

https://dspace.org/

Unsupervised machine learning in the search for dark and semi-visible jets

Gusinow, Roy

2021
https://hdl.handle.net/10539/33053

Downloaded from a DSpace repository. https://dspace.org/



WITS UNIVERSITY

University of the Witwatersrand
Johannesburg, South Africa

Unsupervised Machine Learning in the Search for Dark and Semi-visible Jets

Roy Gusinow *(1430013)*

Supervised by ***Prof. Deepak Kar***

MSc Dissertation

Semester 2, 2021

Declaration

I, Roy Jacob Gusinow, declare that this research report is my own, unaided work. It is being submitted for the degree of Master of Science in the field of Physics at the University of the Witwatersrand, Johannesburg. It has not been submitted for any degree or examination at any other university.



Roy Jacob Gusinow, 8^{th} day of October 2021 at University of the Witwatersrand, Johannesburg.

Abstract

Much of dark matter (DM) research has focused on DM candidate particles which are heavy and have little interaction with baryonic matter. However, many theories have proposed DM candidates that do indeed interact with observable matter, particularly resulting in the formation of jets. In certain models, only a portion of dark hadrons produced in a collision will decay back to SM quarks, while the rest will pass through the detector undetected. Semi-visible jets (SVJ) occur when dark hadrons only partially decay to SM hadrons, while for dark jets, the dark hadrons decay fully. Since the final states involve unusual topologies, searches using traditional methods prove challenging to find evidence of resonant signal. New developments in recent years within machine learning community provides a unique opportunity for high-energy particle physics research. In this work is provided a review of anomaly detection methods and its applicability to dark and semi-visible jets in order to uncover new BSM physics.

For my dedushka.

Acknowledgements

I would first like to thank my supervisor, Deepak Kar, for giving myself the opportunity to satiate the questions in physics which I had since a young age. The appreciation I have for his support and care during this pandemic cannot be overstated, and I am glad to have gained such a friend as Deepak.

Thank you to the members of the ATLAS-DJR group for providing valuable feedback to the presentations of my results, as well as the members of the ATLAS-WITS Jets group, particularly Marvin and Sukanya, for their assistance in the more technical aspects of my dissertation.

My thanks to the National Research Foundation for my funding, which financially enabled me to produce this work.

Shoutout to the boys (you know who you are), and especially Mvelo, for always standing by my side.

Thank you to my family — my mother, father, sisters, Raya, as well as Storm and Azuri, who always supported me with abundant love during my research.

מוֹדֶה אֲנִי לְפָנֶיךָ מֶלֶךְ חַי וְקַיָּם, שֶׁהֶחֱזַרְתָּ בִּי נִשְׁמָתִי בְּחֶמְלָה. רַבָּה אֱמוּנָתֶךָ.

Contents

I Anomaly detection for dark and semi-visible jets **1**

1 Introduction **2**

2 Collider Physics and Searches at the LHC **4**
2.1 The Standard Model . . . 4
2.1.1 Fermions . . . 5
2.1.2 Bosons . . . 5
2.1.3 Limitations of the Standard Model . . . 5
2.2 Large Hadron Collider (LHC) . . . 6
2.2.1 The ATLAS Detector . . . 8
2.3 Collider Physics . . . 9
2.3.1 Mandelstam Variables . . . 10
2.3.2 Luminosity . . . 11
2.3.3 Collider Coordinates . . . 11
2.4 Jets . . . 14
2.4.1 Jet Substructure Observables . . . 14

3 Dark Matter Searches **19**
3.1 Review in general Dark Matter theory . . . 19
3.1.1 Mono-X Signatures . . . 20
3.1.2 Effective Field Theories and Simplified Models . . . 20
3.2 Dark and Semi-visible Jets . . . 22
3.3 Simulation . . . 25
3.3.1 Semi-visible Jet Event Generation . . . 25
3.3.2 Dark Jet Event Generation . . . 29

4 Survey of Anomaly Detection Algorithms **35**
4.1 Advances of anomaly detection algorithms in experimental particle physics . . . 35
4.2 Bump-Hunting . . . 36
4.2.1 Assessing Performance . . . 37
4.3 Deep Neural Networks . . . 38
4.3.1 Focal Loss . . . 40
4.4 Autoencoders . . . 42
4.4.1 Variational Autoencoders . . . 43
4.5 Classification without Labels (CWoLa) . . . 44
4.5.1 Toy example . . . 46
4.6 Tag N'Train . . . 48

4.6.1 Method Outline . . . 49
4.7 Dynamic Graph Convolutional Neural Network . . . 49
4.7.1 Convolutions on Jet Images . . . 50
4.7.2 Graph Neural Network . . . 51
4.7.3 Dynamic Graph Convolutional Neural Network . . . 52
4.8 Combination/Lorentz Layer . . . 53
4.8.1 Combination Layer . . . 54
4.8.2 Lorentz Layer . . . 54
4.9 Simulation Assisted Likelihood Anomaly Detection . . . 55

5 Tagging dark and semi-visible jets 56
5.1 Semi-visible Jets . . . 58
5.1.1 *t-channel* . . . 58
5.1.2 *s-channel* . . . 61
5.2 Dark Jets . . . 67
5.2.1 New Benchmarks . . . 68
5.2.2 Comparisons on utilising n_{trk} . . . 71

6 Conclusions 76
6.1 Future Considerations . . . 77

II Use of Gaussian mixture models in aiding COVID-19 hotspot analysis for the Gauteng Province 79

7 Big Data and Artificial Intelligence-Based Hot-Spot Analysis of COVID-19: Gauteng, South Africa, as a case study 80
7.1 Introduction . . . 81
7.2 Methods . . . 82
7.2.1 Data Description . . . 82
7.2.2 Clustering Cases by Geo-Location . . . 82
7.3 Results and Discussion . . . 84
7.3.1 Hot-Spot Density Definition . . . 84
7.3.2 Hot-Spot Activity Analysis . . . 86
7.3.3 Second Wave Risk Index Definition . . . 88
7.3.4 Applications of Hot-Spot Definition for Second Wave . . . 88
7.4 Conclusions . . . 90

Part I

Anomaly detection for dark and semi-visible jets

Chapter 1

Introduction

Modern physics has proven very successful over the past decades in providing robust theoretical explanations to wide range of physical phenomena. An essential part of today's physics theory is the Standard Model (SM), which describes the physical process of the smallest fundamental particles and forces in existence. The construction of large particle colliders has confirmed the paradigm of Standard Model physics, with predictions corresponding to experimental verification at very high precision levels. As recently as 2012, the largest proton-proton collider in the world – the Large Hadron Collider – found strong evidence of the Higgs boson, a fundamental particle which remained the last elusive piece predicted by Standard Model physics. As of today, the Standard Model is complete, however there is still many important observable phenomena that SM physics does not account for. This leads us to extend the Standard Model to accommodate unexplained physics, which we call research into Beyond the Standard Model physics. The probing questions of BSM relevant to this thesis are explanations for the continual evidence of dark matter (DM), a type of matter which is 'invisible' as it is thought to rarely interact with SM particles.

Much of dark matter research has focused on DM candidate particles which are heavy and have little interaction with baryonic matter. However, many theories have proposed DM candidates that do indeed interact with observable matter, particularly resulting in the formation of jets. In certain models, only a portion of dark sector particles produced in a collision will decay back to SM quarks, while the rest will pass through the detector undetected. Semi-visible jets (SVJ) occur when dark sector particles only partially decay to SM particles, while for dark jets resonances (DJR), the dark sector particles decay fully. It is important to highlight that SVJ and DJR have final-state signatures similar to jets originating from Standard Model processes, and therefore it is difficult to differentiate from. Since the final states involve unusual topologies, searches using traditional methods prove challenging to find evidence of dark and semi-visible jets. One proposed idea is to greatly utilise the jet substructure observables in order to assist such searches.

The computer science field of machine learning is a widely publicised domain which seek to find the best optimised algorithms that are capable of certain human tasks, such as labelling and tagging. Examples of these include the recognition of faces in images, or estimating the probability of financial fraud for particular bank transactions. New developments in recent years within machine learning community provides a unique opportunity for high-energy particle physics research, as there is a growing body of research which develop ML algorithms for collider physics specific problems. One example relevant to this thesis is algorithm which can separate interesting BSM collision events, from events which contain SM physics, like ordinary hadronic jets. It is precisely because dark and semi-visible jets look similar in structure to ordinary hadronic jets that it is worth considering an area of unsupervised machine learning called anomaly detection.

Anomaly detection seeks to tag few events or samples within a dataset which are anomalous to the primary trend of the entire sample set. As we expect dark and semi-visible jets to be few in number and slightly varied in structure compared to jet generated by SM particles, anomaly detection can be used effectively.

In this work is provided a review of anomaly detection methods and its applicability to dark and semi-visible jets, by utilising certain jet substructure observables, in order to uncover new BSM physics. Chapter 2 gives an overview of the Large Hadron Collider and a basis of the collider physics we are handling, such as the formation of jets and the corresponding jet observables. Chapter 3 describes how typical dark matter searches are currently carried out and how dark and semi-visible jets structurally deviate from this. The chapter also displays the distribution of jet substructure observables of the specific models we are exploring. Chapter 4 presents a summary of anomaly detection methods we are utilising, by first an explanation of how deep neural networks function and then proceeding with a literature review of particle physics-specific made anomaly detection algorithms. Chapter 5 is a presentation of the findings when applying these anomaly detection techniques to the dark and semi-visible jet models. Finally, Chapter 6 represents a summary of these results and a list of considerations which can be used for future work within the topic.

Chapter 2

Collider Physics and Searches at the LHC

2.1 The Standard Model

The Standard Model (SM) is the currently accepted theory of particle physics, that is, the physics which describes fundamental microscopic particles, as well as the forces (gravitational, electromagnetic as well as strong and weak forces) which govern them [2]. Developed since the 1970's, the model has proved successful at describing a wide range of physical phenomena of the universe. Similar to Mendeleev's periodic table, it is composed of a set of particles, portrayed in Figure 2.1. Bosons are the fundamental mediators of force, while fermions encapsulate the constituent particles of the model. In this section, we will briefly describe particles and force-carriers most relevant to this thesis.

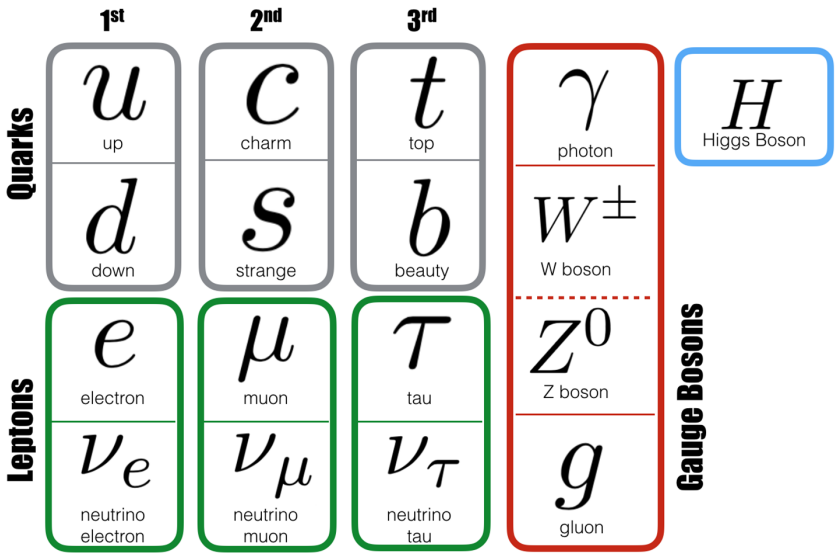


Figure 2.1: Image of the Standard Model of particle physics. Fermions contain three generations of quarks and leptons. The force mediator bosons consist of 4 gauge bosons and 1 scalar boson, the Higgs boson [1].

2.1.1 Fermions

Fermions are particles with half-integer spin, and are split into groups called leptons and quarks. Both these families are further classified into 3 generations each, with each successive generations containing quarks and leptons which are progressively heavier and more unstable than those in the previous generation.

We have six quarks which are as follows: up (u), down (d), charm (c), strange (s), top (t) and bottom (b). Quarks typically interact via the strong force and have quantum numbers such as colour, flavour, spin, charge and also mass.

Similarly, SM also describes six leptons: the electron (e), muon (μ), tau (τ) and its corresponding neutrino pair, the electron neutrino (ν_e), muon neutrino (ν_μ), tau neutrino (ν_τ). The neutrinos have no electric charge and are massless, while their corresponding generation pairs do possess charge and mass.

We introduce a new symmetry to the current set of fermions described. Each Standard Model particle mentioned has a corresponding anti-particle twin with the same characteristics as ordinary matter but opposite attribute. In the case of leptons, the sister particles have opposite the electromagnetic charge, such as the case between electrons and positrons. For quarks, the colour is of the opposite type, and thus anti-quarks can have anti-red or anti-blue colour. Collectively, quarks and gluons are referred to as partons.

2.1.2 Bosons

Bosons are also composed of two families: gauge and scalar bosons. As mentioned previously, particles interact through force mediators, which transmit three fundamental forces. Photons (γ) mediate the electromagnetic force. Gluons (g) mediate the strong force. The rest of the gauge bosons are the Z and $W^{\pm}$, all with spin-1. The only scalar boson, which has spin-0, is the now-famous Higgs boson (H), a manifestation of the Higgs Field. The Higgs give mass to the Z and W bosons, as well as to the non-neutrino fermions [3] [4] [5] [6].

Quantum Electrodynamics (QED) describes the field of all fermions as they experience the electromagnetic force, while Quantum Chromodynamics (QCD) enlightens us on the phenomena of the strong interaction and its effect on quarks. These two bodies of research make up essential contributions to the topic called Quantum Field Theory. An important implication of the strong interactions and QCD, is that colour charged particles cannot exist freely, which means quarks will form colour neutral particles such as mesons. A main consequence is that quarks do not exist isolated, and thus undergo hadronisation, and form hadrons (like neutrons and protons).

2.1.3 Limitations of the Standard Model

Despite the success of the Standard Model in describing many important physical phenomenona of the universe, there are still gaps in knowledge which the current theoretical framework cannot explain. We mention a few important probing questions that may require the extension of the Standard Model in order to answer.

One important issue is known as the hierarchy problem. Initially, this was noted in Quantum Field Theory as a strange discrepancy between the weak nuclear force and the gravitational force — namely, 24 orders of magnitude different. Further investigations into this question lead theoretical physicists to note a serious problem.

It is important to note that the Higgs field contains 250 GeV of energy. It is therefore non-zero and has a non-zero average vaccuum energy throughout the universe. This field gives W and Z bosons masses of around 100 GeV each. Theory tells us that this field is an extremely unstable

state, and we ought to have seen the average vacuum energy of the Higgs field become zero or as large as the Planck energy. These are the two 'natural' values that one would expect the Higgs field to take on, as the Planck mass is the energy level of grand unification theories. In other words, the hierarchy problem outlines that something extremely unlikely occurred. An incredible amount of fine-tuning calculations are required to resolve the discrepancy between the weak nuclear force and gravitational force, at a magnitude of 10^{24} times. Quantum corrections made to the Higgs field cancel out at such fine margins to make the Higgs boson stable [7].

Investigations into this unnaturalness gives evidence that new forces, fields, energies and particles are needed to solve this fine-tuned value, so theories such as supersymmetry are introduced [8]. Philosophic considerations include the anthropic principle, which states that we inhabit a unique universe or section of the universe where the Higgs field is 250 GeV. The reason given is that only this type of inverse is stable enough to host intelligent life that is able to observe such phenomena. Regardless of explanation, these new theories all can be proven or disproven via experiments at the Large Hadron Collider, which discovered the Higgs Boson mentioned previously [9] [10].

As mentioned previously, leptons have corresponding anti-particles pairs, which have the same mass but opposite charge. So, the antiparticle of a hydrogen atom can be constructed by pairing a positron (electron antiparticle) and antiproton (two up anti-quarks and one down anti-quark), which makes the hydrogen atom have a negative charge. When a hydrogen and anti-hydrogen atom collide, they will annihilate. The Big Bang created equal proportions of particles and anti-particles, yet current astrophysical observations of the universe show that much more standard matter than anti-matter. The symmetry of particles and anti-particles behaving the same way is sometimes violated, as was noted in Fermilab's KTeV experiment. Kaons were noted to decay electrons more often than positrons, which highlights the asymmetry we observe currently [11] [12].

Lastly, and most relevant to this thesis, is the existence of dark matter and dark energy, which is described in detail in Section 3.1.

2.2 Large Hadron Collider (LHC)

The Large Hadron Collider is the world's largest proton-proton collider, with its ring's circumference at 27 kilometres, at the European Organization for Nuclear Research (CERN). In Geneva, Switzerland, the LHC hosts various experiments within its complex. Namely, these are the ATLAS, CMS, ALICE and LHCb experiments [13]. The goal of its construction was to probe fundamental questions related to physics, which were not able to be answered by the Standard Model developed in the 1970s. Standard Model (SM) physics has proven successful in accurately describing a wide variety of physical phenomena, but still leaves certain questions unanswered — such as why there is an imbalance between matter and antimatter, as well as the nature of dark matter and others mentioned in Section 2.1.3. The historic discovery of the Higgs boson and Higgs mechanism in 2013, demonstrated the justification for such a massive collider, reaching appropriate energy levels of particles required to make new discoveries in physics.

We collide protons at high energies in order to study the results of the collision. This is because as we move to higher energy collisions, we are able to produce more heavy particles, some of which may be associated with interesting physics processes. The particles produced then interact with the layers of the detector and we may infer the physical processes which have taken place. For particles like electrons, detection is easy, whilst in the case of particles like neutrino, detection is difficult as they rarely interact with ordinary matter.

In order to make such probes into new physics, particle colliders like the LHC work as follows. An electric field is used to strip hydrogen gas of its electrons, thereby producing protons. From

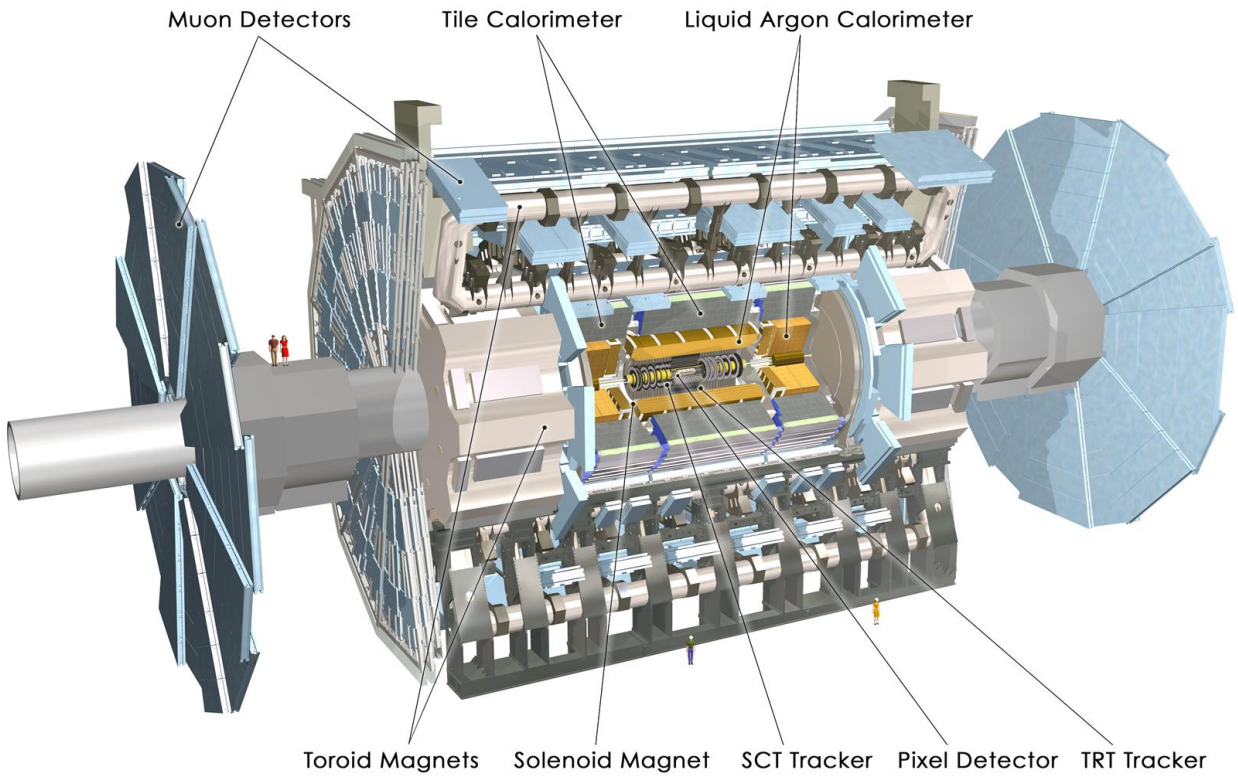


Figure 2.2: Schematic diagram of the ATLAS detector depicting the scale of the entire detector as well as various components which read different physical properties of the detector [20].

this point, the protons are fed into progressively more powerful accelerators in order to achieve appropriate energies required for exploring new physics. The linear accelerator, LINAC reaches an energy of 50 MeV, then its through the Proton Synchotron Booster (PSB), Proton Synchotron (PS) and Super Proton Synchotron (SPS) where the protons are expected to reach 1.4 GeV, 25 GeV and 450 GeV respectively. These beams of accelerated particle are then injected into the LHC ring in clockwise and anti-clockwise directions where each beam may now reach a maximum energy of 6.5 TeV. Thus, two beams may be brought to collide with each other inside an experiment's detector, such as ATLAS's [14].

A run marks the period when the LHC conducts collisions and collects the data from them. The LHC completed its first run for 2009–2013, after which the first long shutdown took place and further repairs and upgrades were made. The discovery of the Higgs boson was due to data collected from this first run [15], as well as the existence of exotic hadrons such as the tetraquark candidate $Z(4430)$ [16]. During the second run from 2015 to 2018, beam energy reached 6.5 TeV, making collisions of 13 TeV now possible [17]. Notably. ATLAS and CMS experiments noted an excess at around 750 GeV for a diphoton invariant mass spectrum, which ultimately did not indicate a new discovery [18]. Run 3 is expected to begin in March 2022, where the integrated luminosity — which is proportional to the number of collisions — will be comparable to the integrated luminosity of Run 2 [19].

2.2.1 The ATLAS Detector

Two beams in the LHC's ring may be directed to collide at any one of the LHC's four detectors mentioned. The largest detector, ATLAS, is the one we focus on in this thesis. The cylinder shape wraps around the point of collision, where ATLAS has diameter of 25 m and a length of 46 m and a weight of approximately 7000 tonnes. Rather than focus on particular physics process, ATLAS is a general-purpose detector designed to read out a wide variety of new physics signal, regardless of what form it may take. This records important aspects of new particles and their decay products, such as the mass, energy, momentum and spin. The detector is constructed in several layers, each dedicated to reading specific physical attributes, and thus require a complex system in order to record them from collisions [21]. The ATLAS detector and its various layers are depicted in Figure 2.2.

Inner Detector

As the name suggest, the Inner Detector is the innermost part of the ATLAS system. It consists of three sections of consecutive surrounding layers.

3.3 cm from the beam axis sits the Pixel Detector composed of 92×10^6 silicon pixel elements. These are designed to read out small energy deposits left over from charged particles interacting with them. The 1744 detector modules are able to reconstruct the momentum of the charged particle.

The Semi-Conductor Tracker (SCT) surrounds this first detector. Similar in concept, the SCT uses silicon strips instead of pixels, in which 4000 modules each contain 6 million strips of silcone sensors. This system allows the charged particle's track to be reconstructed accurately.

The next section is the Transition Radiation Tracker (TRT). Consisting this time of straw tubes filled with a gas mixture instead of silicon strips. Charged particles ionise when interacting with the gas, which then trigger an electrical signal to be sent. From the type of signal sent, we can sometimes deduce the particle which interacted, such as electrons and positron. Also, as its name suggest, we also further reconstruct the charged particle's track [22].

Completed in 2014, a new section of the inner detector was added: the Insertable B-Layer (IBL). It is measures 3.3 radius from the beam axis, making it the new innermost pixel detector. The addition of this layer was made because significant radiation damage due to increasing energy of collision made it possible to damage the components of the inner detector, and thus measuring tracking information of particle would become difficult. The introduction of this layer has ensured high quality b-tagging and tracking information for the inner detector [23]

Magnet System

Surrounding the inner detector are two superconducting magnet systems: the solenoidal and toroidal. We note that charged particles travelling through a magnetic field will experience a Lorentz force, and thus a linear trajectory will begin to bend. From this curvature, we may infer the momentum of the charged particle, as well as the type of charge. The inner solenoid generates a magnetic field of 2 T with the a thickness of only 4.5 cm. The outer toroid magnet consists of eight coils between the calorimeter and muon system, making it the world's largest toroid magnet. It is used to measure the momentum of muons [24].

Calorimeters

Past the inner detector and solenoid magnet system is the calorimeter section composed of two calorimeter detectors: the electromagnetic (EM) detector and the Hadronic Calorimeter detector. They are used to absorb most incoming particles' energy and return measurements related to the particles' energies. Composed of high-density metals and active materials, the calorimeters are capable of reading out particle showers which is important in understanding the underlying collision made.

The EM calorimeter detects photon and charged particles, that is, particles which interact electromagnetically. It is capable of measuring the particle's location and energy deposited to a high precision. An important component of the EM calorimeter is the liquid argon (LAr) calorimeter, which is dedicated to absorbing the energy of the hadrons, electrons and photons created in the collision. Layers of metal with liquid argon in-between converting incoming high-energy particles into many low-energy particles. The absorption causes the liquid argon to ionise, resulting in an electric current being sent.

The Hadronic Calorimeter absorbs the remaining particles that interact through the strong force, which where not deposited in the argon calorimeter. The layers of steel here convert hadron to photons, which may then be absorbed and generate a resulting electrical current. We note that this detector is less precise than the EM calorimeter [25].

Muon Spectrometer

The outermost section of the ATLAS detector is used to measure the momentum of muons which escape all the preceeding detector layers. Muons can interact with electrons of the gas mixture in the Monitored Drift Tube (MDTs). These aluminium tubes then send an electrical signal from the wire at the center of the tube [26].

Trigger System

The trigger system is an important component of data recording. When collisions occur, the trigger system helps the bottlenecking situation by only recording interesting events and writing them to disk. The LHC collision rate is 40 MHz, which is not feasible to store all data produced.

It is split into two stages; the Level 1 hardware trigger works on information from the calorimeter and muon detectors and decision are made in microseconds as to what events are stored and passed. For example, the Level 1 trigger reduces the bunch crossing rate to approximately 100 kHz. Level 2 High-Level trigger utilizes a variety of custom processors to survey specific regions of information and analyse the events in great detail for anything exotic. It is then passed on to offline data storage units. This trigger system provides a filtration process needed to reduce the millions of events passing through the detectors each second to around 200 events per seconds where they can be stored offline and studied for new physics [27].

2.3 Collider Physics

In this section, we briefly overview the physics considerations which dominate particle collision, considering we have described the basic set-up of a particle collider. As we explain in the previous section, the LHC accelerates protons to high energies into two separate beams running clockwise and anti-clockwise, after which we collide the beams inside the ATLAS detector.

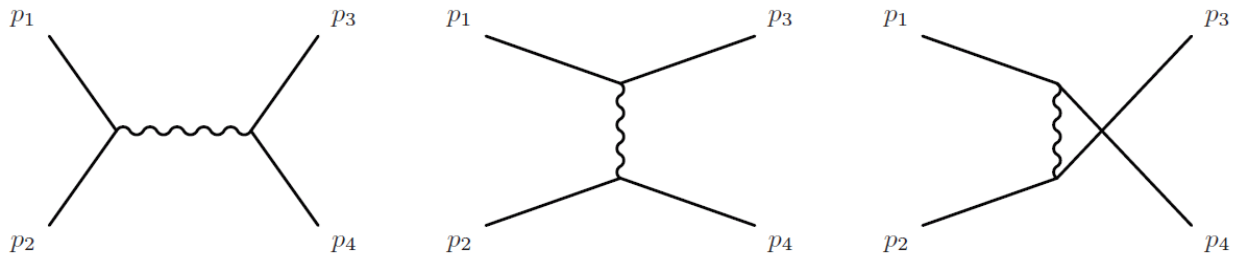


Figure 2.3: Feynman diagrams displaying the three $(2 \to 2)$ scattering processes corresponding to the Mandelstam variables, where there are two incoming particles, p_1 and p_2, as well as two outgoing particles, p_3 and p_4. From left to right, the Feynman diagrams show the s-channel, t-channel and u-channel processes.

2.3.1 Mandelstam Variables

Consider Figure 2.3 that shows the Feynman diagrams of the following three possible scattering events. Let us consider the simple interaction of four particles, where we have two incoming particles with the four-vector, p_1 and p_2, as well as two outgoing particles' four-vector, p_3 and p_4. In the simplified models, we may define this two body scattering $(2 \to 2)$ as three distinctive processes which may take place in a particle collider. In the first case, an s-channel (space) arrangement involves SM quarks which may collide to produce a candidate BSM particle (χ) through the mediator, a Z' boson. We can see this as the particles p_1 and p_2 merging into a virtual particle which immediately decays into p_3 and p_4. In the second scattering process, the t-channel (time) describes the process whereby a SM quark, p_1 possibly decays into p_3, with the mediator (ϕ) now interacting with a second incoming particle, p_2. As p_2 interacts with the emission of p_1, it changes into p_4. Thus, we can look for evidence of the DM particle (χ) and its mediator, ϕ. For the final case of the u-channel, the roles of quarks p_1 and p_2 are simply interchanged from the t-channel process.

Using energy-conservation laws, the Mandelstam variables, are defined as,

$$s = (p_1 + p_2)^2 = (p_3 + p_4)^2, \tag{2.1}$$

$$t = (p_1 - p_3)^2 = (p_4 - p_2)^2, \tag{2.2}$$

$$u = (p_1 - p_4)^2 = (p_3 - p_2)^2. \tag{2.3}$$

In the center-of-mass frame where $p_1 + p_2 = p_3 + p_4 = 0$, $\sqrt{s}$ is the total energy of the colliding particles, whereas t denotes the total squared momentum transferred from p_1 to p_2. It is from these quantities we are able to calculate the scattering angle. We can show that,

$$s + t + u = m_1 + m_2 + m_3 + m_4, \tag{2.4}$$

where m_i is the mass of the i^{th} particle. It is important to point out that only the s-channel provides events with a resonance signal that makes it possible for us to discover the new particle generated. This is because s represents the square of the center-of-mass energy, and so the energy transfer is at a maximum and we are most likely to see an excess due to an interesting process we are studying. In other words, the s-channel represents the highest energy transfer possible, which makes it the most likely for us to discover new heavy particle which undergoes this process. On other hand, the t-channel has no such resonance signal to look for in our invariant mass distributions.

2.3.2 Luminosity

The LHC ring accelerate the proton beams in bunches. Under standard procedures, each beam contains 2808 bunches, and each bunch contains 10^{11}. In order to understand these bunch crossings, it is worth considering the simple case where two protons collide. At the center of mass reference frame, the particles A and B — with negligible masses compared to their energies — have the respective four-vectors,

$$\vec{p}_A = \begin{pmatrix} E \\ 0 \\ 0 \\ E \end{pmatrix} \quad , \quad \vec{p}_B = \begin{pmatrix} E \\ 0 \\ 0 \\ -E \end{pmatrix}. \tag{2.5}$$

Thus, the total energy of the system is the center-of-mass energy which will be $E_{cm} = 2E$, and so we can define the Mandelstam variable $s = (p_A + p_B)^2 = E_{cm}^2 = 4E^2$.

For actual particle colliders, we send many particles to collide head-on in the particle beam. We consider now the beams, A and B which contain numerous particles to directly collide. The probable number of collisions the particles that beam A can make, should be the effective cross-section σ, of an individual particle in the full beam as seen by the colliding particles in beam B, divided by the cross-sectional area, A_A of the actual beam. Clearly, this value should be multiplied by the actual number of particles available, N_A in order to retrieve the total probability of collisions, P_{col},

$$P_{col} = N_A \frac{\sigma}{A_A}. \tag{2.6}$$

Thus, the total number of collisions events, N_{col} should be equal to the number of particles present in beam B multiplied by the probability that these particles collide with the particles in beam A. Thus,

$$N_{col} = N_B P_{col} = \sigma \frac{N_A N_B}{A_A}. \tag{2.7}$$

We expect this result, as when two particle beams collide head-on, the number of collisions is directly proportional the number of particles present in both beams, and inversely proportional to the cross-sectional area, A_A of the beams. The constant of proportionality here is the scattering cross-section σ, which is the effective size of a single particle of beam A.

We define the number of collision per second, R and the respective frequency of collisions, f,

$$R = \sigma \frac{N_A N_B}{A_A} f = \sigma L, \tag{2.8}$$

where $L = \frac{N_A N_B}{A_A} f$ is the instantaneous luminosity. We can relate the luminosity to the number of collisions by integration Equation (2.8) over time,

$$L_{int} = \int L dt = \int \frac{R}{\sigma} dt = \int \frac{1}{\sigma} \frac{dN_{col}}{dt} dt. \tag{2.9}$$

So, L_{int} is an effective measure of the number of collision event between two beams, which can be used to discern the probability of a particular physics process [28].

2.3.3 Collider Coordinates

The surrounding detector is cylindrical in shape, so we must define a coordinate system which exploits this geometry and the relativistic effects of the particle collision. A diagram of the coordinate system described is depicted in Figure 2.4. From the beam axis, commonly denoted as z, we define

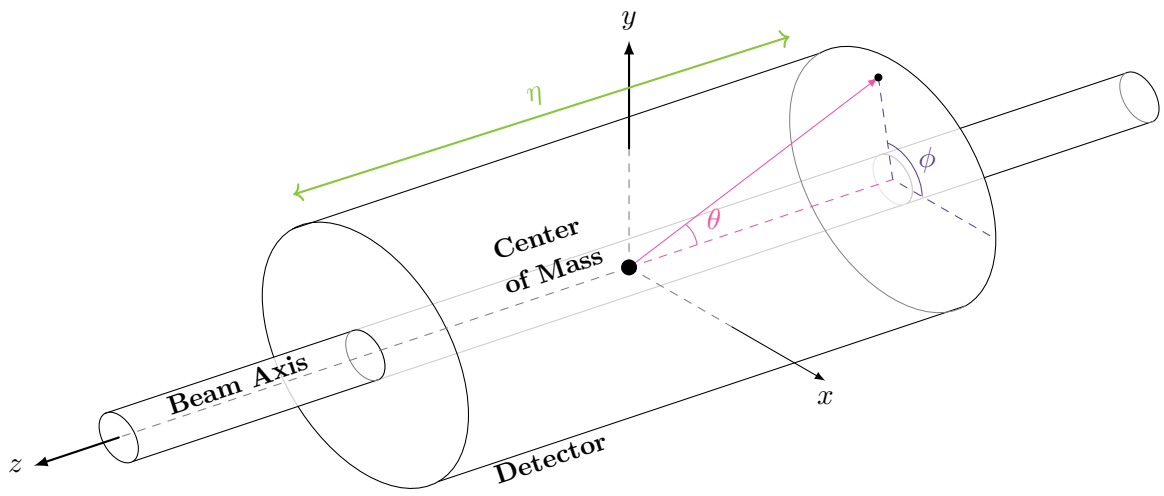


Figure 2.4: Diagram of the detector coordinate system. The Cartesian system (x-y-z plane) is imposed on the detector, with the center of mass point as the origin for both systems. It is critical to note that y is the vertical Cartesian coordinate, and not the rapidity. The z is defined along the beam axis, where we expect to send bunches of protons through for collision within the detector. The azimuthal angle ϕ (purple), wraps around the beam axis on the x-y plane as it would in a cylindrical system. The polar angle θ (pink), rotates about the x-z plane, and is used to construct the psuedorapidity variable, η (green), which runs across the length of the detector.

the azimuthal angle, ϕ which wraps around the z-axis as it would in cylindrical coordinates. This angle is always perpendicular to the transverse momentum, p_T, which measures the momentum of the particle in the radial direction of a cylindrical coordinate system. Thus, a particle created from the collision point with only transverse momentum, will move radially outward of the beam axis, passing through every layer of the detector. If we imagine an x-y plane, situated perpendicular to the z axis as in standard Cartesian coordinates, it is clear that the component of momentum along this plane can be calculated as,

$$p_x = p_T \cos\phi \quad , \quad p_y = p_T \sin\phi \quad , \quad p_T = \sqrt{p_x^2 + p_y^2}. \tag{2.10}$$

In the x-z plane, we can define the polar angel θ similar to how we would in a polar coordinate system. To recall, the azimuthal angle, ϕ is located in the x-y plane. The polar angle is a poor choice in using for relativistic particles in a cylindrical detector, so we may define the $y = 0.5 \ln \frac{E+p_z}{E-p_z}$ as the rapidity. However, the z component of momentum can be difficult to measure, so we define the pseudorapdidity η, as an alternative. For hadron colliders at relativistic speeds, it is important that the psuedorapidity is Lorentz invariant for boosts along the beam axis, z, and so it is defined as,

$$\eta = -\ln \tan \theta/2. \tag{2.11}$$

It is important to note that we define the origin of both angles on the beam axis and the center-of-mass location of the colliding particles. With the definition of psuedorapidity, we can convert back to a Cartesian system by noting Equations (2.10) as well as that $p_z = p_T \sinh\eta$ and $|p| = p_T \cosh\eta$.

Thus, there are four primary kinematic variables used to measure particles in the detector. The energy, E has already been mentioned, and defines the total relativistic energy of the particle.

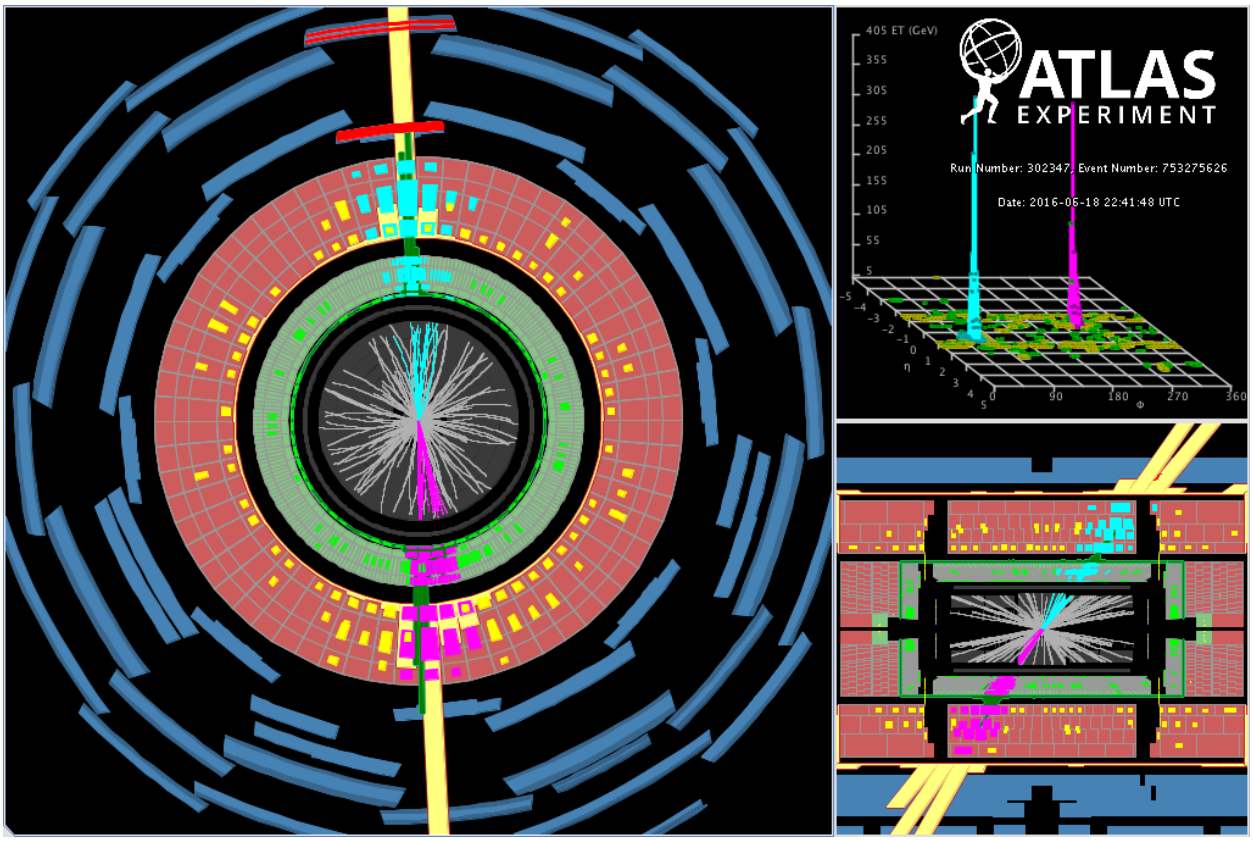


Figure 2.5: Depiction of a dijet event recorded by the ATLAS detector with the dijet invariant mass at 7.5 TeV [29]. The two jet are highlighted in pink and blue colours. From collision, the particles eventually decay into quarks and gluons to form parton showers. Hadronisation occurs, and it is these hadrons which deposit energy into the calorimeter tiles. The left image depicts the view of the detector through the beam axis and the bottom right figure shows a side profile of the detector and collision event. The top right figure displays a three dimensional plot, where the base is the η-ϕ plane of the detector, and the height is the deposited energy of the calorimeter clusters.

The transverse momentum, azimuthal angle and psuedorapidity collectively define a particles four-vector, $[E, p_t, \phi, \eta]$. It is important to remind ourselves that we mainly use this coordinate system instead of a standard Cartesian geometry which represents the four-vector as $[E, p_x, p_y, p_z]$. This is because of the cylindrical nature of the detector, as well as the invariance with respect to the Lorentz boost alongside the z axis, as we are dealing with particles close to the speed of light.

We can demonstrate that defining the four-momenta this way has clear benefits. For example, as we are colliding particle along the beam axis, z, the total transverse momentum of the system must be zero, $p_T = 0$. Not all detectors layers will manage to record the transverse momentum of particles moving through. Thus, a useful parameter known as the missing transverse energy (MET), $\not{E_T}$ is defined as the total transverse momentum not recording by the detectors, or in other words, missing.

2.4 Jets

Collision events can produce many particles which decay before ever reaching the detector. A single particle may produce many decay products, which in turn may decay further, eventually producing a shower of quarks and gluons before reaching the detector. We reiterate that, collectively, quarks and gluons are referred to as partons. The result of this decay chain are collections of high energy quarks and gluons, which are not directly observed in the final state. These collimated bunches, or parton showers, form hadrons through the process of hadronisation before reaching the detector.

This poses the issue — how do we study a new particle created if we only have the readings of its decay products? Because of this phenomena, it is useful to define an object of an event called a jet. A jet is a cluster of hadrons, which have been generated by the decay of a single parton, such as a gluon or quark in a typical collision, which may eventually interact with collider detectors. These hadrons are the actual particles which excite the calorimeter tile detectors and where we may see in which location of the detector geometry that the hadrons came from. By studying the resultant jet objects generated by new physics, we hope to understand what initial physics process happened in the event.

Given the set of hadrons read by the detector, p_i^{had} we seek to cluster these particles into jets, p_j^{jet}. The mass of the system of particles is found by the four-momentum calculation,

$$m^{jet} = \sqrt{\sum E_i^{had} - \sum |\vec{p}_i^{had}|}, \tag{2.12}$$

where m^{jet} is the mass of the jet, and E_i^{had} and $|\vec{p}_i^{had}|$ are the energy and magnitude of momentum of the i^{th} hadron, respectively.

The process of constructing jets is a clustering problem, and so there are many algorithms built for jet construction. A distance measure is needed, and thus we should naturally make use the angular distance, R between two particles in the detector geometry,

$$R = \sqrt{(y_1 - y_2)^2 + (\phi_1 - \phi_2)^2}, \tag{2.13}$$

where y and ϕ are the rapidity and azimuthal angle respectively. The algorithms for clustering hadrons into jets in the LHC in particular, can be handled by considering the processing of one object decaying into two. We take the inverse approach, by continuously merging two particles into a jet if they meet a certain distance criteria selected by the chosen algorithm. In this way, we group the hadrons which deposited energy into the calorimeter tiles in order to create the jet which originates back from a single parton. An alternative to the calorimeter energy readings is to utilise particle flow objects to construct jets. For example, the benefit of methods using inputs like charged particle tracks is to generate finer angular resolutions. Figure 2.5 depicts this process for a dijet event, as read by the ATLAS detector.

In this thesis, we utilise the anti-k_t jet algorithm [30] implemented in FASTJET [31]. Its defining feature is clustering all objects nearby to the largest p_T object, within the specified angular radius, R_0. This results in jets with a circular shape.

2.4.1 Jet Substructure Observables

A standard jet can be clustered with radius $R = 0.4 - 0.6$. However, let us consider a heavy particle which decays into two daughter particles. The angular separation between these two decay products can be quite large, so we increase to the jet radius to $R \sim 1$, which allows us to encapsulate all the decay products of the original heavy particle to a single jet. These jets, which we call large-radius

Name	Variable
Angularity	λ_κ^β
N-subjettiness	τ_i, τ_{32}, τ_{21}
Energy Correlation Functions	$e_N^{(\beta)}$, C-family, D-family
Number of Charged Particle tracks	n_{trk}
Fox-Wolfram Moment	R^{FW}
Splitting Scale Distance Measure	$\sqrt{d_{32}}$, $\sqrt{d_{21}}$
ΔR k_t Algorithm	KtDR
Aplanarity	A
Planar Flow	Pf

Table 2.1: Summary of useful Jet substructure variables.

jets, can therefore be understood as a proxy for the original particle in question. It is because these jets are so large, that their composition has interesting characteristics which can help us tag a large-radius jet originating from, for example, a top quark. These attributes are aptly named the jet's substructure, as they describe the internal structure of the jet. In order to derive these important characteristics of a jet, we look closely at the constituents which compose the large-radius jet.

The study of jet substructure provides us with tools necessary to differentiate between large-radius jets which were created by an underlying interesting physics process (like jets formed from top quarks), against background jets formed by many multijet processes which created quarks and gluons within an event. The following jet substructure variables investigate internal structure of a jet, or jet shape which may help us to tag jets formed by a variety of processes and there has been a growing body of research dedicated to exploring these variables in the machine learning field. It is these variables which are used in this thesis.

Generalised Angularity

This discriminant variable, the generalised angularity, λ_κ^β of a jet, is defined by a two parameters; the angular exponent, $\beta \geq 0$ and the weighting factor, κ [32]. So,

$$\lambda_\kappa^\beta = \sum_{i \in jet} z_i^\kappa \left(\frac{\Delta R_i}{R_0} \right)^\beta, \tag{2.14}$$

where R_0 is the jet radius. The momentum fraction, z_i and distance of the jet constituent to the jet axis, ΔR_i are defined as follows,

$$z_i = \frac{p_{T,i}}{\sum_{i \in jet} p_{T,j}} \qquad , \qquad \Delta R_i = \sqrt{(\eta_i - \eta_{jet})^2 + (\phi_i - \phi_{jet})^2}. \tag{2.15}$$

It is clear in the case where $\beta = 0$ and $\kappa = 0$ that we reduce to the multiplicity of jet constituent particles that make up a jet. For a variety of parameter choices, the angularity can be a strong analogous to a jet shape's properties, such as mass. For example, a measure of squaring the transverse momentum and then summation, p_T^D is simply the case when $\beta = 2$, $\kappa = 0.5$, multiplied by a factor. For $\beta = 1$ and $\kappa = 0.5$, the variable is termed Les Houches angularity (LHA). In the generalised case, we can see that jets with more radiation should have larger generalised angularities. Combined with the utilisation of the distance to the jet axis, angularity can be seen as the measure

of radiation around the jet axis. We note that jets formed by gluons have a larger angularity than quark-initiated jets, thereby making this observable a useful discriminant for such events [33].

N-subjettiness

The number N of subjets within a jet are used to construct this substructure variable, N-Subjettiness, τ_N^β. It is defined by one free parameter, β, but it most cases, including this thesis, we set $\beta = 1$. It is calculated as follows,

$$\tau_N^\beta = \frac{1}{d_0} \sum_{i \in jet} p_{T,i} min\{\Delta R_{i,a_1}^\beta, \ldots, R_{i,a_N}^\beta\}, \tag{2.16}$$

where d_0 is a normalisation factor. The set of axes $\{a_1, \ldots, a_N\}$ are the subjets that can be formed by a variety of mechanisms. In this thesis, we focus forming the axes using the WTA k_t algorithm. We take a single jet, and recluster the constituents of the jet, using the k_t according the *winner-takes-all* (WTA) recombination scheme. This is simply a jet clustering algorithm where we must specify the number of jet we would like, rather than specifying the jet radius, R_0. Thus, in Equation (2.16), we multiply the transverse momentum of the jet constituents composing the jet with the distance between that jet constituent and its nearest subjet, ΔR. This means that N-Subjettiness gives a measure of how closely constituents are aligned to subjet axes, and thus how energy is spread to these axes.

Taking the N-subjettiness ratio,

$$\tau_{N,N-1}^{(\beta)} = \frac{\tau_N^\beta}{\tau_{N-1}^\beta}, \tag{2.17}$$

provides a good discriminator between N-prong signal jets against multijet background [34] because the variable identifies N-prong substructure.

Energy Correlation Functions

Energy Correlation Functions (ECF's) effectively provide the similar information to N-subjettiness without the need for utilising reclustering for subjets and the specification of particular axes. They are defined by the parameters, β and N, and are calculated as follows,

$$\begin{aligned}
e_1^{(\beta)} &= \sum_{i<j \in jet} z_i \\
e_2^{(\beta)} &= \sum_{i<j \in jet} z_i z_j \Delta R_{ij}^\beta \\
e_3^{(\beta)} &= \sum_{i<j<k \in jet} z_i z_j z_k \Delta R_{ij}^\beta R_{jk}^\beta R_{ik}^\beta \\
&\vdots \\
e_N^{(\beta)} &= \sum_{i<\cdots<N \in jet} \left(\prod_j^N z_{i_j} \right) \left(\prod_{k<l=1}^N \Delta R_{i_k i_l}^\beta \right),
\end{aligned}$$

where z_i is the momentum fraction as mentioned previously in Equation (2.15). Similar to N-subjettiness, taking the ratio of these quantities for different choices of N give a strong discriminator

for massive boosted particles. More specifically, we can define the C and D families,

$$C_N = \frac{e^{\beta}_{N+1} e^{\beta}_{N-1}}{\left(e^{\beta}_{N}\right)^2} \qquad , \qquad D_N = \frac{e^{\beta}_{N+1}\left(e^{\beta}_{N-1}\right)^3}{\left(e^{\beta}_{N}\right)^3}, \tag{2.18}$$

for specific values of the parameter β [35]. In this thesis, it is important to state to specific variables that are useful in discriminating two-prong structures within jets. That is,

$$C_2 = \frac{e_3}{(e_2)^2} \qquad , \qquad D_2 = \frac{e_3}{(e_2)^3}. \tag{2.19}$$

It is interesting to note that ECF's have been incorporated into Energy Flow polynomials, as they construct a linear basis for all jet substructure variables which are IR and ICR safe. This leads it to have large applicability for machine-learning tools [36].

KtDR

KtDR is simply the angular distance, ΔR produced in the psuedorapidity-azimuthal, η-ϕ plane, when the final clustering step is proceeded to in the k_t algorithm. So, similar to Equation (2.15),

$$\Delta R_i = \sqrt{(\eta_i - \eta_j)^2 + (\phi_i - \phi_j)^2}. \tag{2.20}$$

For larger energies, quarks will be more closely collimated and thus the distance between jet constituents that compose a jet will be shorter. Thus, KtDR is a strong discriminator for low p_T jets [37].

Fox-Wolfram Moment

For our applications, the Fox-Wolfram moment is defined as,

$$H_l = \sum_{i,j} \frac{|\vec{p_i}||\vec{p_j}|}{E^2} P_l\left(\cos\Omega_{ij}\right), \tag{2.21}$$

where P_l is the Legendre polynomial function, E is the total energy of all clusters of the jet, and where the argument is the distance measure, $\cos\Omega_{ij} = \cos\theta_i\cos\theta_j - \sin\theta_i\sin\theta_j\cos\left(\phi_i - \phi_j\right)$ between the $i-$th and $j-$th jet constituent of a jet.

Usually, Legendre polynomials have a weighting, and so there are many choices for the weighting factor in Equation (2.21), such as utilizing the transverse momentum instead of the total. The resultant quantity expresses a superposition of spherical harmonics, which can be effective at describing the geometry of a jet [38]. The ratio between the 2nd and 0th order moments, $R^{FW}_{20} = \frac{H_2}{H_0}$ tend to be good discriminators.

Splitting Scales

When utilizing the inclusive k_t algorithm, we compute the following distance measure between the i^{th} and j^{th} jet constituents, which has been stated in a similar way,

$$d_{ij} = min\{p^2_{Ti},\, p^2_{Tj}\}\frac{\Delta R^2_{ij}}{R_0}, \tag{2.22}$$

where ΔR has been defined in Equation (2.20). This is used to define the splitting scale, $\sqrt{d_{ij}}$, which records the distances between jet constituents at each step of the clustering process. For example, in the case of merging three jet constituents with moment, p_1, p_2, p_3 to two $(3 \to 2)$, we store the value between jet constituent distance between p_3 and p_2, which leads to $\sqrt{d_{32}}$. Similarly, in the merging process $(2 \to 1)$ of the k_t algorithm, we retrieve $\sqrt{d_{21}}$. Since the merging algorithm is built to undo the branching of particles before reaching the detector, $\sqrt{d_{ij}}$ gives a sense of how 'hard' the splitting was [39].

Planar Flow

The shape tensor of an event, I_w is defined as,

$$I_w^{kl} = \sum_i E_i \frac{p_{Tik}}{E_i} \frac{p_{Til}}{E_i}, \tag{2.23}$$

where E_i is the energy of the i^{th} constituent of the jet. By taking the determinant, $det(I_w)$ and trace, $tr(I_w)$ of this tensor and simplifying using the eigenvalues, λ of I_w, we can derive the Planar Flow variable, Pf which measures how the movement of energy flow inside the jet is linear in direction. It is defined in [40] as,

$$Pf = \frac{4det(I_w)}{tr(I_w)^2} = \frac{4\lambda_1\lambda_2}{(\lambda_1 + \lambda_2)^2}. \tag{2.24}$$

Chapter 3

Dark Matter Searches

3.1 Review in general Dark Matter theory

We have described the mechanisms of the ATLAS detector, as well as the production of jets and corresponding substructure so that we may now detail how we investigate new physics that proton-proton colliders like the LHC were built for.

The Standard Model (SM) of particle physics describes the observable phenomena of the microscopic world, built on the framework of Quantum Field Theory (QFT). Beginning with Mills and Yang in 1954, particle physics and the Standard Model have been continuously built up to describing three fundamental forces (electromagnetic, strong and weak nuclear forces) as well as elementary particles such as generations of leptons and quarks. At the same time, massive experimental projects such as the Large Hadron Collider have been conducted in order to guide and verify our theories. Indeed, experimental confirmation has been extremely successful, such as the discovery of the W and Z bosons in 1983 [41] [42].

SM physics still adequately describes most physical phenomena, as mentioned in Section 2.1. Therefore, we can say that the Standard Model is valid up to a certain energy scale, beyond which we expect to see evidence of Beyond the Standard Model (BSM) physics. This is the reason for upgrades made to the LHC, as we can more readily probe deviations from the Standard Model at higher energies.

A major component missing from the Standard Model is the theoretical predictions to accommodate the indirect evidence of exotic particles which give rise to Dark Matter (DM). The problem first arose as early as 1933, when it was observed that the visible mass of galaxy clusters was too small to account for the gravitationally induced velocity of constituent galaxies [43] that was also observed. Thus, it was postulated that some matter was unseen. Over the following decades, further motivation for the existence of this non-baryonic matter was found, such as explanations of the cosmic microwave background (CMB) favouring the existence of DM [44] [45]. As of 2012 with the discovery of the Higgs Boson [46], the experimental support of the SM is complete — however — it lacks an explanation of DM, and thus new Beyond the Standard Model physics (BSM) is required. It is essential to point out that despite astrophysical observations, evidence of dark matter particles at the quantum scale has not been observed. In other words, particle colliders like the LHC, as well as direct detection experiments [47], are still continuously searching for this elusive matter.

Research over the past decades has focused on explanations of dark matter and the proposed properties such type of matter has. The earliest proponents suggested a type of particles which interacts strongly with the gravitational force in order to explain the astrophysical phenomena mentioned, but also weakly interact with ordinary Standard Model particles. One such particle

which was hoped to be a candidate for this elusive matter were neutrinos, which up until the 1980's, were thought to be massless [48]. However, it was found that the do have mass. A universe with neutrinos as the dark matter candidate, would look very different to the universe we observe. More so, we observe that dwarf irregular galaxies have a large dark matter content, where we also know that neutrinos cannot be dark matter particles in these systems. This would require another type of dark matter to account for our observation in addition to neutrinos [49].

Another prominent theory to provide a model of dark matter particles is that of supersymmetry (SUSY). SUSY proposes an extension to the current theory of the Standard Model in order to fill in gaps in our experimental observations. This is done by introducing extra supersymmetric particles, which interact with SM forces but contain different masses. In the Minimal Supersymmetry Standard Model (MSSM), the Standard Model is extended to accommodate a super symmetric particle of all fermions and bosons with minimum effort [50]. So, these partners of ordinary matter differ by a spin half. Thus, all fermions have a superpartner bosons, and vice versa for ordinary bosons. We denote these MSSM particles by adding a tilde ($\sim$) and prefixing the SM particle names with an 's'. Thus, quarks and leptons become squarks and sleptons respectively, but for the photon, W and Z boson, it is required to proposed extra particles other than the superpartners, in order to complete the theory. That is important is that the result of the model is the existence of weakly interacting massive particles (WIMPs), which provides an explanation on the lack of visible signature in our detectors [51]. Potential candidates for dark matter, WIMPs weakly interact with ordinary SM matter, and so is thus difficult to read off in our detectors. Examples of possible candidate particles include the neutrally charged four neutralinos ($\tilde{N}_1, \tilde{N}_2, \tilde{N}_3, \tilde{N}_4$) [52], as well as the sneutrino, the supersymmetry partner of the electron, muon, and tau neutrinos [53].

3.1.1 Mono-X Signatures

As mentioned previously, we may calculate the missing transverse energy, $\not{E}_T$ by considering energy-momentum conservation laws and noting that the total transverse momentum of the system sums to zero, as the particles collide on the beam axis. It is important to state that neutrinos particles do not interact with detectors and thus also contributes to the value of $\not{E}_T$. Possible candidates for dark matter created in collision will not interact with detectors, and thus we expect an increase in the amount of transverse energy missing. At this point, we should note that if the final state of an event only produced DM particles, we would read no transverse momentum and the event would not be recorded. However, in most collisions, we produce some ordinary matter and a dark matter candidate, where the DM particle contributes to the MET.

We refer to this type of searching for BSM physics, as Mono-X signatures. In this case, X is a Standard Model object, such as a jet, photon, top quark, or Higgs, Z or W boson, which is produced in conjunction with a dark matter candidate which depends on BSM model. In searching for Mono-jet signatures, we focus on jet + $\not{E}_T$ events, which are events which contain a significantly large $\not{E}_T$ — which cannot only be attributed to neutrinos — as well as at least one jet from which we expect the recoil of the event to be derived from. Figure 3.1 depicts an idealised diagram of what such events look like inside the ATLAS detector, without the effect of neutrinos, ν leaving the detector. We emphasize this point as we know that neutrinos leave the detector undetected, and therefore always contribute to the missing energy, $\not{E}_T$.

3.1.2 Effective Field Theories and Simplified Models

Much current DM research has focused on heavy particles that have little interaction with baryonic matter (WIMPs) [54] [55] [56]. Such matter will not be picked up by detectors at the LHC, and thus

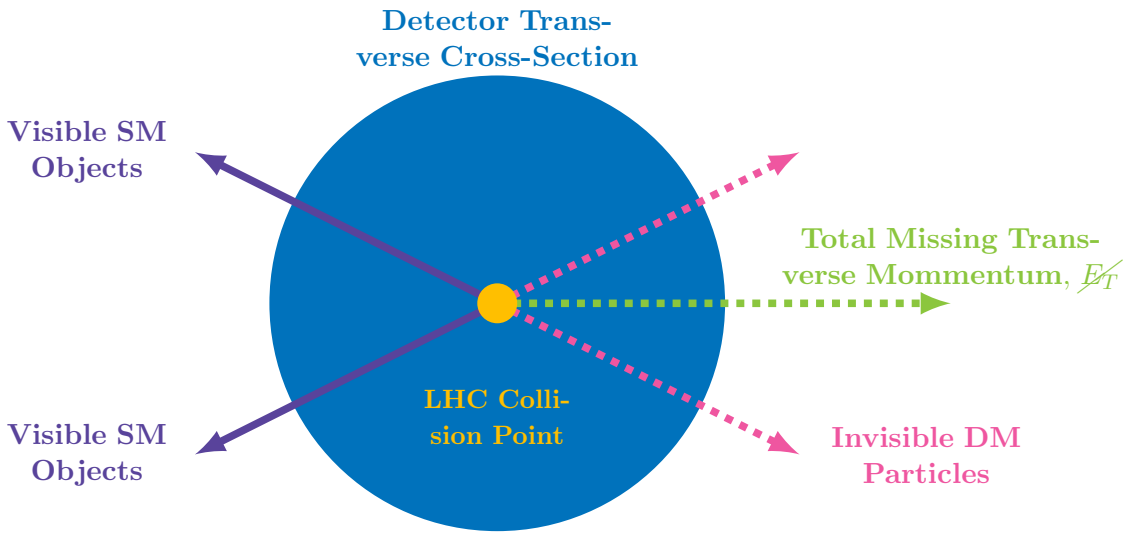


Figure 3.1: Idealised diagram displaying how missing transverse momentum, $\not{E}_T$ occurs within the LHC detector without the effect of neutrinos escaping, from the point of view of the beam axis. Once collision occurs, both Standard Model and dark matter particles are created. The visible SM objects are read by the detector (purple), while DM hadrons escape undetected (pink). We can infer the four-vector of these invisible hadrons by considering the reading we get from the visible particles and using conservation laws. The addition of these four-momenta give the total missing energy of the system (green). As neutrinos are created and also pass through undetected by the detector, they will also contribute to this missing energy, $\not{E}_T$.

we look for missing transverse momentum (MET), $\not{E}_T$ by the use of conservation of momentum in each initial collision. However, for any more complicated theory proposed, the collider signature may require vastly more complex analyses in order to find experimental evidence. Many theories have proposed that indeed DM does interact with usual observable matter. In the 'dark sector' which contain a new set of dark particles and forces governing them, link with Standard Model particles via portal interactions.

Effective field theories (EFT's) describe the framework of approximating complicated physics theories. Low energy interactions of dark matter are thus explained by dark matter effective field theories (DMEFT's), where the interaction between SM and DM is a contact interaction which takes many forms. We expect this description sufficiently accounts for minor details that may occur for high-energy collisions which take place within the LHC, where complex interactions are integrated out of the system. EFT's in this situation are flexible in the sense that it can accommodate a wide range of physical interactions proposed, as it doesn't require a specific description of the mediator mechanism between ordinary matter and that of dark matter.

Simplified models propose simple descriptions of the physics occurring at the point of collision, where we do not concern ourselves with the additional physical processes at higher energy scales than that of the collision energies. These models can describe tree-level interactions, such as pair production of invisible dark matter particles, because of the fact that we only expect a few number of these particles to be created in these interactions. We use these simplified models as a bridge between basic EFT's and more developed complete physics theories [57].

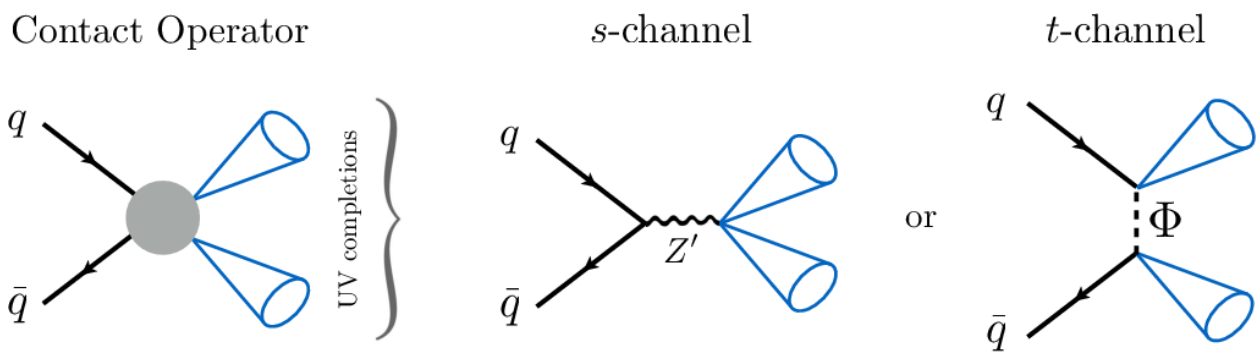


Figure 3.2: Feynman diagrams illustrating the two channels (s and t) of dark hadron production. The left image shows a general contact operation involving q and $\bar{q}$ producing dark hadrons, which in turn form semi-visible jets in the detector. The middle image displays how these semi-visible jets in the s-channel where Z' bosons are the mediators. The right hand side diagram shows how semi-visible jets are produced in the t-channel via the mediator, Φ [58].

3.2 Dark and Semi-visible Jets

The focus of this thesis is that of a search involving dark and semi-visible jets. The current body of research into dark matter and the corresponding searches at the LHC have, so far, not led to a discovery of a dark matter particle. Considering the majority of these searches focused primarily on WIMPs, there is a growing set of phenomenological papers that are investigating alternatives. One such area has proposed accessing the dark sector in the absence of WIMP particles via unusual collider topologies which have not been explored in detail yet.

One such unique proposition is to suggest that the dark sector is strongly coupled to SM particles, instead of the assumption that it weakly interacts. This implies that a large proportion of dark matter particles created in collision event do decay back into Standard Model particles, and thus do not escape the detector. This concept is important as it opens new classes of signatures which have not been explored, as well as providing an explanation on the lack of results in current searches. That is, the dark matter signature would rest in the jets formed by the gluons and quarks that the dark matter particle decays into. If this is the case, these dark jet should contain unique topologies which have not been explored adequately. In summary, dark jets are formed when a heavy particle decays into a pair of dark matter particles, which in turn decay to visible Standard model particles like hadrons, which we read in the detector. As we recall, these hadrons are formed into jets, and thus the final state is deemed 'dark jets' [59].

We can generalise this concept, by describing the range of coupling strength of the dark sector to SM interactions. In the one extreme, strong coupling leads to DM particles created in collision to completely decay into hadrons, which form dark jets. On the other extreme, weak coupling implies that the DM particle never decays back to visible Standard Model particles, and thus escapes undetected. Between these situations, there exists the situation where the DM particles only partially decay back to visible matter, while the rest leave the detector as missing transverse energy. The remaining visible matter still forms jets and, similar to the dark jets, will have unusual topological characteristics compared to ordinary jets generated by SM decay mechanisms. This type of jets is called 'semi-visible' jets, and their production in the s and t-channels is depicted in Figure 3.2.

From the explanation given so far, it is clear that the circumstance of a semi-visible jet should have the jet and $\not{E}_T$ aligned along the same direction. This is because, unlike WIMPs, semi-

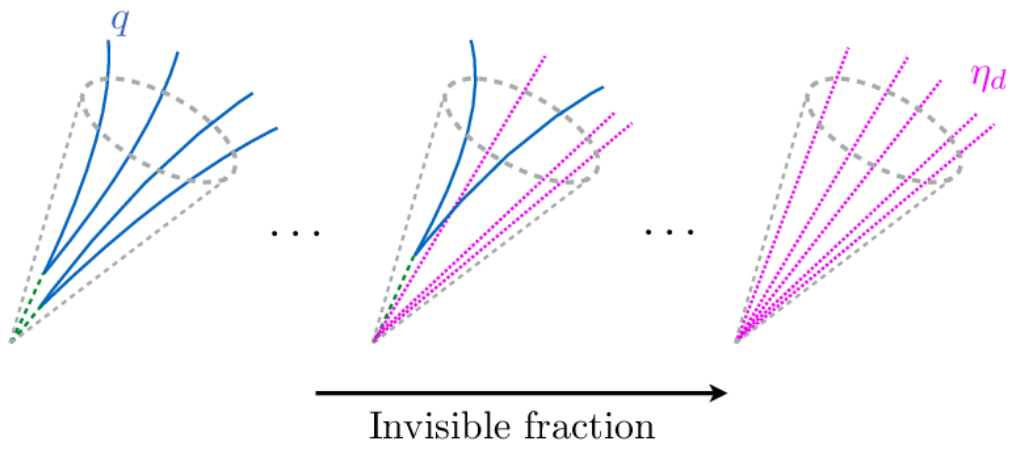


Figure 3.3: Illustration of the structure of a semi-visible jet, which illustrates the structure of a jet when dark hadrons decay. They produce SM quarks, q — but some fraction, η_d of dark hadrons may be stable at the detector, resulting in a combination of a signature object that contains both visible and invisible matter at the detector level. It is worth pointing out that when the jet only contain dark hadrons, the jet is invisible and we expect missing energy, $\not{E_T}$, in this direction as it does not interact with the detector. Green lines indicate the path of dark hadrons which may decay into visible SM particles (blue lines) or the dark hadrons may remain stable (dotted pink lines) [58].

visible jets imply a reasonable amount interaction between dark and regular hadrons. Thus, in the production of multiple jets, one jet will be paired closely with the missing $\not{E_T}$ in the same direction as both DM and SM particles have momentum along the same axis. However, this poses an issue as many $jets + \not{E_T}$ searches require a minimum separation angle, $\Delta\phi$ between any jet and $\not{E_T}$ to reduce multijet background events, and so we require $\Delta\phi > 0.4$ in many cases. The see this when comparing the results of a strongly and weakly coupled models (semi-visible jets and WIMPs respectively) which produce different $\Delta\phi$ and $\not{E_T}$ values. This means that this class of jets have not had dedicated searches in ATLAS, and the type of event which would contain these jets have typically been removed.

To formalize this, [60] begins by defining that some group of dark quarks collect into hadrons below an energy scale, Λ_d. This means a dark quark with transverse momentum, p_T larger than the dark confinement scale ($p_T > \Lambda_d$) will eventually hadronize. These dark hadrons produced are expected to operate similarly to their Standard Model counterparts, creating DM jets. These resultant jets would be invisible to detectors, resulting in MET, as discussed earlier. However, it is possible through portal interactions between the dark sector and SM sector that a fraction of these dark hadron decay back into visible SM states.

We can provide an effective Lagrangian which parametrises BSM effects by capturing all possible interactions,

$$\mathcal{L}_{eff} = \mathcal{L}_{SM} + \Lambda^{-1}\mathcal{L}_1 + \Lambda^{-2}\mathcal{L}_2, \tag{3.1}$$

where $\mathcal{L}_1, \mathcal{L}_2$ represent SM operators which obey $SU(3)_C \times SU(2)_L \times U(1)$ gauge symmetries. These terms have a decreasing effect on the effective Lagrangian, as denoted by the increasing power of the factor Λ.

We must only add dark matter particle production to extend the Standard Model, and so a Dark Matter Effective Field Theory (DMEFT) is utilized. The contact operation between SM

particles and DM hadrons can therefore be represented as,

$$\mathcal{L}_{contact} \supset \left(\frac{c_{ijab}}{\Lambda^2}\right)(\bar{q}_i\gamma^\mu q_j)(\chi_a\gamma_\mu\chi_b)\,. \tag{3.2}$$

As explained in [58], the dark sector is an $SU(2)_D$ gauge theory, with the dark sector coupling defined as $\alpha_d = \frac{g_d^2}{4\pi}$. This has the interaction Lagrangian as,

$$\mathcal{L}_{dark} \supset -\frac{1}{2}\mathrm{tr}\left(G^d_{\mu\nu}G^{d\mu\nu}\right) - \bar{\chi_a}\left(i\not{D} - M_{d,a}\right)\chi_a. \tag{3.3}$$

The intricacies of the dark sector have a minor effect on the final state signature we may encounter in the LHC detectors. By using Simplified Models (explained in Section 3.1.2), which are already widely used for WIMP searches, we are only need to concern ourselves with effects on collider observables, such as the multiplicity of particles produced, the fraction of these particles which are invisible to the detector due to the presence of dark matter, as well as the average p_T of these objects. Thus, three parameters are introduced to the model which account for the variety of signature which we may encounter in the ATLAS detector.

The number of stable and unstable states which are produced by the dark sector model is controlled by the ratio of the rate of stable hadrons to the rate of total hadrons,

$$r_{inv} \equiv \frac{\text{Number of stable hadrons}}{\text{Number of total hadrons}}. \tag{3.4}$$

As $r_{inv} \to 0$, we expect the dark hadrons produced to decay back to visible SM particles. As $r_{inv} \to 1$, no dark matter particles decay to visible matter, and thus escape undetected. This situation is virtually indistinguishable from WIMPs. For semi-visible jets to occur, we look for the intermediate case where r_{inv} is not at either extremes. The other two parameters which control the final state signature are the characteristic dark hadron mass scale M_d, and the strong coupling to the dark sector, α_d.

Figure 3.4 displays the effect that these three parameters (r_{inv}, M_d, α_d) have on the output of the models. What we observe is how these three parameters change the multiplicity of dark hadrons produces. As we have stated, it is clear from the 2-dimensional histograms that the number of dark hadrons, n_{dark} increases with as r_{inv}. Furthermore, the overall number of dark hadrons produced, decreases as we increase the mass scale, M_d. Generally, the number of dark hadrons produced increases as the confinement scale Λ_d decreases. This increase stops when $\Lambda_d \lessapprox M_d$, or the dark quark mass cuts off the dark shower production. This cut-off occurs at $\alpha_d \approx 0.23$ and $\alpha_d \approx 0.45$ at $M_d = 10$ GeV and $M_d = 100$ GeV respectively. For this reason, we can see why there is a peak of dark hadrons in both plots of Figure 3.4.

[61] shows that the effects of dark hadron decay depend much on their spin, mass and mediator the Standard Model sector. Two dark quarks with different flavours, q'_1 and q'_2 can combine to form either the dark pion, π_d or the dark rho meson, ρ_d. The spin-0 dark pions are much lighter than other dark hadrons, and until they decay, will make up a considerable percentage of the formed dark jet. It may decay into a pair of heavy quarks, where the quarks lifetime are closely correlated with the mass of the dark pion, m_{π_d}. Under certain cases, dark pions may also decay into a pair of dark photons, γ'. Dark mesons quickly decay through the process $\rho_d \to \pi_d\pi_d$, except for π^0_d, which decays to visible matter. In totality, this implies all dark mesons, ρ_d (except π^0_d) decay to dark pions, π_d, which if stable form the invisible matter of a semi-visible jet, and π^0_d contributes to the visible material of the jet. The extent of the final state signature thus depends on the flavoured quarks mass, $m_{q'_d}$, the dark meson mass, m_{ρ_d} and the dark pion mass, m_{π_d}.

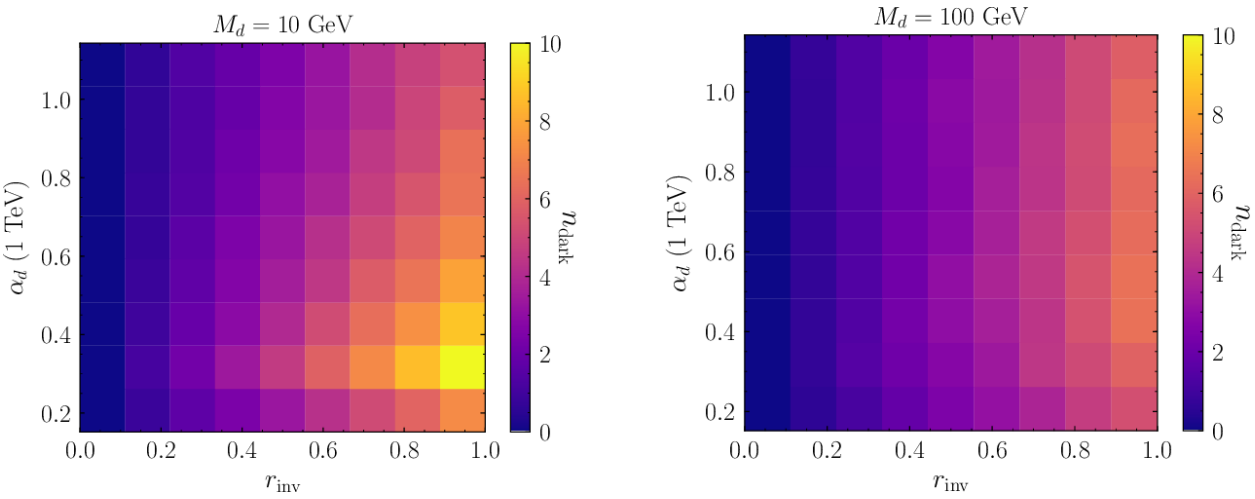


Figure 3.4: 2-dimensional plots showing the effect of the three parameters (r_{inv}, M_d, α_d) on the multiplicity of dark hadrons produced. The left plot with $M_d = 10$ GeV displays in general a larger amount of dark hadrons than the right plot with $M_d = 100$ GeV. We note in both plots that increasing r_{inv} leads to an increased production of dark hadrons, while varying the coupling strength has a minor effect [58].

As we have alluded to, the production of dark and semi-visible jets results in unique topologies as they are the result of dark hadron decay into SM particles that forms jets. Since these signatures look similar to jets produces by ordinary physics processes, it is distinctive challenge to differentiate dark and semi-visible jets from multijet background. However, [62] has been shown that the use of jet substructure can aide in searches for semi-visible jets. We also note that it may very well be that these jets occur very rarely, and thus there are a small number of signal events.

3.3 Simulation

Monte Carlo event generators are used to simulate the various hard-scattering processes that occur for high-energy proton-proton collisions in the LHC. The role of MC event generators is to generate events based on some underlying Standard Model processes that particles undergo before reaching the detector in their final state. Simulation is composed of two parts. First, we start with $2 \to 2$ hard scattering processes given by matrix element calculations up to some given order. Second, the $2 \to N$ final state processes is realised by the phenomenologically driven parton shower approach.

3.3.1 Semi-visible Jet Event Generation

Signal samples generation of semi-visible jets is made with *MADGRAPH* [63] event generator at leading order, and showered with *PYTHIA8* [64]. As described in Section 3.2, semi-visible jets have three important parameters which gives rise to their final state signature. We consider samples generated with set parameters, the ratio of stable hadrons $r_{inv} = 0.6$, mediator mass $M_\phi = 1500$ GeV and dark mass hadron, $M_D = 10$ GeV. The *Hidden Valley* module [65] of *PYTHIA8* is used to simulate and study the effects of Beyond the Standard Model physics, and provides a generalised framework to simulate BSM physics, such as that which generate dark and semi-visible jets. Parameters specific to *Pythia* Hidden Valley module are set to $\lambda = 1$ and $\alpha_O = 1$ in order to run α_ν, and $\alpha_{FSR} = 1$ represents the constant coupling strength. The background process samples

which are generated represent multijet background.

Reconstruction

Jets are constructed using the anti-k_t algorithm with a specified radius of $R = 0.4$. We take these constructed jets, and use them as the inputs to recluster them into jets with radius, $R = 1.5$. The reclustering process is needed as this represent the final state topology better. This is because it can be shown that from jet images on the $\eta\phi$ plane that jets with radii larger than $R = 1.0$ optimally encloses energy deposits of a semi-visible jet. However, large jets encapsulate the dark hadron produced. Thus, a reclusering approach is chosen for the semi-visible jet model.

We state the two important object selections for each event in order to appropriately analyse semi-visible jet signatures. That is, we only consider events with $\not{E}_T > 300$ GeV and the reclustered $R = 1.5$ jets where the leading jet $p_T > 300$ GeV and the remaining jets containing $p_T > 30$ GeV and jet mass, $m_j > 40$ GeV.

Distributions of Jet Substructure Observables

The following figures display the jet substructure distributions for semi-visible jet with parameters which we have detailed above, after object selections. Figure 3.5 depicts these substructure observables for the s-channel process of SVJ, while Figure 3.6 shows the t-channel distributions.

The final weighting is calculated as follows. Each individual event weight is multiplied by the luminosity of the data sample, as well as the corresponding filter efficiency and cross-section of the JX sample it belongs to. The JX samples and relevant metadata are retrieved from the ATLAS Metadata Interface (AMI). This value is then divided by the total number of events in the JX sample. The equation below displays how this final weighting is calculated,

$$\text{Final Weighting} = w_0 \frac{XS \cdot FE \cdot L}{\text{JX No. Events}}, \tag{3.5}$$

where w_0 is the event weight, XS is the JX cross-section, FE is the JX filter efficiency, and L is the luminosity, which is equal to $139\,\text{fb}^{-1}$. Then, the signal and background distributions are normalized by their cross-section.

We may highlight that these distributions indicate the low amounts of signal present, relative to the multijet background. We see that both signal and background distribution occupy similar ranges in these jet substructure observables, with no obvious cuts to make which separate the samples into bins which increase our proportion of signal to background.

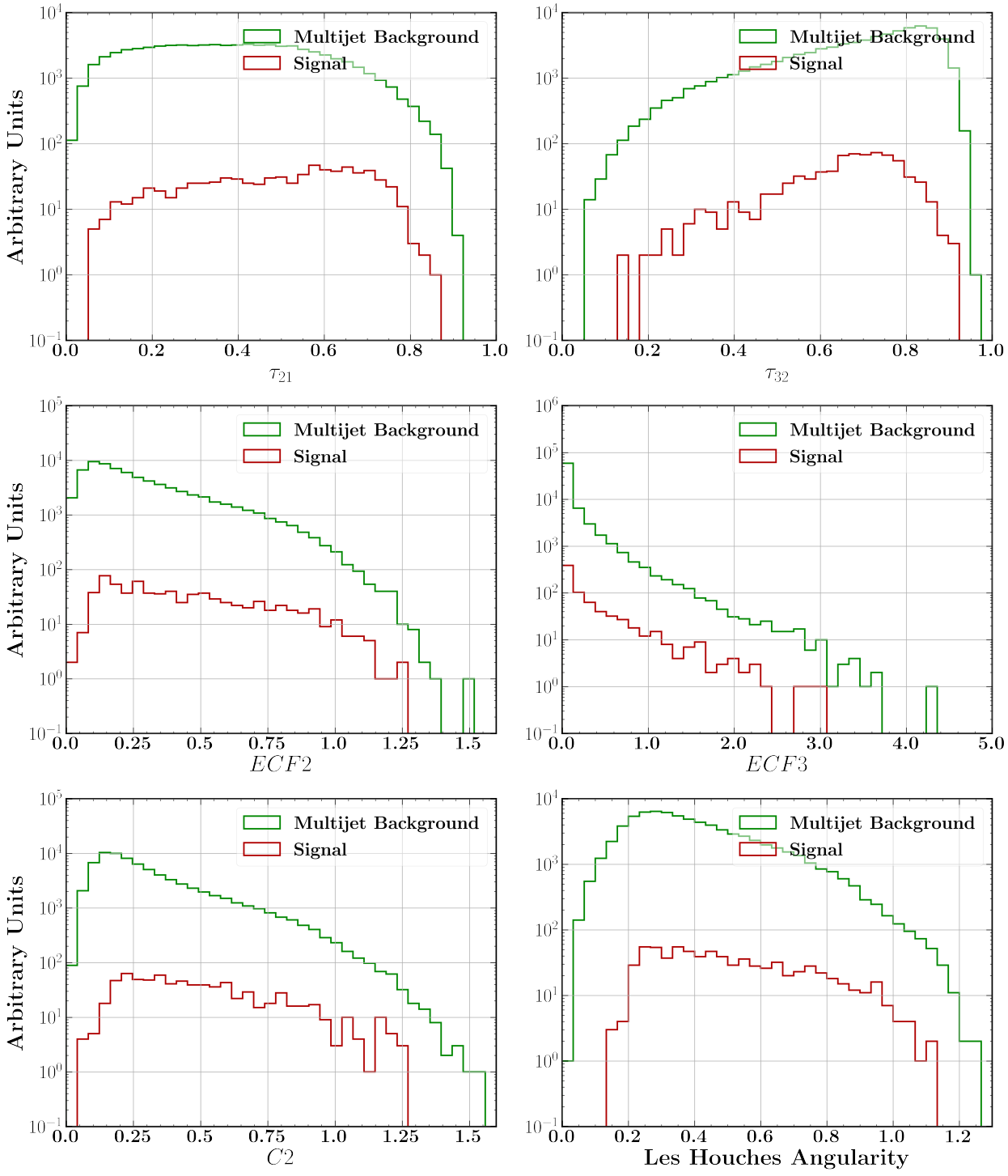


Figure 3.5: Distributions of jet substructure observables for the s-channel semi-visible jet production.

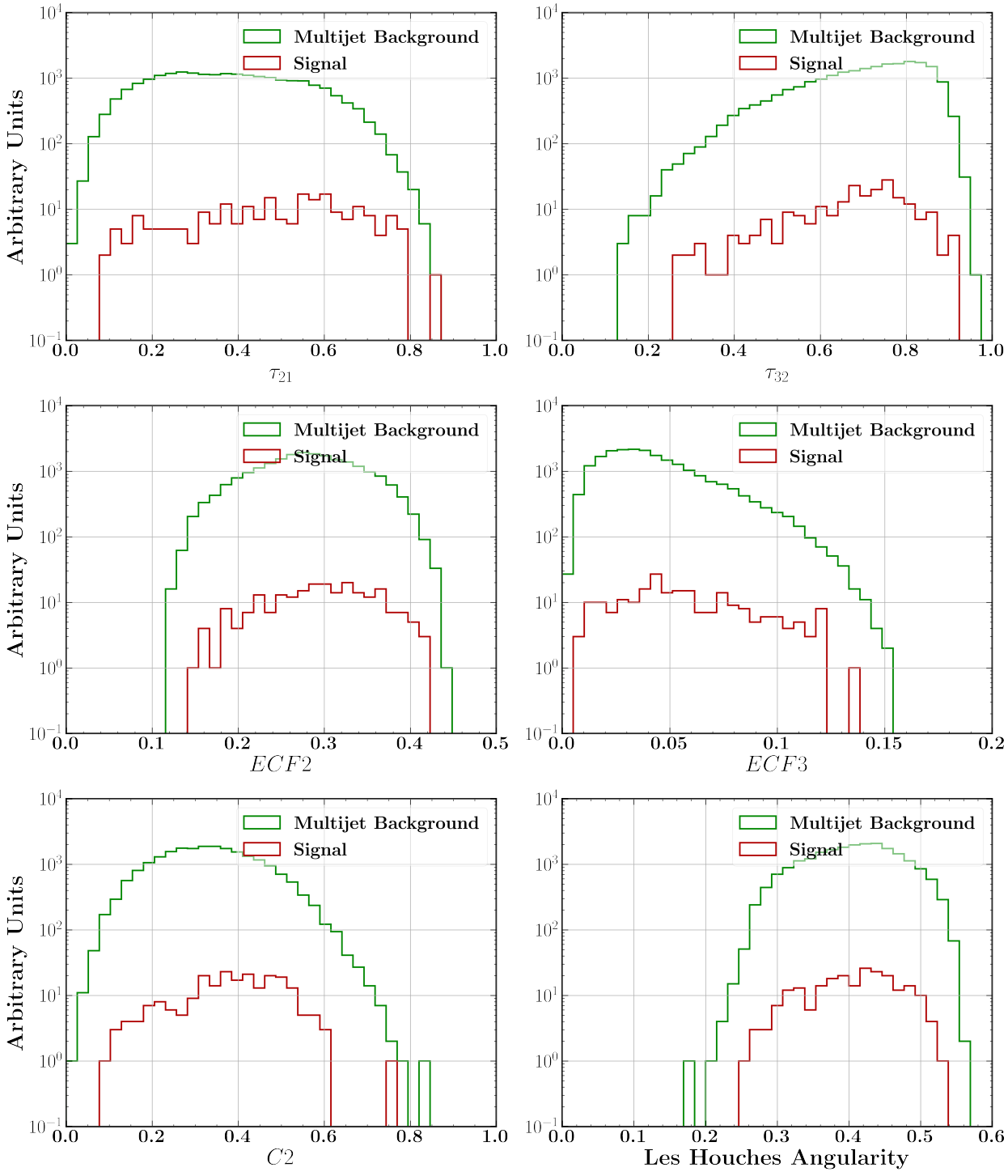


Figure 3.6: Distributions of jet substructure observables for the t-channel semi-visible jet production.

Model	N_d	n_f	Λ_d (GeV)	$\tilde{m_q}$ (GeV)	m_{π_d} (GeV)	m_{ρ_d} (GeV)	π_d Decay Mode	ρ_d Decay Mode
A	3	2	15	20	10	50	$\pi_d \to c\bar{c}$	$\rho_d \to \pi_d\pi_d$
B	3	6	2	2	2	4.67	$\pi_d \to s\bar{s}$	$\rho_d \to \pi_d\pi_d$
C	3	2	15	20	10	50	$\pi_d \to \gamma'\gamma'$ with $m_{\gamma'} = 4$ GeV	$\rho_d \to \pi_d\pi_d$
D	3	6	2	2	2	4.67	$\pi_d \to \gamma'\gamma'$ with $m_{\gamma'} = 0.7$ GeV	$\rho_d \to \pi_d\pi_d$

Table 3.1: Overview of the four models considered, with corresponding dark jet parameters as taken from [61].

3.3.2 Dark Jet Event Generation

Dark jet signal samples are generated with four possible situations depending on the choice of parameters mentioned in Section 3.2. The *PYTHIA8* program is used to simulate these events. Table 3.1 displays the four models' (A, B, C, D) parameter details, which are considered in this thesis. We also explore each model at different mass peaks, (500 GeV, 1000 GeV, 1500 GeV, 2000 GeV, 2500 GeV). The primary background samples are that of multijet processes, generated by *PYTHIA* 8.186. Simulation of parton showering and decay into final states produce events which are then sent to detector simulators. These Monte Carlo samples are passed through the ATLAS detector simulation by use of *GEANT4* [66]. Pile-up is then added by simulating more proton-proton collision in *PYTHIA8*.

Reconstruction

After moving through the detector simulator, the events are clustered into jets. In this case, the anti-k_t algorithm is used for the formation of jets, with the requirement of a jet radius, $R = 1.0$. Reclustering was not advantageous for these dark jet models, in contract to the semi-visible jet model. So, in order to remove pile-up effects and soft particles from the underlying event, jet grooming is performed. This ensures that the jet substructure observables which will be studied will not be concealed by these effects. For jet trimming, each jet is taken and reclustered using the anti-k_t algorithm once more, but now requiring jet radius $R = 0.2$, which form the group of subjets. The ratio of the transverse momentum of the subjet to the original, $\frac{p_{T,i}^{sub}}{p_T^{original}}$, must meet a threshhold of 5% or the subjet will be discarded in order to remove these soft objects. In other words, all remaining subjet's transverse momentum is $p_{T,i}^{sub} > f_{cut}$ where $f_{cut} = 0.05$.

Finally, the nature of the search requires that we focus on jets with high transverse momentum, that are within the central region of the detector. Thus, we ensure that all jets require $p_T > 200$ GeV and $|\eta| < 2$. The kinematic plots in Figures 3.7, 3.8, 3.9 and 3.10 display the distributions of the events after this selection.

Kinematic distributions

The following plots display important kinematic distributions related to Models A, B, C, D, with mass peaks of 1000, 1500, 2000, 2500 GeV. These are the distributions after the cuts of jets 200 GeV $< p_T$ and $|\eta| < 2$ are applied. We state again that all distributions are weighted by Equation 3.5 and are normalized by their cross-section.

Similar to what was stated with the semi-visible jet substructure observables distribution, it is difficult to motivate obvious cuts which may help our analysis. When comparing the different mass peaks, we can see that the larger masses have invariant mass peaks at higher numbers, which is to be expected. Other observables seem to show low degrees of separation for the signal and background distributions. More so, the Models A–D do you not show much deviation in their jet observable distributions, which lead to difficulty in identifying whether signal came from a particular model when an anomaly detection algorithm is used. However, the number of charged tracks n_{trk} in each model is considerably large when compared with the multijet background, especially at the 2500 GeV mass peak. As will be discussed in later sections, n_{trk} proves to be a powerful discriminator for many of the anomaly detection algorithms used.

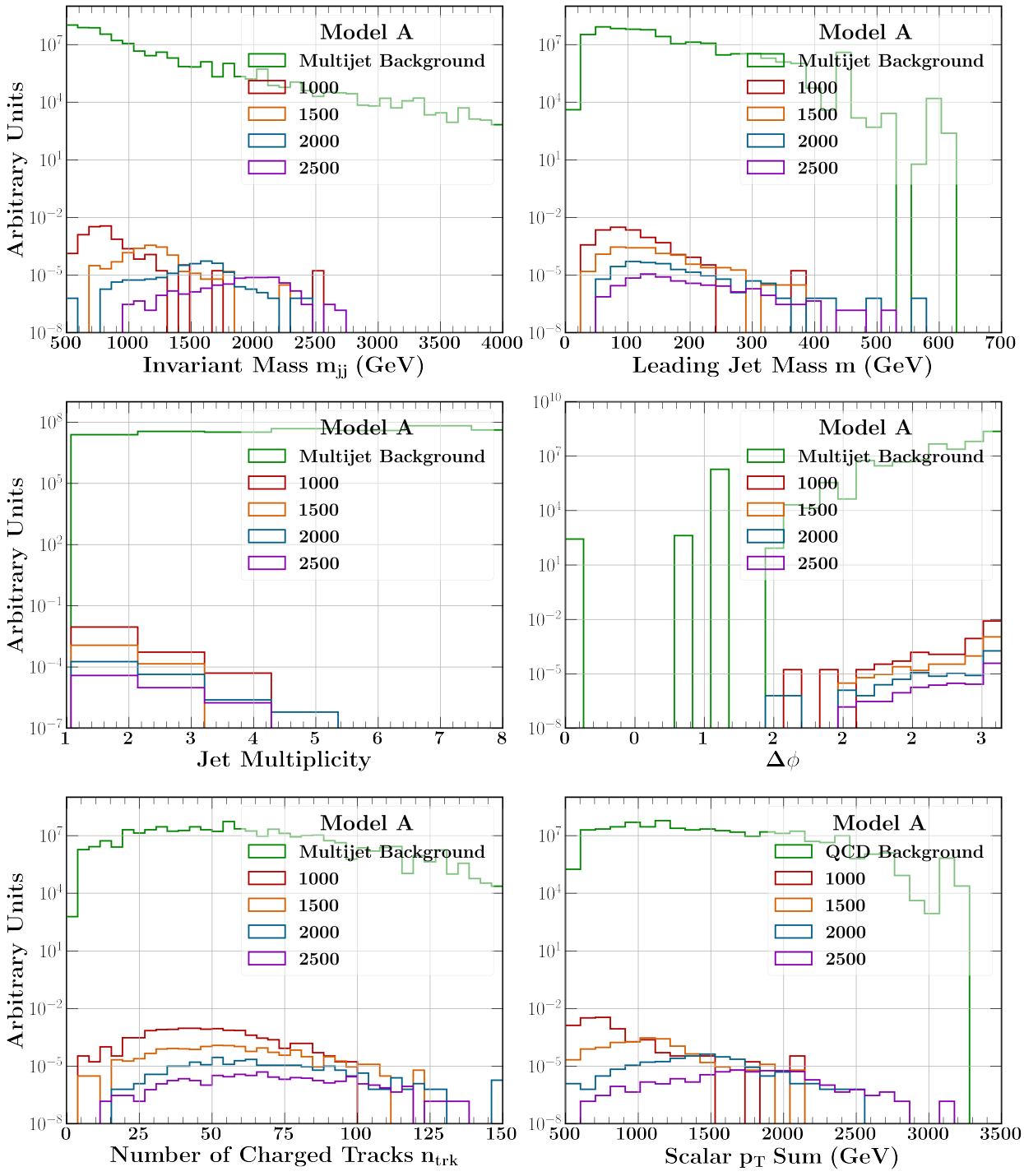


Figure 3.7: Kinematic distributions of simulated dark jet events with Model A parameters.

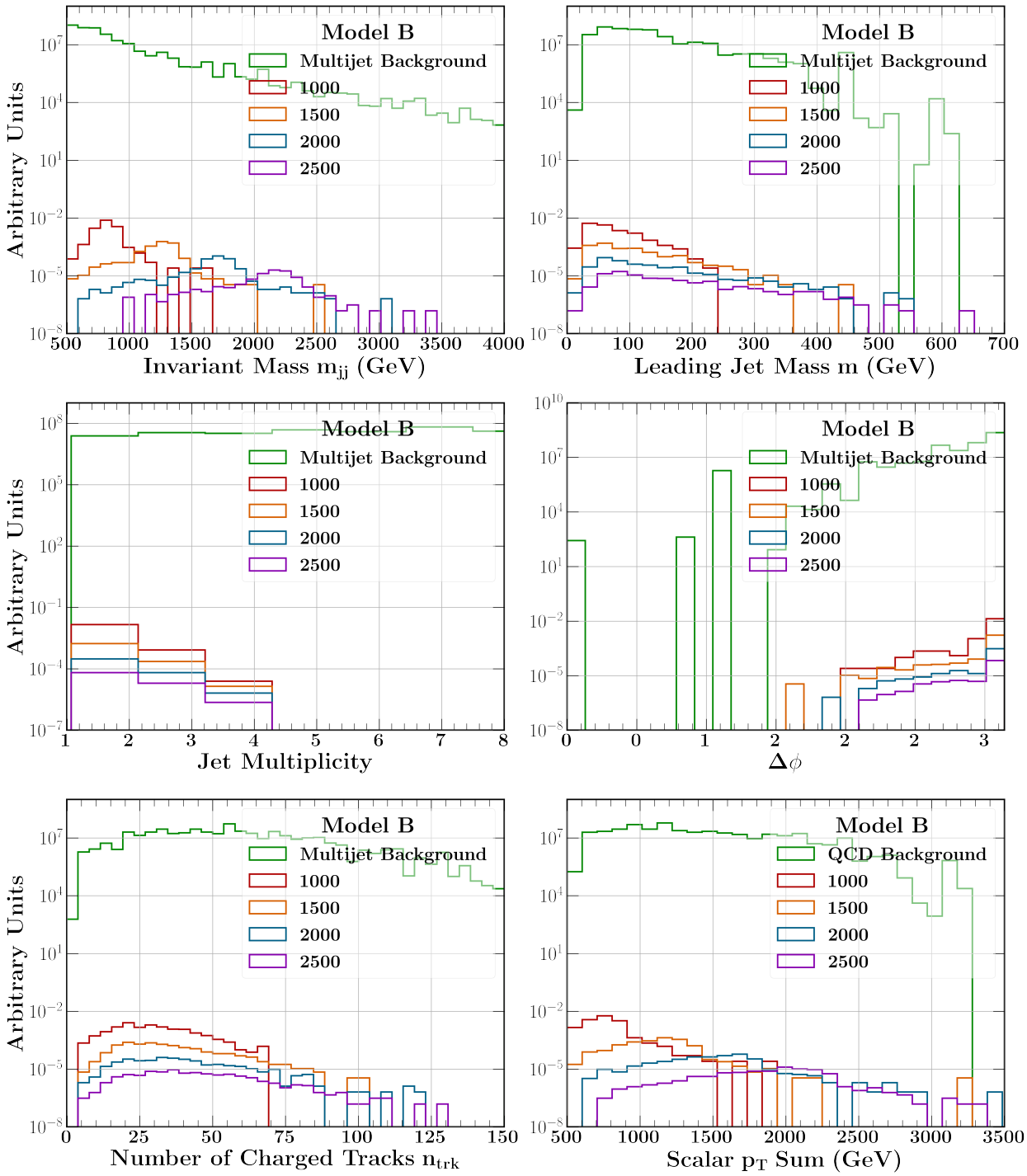


Figure 3.8: Kinematic distributions of simulated dark jet events with Model B parameters.

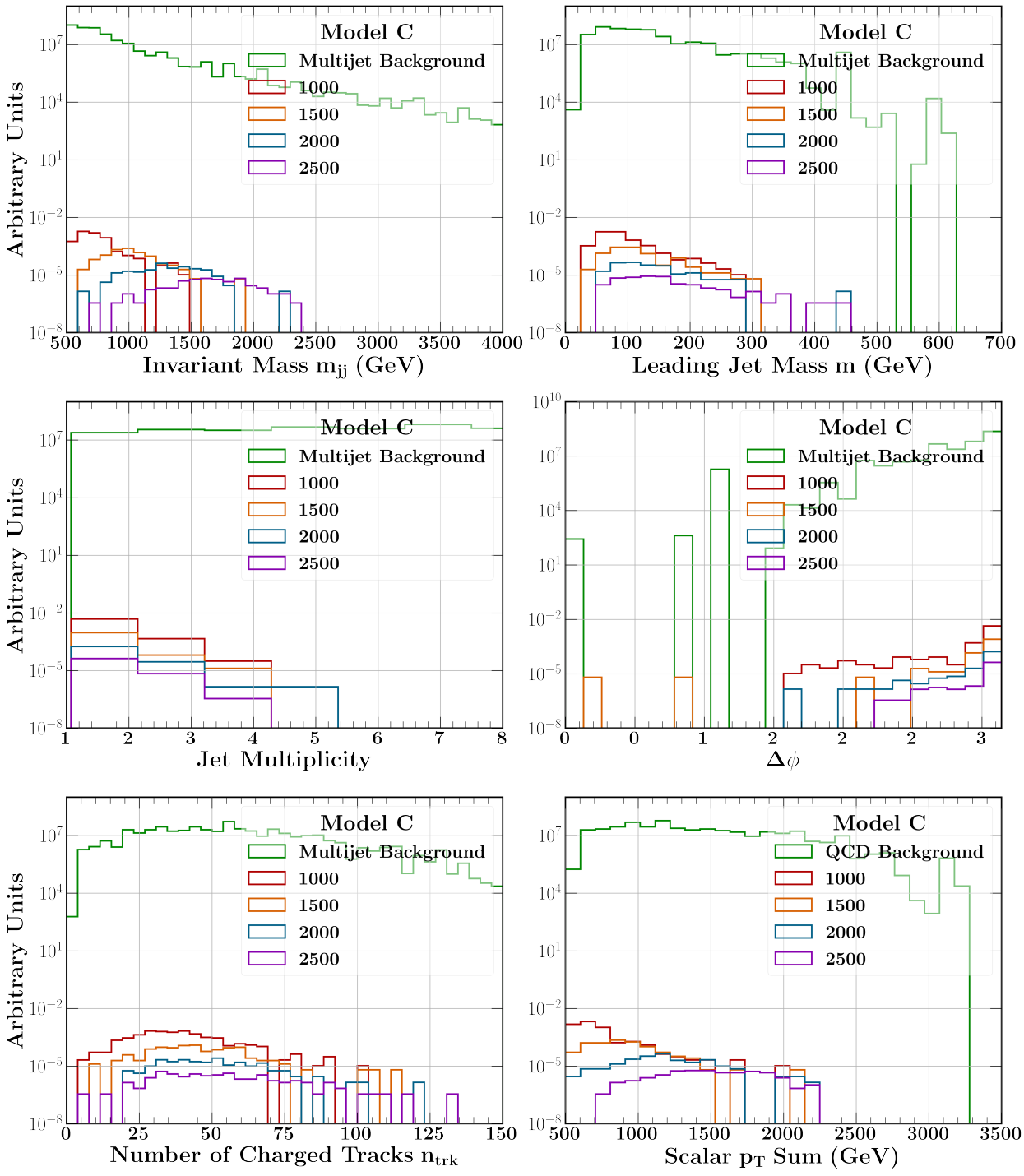


Figure 3.9: Kinematic distributions of simulated dark jet events with Model C parameters.

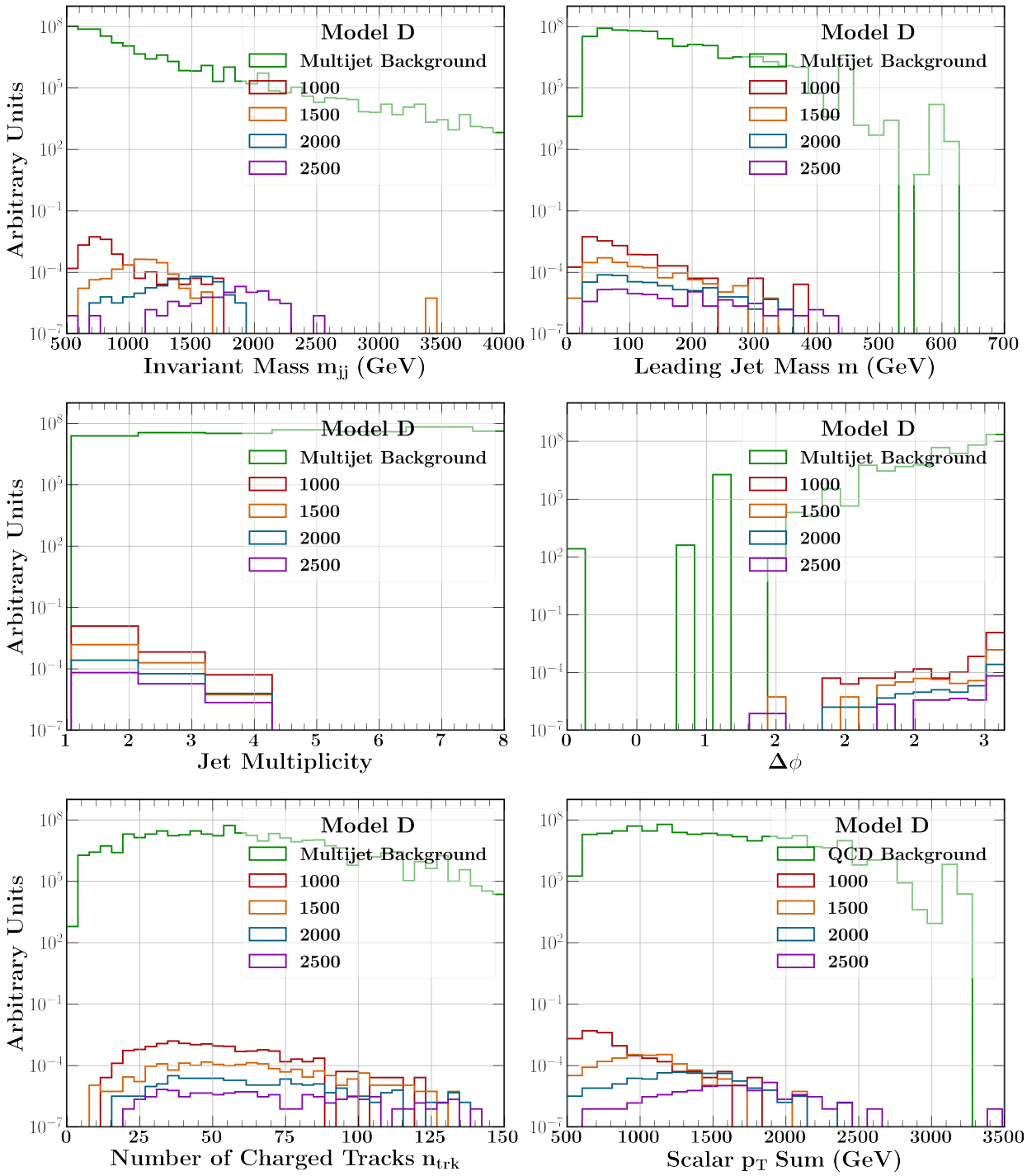


Figure 3.10: Kinematic distributions of simulated dark jet events with Model D parameters.

Chapter 4

Survey of Anomaly Detection Algorithms

4.1 Advances of anomaly detection algorithms in experimental particle physics

In this chapter, we aim to see whether novel anomaly detection algorithms developed in recent years may be useful for exploring the presence of BSM in dark and semi-visible jets. We expect dark and semi-visible jets to be difficult to distinguish from background jets in any single event. Thus, the recent breakthrough in ML research provides a good opportunity to show if these techniques can be used effectively in aiding current analyses being done for dark and semi-visible jets, such as improving sensitivity.

Out of possible millions of collisions at the LHC, only a small minority of events will contain a new exotic physics process (such as that from WIMPs or semi-visible jets) that we are interested in investigating. Thus, we ask given any event — what is the likelihood that is a signal or background event? What we seek to find is the optimal function f that can return a probability of signal or background, $f(\vec{x})$, given a single event which contains object such as jets and the jets' corresponding observables, $\vec{x}$. The classifier f can take many forms, with the main tool focused in this thesis being a deep neural network. Other techniques that have been used in within the field include Boosted Decision Trees for variable-choice optimization [67], Bayesian neural networks [68], as well as many complex clustering algorithms.

There are two important notes to discuss when considering a search for BSM physics using machine learning methods. Firstly, we expect the number of background events to far exceed the number of signal events. This is an important consideration to the problem as a large domain of ML use datasets that handle problems which contain an equal balance of signal and background processes. Therefore, there are not many examples for the classifier to correctly fit signal. Secondly, for a model-agnostic classifier trained purely on actual collision data rather than simulated counterparts, our dataset is *unlabelled*. These characteristics make up the domain of *unsupervised* machine learning called *Anomaly Detection* (AD).

The recent popularity of Machine Learning (ML) techniques has led to many uses within High Energy Particle Physics. These include utilizing Generative Adversarial Networks (GANs) to simulate background events [69] and the use of Deep Neural Networks and Boosted Decision Trees in classifying W Bosons and Top Quarks [70]. Most research both within the HEPP and ML communities has revolved around the use of *supervised* learning, whereby, collision events are *labelled* as either background or signal which requires a preconceived model in a simulation. For a search that

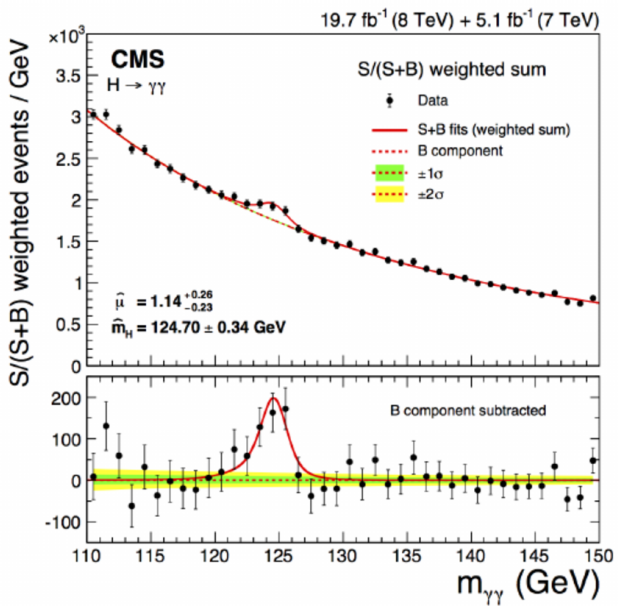


Figure 4.1: Example of a bump-hunt. CMS diphoton mass spectrum, $m_{\gamma\gamma}$. There is a signal resonance near 125 GeV where we see the extra collection of events — due to a new physics process — form a bump. Solid red lines indicate the polynomial fit of the signal (S) and background (B), while dashed red shows only the background fit. The bottom graph displays the spectrum with the background subtracted, which exaggerates the signal which forms the bump [73].

is truly model-agnostic, there is growing community within ATLAS to make use of *unsupervised* learning or Anomaly Detection (AD), whereby only collider data is sufficient for distinguishing events [71] [72]. Anomalies may be produced from a different mechanism compared to standard processes, and thereby point to the apparency of exotic matter such as DM.

4.2 Bump-Hunting

Before describing the anomaly detection algorithms investigated, it is important to state how we can evaluate the performance of any such method in separating signal and background. Chapter 2 focuses on the details and data of physical objects we get in the final state of a single collision of particles, or events. In actuality, we collide a huge number of particles that in turn produce a huge number of collision events. Even after selection by the Trigger system mentioned in Section 2.2.1, only a small fraction of these events will actually contain interesting physics mechanism which may hint at evidence of BSM theories.

In trying to investigate how BSM physics works, we look for signs of signal in our collision data. We conduct a search of a new physics model, which we know will produce a specific final state signature that is different to the typically predicted SM, in our events. However, a combination of processes that we are not interested may produce the same signatures, which we deem as background. Thus, in a collection of collision events, we must disentangle the distributions of signal from multijet background.

Multijet background here would produce a smooth continuous distribution of, for example, invariant mass which is constructed by the many collision events. If we propose a new particle with an invariant mass of 100 GeV, which decays into SM particles before reaching the detector, then we

will expect some collisions to have produced this particle. This means we should see a distribution of collision events for invariant mass where it is smooth due to background processes, but a small peak around 100 GeV, as there is an excess of events which were produced by this new particle. The method of searching for this unevenness of events (resonance) is aptly named bump-hunting and an example is shown in Figure 4.1. Other distributions can also be used, such as the case of a specified particle rarely interacting with ordinary matter and the detectors, and so we should see a resonance signal in the transverse missing energy distribution, $\not{E}_T$.

Relevant to this thesis, we highlight the fact that we conduct searches using new proposed models which produce a final state signature that we look out for. These models include supersymmetry (SUSY) as well as models with extra dimensions. Considering the previous chapter, we analyse the jet kinematic observables as well as the kinematic observables of dark and semi-visible jets presented in Sections 3.3.1 and 3.3.2. Each model and relevant mass peak for dark jet production may create a different final state resonance which we hope to discriminate against the multijet background. Useful discriminating observables such as n_{trk} are used in order to make signal events more prominent in the distributions of the models we are considering in this work.

4.2.1 Assessing Performance

After the relevant object and event selections are made, the goal of an analysis is to keep signal events, S while reducing the number of background events B. Assuming the total number of events, N is Poisson distributed, then the corresponding uncertainty is $\sqrt{N}$. Given a certain cut that we make on an observable, we would like to increase the value of the significance, $\frac{S}{\sqrt{B+S}}$.

Here arises the problem for any good analysis, in that we would like to gain a low uncertainty, but large significance. When applying cuts to the observables, we may increase the significance, but the remaining events here give a high uncertainty as $N = B + S$. On the other hand, a cut which retains many events has low uncertainty, but the ratio of signal to background is too low for any conclusions to be made.

It is essential to explain how we do evaluate the quality of the cuts being made and this is where the ROC curve as well as its partner, Area Under the Curve (AUC) is required [74]. This graph is necessary to both visualize and check the execution of our cuts. We define the true positive rate (TPR) to be the rate at which the values above the cut correctly predicts signal when the actual event does contain signal. This is in contrast to the false positive rate (FPR), which defines cuts predicting signal when the result is background. Their inverses when predicted background is actually background and when the events are signal are respectively labelled false negative (FN) and true negative (TN). Thus, we can now define,

$$TPR = \frac{TP}{TP + FN} \quad , \quad FPR = \frac{FP}{FP + TN}. \tag{4.1}$$

Thus, we plot TPR vs. FPR to form the ROC probability curve, with the area underneath being the AUC. The AUC's value represents the percentage accuracy the cuts are able to separate one outcome variable from another. A good model expects to have an AUC close to 1 (100 percent correct classification) whilst 0.5 implies the cut is equally likely to suggest an event as signal versus background. We tend to prefer a background rejection, ϵ_B (1−FPR) versus signal efficiency, ϵ_S (TPR) curve.

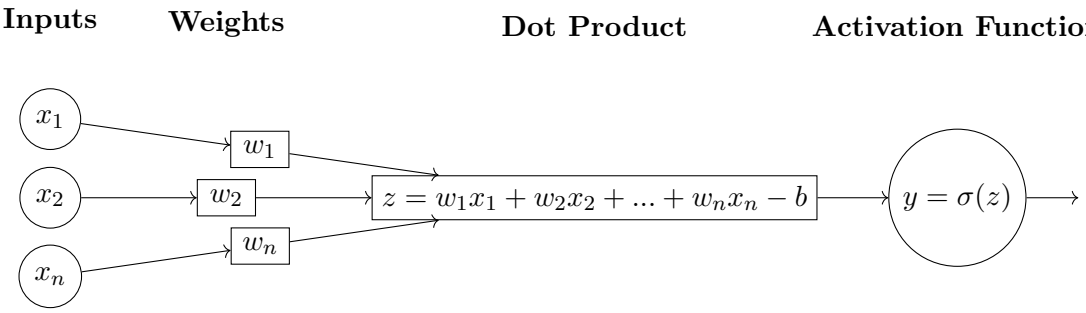


Figure 4.2: Diagram of a Perceptron. The input variables, which are the jet substructure observables, $x_1, \ldots, x_n$ are weighted respectively by $w_1, \ldots, w_n$. We subtract this linear combination by the bias value, b to arrive at z. The activation function, $\sigma(z) \in [0, 1]$ is typically portrayed as monotonically increasing function with a steep slope centred around 0. Examples include the logistic function, $\sigma(z) = \frac{1}{1+e^{-z}}$ or increased range function, $\sigma(z) = \tanh z$ as shown in Table 4.1. We describe lower values of y as large probability of background, and higher values closer to 1 as an event more indicative of signal present.

4.3 Deep Neural Networks

Neural networks are a class of machine learning algorithms that are designed to roughly resemble how neurons in the brain operate. Considering the wide-scale success in recent years, as well as its flexibility to adapt to a wide range of tasks, it is important to briefly state how the tool works. The building block of any network is a *perceptron*. A perceptron is consists of three main components: the variable inputs, $\vec{x}$, learnable weights vector $\vec{w}$ and bias b, and an activation function σ. A perceptron defines the process of calculation:

$$y = \sigma\left(\vec{w} \cdot \vec{x} - b\right). \tag{4.2}$$

This is visualized in Figure (4.2). The activation function is usually chosen to be monotonically increasing and its range $y \in [0, 1]$. For a given input $\vec{x}$, such as the observables of a jet in an event, we can weight the importance of the observables with weights, $\vec{w}$. If the dot product total is large enough to reach a given threshold b, the activation function will output a value closer to 1. However, if the given weights and variables does not overcome the required bias, the outputted value is will be closer to 0 (a lower value). We may also stack multiple perceptrons into a single *layer*, where each node may have the same input but different weights and biases. Although straightforward, the importance here is that the outputs of a layer can be used as the input for another layer of perceptron. In fact, each layer may contain an arbitrary amount of nodes, and we are free to choose how many layers we may connect to each other. A collection of these objects can thus be called a neural network as they can be understood to resemble neurons firing off in the brain.

The layers are classically split into three sections and is visualised in Figure (4.3). An *input layer* contains a column of neurons that can readily take the variables of a jet to be fed into the network. From this layer, we may have an arbitrary length of *hidden layers*, where the result of each neuron is used as the input to every neuron in the next layer each connected to each other. A configuration where every node's input is the combination of the previous layer's output defines a *dense network*, where we note that it is not always in best interests to connect every node this way. The final section is the *output layer* which typically contains a few nodes which are fed all results of the final hidden layer's perceptrons. The outputs of this final layer are expected to be

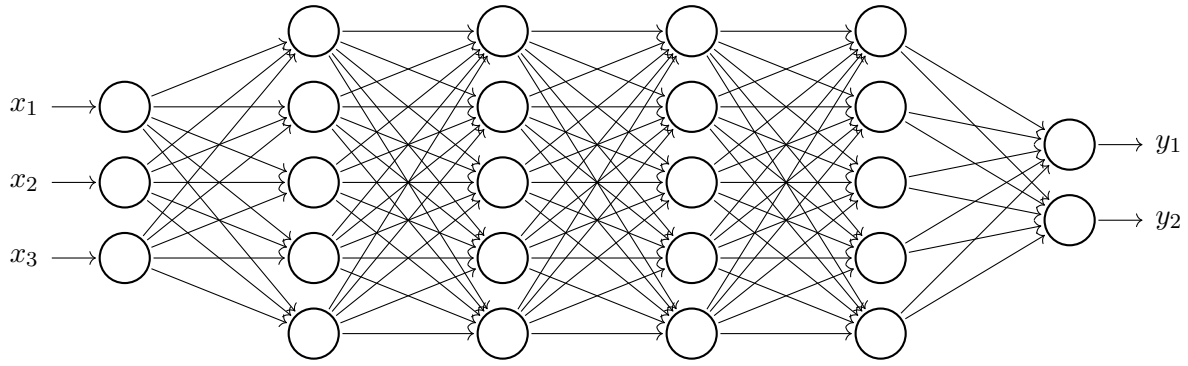


Figure 4.3: Dense Neural Network Architecture. Each node represents one perceptron, as depicted in Figure (4.2), while the arrows drawn typically depict the weight multiplications on the output of the activation function. Variables, x_1, x_2 and x_3 are fed through the input layer initially. Following this, there is a section of four hidden layers, each layer containing 5 nodes each. The final output layer returns the value for the probability of background, y_1 or probability of signal, y_2. The depicted neural is described as *dense*, as every node in one layer is connected to every node in the preceding layer, which may not be the most optimal architecture for a variety of problems.

meaningful results for researchers to interpret, such as the likelihood of signal present in a given event which contains jets. We may generalize Equation (4.2) with this neural network architecture when applying matrix operation to input $\vec{x}$ from a layer of length, m nodes, to the next layer of n nodes;

$$z = \begin{pmatrix} w_{11} & w_{12} & \dots & w_{1m} \\ w_{21} & w_{22} & \dots & w_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \dots & w_{nm} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_m \end{pmatrix} - \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}, \tag{4.3}$$

where $y = \sigma(z)$ makes use of the activation function which recall will then be used as the input of the next layer.

In summary, we can describe a neural network as a function of matrix operations. Therefore, we define this setup as the classifier, $f_w(\vec{x})$. In a standard classification task such as tagging jets, $\vec{x}$ would be the jet substructure observables and $f_w \in [0, 1]$ the probability of signal in the event. We note that the free parameter to choose here is the set of weight matrices, w. Thus, we may pose the problem of distinguishing signal from background in an event as the finding of the optimal parameter weights, w required to construct a model, f_w.

In brief, we may find the optimal weights by gathering the example dataset of jets $\vec{x}$ for N events and a corresponding signal/background target $y \in \{0, 1\}$. Then, we may construct the following classifier f_w which seeks to minimize the objective function, $\mathcal{L}$. For a standard technique such as linear regression, the objective function would be the mean squared error,

$$\mathcal{L} = MSE(\vec{x}) \equiv \frac{1}{N} \sum_{i=0}^{N} \left(f\left(\vec{x_i}\right) - y_i \right)^2, \tag{4.4}$$

Name	Function
Linear	$\sigma(x) = x$
Sigmoid	$\sigma(x) = \frac{1}{1+e^{-x}}$
Hyperbolic Tangent	$\sigma(x) = \tanh x$
Linear Rectifier (ReLu)	$\sigma(x) = x$

Table 4.1: Overview of activation functions.

where we total the residuals of the i-th event. However, different objective functions in this thesis have been explored due to the different properties they have to the given problem and goals of the architecture. In general, the objective function will be minimized when the optimal weights, w ensure that the function f_w accurately matches the signal/background target of the event dataset, y from a given set of event variables, $\vec{x}$.

A large amount of machine learning theory has been devoted to methods on how to find these optimal parameters. It is important to note that this thesis primarily utilized gradient-based optimisers such stochastic gradient descent.

Activation Functions

The choice of activation function, $\sigma(x)$ for each layer in a neural network plays an important in the results of the network. Table 4.1 displays the most popular choices for the activation function. We tend to avoid using a linear activation function as it is not well suited to classification problems, as its output range can vary dramatically. Also, we would like to exploit the non-linearity of a particular process and system, and stacked layers of linear activations, which may as well be reduced to a single linear layer, is not optimal for most problems.

It is important to note the monotonically increasing nature of the sigmoidal and hyperbolic tangent function, which have a steep increase at the center of its domain. This mimics the smooth but sharp increase of neuron exciting. It is also important to highlight that they both have a confined range of $[0, 1]$ and $[-1, 1]$, so they are well-suited for classification problems.

ReLu is the most popularly used activation function currently and is used widely throughout this thesis. The effect of deactivating a neuron when the input is too negative, is useful in preventing unnecessary layers from influencing the classification of events.

4.3.1 Focal Loss

As stated previously, both dark and semi-visible models produce a low number of signal events compared to background event. Because of the low amounts of signal present, the classifier, f_w found may discard labelling signal accurately in order to label background processes correctly. This class imbalance problem appears in a variety of circumstances within machine learning and specific tools have been developed. Knowing this in advance allows us the opportunity to provide an objective function, $\mathcal{L}$, that can emphasize the importance of finding the optimal classifier, f that prioritise labelling signal events correctly, rather than background events.

Consider the popularly used *Binary Cross-Entropy Loss*,

$$\mathcal{CE} = -\frac{1}{N}\sum_{i=0}^{N} y_i \log(p_i) + (1 - y_i)\log(1 - p_i), \tag{4.5}$$

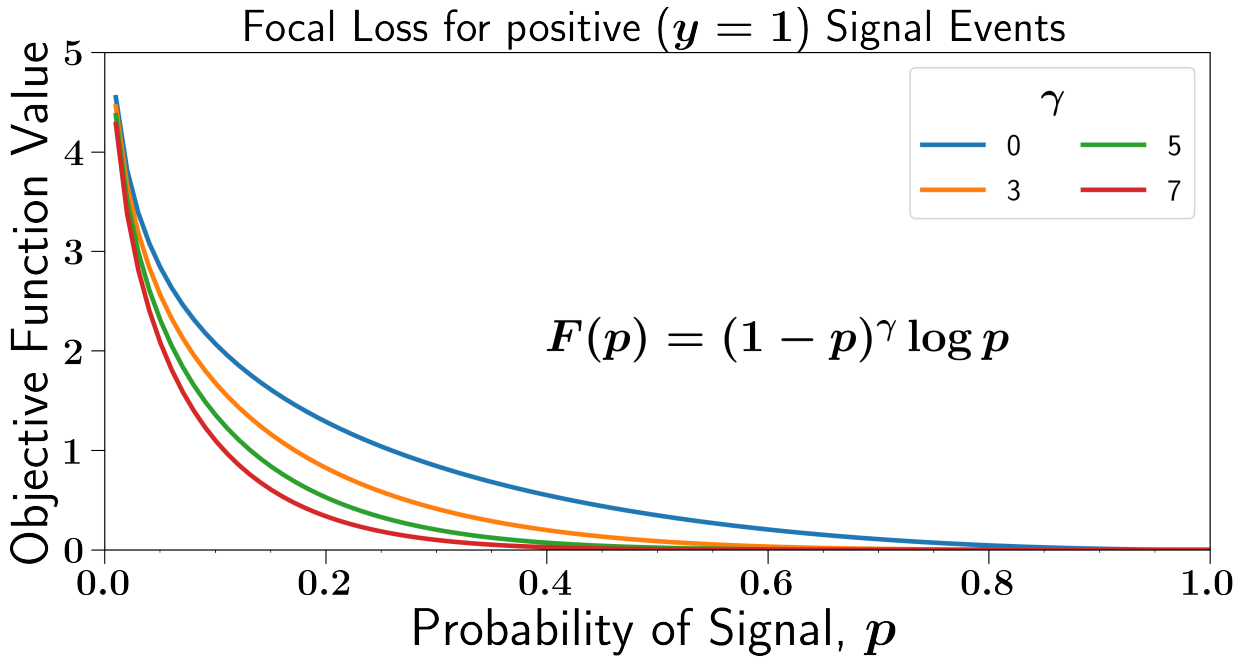


Figure 4.4: Focal Loss plot for signal events and $\alpha = 1$. In the case of a positive event ($y = 1$), the Focal Loss function in Equation (4.6) retains only the first term which is the depicted here. For $\gamma = 0$, the function is equivalent to the first term of Binary Cross-Entropy Loss (Equation (4.5)). As γ increases, the objective function value for p values closer to 1 decreases far more rapidly. This implies that for any positive signal sample, the neural network classifier will be more readily inclined to distinguish the event as signal, as the objective function will decrease faster.

which is typically used for binary classification tasks such as object detection [75]. $p_i = f_w(\vec{x}_i)$ is the probability of classification of signal and y_i is the truth label of the i^{th} event. It is clear that when the event in question is background ($y_i = 0$), we retain only the second term, and only the first term will be kept if it is signal ($y_i = 1$). Therefore, an event which is wrongly classified by the function, f_w will heavily contribute to the total loss, as we take the log of small values of p_i or $1 - p_i$. However, as previously mention, a typical loss function such as this will prioritize the ease of classifying background compared to the difficulty and rarity of distinguishing signal events.

One remedy to this issue is to modify Equation (4.5) slightly, as proposed by [76] in 2018. [76] describes the need to identify rare objects in images. They develop the *Focal Loss* function,

$$\mathcal{F} = -\frac{1}{N}\sum_{i=0}^{N}(1-p_i)^\gamma \left(\alpha y_i \log\left(p_i\right) + (1-\alpha)(1-y_i)\log\left(p_t\right)\right), \tag{4.6}$$

where $\alpha \in [0, 1]$ is weighting term with γ a positive scalar. The introduction of the weighting term, $\alpha(1 - p_i)^\gamma$ allows us to more finely focus on signal events, by choosing larger values of α and γ which prioritize the first term of Equation (4.6). Figure (4.4) displays the effect of adjust the γ parameter, as the term $(1-p_i)^\gamma$ strongly incentives the neural network to correct distinguish signal samples. A larger value of γ decrease the weight exponentially when any signal sample is under question.

It is clear that using the focal loss function with $\alpha > 0.5$ will aid in the identification of signal in a supervised learning framework.

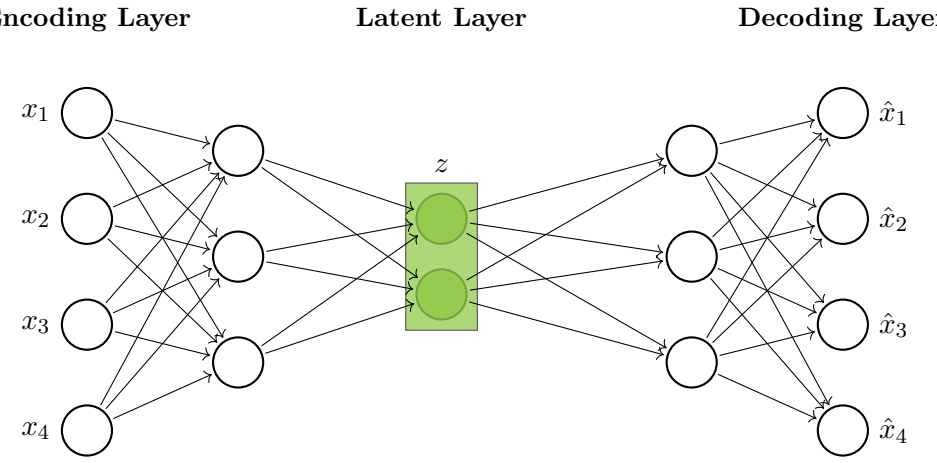


Figure 4.5: Architecture of a dense Autoencoder. Input variables, $\vec{x} = [x_1, x_2, x_3, x_4]$ are fed into the encoding layers. These layers decrease in node count from one layer to another, which compresses the dimensions of the event feature space until it reaches the latent layer, z which contains just two nodes (green block). We expect z to contain the majority of $\vec{x}$ variable information despite the reduction in dimension. From the latent layer, z the jet variables are reconstructed by mirroring the encoder layer. In this section, we increase the number of nodes in each subsequent layer until the event is fully reconstructed in the form of $\vec{x}' = [x'_1, x'_2, x'_3, x'_4]$.

4.4 Autoencoders

An Autoencoder (AE) is a particular type of neural network which seeks to accurately recreate its input as much as possible, through a compression/decompression architecture.

For any input $\vec{x}$, the autoencoder will seek to recreate vector and returned the recreated output, $\vec{x}'$. This would be as trivial operation as multiplying by unity, however the architecture of the AE makes this not as obvious. The AE is split into two segments: an encoder and decoder.

The encoder takes an event and passes it through hidden layers which are progressively shorter in node length compared to the input length. This continues until we reach the *latent* layer, often denoted as z. This hidden layer contains the shortest node length in the network and is understood to compress our event into a latent space. From this layer, the encoder architecture is reflected. That is, from the shortest node length in the latent space, the node length increases from layer to layer until we reach the original length of the event. This decoder section takes the compressed layer, z and reconstructs the input. This is the decoder and its final output is the reconstruction, $\vec{x}'$ of the original input of the encoder, $\vec{x}$. A diagram of the architecture is given in Figure 4.5.

The objective function, called the *reconstruction error*, $\mathcal{L}$ used here to minimize, operates similarly to the supervised learning case in Equation (4.4),

$$\mathcal{L} = \frac{1}{N} \sum_{i=0}^{N} \left(\vec{x_i}' - \vec{x_i} \right)^2 . \tag{4.7}$$

That is, we train the autoencoder to minimize the difference between the inputed $\vec{x}$ and the reconstrution, $\vec{x}'$, despite compressing the event initially around the latent layer z. After training, the autoencoder gives the optimal way to reconstruct the majority of events given a compressed form in z. Thus, given a collection of events that largely contain background, the AE will prioritise correctly recreate these events of the majority-produced process (multijet background), rather

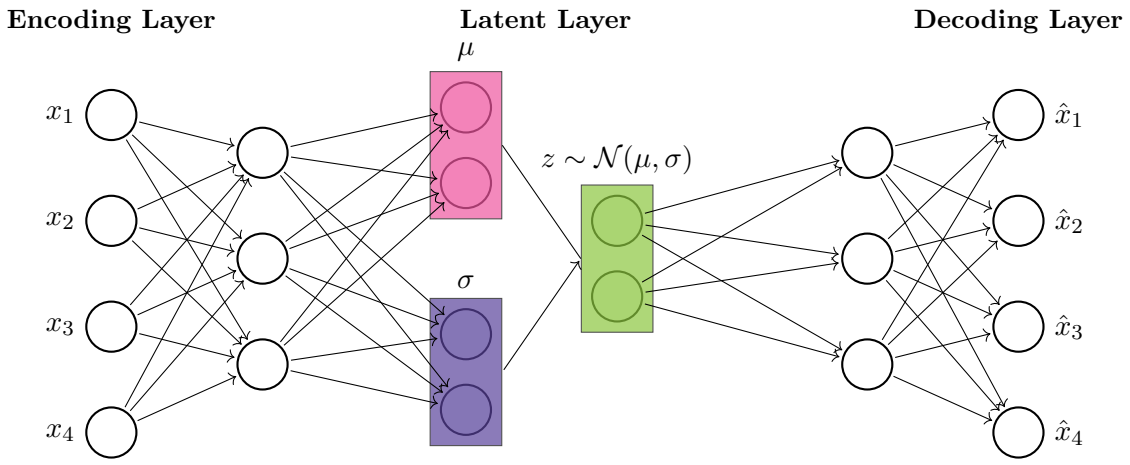


Figure 4.6: Diagram of a dense Variational Autoencoder. Similar to the architecture of an autoencoder displayed in Figure (4.5), a VAE contains an encoding and decoding layers utilized to reconstruct a jet event. The two networks differ in the latent layer, z (green block) which is now described as a random variable sampled from the normal distribution, $\mathcal{N}(\mu_i, \sigma_i^2)$. The optimal means, μ_i (pink block) and standard deviations σ_i (purple block) are learnt whilst training the VAE, similar to finding the optimal weights, w. Since the variational autoencoder introduces a distribution rather than using discrete variables, gaps that may have caused a standard autoencoder difficultly to reconstruct certain jet events, are now corrected.

than the anomalies produced, like signal events. This is because the dimensionality of the event has been reduced in the latent layer, z. This implies that a background event, $\vec{x}$ will look similar to the learned AE's reconstructed event, $\vec{x}'$, but will reconstruct the signal events poorly.

Thus, for any one event, we may calculate the Euclidean norm between $\vec{x}$ and $\vec{x}'$, that is $\|\vec{x}' - \vec{x}\|$. Larger values indicate a higher likelihood of said event being an outlier, as the event would be reconstructed poorly. By plotting a histogram of the Euclidean norm for all events, we may define a threshold value where event with a larger MSE value are classified as signal.

It is worth noting that the encoder and decoder can be used in a variety of other situations not related to anomaly detection but experimental analysis for physics. Such ideas include using the encoder to compress events for faster file transfer [77] and the decoder segment for construction of new events given a seed at the latent layer [78].

4.4.1 Variational Autoencoders

The Variational Autoencoder (VAE) builds upon the architecture of the autoencoder by combining it with the use of more probabilistic models [79]. Rather than have the nodes of the latent layer, z simply output the most compressed form of the event topology, we replace this with a normalized distribution instead. That is, each node in z is replaced by two nodes composed of representation of the mean, μ and standard deviation, σ of the normal distribution, $\mathcal{N}$. Thus, our latent space now consists of random variable, z sampled from the distribution $z \sim \mathcal{N}(\mu, \sigma^2)$, and a standard VAE is depicted in Figure (4.6). From this, we introduce a new term into the loss function which we seek to minimise. The Kullback-Leibler divergence (KL divergence), D_{KL} is an effective measure of how large the difference is between two given probability distributions. We minimize this value by ensuring each distribution in the latent space closely resembles the standard normal distribution,

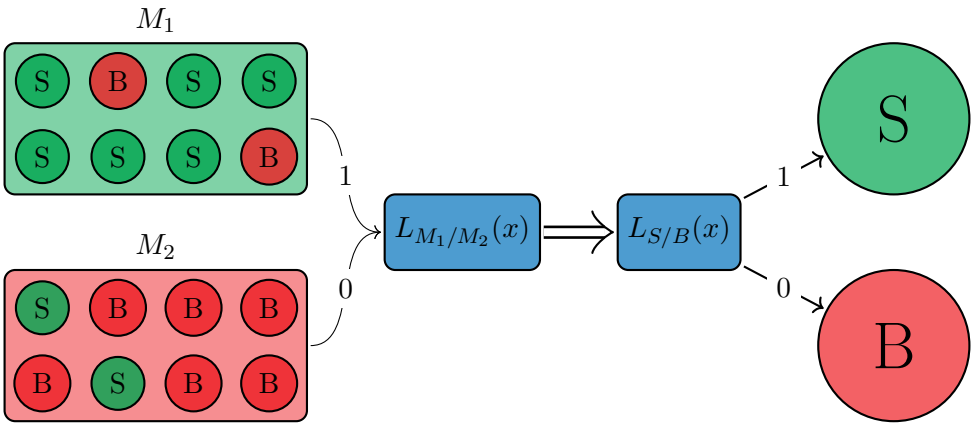


Figure 4.7: Diagram of the process for Classification without Labels (CWoLa). Consider two datasets which contain both signal, S and background, B samples. A classifier, L_{M_1/M_2} like a neural network architecture, is trained to distinguish between a signal-rich set, M_1 and background-rich set, M_2. If the concentration of signal is significantly large in M_1, L_{M_1/M_2} approaches the optimal classifier, $L_{S/B}$ that distinguish signal samples, S from background samples, B.

$\mathcal{N}(0,\,1)$ as possible. The loss function is then a combination of the Euclidean distance and KL divergence,

$$\mathcal{L} = \frac{1}{N}\sum_{i=0}^{N}\left(\vec{x_i}' - \vec{x_i}\right)^2 + \sum_{j=0}^{z} D_{KL}\left[N\left(\mu_j, \sigma_j\right)||N\left(0,1\right)\right]. \tag{4.8}$$

The result of this change leads to the autoencoder's accuracy in recreation of $\vec{x}$ improving vastly as it ensures that the latent variable z is clustered together closely in latent space as well as overlapping, yet globally centred around its target when averaged across all distributions. In effect, gaps that would otherwise be present in the AE are 'smoothed' and 'filled' by this statistical process, resulting in better reconstruction. The introduction of this aspect into the latent layer takes into account the different processes that produce jet-physics samples, and how various physics processes may combine to produce the final event sample. A classic autoencoder will fail to accurately represent many unknown physics processing occurring at the same time in particle collisions, but a variational autoencoder will improve on this aspect.

The process for finding a signal event is the same as the autoencoder, where we find large values of the Euclidean norm, but now with a more accurately reconstructed $\vec{x}'$.

4.5 Classification without Labels (CWoLa)

Classification without Labels (CWoLa) [80] present one of the strongest AD methods produced recently for aiding in the separation of signal to background events. Its versatility and applicability within the field has lead to the algorithm be utilized as a natural extension to a standard bump-hunt which is explained in Section 4.2. In many cases it has been used as a standard benchmark on which to compare new methods, and has already begun to be used for new ATLAS results [81]. Although simple, it is extremely effective in assisting in any search as it can easily be built upon with other existing methods, as will be discussed shortly.

The concept works as follows. In any typical analysis, a histogram plot (with signal and background events together) of the invariant mass, m_0 of the jets in an event, E is created. Through

this plot, we can define a region where we expect to find signal in the spectrum. Declaring a lower, m_{low} and upper mass bound, m_{up}, we find two sets of events:

$$M_1 = E(m_{low} < m_0 < m_{up}) \qquad \text{(Signal Region)} \tag{4.9}$$

$$M_2 = E(m_{low} > m_0;\, m_{up} < m_0) \qquad \text{(Sideband Region)} \tag{4.10}$$

It is important to highlight that events, E, contain jets which have many variables such as the psuedorapidity, η and ECF as well as other jet substructure elements, but we split the set of events based on the variable of invariant mass, m_0. These two mixed samples, M_1 and M_2 which contain both signal (S) and background (B) events. However, we expect higher proportion of signal events in the signal region, rather than the sideband region. The composition of them is such that,

$$p_{M_1}(\vec{x}) = f_1 p_S(\vec{x}) + (1 - f_1)\, p_B(\vec{x}), \tag{4.11}$$

$$p_{M_2}(\vec{x}) = f_2 p_S(\vec{x}) + (1 - f_2)\, p_B(\vec{x}), \tag{4.12}$$

where f_1, f_2 are the respective signal fractions. That is, the ratio of signal event to total events within the given set (i.e $f_i = \frac{\text{No. of Signal Events in } M_i}{\text{Total No. of Events in } M_i} < 1$). Now, it is clear that M_1 contains a larger percentage of signal, $(f_1 > f_2)$.

The optimal classifier that distinguishes samples drawn from the two distribution is the likelihood ratio, L. For the mixed samples and signal/background distributions, the likelihood ratio is, respectively,

$$L_{M_1/M_2}(\vec{x}) = \frac{p_{M_1}(\vec{x})}{p_{M_2}(\vec{x})}, \qquad L_{S/B}(\vec{x}) = \frac{p_S(\vec{x})}{p_B(\vec{x})}. \tag{4.13}$$

where p_S and p_B are the distributions of the signal and background processes respectively. So, we may relate the two likelihood ratios as follows,

$$L_{M_1/M_2} = \frac{p_{M_1}}{p_{M_2}} = \frac{f_1 p_S(\vec{x}) + (1 - f_1)\, p_B(\vec{x})}{f_2 p_S(\vec{x}) + (1 - f_2)\, p_B(\vec{x})} = \frac{f_1 L_{S/B} + (1 - f_1)}{f_2 L_{S/B} + (1 - f_2)}. \tag{4.14}$$

The last term is simply a rescaling of $L_{S/B}$ as long as $f_1 > f_2$. This displays that a classifier that learns the difference between mixed samples, L_{M_1/M_2} gives similar performance to a classifier that differentiates signal to background (fully supervised), $L_{S/B}$. This can be seen in finding the gradient,

$$\begin{aligned}
\frac{\partial L_{M_1/M_2}}{\partial L_{S/B}} &= \frac{\partial}{\partial L_{SB}} \left(\frac{f_1 L_{S/B} + (1 - f_1)}{f_2 L_{S/B} + (1 - f_2)} \right) \\
&= \frac{f_1}{f_2 L_{S/B} + (1 - f_2)} - f_2 \frac{f_1 L_{S/B} + (1 - f_1)}{\big(f_2 L_{S/B} + (1 - f_2)\big)^2} \\
&= \frac{f_1 - f_2}{\big(f_2 L_{S/B} + (1 - f_2)\big)^2}.
\end{aligned}$$

A diagram of the process of CWoLa in utilizing the mixed sets as well as likelihood ratio is portrayed in Figure (4.7). So, as long as we expect $f_1 > f_2$, then $\frac{\partial L_{M_1/M_2}}{\partial L_{S/G}} > 0$ and the function monotonically increases, and thus is an adequate rescaling of the actual optimal classifier, $L_{S/B}$. In the case of $f_1 < f_2$, the reverse classifier, $L_{B/S}$ can be approximated by L_{M_2/M_1}. An important highlight

compared to other methods of similar performance is that the fractions (f_1, f_2) do not need to be known explicitly beforehand.

It is important to note that the likelihood ratio's dependant variables, $\vec{x}$, does not include the invariant mass, m_0, which was used to define the two regions, as including this information would make the process of finding the likelihood ratio extremely trivial with only on variable. Rather, the set of other random auxiliary variables within the event should be used and contain as much discerning information as possible information surrounding the invariant mass without it being part of its calculation. Thus, we are using all aspects of the event, while utilising the different regions. The benefit of this method is that it can use all variables available to the event, rather than making cuts on the distribution, as can be difficult in higher dimensions with many variables.

4.5.1 Toy example

[80] contains a small example illustration of the method, which can also display a great visualisation of the technique. We consider an event with just three variables; the invariant mass m_0, and 'jet-substructure' variables in x, y coordinates, or $E(m_0, \vec{Y})$ where $\vec{Y} = \{x, y\}$ is the arbitrary-sized auxiliary information used to classify the event.

$f_1 = N_S = 100$ signal and $f_2 = N_B = 1000$ background events are generated, where the background is uniformly distributed over the square domain , $|x|, |y| < 0.5$ and mass range,

$$M_2 = E(|m_0 - m_{res}| < 2) \qquad \text{(sideband region)}.$$

The signal is confined to $|x|, |y| < 0.1$ and with signal region bin,

$$M_1 = E(|m_0 - m_{res}| < 1) \qquad \text{(signal region)}. \tag{4.15}$$

m_{res} represents a point in the mass distribution where we are at the center of the signal region. As described previously, we now have two sets, one which contains both signal and background, M_1, and another with just background, M_2. Thus, in implementation, this means we can train a DNN that distinguishes between the sideband and signal regions in order to estimate L_{M_1/M_2}, which may then be used as a classifier for signal against background process.

Figure (4.8) visualizes the process of CWoLa. The three variables for this illustrative example are plotted. The invariant mass distribution, m_0 is drawn on a histogram plot, where it is marked where the split into signal/sideband regions occurs. In this example, $m_{low} = -1$ and $m_{up} = 1$. A 2-dimension square domain displays a scatter plot of x, y variables. It is clearly shown there is a uniform distribution of background (purple), with the signal (orange) contained only within a square region at the center of the domain, as described previously.

A neural network is trained to distinguish M_1 from M_2 events, and thereby expect the same neural network to differentiate signal from background. The 2-dimensional histogram as the right-side plot of Figure (4.8) displays the probability of events classified as signal/background within different regions of the x, y space. As clearly seen, regions closer to the center of the domain space are far more likely to be classified as signal. The presence of high signal probability in the auxiliary variables, $\vec{Y}$, despite splitting the signal background on invariant mass, indicates the ability of CWoLa to use all information of an event, such as jet-substructure variables in order to accurately classify events.

It is noted in the paper that performance of this classifier greatly depends on the proportion of signal present in M_1. In other words, lowering the value of N_S, the number of signal events within M_1 has great effect on the outcome. A larger ration of signal to background, N_S/N_B in this distribution implies it is more likely that a classifier is able to locate the presence of signal events

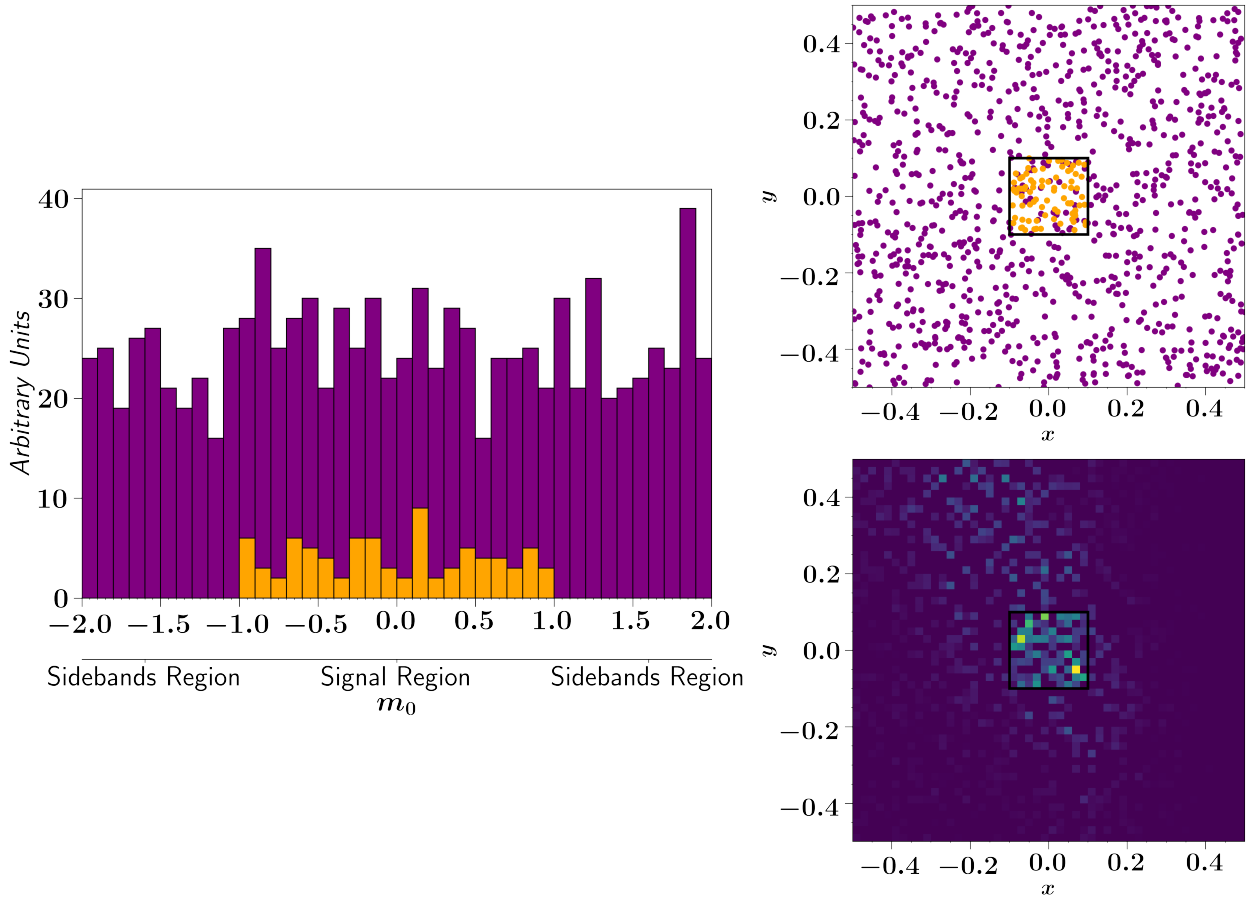


Figure 4.8: (Left) Histogram of example invariant mass, m_0 centered around 0, with signal in yellow and background events in purple. The background and signal are sampled randomly from the uniform distribution. The signal generated is only in the region, $-1 < m_0 < 1$ (signal region), while $m_0 < -1$ and $m_0 > 1$ are deemed the sideband region, As displayed in Figure (4.7), this poses the correct conditions for CWoLa by supplying M_1 and M_2. (Right) The top scatter plot portrays the example toy events in the $\vec{Y} = \{x, y\}$ coordinates. Again, both signal and background are randomly generated, but the signal events are confined to the smaller square domain, $-0.1 < x < 0.1$ and $-0.1 < y < 0.1$. The bottom 2-dimensional histogram plot visually describes the results of training a neural network to distinguish between in the signal and background regions from the invariant mass histogram. Yellow regions indicate larger probability of signal, which is largely located in the smaller square domain where signal events are dominant.

as the defining aspect that differentiates the signal from sideband regions. However, if there is a low amount of signal ratio, either due to the physical processes being studied that generate it, or a poor choice of m_{low} and m_{up} to define the two region,t he performance of the classifier drops as it finds more significant deviating aspects in the samples, that is not the presence of signal.

Considering how general the method and its strong performance in unsupervised learning, it has a wide range of expansion in its usage. It is not necessary to rely particularly on invariant mass, m_0 in order to gather sets M_1 and M_2. One simply needs some way of splitting a set once into sets of lower and higher proportion of signal. This can be done by choosing another variable, like other jet observables, or a more involved process such as Tag'N'Train. Further discussions in

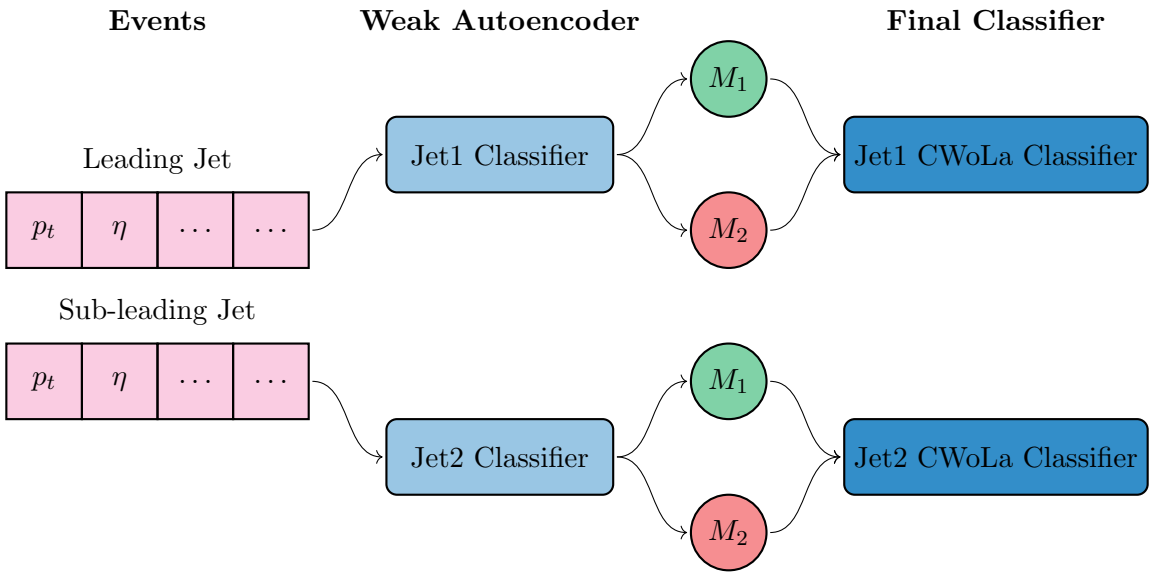


Figure 4.9: Diagram of the Tag N'Train process. A modification of the typical CWoLa process, TNT seeks to locate signal and background-rich datasets, M_1 and M_2 respectively, in the absence of a key variable, such as the Signal/Sideband regions found in the invariant mass distribution with resonant signal. A weak unsupervised classifier, such as an autoencoder, generates the required sets from a leading and sub-leading jet observables. A neural network is then solely trained on the opposing jet that was used for weak supervision. The results are two CWoLa-like classifiers trained on both jets, which we expect to be close to the optimal classifier between signal and background events.

[82] discuss its use in combination with a classic bump-hunt (mentioned in Section 4.2).

4.6 Tag N'Train

The Tag N'Train (TNT) method [83] takes direct inspiration from CWoLa as discussed in its paper. TNT discuss two shortcomings present in the CWoLa method which their algorithm seeks to address. Mainly, CWoLa requires the use of some signal mass region. If a new physics process does not produce a resonant signal in the invariant mass distribution, or any other kinematic or jet substructure variables, then we cannot define the sets M_1 and M_2 that are required to begin classification of signal from background. The other issue can be that the algorithm produces poor results when the auxiliary information in $\vec{Y}$ is strongly correlated with the invariant mass. The paper highlight that another anomaly detection algorithm, autoencoders are also vulnerable to poor performance due to its architecture as an encoder-decoder. Even positive results of an autoencoder may tell us very little about the nature of the signal events themselves.

The Tag N'Train approach seeks to resolve both problematic aspects of the respective algorithms by effectively combining the positive aspects of both. The requirement for TNT is the existence of two separate objects within an event which can be registered as signal-like or background-like. For most applications of this method, as well as all the use cases described in this thesis, these objects can simply be the leading and sub-leading jets in a typical dijet analysis. The method is visualised in Figure (4.9).

4.6.1 Method Outline

- We consider the first object, or the leading p_T jet of an event, which contains the jet substructure observables of the leading jet. Take a weak unsupervised function that is able to label the events into 2 sets — a signal-rich set of events M_1 and background-rich sample set, M_2. We note that the function does not have to perform particularly well, but simply have above-average performance in discriminating signal from background.

- We now take the set of variables in the second object, or sub-leading jet. Since these objects are sorted into two sets using the variables of the first jet in the corresponding events, we are not utilising the same jet substructure as the previous step. In other words, we train a supervised classifier to distinguish sub-leading jets in sets M_1 from M_2, where we remind ourselves that we retrieve these sets from the observables of the leading jets.

- We repeat the process with the roles of the leading and sub-leading jet reversed. In other words, we develop the sets M_1 and M_2 from the sub-leading jet this time, and then proceed to learn a supervised classifier to distinguish between the two event datasets. It is clear then that two deep neural networks will be trained, one based one the leading jet, and another on the sub-leading jet. It is expected that the performance will be roughly the same (same AUC scores).

In summary, Tag N'Train can be seen as a specialized expansion of CWoLa, applied particularly to dijet scans. Instead of relying on signal and background dominant distributions found by locating the signal and sideband regions of the invariant mass histogram, we construct the signal/background-rich distributions from any type event structure or distribution layout by using a weak classifier. This weak unsupervised classifier is usually an autoencoder, which will be used in this thesis. From understanding of CWoLA, it is clear that the results of this algorithm greatly depends as the strength of the initial weak classifier, as this the process which produces the signal-dominant set, M_1.

4.7 Dynamic Graph Convolutional Neural Network

A major topic to highlight in using unsupervised machine learning is the use of graphs. Many problems such as fraud detection [84] and traffic forecasting [85] have datasets of unordered groups. In the theory presented so far, the relationship between different jets in an event have not been explicitly presented to the neural network in order to be taken advantage of. Qu and Gouskos' paper [86] proposes the utility of structuring jet events as undirected graphs. These graphs can be useful in more-readily extracting information from a given event, and thus lead to better classification. Before summarising the algorithm presented, it is worth having an overview of how jets are presented as images, and the corresponding convolutional operations that have been used for jet tagging.

It is important to highlight that for this work we did not utilize jet images in the algorithms presented, and we do not gather any results which utilise jet images as the inputs to our algorithms. This means that we do not perform standard preprocessing operations on jet images such as translation, rotation and normalisation, in order to ensure optimal performance. Thus, the following discussion on jet images simply highlight how we can construct events in a variety of formats and then apply specific operations on these structures. In this case, jet graphs can be seen as a better representation of jet images, while utilising similar operations.

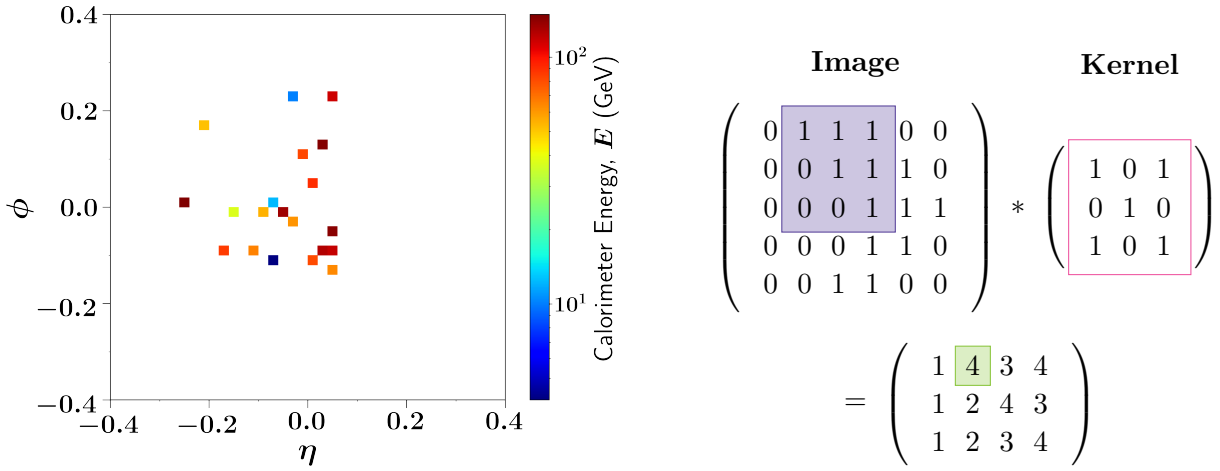


Figure 4.10: (Left) A typical jet image, generated as an example. The calorimeter energy, E values are plotted as colour pixel, with more blue indicating lower energy, whilst more red indicates higher values. Plotted on the η-ϕ plane, constructed is an image which is a 20×20 matrix, with entries corresponding to the calorimeter energy, E. (Right) Visual depiction of a convolution operation on an image. A region (purple block) of the the jet image, I is multiplied element-wise by the kernel, K (pink block) and the resulting values are aggregated in the bottom matrix (green block). The bottom matrix depicts the entire resulting convolution, $I * K$, where summation has been used as the aggregation operation. This matrix displays the results of shifting the kernel multiplication across the entire image, I. Note that in a neural network, the kernel K is composed of trainable weights, w which we seek to optimise.

4.7.1 Convolutions on Jet Images

Jet events can be represented as 2-dimensional images, typically displayed with a horizontal η and vertical ϕ axis. The left plot of Figure (4.10) displays an example image of 20 jets, where the colour represents the value of the calorimeter energy, E. Images are a well-used dataset standard within machine learning, and they are fed into a typical neural network through a special input layer called a *convolutional layer*. This layer is required, as many problems such as object detection require not only information like the intensity/colour of the pixels, but also knowledge on the arrangement of pixels within an image. A typical vectorised input, such as a list of jet substructure variables, does not indicate any spatial awareness that occurs between different jets. Thus, by constructing a jet image and performing convolution operations, we hope to retrieve useful information in the arrangements of the jets within an event, which may lead to better classification.

The right hand-side diagram of Figure (4.10) demonstrates how we may perform a convolutional operation on a jet image. We recall that a simple black and white image is essentially a 2-dimensional matrix, I where each entry is a value between 0 and 1. The kernel/filter, K is also a 2-dimensional matrix, composed of weights. To convolve an image, we consider a region of I with the same dimensions as K. The convolving operation, $*$ takes the element-wise product of the 2 matrices and then uses an aggregating operation such as averaging to return a scalar value. This scalar is saved into a resulting matrix, and we repeat this process until we consider all regions of the input image, I. Thus, the resultant $I * K$ is another 2-dimensional matrix, where each entry is weighted average or summation of pixels within a given section of I. Considering the output is also matrix, it too can be convolved, leading to a stack of layers within a DNN named *convolutional* layers.

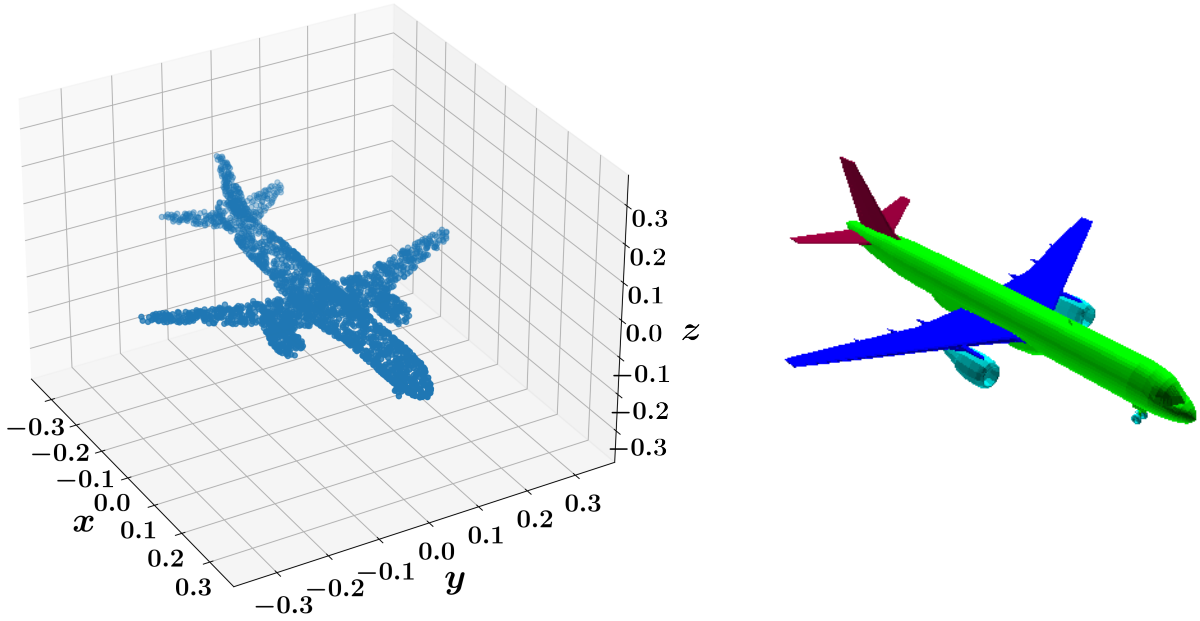


Figure 4.11: (Left) 3-dimensional point cloud plot of an aeroplane taken from the ShapeNet database [87]. A typical use of the graph neural networks is in point clouds, where object identification through segmentation is made. (Right) Segmentation of the point cloud aeroplane. As clearly depicted via colour, a classifier highlights regions of the aeroplane that is used for identification, such as the wings and tail. Utilization of the geometry of these point clouds in specific sections of the objects inspires [86] to apply the usage to jets in an event. Jets may also be represented as point clouds in higher dimensional space, similar to the left-hand side plot.

To incorporate this into our neural network framework, we note that the kernel, K is composed of the learnable weights, w which we seek to optimize in order to improve classification for signal/background events. With this paradigm, it is clear that *convolutional* neural networks (CNN) can take advantage of the spatial relation of pixels in an image. This is important in jet images as the CNN, as the arrangement of jets within an event may be of use when locating signal resonance.

There are shortcomings of applying a CNN as a process of image recognition for jet images. Jets that have a high pixel intensity (i.e. large E value) may be sparsely spread in the image, such as in Figure (4.10). Jet images are also created at different resolutions which may be problematic for the network to be applied generally. These problems lead us to search for a better data-structure that is able to represent jet objects similarly to images in a geometric or spatial aware sense, where we may perform convolutional operations, but avoid the issue of large 'gaps' between jets that are created in a high-resolution sparse jet image.

4.7.2 Graph Neural Network

Instead of a jet image, we may consider constructing a single event as an undirected graph, where each node is a jet with corresponding jet substructure and kinematic variables as the attributes of the node. Reconfiguring datasets into this graph structure has become increasingly popular within the machine learning community, as well as has been experimented with to produce great results

[86] [88] when handling objects such as jets in an event.

In a typical graph neural network, we consider the graph input, $G = (V, E)$ where V are the vertices/nodes, and E are the edge properties between vertices. To represent this structure mathematically, we require a feature matrix X which describes the vertices, V of our graph. X is a $N \times M$ matrix, where N is the number of nodes or jets in the graph, and M is the number of feature variables or often jet substructure variables. The edges, E are represented through a $N \times N$ adjacency matrix, A which represents the connection between nodes in the graph, G. For example, if the entry of the adjacency matrix, $A_{32} = 1$ then we expect the third vertex to be connected to the second vertex. Comparatively, $A_{14} = 0$, implies that the $1st$ and $4th$ node are not connected.

Using this paradigm, the operation in a typical layer of graph neural network, similar to Equation (4.2), is

$$X^{i+1} = \sigma(D^i \hat{A} X^i W^i), \tag{4.16}$$

where $\hat{A} = A + I$ is a modified adjacency matrix that ensure that a node and its neighbours features are propagated through layers in a network. D^i is simply the inverse of the degree matrix, $\hat{A}$ which is added as a normalizing operation. X^i is a feature representation of our graph at the i^{th} layer within the neural network, where we note that $X^0 = X$ is the initial inputed graph. W^i is again, our matrix of learnable weights which will be optimized.

We see in Equation (4.16) that the weight matrix, W performs as similar role to the kernel in a convolutional neural network, by aggregating a node and its neighbours in a graph. In the case of a CNN, a pixel and its neighbouring regions are taken into account by applying a convolution operation. In comparison, a node and its connected vertices, as well as their corresponding features, are taken into account by multiplying by the same weight matrix.

It is clear in this way that graph convolutional neural network can remedy the issues highlighted for jet image convolutions. Namely, as the structure of the jet events are now graphs, there is no concern now of sparsity or resolution issues as there were in constructed jet images.

4.7.3 Dynamic Graph Convolutional Neural Network

Because of the applicability of this framework, [86] take inspiration from the work of [89] in proposing an architecture used in classification and object segmentation of 3-dimensional point clouds, called a *dynamic graph convolutional neural network* (DGCNN). As reflected in both papers, this work is the result of an evolution of the PointNet architecture [90], which utilized each point set independently of the surrounding points in the neighbourhood. While the summary of GNN's presented thus far present a standard method for classifying nodes within a graph, [86] details how to exploit the geometric structure of vertices of a graph, in order to classify the full graph or signal/background event. This implies the differentiating of signal to background events by the arrangement of jets within a local region. For example, in the original paper, [86] use this architecture to classify an object as a aeroplane from the ShapeNet database [87], by identifying the individual plane-wings and turbines. The usage of this set is described in Figure (4.11)

Central to the architecture of a DGCNN, is the *EdgeConv* operation which is portrayed in Figure (4.12). Considering an event with 20 jets, it is processed by the neural network as follows:

- For each jet, x_i, a *K-nearest neighbour* clustering algorithms is run in order to locate the K nearest jets, x_{j_k} measured by the Euclidean distance of variables of the jets. The diagram in Figure (4.12) portray the 5-nearest neighbours of the jet, x_i.

- We define an operation, $e_{ij} = h_\Theta(x_i, x_j)$, named the *edge function* which is a function of the main jet, x_i and nearest jets, x_j. It is characterized by the learnable weights, Θ that will be

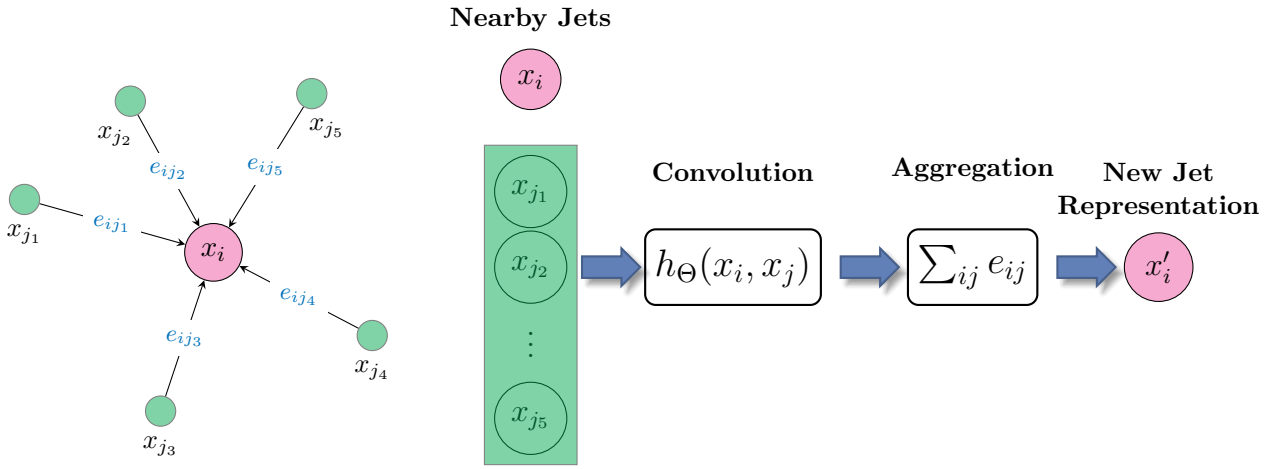


Figure 4.12: General process of Dynamic graph convolutional neural network. (Left) A diagram of $K = 5$ nearest neighbour jets, x_j (green) for the considered jet, x_i (pink). The edge features, e_{ij} (blue) are defined as a relationship between the considered jet, x_i and its neighbours, x_j. (Right) Summary of DGCNN in the paradigm of a neural network. The considered jet and its neighbours are used to find the edges, through a multi-layer perceptron, $h_\Theta(x_i, x_j)$. Θ is the set of trainable weights that are set to be optimized to produces the most relevant edge features. These edges are then aggregated (e.g summation), in order to produce a new feature representation of the considered jet, x'_i. We repeat this process to derive a new feature space for all jets in an event.

optimized during training of the network. In other words, h_Θ is a perceptron which takes in jets x_i and x_j and outputs the *edge feature* e_{ij} between them. We can also note that this function may take many forms. In the case of only utilising the considered jet, $h_\Theta(x_i)$ the algorithm is equivalent to PointNet [90].

- We retrieve the K edge features for the considered jet, x_i and apply an aggregation operation $\square_j e_{ij} = x'_i$ for the K edge features. Typically, $\square_j$ is the summation, $\sum_j$ or maximum operator. The vector x'_i is considered a new feature representation of the original jet, x'_i. We may then repeat this operation to construct a new feature representation of all 20 jets in the event.

The above instructions define a method of gathering a new feature representation, x'_i of an inputed set of jets within an event, by utilizing convolution operations on information between jets and their nearby neighbours (the edge features). Considering that the result of this process is another feature representation if a jet, we continuously stack *EdgeConv* layers in order to more accurately make use of the geometry of the jet events.

It is important to note that multiple layers of *EdgeConv* recompute the K nearest jet neighbours at the beginning of each layer. This makes the process *dynamic*, as the neural network is continuously tasked to exploit the relationship between jets at each feature representation in the layers, which may lead to better classification. By only utilising edge features of jets closest to each other, we seek structures which may hint at an event containing signal.

4.8 Combination/Lorentz Layer

[91] introduce a new type of layer architecture, called *DeepLolaTagger* which appreciate the underlying collision physics present when developing new machine learning technique that use jet

substructures properties. They propose this method as a way to address the deficiency present in using convolutional neural networks for jet images constructed in the η-ϕ plane. That is, the method produces poor results in the case of sparsely distributed pixels, which is the case when we gain better experimental resolution of the images. [91] show that *DeepLolaTagger* can handle finer granularity from calorimeter readings, as well as perform better than a CNN. *DeepLolaTagger* consists of two layers; the *Lorentz* and *Combination* Layers.

The setup of the proposed machine learning method is as follows. Any jet are composed of 4-momentum vector. For a given event, we may have N jets, sorted by largest transverse momentum p_T, that can be notated as the tensor, $k_{\mu,i}$,

$$k_{\mu,i} = \begin{pmatrix} k_{0,0} & k_{0,1} & \dots & k_{0,N} \\ k_{1,0} & k_{1,1} & \dots & k_{1,N} \\ k_{2,0} & k_{2,1} & \dots & k_{2,N} \\ k_{3,0} & k_{3,1} & \dots & k_{3,N} \end{pmatrix} . \tag{4.17}$$

The first column represent the 4-momentum of the leading jet in the event, with the following columns descending in order of p_T. In many cases, we may say that the first row is the calorimeter energy, E, while rows 1–3 represent the momentum, p of the jet in the x, y and z directions respectively. In other words, $k_{\mu,i} = [E_i, p_{x_i}, p_{y_i}, p_{z_i}]$.

4.8.1 Combination Layer

The first layer proposed simply multiplies the $k_{\mu,i}$ by a learnable matrix C_{ij}. It has the form,

$$C_{ij} = \begin{pmatrix} 1 & 0 & \dots & 0 & C_{1,N+2} & \dots & C_{1,M} \\ 0 & 1 & & \vdots & C_{2,N+2} & \dots & C_{2,M} \\ \vdots & \vdots & \ddots & 0 & \vdots & & \vdots \\ 0 & 0 & \dots & 1 & C_{N,N+2} & \dots & C_{N,M} \end{pmatrix} \tag{4.18}$$

The output expected at the end of the layer is

$$\tilde{k}_{\mu,j} \leftarrow k_{\mu,i}\, C_{ij} \, . \tag{4.19}$$

That is, $\tilde{k}_{\mu,j}$ is a 2-dimensional tensor that contains the initial 4-vector of N jets, $k_{\mu,i}$ (as it is multiplied by the identity matrix in the left section of C_{ij}), as well as M–N set of linear combinations of the jets and their respective momentum. We can say that each column represents the 4-momentum of a jet, k_j, with M jets assembling the outputted matrix, $\tilde{k}_{\mu,j}$. The coeffocients, C are simply the set of learnable weights, w as described in Equation (4.3), which are optimized by minimized the objective function of the neural network. The length of the j in C_{ij} (in other words, M) can be chosen on implementation and is discussed in [91]. They found that varying the number of trainable parameters in C_{ij} had no significant effect.

4.8.2 Lorentz Layer

This next layer seeks to transform the output of the *Combination* layer, $\tilde{k}_{\mu,j}$ into more physically meaningful objects than the linear combination of jets within an event, from which a DNN may more readily find important information. The 4 variables [91] chose to consider are the invariant mass, m^2, transverse momentum, p_T, a linear combination of energies, $w^{(E)}E$, and Minkowski distance between two jets k_j and k_m, $w^{(d)}d^2$. Explicitly, this distance value is calculated as,

$$d^2_{mj} = \eta_{\mu\nu}(\tilde{k}_m - \tilde{k}_j)_\mu(\tilde{k}_m - \tilde{k}_j)_\nu, \tag{4.20}$$

where $\eta_{\mu\nu}$ is the Minkowski metric. Thus, we produce M set of $\tilde{l}_j$,

$$\tilde{l} = \begin{pmatrix} m^2(\tilde{k}_1) & m^2(\tilde{k}_2) & \dots & m^2(\tilde{k}_M) \\ p_T(\tilde{k}_1) & p_T(\tilde{k}_2) & \dots & p_T(\tilde{k}_M) \\ w^{(E)}_{m1} E(\tilde{k}_1) & w^{(E)}_{m2} E(\tilde{k}_2) & \dots & w^{(E)}_{mM} E(\tilde{k}_M) \\ w^{(d)}_{m1} d^2_{m1} & w^{(d)}_{m2} d^2_{m2} & \dots & w^{(d)}_{mM} d^2_{mM} \end{pmatrix}, \tag{4.21}$$

where $w^{(E)}$ and $w^{(d)}$ are learnable weights found through the DNN. In summary, the first row calculates the invariant mass for each jet from the Combination Layer, $\tilde{k}_j$, and the second row represents the transverse momentum of each jet. Every entry in the third row each represent a summation of the energies of each jet, weighted differently by w^E_{mj}. Similarly, the last row are entries that each contain the linear combination of the distances and are weighted by $w^{(d)}_{mj}$.

As stated in the paper, the goal of introducing these two layers before any dense layer in a deep neural network, is to replicate a preprocessing stage that makes the network easier to learn defining features of a particle physical process. In short, the *Combination Layer* combines different aspects of the jets in the event and arranges as many possible relevant combinations. The *Lorentz Layer* then takes this set of jets and transforms them into more physics-related objects typically used to classify events.

We expect slightly higher performance using these initial layers, and faster training times. As said aptly within the paper: '*One can think of them as a rotation in the observable space, making the relevant information more accessible to the neural network.*'

4.9 Simulation Assisted Likelihood Anomaly Detection

Lastly, the final method presented is unique as it tries to incorporate the availability of simulated collisions in order to improve its accuracy. Simulation-Assisted Likelihood-free Anomaly Detection (SALAD) [92] again utilises a signal localized in a feature, m of the jet's distributions. That is, we split the distribution in m into a signal and sideband regions. By reweighting events using assistance from simulation, it is possible to perform a better bump hunt.

Begin by training a classifier f to distinguish simulation from data collision sets for only the sideband region. It can be shown that,

$$w(\vec{x}) \equiv \frac{f(\vec{x})}{1 - f(\vec{x})} = L_{\text{data/sim}} \cdot \frac{p(\vec{x}|\text{data})}{p(\vec{x}|\text{simulation})}, \tag{4.22}$$

where $L_{\text{data/sim}}$ is the likelihood ratio of actual collision data to simulation. This equation relates how to reweight any event in our simulation set to our collision set or vice versa, by multiplying of the weight, $w = f/1 - f$. Now, within the signal region, we reweight the simulated events. If the classifier, f is a neural network, then the function weight, w will automatically interpolate into the signal region. Finally, we train a second classifier, g to distinguish reweighted simulation from collision data within the signal region. We do this as $g(\vec{x})$ is related to the optimal classifier, as can be displayed by the monotonically increasing relation,

$$\frac{g(\vec{x})}{1 - g(\vec{x})} \propto \frac{p(\vec{x}|\text{S+B})}{p(\vec{x}|\text{B})}, \tag{4.23}$$

where the right hand side relates the ratio of signal to background in the actual collision dijet events.

Chapter 5

Tagging dark and semi-visible jets

This chapter highlights the results and findings of applying the various anomaly detection algorithms described in Section 4.1 on dark and semi-visible jet samples (Section 3.2). The aim is to find variables and/or anomaly detection algorithms which work particularly well enough to aid in a search for these dark matter signatures. As mentioned, improving the significance in a search like this is difficult, as the jets generated by these dark matter particles are expected to look very similar hadronic jets produced by QCD processes.

In this chapter, we share results concerning the t- and s-channel production of semi-visible jets. The datasets we consider here are the simulated events generated according to specifications of the SVJ model given in Section 3.3.1. The corresponding jet substructure distribution for this set of samples is displayed in Section 3.3.1. For dark jets, we investigate the s-channel production Models A, B, C and D at differing mass peak signals 1000 GeV, 1500 GeV, 2000 GeV, 2500 GeV, as detailed in Table 3.1 of Section 3.3.2. The kinematic distributions of the simulated dark jet events are given in Section 3.3.2.

All algorithms and supporting code like the generation of ROC plots, was implemented in Python 3.0. As the algorithms all have a foundation in deep neural networks, the Python package Tensorflow 2.0 [93] provides the back-end heavy computation and handling of optimisation methods, while Keras [94] provides the front-end support for the building of the various neural network architectures required. Unless otherwise stated, the ADAM optimizer [95] is used in order to minimize the loss function of the network

It is important to note that this thesis does not intricately investigate the optimisation of hyperparamters of these architectures. Hyperparamters such as batch size, epoch length, the use of dropout layers and much more can have profound impact on the performance of classification. We keep all hyperparameters mostly consistent in order to have meaningful comparisons between aspects of the algorithms we are comparing.

Implementations of the algorithms as well as the generation of figures can be found at `https://github.com/roygusinow/ANOM_JETS`.

$k-$Folds

Machine learning and its sub-disciplines involve statistical processes on large datasets, such as the petabytes of collision events generated by the LHC. As such, it is necessary to apply our anomaly detection algorithms multiple times on different subsets of collision events in order to avoid statistical fluctuations present, and verify the performance of these methods. This implies using multiple different training and testing sample sets in an appropriate manner which does not compromise the performance of the algorithm.



Figure 5.1: Diagram of k-folds cross-validation for $k = 5$. Each row represent the total number of simulated events for a dataset. We split the set into 5 equally distributed subsets, and take one subset to be the testing set (pink cells). The rest of the subsets are used as training samples (green cells) for the neural network.

A popular technique to address these issues is the k-folds cross-validation method which is visualised in Figure 5.1. The entire set of collision events is again split into testing and training data, however, in a specified way. We split the total sample set into k equally distributed subsets, each containing $\frac{1}{k}$ fraction of the entire set of events. In the first iteration, the four subsets are taken and used in their entirety as the training set, with the remaining subset used as the testing set. After we gather the performance measures such as the ROC curve and corresponding AUC stated in Section 4.2, we must repeat the process again, where we will now choose a different subset to be the testing subset. We repeat this action k times, until each subset has had the opportunity to be the testing sample set.

In this way, we may retrieve k results from the same algorithm, learned on different collision events. From this, we can aggregate the performances so that we may properly evaluate said algorithm. In this chapter, all result display the average or mean curve of the ROC plots, and AUC values. We may also display the standard deviation of these k ROC curves, and AUC scores as the scale of uncertainty that we expect. Unless otherwise stated, we choose $k = 5$ folds for this thesis.

Normalisation

The training of neural network can be erratic when comparing input variables of vastly different scales. In this work for example, the azimuthal angle, ϕ occupies the domain between $-\pi$ and π, whilst the transverse momentum, p_T of a jet can range from 100's into the 1000's GeV. Therefore, it is typical in situations to normalise the inputs of the jet. For any given kinematic variable or jet substructure variable, x we may normalise the distribution as follows,

$$x_{normalised} = \frac{x - \text{minimum}(x)}{\text{maximum}(x) - \text{minimum}(x)}. \tag{5.1}$$

This normalisation is also necessary as we recall in the autoencoder architecture that we recreate the inputted jet observables, and since the relevant activation functions have a typical range between 0 and 1, it is therefore absolutely necessary to apply this normalisation to ensure all jet observables, $x \in [0, 1]$.

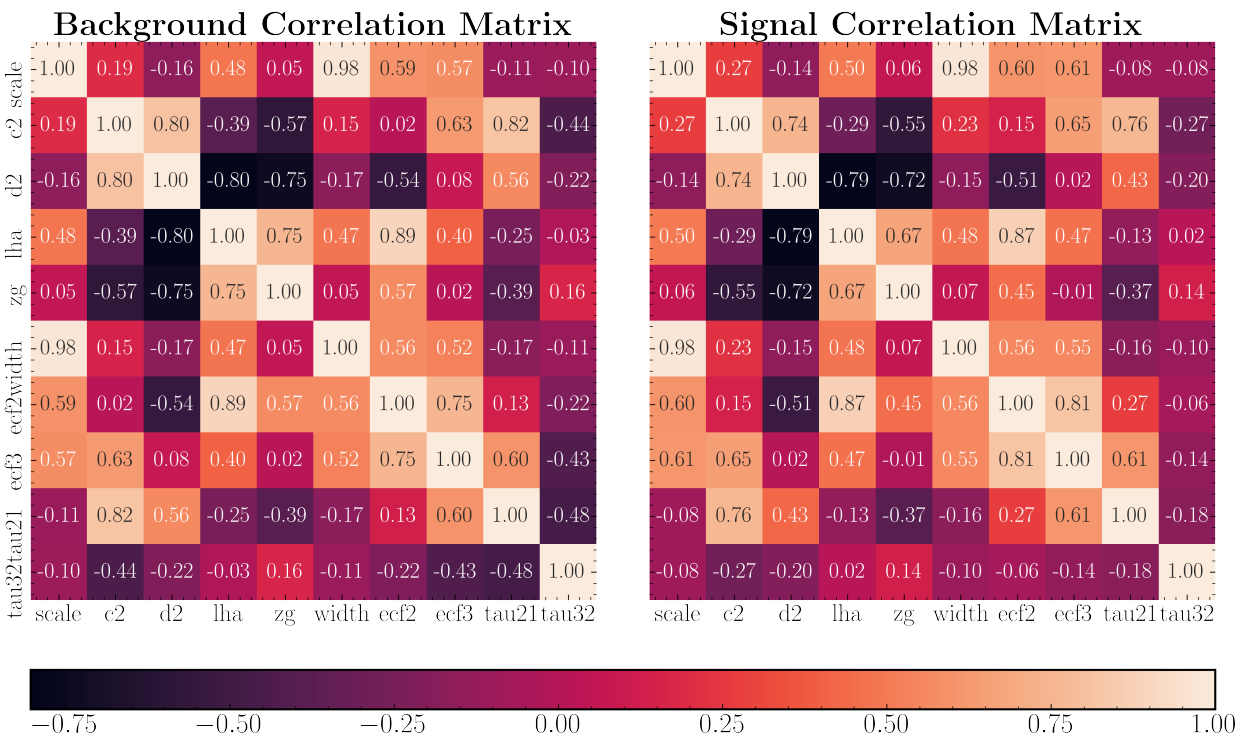


Figure 5.2: Linear correlation matrices for signal and background processes in the t-channel of semi-visible jets, using jet substructure observables. The colour bar indicates large negative values as tending towards black, and values closer to positive 1 approaching white. We look for uncorrelated observables where the linear correlation value is equal to 0 (red), in order to aid our analysis.

5.1 Semi-visible Jets

5.1.1 *t-channel*

Figure 5.2 displays the correlation matrices of jet substructure variables for the t-channel. We look for substructure distributions which are strongly uncorrelated with each other, as these variables combined can have strong discriminating power when we make distribution cuts on them. We see on the figure that *Les Houches Angularity (LHA)* and the N-subjettiness ratio, *tau32* (τ_{32}) have low correlation with each other. Contrary to this example, LHA and D_2 are strongly correlated with each other. This would imply that there is not much use in training on both these substructure variables, as little added information can be gained in order to discriminate between multijet background and semi-visible jet resonances.

It is worth noting that the two correlation matrices for both signal and background appear very similar to each other. That is, the linear correlation values for every distribution in both processes, have close values to one another. It has already been stated that semi-visible jets will be difficult to differentiate against hadronic jets originating from Standard Model physics, and so this occurrence between the matrices is not useful, but not entirely unexpected.

Latent Dimension z of Autoencoder and Variational Autoencoder

This section compares the results of the unsupervised anomaly detection algorithms, a standard autoencoder (AE), with a variational autoencoder (VAE), as well as commenting on the final state

$\frac{m}{p_T}$ **Bin**	**Benchmark NN AUC Score**
0.20–0.23	0.6461
0.23–0.26	0.63848
0.26–0.29	0.63520
Full Range	0.658

Table 5.1: Table highlighting AUC scores of a benchmark neural network trained on different scale bins

signatures of semi-visible jets in the t-channel.

A supervised deep neural network (NN) is trained to distinguish signal from multijet background events. This neural network consists of 10 layers, each layers consisting of 8 neurons. The Linear Rectifier (ReLu) activation function is used for each layer, except for the final layer which utilizes the sigmoidal activation function to ensure we achieve a result between 0 (background) and 1 (signal). We run train the network for 50 epochs. The results of the NN is simply used as a benchmark on which to compare the unsupervised algorithms. In an actual search, we would not know which simulated events contain signal and thus could not train a supervised algorithm.

There is a total of $\sim 24 \times 10^3$ simulated events for the t-channel process of this semi-visible jet model, after the relevant cuts as described in Section 3.3.1, have been made. For the implemented algorithms in this section, we train on the entire leading jet p_T range in order to highlight the wide applicability to different situations which may be encountered in an ATLAS analysis. However, we demonstrate that essential discriminating power is not lost by only training a benchmark neural network on different p_T bins and comparing these results to an NN trained on the entire p_T spectrum.

Table 5.1 highlights how little variation occurs of performance for differing scale bins and the full range. These results indicate that using the entire p_T range does not change our conclusions, compared to conclusions made by training on the specified p_T bins.

The AE and VAE have the same architecture except for the latent layer. We begin with 8 nodes in the input layer, and each 4 subsequent layers have one less neuron than the previous layer. After the latent layer, the autoencoders are reflected, where each subsequent layer now increases in node count until we reach 8 nodes in the output layer, for reconstruction. The latent layer, z is the smallest layer in the network, which compresses our event. We vary the number of nodes in this layer in order to see the effects on performance. Each layer in this network uses a sigmoid activation function — including the hidden layers — and we train for 50 epochs once more.

Figure 5.3 displays the ROC plots for the AE and VAE at varying latent layer lengths, $z = \{1, 2, 3, 4\}$. We see that the VAE performs worse than the AE for all latent dimension sizes. It's AUC score close to 0.52 indicate a classifier only slightly better than random. An autoencoder performing closer to the benchmark is surprising, as we expect a VAE to make improvements in the reconstructed output of the final layer, by constructing overlapping distributions in the latent space. This may point to an underlying failure of the VAE to correctly discern discriminating processes in the latent space, which leads to near uniform reconstruction of multijet and semi-visible jets substructures.

Focusing on the superior results of the standard autoencoder, varying the length of nodes in the latent space has a significant impact in indicating if an event is anomalous. $z = 3$ is optimal length whereby the network is trained on thousands of simulated events in order to suppress signal events and only accurately reconstruct background events. For lengths below $z = 3$, the events are too far

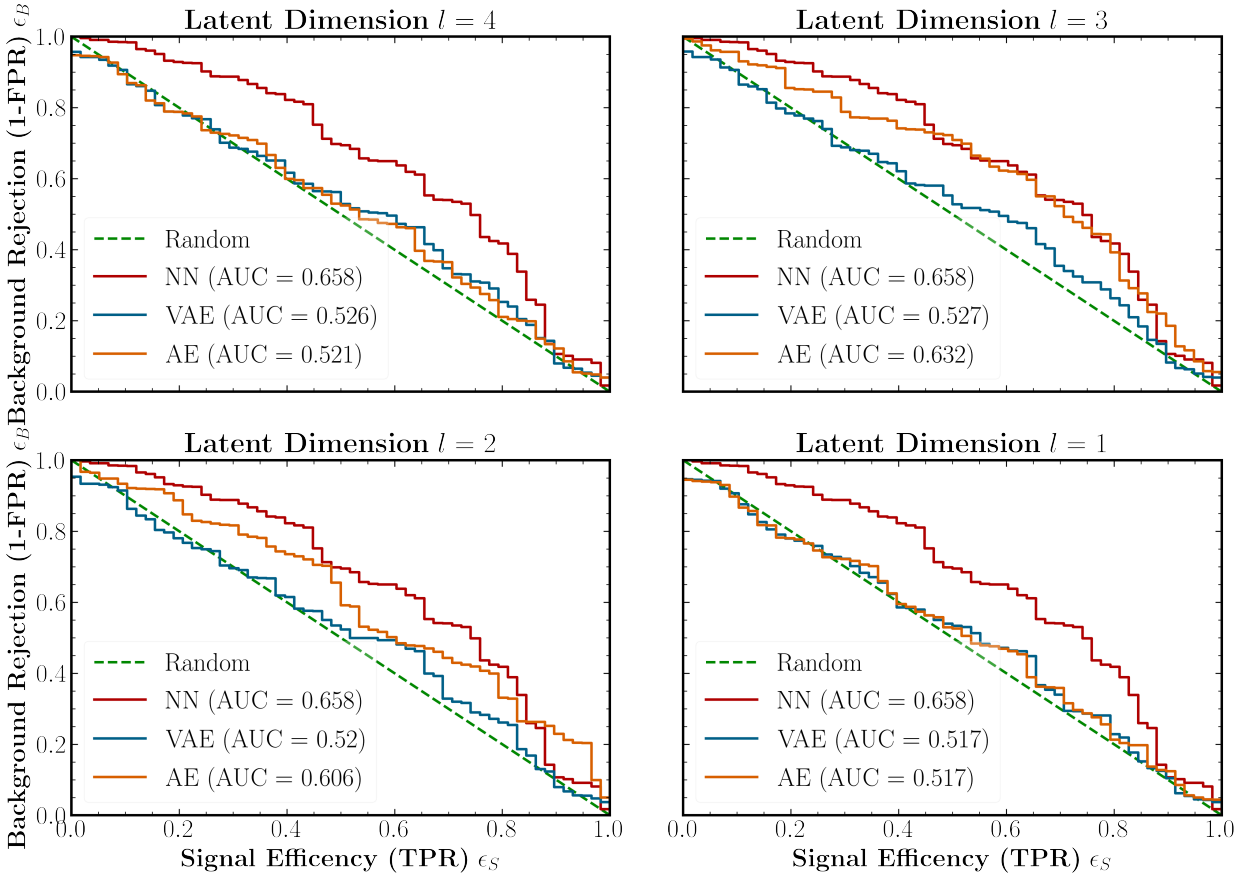


Figure 5.3: Comparison of AE and VAE ROC plots for various latent dimension lengths, $z = \{1, 2, 3, 4\}$. The dashed line indicates a random classifier which has a 50 % chance to classify an event as either signal or background, and therefore indicates the minimum baseline our algorithms should achieve. On the other end of the spectrum, the benchmark neural network (NN) (red) represents the maximum AUC score the unsupervised algorithms can achieve. ROC curves closer to the benchmark indicate a better classifier.

compressed to reconstruct any simulated event accurately, leading to a more random discriminator. Larger than $z = 3$, the network has too little compression and can thus easily reconstruct both multijet background as well as semi-visible jet substructures.

Figure 5.4 displays the MSE plots for the autoencoder (AE), which corresponds to the ROC results in Figure 5.3. We recall that the MSE is defined as the Euclidean distance, $|\bar{x} - \bar{x}'|$ where $\bar{x}$ is the inputted jet substructure and $\bar{x}'$ is the output reconstruction. We look for the optimal threshhold for the MSE where we can classify all simulated events lower than this value as background, and all event greater as indicative of semi-visible jets. However, in practice, this optimal threshold is cannot be known as we do not know what the labels of our events are (signal/background). So, Figure 5.4 should hope to display signal events tending to contain larger MSE score which is the case. Therefore, in these distributions, we hope to have all background events clustered around 0, whilst large MSE values contain signal. Thereby, a reasonably chosen threshold will not differ greatly in performance compared to the optimal threshold. We can see in these figures that this is not the case, leading to difficulty in choosing any threshold which gives a strong discriminator. This is the reason why the AUC scores are low. At $z = 3$, gives the best results, as we can a more

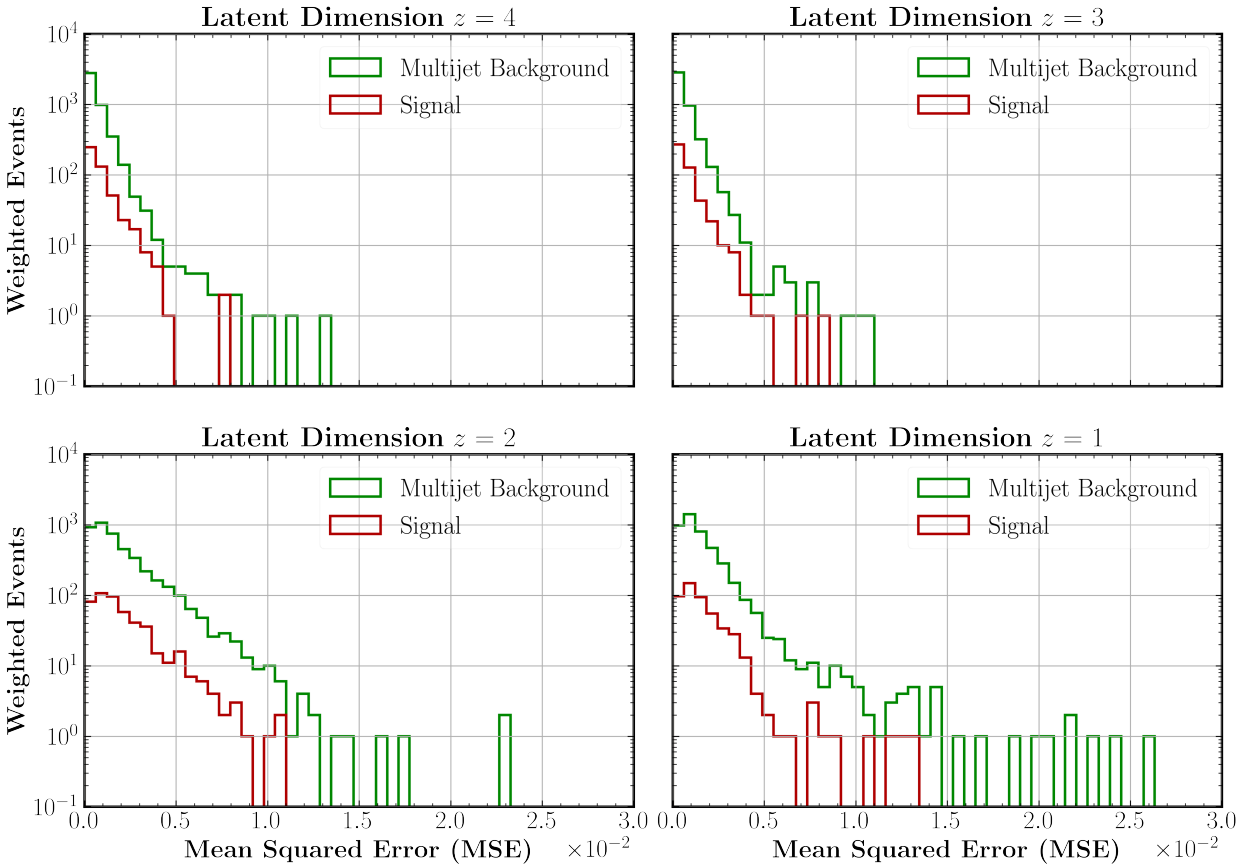


Figure 5.4: Comparison of autoencoder mean squared loss (MSE) plots for various latent dimension lengths, $z = \{1, 2, 3, 4\}$ corresponding to Figure 5.3. In the optimal classifier, we would like the background (green) to all be centred close to zero, and signal (red) be distributed at larger values of MSE.

significant spike in signal events at larger values of MSE.

We state that we did not investigate autoencoder architecture where the number of dimensions or nodes is increased, prior to reducing the number of dimensions in the latent layer. Such an architecture would expect to inflate the jet event representation, which may make the network more easily compress the event and lead to better reconstruction for multijet background.

It is essential to point out here that, although the benchmark network is supervised, the classifier learnt performs somewhat badly with an AUC score equal to 0.632. This highlights a significant difficulty in discriminating semi-visible jets from ordinary hadronic jets. We may say that even if the standard autoencoder can approach the benchmark's performance, the function may still not be suitable to aid in a search or improve significance.

5.1.2 *s-channel*

In the s-channel, we explore the performances of Classification without Label (CWoLa), Tag N'Train (TNT) and DGCNN, where their explanations are given in Sections 4.5, 4.6 and 4.7 respectively. As the s-channel does contain a clear signal resonance, we utilise the leading and sub-leading p_T jets in an event as the focused training samples. As explained in Chapter 4.1, these algorithms require these event properties in order to implement correctly.

In total, 9×10^4 dijet events were produced for this analysis. Similar to what was stated for the t-channel results, we utilise the entire range of the leading p_T when training the algorithms. We ensure that this will give similar results to training on specified bins by again comparing AUC scores of the benchmark NN trained on different scale ranges. Bins 0.25–0.45 and 0.45–0.65 gives an AUC scores of 0.949623 and 0.946912 respectively. Comparatively, training on the entire range gives a similar score of 0.9661.

CWoLa

In order for the conditions of CWoLa to be met, we require two sample sets; one which contains signal-rich events, and another which contains background-rich events. As we are dealing with the s-channel which has a strong resonance in the invariant mass distribution, we can define the signal and sideband regions, which generate the two sets required. It is important to highlight that although we define these sets based on thresholds to the m_{jj} distribution, it is essential that we do not actually use this observable or any other substructure observable highly correlated with the invariant mass, during training. We reiterate the need for what was stated in Section 3.3.1 on how the final event weights are calculated. The event weights are a particularly essential part of the CWoLa algorithm as it relies heavily on the concentrate of signal and background events, which are weighted.

This section explores the importance of defining an appropriate signal region that splits the events appropriate into signal/background rich sets. m_{res} represents a point in the invariant mass distribution where we are at the center of a proposed SR, while m_{width} defines half the width of this SR. Both these parameters define what constructs the signal region, and therefore what composes the sideband region. For example, $m_{res} = 1000$ GeV and $m_{width} = 150$ GeV implies that the signal region is all invariant mass events between 850 GeV and 1150 GeV ($850 < m_{jj} < 1150$), while the sideband region covers all event with $m_{jj} < 850$ GeV and $m_{jj} > 1150$ GeV. The supervised deep neural network is trained to differentiate SR events from SB events in the absence of the invariant mass, where the NN is composed of 10 hidden layers, each consisting of 18 nodes, before a single node activation in the output layer. We recall that this learnt NN approaches the optimal classifier that discriminates semi-visible jet events from background events.

Figure 5.5 displays the AUC scores of CWoLa as we vary the signal region window. This is done by setting a constant width, $m_{width} = 100$ GeV, and varying the m_{res} from 200 GeV to 3500 GeV in steps of 100 GeV. Well-chosen signal regions result in a high AUC score, as the concentration of signal events in the signal-rich sets are significantly larger than background-rich sample sets. As stated in Section 4.5, the strength of the learnt neural network to differentiate between signal and background events, is proportional to the percentage of signal events in the signal-rich sample sets. Thus, the more semi-visible jet events contained within the SR will result in a stronger classifier. As we move to larger invariant mass windows, the SR begins to contain effectively no signal, whilst the sideband region contains almost all signal events. This leads to AUC scores below 0.5, as the NN has learnt the inverse solution. That is, most background is tagged as signal, and vice versa. We note that in for $m_{res} = \{700, 800, 900\}$, the fraction signal events is largest in the signal region compared to the sidebands region, and thus has a large AUC score — significantly larger that what is capable from autoencoders. We may also notice that there is a second spike at 1700 GeV in Figure 5.5. This can be explained as the multijet background distribution tails off around this region at a significantly faster rate than our signal distribution. So, the region around 1700 GeV represents a second maximum of signal to background ratio in the signal region, compared to the first maximum is around 800 GeV.

An overview of the relationship between various chosen jet substructure observables to the jet

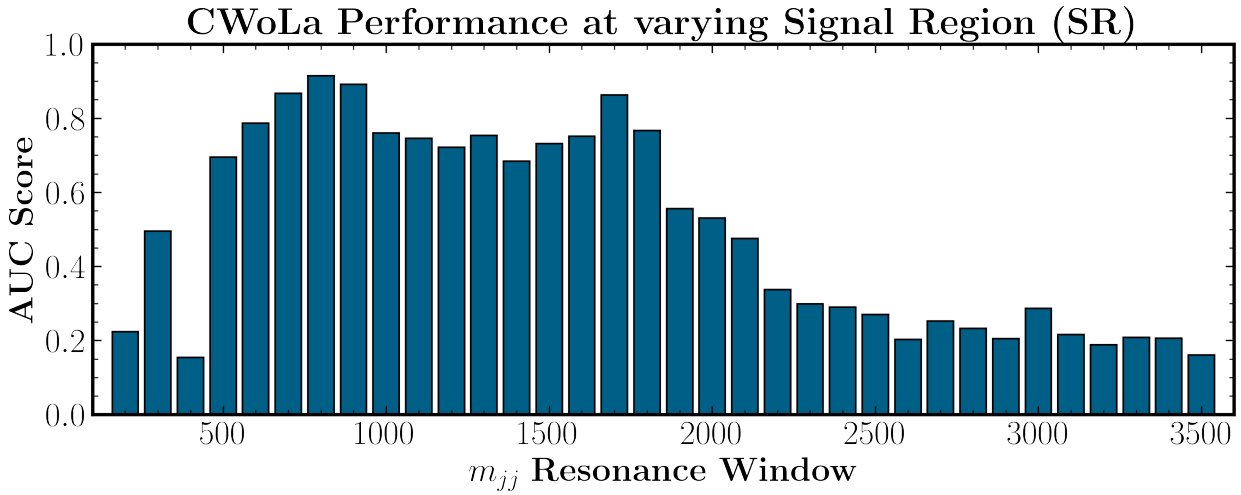


Figure 5.5: AUC Scores of the Classification without Labels (CWoLa) as function of varying the m_{jj} signal region window. We set $m_{width} = 100$ GeV, and vary the m_{res} from 200 GeV to 3500 GeV in steps of 100 GeV. High AUC scores around $m_{res} = 800$ GeV are expected as the fraction of signal in the SR is the largest there.

invariant mass for the leading and sub-leading p_T jets is shown in Figure 5.6. We highlight that although m_{jj} was not directly used during training, the CWoLa selection displays a remarkable correlation with the particle level plots on the invariant mass axis. This displays CWoLa's ability to use auxiliary variables such the semi-visible jet substructure in order to discern signal events. In all appropriate comparisons of the scatter plots in Figure 5.6, the scatter points of signal events and CWoLa selected are in similar regions, which results in the high AUC score.

Tag N'Train

We take the two leading p_T jets within an event to be used for Tag N'Train classification. For the initial weak unsupervised classifier, a standard autoencoder with varying latent dimension, z is used. The two jets are independently trained on separate AE's to produce two different sets of signal and background-rich events. These two sets meet the requirements for CWoLa and thus we train two supervised neural network classifiers with signal/background indications from the AE. In this supervised training section, the jet substructure observables from the jet has not been utilised, is now used for training. In other words, an initial autoencoder which uses the leading jet substructure distribution, will now us the sub-leading jet for the supervised classifier and vice versa, as per the steps detailed in 4.6. Thus, we produce two improved classifiers which are both utilise all aspects of the dijet events

The autoencoders are similar in architecture to the AE explored in Section 5.1.1, while supervised neural networks consists of 10 layers with 9 neurons each. We train each architecture for 100 epochs.

Figure 5.7 displays the results of a good performance with the latent dimension of the AE, $z = 3$. The MSE plots for both the leading and sub-leading p_T both show indicate above-average separation of signal and background distributions. However, the vast majority of signal events are still 'hidden' in the region closer to 0, where background events are dominant. In this situation, much like in the t-channel, AE's do not provide enough a strong indication of the presence of signal, compared to the thousands of background events surrounding it.

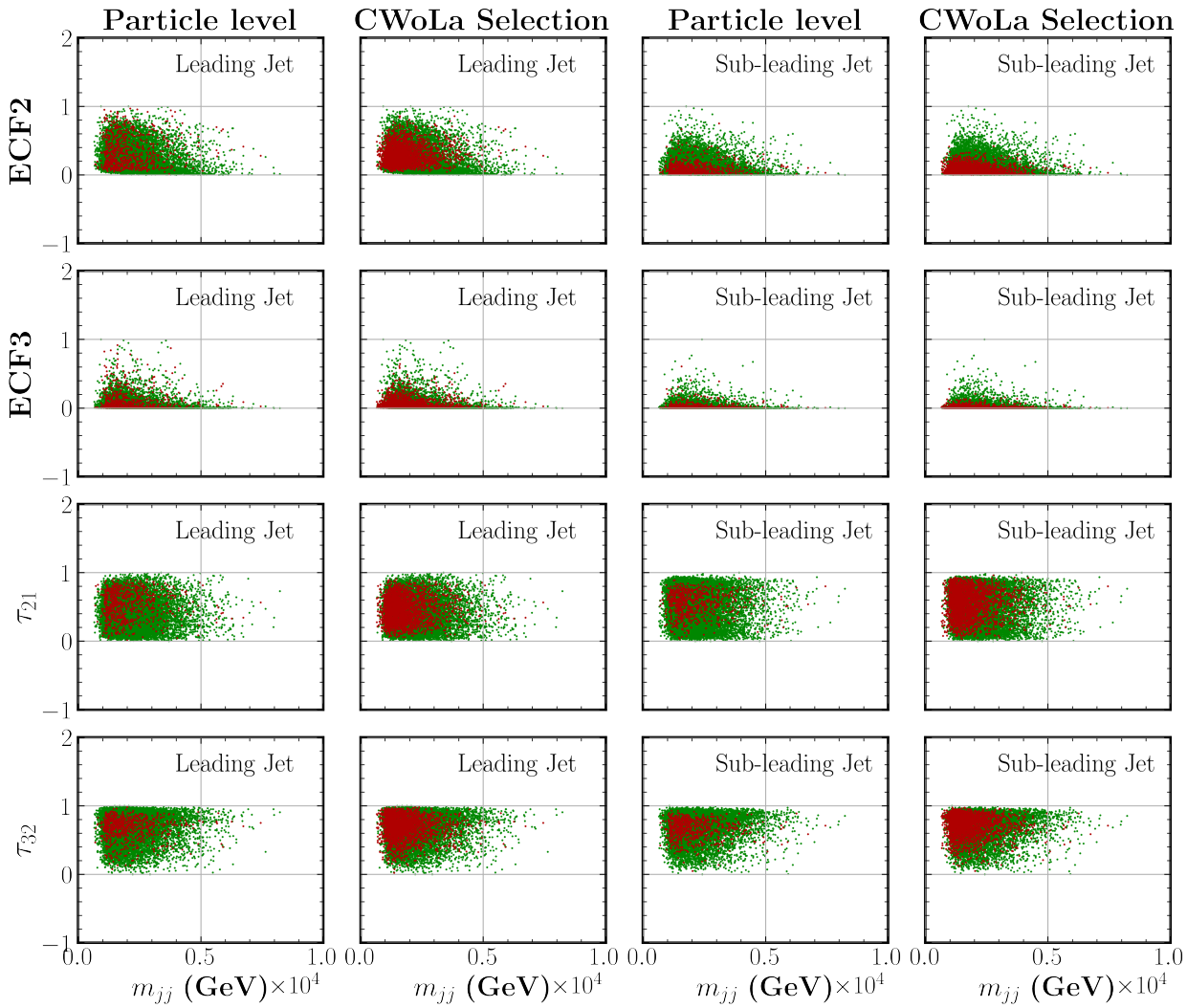


Figure 5.6: Scatter plots of jet substructure variables against the jet invariant mass for the two leading p_T jets. The signal region window for CWoLa has parameters, $m_{res} = 800$ GeV and width, $m_{width} = 100$ GeV, giving a high AUC value of 0.893. Green scatter points indicate multijet background, while red scatter points indicate particle-level/CWoLa selection events in the respective columns. Along the m_{jj} axis, CWoLa locates signal in the correct regions despite only using auxiliary jet substructure observables.

The second row of Figure 5.7 shows the values of the output layer of the supervised neural network, trained on the signal and background-rich samples generated by the autoencoders. We can see clearly that this generates distributions that make it far easier to separate events which contain semi-visible jets from ordinary jets from SM processes. It is clear in the leading jet activation function plot that we may define a threshold near 0.6, where the majority of events above the threshold will contain signal. However, the sub-leading jet shows little improvement from the MSE plot, as we can see there is only a small amount of separation power between the two distributions.

The bottom row displays the ROC curves of the weak unsupervised autoencoder, and the respective improved supervised discriminator. As we have indicated, utilising the sub-leading p_T jet's substructure for leading jet driven AE, shows great improvement in performance. In comparison,

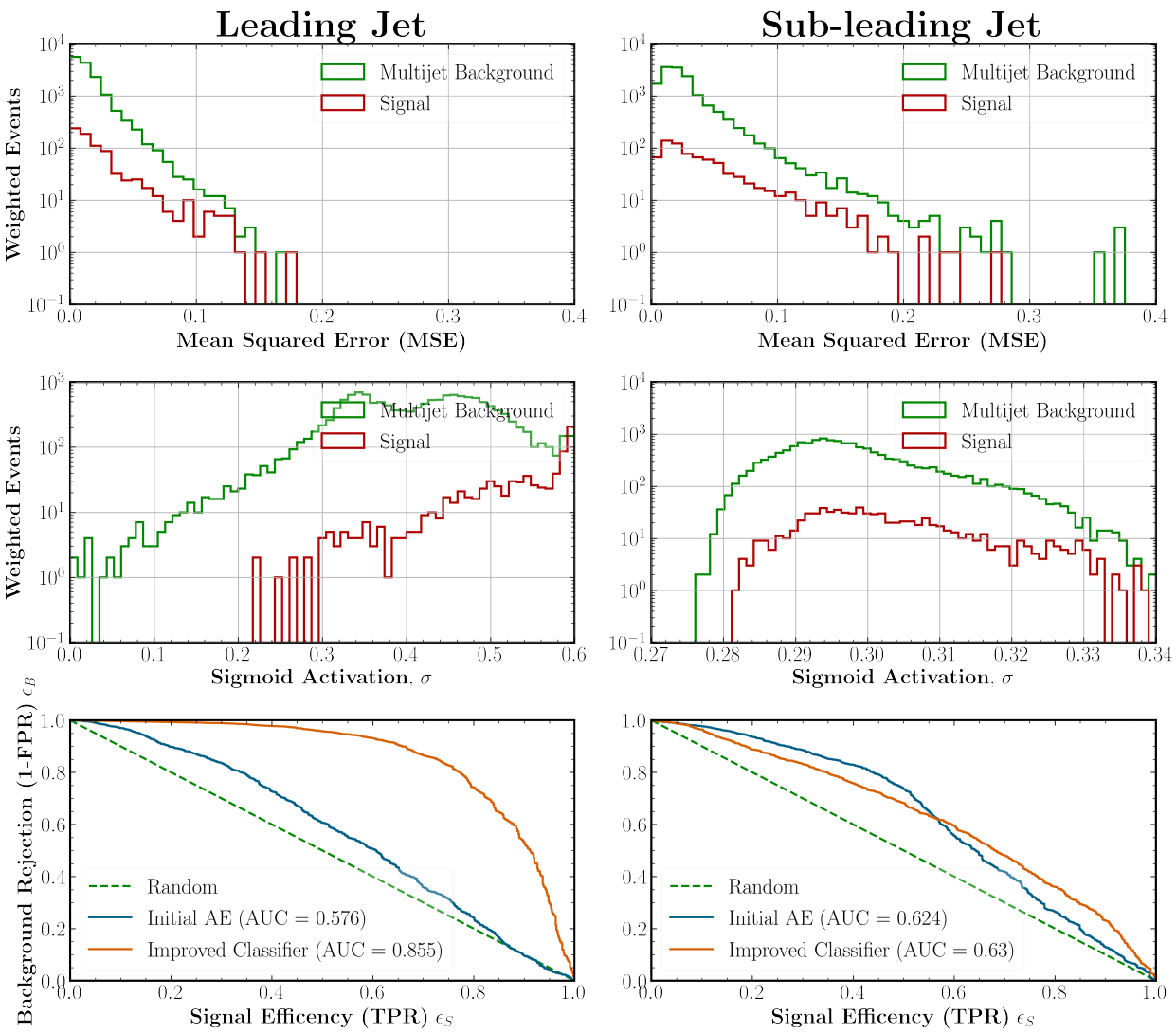


Figure 5.7: Comparison of Tag N'Train using latent dimension size, $z = 3$ for the initial autoencoder. The first row displays MSE plots for the weak unsupervised classifier (autoencoder). The middle row indicates the final values of the output activation layer of the supervised neural network, trained on the results given by the autoencoders. The final row displays ROC curves comparing the initial weak classifier (blue line) with it's corresponding improvement (orange line) via supervised training signal/background-rich sample sets generated by the autoencoders.

the sub-leading jet did not gain much discriminating power using the leading jet's substructure observables. This may be due a poor selection of the initial AE, which can have a large impact on results.

Figure 5.8 is similar in explanation to Figure 5.7, but the results are obtained by setting the autoencoder's latent size to $z = 1$. As was noted before, this leads to poor discriminating power, and we can see this in the MSE histograms of the initial AE's. It is because this latent space length cannot provide signal and background-rich samples, the supervised neural networks trained on these samples make separation signal and background even more difficult, as indicated by the activation function histograms. The ROC curves here highlight the final performances of such a

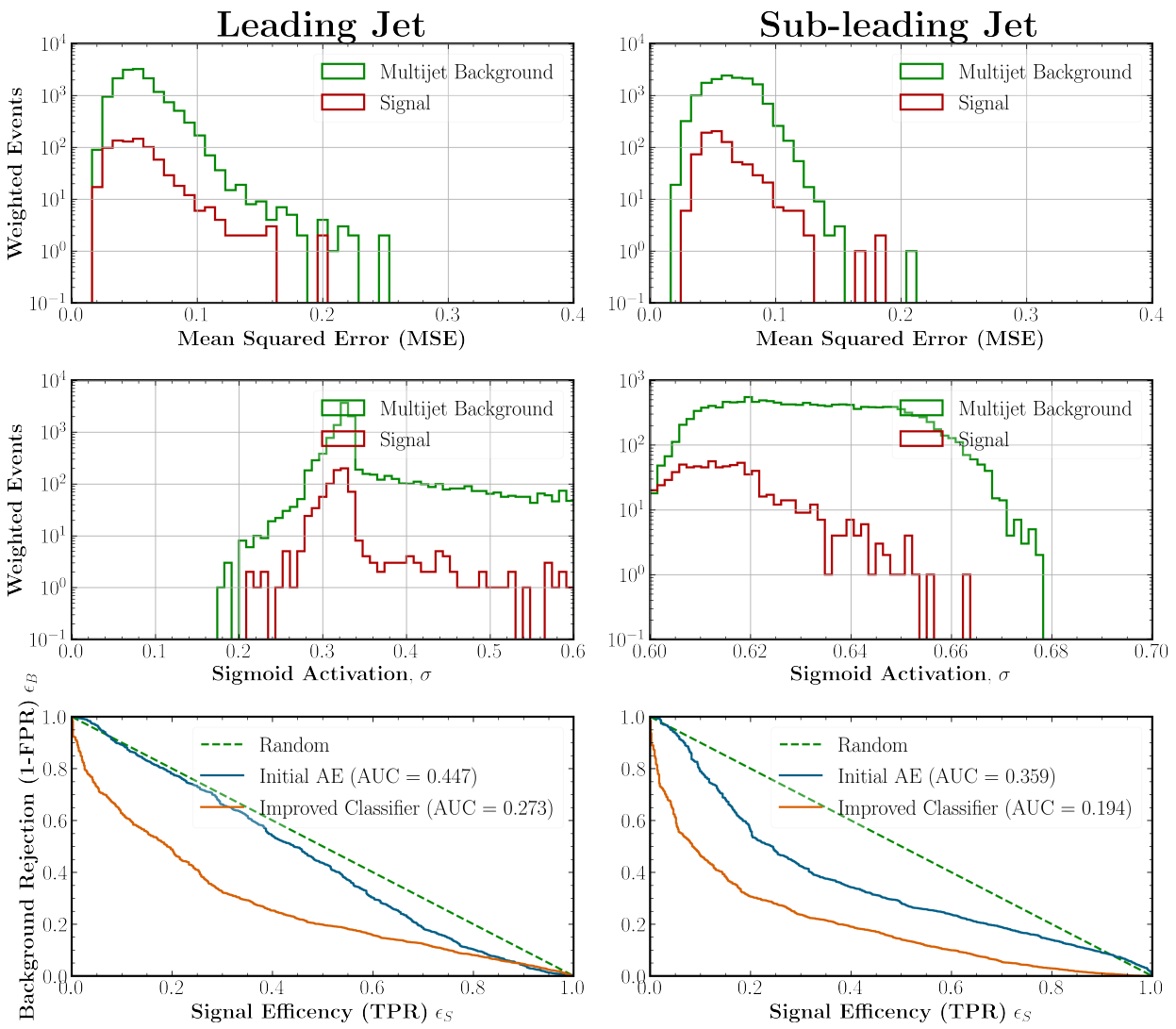


Figure 5.8: Comparison of Tag N'Train using latent dimension size, $z = 1$ for the initial autoencoder. A poor degree of separation in the initial MSE histograms leads to even worse results by the supervised discriminators.

situation, where a poorly run AE autoencoder results in even worse performance by the supervised classifier, as signal/background rich samples have been adequately generated

Figure 5.9 are two scatter plots that compares the leading and sub-leading jet scores derived from the results of the final activation layer of the supervised network. When $z = 3$, events containing semi-visible jets are clustered around the right side, while the majority of background events are located closer to 0. We recall that signal events should have activation function values close to 1 in the ideal classifier, and so we see this in the scatter plot. However, the opposite occurs in the situation where $z = 1$, as most signal located in the bottom left corner of the scatter plot, indicate leading and sub-leading scores closer to 0, which is the opposite of we hope occurs. This inverse is the reason why Figure 5.8 contains AUC scores below 0.5.

Overall, we hope to demonstrate how Tag N'Train has potential be utilised as a strong tagger for dijet events, but is very volatile to poor initial autoencoder results, which will have dramatic

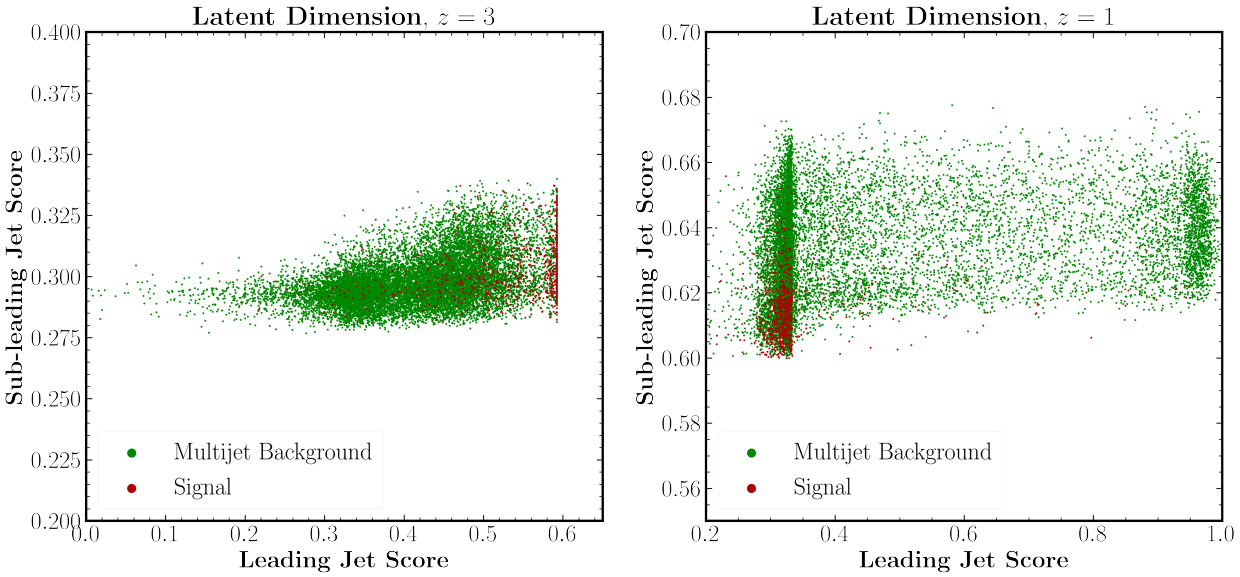


Figure 5.9: Scatter plots of sigmoid activation scores for the leading and sub-leading p_T jets in the Tag N'Train algorithm. When the latent dimension, $z = 3$, there is a strong separation between signal and background events. At $z = 1$, most events that contain semi-visible jets are described as background, while the cluster of background events closer to the leading jet's score of 1 are misleading labelled as signal.

influence on overall performance.

We compare the performances of each of the algorithms in Figure 5.10. We can note that in the s-channel, the benchmarked neural network has a significantly larger AUC score of 0.966. This may indicate that it is easier to discriminate events which contain semi-visible jets from multijet background, as the t-channel results are poor in comparison. The standard autoencoder now performs adequately above the random classifier, and we highlight that both Tag N'Train and CWoLa approach the supervised benchmark levels of performance, despite being wholly unsupervised. It is clear that if the signal region is chosen correctly so as to emphasise the final state resonance of semi-visible jets, it provides a powerful discriminator against background.

5.2 Dark Jets

As detailed in Section 3.2, the models A, B, C and D at different mass peaks (1000, 1500, 2000, 2500 GeV) and after detector simulation effects (detector-level) have been taken into account, are investigated here. This section displays the uses that various anomaly detection algorithms may have in differentiating jets made by Standard Model processes against dark jets created by BSM physics which decayed back in a visible SM final state.

Similar to the reasoning in Section 5.1, we consider a p_T bin in comparison to the entire spectrum to ensure they produce similar results. Figure 5.11 demonstrates that the AUC scores are similar for Model A at its respective mass peaks. The dashed lines indicate the benchmark neural netowrk trained on p_T bins, 200–400 GeV, 400–600 GeV, 600–800 GeV and 800–1000 GeV for mass peaks 1000 GeV, 1500 GeV, 2000 GeV and 2500 GeV respectively. The close AUC score indicate that training on the entire p_T spectrum for these dark jet models should perform similarly to models trained on these enclosed bins.

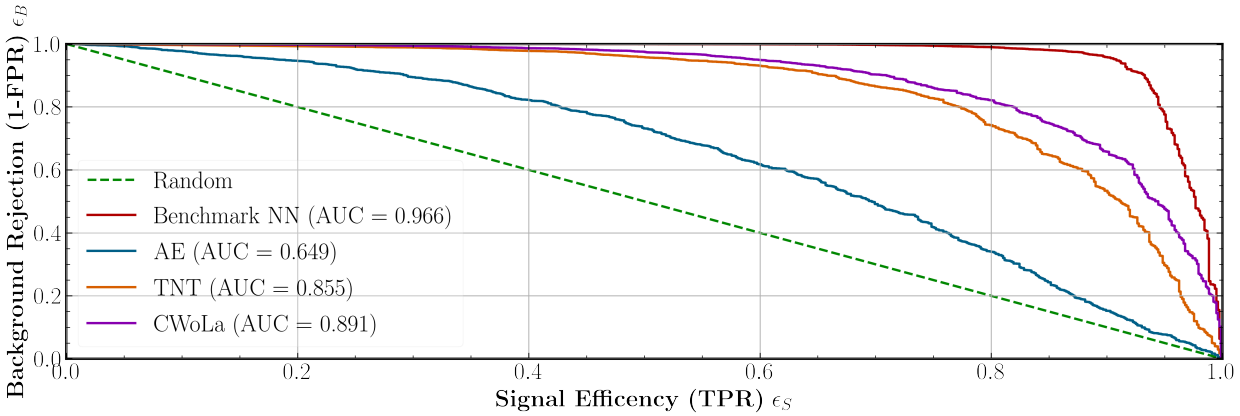


Figure 5.10: ROC curve highlighting various anomaly detection algorithms for semi-visible s-channel production by utilising jet substructure observables. The Tag N'Train curve displays the leading jet's improved classifier from an initial autoencoder woth latent dimension, $z = 3$. The CWoLa curve is generated with choosing SR with $m_{res} = 900$ GeV. Overall performance of techniques (including the supervised benchmark) is improved from the t-channel process events.

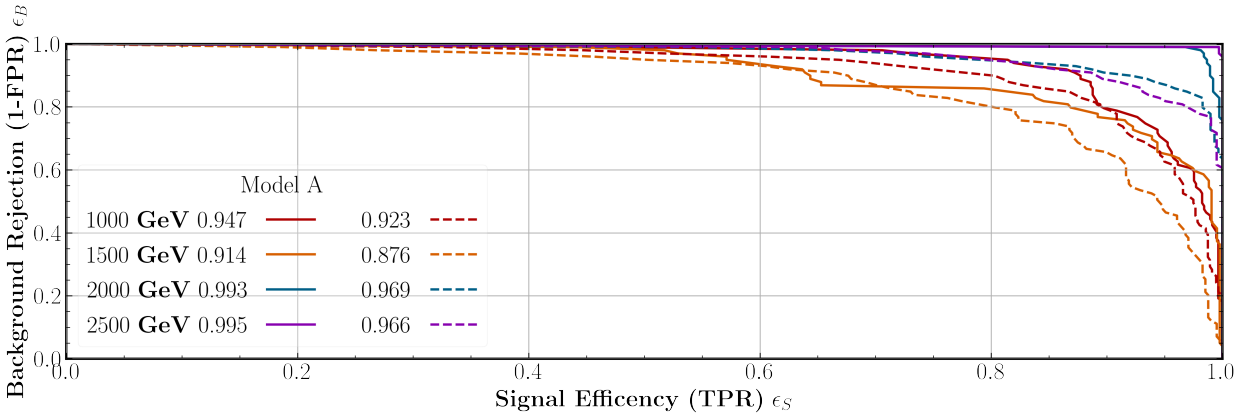
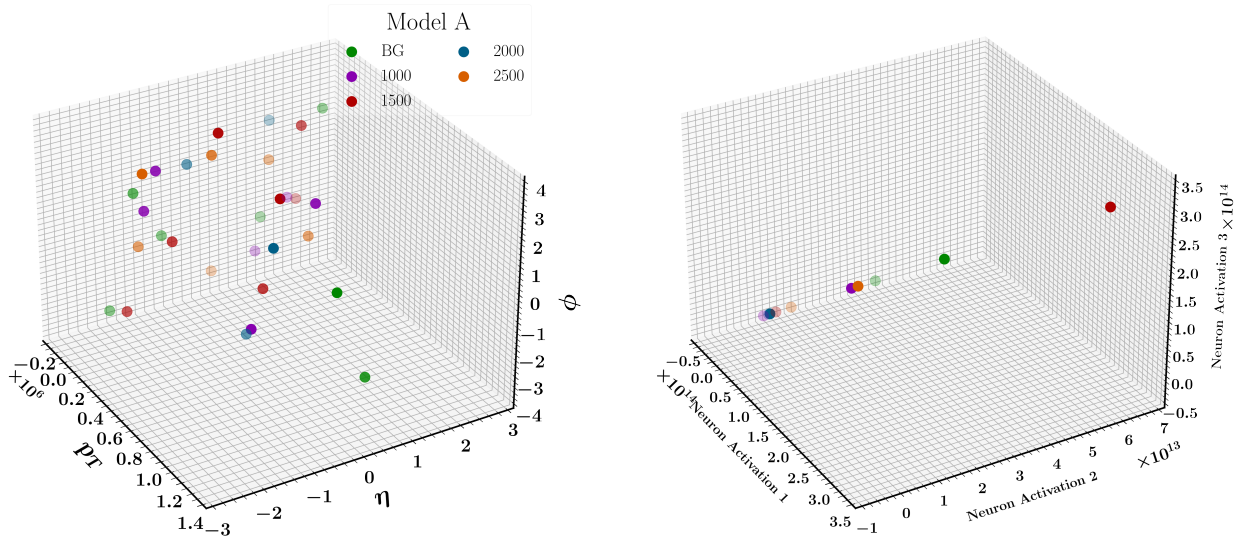


Figure 5.11: ROC curves for a benchmark neural network trained on the p_T range (solid lines) and specified p_T bins (dashed lines). AUC scores are indicated in the legend. Performance at the same mass peaks is similar.

5.2.1 New Benchmarks

It is worth improving the supervised benchmark which is currently a standard deep neural network. The need for this is two-fold. Firstly, better benchmark encourage us to look for improved anomaly detection algorithms as we are motivated to reach a higher ceiling. Secondly, the new different architecture employed on the supervised neural network may indicate the need for an unsupervised expansion which utilise the same concepts. For example, aspects of a combination layer be added to anomaly detection algorithms already mention here to provide better result.

In this section, we compare how a supervised deep graph convolutional neural network (DGCNN) and a standard deep neural network with an initial combination and Lorentz layer compare with a fully-connected neural network composed of no unique layers.



(a) Plot of example events in the jet's p_T-ϕ-η plane. Green points display a single example background event which contains 5 jets depicted on the plot. The other various colours depict jets generated by the Model A process at mass peaks, (1000, 1500, 2000, 2500 GeV). It is these jets which we send to the initial input layer of the *EdgeConv* layer.

(b) Graph displaying the feature space of the jets in Figure 5.12a after processing by one *EdgeConv* layer. This layer transforms every collection of jets in an event into a new jet representation, which is two jets, each with 20 new variables. The scatter plot depicts three of these new represented features, but we expect background and Model A to be more easily separated at higher dimensions than 3.

Figure 5.12: 3-dimensional scatter plots of jet feature representation before and after the *EdgeConv* layer.

Supervised Deep Graph Convolutional Neural Network

The DGCNN implemented is composed of several layers on process. Firstly, the initial set of jets within an event are taken and the pairwise Euclidean distance is calculated for every jet to each other, using the corresponding jet substructure observables. This is used to find the $K-$nearest neighbour of each jet, where we set $K = 3$. The edge features of each jet are then calculated, and are moved through a layer of perceptrons. We choose the aggregation function to be the maximum operator, and output 2 new feature representation of the jet's substructure. This defines the *EdgeConv* layer and we repeat this layer three times before adding a fully-connected section of 10 layers. The final output layer is a single sigmoidal neuron which will indicate the presence of signal.

Figures 5.12a and 5.12b display the jet distributions at two layers of the DGCNN. In Figure 5.12a, the jet's kinematic variables are plotted in three dimensions, where these variables move into the first inputted layer of *EdgeConv*. Figure 5.12b displays the representation of the jet after a single edge feature convolution, where each point in the 3-dimensional plot is newly constructed 'jet' in a different feature space. We expect this new representation to help us easily distinguish between different dark jet models against multijet background. The plot can only show two jets per event in three dimensional space, but there are 20 new 'jet substructure' representational dimensions. It is in this dimensional space that we may easily separate jets constructed by different initial process, such as dark jets.

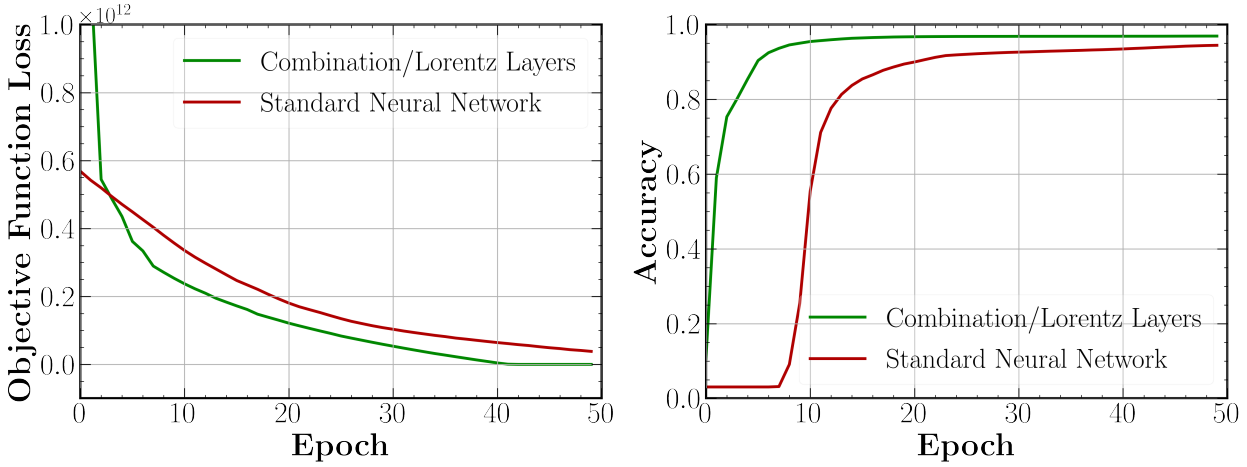


Figure 5.13: Objective function loss curve (left) and accuracy curve (right) per the number of epochs completed, for training on Model A at mass peak, 1500 GeV. The graphs compare a fully-connect NN and one aided by the use of combination and Lorentz layers. Both objective functions decrease as expected, however, the Lola/Cola network reaches the optimal minima in less epochs required.

Combination/Lorentz Layer Network

We expect the addition of a combination and Lorentz layer to slightly improve performance on a standard neural network, as the architecture of the layers more readily take advantage of the physical processes that create the final states we see in jets in events. A combination layer is constructed so that the learnable matrix, C_{ij} is 4×5 dimensions, where we utilise at least three jets per event and the remaining two are combinations of these, which together are weighted by C_{ij}. The output of this layer is then fed into a Lorentz layer, where extracts the essential physics-based discernible information of the event which can be used for classification. Following this, we implement 10 fully connected hidden layers with a single neuron activation as our output layer.

Figure 5.13 highlights the efficiency of adding the combination and Lorentz layers to a standard fully connected network. Comparing the addition of these layers, we see that although the objective function loss initially begins at a larger value, it decreases significantly faster than the ordinary deep neural network as the layers aid in capturing the jets' physics more readily. Both loss curve decrease along the epoch count as expected, but the combination/Lorentz layer reaches the optimal minimum value far more quickly. Similarly, the accuracy by epoch count curve in Figure 5.13 show that these physics-based layer reach the highest accuracy much earlier than a standard architecture network, although it is worth noting that accuracy value for imbalanced sets such as what we deal with in dark jets is not a viable measure of performance. It is clear that with enough epochs, the standard NN should approach the same minimum loss and give similar results to a network assisted by the combination and Lorentz layers. However, the number of epochs required to match this performance may be realistically infeasible in terms of computing resources and time.

In Figure 5.14, we depict the ROC curves of a supervised fully-connected neural network, the DGCNN and combination/Lorentz layer aided network described throughout this section, where we train for Model A at mass resonance, 1500 GeV against multijet background processes. All three algorithms perform at similarly high levels with both DGCNN and LoLa/Cola networks at effectively equal results. Although supervised, the slight improvements to these networks indicate more possibilities available for us to represent dark jets within deep neural network architectures

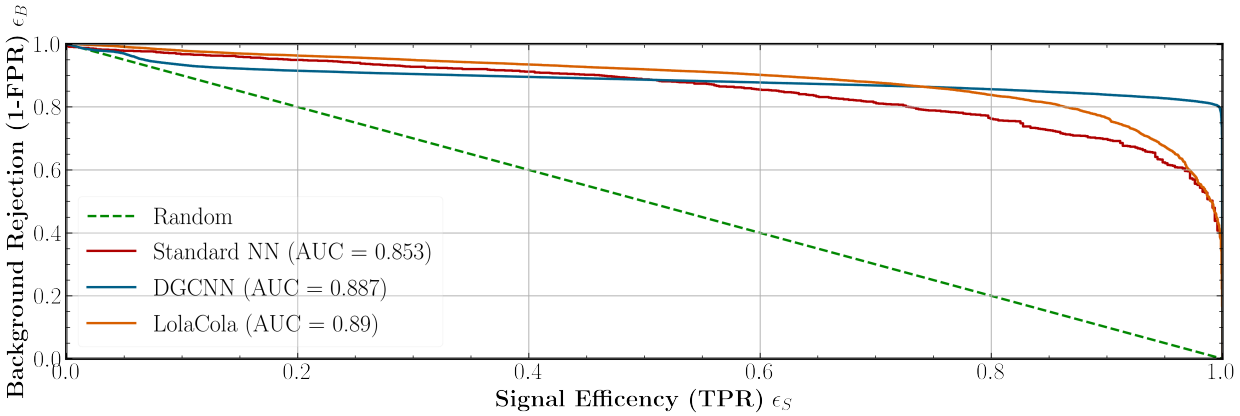


Figure 5.14: ROC curve results for a standard neural network, DGCNN as well as a neural network with combination and Lorentz layer (Lola/Cola) network for Model A at mass peak , 1500 GeV. Both non-standard architecture give slightly improved results and are similarly comparable.

which more easily capture the underlying physics taking place here, compared to more generalised methods. In future, we may consider finding ways using these algorithms for unsupervised use in order to improve current algorithms.

5.2.2 Comparisons on utilising n_{trk}

In this section, we investigate dark jet models A, B, C and D at mass peaks 1000, 1500, 2000, 2500 GeV utilising all the anomaly detection methods we have introduced. Importantly, the high number of tracks, n_{trk} within a jet, is thought to be a strong indicator of the presence of dark jet, and so we test our methods both with and without utilising the n_{trk} observable. In this study, the matching of tracks is done via the ghost-matching technique [30]. The full list of the models evaluated with corresponding AUC scores is displayed in Table 5.2. We visualised the effect of adding the number of tracks through ROC curves visualised in Figures 5.15, 5.16, 5.17 and 5.18, corresponding to models A, B, C and D respectively. We mention that both the standard autoencoder and Tag N'Train use a latent dimension, $z = 10$, and the signal region for CWoLa is formed by using the model's corresponding resonant mass indicated, with a width, $m_{width} = 100$ GeV.

The benchmark results represent the peak performance attainable in each of the models. As shown in the semi-visible jet section, the autoencoder performs the worst of the three anomaly detection methods, while CWoLa results in the best performance. This is seen the mean AUC score row (highlighted in green) of each model in Table 5.2. We highlight the powerful use of CWoLa, as the averages show performance close to the optimal benchmark performance, despite being unsupervised. CWoLa is successful enough to be used in ATLAS analyses [81], and it is shown here that it is also capable of utilising jet substructure observables to predict the presence of dark jet.

We can also see from the mean AUC scores in Table 5.2 that there is a clear increase in performance by adding the number of tracks, n_{trk} to the list of variables used for training the anomaly detection algorithm. In almost every case with every model, the AUC score increased when the track multiplicity observable was introduced, with the only exception to this being rare erratic results for Tag N'Train and CWoLa. We can visualise this effect by considering the ROC curves for all four models, with solid lines indicating no use of n_{trk}, and dashed lines utilising the track multiplicity. In most cases, the dashed lines perform better. This indicates that a jet's n_{trk}

Model	Mass Peak (GeV)	Benchmark		Autoencoder		Tag N'Train		CWoLa	
		No n_{trk}	n_{trk}	No n_{trk}	n_{trk}	No n_{trk}	n_{trk}	No n_{trk}	n_{trk}
Model A	1000	0.7928	0.9652	0.6171	0.7629	0.4692	0.5242	0.5000	0.5000
	1500	0.9584	0.9582	0.6577	0.8770	0.9364	0.5396	0.9190	0.9489
	2000	0.9975	0.9978	0.7656	0.9422	0.5389	0.5437	0.9827	0.5000
	2500	0.9994	0.9994	0.7967	0.9650	0.4766	0.9418	0.8764	0.5000
	Mean AUC	0.9163	0.9671	0.6802	0.8607	0.6482	0.5358	0.8006	0.6496
Model B	1000	0.8133	0.7381	0.4504	0.6007	0.7756	0.6565	0.7809	0.6125
	1500	0.9642	0.9593	0.6085	0.7938	0.9654	0.7937	0.8882	0.9426
	2000	0.9911	0.9893	0.7117	0.9046	0.1682	0.7129	0.9900	0.9732
	2500	0.9977	0.9987	0.7982	0.9498	0.5931	0.1748	0.9969	0.9973
	Mean AUC	0.9413	0.9172	0.6422	0.8122	0.6256	0.5845	0.9143	0.8856
Model C	1000	0.7004	0.8961	0.4439	0.5157	0.5629	0.5192	0.5036	0.8488
	1500	0.9100	0.9290	0.5873	0.7492	0.6440	0.7813	0.7740	0.9219
	2000	0.9727	0.9918	0.6741	0.8477	0.7613	0.6419	0.9520	0.9684
	2500	0.9986	0.9975	0.7645	0.9399	0.9821	0.9400	0.9928	0.9837
	Mean AUC	0.8563	0.9536	0.6174	0.7631	0.7376	0.7206	0.8448	0.9307
Model D	1000	0.8562	0.9721	0.4185	0.6634	0.7262	0.6250	0.4961	0.4177
	1500	0.9378	0.9866	0.5220	0.7025	0.8611	0.6017	0.9389	0.9575
	2000	0.9901	0.9886	0.6691	0.9093	0.9623	0.3383	0.9861	0.9711
	2500	0.9964	0.9967	0.7691	0.9798	0.9592	0.3488	0.9953	0.9921
	Mean AUC	0.9401	0.9860	0.5947	0.8138	0.8772	0.4785	0.8541	0.8346

Table 5.2: Display of AUC scores for various algorithms applied to different models. The green rows highlight the mean scores for each model corresponding to each method. From this table, it is clear that the use of n_{trk} does aide in discriminating signal.

is a useful discriminant in finding dark jets. We expected this result as we know that the track multiplicity is closely related to dark meson decay which produces dark jets. The processes which correspond to n_{trk} numbers for gluon and quark jets are different to the dark portal interaction, and so it was expected the track multiplicity would be different between dark jets and multijet background.

We may also note a weak trend in the mass resonances of all these models, in that performance generally increases as we move from 1000 GeV to 2500 GeV which indicate that heavier particles are usually easier to discriminate from multijet background. Large radius jets are the result of dark matter decay product resonances, but are also formed here by backgrounds of lepton and quark jets. Heavier resonances may result in more unique jet substructure to dark jets, leading to an easier distinction between every model with mass resonance around 2500 GeV, and multijet background.

Across all models, we see no significant differences between the average performances. This highlights the general applicability of neural network based anomaly detection, as each model uses different decay modes and masses of π_d. In some ways this is expected, as we are not classifying the underlying processes that each model contains against SM hadronic jets, but rather alerting us whether a given event contains anomalous dark jets. Simply put, these algorithms notify us of anomalous events, but not in which way they are anomalous.

A few below 0.5 AUC values for the Tag N'Train method and values near random (AUC = 0.5) for CWoLa highlight the erratic nature of many anomaly detection algorithms. As discussed earlier in Section 4.6, an initial poor choice of the autoencoder results in an amplification of poor choice of

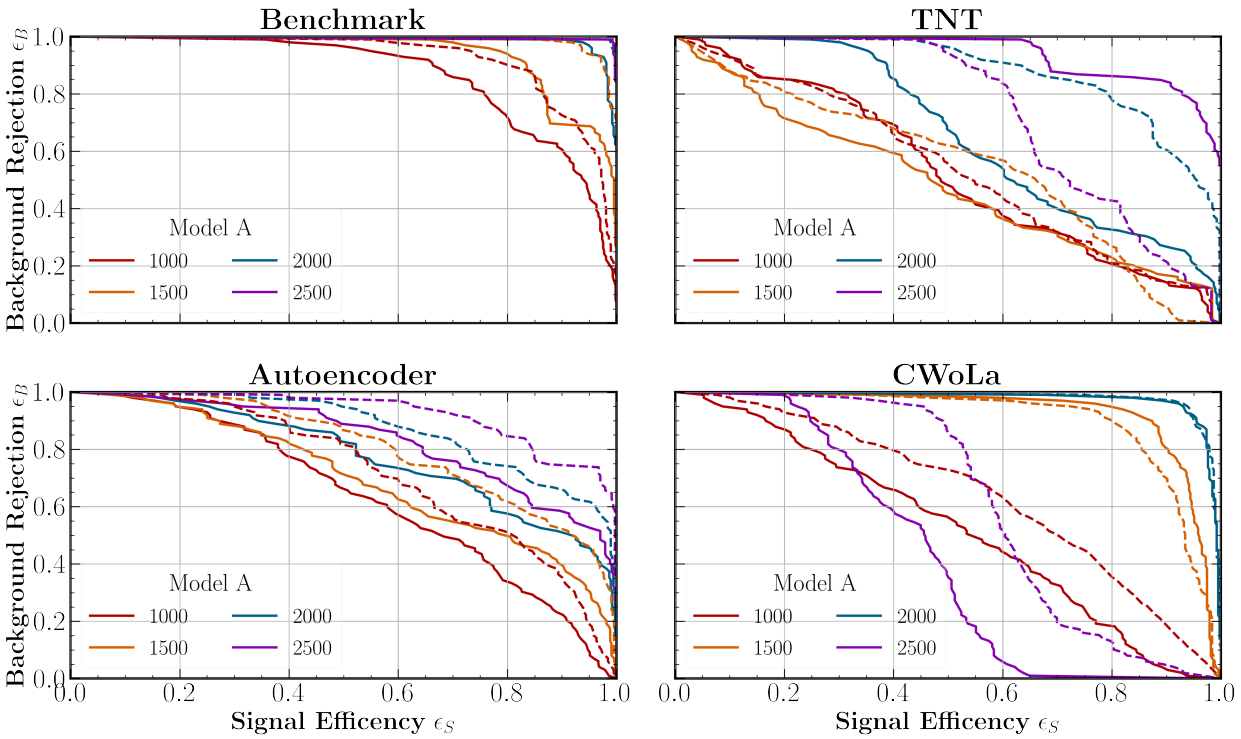


Figure 5.15: ROC curves for Model A at various mass peaks. Solid lines indicate no use of the track multiplicity, n_{trk}, the dashed line indicates use of n_{trk} in the respective algorithm.

signal for TNT, which leads to reverse classification whereby signal is labelled as background and vice versa. A poor choice of signal region for CWoLa can lead to near random results, as indicated by Model B at mass 1000 GeV in both Figure 5.16 and Table 5.2. This is not to say that the algorithms are unusable, as these AUC scores do indicate some information was gained in using the jet substructure to discriminate signal, but in order to correctly misclassify events. However, in practice whereby we are blinded and do not know beforehand whether an event contains dark jets, these results are extremely misleading and so must be handled with caution.

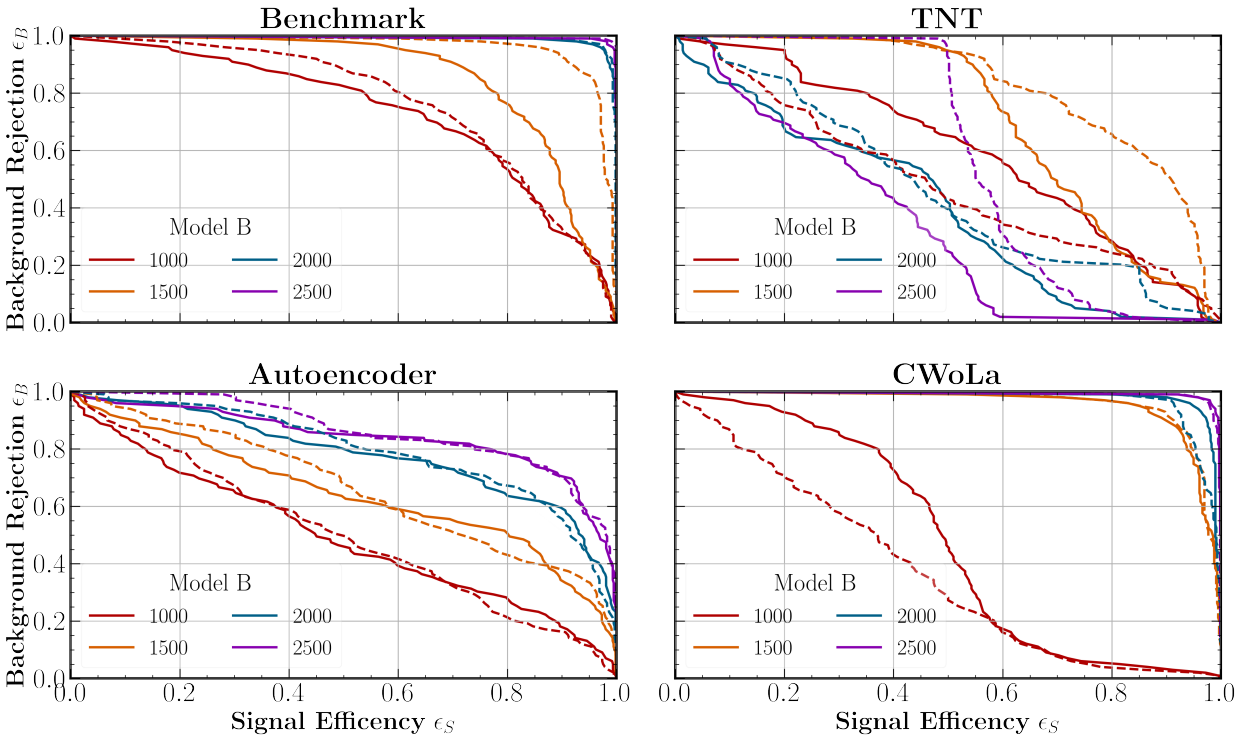


Figure 5.16: ROC curves for Model B at various mass peaks with (solid line) and without (dashed line) the use of n_{trk}.

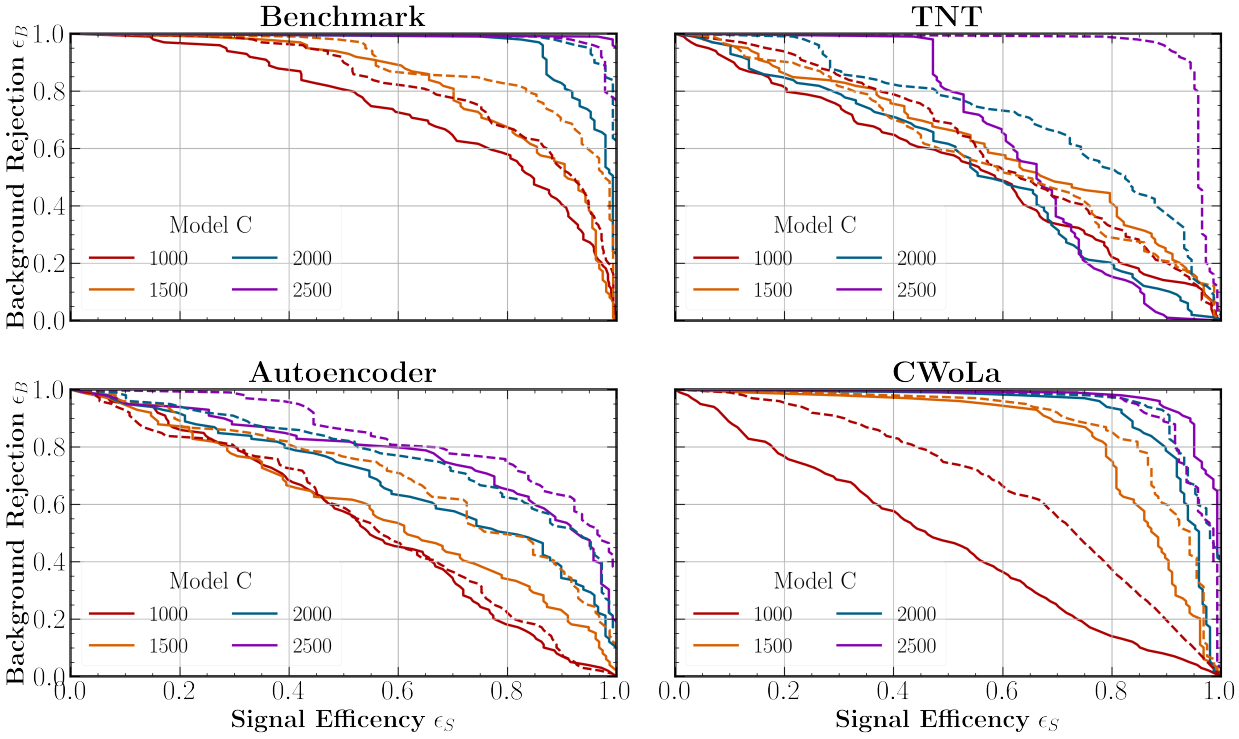


Figure 5.17: ROC curves for Model C at various mass peaks with (solid line) and without (dashed line) the use of n_{trk}.

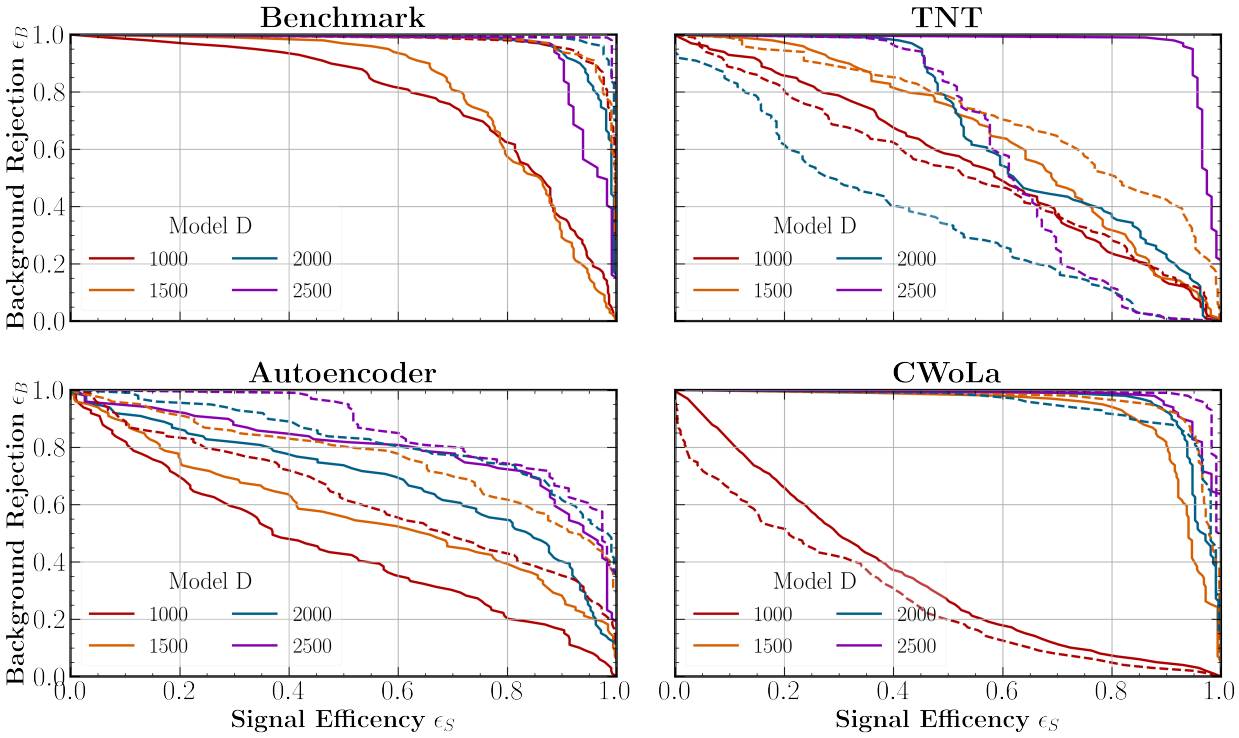


Figure 5.18: ROC curves for Model D at various mass peaks with (solid line) and without (dashed line) the use of n_{trk}.

Chapter 6

Conclusions

In this thesis, we have introduced theoretical models of dark and semi-visible jets and how they differ to usual dark matter searches. Work in the field indicated that the unique topologies of SVJ and DJR final states could be seen in the jet substructure observables. An overview of the topic of anomaly detection and various algorithms developed for particle physics research, was then given. We then used these unsupervised algorithms to see whether they could discriminate collision events that contained dark or semi-visible jet final state signatures, against events multijet background.

First, we compared a standard autoencoder and variational autoencoder in the t-channel production mode for a semi-visible jet models by varying the latent dimension, then we introduced the Tag N'Train and CWoLa algorithms for use in the s-channel of the SVJ model. For the dark jet models, we compared how supervised benchmark performance could be improved via a dynamic graph convolutional neural network, as well as the introduction of an initial combination and Lorentz layers. Finally, we inspected the use of the number of tracks within a jet, n_{trk} and how that may improve some of the unsupervised machine learning algorithms mentioned. The performance of these anomaly detection methods were visualised using ROC curves, and quantitatively assessed against each other and a supervised benchmark using their corresponding AUC scores.

On the topic of semi-visible jet, the lack of clear resonance signal in the t-channel production mode makes it difficult for any anomaly detection algorithm to be applied well. Although it was found that the standard autoencoder performs seemingly performs better than its variational counterpart, both have low AUC scores which make it difficult to recommend for use. The choice of latent dimension length, z also can have a large impact and would need to be chosen carefully when using autoencoders. However, better-suited AD techniques can improve results when working on the s-channel events. As suggested, CWoLa is powerful unsupervised algorithm for use in particle physics, and performs well in locating events which contain semi-visible jets in the s-channel if an appropriate signal region is chosen. Tag N'Train is equally capable of providing the same excellent results, but suffers extreme swings in performance if the initial weak autoencoder is poor.

For the dark jet models, both the DGCNN and LoLa/CoLa layers perform better than a standard deep neural network, showing evidence that there is more discriminating information to be found by the use of these two methods, at least for Model A at mass resonance, 1000 GeV. The use of the track multiplicity, n_{trk} improved the AUC scores of all dark jet models at almost all mass peaks, indicating its use powerful use in distinguishing between dark jets and multijet background. As expected, it is also easier to locate dark jets when the mass resonances are larger. What is remarkably to highlight is how close the CWoLa method can reach to the performance of a deep supervised neural network benchmark, considering its unsupervised nature. Despite its strong potential for use in indicating dark jets, a signal region needs to be chosen before training, in order

for the method to work. A poor choice of SR can lead to an AUC scores close to 0.5, or at worst a score below 0.5 where we gain the inverse classifier that tags background as signal and vice versa. The same can be said of the TNT technique, where extremely poor results can be produced by proliferating the bad performance of the initial autoencoder.

As of the time of writing, classical searches into dark and semi-visible jets are under way. However, recommended search strategies are in agreement with the finding presented thus far. That is, the use of jet substructure observables and, in particular, the number of tracks is of large importance in discrimination for dark jets [58]. For semi-visible jets, [62] found that SVJ's contain a more multi-pronged structure, resulting in larger C_2 and LHA values. However, this work utilised both without inspecting the discriminating power of these particular JSS observables.

We highlight that it is only because of simulation that we are able to measure this performance, which leads us to know which events contain dark jets before any algorithms are run. If there was ever use with actual collision data, we would not be able to evaluate this performance. However, we may comment further on suitable changes that would be necessary to implement these algorithms on assisting a search. For autoencoders, the required threshold needed for adequate performance may prove too difficult to select based on the results retrieved from simulated data. An optimal threshold can only be chosen a posteriori, and therefore cannot be used when dealing with collision data. Extending this issues to other algorithms, we may only reasonably say that only CWoLa can allow us a broad range of the threshold as it separates background and signal samples so well. We may utilise these models learnt on simulation data and applied to collision events, however, considering the unsupervised nature of the training, it is worthwhile to train directly on collision data, and its output utilized in order to assist a search. As stated within many of papers implemented such as Tag N'Train and CWoLa, a modified version of these algorithms can be used for background estimation on collision data. Comparisons to other background estimation techniques can then be made in order to assist a new search

6.1 Future Considerations

Serious implementation of neural network architectures, such as all the methods considered in this thesis, require an extensive search of optimal hyperparameters that ensure we may get the best performance out of these methods. Such hyperparameters include the use of dropout layers, early stopping, batch and epoch size, the number of fully connected layers, tuning of the learning rate, choice of gradient optimisers and objective functions, can all have profound impact on results, but were not explored in this work.

Specifically for jets, it is plausible to more closely examine the use of jet substructure observables. The substructure distributions used in this thesis is not exhausted, and new interesting observables related to dark and semi-visible jets may be proposed in the future, which can have better discriminating power than ones currently available.

A general suggestion for all dark matter searches involving jets is a motivation to develop both the DGCNN and LoLa/CoLa methods into unsupervised versions. It is shown in this work that both these algorithms perform better than a standard neural network, which indicates that these architectures more readily exploit the underlying physics processes presents. So, utilising these ideas for new anomaly detection techniques may prove very fruitful. One specific suggestion is to predicate an autoencoder with an inputted combination and Lorentz layer, thereby assisting the unsupervised architecture in grasping the essential physics that is used to discriminate between collision events.

Overall, we can state that, in general, it is difficult to derive unsupervised techniques which

use jet substructure variables to differentiate dark and semi-visible jets from multijet background. However, the CWoLa algorithm has large potential to aid a search for either dark or semi-visible jet models, considering its strengths and versatility. For dark jet models specifically, the track multiplicity variable is of good use on its own for helping to differentiate these models at all mass resonances from multijet background.

Part II

Use of Gaussian mixture models in aiding COVID-19 hotspot analysis for the Gauteng Province

Chapter 7

Big Data and Artificial Intelligence-Based Hot-Spot Analysis of COVID-19: Gauteng, South Africa, as a case study

Authors: Benjamin Lieberman, Roy Gusinow, Ali Asgaryand, Nicola Luigi Bragazzi, Joshua Choma, Salah-Eddine Dahbi, Kentaro Hayasi, Deepak Kar, Mary Kawonga, Jude Dzevela Kong, Mduduzi Mbada, Bruce Mellado, Kgomotso Monnakgotla, James Orbinski1, Xifeng Ruan, Finn Stevenson, and Jianhong Wu

This chapter is a paper submitted for publication to the ISPRS International Journal of Geo-Information, which highlights some of the work that I conducted for the Gauteng provincial government for the monitoring COVID-19 hotspots, using Gaussian mixture models. The implementation of this unsupervised anomaly detection algorithm was completed during my MSc research.

My contributions to this paper are as followed. I conducted the preprocessing of the COVID-19 cases, and implemented the scripts which utilise the Gaussian Mixture Model algorithm in Python 3. Fitting of the SI curves to each cluster was also handled by coding scripts I created. My work is highlighted in the Section 7.2.

7.1 Introduction

In late December 2019, a novel coronavirus, named "Severe Acute Respiratory Syndrome-related Coronavirus type 2" (SARS-CoV-2), emerged in the city of Wuhan, Hubei province of People's Republic of China [96]. The virus rapidly spread by the 11^{th} of March 2020, resulting in a confirmed global pandemic, known as "Coronavirus Disease 2019" (COVID-19). As of the 5^{th} of March 2021, the virus is affecting more than 218 countries, with the total number of confirmed cases exceeding 116 million and approximately 2.6 million fatalities worldwide being attributed to the effects of the virus. A large, worldwide modeling effort is currently underway to improve public health policy decision-making with regards to the still ongoing COVID-19 pandemic. Many research groups and national response teams have looked into country specific intervention strategies.

On the 5^{th} of March 2020, South Africa recorded its first COVID-19 case and three weeks later, on the 27^{th} of March, South Africa entered a full, government-enforced lockdown [97]. This formed part of a five tier risk-adjusted alert levels system [98]. The full list of South Africa's moves between lockdown levels can be seen below in Table 7.1 [99]. The first wave of COVID-19 continued in South Africa until October 2020 where the number of new cases had settled to a manageable amount. By late November 2020, South Africa's number of cases started to increase and the second wave of the pandemic began. The risk-adjusted system implemented allowed a controlled reopening/closing of the economy influenced by a set of factors, including the virus transmission rate, number of infectious cases, capacity of health facilities, the extent of the effectiveness of the implemented public health interventions and the economic and societal impact of continued restrictions.

Alert Level	**Wave**	**Start Date**	**Total Cases**	**Recoveries**	**Fatalities**
5	1	27 March	927	12	0
4	1	1 May	5951	2382	116
3	1	1 June	34357	17291	705
2	1	18 August	592144	485468	12264
1	2	21 September	661936	591208	15992
3	2	29 December	1021451	858456	27568

Table 7.1: South Africa's 2020 Alert Level Progression

The University of Witwatersrand and Ithemba Labs COVID-19 modelling group have formed part of the Gauteng's Department of Health advisory council, providing an in-depth analysis of the province's progress in the pandemic [100]. As part of the Gauteng Department of Health/Premier's advisory council, our modeling efforts provide information that government stakeholders use to inform their decisions, thus allowing a statistical grounds for changes in alert levels and distribution of resources.

COVID-19 data contain many complexities that must be taken into account when extracting information to guide public health decision- and policy-makers [101]. This complexity includes factors such as the large number of misclassified or under-reported infections, inconsistency and limitations in testing as well as fluctuating infection and fatality rates as influenced by social/behavioral dynamics.

As this data is the basis for modeling and therefore informing decisions around the risk-adjusted policies, understanding and accommodating these complexities in the model is vital. In generalising the spread of a virus over a large area, such as a province, it must be assumed that the transmission occurs as a stochastic process. This statistically random spread of a virus through a population

is the core of the majority of Susceptible-Infectious-Recovered-Deceased (SIRD) models and is dependent on factors such as number of infected cases, infection rate, level of social interactions, susceptible population and total population[100]. However the spread of COVID-19 and, therefore, the data representing the virus progression do not always conform to a stochastic model. In this paper, we will focus on the most influential non-stochastic dynamics of COVID-19 hot-spots.

A virus hot-spot can be defined as a cluster of cases within an area whose spreading dynamics do not conform to the general growth of the pandemic, exhibiting an exponential, short-lived growth. As these collections of cases do not conform to the macro-dynamics of the location, they need to be clearly defined and understood in order to accurately understand and model the virus progression. The geo-localization and clustering analyses of cases for this purpose are therefore vital and can be done using advanced artificial intelligence (AI) geo-clustering methods. This clustering approach can, therefore, be used to define individual clusters as hot-spots and allows the corresponding cases to be removed from the stochastic model - providing stochastic predictions that are not biased by the hot-spot dynamics [102].

7.2 Methods

7.2.1 Data Description

The data required for the hot-spot geo-localization analysis needs to be of a high level of detail. Therefore for this paper highly classified NICD COVID-19 Gauteng case data containing: ID (Person Case), recorded address, test date and geo-localization data (including latitude and longitude coordinates). The data has been prepossessed to remove geo-localization data that has incorrect address recorded or issues interpreting/processing the address.

7.2.2 Clustering Cases by Geo-Location

In order to analyse the area distribution of COVID-19 cases, AI techniques provide an excellent tool in grouping cases geographically. In this paper we focus on the semi-supervised machine learning method, Gaussian mixture model. This model allows us to accurately analyse the dynamics of the virus within a determined area.

AI and Clustering

The given problem is using the location of residence of each COVID-19 case in Gauteng to produce clusters. Once defined, these clusters can be analysed and accurately labelled as hot-spots or non-hot-spots.

There exists various clustering methods within AI and semi-supervised machine learning algorithms that can be implemented to solve a 2-dimensional (latitude/longitude co-ordinates) problem such as the present one. After evaluating various methods including the k-means algorithm, the Gaussian mixture model was chosen.

Gaussian Mixture models provide a probability-based approach to the likelihood of a COVID cases being within a cluster by producing a 2-dimensional Gaussian probability model overlayed onto the Gauteng map area. The clusters produced can overlap with each other, which encapsulates the possibility that hot-spots may very well also overlap with each other. The corresponding weight, ϕ generated for each cluster, provides a simplistic estimate of the importance of the cluster, as well as another variable for filtering false clusters from actual hot-spots.

Gaussian Mixture Model (GMM)

A Gaussian Mixture model is an algorithm which operates by generating k 2-dimensional Gaussian probability distributions, where k is a hyper-parameter specified. Thus, we are required to generate means, μ_j, covariance Σ_j and weighting, ϕ_j where the index specifies the $j-th$ Gaussian cluster. So, the probability of a new case, $p(x)$ occurring at a given point x is a linear combination of probabilities from all generated clusters,

$$p(x) = \sum_{j=1}^{k} \phi_i \, \mathcal{N}\left(x \mid \mu_j, \Sigma_j\right), \tag{7.1}$$

where $\mathcal{N}$ is the normal distribution. We generate the set of normal distributions (with associated weights, means and covariances) with an algorithm which optimally fits the probability distributions given the set of already known COVID-19 cases and their coordinates.

In order to generate k-Gaussian probability distribution, the **Expectation-Maximisation** algorithm is employed.

At the expectation step, we calculate the probability that a point x_i was generated by the $j-th$ Gaussian for all k distributions:

$$\gamma_{ij} = \frac{\phi_j N\left(x_i \mid \mu_j, \Sigma_j\right)}{\sum_{q=1}^{k} \phi_j N\left(x_i \mid \mu_q, \Sigma_q\right)}. \tag{7.2}$$

In the maximisation step, the probabilities γ_{ij} are used to generate new cluster parameters. That is, new mean μ_j, co-variance Σ_j and weight ϕ_j are updated as follows:

$$\phi_j = \sum_{i}^{N} \frac{\gamma_{ij}}{N} \quad , \quad \mu_j = \sum_{i}^{N} \frac{\sum_i^N \gamma_{ij} x_i}{\sum_i^N \gamma_{ij}} \quad , \quad \Sigma_j^2 = \frac{\sum_i^N \gamma_{ij}(x_i - \mu_j)(x_i - \mu_j)^T}{\sum_i^N \gamma_{ij}}. \tag{7.3}$$

These steps are iterated until a convergence criteria is met. In our case, the variable $x = \{x, y\}$ is the set of longitudinal, y and latitudinal coordinate, x.

Susceptible-Infection Curve

Once the latent variables of the Gaussian probabilities distributions (weights, means, standard deviation) have been found through the processing of COVID-19 cases in Gauteng, it is important to verify which clusters are hot-spots, or highly infectious areas/districts of the province. In order to accomplish this, the time dependent progression of cases is inspected for each cluster independently. That is, the cumulative number of cases was computed as a function of the date the patients were first recorded to have contracted the virus.

An aspect to consider is whether the clusters found follow the *Susceptible-Infection (SI) Curve*, which model the number of susceptible people who get infected, $SI(t)$, over time, within a given area/cluster. The SI equation is shown below in Equation 7.4;

$$SI(t) = \frac{SI_0}{1 + e^{-SI_1(t - SI_2)}}. \tag{7.4}$$

SI_0 is the total number of predicted cases within a cluster once it has saturated the susceptible population, SI_1 represents the rate of infection of the virus, and SI_2 is the number of days before the peak of growth of the cluster. This function is a solution to the logistic differential equation, a simple system which describes the number of infected cases in a given population. The model is applicable as we expect a small increase of infection cases in the early stages of a susceptible

population, and then a sharp increase as the disease spreads rapidly throughout the cluster. A plateau is expected once all susceptible people within a cluster are infected.

The SI curve can therefore be fitted to the time-series of each cluster in order to generate these parameters for the $j-th$ cluster, giving more properties to accurately filter clusters into hot-spots. A poorly fit SI curve can indicate that the cluster is not a COVID-19 hot-spot, as it does not follow an accurate description of disease spread.

7.3 Results and Discussion

Once the cases throughout Gauteng province have been clustered and described, each area can be described through the following parameters; Total Cumulative Cases (N_{TC}), 1^{st} and 2^{nd} standard deviation area (A_{1sd} and A_{2sd} respectively) and the susceptible-infection parameters (SI_0, SI_1 and SI_2).

7.3.1 Hot-Spot Density Definition

Taking the distinct specifications of each cluster into account a formal definition of a hot-spot cluster can be established. The density distribution of the clusters, shown in Figure 7.1, forms a Gaussian like shape at densities from 0 - 350 $\frac{cases}{km^2}$ followed by a sporadic tail of densities of 350 to more than 30000 $\frac{cases}{km^2}$. The uniformity of low density clusters is found to be associated with stochastic growth. Thus by cutting the densities at the two Sigma interval we are able to produce a density threshold of **196.05** $\frac{cases}{km^2}$. Clusters with densities greater than the threshold are found to have rapid, non-stochastic growth. This density threshold therefore allows us to define hot-spot clusters as any cluster whose density exceeds the determined density threshold.

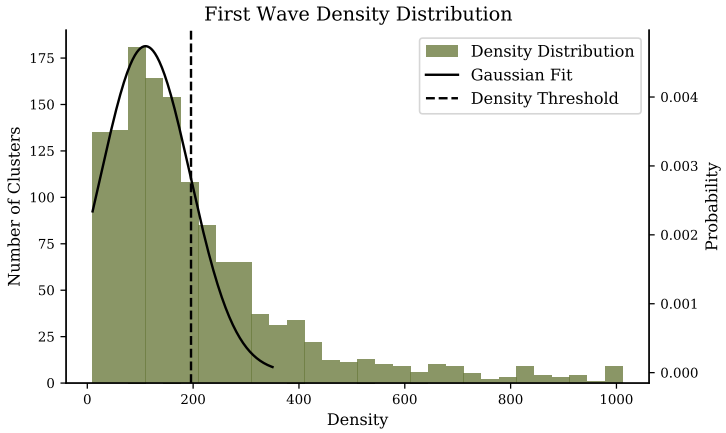


Figure 7.1: Gauteng Cluster Density Distribution with Gaussian Fit.

Therefore a cluster is defined as a hot-spot when the cluster's density, $\rho_c(t)$, is greater or equal to the threshold density, ρ_{th}.

We applied this criterion to the first wave of COVID-19 in Gauteng Province where $\rho_{cluster}(t)$ is the case density of a given cluster at a given day and ρ_{th} is the minimum density stipulating hot-spot dynamics. Out of 1,500 clusters, once split on the density threshold 607 of the clusters are defined as hot-spots and the remaining 893 clusters are defined as stochastic.

In order to evaluate this definition further we can compare the susceptible-infection parameters of the clusters defined as hot-spots against the stochastic or non-hot-spot clusters.

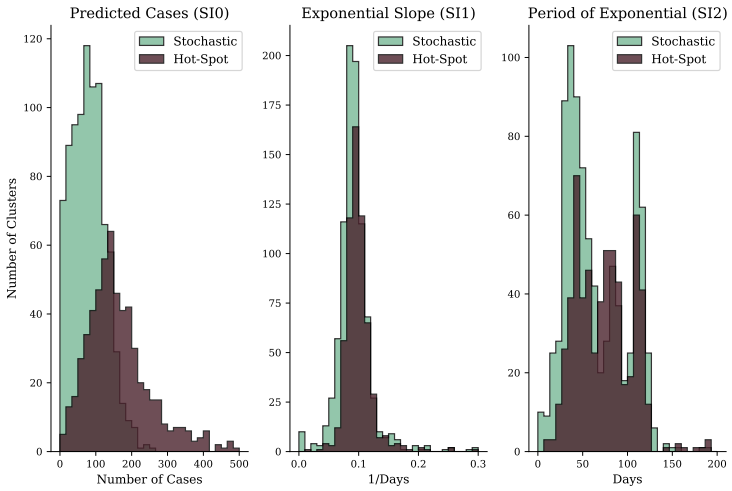


Figure 7.2: Gauteng Cluster parameter Distributions Comparison.

Once Hot-Spot clusters are defined by their density, it can be seen, in Figure 7.2, that Hot-Spot clusters have on average an increased number of total cases, ±180, than the stochastic clusters, ±90. Hot-Spot clusters also have a slightly increased exponential slope with a period of ±10 days where stochastic clusters period of exponential slope can be seen to be ±11 days.

An evaluation of this hot-spot definition can be seen using a comparison of the total cases in stochastic clusters and hot-spot clusters during the first wave. Figure 7.3 reflects that during the first wave approximately two thirds of the cases in Gauteng occurred in hot-spot clusters. This case distribution shows excellent coherence with first wave stochastic predictions (Using a Di-SIRD linear control model [100]) compared to data, as shown in Figure 7.4. This example of stochastic prediction demonstrates how the emergence of hot-spots in June 2020 did not follow the expected stochastic progression of the virus.

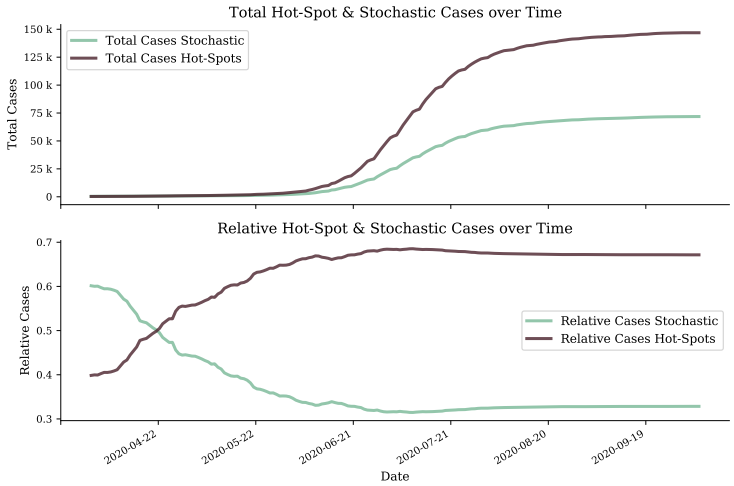


Figure 7.3: Number of Hot-Spot Cases Over Time During the First Wave.

Therefore, it can be seen that the density cut-off value defining hot-spot clusters successfully is able to extract the clusters growing more exponentially and sporadically from those with a more

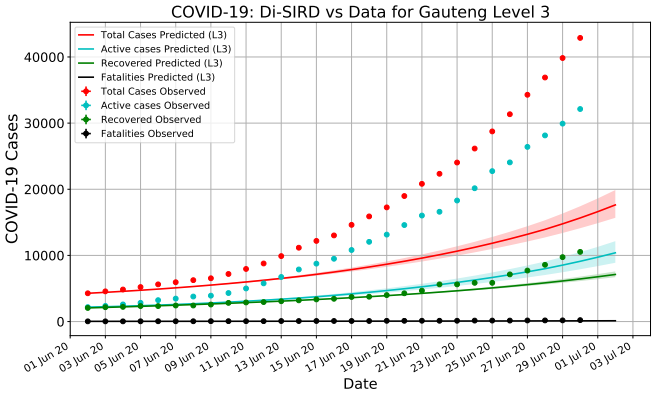


Figure 7.4: First Wave Stochastic Prediction Vs Data.

uniform, random growth.

7.3.2 Hot-Spot Activity Analysis

Once a hot-spot cluster's total cases reaches the plateau or passes the peak of a surge, it can be said that the dynamics of the cluster is no longer that of a hot-spot. The activity of a cluster at any point in time can therefore be quantified as the ratio of the total cases in the cluster at the respective time divided by the total predicted cases of the cluster, SI_0, described in Criteria 7.5.

$$\frac{N_{TC}(t)}{SI_0} < L_{th} \tag{7.5}$$

where the activity threshold, L_{th}, represents the upper bound on actively growing clusters using the ratio of Total Cases to Total Predicted Cases. As one would expect all the clusters that where defined as hot-spots during the first wave have returned to stochastic dynamics after the first wave completion. More specifically, we are able to determine an activity threshold. The activity threshold assumes that only 1% of clusters remain active in the subsequent period of the first wave with a corresponding proportional error. Therefore as shown in the activity distribution, Figure 7.5, the activity threshold is determined to be **0.85**.

The time dependent evolution of newly defined hot-spots as well as hot-spots that are returning to stochastic dynamics in Gauteng during the first wave can be analysed using the above defined criteria. These dynamics are visualised in Figures 7.6 and 7.7 respectively.

To understand the growth of the hot-spot clusters an SI curve is fit to the cumulative number of hot-spot clusters shown in Figure 7.6. The daily increase of hot-spot clusters peaks in mid July, which is confirmed by the SI_2 parameter which determines the inflection of exponential growth to occur on the 10th of July, 101 days after the 1st of April. The cumulative hot-spot clusters reaches its plateau in mid August coinciding with South Africa's move from level 3 to level 2, with 594 of the total 1,500 clusters having already developed into hot-spots. The SI fit to the cumulative number of hot-spot clusters describes the period of the exponential growth to be approximately 12 days ($1/SI_1$).

Figure 7.7 shows not only the emergence of hot-spot clusters but also when hot-spots progress back to a stochastic dynamics, described by Equation 7.5. Clusters experiencing hot-spot dynamics start to reach their peak, and therefore progress back to stochastic clusters, from mid July. By the

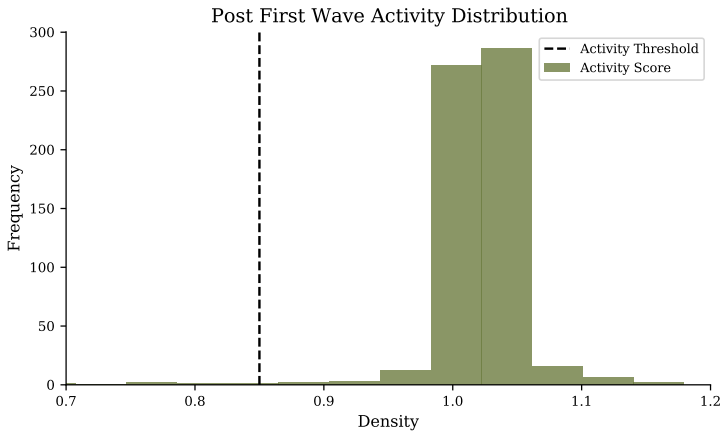


Figure 7.5: Gauteng Activity Distribution After First Wave Completion.

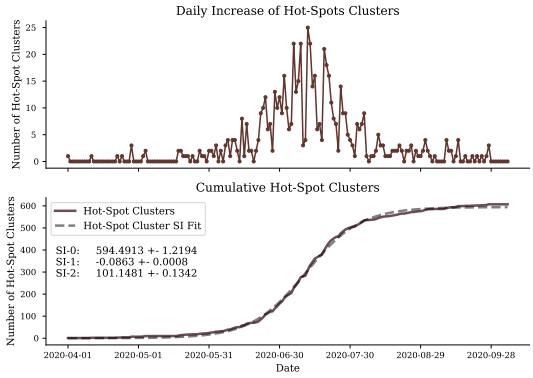


Figure 7.6: Cumulative and Emerging COVID-19 Hot-Spot Clusters in Gauteng

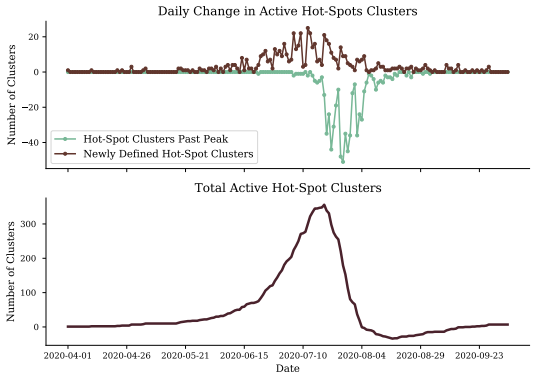


Figure 7.7: Number of Active Hot-Spot Clusters.

end of August a maximum of 39 hot-spots have reached their peak and by the end of September all but 21 cluster have progressed back to stochastic dynamics.

7.3.3 Second Wave Risk Index Definition

The second wave risk index quantifies the deviation of the data from the hypothesis of a single wave. This index can be used to measure the risk of a second wave occurring and/or the severity of a given cluster.

The second wave risk index, RI, is calculated using Equation 7.6.

$$RI = \begin{cases} RI_0 + RI_1, & \text{if } RI_1 > 8 \\ RI_0, & \text{if } RI_1 \leq 8 \end{cases} \tag{7.6}$$

Where RI_0 and RI_1 are determined using Equations 7.7 and 7.8 respectively.

$$RI_0 = 100 * \frac{N_{TC}(t) - SI_0(t)}{SI_0(t)} \tag{7.7}$$

$$RI_1 = 10 * \frac{1}{1 + \frac{N_{TC}(t)}{SI_0(t)}} \tag{7.8}$$

In order to understand the pandemic severity as reflected by the RI, the first wave's RI distribution is used to define the extent of risk of a given cluster. Analysing the clusters after the completion of the first wave allows for a boundary to be determined for both stochastic and hot-spot distributions reflecting at which value of the RI a cluster can be defined as at risk, as shown in Figure 7.8. The boundary is determined to allow for 1% of clusters to remain at high risk even after the completion of the first wave.

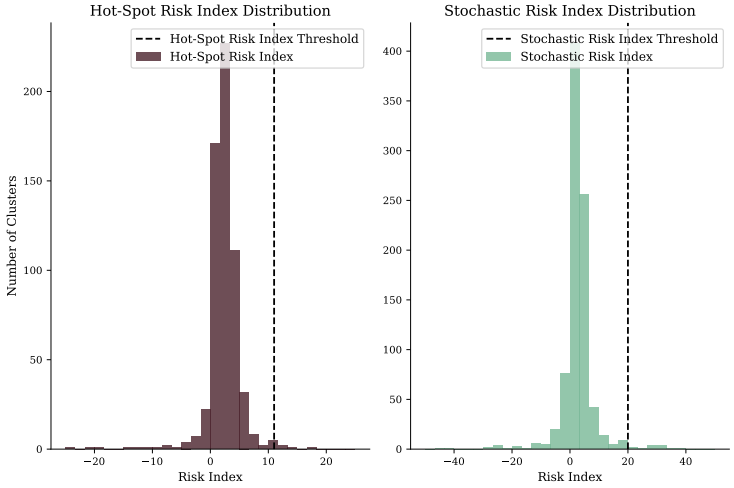


Figure 7.8: Gauteng First Wave Distributions of Second Wave Risk Index split into Hot-Spots and Stochastic Clusters

Therefore, in the second wave analysis, a hot-spot cluster with a RI greater than **11** can be classified as a high risk hot-spot. Similarly a stochastic cluster with a RI greater than **20** can be classified as a developing high risk cluster.

7.3.4 Applications of Hot-Spot Definition for Second Wave

The definition and parameterization of clustered cases provides various applications in informing stakeholders in their decisions related to COVID-19 interventions and preventative measures. Sec-

tion 7.3.4 discusses two of these applications. The first application allows for the hot-spot dynamics to be integrated into epidemiological models, while the second and more vital role is to expose potentially problematic areas in order to inform intervention strategies and advance social awareness and adoption of proper behaviors.

Implementation of Hot-Spot Analysis into SIRD Model

A problem encountered in modeling the COVID-19 pandemic is that SIRD models generally function stochastically (random β dependent spread through susceptible population). However pockets of cases developing usually in high density areas undergo independent, rapid infection that does not fit into larger model. This micro-system cluster is referred to as a hot-spot and undergoes independent non-stochastic hot-spot dynamics. In order to classify a specific group of cases in an area as a hot-spot the cases must first be grouped and their characteristics modeled, using each groupings characteristics to define a hot-spot cluster.

It therefore follows that in order to produce informative predictions for governmental policy- and decision-makers, such as estimate numbers of hospital beds, use of intensive care units (ICUs) wards and when the peak will occur, the hot-spot cluster cases must be extracted from the data the stochastic SIRD model is calibrated on. The model then is able to interpret the progression of COVID-19 without the inconsistencies incurred by the non-conforming hot-spot cases.

This is done by extracting the daily ratio of stochastic cumulative cases from the total cases in all clusters and applying this ratio to the recorded data before it is used to inform the model.

$$I_{stoch} = \frac{I_s}{I_s + I_{hs}} * I_d \tag{7.9}$$

where I_{stoch} is the stochastic active cases, I_s is the active cases in stochastic clusters, I_h is the active cases in hot-spot clusters and I_d is the active cases recorded.

Exposing Hot-Spot and High Risk Clusters

The primary need for COVID-19 Hot-Spot classification is to target clusters/areas where non-conforming, exponential growth is occurring. Using the definition of hot-spot clusters developed in Section 7.3, clusters can effectively be classified and their progression and dynamics described. Table 7.2 summarises descriptive parameters of a classified cluster.

Hot-Spot Classification	**Cluster Activity**	**Risk Index**
If cluster can be described as a hot-spot	At what point through its progression a cluster is	The severity of infection rate and scale of a cluster

Table 7.2: Summary of specifications of classified clusters

These three parameters describing each cluster are able to inform stakeholders not only on what areas are considered COVID-19 high growth areas but also the period of time the cluster will last and how severe the dynamics of the cluster is. This can then be visualised in an interactive map for stakeholders as shown below in Figure 7.9.

Emerging spatio-temporal hot-spot analysis is of crucial importance for public health policy- and decision-makers and can provide valuable information that would not possible to achieve with other techniques, enabling to capture specific clustering patterns in terms of particular districts and areas that would be otherwise classified as being at low risk for spreading COVID-19. Hot-spot

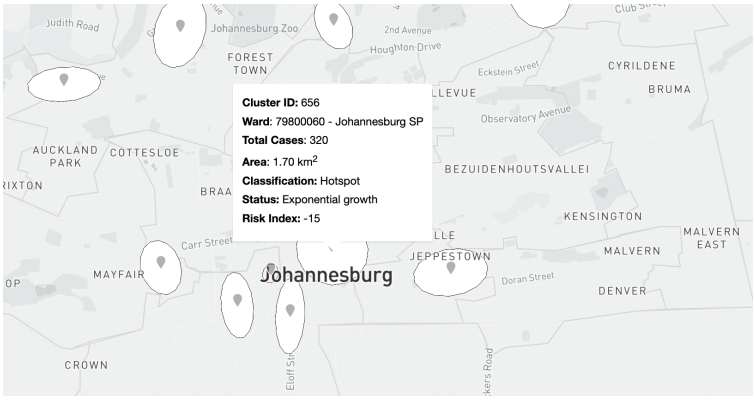


Figure 7.9: Hot-Spot visualisation on gpcoronavirus.co.za

analysis can complement classical epidemiological and surveillance approaches, shedding light on COVID-19 spatio-temporal trends and the possible evolution of its trajectories. Furthermore, the hot-spot analysis enables to easily visualize data in a way that is accessible for stakeholders and helps them in the decision-making process.

In the existing scholarly literature, some studies have performed a hot-spot analysis of COVID-19. For instance, Shariati and colleagues [103] have computed Anselin Local Moran's I indices to identify high- and low-risk clusters of COVID-19 worldwide. Authors were able to locate San Marino and Italy as territories characterized by a dramatically high toll of deaths, with infectious hot-spots widespread in northern Africa as well as southern, northern and western Europe. Noteworthy, infectious cases occurring in these hot-spots represent about 70 percent of all global infectious cases.

Other hot-spot analyses have been carried out at the nation level. Mo and coworkers [104] coupled local outlier analysis with hot-spot analysis based on space-time cube metrics in mainland China. Authors were able to demonstrate a rather quick, uneven spreading of the outbreak from the cities of Wuhan and Shiyan to the neighbouring areas and provinces. In Italy, combining a variety of geospatial analytical methods (spatial autococorrelation, spatio-temporal clustering and kernel density techniques), infodemiology (Google Trends and web searches analysis) and AI methods (machine learning and Adaboost algorithm for single-factor modelling), Niu and collaborators [105] were able to provide an in-depth assessment of the COVID-19 outbreak, in terms of its distribution and spreading characteristics. Hot-spots could be identified mainly in northern Italy.

Purwanto and colleagues [106] explored COVID-19 distribution patterns in East Java (Indonesia). Authors were able to identify Surabaya as major hot-spot, from which the outbreak reached cities characterized by high density of roads, food venues, and commercial and financial facilities.

In the present investigation, we have provided a robust statistical method for distinguishing between hot-spots and areas characterized by stochastic spreading of COVID-19 cases. We applied this analytical framework to the first and second waves, taking Gauteng province, South Africa, as a case study. These methods are general-purpose and can be, as such, applied to other countries as well.

7.4 Conclusions

Hot-spot analyses can represent sophisticated techniques, that can be further supplemented by ad hoc epidemiological surveys [107].

Hot-spot analysis represents an advanced statistical approach that can be effectively utilized for outbreak analytics and visualization. It can equip public health policy- and decision-makers with updated, real-time assessment of the pandemic trends and its future projected trajectories. Furthermore, it can complement classical epidemiological surveys, leading to the identification of patterns that would be otherwise classified as low-risk ones. In conclusion, hot-spot analysis has been highly helpful in promptly recognizing high-risk clusters, and to adopt/adjust proper public health measures. Since the COVID-19 pandemic is a highly changeable and constantly under flux situation, we can anticipate that hot-spot analysis can aid stakeholders in making informed, evidence-based and data-driven decisions, while several countries are currently facing the third wave of the outbreak and are making efforts in vaccine roll-out.

Bibliography

[1] University of Zurich — Standard Model. "`https://www.physik.uzh.ch/en/researcharea/lhcb/outreach/StandardModel.html`".

[2] Mary K. Gaillard, Paul D. Grannis, and Frank J. Sciulli. The standard model of particle physics. *Rev. Mod. Phys.*, 71:S96–S111, Mar 1999.

[3] P.W. Higgs. Broken symmetries, massless particles and gauge fields. *Physics Letters*, 12(2):132–133, 1964.

[4] Peter W. Higgs. Broken symmetries and the masses of gauge bosons. *Phys. Rev. Lett.*, 13:508–509, Oct 1964.

[5] F. Englert and R. Brout. Broken symmetry and the mass of gauge vector mesons. *Phys. Rev. Lett.*, 13:321–323, Aug 1964.

[6] G. S. Guralnik, C. R. Hagen, and T. W. B. Kibble. Global conservation laws and massless particles. *Phys. Rev. Lett.*, 13:585–587, Nov 1964.

[7] Nima Arkani–Hamed, Savas Dimopoulos, and Gia Dvali. The hierarchy problem and new dimensions at a millimeter. *Physics Letters B*, 429(3-4):263–272, Jun 1998.

[8] Stephen P. Martin. *A Supersymmetry Primer*, pages 1–153.

[9] 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*, 716(1):1–29, Sep 2012.

[10] CMS Collaboration. Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC. *Physics Letters B*, 716(1):30–61, Sep 2012.

[11] KTeV Collaboration. Measurements of direct CP violation, CPT symmetry, and other parameters in the neutral kaon system. *Phys. Rev. D*, 67:012005, Jan 2003.

[12] KTeV Collaboration. Erratum: Measurements of direct *cp* violation, *cpt* symmetry, and other parameters in the neutral kaon system [phys. rev. d 67, 012005 (2003)]. *Phys. Rev. D*, 70:079904, Oct 2004.

[13] The Large Hadron Collider. "`https://home.cern/science/accelerators/large-hadron-collider`".

[14] CERN's Accelerator Complex. "`https://home.cern/science/accelerators/accelerator-complex`".

[15] The first LHC protons run ends with new milestone. "https://home.cern/news/press-release/cern/first-lhc-protons-run-ends-new-milestone".

[16] LHCb Collaboration. Observation of the resonant character of thez(4430)-state. *Physical Review Letters*, 112(22), Jun 2014.

[17] CERN Timeline. "https://www.lhc-closer.es/taking_a_closer_look_at_lhc/0.cern_timeline".

[18] CMS Collaboration. Search for resonant production of high-mass photon pairs in proton-proton collisions at $\sqrt{s} = 8$ and 13 tev. *Phys. Rev. Lett.*, 117:051802, Jul 2016.

[19] New Schedule for CERN's Accelerators and Experiments. "https://home.cern/news/news/accelerators/new-schedule-cerns-accelerators-and-experiments".

[20] Computer Generated Image of the whole ATLAS Detector. "https://cds.cern.ch/record/1095924".

[21] The ATLAS Detector. "https://atlas.cern/discover/detector".

[22] The Inner Detector. "http://cds.cern.ch/record/2759658/files/The%20Inner%20Detector.pdf".

[23] Alessandro La Rosa. The ATLAS Insertable B-Layer: from construction to operation, 2016.

[24] Magnet System. "http://cds.cern.ch/record/2759664/files/Magnet%20System.pdf".

[25] Calorimeters. "http://cds.cern.ch/record/2759659/files/Calorimeters.pdf".

[26] Muon Spectrometer. "http://cds.cern.ch/record/2759662/files/Muon%20Spectrometer.pdf".

[27] Trigger and Data Acquisition. "https://atlas.cern/discover/detector/trigger-daq".

[28] Maxim Perelstein. Introduction to Collider Physics. *Physics of the Large and the Small*, Mar 2011.

[29] ATLAS Collaboration. ATLAS events at 13 TeV - 2016 conferences. General Photo, Jul 2016.

[30] Matteo Cacciari, Gavin P Salam, and Gregory Soyez. The Catchment Area of Jets. *Journal of High Energy Physics*, 2008(04):005–005, Apr 2008.

[31] Matteo Cacciari, Gavin P. Salam, and Gregory Soyez. FastJet User Manual. *The European Physical Journal C*, 72(3):1896, Mar 2012.

[32] Andrew J. Larkoski, Jesse Thaler, and Wouter J. Waalewijn. Gaining (mutual) information about quark/gluon discrimination. *Journal of High Energy Physics*, 2014(11), Nov 2014.

[33] Jason Gallicchio and Matthew D. Schwartz. Quark and Gluon Tagging at the LHC. *Physical Review Letters*, 107(17), Oct 2011.

[34] Jesse Thaler and Ken Van Tilburg. Identifying boosted objects with N-subjettiness. *Journal of High Energy Physics*, 2011(3), Mar 2011.

[35] Andrew J. Larkoski, Gavin P. Salam, and Jesse Thaler. Energy correlation functions for jet substructure. *Journal of High Energy Physics*, 2013(6), Jun 2013.

[36] Patrick T. Komiske, Eric M. Metodiev, and Jesse Thaler. Energy flow networks: deep sets for particle jets. *Journal of High Energy Physics*, 2019(1), Jan 2019.

[37] S Catani, Yu L Dokshitzer, Michael H Seymour, and Bryan R Webber. Longitudinally-invariant $k_\perp$-clustering algorithms for hadron-hadron collisions. *Nucl. Phys. B*, 406:187–224. 38 p, Feb 1993.

[38] Geoffrey C. Fox and Stephen Wolfram. Observables for the Analysis of Event Shapes in e^+e^- Annihilation and Other Processes. *Phys. Rev. Lett.*, 41:1581–1585, Dec 1978.

[39] ATLAS Collaboration. Measurement of kT splitting scales in W events at $\sqrt{s} = 7$ TeV with the ATLAS detector. *The European Physical Journal C*, 73(5), May 2013.

[40] Leandro G. Almeida, Seung J. Lee, Gilad Perez, Ilmo Sung, and Joseph Virzi. Top quark jets at the LHC. *Physical Review D*, 79(7), Apr 2009.

[41] UA1 Collaboration. Experimental Observation of Isolated Large Transverse Energy Electrons with Associated Missing Energy at $\sqrt{s} = 540$ GeV. *Physics Letters B*, 122(1):103 – 116, 1983.

[42] UA2 Collaboration. Observation of Single Isolated Electrons of High Transverse Momentum in Events with Missing Transverse Energy at the CERN anti-pp Collider. *Physics Letters B*, 122(5):476 – 485, 1983.

[43] Fritz Zwicky. Die Rotverschiebung von extragalaktischen Nebeln (The Redshift of Extragalactic Nebulae). *Helvetica Physica Acta*, 1933.

[44] E. Komatsu, J. Dunkley, and M. R. Nolta. Five-Year Wilkinson Microwave Anisotropy Probe Observations: Cosmological Interpretation. *apjs*, 180(2):330–376, February 2009.

[45] C. L. Bennett, R. S. Hill, G. Hinshaw, and M. R. Nolta. First-Year Wilkinson Microwave Anisotropy Probe (WMAP) Observations: Foreground Emission. *apjs*, 148(1):97–117, September 2003.

[46] 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*, 716(1):1 – 29, 2012.

[47] R. Bernabei, P. Belli, F. Cappella, R. Cerulli, A. d'Angelo, C.J. Dai, H.L. He, A. Incicchitti, H.H. Kuang, X.H. Ma, F. Montecchia, F. Nozzoli, D. Prosperi, X.D. Sheng, and Z.P. Ye. Direct detection of dark matter particles. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 630(1):279–284, 2011. Proceedings of the 2nd Roma International Conference on Astroparticle Physics (RICAP 2009).

[48] Scott Tremaine and James E. Gunn. Dynamical role of light neutral leptons in cosmology. *Phys. Rev. Lett.*, 42:407–410, Feb 1979.

[49] Ortwin E. Gerhard and David N. Spergel. Dwarf Spheroidal Galaxies and the Mass of the Neutrino. *APJL*, 389:L9, April 1992.

[50] Csaba Csaki. The Minimal Supersymmetric Standard Model. *Modern Physics Letters A*, 11(08):599–613, Mar 1996.

[51] J. Swart, G. Bertone, and J. V. Dongen. How dark matter came to matter. *Nature Astronomy*, 1:0059, 2017.

[52] Toby Falk. Neutralinos as Dark Matter. In Kingman Cheung, John F. Gunion, and Stephen Mrenna, editors, *Particles, Strings and Cosmology*, page 394, January 2000.

[53] C. A. de S. Pires, P. S. Rodrigues da Silva, A. C. O. Santos, and Clarissa Siqueira. Higgs mass and right-handed sneutrino WIMP in a supersymmetric 3-3-1 model. *Physical Review D*, 94(5), Sep 2016.

[54] Kim Griest. The search for the dark matter: WIMPs and MACHOs. *Annals of the New York Academy of Sciences*, 688(1):390–407, Jun 1993.

[55] J. Grube and VERITAS Collaboration. Veritas limits on dark matter annihilation from dwarf galaxies. *AIP Conference Proceedings*, 1505(1):689–692, 2012.

[56] Francesc Ferrer, Lawrence M. Krauss, and Stefano Profumo. Indirect detection of light neutralino dark matter in the next-to-minimal supersymmetric standard model. *Physical Review D*, 74(11), Dec 2006.

[57] Antonio Boveia and Caterina Doglioni. Dark Matter Searches at Colliders. *Annual Review of Nuclear and Particle Science*, 68(1):429–459, Oct 2018.

[58] Timothy Cohen, Mariangela Lisanti, Hou Keong Lou, and Siddharth Mishra-Sharma. LHC searches for dark sector showers. *Journal of High Energy Physics*, 2017(11), Nov 2017.

[59] Elias Bernreuther, Felix Kahlhoefer, Michael Krämer, and Patrick Tunney. Strongly interacting dark sectors in the early Universe and at the LHC through a simplified portal. *Journal of High Energy Physics*, 2020(1), Jan 2020.

[60] Timothy Cohen, Mariangela Lisanti, and Hou Keong Lou. Semivisible Jets: Dark Matter Undercover at the LHC. *Physical Review Letters*, 115(17), Oct 2015.

[61] Myeonghun Park and Mengchao Zhang. Tagging a jet from a dark sector with jet substructures at colliders. *Physical Review D*, 100(11), Dec 2019.

[62] Deepak Kar and Sukanya Sinha. Exploring jet substructure in semi-visible jets. *SciPost Physics*, 10, 04 2021.

[63] J. Alwall, R. Frederix, S. Frixione, V. Hirschi, F. Maltoni, O. Mattelaer, H.-S. Shao, T. Stelzer, P. Torrielli, and M. Zaro. The automated computation of tree-level and next-to-leading order differential cross sections, and their matching to parton shower simulations. *Journal of High Energy Physics*, 2014(7), Jul 2014.

[64] Torbjörn Sjöstrand et al. An introduction to PYTHIA 8.2. *Computer Physics Communications*, 191:159–177, Jun 2015.

[65] Morgan Svensson Seth. A first study of Hidden Valley models at the LHC. *arXiv: High Energy Physics - Phenomenology*, 2011.

[66] S. Agostinelli, J. Allison, K. Amako, and et. al. Geant4—a simulation toolkit. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, 506(3):250–303, 2003.

[67] V. M. Abazov and B. Abbott. Evidence for production of single top quarks. *Phys. Rev. D*, 78:012005, Jul 2008.

[68] B. S. Kronheim, M. P. Kuchera, H. B. Prosper, and A. Karbo. Bayesian Neural Networks for Fast SUSY Predictions. *Phys. Lett. B*, 813:136041, 2021.

[69] Riccardo Di Sipio, Michele Faucci Giannelli, Sana Ketabchi Haghighat, and Serena Palazzo. DijetGAN: a Generative-Adversarial Network approach for the simulation of QCD dijet events at the LHC. *Journal of High Energy Physics*, 2019(8), Aug 2019.

[70] ATLAS Collaboration. Identification of Hadronically-Decaying W Bosons and Top Quarks Using High-Level Features as Input to Boosted Decision Trees and Deep Neural Networks in ATLAS a $\sqrt{s} = 13$ TeV. Technical report, CERN, 2017.

[71] Gregor Kasieczka, Benjamin Nachman, and David Shih. The LHC Olympics 2020: A Community Challenge for Anomaly Detection in High Energy Physics, 2021.

[72] Anomaly Detection Session, Machine Learning for Jets Workshop. "`https://indico.cern.ch/event/809820/sessions/329216/#20200116`".

[73] Evidence for a new state decaying into two photons in the search for the standard model Higgs boson in pp collisions. Technical report, CERN, Geneva, 2012.

[74] Seyon Sivarijah. Machine Learning Based Simulation of Particle Physics Detectors. Master's thesis, University of Cambridge, 2017.

[75] Trong Huy Phan and Kazuma Yamamoto. Resolving Class Imbalance in Object Detection with Weighted Cross Entropy Losses. *ArXiv*, abs/2006.01413, 2020.

[76] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal Loss for Dense Object Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327, 2020.

[77] Lucas Theis, W. Shi, A. Cunningham, and Ferenc Huszár. Lossy Image Compression with Compressive Autoencoders. *ArXiv*, abs/1703.00395, 2017.

[78] Dalila Salamani, Stefan Gadatsch, Tobias Golling, Graeme Andrew Stewart, Aishik Ghosh, David Rousseau, Ahmed Hasib, and Jana Schaarschmidt. Deep Generative Models for Fast Shower Simulation in ATLAS. In *2018 IEEE 14th International Conference on e-Science (e-Science)*, pages 348–348, 2018.

[79] Diederik P. Kingma and M. Welling. Auto-Encoding Variational Bayes. *CoRR*, abs/1312.6114, 2014.

[80] Eric M. Metodiev, Benjamin Nachman, and Jesse Thaler. Classification without labels: learning from mixed samples in high energy physics. *Journal of High Energy Physics*, 2017(10), Oct 2017.

[81] ATLAS Collaboration. Dijet Resonance Search with Weak Supervision Using $\sqrt{s} = 13$ TeV *pp* Collisions in the ATLAS Detector. *Phys. Rev. Lett.*, 125:131801, Sep 2020.

[82] Jack H. Collins, Kiel Howe, and Benjamin Nachman. Extending the search for new resonances with machine learning. *Physical Review D*, 99(1), Jan 2019.

[83] O. Amram and C. Suarez. Tag N' Train: a technique to train improved classifiers on unlabeled data. *Journal of High Energy Physics*, 2021:1–21, 2020.

[84] Si Zhang, Dawei Zhou, Mehmet Yigit Yildirim, Scott Alcorn, Jingrui He, Hasan Davulcu, and Hanghang Tong. HiDDen: Hierarchical dense subgraph detection with application to financial fraud detection. Proceedings of the 17th SIAM International Conference on Data Mining, SDM 2017, pages 570–578. Society for Industrial and Applied Mathematics Publications, 2017.

[85] Yaguang Li, Rose Yu, C. Shahabi, and Y. Liu. Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting. *arXiv: Learning*, 2018.

[86] Huilin Qu and Loukas Gouskos. Jet tagging via Particle Clouds. *Physical Review D*, 101(5), Mar 2020.

[87] Angel X. et al. Chang. ShapeNet: An Information-Rich 3D Model Repository. Technical Report arXiv:1512.03012 [cs.GR], Stanford University — Princeton University — Toyota Technological Institute at Chicago, 2015.

[88] Murat Abdughani, Jie Ren, Lei Wu, and Jin Min Yang. Probing stop pair production at the LHC with graph neural networks. *Journal of High Energy Physics*, 2019(8), Aug 2019.

[89] Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, and Justin M. Solomon. Dynamic Graph CNN for Learning on Point Clouds. *ACM Transactions on Graphics (TOG)*, 2019.

[90] C. Qi, Hao Su, Kaichun Mo, and L. Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 77–85, 2017.

[91] Anja Butter, Gregor Kasieczka, Tilman Plehn, and Michael Russell. Deep-learned Top Tagging with a Lorentz Layer. *SciPost Physics*, 5(3), Sep 2018.

[92] A. Andreassen, B. Nachman, and D. Shih. Simulation assisted likelihood-free anomaly detection. *Physical Review D*, 101, 2020.

[93] Martín Abadi et al. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems, 2015. Software available from tensorflow.org.

[94] François Chollet et al. Keras. `https://github.com/fchollet/keras`, 2015.

[95] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *CoRR*, abs/1412.6980, 2015.

[96] Wang L Lai A Ji X Zhai X Li G Suchard MA Tian J Zhou J Veit M Su S. Sun J, He WT. COVID-19: Epidemiology, Evolution, and Cross-Disciplinary Perspectives. *Trends Mol Med.*, 26(5):483–495, 2020.

[97] Shabir Ahmad Lone and Aijaz Ahmad. Covid-19 pandemic – an african perspective. *Emerging Microbes & Infections*, 9(1):1300–1308, 2020. PMID: 32458760.

[98] COVID-19 risk adjusted strategy - SA Coronavirus. `https://sacoronavirus.co.za/covid-19-risk-adjusted-strategy`, accesed on 27 April 2020, 2020.

[99] President Cyril Ramaphosa: South Africa's response to Coronavirus COVID-19 pandemic — South African Government. https://www.gov.za/speeches/president-cyril-ramaphosa-south-africas-response-coronavirus-covid-19, accessed on 26 April 2020, 2020.

[100] Joshua Choma, Fabio Correa, Salah-Eddine Dahbi, Barry Dwolatzky, Leslie Dwolatzky, Kentaro Hayasi, Benjamin Lieberman, Caroline Maslo, Bruce Mellado, Kgomotso Monnakgotla, et al. Worldwide Effectiveness of Various Non-Pharmaceutical Intervention Control Strategies on the Global COVID-19 Pandemic: A Linearised Control Model. *medRxiv*, 2020.

[101] Weston C. Roda, Marie B. Varughese, Donglin Han, and Michael Y. Li. Why is it difficult to accurately predict the covid-19 epidemic? *Infectious Disease Modelling*, 5:271–281, 2020.

[102] C. Nowzari, V. M. Preciado, and G. J. Pappas. Analysis and Control of Epidemics: A Survey of Spreading Processes on Complex Networks. *IEEE Control Systems Magazine*, 36:26–46, 2016.

[103] Kasraee M Jahangiri-Rad M. Shariati M, Mesgari T. Spatiotemporal analysis and hotspots detection of COVID-19 using geographic information system (March and April, 2020). 18(2):1–9, 2020.

[104] Mai T Bei C Qin J Pang W Zhang Z. Mo C, Tan D. An analysis of spatiotemporal pattern for COIVD-19 in China based on space-time cube. 92(9):1587–1595, 2020.

[105] Bing Niu, Ruirui Liang, Shuwen Zhang, Hui Zhang, Xiaosheng Qu, Qiang Su, Linfeng Zheng, and Qin Chen. Epidemic analysis of covid-19 in italy based on spatiotemporal geographic information and google trends. *Transboundary and Emerging Diseases*, n/a(n/a).

[106] Purwanto Purwanto, Sugeng Utaya, Budi Handoyo, Syamsul Bachri, Ike Sari Astuti, Kresno Sastro Bangun Utomo, and Yulius Eka Aldianto. Spatiotemporal analysis of covid-19 spread with emerging hotspot analysis and space–time cube models in east java, indonesia. *ISPRS International Journal of Geo-Information*, 10(3), 2021.

[107] Francesca Cito, Laura Amato, Alessandra Di Giuseppe, Maria Luisa Danzetta, Simona Iannetti, Antonio Petrini, Alessio Lorusso, Barbara Bonfini, Alessandra Leone, Romolo Salini, et al. A COVID-19 hotspot area: activities and epidemiological findings. *Microorganisms*, 8(11):1711, 2020.