^{min}_{ll} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	7.81×10^{-6}	ee	0.13 ± 0.03	$2.76 \times 10^{-5} \pm 5.68 \times 10^{-6}$	61.2 ± 14.7
		$\mu\mu$	0.11 ± 0.02	$2.28 \times 10^{-5} \pm 4.71 \times 10^{-6}$	79.9 ± 16.8
200	6.81×10^{-7}	ee	0.19 ± 0.04	$3.52 \times 10^{-6} \pm 7.20 \times 10^{-7}$	64.2 ± 14.6
		$\mu\mu$	0.14 ± 0.03	$2.68 \times 10^{-6} \pm 5.50 \times 10^{-7}$	81.6 ± 17.3
300	6.01×10^{-8}	ee	0.12 ± 0.02	$2.06 \times 10^{-7} \pm 4.26 \times 10^{-8}$	68.0 ± 14.6
		$\mu\mu$	0.08 ± 0.02	$1.26 \times 10^{-7} \pm 2.66 \times 10^{-8}$	83.6 ± 17.5
400	8.42×10^{-9}	ee	0.13 ± 0.03	$3.12 \times 10^{-8} \pm 6.43 \times 10^{-9}$	59.4 ± 13.3
		$\mu\mu$	0.07 ± 0.01	$1.75 \times 10^{-8} \pm 3.68 \times 10^{-9}$	74.2 ± 15.7
500	1.80×10^{-9}	ee	0.08 ± 0.02	$4.08 \times 10^{-9} \pm 8.56 \times 10^{-10}$	66.6 ± 14.7
		$\mu\mu$	0.05 ± 0.01	$2.71 \times 10^{-9} \pm 5.80 \times 10^{-10}$	77.0 ± 16.2
600	4.83×10^{-10}	ee	0.08 ± 0.02	$1.02 \times 10^{-9} \pm 2.15 \times 10^{-10}$	64.6 ± 14.2
		$\mu\mu$	0.05 ± 0.01	$6.78 \times 10^{-10} \pm 1.46 \times 10^{-10}$	80.1 ± 16.9
700	1.49×10^{-10}	ee	0.05 ± 0.01	$2.05 \times 10^{-10} \pm 4.44 \times 10^{-11}$	72.9 ± 16.0
		$\mu\mu$	0.04 ± 0.01	$1.54 \times 10^{-10} \pm 3.39 \times 10^{-11}$	74.5 ± 15.7
800	5.12×10^{-11}	ee	0.05 ± 0.01	$7.33 \times 10^{-11} \pm 1.58 \times 10^{-11}$	62.2 ± 13.7
		$\mu\mu$	0.03 ± 0.01	$4.02 \times 10^{-11} \pm 9.06 \times 10^{-12}$	78.7 ± 16.6
900	1.90×10^{-11}	ee	0.04 ± 0.01	$2.00 \times 10^{-11} \pm 4.42 \times 10^{-12}$	64.3 ± 14.2
		$\mu\mu$	0.03 ± 0.01	$1.53 \times 10^{-11} \pm 3.47 \times 10^{-12}$	74.3 ± 15.7
1000	7.47×10^{-12}	ee	0.04 ± 0.01	$7.97 \times 10^{-12} \pm 1.76 \times 10^{-12}$	56.8 ± 14.5
		$\mu\mu$	0.02 ± 0.00	$4.10 \times 10^{-12} \pm 9.77 \times 10^{-13}$	82.6 ± 17.4
1100	3.07×10^{-12}	ee	0.03 ± 0.01	$2.51 \times 10^{-12} \pm 5.70 \times 10^{-13}$	56.0 ± 14.7
		$\mu\mu$	0.02 ± 0.00	$1.73 \times 10^{-12} \pm 4.09 \times 10^{-13}$	79.3 ± 16.6
1200	1.31×10^{-12}	ee	0.02 ± 0.00	$8.04 \times 10^{-13} \pm 1.91 \times 10^{-13}$	67.0 ± 14.6
		$\mu\mu$	0.02 ± 0.00	$6.02 \times 10^{-13} \pm 1.48 \times 10^{-13}$	77.7 ± 16.3
1300	5.80×10^{-13}	ee	0.02 ± 0.00	$3.16 \times 10^{-13} \pm 7.56 \times 10^{-14}$	56.8 ± 13.0
		$\mu\mu$	0.01 ± 0.00	$1.53 \times 10^{-13} \pm 4.17 \times 10^{-14}$	80.6 ± 17.0
1400	2.63×10^{-13}	ee	0.02 ± 0.00	$1.29 \times 10^{-13} \pm 3.21 \times 10^{-14}$	63.0 ± 14.0
		$\mu\mu$	0.01 ± 0.00	$6.20 \times 10^{-14} \pm 1.77 \times 10^{-14}$	81.6 ± 17.1
1500	1.22×10^{-13}	ee	0.01 ± 0.00	$4.76 \times 10^{-14} \pm 1.22 \times 10^{-14}$	49.9 ± 13.8
		$\mu\mu$	0.01 ± 0.00	$2.68 \times 10^{-14} \pm 7.97 \times 10^{-15}$	82.2 ± 17.1

Table I.20: Inputs for the EFT $\rightarrow Z'\chi\chi$ HDS σB calculations in SR3. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	7.81×10^{-6}	ee	0.09 ± 0.02	$1.90 \times 10^{-5} \pm 3.96 \times 10^{-6}$	19.1 ± 5.3
		$\mu\mu$	0.05 ± 0.01	$9.78 \times 10^{-6} \pm 2.10 \times 10^{-6}$	21.2 ± 5.3
200	6.81×10^{-7}	ee	0.36 ± 0.07	$6.75 \times 10^{-6} \pm 1.37 \times 10^{-6}$	22.4 ± 6.0
		$\mu\mu$	0.20 ± 0.04	$3.85 \times 10^{-6} \pm 7.84 \times 10^{-7}$	19.9 ± 4.7
300	6.01×10^{-8}	ee	0.61 ± 0.12	$1.03 \times 10^{-6} \pm 2.07 \times 10^{-7}$	17.5 ± 5.0
		$\mu\mu$	0.36 ± 0.07	$6.09 \times 10^{-7} \pm 1.23 \times 10^{-7}$	19.4 ± 4.5
400	8.42×10^{-9}	ee	1.03 ± 0.21	$2.42 \times 10^{-7} \pm 4.86 \times 10^{-8}$	16.0 ± 6.5
		$\mu\mu$	0.70 ± 0.14	$1.63 \times 10^{-7} \pm 3.28 \times 10^{-8}$	19.7 ± 4.9
500	1.80×10^{-9}	ee	1.19 ± 0.24	$5.95 \times 10^{-8} \pm 1.19 \times 10^{-8}$	18.0 ± 5.0
		$\mu\mu$	0.83 ± 0.17	$4.17 \times 10^{-8} \pm 8.38 \times 10^{-9}$	21.0 ± 5.0
600	4.83×10^{-10}	ee	1.41 ± 0.28	$1.89 \times 10^{-8} \pm 3.78 \times 10^{-9}$	19.1 ± 5.1
		$\mu\mu$	0.96 ± 0.19	$1.29 \times 10^{-8} \pm 2.60 \times 10^{-9}$	16.7 ± 4.5
700	1.49×10^{-10}	ee	1.47 ± 0.29	$6.09 \times 10^{-9} \pm 1.22 \times 10^{-9}$	26.2 ± 6.3
		$\mu\mu$	1.03 ± 0.21	$4.28 \times 10^{-9} \pm 8.60 \times 10^{-10}$	18.6 ± 4.7
800	5.12×10^{-11}	ee	1.58 ± 0.32	$2.25 \times 10^{-9} \pm 4.52 \times 10^{-10}$	20.1 ± 5.7
		$\mu\mu$	1.11 ± 0.22	$1.58 \times 10^{-9} \pm 3.18 \times 10^{-10}$	19.9 ± 4.9
900	1.90×10^{-11}	ee	1.67 ± 0.33	$8.82 \times 10^{-10} \pm 1.77 \times 10^{-10}$	21.4 ± 5.6
		$\mu\mu$	1.12 ± 0.22	$5.93 \times 10^{-10} \pm 1.19 \times 10^{-10}$	19.0 ± 4.8
1000	7.47×10^{-12}	ee	1.69 ± 0.34	$3.52 \times 10^{-10} \pm 7.05 \times 10^{-11}$	18.5 ± 6.9
		$\mu\mu$	1.18 ± 0.24	$2.46 \times 10^{-10} \pm 4.93 \times 10^{-11}$	20.5 ± 5.1
1100	3.07×10^{-12}	ee	1.69 ± 0.34	$1.44 \times 10^{-10} \pm 2.90 \times 10^{-11}$	21.2 ± 5.7
		$\mu\mu$	1.19 ± 0.24	$1.01 \times 10^{-10} \pm 2.03 \times 10^{-11}$	17.4 ± 4.5
1200	1.31×10^{-12}	ee	1.72 ± 0.34	$6.28 \times 10^{-11} \pm 1.26 \times 10^{-11}$	18.3 ± 5.0
		$\mu\mu$	1.18 ± 0.24	$4.32 \times 10^{-11} \pm 8.68 \times 10^{-12}$	18.7 ± 4.8
1300	5.80×10^{-13}	ee	1.72 ± 0.34	$2.77 \times 10^{-11} \pm 5.55 \times 10^{-12}$	24.1 ± 5.9
		$\mu\mu$	1.19 ± 0.24	$1.91 \times 10^{-11} \pm 3.84 \times 10^{-12}$	18.1 ± 4.9
1400	2.63×10^{-13}	ee	1.74 ± 0.35	$1.28 \times 10^{-11} \pm 2.56 \times 10^{-12}$	17.1 ± 5.1
		$\mu\mu$	1.18 ± 0.24	$8.66 \times 10^{-12} \pm 1.74 \times 10^{-12}$	20.4 ± 4.7
1500	1.22×10^{-13}	ee	1.81 ± 0.36	$6.14 \times 10^{-12} \pm 1.23 \times 10^{-12}$	15.6 ± 6.6
		$\mu\mu$	1.16 ± 0.23	$3.92 \times 10^{-12} \pm 7.87 \times 10^{-13}$	23.0 ± 5.6

Table I.21: Inputs for the EFT $\rightarrow Z'\chi\chi$ LDS σB calculations in SR1. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	3.35×10^{-5}	ee	0.02 ± 0.00	$2.22 \times 10^{-5} \pm 4.78 \times 10^{-6}$	303.7 ± 62.1
		$\mu\mu$	0.02 ± 0.00	$1.70 \times 10^{-5} \pm 3.72 \times 10^{-6}$	271.4 ± 58.6
200	2.67×10^{-6}	ee	0.08 ± 0.02	$6.24 \times 10^{-6} \pm 1.28 \times 10^{-6}$	268.5 ± 54.9
		$\mu\mu$	0.05 ± 0.01	$3.45 \times 10^{-6} \pm 7.23 \times 10^{-7}$	289.9 ± 58.6
300	2.06×10^{-7}	ee	0.08 ± 0.02	$4.55 \times 10^{-7} \pm 9.42 \times 10^{-8}$	269.1 ± 55.0
		$\mu\mu$	0.04 ± 0.01	$2.02 \times 10^{-7} \pm 4.33 \times 10^{-8}$	304.0 ± 61.5
400	2.65×10^{-8}	ee	0.25 ± 0.05	$1.84 \times 10^{-7} \pm 3.73 \times 10^{-8}$	286.9 ± 58.7
		$\mu\mu$	0.12 ± 0.02	$8.56 \times 10^{-8} \pm 1.75 \times 10^{-8}$	284.0 ± 57.5
500	5.46×10^{-9}	ee	0.20 ± 0.04	$2.99 \times 10^{-8} \pm 6.09 \times 10^{-9}$	274.1 ± 55.8
		$\mu\mu$	0.09 ± 0.02	$1.38 \times 10^{-8} \pm 2.86 \times 10^{-9}$	303.3 ± 61.3
600	1.47×10^{-9}	ee	0.28 ± 0.06	$1.13 \times 10^{-8} \pm 2.28 \times 10^{-9}$	270.9 ± 55.7
		$\mu\mu$	0.14 ± 0.03	$5.82 \times 10^{-9} \pm 1.19 \times 10^{-9}$	299.1 ± 60.5
700	4.72×10^{-10}	ee	0.27 ± 0.05	$3.54 \times 10^{-9} \pm 7.17 \times 10^{-10}$	278.1 ± 56.7
		$\mu\mu$	0.14 ± 0.03	$1.80 \times 10^{-9} \pm 3.68 \times 10^{-10}$	291.6 ± 58.9
800	1.73×10^{-10}	ee	0.27 ± 0.05	$1.30 \times 10^{-9} \pm 2.63 \times 10^{-10}$	288.3 ± 58.7
		$\mu\mu$	0.14 ± 0.03	$6.80 \times 10^{-10} \pm 1.39 \times 10^{-10}$	297.4 ± 60.2
900	7.02×10^{-11}	ee	0.27 ± 0.05	$5.35 \times 10^{-10} \pm 1.08 \times 10^{-10}$	279.1 ± 56.9
		$\mu\mu$	0.15 ± 0.03	$2.84 \times 10^{-10} \pm 5.81 \times 10^{-11}$	296.5 ± 60.0
1000	3.09×10^{-11}	ee	0.26 ± 0.05	$2.24 \times 10^{-10} \pm 4.54 \times 10^{-11}$	288.4 ± 59.0
		$\mu\mu$	0.13 ± 0.03	$1.12 \times 10^{-10} \pm 2.30 \times 10^{-11}$	296.4 ± 60.0
1100	1.45×10^{-11}	ee	0.25 ± 0.05	$1.01 \times 10^{-10} \pm 2.05 \times 10^{-11}$	290.0 ± 59.5
		$\mu\mu$	0.14 ± 0.03	$5.63 \times 10^{-11} \pm 1.15 \times 10^{-11}$	278.3 ± 56.3
1200	7.19×10^{-12}	ee	0.25 ± 0.05	$5.01 \times 10^{-11} \pm 1.02 \times 10^{-11}$	287.9 ± 59.1
		$\mu\mu$	0.13 ± 0.03	$2.57 \times 10^{-11} \pm 5.27 \times 10^{-12}$	299.4 ± 60.6
1300	3.74×10^{-12}	ee	0.26 ± 0.05	$2.67 \times 10^{-11} \pm 5.41 \times 10^{-12}$	278.1 ± 56.7
		$\mu\mu$	0.12 ± 0.02	$1.28 \times 10^{-11} \pm 2.63 \times 10^{-12}$	293.2 ± 59.4
1400	2.02×10^{-12}	ee	0.23 ± 0.05	$1.30 \times 10^{-11} \pm 2.65 \times 10^{-12}$	283.7 ± 57.9
		$\mu\mu$	0.13 ± 0.03	$7.09 \times 10^{-12} \pm 1.46 \times 10^{-12}$	290.4 ± 58.7
1500	1.13×10^{-12}	ee	0.23 ± 0.05	$7.19 \times 10^{-12} \pm 1.46 \times 10^{-12}$	268.5 ± 54.8
		$\mu\mu$	0.11 ± 0.02	$3.56 \times 10^{-12} \pm 7.35 \times 10^{-13}$	292.2 ± 59.1

Table I.22: Inputs for the EFT $\to Z'\chi\chi$ LDS σB calculations in SR2. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	3.35×10^{-5}	ee	0.04 ± 0.01	$3.40 \times 10^{-5} \pm 7.15 \times 10^{-6}$	66.7 ± 14.4
		$\mu\mu$	0.04 ± 0.01	$3.32 \times 10^{-5} \pm 6.95 \times 10^{-6}$	72.9 ± 15.4
200	2.67×10^{-6}	ee	0.07 ± 0.01	$5.13 \times 10^{-6} \pm 1.06 \times 10^{-6}$	54.6 ± 13.4
		$\mu\mu$	0.05 ± 0.01	$3.90 \times 10^{-6} \pm 8.12 \times 10^{-7}$	77.0 ± 16.2
300	2.06×10^{-7}	ee	0.09 ± 0.02	$5.01 \times 10^{-7} \pm 1.04 \times 10^{-7}$	61.9 ± 13.6
		$\mu\mu$	0.06 ± 0.01	$3.45 \times 10^{-7} \pm 7.21 \times 10^{-8}$	73.4 ± 15.4
400	2.65×10^{-8}	ee	0.18 ± 0.04	$1.36 \times 10^{-7} \pm 2.76 \times 10^{-8}$	54.9 ± 14.3
		$\mu\mu$	0.11 ± 0.02	$8.41 \times 10^{-8} \pm 1.72 \times 10^{-8}$	71.4 ± 15.0
500	5.46×10^{-9}	ee	0.17 ± 0.03	$2.51 \times 10^{-8} \pm 5.12 \times 10^{-9}$	51.7 ± 13.8
		$\mu\mu$	0.11 ± 0.02	$1.71 \times 10^{-8} \pm 3.51 \times 10^{-9}$	86.1 ± 18.2
600	1.47×10^{-9}	ee	0.25 ± 0.05	$1.03 \times 10^{-8} \pm 2.09 \times 10^{-9}$	50.0 ± 11.3
		$\mu\mu$	0.16 ± 0.03	$6.40 \times 10^{-9} \pm 1.31 \times 10^{-9}$	88.0 ± 18.5
700	4.72×10^{-10}	ee	0.24 ± 0.05	$3.09 \times 10^{-9} \pm 6.27 \times 10^{-10}$	61.6 ± 13.9
		$\mu\mu$	0.16 ± 0.03	$2.05 \times 10^{-9} \pm 4.18 \times 10^{-10}$	87.9 ± 18.5
800	1.73×10^{-10}	ee	0.26 ± 0.05	$1.24 \times 10^{-9} \pm 2.52 \times 10^{-10}$	56.0 ± 12.7
		$\mu\mu$	0.17 ± 0.03	$8.40 \times 10^{-10} \pm 1.71 \times 10^{-10}$	76.9 ± 16.1
900	7.02×10^{-11}	ee	0.28 ± 0.06	$5.53 \times 10^{-10} \pm 1.12 \times 10^{-10}$	51.5 ± 14.2
		$\mu\mu$	0.17 ± 0.03	$3.38 \times 10^{-10} \pm 6.90 \times 10^{-11}$	78.3 ± 16.7
1000	3.09×10^{-11}	ee	0.27 ± 0.05	$2.28 \times 10^{-10} \pm 4.63 \times 10^{-11}$	62.4 ± 14.0
		$\mu\mu$	0.18 ± 0.04	$1.52 \times 10^{-10} \pm 3.10 \times 10^{-11}$	74.9 ± 15.7
1100	1.45×10^{-11}	ee	0.28 ± 0.06	$1.11 \times 10^{-10} \pm 2.26 \times 10^{-11}$	65.0 ± 14.6
		$\mu\mu$	0.16 ± 0.03	$6.63 \times 10^{-11} \pm 1.35 \times 10^{-11}$	78.7 ± 16.4
1200	7.19×10^{-12}	ee	0.27 ± 0.05	$5.43 \times 10^{-11} \pm 1.10 \times 10^{-11}$	64.6 ± 14.5
		$\mu\mu$	0.17 ± 0.03	$3.32 \times 10^{-11} \pm 6.78 \times 10^{-12}$	80.5 ± 16.9
1300	3.74×10^{-12}	ee	0.25 ± 0.05	$2.55 \times 10^{-11} \pm 5.17 \times 10^{-12}$	49.7 ± 13.8
		$\mu\mu$	0.15 ± 0.03	$1.59 \times 10^{-11} \pm 3.26 \times 10^{-12}$	80.5 ± 16.8
1400	2.02×10^{-12}	ee	0.25 ± 0.05	$1.42 \times 10^{-11} \pm 2.88 \times 10^{-12}$	68.5 ± 15.0
		$\mu\mu$	0.13 ± 0.03	$7.47 \times 10^{-12} \pm 1.53 \times 10^{-12}$	81.7 ± 17.2
1500	1.13×10^{-12}	ee	0.25 ± 0.05	$7.93 \times 10^{-12} \pm 1.61 \times 10^{-12}$	69.5 ± 15.3
		$\mu\mu$	0.13 ± 0.03	$4.17 \times 10^{-12} \pm 8.57 \times 10^{-13}$	83.9 ± 17.6

Table I.23: Inputs for the EFT $\rightarrow Z'\chi\chi$ LDS σB calculations in SR3. The first three columns are the Z' mass, the theoretical cross-section times branching ratio σB, and what Z' decay channel we are looking at. The next two are ε_{sig}, which is the signal selection efficiency, and N_{sig}, which is the theoretical number of signal events after the cuts. The last two columns are the number of background events, N_{bkg}, and the events observed in the data, N_{obs}. The uncertainties of ε_{sig}, N_{sig} and N_{bkg} are statistical with an assumed 20% systematic uncertainty. The MET threshold is $E_{T,min}^{miss} = 50GeV$ and the invariant mass threshold is $m_{ll}^{min} = 110GeV$ and is the same for all inputs.

$m_{Z'}$ [GeV]	σB [fb]	Channel	ε_{sig} [$\times 10^{-1}$]	N_{sig}	N_{bkg}
130	3.35×10^{-5}	ee	0.02 ± 0.00	$1.86 \times 10^{-5} \pm 4.05 \times 10^{-6}$	26.5 ± 6.5
		$\mu\mu$	0.01 ± 0.00	$8.90 \times 10^{-6} \pm 2.08 \times 10^{-6}$	15.7 ± 3.9
200	2.67×10^{-6}	ee	0.07 ± 0.01	$5.06 \times 10^{-6} \pm 1.05 \times 10^{-6}$	23.7 ± 6.1
		$\mu\mu$	0.05 ± 0.01	$3.46 \times 10^{-6} \pm 7.24 \times 10^{-7}$	22.6 ± 5.5
300	2.06×10^{-7}	ee	0.13 ± 0.03	$7.71 \times 10^{-7} \pm 1.57 \times 10^{-7}$	14.6 ± 6.5
		$\mu\mu$	0.08 ± 0.02	$4.33 \times 10^{-7} \pm 8.95 \times 10^{-8}$	18.3 ± 4.4
400	2.65×10^{-8}	ee	0.27 ± 0.05	$2.00 \times 10^{-7} \pm 4.04 \times 10^{-8}$	15.9 ± 4.9
		$\mu\mu$	0.17 ± 0.03	$1.27 \times 10^{-7} \pm 2.57 \times 10^{-8}$	20.6 ± 4.8
500	5.46×10^{-9}	ee	0.31 ± 0.06	$4.75 \times 10^{-8} \pm 9.60 \times 10^{-9}$	18.2 ± 4.8
		$\mu\mu$	0.21 ± 0.04	$3.15 \times 10^{-8} \pm 6.39 \times 10^{-9}$	18.5 ± 4.7
600	1.47×10^{-9}	ee	0.46 ± 0.09	$1.89 \times 10^{-8} \pm 3.81 \times 10^{-9}$	16.7 ± 4.9
		$\mu\mu$	0.31 ± 0.06	$1.27 \times 10^{-8} \pm 2.57 \times 10^{-9}$	21.1 ± 5.2
700	4.72×10^{-10}	ee	0.49 ± 0.10	$6.40 \times 10^{-9} \pm 1.29 \times 10^{-9}$	21.5 ± 5.5
		$\mu\mu$	0.35 ± 0.07	$4.56 \times 10^{-9} \pm 9.20 \times 10^{-10}$	17.2 ± 4.3
800	1.73×10^{-10}	ee	0.63 ± 0.13	$3.05 \times 10^{-9} \pm 6.14 \times 10^{-10}$	21.5 ± 5.8
		$\mu\mu$	0.42 ± 0.08	$2.04 \times 10^{-9} \pm 4.11 \times 10^{-10}$	18.7 ± 4.8
900	7.02×10^{-11}	ee	0.72 ± 0.14	$1.41 \times 10^{-9} \pm 2.84 \times 10^{-10}$	18.8 ± 6.9
		$\mu\mu$	0.49 ± 0.10	$9.62 \times 10^{-10} \pm 1.94 \times 10^{-10}$	20.1 ± 4.9
1000	3.09×10^{-11}	ee	0.75 ± 0.15	$6.47 \times 10^{-10} \pm 1.30 \times 10^{-10}$	15.3 ± 6.8
		$\mu\mu$	0.53 ± 0.11	$4.51 \times 10^{-10} \pm 9.08 \times 10^{-11}$	20.8 ± 4.9
1100	1.45×10^{-11}	ee	0.83 ± 0.17	$3.37 \times 10^{-10} \pm 6.76 \times 10^{-11}$	21.1 ± 5.6
		$\mu\mu$	0.59 ± 0.12	$2.40 \times 10^{-10} \pm 4.82 \times 10^{-11}$	23.4 ± 5.6
1200	7.19×10^{-12}	ee	0.90 ± 0.18	$1.79 \times 10^{-10} \pm 3.60 \times 10^{-11}$	19.7 ± 5.6
		$\mu\mu$	0.62 ± 0.12	$1.23 \times 10^{-10} \pm 2.48 \times 10^{-11}$	21.9 ± 5.2
1300	3.74×10^{-12}	ee	0.93 ± 0.19	$9.63 \times 10^{-11} \pm 1.93 \times 10^{-11}$	12.4 ± 6.2
		$\mu\mu$	0.65 ± 0.13	$6.77 \times 10^{-11} \pm 1.36 \times 10^{-11}$	14.0 ± 3.7
1400	2.02×10^{-12}	ee	0.97 ± 0.19	$5.46 \times 10^{-11} \pm 1.10 \times 10^{-11}$	9.2 ± 6.2
		$\mu\mu$	0.66 ± 0.13	$3.70 \times 10^{-11} \pm 7.44 \times 10^{-12}$	18.4 ± 4.5
1500	1.13×10^{-12}	ee	1.02 ± 0.20	$3.20 \times 10^{-11} \pm 6.43 \times 10^{-12}$	24.4 ± 6.5
		$\mu\mu$	0.65 ± 0.13	$2.04 \times 10^{-11} \pm 4.11 \times 10^{-12}$	19.4 ± 4.6

Appendix J

BDT of depth 30

Before conducting the final grid search for the final networks used on the results, we experimented with different hyperparameters to change. On one of grid searches we set the learning rate as $\eta = 0.1$ as the trend showed this giving the best results with less overtraining, and $\lambda = 10^{-5}$. This grid search had `n_estimators` $\in [10, 100, 500, 1000]$ and depth $\in [3, 4, 5, 6]$. The expected significance is shown in Figure J.1. The testing and training AUC can be seen in Figure J.2.

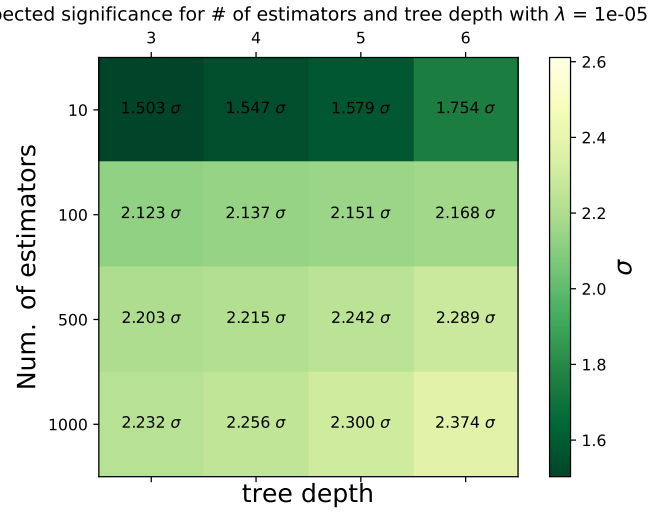
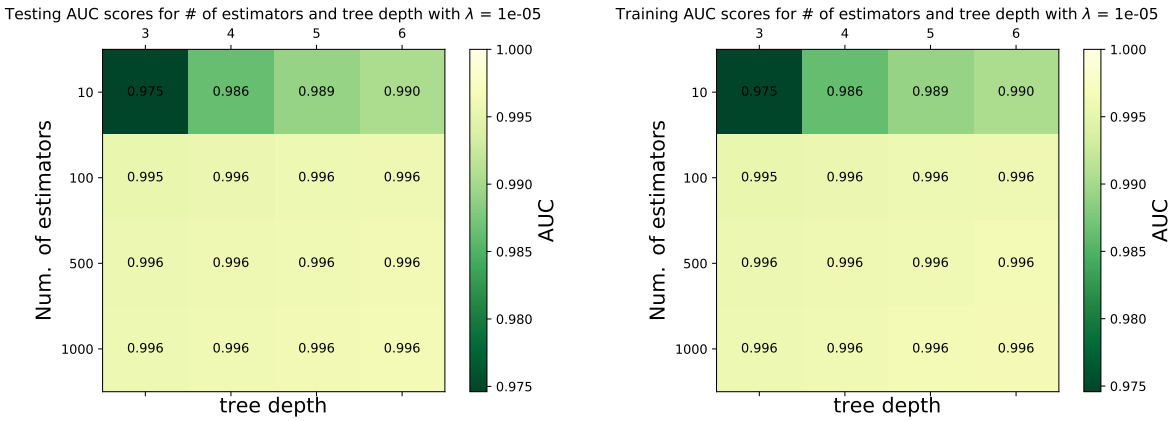


Figure J.1: Grid search expected significance when setting $\lambda = 10^{-5}$ and $\eta = 0.1$

The best networks (depth of 6 and 1000 estimators) feature importance plots is shown in Figure J.3.

However, as we can on Figure J.1 the expected significance does not seem to converge. Because of this, and out of curiosity we decided to keep training the networks with a



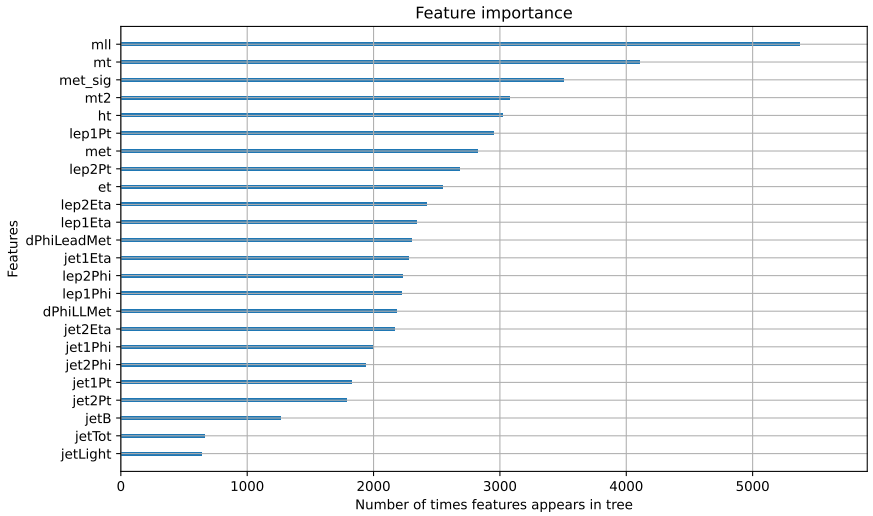
(a) Testing AUC *(b) Training AUC*

Figure J.2: Grid search AUC when setting $\lambda = 10^{-5}$ *and* $\eta = 0.1$

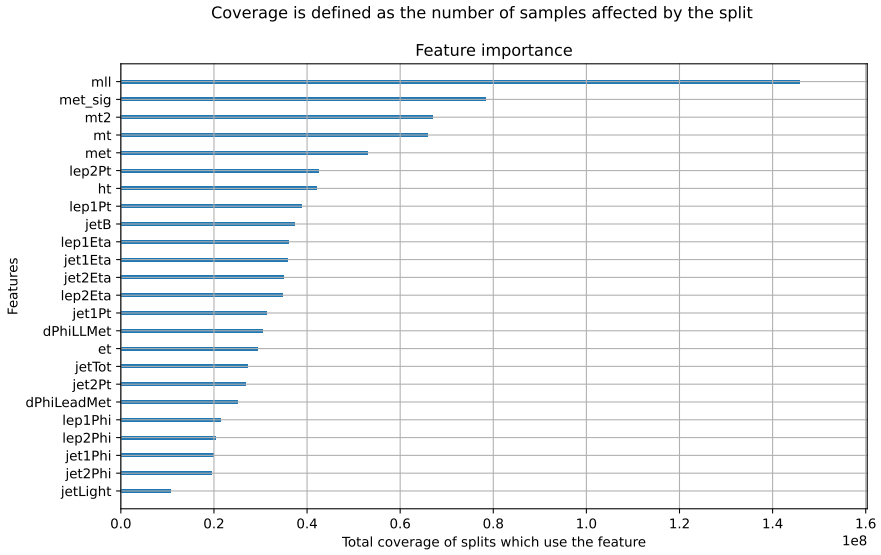
greater depth. The results can be shown in Figure J.4.

In the Figure we see that the networks expected significance does not seem to converge, and rather just keeps getting greater and greater. This is however highly radical as the convention is to normally not have a depth greater than 7, which already is radical. The reason for not having deep BDTs being that the network is highly likely to overtrain and give wrong predictions. However, this was not the case for us as seen for example in Figure J.5.

I understand however that this is controversial since we are splitting a data set, that is at best of size 2^{27}, 30 times. That means that after a depth of 27 there is exactly one event pr branch. So how does a depth of 30 make sense? To help with this we could use a feature in XGBoost to see which features are most important when evaluating a signal. When testing the network trained on the FULL Z' DM data set on a DH HDS $m_{Z'} = 130$ GeV model we get the features shown in Figure J.6 as most important.



(a) Using "weight" metric



(b) Using "coverage" metric

Figure J.3: Feature importance of depth 6 network trained on FULL Z' DM data set when testing it on DH HDS $m_{Z'} = 130$ GeV model.

As we can see these features vary a lot depending on which metric we use to evaluate the importance. When using the "coverage" metric, which as stated is defined as the number of samples affected by the split, we get the features we physically expect to be important when trying to single out a DM model. And this metric is arguably the one we need to use to define what features are important. Since the more samples a feature split, the more powerful it is to separate signal from background.

We can see however that when we use "weight" as a metric, which is the XGBoost

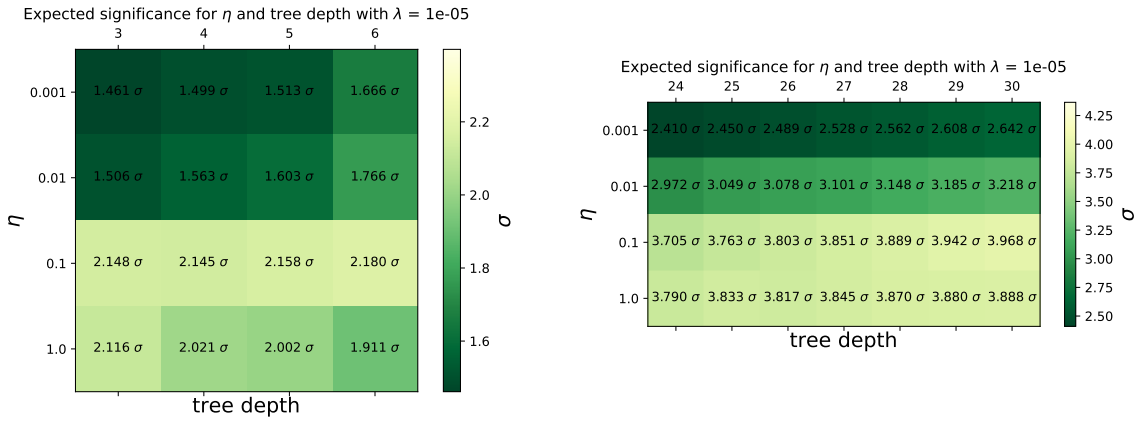
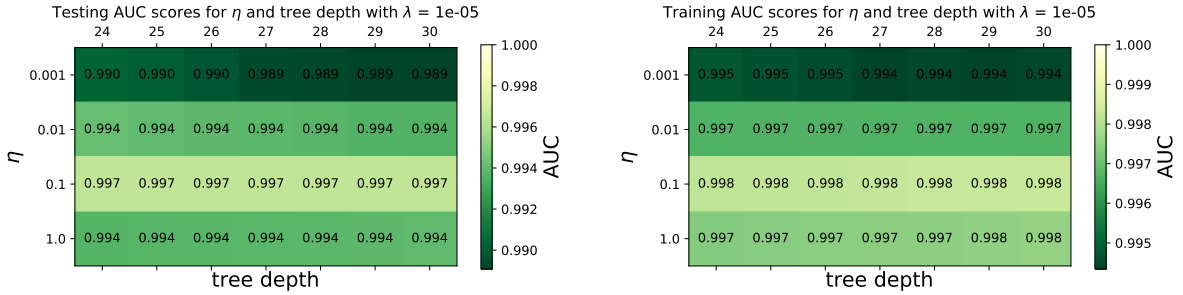


Figure J.4: Grid search expected significance going to a depth of up to 30



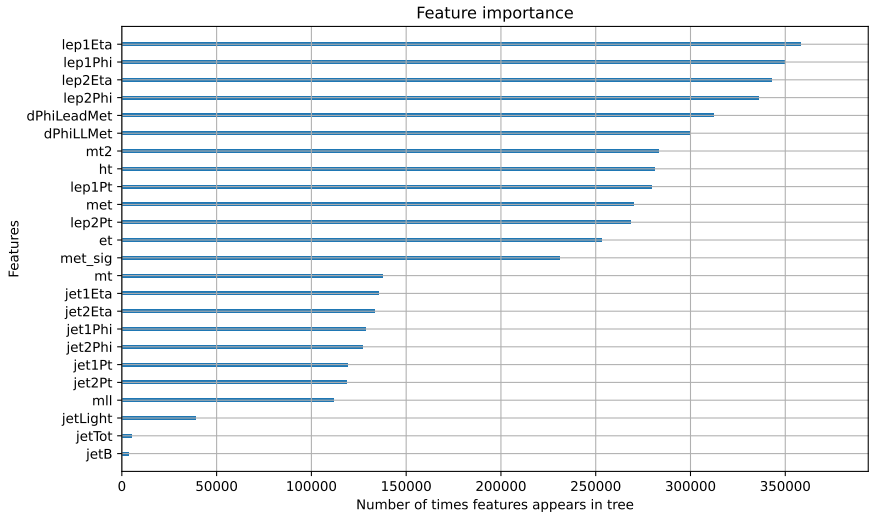
(a) Testing AUC *(b) Training AUC*

Figure J.5: Grid search AUC going to a depth of up to 30

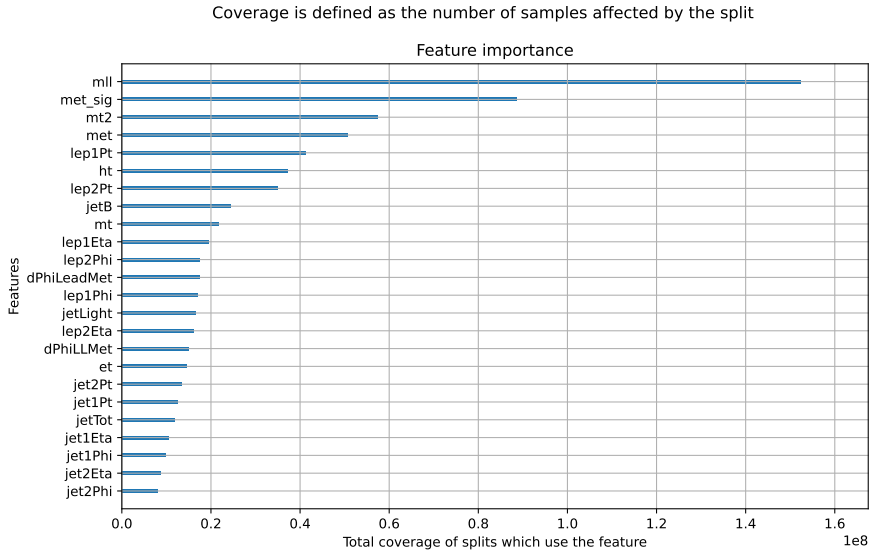
standard metric, we get completely unexpected features that we physically don't expect to be important when trying to single out a DM model. But as described by the metric, the "weight" is the number of times a feature appears in a tree. Which might explain that the reason the pseudorapidity and ϕ range so high on this list, is simply because the tree is struggling to find a pattern here and is trying extra hard to single out DM from SM.

To showcase the difference in signal recognition between the monstrous 30 depth BDT to the more sensible 6 depth BDT, I again tested the networks on the good old DH HDS $m_{Z'} = 130$ GeV model. The results as well as their expected significance can be seen below.

The difference is extreme, when looking at the monster of depth 30 we can get an expected significance of 1.2 σ (without uncertainties) on our model of max 15 events, only



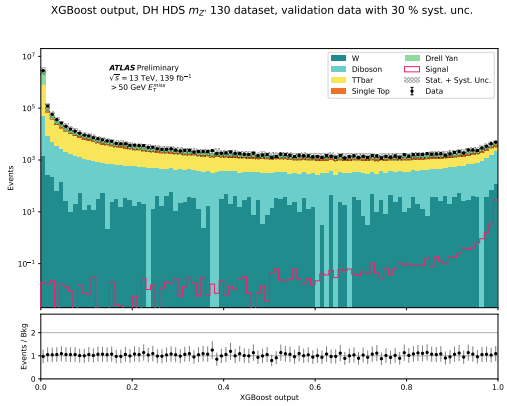
(a) Using "weight" metric



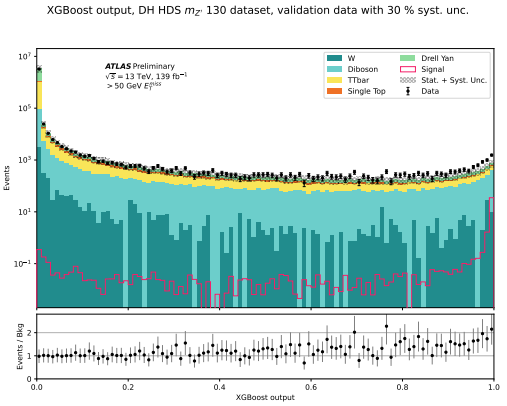
(b) Using "coverage" metric

Figure J.6: Feature importance of depth 30 network trained on FULL Z' DM data set when testing it on DH HDS $m_{Z'} = 130$ GeV model.

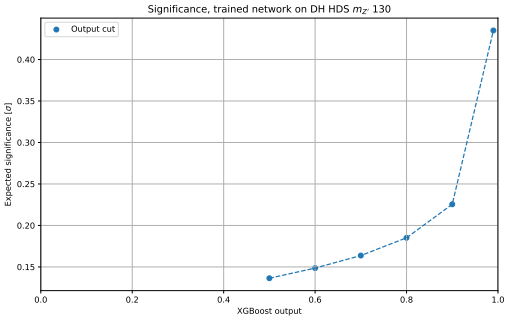
having made a cut of 50 GeV on the missing transverse energy. We can however see that the data and background do not agree to the same degree of the network with depth 6. Using purely statistical uncertainty and assuming a systematic uncertainty of 30%, we see that a few data points do not agree with the MC background. These data points are points the network classified as signal, so if we completely trusted the network this would be a hint of new physics! However, this is the last thing we should assume, and rather take this as a hint that the network is doing something fishy.



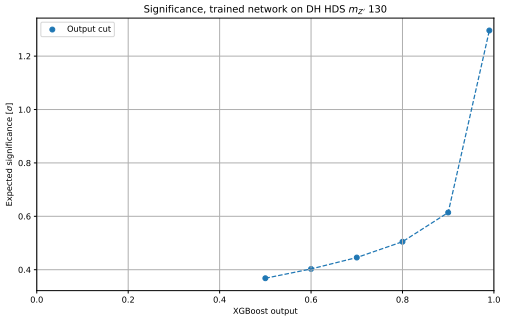
(a) Depth of 6



(b) Depth of 30



(c) The expected significance of a)



(d) The expected significance of b)

Figure J.7: Comparison of the network performance when having a depth of 6 and 30. Figure a) and b) show the validation data of both cases, c) and d) show the expected significance of the validation plots when making a cut on the output.

If I had access to XGBoost with built GPU support, I would increase the number of estimators even more to check if this increases the significance while still having a depth of maximum 6. However, as of now this is not possible. As the weighting method explained in the previous section was not included here, we will drop going to a tree depth of 30, and have a maximum of 6.

Bibliography

1. ATLAS-Collaboration. Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC. Physics Letters B 2012 Sep; 716:1–29. DOI: 10.1016/j.physletb.2012.08.020. Available from: https://doi.org/10.1016%2Fj.physletb.2012.08.020

2. Lee BW and Weinberg S. Cosmological Lower Bound on Heavy-Neutrino Masses. Phys. Rev. Lett. 1977 Jul; 39(4):165–8. DOI: 10.1103/PhysRevLett.39.165. Available from: https://link.aps.org/doi/10.1103/PhysRevLett.39.165

3. Preskill J, Wise MB, and Wilczek F. Cosmology of the Invisible Axion. Phys. Lett. B 1983; 120. Ed. by Srednicki MA:127–32. DOI: 10.1016/0370-2693(83)90637-8

4. Autran M, Bauer K, Lin T, and Whiteson D. Searches for dark matter in events with a resonance and missing transverse energy. Phys. Rev. D 2015 Aug; 92(3):035007. DOI: 10.1103/PhysRevD.92.035007. Available from: https://link.aps.org/doi/10.1103/PhysRevD.92.035007

5. Bauer M, Haisch U, and Kahlhoefer F. Simplified dark matter models with two Higgs doublets: I. Pseudoscalar mediators. JHEP 2017; 05:138. DOI: 10.1007/JHEP05(2017)138. arXiv: 1701.07427 [hep-ph]

6. Jungman G, Kamionkowski M, and Griest K. Supersymmetric dark matter. Phys. Rept. 1996; 267:195–373. DOI: 10.1016/0370-1573(95)00058-5. arXiv: hep-ph/9506380

7. Peskin ME and Schroeder DV. An Introduction to Quantum Field Theory. Reading, USA: Addison-Wesley (1995) 842 p. Westview Press, 1995

8. Thomson M. Modern Particle Physics. Cambridge University Press, 2013

9. Georgi H. LIE ALGEBRAS IN PARTICLE PHYSICS. FROM ISOSPIN TO UNIFIED THEORIES. Vol. 54. 1982

10. Wikipedia. Standard Model. Wikipedia. 2023. Available from: https://en.wikipedia.org/wiki/Standard_Model

11. Peebles PJE and Ratra B. The Cosmological Constant and Dark Energy. Rev. Mod. Phys. 2003; 75. Ed. by Hsu JP and Fine D:559–606. DOI: 10.1103/RevModPhys.75.559. arXiv: astro-ph/0207347

12. Bertone G, Hooper D, and Silk J. Particle dark matter: Evidence, candidates and constraints. Phys. Rept. 2005; 405:279–390. DOI: 10.1016/j.physrep.2004.08.031. arXiv: hep-ph/0404175

13. Einasto J. Dark Matter. 2010. arXiv: 0901.0632 [astro-ph.CO]

14. Dienes KR, Dudas E, and Gherghetta T. Grand unification at intermediate mass scales through extra dimensions. Nuclear Physics B 1999; 537:47–108. DOI: https://doi.org/10.1016/S0550-3213(98)00669-5. Available from: https://www.sciencedirect.com/science/article/pii/S0550321398006695

15. Arkani-Hamed N, Cohen AG, and Georgi H. Electroweak symmetry breaking from dimensional deconstruction. Physics Letters B 2001; 513:232–40. DOI: https://doi.org/10.1016/S0370-2693(01)00741-9. Available from: https://www.sciencedirect.com/science/article/pii/S0370269301007419

16. Search for direct pair production of sleptons and charginos decaying to two leptons and neutralinos with mass splittings near the W-boson mass in $\sqrt{s} = 13\,\text{TeV}$ pp collisions with the ATLAS detector. 2022 Sep. arXiv: 2209.13935 [hep-ex]

17. Branco GC, Ferreira PM, Lavoura L, Rebelo MN, Sher M, and Silva JP. Theory and phenomenology of two-Higgs-doublet models. Phys. Rept. 2012; 516:1–102. DOI: 10.1016/j.physrep.2012.02.002. arXiv: 1106.0034 [hep-ph]

18. Combination and summary of ATLAS dark matter searches using 139 fb^{-1} of $\sqrt{s} =$ 13 TeV p p collision data and interpreted in a two-Higgs-doublet model with a pseudoscalar mediator. Tech. rep. All figures including auxiliary figures are available at https://atlas.web.cern.ch/Atlas/GROUPS/PHYSICS/CONFNOTES/ATLAS-CONF-2021-036. Geneva: CERN, 2021. Available from: https://cds.cern.ch/record/2777863

19. Jackson JD. "Kinematics". Particle Data Group 2008. Available from: https://pdg.lbl.gov/2008/reviews/kinemarpp.pdf

20. Vadla KOH. Search for production of charginos and neutralinos in dilepton final states with the ATLAS detector at the LHC. PhD thesis. University of Oslo, 2022. Available from: https://www.duo.uio.no/handle/10852/97304

21. Barr A, Lester C, and Stephens P. A variable for measuring masses at hadron colliders when missing energy is expected; m_{T2}: the truth behind the glamour. Journal of Physics G: Nuclear and Particle Physics 2003 Sep; 29:2343–63. DOI: 10.1088/0954-3899/29/10/304. Available from: https://doi.org/10.1088%2F0954-3899%2F29%2F10%2F304

22. Martin AD, Stirling WJ, Thorne RS, and Watt G. Parton distributions for the LHC. The European Physical Journal C 2009 Jul; 63:189–285. DOI: 10.1140/epjc/s10052-009-1072-5. Available from: https://doi.org/10.1140%2Fepjc%2Fs10052-009-1072-5

23. Jadach S, Płaczek W, Sapeta S, Siodmok A, and Skrzypek M. Parton distribution functions in Monte Carlo factorisation scheme. Eur. Phys. J. C 2016; 76. Comments: 25 pages, 6 figures:649. DOI: 10.1140/epjc/s10052-016-4508-8. arXiv: 1606.00355. Available from: https://cds.cern.ch/record/2157419

24. The ATLAS Collaboration. The ATLAS Experiment at the CERN Large Hadron Collider. JINST 2008; 3. Also published by CERN Geneva in 2010:S08003. DOI: 10.1088/1748-0221/3/08/S08003. Available from: https://cds.cern.ch/record/1129811

25. Pequenao J and Schaffner P. How ATLAS detects particles: diagram of particle paths in the detector. 2013. Available from: https://cds.cern.ch/record/1505342

26. Guevara R. The HiggsML and a simple Quantum-enhanced Machine Learning algorithm. 2022. Available from: https://github.com/rubenguevara/QuantumMachineLearning/blob/main/Quantum_Machine_Learning.pdf

27. ATLAS Collaboration. Search for new particles in events with one lepton and missing transverse momentum in pp collisions at $\sqrt{s} = 8$ TeV with the ATLAS detector. Journal of High Energy Physics 2014 Sep; 2014. DOI: 10.1007/jhep09(2014)037. Available from: https://doi.org/10.1007%5C%2Fjhep09%5C%282014%5C%29037

28. Cowan G. Discovery sensitivity for a counting experiment with background uncertainty. Tech. rep. http://www.pp.rhul.ac.uk/∼cowan/stat/medsig/medsigNote.pdf. London: Royal Holloway, 2012. Available from: http://www.pp.rhul.ac.uk/~cowan/stat/medsig/medsigNote.pdf

29. Baldi P, Cranmer K, Faucett T, Sadowski P, and Whiteson D. Parameterized neural networks for high-energy physics. The European Physical Journal C 2016 Apr; 76. DOI: 10.1140/epjc/s10052-016-4099-4. Available from: https://doi.org/10.1140%2Fepjc%2Fs10052-016-4099-4

30. Baldi P, Sadowski P, and Whiteson D. Searching for exotic particles in high-energy physics with deep learning. Nature Communications 2014 Jul; 5. DOI: 10.1038/ncomms5308. Available from: https://doi.org/10.1038%2Fncomms5308

31. Hjorth-Jensen M. Week 40: From Stochastic Gradient Descent to Neural networks. 2021 Nov. Available from: https://compphysics.github.io/MachineLearning/doc/pub/week40/html/week40.html

32. Hjorth-Jensen M. Week 41 Constructing a Neural Network code, Tensor flow and start Convolutional Neural Networks. 2022 Aug. Available from: https://compphysics.github.io/MachineLearning/doc/pub/week41/html/week41.html

33. Hjorth-Jensen M. Applied Data Analysis and Machine Learning, 6. Logistic Regression. 2021 Aug. Available from: https://compphysics.github.io/MachineLearning/doc/LectureNotes/_build/html/chapter4.html

34. Kingma DP and Ba J. Adam: A Method for Stochastic Optimization. 2017. arXiv: 1412.6980 [cs.LG]

35. Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Jia Y, Rafal Jozefowicz, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dandelion Mané, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Mike Schuster, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-

Scale Machine Learning on Heterogeneous Systems. Software available from tensorflow.org. 2015. Available from: https://www.tensorflow.org/

36. Morgan JN and Sonquist JA. Problems in the Analysis of Survey Data, and a Proposal. Journal of the American Statistical Association 1963; 58:415–34

37. Quinlan JR. Induction of Decision Trees. Mach Learning 1986; 1:81–106. DOI: https://doi.org/10.1007/BF00116251

38. Friedman JH. Greedy function approximation: A gradient boosting machine. The Annals of Statistics 2001; 29:1189–232. DOI: 10.1214/aos/1013203451. Available from: https://doi.org/10.1214/aos/1013203451

39. Adam-Bourdarios C, Cowan G, Germain C, Guyon I, Kégl B, and Rousseau D. The Higgs boson machine learning challenge. *Proceedings of the NIPS 2014 Workshop on High-energy Physics and Machine Learning*. Ed. by Cowan G, Germain C, Guyon I, Kégl B, and Rousseau D. Vol. 42. Proceedings of Machine Learning Research. Montreal, Canada: PMLR, 2015 Dec :19–55. Available from: https://proceedings.mlr.press/v42/cowa14.html

40. Chen T and Guestrin C. XGBoost: A Scalable Tree Boosting System. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD '16. San Francisco, California, USA: ACM, 2016 :785–94. DOI: 10.1145/2939672.2939785. Available from: http://doi.acm.org/10.1145/2939672.2939785

41. Hjorth-Jensen M. Applied Data Analysis and Machine Learning, 9. Decision trees, overarching aims. 2021 Aug. Available from: https://compphysics.github.io/MachineLearning/doc/LectureNotes/_build/html/chapter6.html#

42. Hjorth-Jensen M. Applied Data Analysis and Machine Learning, 10. Ensemble Methods: From a Single Tree to Many Trees and Extreme Boosting, Meet the Jungle of Methods. 2021 Aug. Available from: https://compphysics.github.io/MachineLearning/doc/LectureNotes/_build/html/chapter7.html

43. TensorFlow. Train and evaluate. Sample weight definition. Available from: https://www.tensorflow.org/guide/keras/train_and_evaluate#sample_weights [Accessed on: 2023 Apr 27]

44. Wikipedia. Receiver operating characteristic. Wikipedia. 2023. Available from: https://en.wikipedia.org/wiki/Receiver_operating_characteristic

45. Gleisberg T, Höche S, Krauss F, Schönherr M, Schumann S, Siegert F, and Winter J. Event generation with SHERPA 1.1. Journal of High Energy Physics 2009 Feb; 2009:7–7. DOI: 10.1088/1126-6708/2009/02/007. Available from: https://doi.org/10.1088%2F1126-6708%2F2009%2F02%2F007

46. Alioli S, Nason P, Oleari C, and Re E. A general framework for implementing NLO calculations in shower Monte Carlo programs: the POWHEG BOX. Journal of High Energy Physics 2010 Jun; 2010. DOI: 10.1007/jhep06(2010)043. Available from: https://doi.org/10.1007%2Fjhep06%282010%29043

47. Sjöstrand T, Ask S, Christiansen JR, Corke R, Desai N, Ilten P, Mrenna S, Prestel S, Rasmussen CO, and Skands PZ. An introduction to PYTHIA 8.2. Computer Physics Communications 2015 Jun; 191:159–77. DOI: 10.1016/j.cpc.2015.01.024. Available from: https://doi.org/10.1016%2Fj.cpc.2015.01.024

48. Brun R and Rademakers F. ROOT: An object oriented data analysis framework. Nucl. Instrum. Meth. A 1997; 389. Ed. by Werlen M and Perret-Gallix D:81–6. DOI: 10.1016/S0168-9002(97)00048-X

49. McKinney W. Data Structures for Statistical Computing in Python. *Proceedings of the 9th Python in Science Conference*. Ed. by Walt S van der and Millman J. 2010 :56–61. DOI: 10.25080/Majora-92bf1922-00a

50. Soyez G. Pileup mitigation at the LHC: A theorist's view. Phys. Rept. 2019; 803:1–158. DOI: 10.1016/j.physrep.2019.01.007. arXiv: 1801.09721 [hep-ph]

51. Catani S, Florian D de, and Grazzini M. Direct Higgs production and jet veto at the Tevatron and the LHC in NNLO QCD. JHEP 2002; 01:015. DOI: 10.1088/1126-6708/2002/01/015. arXiv: hep-ph/0111164

52. Denner A, Franken R, Schmidt T, and Schwan C. NLO QCD and EW corrections to vector-boson scattering into W^+W^- at the LHC. JHEP 2022; 06:098. DOI: 10.1007/JHEP06(2022)098. arXiv: 2202.10844 [hep-ph]

53. Czakon M, Heymes D, and Mitov A. High-precision differential predictions for top-quark pairs at the LHC. Phys. Rev. Lett. 2016; 116:082003. DOI: 10.1103/PhysRevLett.116.082003. arXiv: 1511.00549 [hep-ph]

54. Aad G et al. Jet energy measurement and its systematic uncertainty in proton-proton collisions at $\sqrt{s} = 7$ TeV with the ATLAS detector. Eur. Phys. J. C 2015; 75:17. DOI: 10.1140/epjc/s10052-014-3190-y. arXiv: 1406.0076 [hep-ex]

55. Aad G et al. ATLAS b-jet identification performance and efficiency measurement with $t\bar{t}$ events in pp collisions at $\sqrt{s} = 13$ TeV. Eur. Phys. J. C 2019; 79:970. DOI: 10.1140/epjc/s10052-019-7450-8. arXiv: 1907.05120 [hep-ex]

56. Pedregosa F, Varoquaux G, Gramfort A, Michel V, Thirion B, Grisel O, Blondel M, Prettenhofer P, Weiss R, Dubourg V, Vanderplas J, Passos A, Cournapeau D, Brucher M, Perrot M, and Duchesnay E. Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research 2011; 12:2825–30

57. Mullenbach J. Keras. Source code. 2023. Available from: https://github.com/keras-team/keras/blob/master/keras/engine/training.py [Accessed on: 2023 Apr 27]

58. tedsandler. Does xgboost support *negative* instance weights? Discussion forum. Available from: https://discuss.xgboost.ai/t/does-xgboost-support-negative-instance-weights/1925 [Accessed on: 2023 Apr 27]

59. hcho3. Forbid negative Hessian values. Issue reported on github. Available from: https://github.com/dmlc/xgboost/issues/4372 [Accessed on: 2023 Apr 27]

60. Danziger K, Höche S, and Siegert F. Reducing negative weights in Monte Carlo event generation with Sherpa. 2021. DOI: 10.48550/ARXIV.2110.15211. Available from: https://arxiv.org/abs/2110.15211

61. Abbott B, Alhroob M, Bhopatkar VS, Hopkins WH, Lambert JE, Metcalfe J, Serkin L, Xu W, and Zhu J. Search for Triboson $W^{\pm}W^{\mp}W^{\mp}$ Production Using 13 TeV *pp* Collision Data at ATLAS. Tech. rep. Geneva: CERN, 2020. Available from: https://cds.cern.ch/record/2714377